

DTHRCTL	Bit	Description	R/W	Initial State
		The recommended value for ThrLen is to be the same as the programmed AHB Burst Length (GAHBCFG.HBstLen).		
ISOThrEn	[1]	ISO IN Endpoints Threshold Enable When this bit is set, the core enables thresholding for isochronous IN endpoints	R/W	1'b0
NonISOThrEn	[0]	Non-ISO IN Endpoints Threshold Enable When this bit is set, the core enables thresholding for Non Isochronous IN endpoints.	R/W	1'b0

5.8.7.11 Device IN Endpoint FIFO Empty Interrupt Mask Register (DIEPEMPMSK, R/W, Address = 0xEC00_0834)

This register is used to control the IN endpoint FIFO empty interrupt generation (DIEPINTn.TxfEmp).

- Mask interrupt: 1'b0
- Unmask interrupt: 1'b1

DVBUSPULSE	Bit	Description	R/W	Initial State
Reserved	[31:16]	-	-	16'h0
InEpTxfEmpMsk	[15:0]	IN EP Tx FIFO Empty Interrupt Mask Bits These bits acts as mask bits for DIEPINTn. TxFEmp interrupt One bit per IN Endpoint: Bit 0 for IN EP 0, bit 15 for IN EP 15	R/W	16'h0

5.8.7.12 Device Logical Endpoint-Specific Registers

A logical endpoint is unidirectional: it is either IN or OUT. To represent a bidirectional endpoint, two logical endpoints are required, one for the IN direction and the other for the OUT direction. This is also true for control endpoints. The registers and register fields described in this section may pertain to IN or OUT endpoints, or both, or specific endpoint types are noted.

5.8.7.13 Device Control IN Endpoint 0 Control Register (DIEPCTL0, R/W, Address = 0xEC00_0900)

This section describes the Control IN Endpoint 0 Control register. Nonzero control endpoints use registers for endpoints 1-15.

DIEPCTL0	Bit	Description	R/W	Initial State
EPEna	[31]	Endpoint Enable Indicates that data is ready to be transmitted on the endpoint. The core clears this bit before setting any of the following interrupts on this endpoint. <ul style="list-style-type: none">• Endpoint Disabled• Transfer Completed	R_WS_SC	1'b0
EPDis	[30]	Endpoint Disable The application sets this bit to stop transmitting data on an endpoint, even before the transfer for that endpoint is complete. The application must wait for the Endpoint Disabled interrupt before treating the endpoint as disabled. The core clears this bit before setting the Endpoint Disabled Interrupt. The application must set this bit only if Endpoint Enable is already set for this endpoint.	R_WS_SC	1'b0
Reserved	[29:28]	-	-	2'b0
SetNAK	[27]	Set NAK A write to this bit sets the NAK bit for the endpoint. Using this bit, the application controls the transmission of NAK handshakes on an endpoint. The core also sets this bit for an endpoint after a SETUP packet is received on that endpoint.	W	1'b0
CNAK	[26]	Clear NAK A write to this bit clears the NAK bit for the endpoint.	W	1'b0
TxFNum	[25:22]	TxFIFO Number This value is set to the FIFO number that is assigned to IN Endpoint 0.	R	4'h0
Stall	[21]	STALL Handshake The application sets this bit, and the core clears it, if a SETUP token is received for this endpoint. If a NAK bit, Global Non-Periodic IN NAK, or Global OUT NAK is set along with this bit, the STALL bit takes priority.	R_WS_SC	1'b0
Reserved	[20]	-	-	1'b0
EPType	[19:18]	Endpoint Type Hardcoded to 00 for control	R	2'h0

DIEPCTL0	Bit	Description	R/W	Initial State
NAKsts	[17]	<p>NAK Status Indicates the following:</p> <ul style="list-style-type: none"> • 1'b0: The core is transmitting non-NAK handshakes based on the FIFO status • 1'b1: The core is transmitting NAK handshakes on this endpoint <p>If this bit is set, either by the application or core, the core stops transmitting data, even if there is data available in the TxFIFO. Irrespective of this bit's setting, the core always responds to SETUP data packets with an ACK handshake.</p>	R	1'b0
Reserved	[16]	-	-	1'b0
USBActEP	[15]	<p>USB Active Endpoint This bit is always set to 1, indicating that control endpoint 0 is always active in all configurations and interfaces.</p>	R	1'b1
Reserved	[14:2]	-	-	13'h0
MPS	[1:0]	<p>Maximum Packet Size Applies to IN and OUT endpoints. The application must program this field with the maximum packet size for the current logical endpoint.</p> <ul style="list-style-type: none"> • 2'b00: 64 bytes • 2'b01: 32 bytes • 2'b10: 16 bytes • 2'b11: 8 bytes 	R/W	2'h0

5.8.7.14 Device Control OUT Endpoint 0 Control Register (DOEPCTL0, R/W, Address =0xEC00_0B00)

This section describes the Control OUT Endpoint 0 Control register. Nonzero control endpoints use registers for endpoints 1-15.

DOEPCTL0	Bit	Description	R/W	Initial State
EPEna	[31]	<p>Endpoint Enable When Scatter/Gather DMA mode is enabled, for OUT endpoints this bit indicates that the descriptor structure and data buffer to receive data is setup.</p> <ul style="list-style-type: none"> • When Scatter/Gather DMA mode is disabled—(such as for buffer-pointer based DMA mode)—this bit indicates that the application has allocated the memory to start receiving data from the USB. <p>The core clears this bit before setting any of the following interrupts on this endpoint:</p> <ul style="list-style-type: none"> • SETUP Phase Done • Endpoint Disabled • Transfer Completed <p>Note: In DMA mode, this bit must be set for the core to transfer SETUP data packets into memory.</p>	R_WS_SC	1'b0
EPDis	[30]	Endpoint Disable The application cannot disable control OUT endpoint 0.	R	1'b0
Reserved	[29:28]	-	-	2'b0
SetNAK	[27]	<p>Set NAK A write to this bit sets the NAK bit for the endpoint. Using this bit, the application controls the transmission of NAK handshakes on an endpoint. The core sets this bit on a Transfer Completed interrupt, or after a SETUP is received on the endpoint.</p>	W	1'b0
CNAK	[26]	Clear NAK A write to this bit clears the NAK bit for the endpoint.	W	1'b0
Reserved	[25:22]	-	-	4'h0
Stall	[21]	<p>STALL Handshake The application sets this bit, and the core clears it, if a SETUP token is received for this endpoint. If a NAK bit or Global OUT NAK is set along with this bit, the STALL bit takes priority. Irrespective of this bit's setting, the core always responds to SETUP data packets with an ACK handshake.</p>	R_WS_SC	1'b0
Snp	[20]	<p>Snoop Mode This bit configures the endpoint to Snoop mode. In Snoop mode, the core does not check the correctness of OUT packets before transferring them to application memory.</p>	R/W	1'b0
EPType	[19:18]	Endpoint Type Hardcoded to 2'b00 for control.	R	2'h0

DOEPCTL0	Bit	Description	R/W	Initial State
NAKsts	[17]	<p>NAK Status Indicates the following:</p> <ul style="list-style-type: none"> • 1'b0: The core is transmitting non-NAK handshakes based on the FIFO status • 1'b1: The core is transmitting NAK handshakes on this endpoint <p>If application or the core sets this bit, the core stops receiving data, even if there is space in the RxFIFO to accommodate the incoming packet. Irrespective of this bit's setting, the core always responds to SETUP data packets with an ACK handshake</p>	R	1'b0
Reserved	[16]	-	-	1'b0
USBActEP	[15]	<p>USB Active Endpoint This bit is always set to 1, indicating that a control endpoint 0 is always active in all configurations and interfaces.</p>	R	1'b1
Reserved	[14:2]	-	-	13'h0
MPS	[1:0]	<p>Maximum Packet Size The maximum packet size for control OUT endpoint 0 is the same as what is programmed in control IN Endpoint 0.</p> <ul style="list-style-type: none"> • 2'b00: 64 bytes • 2'b01: 32 bytes • 2'b10: 16 bytes • 2'b11: 8 bytes 	R	2'h0

5.8.7.15 Device Endpoint-n Control Register (DIEPCTLn/DOEPCTLn, R/W, Address = 0xEC00_0900+ n*20h, 0xEC00_0B00+ n*20h)

Endpoint_number: 1 ≤ n ≤ 15

The application uses this register to control the behavior of each logical endpoint other than endpoint 0.

DIEPCTLn/ DOEPCTLn	Bit	Description	R/W	Initial State
EPEna	[31]	<p>Endpoint Enable Applies to IN and OUT endpoints.</p> <ul style="list-style-type: none"> When Scatter/Gather DMA mode is enabled, For IN endpoints this bit indicates that the descriptor structure and data buffer with data ready to transmit is setup. For OUT endpoint it indicates that the descriptor structure and data buffer to receive data is setup. When Scatter/Gather DMA mode is enabled—such as for buffer-pointer based DMA mode: <ul style="list-style-type: none"> - For IN endpoints, this bit indicates that data is ready to be transmitted on the endpoint. - For OUT endpoints, this bit indicates that the application has allocated the memory to start receiving data from the USB. - The core clears this bit before setting any of the following interrupts on this endpoint: • SETUP Phase Done • Endpoint Disabled • Transfer Completed <p>Note: For control endpoints in DMA mode, this bit must be set to be able to transfer SETUP data packets in memory.</p>	R_WS _SC	1'b0
EPDis	[30]	Endpoint Disable Applies to IN and OUT endpoints. The application sets this bit to stop transmitting/receiving data on an endpoint, even before the transfer for that endpoint is complete. The application must wait for the Endpoint Disabled interrupt before treating the endpoint as disabled. The core clears this bit before setting the Endpoint Disabled interrupt. The application must set this bit only if Endpoint Enable is already set for this endpoint.	R_WS _SC	1'b0
SetD1PID	[29]	Set DATA1 PID Applies to interrupt/bulk IN and OUT endpoints only. Writing to this field sets the Endpoint Data PID (DPID) field in this register to DATA1. This field is applicable both for Scatter/Gather DMA mode and non- Scatter/Gather DMA mode.	W	1'b0
SetOddFr		Set Odd (micro) frame Applies to isochronous IN and OUT endpoints only. Writing to this field sets the Even/Odd (micro) frame (EO_FrNum) field to odd (micro) frame. This field is not applicable for Scatter/Gather DMA mode.		
SetD0PID	[28]	Set DATA0 PID Applies to interrupt/bulk IN and OUT endpoints only. Writing to this field sets the Endpoint Data PID (DPID) field in this register to	W	1'b0

DIEPCTLn/ DOEPCTLn	Bit	Description	R/W	Initial State
SetEvenFr		<p>DATA0. This field is applicable both for Scatter/Gather DMA mode and non- Scatter/Gather DMA mode.</p> <p>In non-Scatter/Gather DMA mode: Set Even (micro) frame (SetEvenFr)</p> <p>Applies to isochronous IN and OUT endpoints only. Writing to this field sets the Even/Odd (micro) frame (EO_FrNum) field to even (micro) frame. When Scatter/Gather DMA mode is enabled, this field is reserved. The frame number in which to send data is in the transmit descriptor structure. The frame in which to receive data is updated in receive descriptor structure.</p>		
SNAK	[27]	<p>Set NAK</p> <p>Applies to IN and OUT endpoints. A write to this bit sets the NAK bit for the endpoint. Using this bit, the application can control the transmission of NAK handshakes on an endpoint. The core can also set this bit for OUT endpoints on a Transfer Completed interrupt, or after a SETUP is received on the endpoint.</p>	W	1'b0
CNAK	[26]	<p>Clear NAK</p> <p>Applies to IN and OUT endpoints. A write to this bit clears the NAK bit for the endpoint.</p>	W	1'b0
TxFNum	[25:22]	<p>TxFIFO Number</p> <p>These bits specify the FIFO number associated with this endpoint. Each active IN endpoint must be programmed to a separate FIFO number. This field is valid only for IN endpoints.</p>	R/W	4'h0
Stall	[21]	<p>STALL Handshake</p> <p>Applies to non-control, non-isochronous IN and OUT endpoints only. The application sets this bit to stall all tokens from the USB host to this endpoint. If a NAK bit, Global Non-periodic IN NAK, or Global OUT NAK is set along with this bit, the STALL bit takes priority. Only the application can clear this bit, never the core.</p> <p>Applies to control endpoints only</p> <p>The application can only set this bit, and the core clears it, when a SETUP token is received for this endpoint. If a NAK bit, Global Non-periodic IN NAK, or Global OUT NAK is set along with this bit, the STALL bit takes priority. Irrespective of this bit's setting, the core always responds to SETUP data packets with an ACK handshake.</p>	R/W R_WS_SC	1'b0
Snp	[20]	<p>Snoop Mode</p> <p>Applies to OUT endpoints only.</p> <p>This bit configures the endpoint to Snoop mode. In Snoop mode, the core does not check the correctness of OUT packets before transferring them to application memory.</p>	R/W	1'b0
EPType	[19:18]	<p>Endpoint Type</p> <p>Applies to IN and OUT endpoints.</p> <p>This is the transfer type supported by this logical endpoint.</p> <ul style="list-style-type: none"> • 2'b00: Control 	R	2'h0

DIEPCTLn/ DOEPCTLn	Bit	Description	R/W	Initial State
		<ul style="list-style-type: none"> • 2'b01: Isochronous • 2'b10: Bulk • 2'b11: Interrupt 		
NAKsts	[17]	<p>NAK Status Applies to IN and OUT endpoints. Indicates the following:</p> <ul style="list-style-type: none"> • 1'b0: The core is transmitting non-NAK handshakes based on the FIFO status. • 1'b1: The core is transmitting NAK handshakes on this endpoint. When either the application or the core sets this bit: <ul style="list-style-type: none"> • The core stops receiving any data on an OUT endpoint, even if there is space in the RxFIFO to accommodate the incoming packet. • For non-isochronous IN endpoints: The core stops transmitting any data on an IN endpoint, even if there data is available in the TxFIFO. • For isochronous IN endpoints: The core sends out a zero-length data packet, even if there data is available in the TxFIFO. <p>Irrespective of this bit's setting, the core always responds to SETUP data packets with an ACK handshake.</p>	R	1'b0
DPID	[16]	<p>Endpoint Data PID Applies to interrupt/bulk IN and OUT endpoints only. Contains the PID of the packet to be received or transmitted on this endpoint. The application must program the PID of the first packet to be received or transmitted on this endpoint, after the endpoint is activated. The applications use the SetD1PID and SetD0PID fields of this register to program either DATA0 or DATA1 PID.</p> <ul style="list-style-type: none"> • 1'b0: DATA0 • 1'b1: DATA1 This field is applicable both for Scatter/Gather DMA mode and non- Scatter/Gather DMA mode. 	R	1'b0
EO_FrNum		<p>Even/ Odd (Micro) Frame In non-Scatter/Gather DMA mode: Applies to isochronous IN and OUT endpoints only. Indicates the (micro) frame number in which the core transmits/receives isochronous data for this endpoint. The application must program the even/odd (micro) frame number in which it intends to transmit/receive isochronous data for this endpoint using the SetEvnFr and SetOddFr fields in this register.</p> <ul style="list-style-type: none"> • 1'b0: Even (micro) frame • 1'b1: Odd (micro) frame When Scatter/Gather DMA mode is enabled, this field is reserved. The frame number in which to send data is provided in the transmit descriptor structure. The frame in which data is received is updated in receive descriptor structure. 		
USBActEP	[15]	<p>USB Active Endpoint Applies to IN and OUT endpoints. Indicates whether this endpoint is active in the current configuration and interface. The core clears this bit for all endpoints after detecting a USB reset. After receiving the SetConfiguration and SetInterface commands, the application</p>	R_W_SC	1'b0

DIEPCTLn/ DOEPCTLn	Bit	Description	R/W	Initial State
		must program endpoint registers accordingly and set this bit.		
Reserved	[14:11]	-	R/W	4'h0
MPS	[10:0]	Maximum Packet Size Applies to IN and OUT endpoints. The application must program this field with the maximum packet size for the current logical endpoint. This value is in bytes.	R/W	11'h0

5.8.7.16 Device Endpoint-n Interrupt Register (DIEPINTn/DOEPINTn, R/W, Address = 0xEC00_0908 +n*20h, 0xEC00_0B08 +n*20h)

Endpoint_number: $0 \leq n \leq 15$

This register indicates the status of an endpoint with respect to USB- and AHB-related events. The application must read this register if the OUT Endpoints Interrupt bit or IN Endpoints Interrupt bit of the Core Interrupt register is set. Before the application reads this register, it must first read the Device All Endpoints Interrupt (DAINT) register to get the exact endpoint number for the Device Endpoint-n Interrupt register. The application must clear the appropriate bit in this register to clear the corresponding bits in the DAINT and GINTSTS registers.

DIEPINTn/ DOEPINTn	Bit	Description	R/W	Initial State
EPEna	[31:15]	Reserved	-	17'h0
NYETIntrpt	[14]	NYET interrupt (NYETIntrpt) The core generates this interrupt when a NYET response is transmitted for a non isochronous OUT endpoint.	R_SS _WC	1'b0
NAKIntrpt	[13]	NAK interrupt (NAKIntrpt) The core generates this interrupt when a NAK is transmitted or received by the device. In case of isochronous IN endpoints the interrupt gets generated when a zero length packet is transmitted due to un-availability of data in the TXFifo.	R_SS _WC	1'b0
BbleErrIntrpt	[12]	BbleErr (Babble Error) interrupt (BbleErrIntrpt) The core generates this interrupt when babble is received for the endpoint.	R_SS _WC	1'b0
Packet Dropped Status	[11]	PktDrpSts (Packet Dropped Status) This bit indicates to the application that an ISOC OUT packet has been dropped. This bit does not have an associated mask bit and does not generate an interrupt. Dependency: This bit is valid in non Scatter/Gather DMA mode when periodic transfer interrupt feature is selected.	R_SS _WC	1'b0
Reserved	[10]	-	-	-
BNAIntr	[9]	Buffer Not Available Interrupt This bit is valid only when Scatter/Gather DMA mode is enabled. The core generates this interrupt when the descriptor accessed is not ready for the Core to process, such as Host busy or DMA done	R_SS _WC	
TxfifoUndrn	[8]	Fifo Underrun Applies to IN endpoints Only This bit is valid only when thresholding is enabled. The core generates this interrupt when it detects a transmit FIFO underrun condition for this endpoint.	R_SS _WC	1'b0
OutPktErr		OUT Packet Error Applies to OUT endpoints Only This interrupt is valid only when thresholding is enabled. This interrupt is asserted when the core detects an overflow or a CRC error for non-Isochronous OUT packet.		

DIEPINTn/ DOEPINTn	Bit	Description	R/W	Initial State
TxFEmp	[7]	<p>Transmit FIFO Empty</p> <p>This bit is valid only for IN Endpoints This interrupt is asserted when the TxFIFO for this endpoint is either half or completely empty.</p> <p>The half or completely empty status is determined by the TxFIFO Empty Level bit in the Core AHB Configuration register (GAHBCFG.NPTxFEmpLvl)).</p>		
INEPNakEff	[6]	<p>IN Endpoint NAK Effective</p> <p>Applies to periodic IN endpoints only.</p> <p>This bit can be cleared when the application clears the IN endpoint NAK by writing to DIEPCTLn.CNAK. This interrupt indicates that the core has sampled the NAK bit set (either by the application or by the core). The interrupt indicates that the IN endpoint NAK bit set by the application has taken effect in the core. This interrupt does not guarantee that a NAK handshake is sent on the USB. A STALL bit takes priority over a NAK bit.</p>	R	1'b0
Back2Back SETup		<p>Back-to-Back SETUP Packets Receive</p> <p>Applies to Control OUT endpoints only.</p> <p>This bit indicates that the core has received more than three back-to-back SETUP packets for this particular endpoint.</p> <p>For information about handling this interrupt,</p>	R/W	
INTknEPMis	[5]	<p>IN Token Received with EP Mismatch</p> <p>Applies to non-periodic IN endpoints only.</p> <p>Indicates that the data in the top of the non-periodic TxFIFO belongs to an endpoint other than the one for which the IN token was received. This interrupt is asserted on the endpoint for which the IN token was received.</p>	R_SS _WC	1'b0
StsPhseRcvd		<p>Status Phase Received For Control Write</p> <p>This interrupt is valid only for Control OUT endpoints and only in Scatter Gather DMA mode.</p> <p>This interrupt is generated only after the core has transferred all the data that the host has sent during the data phase of a control write transfer, to the system memory buffer. The interrupt indicates to the application that the host has switched from data phase to the status phase of a Control Write transfer. The application can use this interrupt to ACK or STALL the Status phase, after it has decoded the data phase. This is applicable only in case of Scatter Gather DMA mode.</p>		
INTknTXFEmp	[4]	<p>IN Token Received When TxFIFO is Empty</p> <p>Applies to non-periodic IN endpoints only. Indicates that an IN token was received when the associated TxFIFO (periodic/non-periodic) was empty. This interrupt is asserted on the endpoint for which the IN token was received.</p>	R_SS _WC	1'b0

DIEPINTn/ DOEPINTn	Bit	Description	R/W	Initial State
OUTTknEPdis		OUT Token Received When Endpoint Disabled Applies only to control OUT endpoints. Indicates that an OUT token was received when the endpoint was not yet enabled. This interrupt is asserted on the endpoint for which the OUT token was received.		
TimeOUT	[3]	Timeout Condition • In dedicated FIFO mode, applies only to Control IN endpoints. • In Scatter/Gather DMA mode, the TimeOUT interrupt is not asserted. Indicates that the core has detected a timeout condition on the USB for the last IN token on this endpoint.	R_SS _WC	1'b0
setUp		SETUP Phase Done Applies to control OUT endpoints only. Indicates that the SETUP phase for the control endpoint is complete and no more back-to-back SETUP packets were received for the current control transfer. On this interrupt, the application can decode the received SETUP data packet.	-	
AHBErr	[2]	AHB Error Applies to IN and OUT endpoints. This is generated only in Internal DMA mode if there is an AHB error during an AHB read/write. The application reads the corresponding endpoint DMA address register to get the error address.	R_SS _WC	1'b0
EPDisbld	[1]	Endpoint Disabled Interrupt Applies to IN and OUT endpoints. This bit indicates that the endpoint is disabled per the application's request.	R_SS _WC	1'b0
XferCompl	[0]	Transfer Completed Interrupt (XferCompl) Applies to IN and OUT endpoints. • When Scatter/Gather DMA mode is enabled • For IN endpoint this field indicates that the requested data from the descriptor is moved from external system memory to internal FIFO. • For OUT endpoint this field indicates that the requested data from the internal FIFO is moved to external system memory. This interrupt is generated only when the corresponding endpoint descriptor is closed, and the IOC bit for the corresponding descriptor is set. • When Scatter/Gather DMA mode is disabled, this field indicates that the programmed transfer is complete on the AHB as well as on the USB, for this endpoint.	R_SS _WC	1'b0

5.8.7.17 Device Endpoint 0 Transfer Size Register (DIEPTSIZ0, R/W, Address = 0xEC00_0910)

The application must modify this register before enabling endpoint 0. Once endpoint 0 is enabled using Endpoint Enable bit of the Device Control Endpoint 0 Control registers (DIEPCTL0.EPEna/DOEPCTL0.EPEna), the core modifies this register. The application can only read this register once the core has cleared the Endpoint Enable bit.

Nonzero endpoints use the registers for endpoints 1-15.

When Scatter/Gather DMA mode is enabled, this register must not be programmed by the application. If the application reads this register when Scatter/Gather DMA mode is enabled, the core returns all zeros.

DIEPTSIZ0	Bit	Description	R/W	Initial State
Reserved	[31:21]	-	-	11'h0
PktCnt	[20:19]	Packet Count Indicates the total number of USB packets that constitute the Transfer Size amount of data for endpoint 0. This field is decremented every time a packet is read from the TxFIFO.	R/W	2'b0
Reserved	[18:7]	-	-	12'h0
XferSize	[6:0]	Transfer Size Indicates the transfer size in bytes for endpoint 0. The core interrupts the application only after it has exhausted the transfer size amount of data. The transfer size can be set to the maximum packet size of the endpoint, to be interrupted at the end of each packet. The core decrements this field every time a packet from the external memory is written to the TxFIFO.	R/W	7'h0

5.8.7.18 Device OUT Endpoint 0 Transfer Size Register (DOEPTSIZ0, R/W, Address = 0xEC00_0B10)

DOEPTSIZ0	Bit	Description	R/W	Initial State
Reserved	[31]	-	-	1'b0
SUPCnt	[30:29]	SETUP Packet Count This field specifies the number of back-to-back SETUP data packets the endpoint can receive. <ul style="list-style-type: none">• 2'b01: 1 packet• 2'b10: 2 packets• 2'b11: 3 packets	R/W	2'h0
Reserved	[28:21]	-	-	9'h0
PktCnt	[20:19]	Packet Count This field is decremented to zero after a packet is written into the RxFIFO.	R/W	2'b0
Reserved	[18:7]	-	-	12'h0
XferSize	[6:0]	Transfer Size Indicates the transfer size in bytes for endpoint 0. The core interrupts the application only after it has exhausted the transfer size amount of data. The transfer size can be set to the maximum packet size of the endpoint, to be interrupted at the end of each packet. The core decrements this field every time a packet is read from RxFIFO and written to the external memory.	R/W	7'h0

5.8.7.19 Device Endpoint-n Transfer Size Register (DIEPTSI_n/DOEPTSI_n, R/W, Address = 0xEC00_0910 +n*20h, 0xEC00_0B10 +n*20h)

Endpoint_number: 1 ≤ n ≤ 15

The application must modify this register before enabling the endpoint. Once the endpoint is enabled using Endpoint Enable bit of the Device Endpoint-n Control registers (DIEPCTL_n.EPEna/DOEPCTL_n.EPEna), the core modifies this register. The application can only read this register once the core has cleared the Endpoint Enable bit. This register is used only for endpoints other than Endpoint 0.

When Scatter/Gather DMA mode is enabled, this register must not be programmed by the application. If the application reads this register when Scatter/Gather DMA mode is enabled, the core returns all zeros

DIEPTSI _n /DOEPTSI _n	Bit	Description	R/W	Initial State
Reserved	[31]	-	-	1'b0
MC	[30:29]	Multi Count Applies to IN endpoints only. For periodic IN endpoints, this field indicates the number of packets that must be transmitted per microframe on the USB. The core uses this field to calculate the data PID for isochronous IN endpoints. <ul style="list-style-type: none"> • 2'b01: 1 packet • 2'b10: 2 packets • 2'b11: 3 packets 	R/W	2'b0
		For non-periodic IN endpoints, this field is valid only in Internal DMA mode. It specifies the number of packets the core must fetch for an IN endpoint before it switches to the endpoint pointed to by the Next Endpoint field of the Device Endpoint-n Control register (DIEPCTL _n .NextEp).		
		Received Data PID Applies to isochronous OUT endpoints only. This is the data PID received in the last packet for this endpoint. <ul style="list-style-type: none"> • 2'b00: DATA0 • 2'b01: DATA1 • 2'b10: DATA2 • 2'b11: MDATA 		
SUPCnt		SETUP Packet Count Applies to control OUT Endpoints only. This field specifies the number of back-to-back SETUP data packets the endpoint can receive. <ul style="list-style-type: none"> • 2'b01: 1 packet • 2'b10: 2 packets • 2'b11: 3 packets 	R/W	
		Packet Count Indicates the total number of USB packets that constitute the Transfer Size amount of data for this endpoint. <ul style="list-style-type: none"> • IN Endpoints: This field is decremented every time a packet (maximum size or short packet) is read from the TxFIFO. • OUT Endpoints: This field is decremented every time a packet (maximum size or short packet) is written to the RxFIFO. 		
PktCnt	[28:19]		R/W	10'h0

DIEPTSIzn/ DOEPTSIzn	Bit	Description	R/W	Initial State
XferSize	[18:0]	<p>Transfer Size</p> <p>This field contains the transfer size in bytes for the current endpoint. The core only interrupts the application after it has exhausted the transfer size amount of data. The transfer size can be set to the maximum packet size of the endpoint, to be interrupted at the end of each packet.</p> <ul style="list-style-type: none"> • IN Endpoints: The core decrements this field every time a packet from the external memory is written to the TxFIFO. • OUT Endpoints: The core decrements this field every time a packet is read from the RxFIFO and written to the external memory. 	R/W	19'h0

5.8.7.20 Device Endpoint-n DMA Address (DIEPDMA_n/DOEPDMA_n, R/W, Address = 0xEC00_0914 +n*20h, 0xEC00_0B14 +n*20h)

Endpoint_number : 0 ≤ n ≤ 15

The starting DMA address must be DWORD-aligned.

DIEPDMA _n / DOEPDMA _n	Bit	Description	R/W	Initial State
DMAAddr	[31:0]	<p>DMA Address</p> <p>Holds the start address of the external memory for storing or fetching endpoint data.</p> <p>Note: For control endpoints, this field stores control OUT data packets as well as SETUP transaction data packets. When more than three SETUP packets are received back-to-back, the SETUP data packet in the memory is overwritten.</p> <p>This register is incremented on every AHB transaction. The application can give only a DWORD-aligned address.</p> <ul style="list-style-type: none"> • When Scatter/Gather DMA mode is not enabled, When Scatter/Gather DMA mode is not enabled, the application programs the start address value in this field. <p>When Scatter/Gather DMA mode is enabled, this field indicates the base pointer for the descriptor list.</p>	R/W	32'h0

5.8.7.21 Device IN Endpoint Transmit FIFO Status (DTXFSTS_n, R/W, Address = 0xEC00_0918 +n*20h)

Endpoint_number: 0 ≤ n ≤ 15

This read-only register contains the free space information for the Device IN endpoint TxFIFO

DTXFSTS _n	Bit	Description	R/W	Initial State
Reserved	[31:16]	-		16'h0
INEPTxFSpca vail	[15:0]	IN Endpoint TxFIFO Space Avail Indicates the amount of free space available in the Endpoint TxFIFO. Values are in terms of 32-bit words. <ul style="list-style-type: none"> • 16'h0: Endpoint TxFIFO is full • 16'h1: 1 word available • 16'h2: 2 words available • 16'hn: n words available (where 0 ≤ n ≤ 32,768) • 16'h8000: 32,768 words available • Others: Reserved 	R	16'h100

5.8.7.22 Device Endpoint-n DMA Buffer Address Register (DIEPDMA_{Bn}/DOEPDMA_{Bn}, R/W, Address = 0xEC00_091C +n*20h, 0xEC00_B1C +n*20h)

Endpoint_number: 0 ≤ n ≤ 15

These fields are present only in case of Scatter/Gather DMA.

DIEPDMA _{Bn} /D OEPDMA _{Bn}	Bit	Description	R/W	Initial State
DMABufferAddr	[31:0]	DMA Buffer Address Holds the current buffer address. This register is updated as and when the data transfer for the corresponding endpoint is in progress. This register is present only in Scatter/Gather DMA mode. Otherwise this field is reserved.	R	32'h0

6 MODEM INTERFACE

6.1 OVERVIEW OF MODEM INTERFACE

This chapter defines the interface between the Base-band Modem (like MSM) and the Application Processor to facilitate data-exchange between these two devices. To facilitate data-exchange, S5PV210 include a dual-ported SRAM buffer (on-chip). To access SRAM buffer, the Modem chip uses a typical asynchronous-SRAM interface.

The size of the SRAM buffer is 16 KB. This specification specifies a few pre-defined special addressees for the buffer status and interrupts requests.

Modem chip writes data in the data buffer (Internal dual port SRAM buffer) and request interrupt to AP. When the interrupt is asserted, AP reads data in data buffer, and then clears the interrupt. In the same manner, AP writes data in the data buffer and then asserts interrupt to modem chip to notify.

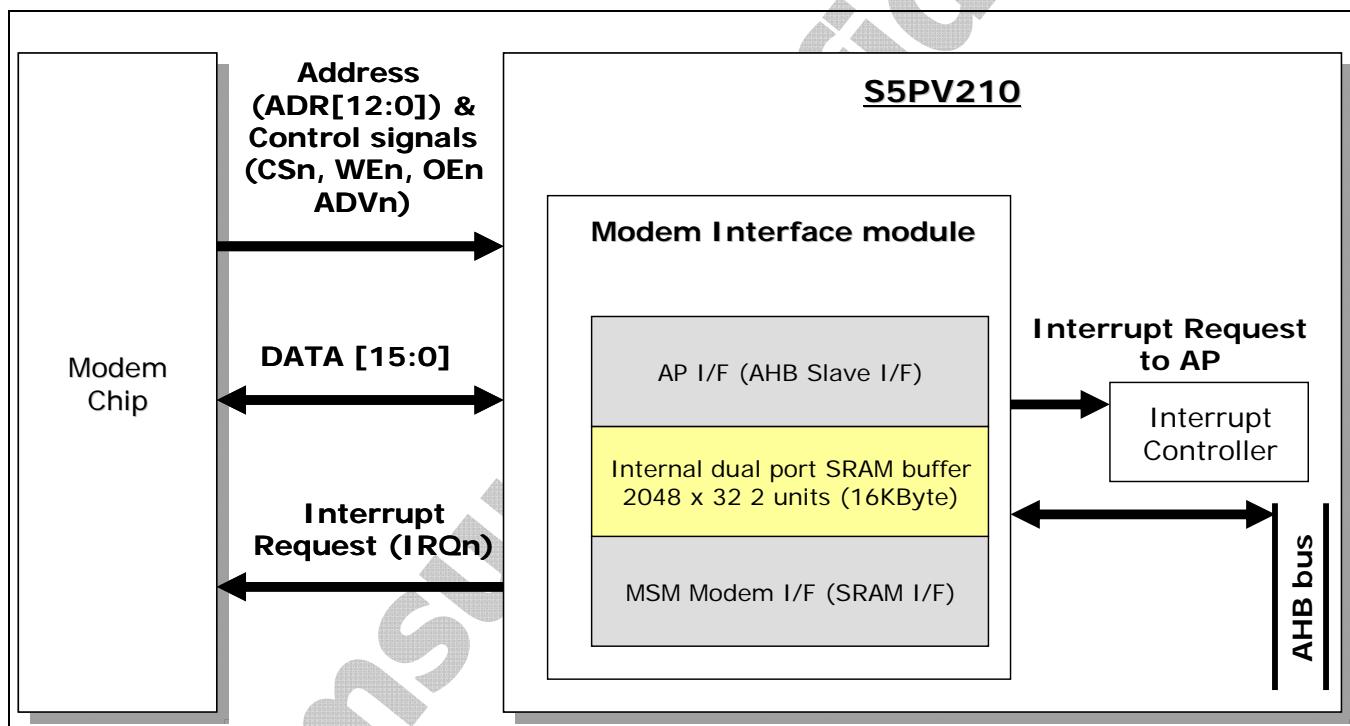


Figure 6-1 Interface with the Modem Chip and the MODEM I/F Block Diagram

6.2 KEY FEATURES OF MODEM INTERFACE

- Asynchronous SRAM interface style interface
- Supports both Standard mode and Address Muxed mode
- Supports 16-bit parallel bus for data transfer
- Supports 16 KB internal dual-port SRAM buffer
- Supports Interrupt request for data exchange
- Programmable interrupt port address
- Supports DMA for data transfer without intervention of CPU

6.3 INTERRUPT PORTS

If the Modem chip or AP accesses the interrupt-port (predefined special addresses) interrupts are requested or cleared. The S5PV210 configures the special address and the default address-map is described in the [Table 6-1](#).

Table 6-1 Interrupt Request and Clear Conditions

Interrupt	An Interrupt is requested, when	Interrupt is cleared, when
To AP	Modem chip writes at least 1 to 0x1FFF through ADR.	AP writes at least 1 to MSMINTCLR register in MODEM IF (2).
To Modem	AP writes 1 to 0xED00_3FFC through internal-chip AHB bus.	Modem chip writes 1 to the bits at 0x1FFE through ADR.

NOTE:

1. There are two address views for MODEMIF, namely, MSM address (ADR) for MODEM chip, and AHB address for S5PV210. AHB address is twice the size of ADR. For example, 0x3FFC at AHB bus is 0x1FFE at ADR. helps you to understand it.
This is default value. To change the value use SFR (INT2AP and INT2MSM).
2. Modem interface block has one Interrupt Clear Registers; MSMINTCLR. Modem interface block generates level type interrupt request and is sustained until the S5PV210 clears the interrupt clear registers by writing any value to the registers.

Modem chip or S5PV210 reads the data that indicates what event occurred, namely, data transfer requested, data transfer done, special command issued, etc. – from interrupt port address. That data format should be defined for communication between the modem chip and S5PV210.

6.3.1 WAKEUP

* S5PV210 MODEM_IF does not support Wakeup Interrupt mode

6.4 ADDRESS MAPPING

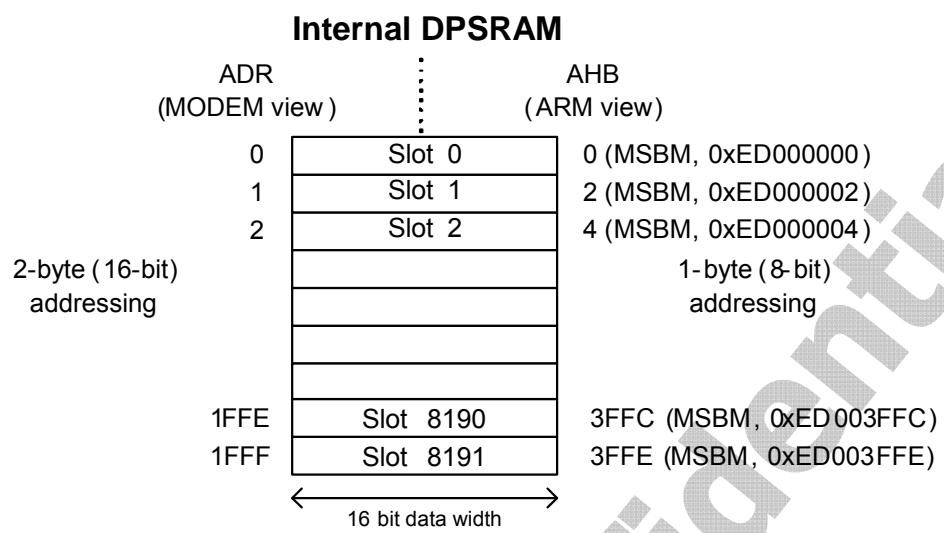


Figure 6-2 MODEM I/F Address Mapping

6.5 TIMING DIAGRAM

6.5.1 STANDARD MODE WRITE, READ TIMING

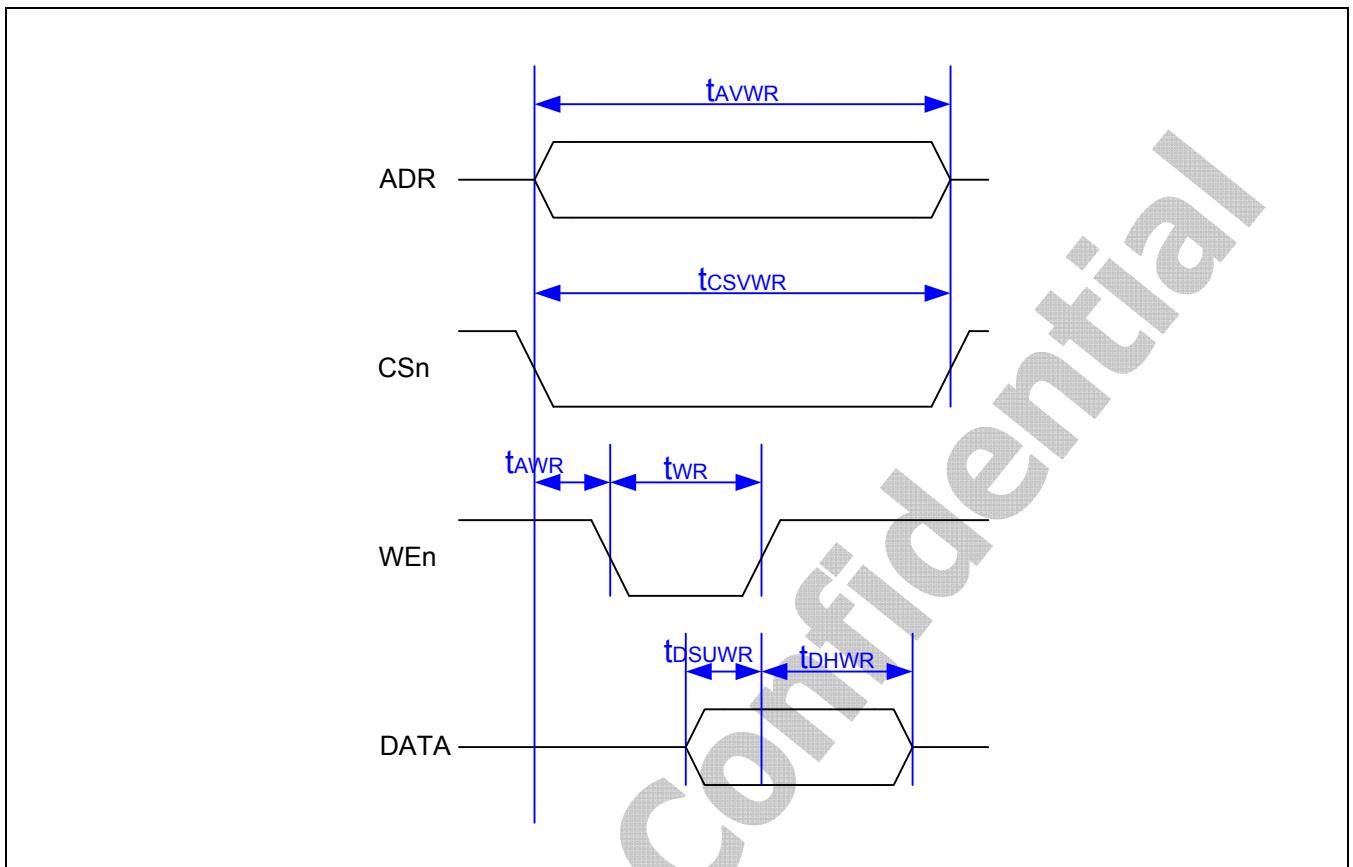


Figure 6-3 Modem Interface Write Timing Diagram (Standard Mode)

Table 6-2 Modem Interface Write Timing (Standard Mode)

Parameter	Description	Min (ns)	Max (ns)	Notes
t_{AVWR}	Address valid to address invalid	16 ns	-	-
t_{CSVWR}	Chip select active	16 ns	-	-
t_{AWR}	Address valid to write active	4 ns	-	-
t_{WR}	Write active	8 ns	-	-
t_{DSUWR}	Write data setup	8 ns	-	-
t_{DHWR}	Write data hold	4 ns	-	-

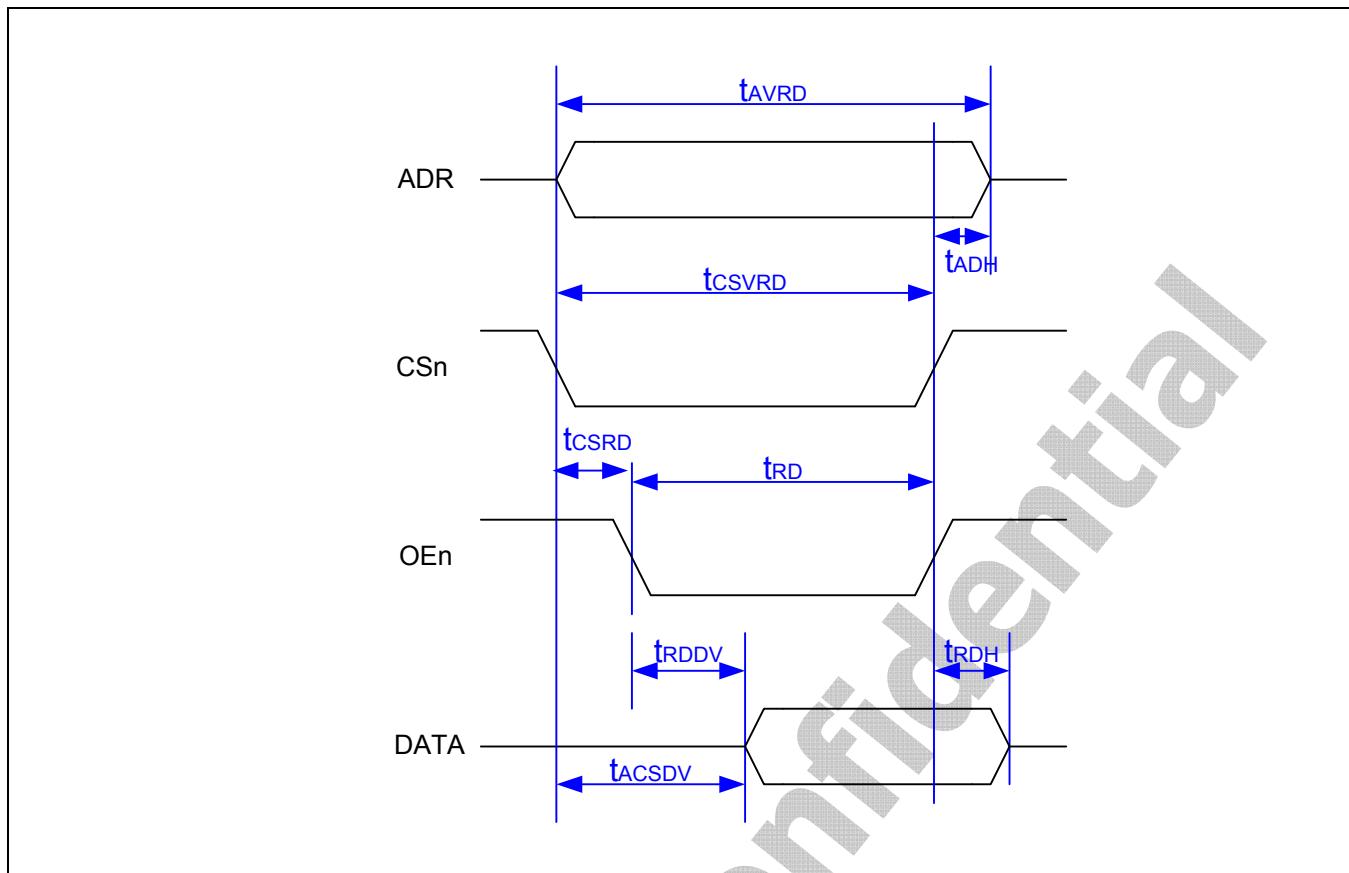


Figure 6-4 Modem Interface Read Timing Diagram (Standard Mode)

Table 6-3 Modem Interface Read Timing (Standard Mode)

Parameter	Description	Min (ns)	Max (ns)	Notes
t_{AVRD}	Address valid to address invalid	50 ns	-	-
t_{ADH}	Address hold	0 ns	-	-
t_{CSVRD}	Chip select active	50 ns	-	-
t_{CSRD}	Chip select active to read active	14 ns	-	-
t_{RD}	Read active	36 ns	-	-
t_{RDDV}	Read active to data valid	-	35 ns	-
t_{RDH}	Read data hold	6 ns	-	-
t_{ACSDV}	Address and chip select active to data valid	-	49 ns	-

NOTE: Output load is 30pF in room temperature (25 Degree)

6.5.2 ADDRESS MUXED MODE WRITE, READ TIMING

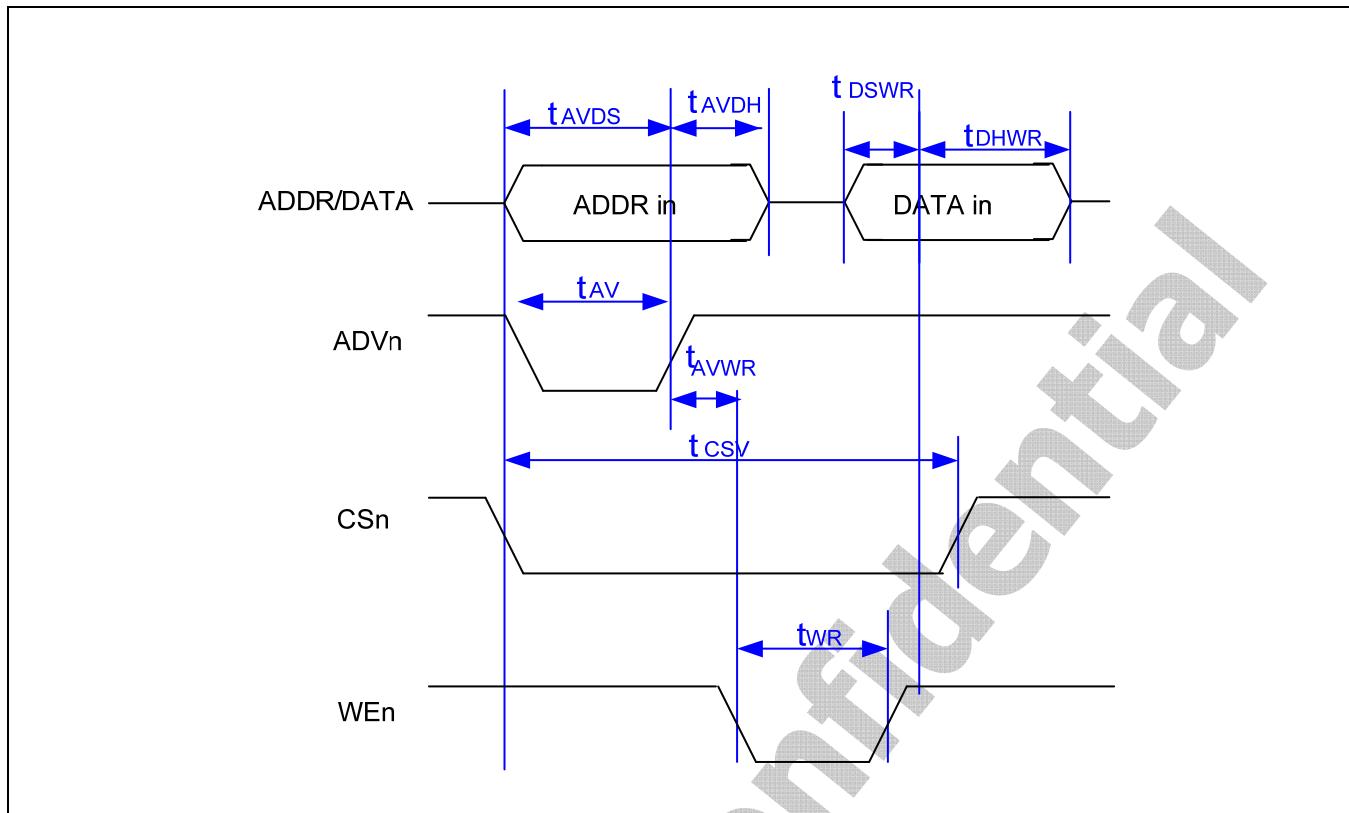


Figure 6-5 Modem Interface Write Timing Diagram (Address Muxed mode)

Table 6-4 Modem Interface Write Timing (Address Muxed mode)

Parameter	Description	Min (ns)	Max (ns)	Notes
t_{AVDS}	Address valid setup	15 ns	-	-
t_{AVDH}	Address valid hold	5 ns	-	-
t_{AV}	Address valid duration	15 ns	-	-
t_{AVWR}	Address valid to write enable	0 ns	-	-
t_{CSV}	Chip select duration	20 ns	-	-
t_{WR}	Write enable duration	5 ns	-	-
t_{DSWR}	Write data setup	8 ns	-	-
t_{DHWR}	Write data hold	4 ns	-	-

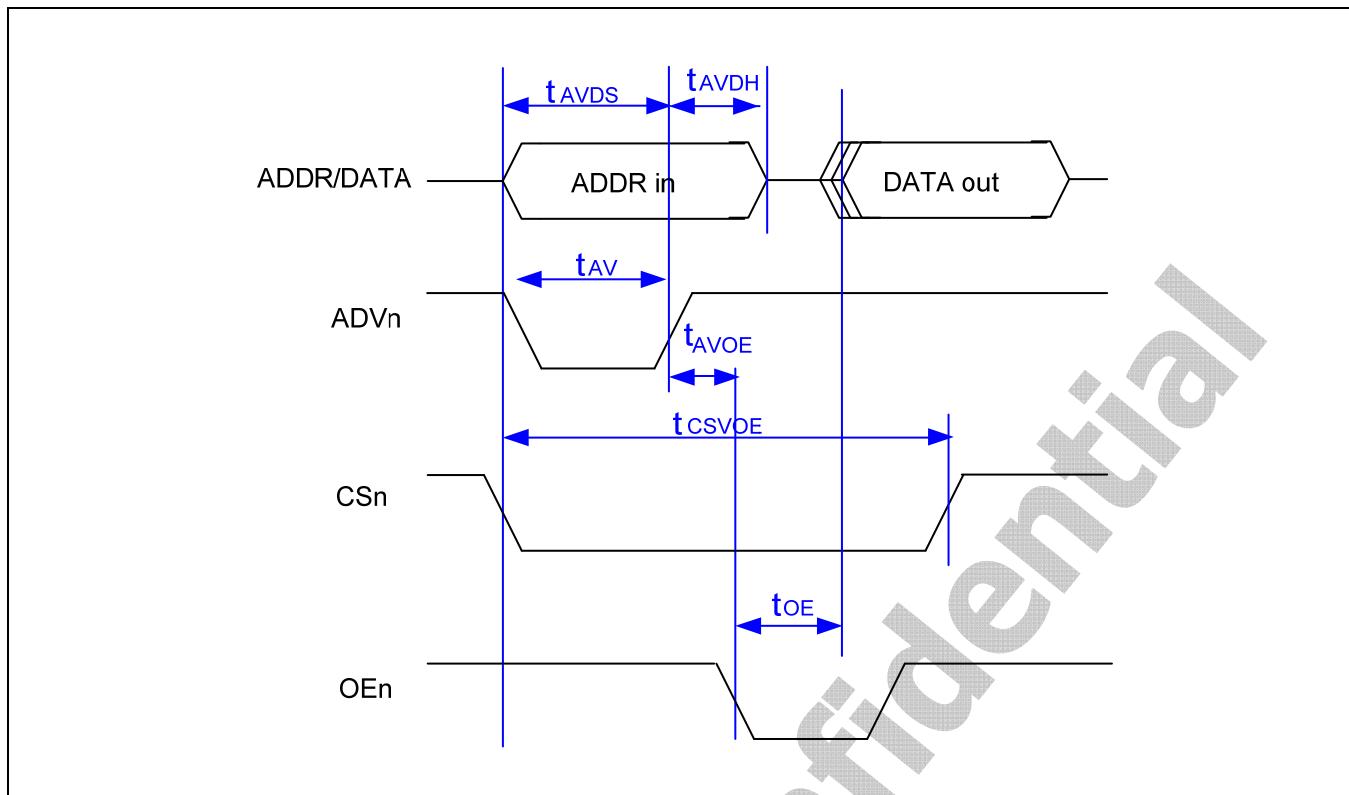


Figure 6-6 Modem Interface Read Timing Diagram (Address Muxed mode)

Table 6-5 Modem Interface Read Timing (Address Muxed mode)

Parameter	Description	Min (ns)	Max (ns)	Notes
t_{AVDS}	Address valid setup	15 ns	-	-
t_{AVDH}	Address valid hold	5 ns	-	-
t_{AV}	Address valid duration	15 ns	-	-
t_{AVOE}	Address valid to read enable	5 ns	-	-
t_{CSVOE}	Chip select duration (Read mode)	45 ns	-	-
t_{OE}	Output enable(Read Active) to data valid	-	35 ns	-

NOTE: Output load is 30pF in room temperature (25 Degree)

6.6 I/O DESCRIPTION

Signal	I/O	Description	Pad	Type
ADR[12:0]	Input	Address from MODEM Chip	XmsmADDR[12:0]	muxed
CSn	Input	Chip Select Signal from MODEM Chip	XmsmCSn	muxed
WE _n	Input	Write Enable Signal from MODEM Chip	XmsmWE _n	muxed
OE _n	Input	Read Enable Signal from MODEM Chip	XmsmR _n	muxed
DATA[15:0]	In/Out	Data from/to MODEM Chip	XmsmDATA[15:0]	muxed
IRQn	Output	Interrupt Request to MODEM Chip	XmsmIRQn	muxed
ADVn	Input	Address Valid from MODEM Chip (Only Address Muxed mode)	XmsmADVn	muxed

NOTE: Type field indicates whether pads are dedicated to the signal or pads are connected to the multiplexed signals.

6.7 SOFTWARE INTERFACE AND REGISTERS

This modem interface provides a generic data-exchange method. This interface does not implement any other complex features except for the interrupt-request/ clear such as automatic FIFO managements, and so on. The software should be responsible for required functionalities for the data exchange between the modem chip and the S5PV210, namely, data exchange protocol, data buffer managements, and so on.

6.8 REGISTER DESCRIPTION

6.8.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
MSBM	0xED00_0000 ~ 0xED00_3FFC	R/W	Specifies the MODEM I/F SRAM Buffer Memory (AP side)	-
INT2AP	0xED00_8000	R/W	Specifies the Interrupt Request to AP Register	0x00003FFE
INT2MSM	0xED00_8004	R/W	Specifies the Interrupt Request to MSM Modem Register	0x00003FFC
MIFCON	0xED00_8008	R/W	Specifies the Modem Interface Control Register	0x00100008
MIFPCON	0xED00_800C	R/W	Specifies the Modem Interface Port Control register	0x00000000
MSMINTCLR	0xED00_8010	W	Specifies the MSM Modem Interface Pending Interrupt Request Clear	-
DMA_TX_ADR	0xED00_8014	R/W	Specifies the DMA TX Request Address Register	0x17FE_13FE
DMA_RX_ADR	0xED00_8018	R/W	Specifies the DMA RX Request Address Register	0x1FFE_1BFE

6.8.1.1 Interrupt Request to AP Register (INT2AP, R/W, Address = 0xED00_8000)

INT2AP	Bit	Description	Initial State
Reserved	[31:14]	Reserved	0
INT2AP_ADR	[13:0]	Modem interface requests the interrupt to S5PV210 if modem chip writes this address. Interrupt controller of S5PV210 and write access to the MSMINTCLR register clears this interrupt.	0x3FFE

6.8.1.2 Interrupt Request to Modem Register (INT2MSM, R/W, Address = 0xED00_8004)

INT2MSM	Bit	Description	Initial State
Reserved	[31:14]	Reserved	0
INT2MSM_ADR	[13:0]	Modem interface (in this module) requests the interrupt to modem chip if AP writes this address and clears the interrupt if modem chip write 1 to the corresponding bit field.	0x3FFC

NOTE: It is recommended that S5PV210 write data with half-word access on the interrupt port because S5PV210 overwrites the data in INT2AP if there are INT2AP and INT2MSM sharing the same word.

6.8.1.3 Modem Interface Control Register (MIFCON, R/W, Address = 0xED00_8008)

MIFCON	Bit	Description	Initial State
Reserved	[31:21]	-	0
Fixed	[20]	Should write as 1	1
DMARXREQEN_1	[19]	Enables MSM Write DMA Request (RX 1) to AP (DMA Controller)	0
DMARXREQEN_0	[18]	Enables MSM Write DMA Request (RX 0) to AP (DMA Controller)	0
DMATXREQEN_1	[17]	Enables MSM Read DMA Request (TX 1) to AP (DMA Controller)	0
DMATXREQEN_0	[16]	Enables MSM Read DMA Request (TX 0) to AP (DMA Controller)	0
Reserved	[15:4]	-	0
INT2MSMEN	[3]	Enables Interrupt to MSM (Modem) : MSM_nIRQ is interrupt signal enable. 0 = Disables 1 = Enables	1
INT2APEN	[2]	Enables MSM (Modem) write interrupt to AP 0 = Disable s 1 = Enables	0
Reserved	[1]	Reserved	0
Fixed	[0]	Fixed to 0	0

6.8.1.4 Modem Interface Port Control Register (MIFPCON, R/W, Address = 0xED00_800C)

MIFPCON	Bit	Description	Initial State
Reserved	[31:7]	-	0
ADM_MODE	[6]	Address Muxed mode selection 0 = Disables 1 = Enables	0
Reserved	[5]	Fixed to 0	0
INT2M_LEVEL	[4]	Fixed to 0	0
Fixed	[3:0]	Fixed to 0	0

6.8.1.5 MSM Interrupt Clear Register (MSMINTCLR, W, Address = 0xED00_8010)

MSMINTCLR	Bit	Description	Initial State
MSMINTCLR	[31:0]	Write access to this register with any data will clear the interrupt pending register of MSM modem interface.	-

NOTE: The interrupt controllers of S5PV210 receive level-triggered type interrupt requests. Therefore, interrupt requests from MSM interface are maintained until the interrupt service routine clears the interrupt pending register by writing any data into this register.

6.8.1.6 DMA Request TX Address Register used at Direct Mode (DMAREQ_TX_ADR, R/W, Address = 0xED00_8014)

INT2AP	Bit	Description	Initial State
Reserved	[31:30]	Reserved	0
DMA_TX_ADR_1	[29:16]	Modem interface requests the DMA to AP (DMA Controller) if modem chip reads this address. Source: DMA_MSM_Req[1]	0x17FE
Reserved	[15:14]	Reserved	0
DMA_TX_ADR_0	[13:0]	Modem interface requests the DMA to AP (DMA Controller) if modem chip reads this address. Source: DMA_MSM_Req[0]	0x13FE

6.8.1.7 DMA Request RX Address Register used at Direct Mode (DMAREQ_RX_ADR, R/W, Address = 0xED00_8018)

INT2AP	Bit	Description	Initial State
Reserved	[31:30]	Reserved	0
DMA_RX_ADR_1	[29:16]	Modem interface requests the DMA to AP (DMA Controller) if modem chip writes this address. Source: DMA_MSM_Req[3]	0x1FFE
Reserved	[15:14]	Reserved	0
DMA_RX_ADR_0	[13:0]	Modem interface requests the DMA to AP (DMA Controller) if modem chip writes this address. Source: DMA_MSM_Req[2]	0x1BFE

7 SD/MMC CONTROLLER

This chapter describes the Secure Digital (SD/ SDIO), MultiMediaCard (MMC), CE-ATA host controller and related registers supported by S5PV210 RISC microprocessor.

7.1 OVERVIEW OF SD/ MMC CONTROLLER

The SD/ MMC host controller is a combo host for Secure Digital card and MultiMediaCard. This host controller is based on SD Association's (SDA) Host Standard Specification.

The SD/ MMC host controller is a interface between system and SD/MMC. The performance of this host is very powerful, as clock rate is 52-MHz and access 8-bit data pin simultaneously.

7.2 KEY FEATURES OF SD/ MMC CONTROLLER

The High-Speed MMC controller supports:

- SD Standard Host Specification Version 2.0 standard
- SD Memory Card Specification Version 2.0 / High Speed MMC Specification Version 4.3 standard
- SDIO Card Specification Version 1.0 standard
- 512 bytes FIFO for data Tx/ Rx
- CPU Interface and DMA data transfer mode
- 1-bit / 4-bit / 8-bit mode switch
- 8-bit 2 channel, or 4-bit 4 channel
- Auto CMD12
- Suspend/ Resume
- Read Wait operation
- Card Interrupt
- CE-ATA mode

7.3 BLOCK DIAGRAM OF SD/ MMC CONTROLLER

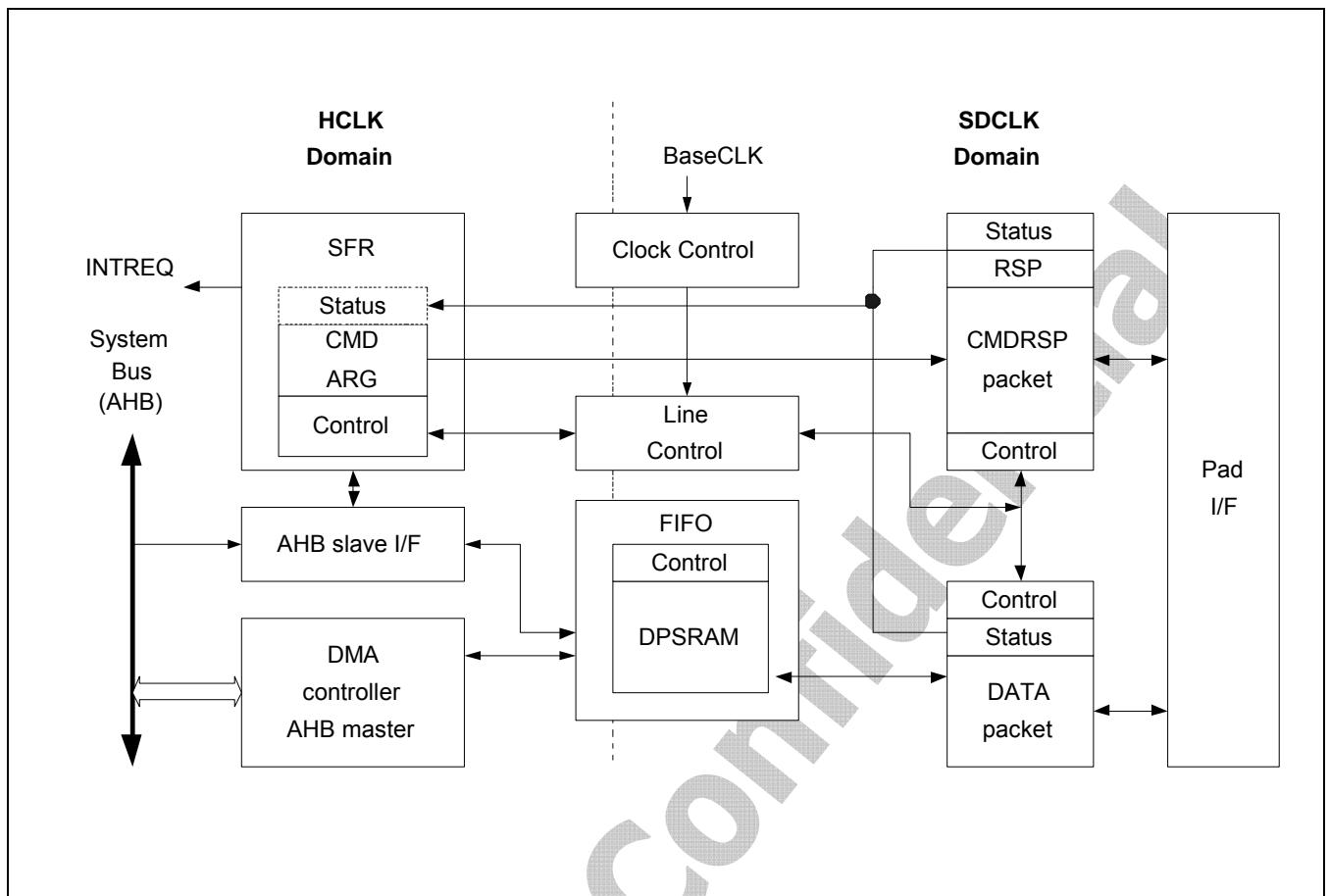


Figure 7-1 SDMMC Clock Domain

7.4 OPERATION SEQUENCE

This section defines basic operation flow chart divided into several sub sequences. "Wait for interrupts" is used in the flow chart. This means the Host Driver waits until specified interrupts are asserted. If already asserted, then follow the next step in the flow chart. Timeout checking is only available when interrupt is not generated, and it is not described in the flow chart.

7.4.1 SD CARD DETECTION SEQUENCE

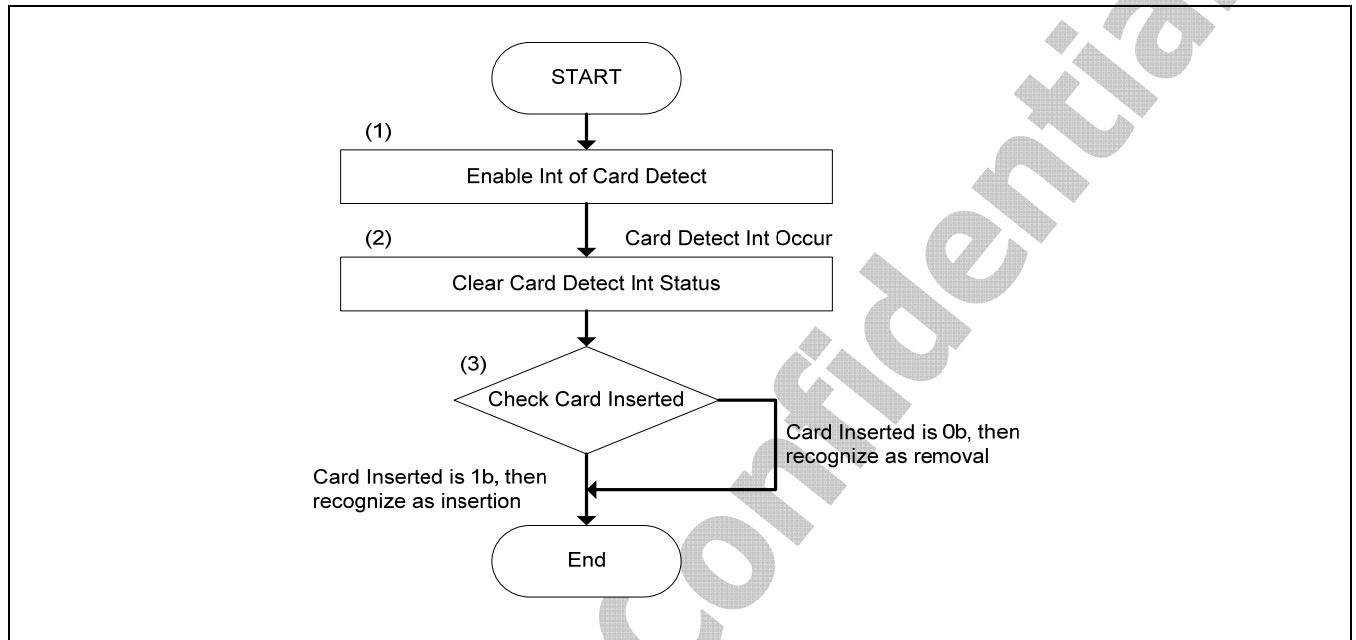


Figure 7-2 SD Card Detect Sequence

The flow chart to detect a SD card is shown in [Figure 7-2](#). The steps to detect SD card:

1. To enable interrupt for card detection, write 1 to the following bits:
 Card Insertion Status Enable (ENSTACARDNS) in the Normal Interrupt Status Enable register
 Card Insertion Signal Enable (ENSIGCARDNS) in the Normal Interrupt Signal Enable register
 Card Removal Status Enable (ENSTACARDREM) in the Normal Interrupt Status Enable register
 Card Removal Signal Enable (ENSIGCARDREM) in the Normal Interrupt Signal Enable register
2. If Host Driver detects the card insertion or removal, it clears the interrupt statuses. If Card Insertion interrupt (STACARDINS) is generated, write 1 to Card Insertion in the Normal Interrupt Status register. If Card Removal interrupt (STACARDREM) is generated, write 1 to Card Removal in the Normal Interrupt Status register.
3. Check Card Inserted in the Present State register. If Card Inserted (INSCARD) is 1, the Host Driver supplies power and clock to the SD card. If Card Inserted is 0, it stops the executing process of the Host Driver.

7.4.2 SD CLOCK SUPPLY SEQUENCE

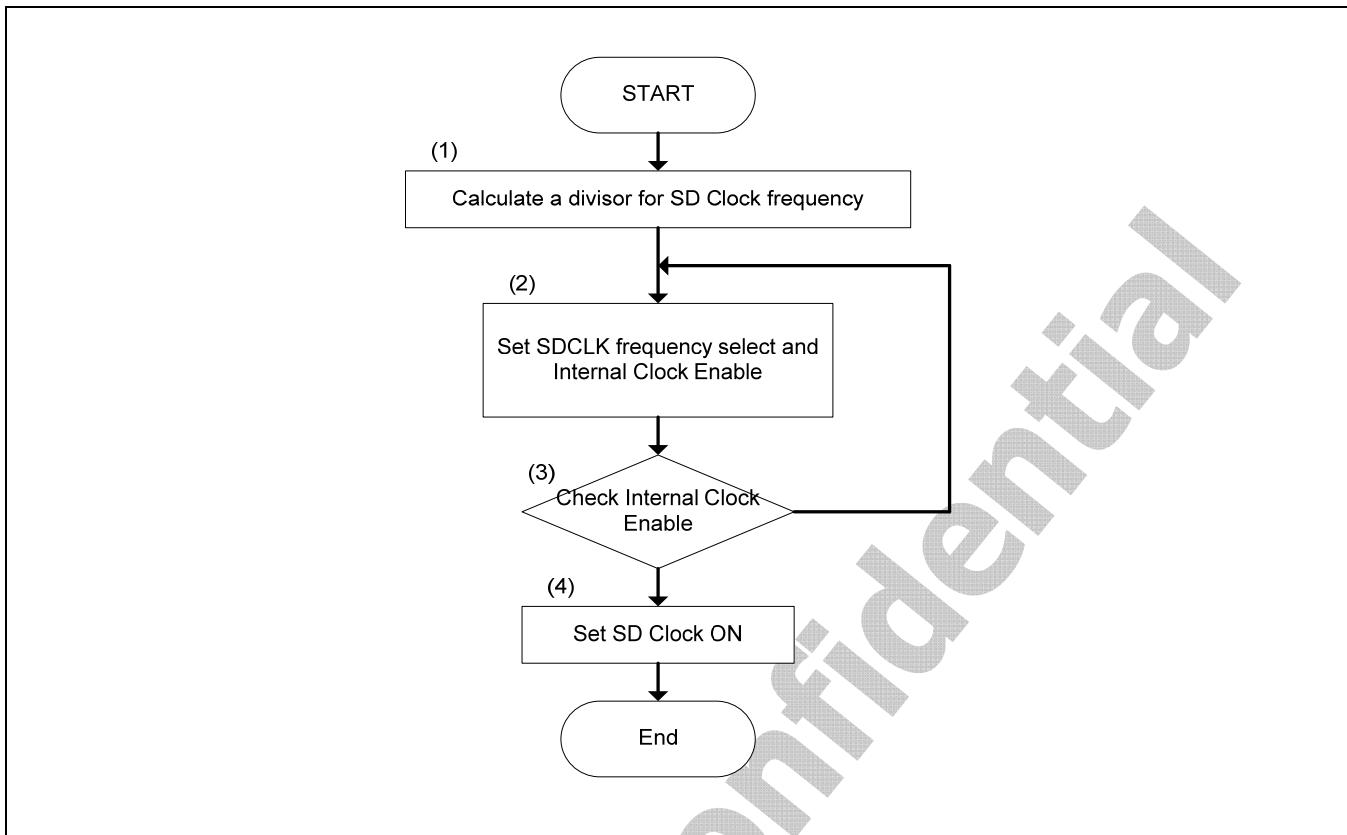


Figure 7-3 SD Clock Supply Sequence

The sequence to set SD Clock to a SD card is shown in [Figure 7-3](#). The clock is enabled before one of the following actions:

- a) Issuing a SD command
- b) Detect an interrupt from a SD card in 4-bit mode.

The steps to set SD Clock to a SD card:

1. Calculate a divisor to determine SD Clock frequency for SD Clock by reading Base Clock Frequency. Refer to clock control register (9.15).
2. Set Internal Clock Enable (ENINTCLK) and SDCLK Frequency Select in the Clock Control register in accordance with the calculated result of step (1).
3. Check Internal Clock Stable (STBLINTCLK) in the Clock Control register. Repeat this step until Clock Stable is 1.
4. Set SD Clock Enable (ENSDCLK) in the Clock Control register to 1. After ENSDCLK is set, the Host Controller starts SD Clock.

7.4.3 SD CLOCK STOP SEQUENCE

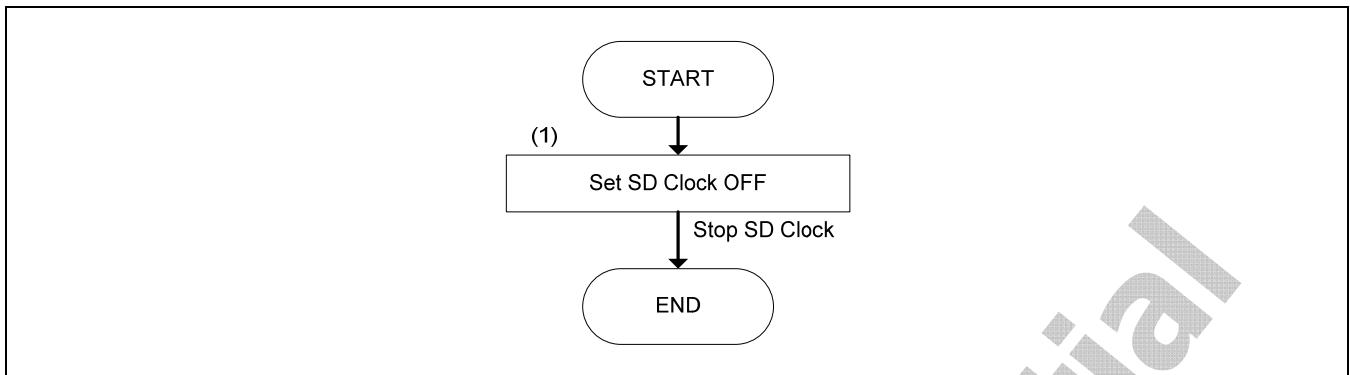


Figure 7-4 SD Clock Stop Sequence

The flow chart to stop the SD Clock is shown in [Figure 7-4](#). The Host Driver does not stop the SD Clock if a SD transaction takes place on the SD Bus -- namely, either Command Inhibit (DAT) or Command Inhibit (CMD) in the Present State register is set to 1.

1. Set SD Clock Enable (ENSDCLK) in the Clock Control register to 0. After ENSDCLK is set, the Host Controller stops SD Clock.

7.4.4 SD CLOCK FREQUENCY CHANGE SEQUENCE

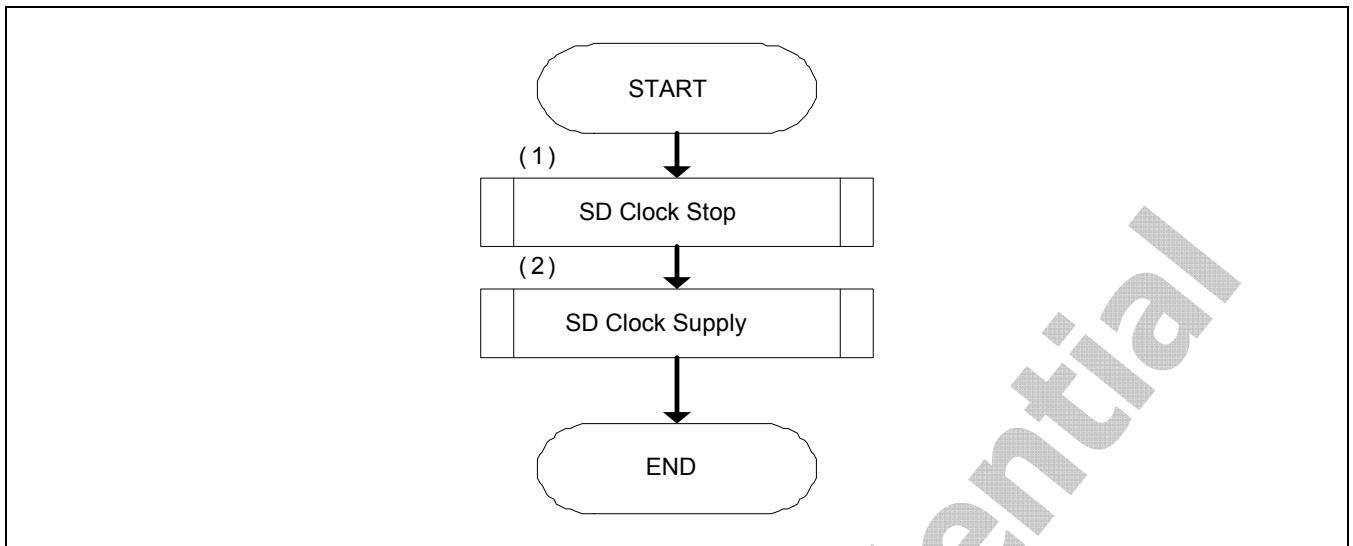


Figure 7-5 SD Clock Frequency Change Sequence

The sequence to change SD Clock frequency is shown in [Figure 7-5](#). If SD Clock is still off, skip step (1).

The steps to change SD Clock frequency:

1. Perform SD Clock Stop Sequence. Refer to [7.4.2 SD Clock Supply Sequence](#).
2. Perform SD Clock Supply Sequence. Refer to [7.4.3 SD Clock Stop Sequence](#).

7.4.5 SD BUS POWER CONTROL SEQUENCE

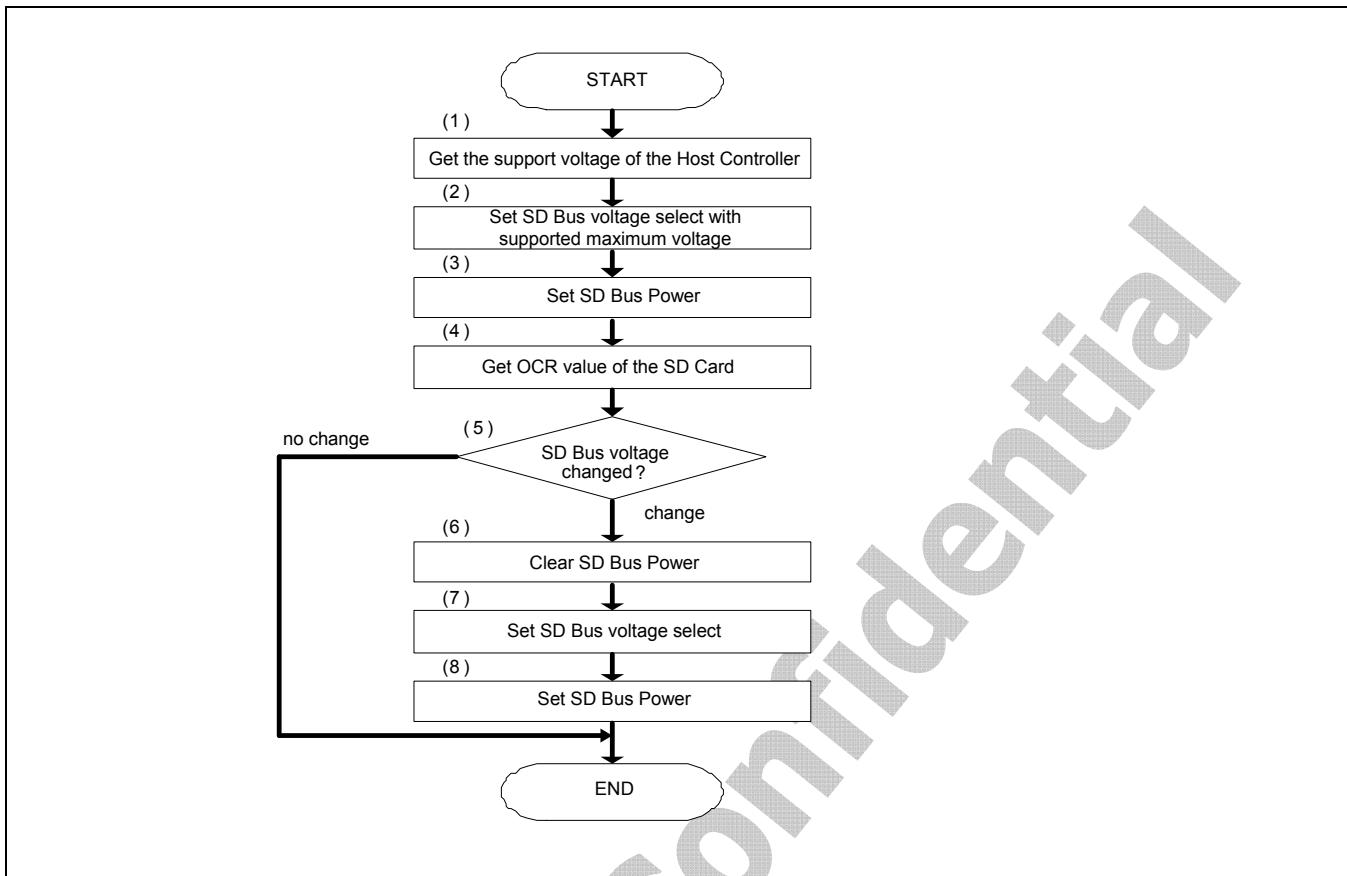


Figure 7-6 SD Bus Power Control Sequence

The sequence to control SD Bus Power is shown in [Figure 7-6](#). Steps to control SD Bus Power:

1. To get the support voltage of the Host Controller, read the Capabilities register.
2. Set SD Bus Voltage Select in external power regulator (optional) with maximum voltage that the Host Controller supports.
3. Set SD Bus Power (PWRON) in the Power Control register to 1.
4. Get the OCR value of all function internal of SD card.
5. Judge whether SD Bus voltage must be changed or not. If SD Bus voltage must be changed, continue with step (6). If SD Bus voltage is not to be changed, go to 'End'.
6. Set SD Bus Power in the Power Control register to 0 for clearing this bit. The card requires voltage rising from 0 volt to detect it correctly. The Host Driver sets SD Bus Voltage Select to clear SD Bus Power before changing voltage.
7. Set SD Bus Voltage Select (SELPWRLVL) in the Power Control register.
8. Set SD Bus Power (PWRON) in the Power Control register to 1.

NOTE: Step (2) and step (3) can be executed at same time. Also, step (7) and step (8) can be executed at same time.

7.4.6 CHANGE BUS WIDTH SEQUENCE

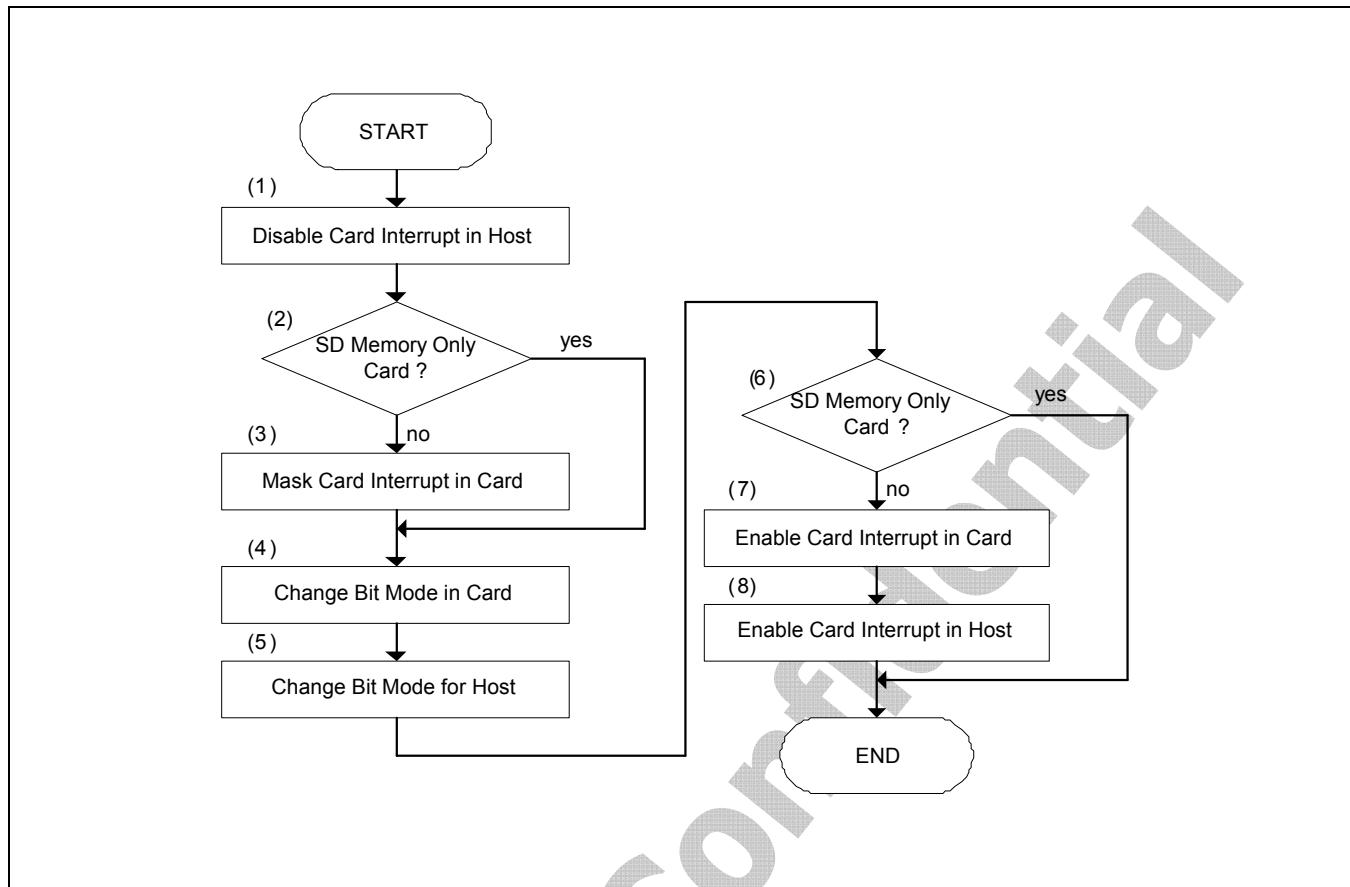


Figure 7-7 Change Bus Width Sequence

The sequence to change bit mode on SD Bus is shown in [Figure 7-7](#). Steps to change bit mode on SD Bus:

1. To mask incorrect interrupts that may occur while changing the bus width, Set Card Interrupt Status Enable (STACARDINT) in the Normal Interrupt Status Enable register to 0.
2. If SD memory card is used, go to step (4). In case of other card, go to step (3).
3. Use CMD52 to set "IENM" of the CCCR in a SDIO or SD combo card to 0.
4. Change the bit mode for a SD card. To change SD memory card bus width by ACMD6 (Set bus width) and SDIO card bus width set Bus Width of Bus Interface Control register in CCCR.
5. If you want to change to 4-bit mode, set Data Transfer Width (WIDE4) to 1 in the Host Control register. In another case (1-bit mode), set this bit to 0.
6. If SD memory card is used, go to 'End'. In case of other card, go to step (7).
7. Set "IENM" of the CCCR in a SDIO or SD combo card to 1 by CMD52.
8. Set Card Interrupt Status Enable to 1 in the Normal Interrupt Status Enable register.

7.4.7 TIMEOUT SETTING FOR DAT LINE

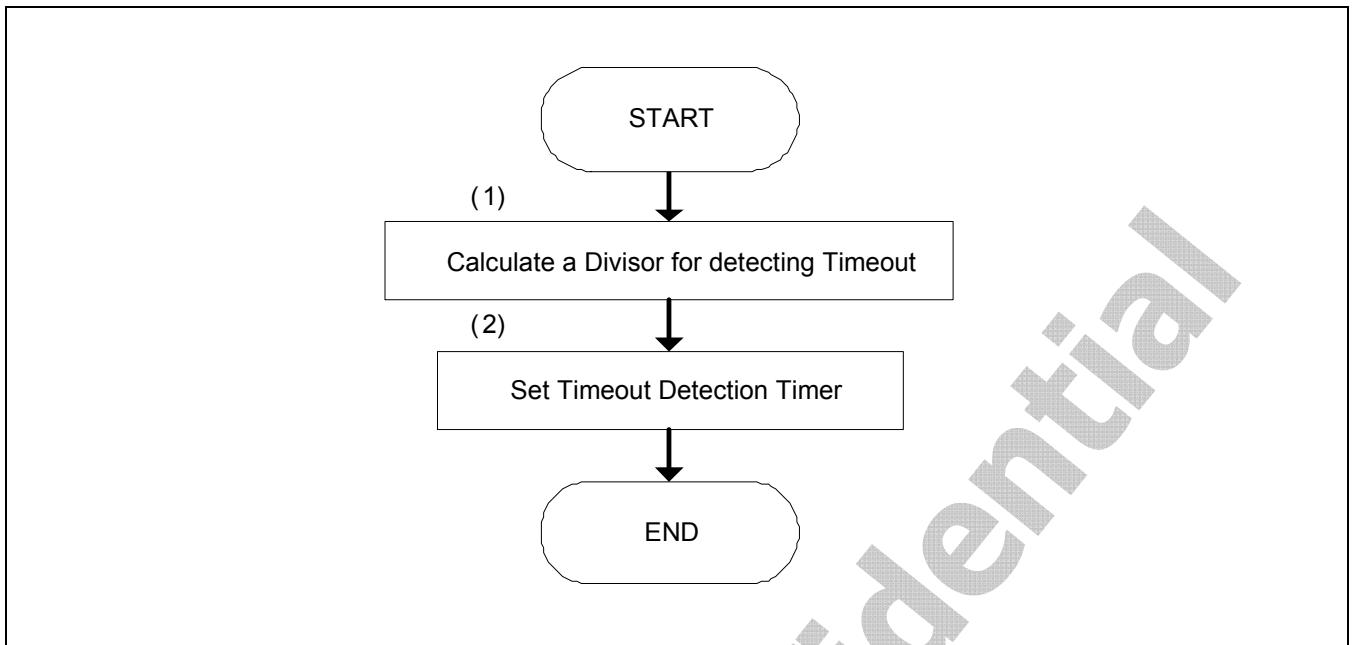


Figure 7-8 Timeout Setting Sequence

In order to detect timeout errors on DAT line, the Host Driver executes the following two steps before any SD transaction.

1. To calculate a divisor for detecting timeout, refer to Timeout Control Register (9.16).
2. (Set Data Timeout Counter Value (TIMEOUTCON) in the Timeout Control register in accordance with the value of step (1).

7.4.8 SD TRANSACTION GENERATION

This section describes the sequence to generate and control various kinds of SD transactions. SD transactions are classified into three cases, namely:

1. Transactions that do not use the DAT line.
2. Transactions that use the DAT line for the busy signal.
3. Transactions that use the DAT line for transferring data.

In this specification the first and the second case's transactions are classified as "Transaction Control without Data Transfer using DAT Line", the third case's transaction is classified as "Transaction Control with Data Transfer using DAT Line".

Refer to the specifications below for the detailed specifications on the SD Command itself:

- SD Memory Card Specification Part 1
PHYSICAL LAYER SPECIFICATION Version 1.01
- SD Card Specification PART E1
Secure Digital Input/Output (SDIO) Specification Version 1.00

7.4.9 SD COMMAND ISSUE SEQUENCE

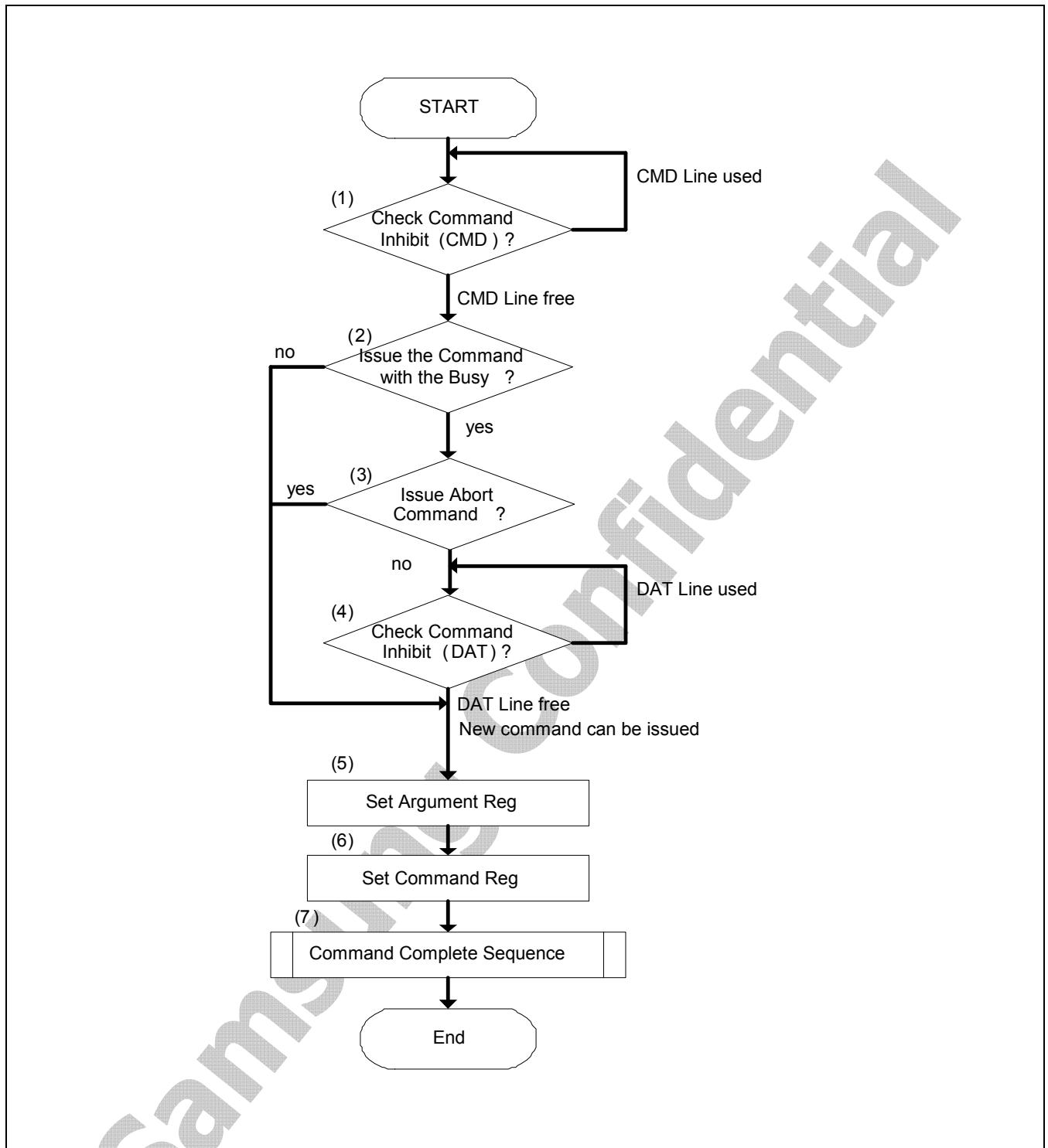


Figure 7-9 Timeout Setting Sequence

Steps to set Timeout:

1. Check Command Inhibit (CMD) in the Present State register. Repeat this step until Command Inhibit (CMD) is 0. If Command Inhibit (CMD) is 1, the Host Driver does not issue a SD Command.
2. If the Host Driver issues a SD Command with busy signal, go to step (3). If without busy signal, go to step (5).
3. If the Host Driver issues an abort command, go to step (5). If no abort command is issued, go to step (4).
4. Check Command Inhibit (DAT) in the Present State register. Repeat this step until Command Inhibit (DAT) is 0.
5. Set the value corresponding to the issued command in the Argument register.
6. Set the value corresponding to the issued command in the Command register.

NOTE: If the upper byte is written in the Command register, it issues a SD command.

7. Perform Command Complete Sequence

7.4.10 COMMAND COMPLETE SEQUENCE

The sequence to complete the SD Command is shown in [Figure 7-7](#), [Figure 7-8](#), [Figure 7-9](#) and [Figure 7-10](#). The following errors can occur during this sequence: Command Index/ End bit/ CRC/ Timeout Error.

Steps to complete the SD command:

1. Wait for the Command Complete Interrupt. If the Command Complete Interrupt occurs, go to step (2).
2. Write 1 to Command Complete (STACMDCMPLT) in the Normal Interrupt Status register to clear this bit.
3. Read the Response register and get necessary information in accordance with the issued command.
4. Judge whether the command uses the Transfer Complete Interrupt or not. If it uses Transfer Complete, proceed with step (5). If not, go to step (7).
5. Wait for the Transfer Complete Interrupt. If the Transfer Complete Interrupt occurs, go to step (6).
6. Write 1 to Transfer Complete (STATRANCMPLT) in the Normal Interrupt Status register to clear this bit.
7. Check for errors in Response Data. If there is no error, proceed with step (8). If there is an error, go to step (9).
8. Return Status of "No Error".
9. Return Status of "Response Contents Error".

NOTE:

1. While waiting for the Transfer Complete interrupt, the Host Driver issues commands that do not use the busy signal.
2. The Host Driver monitors Transfer Complete to judge the Auto CMD12 (Stop Command) complete.
3. If the last block of un-protected area is read using memory multiple blocks read command (CMD18), OUT_OF_RANGE error may occur even if the sequence is correct. The Host Driver must ignore this error. This error appears in the response of Auto CMD12 or in the response of the next memory command.

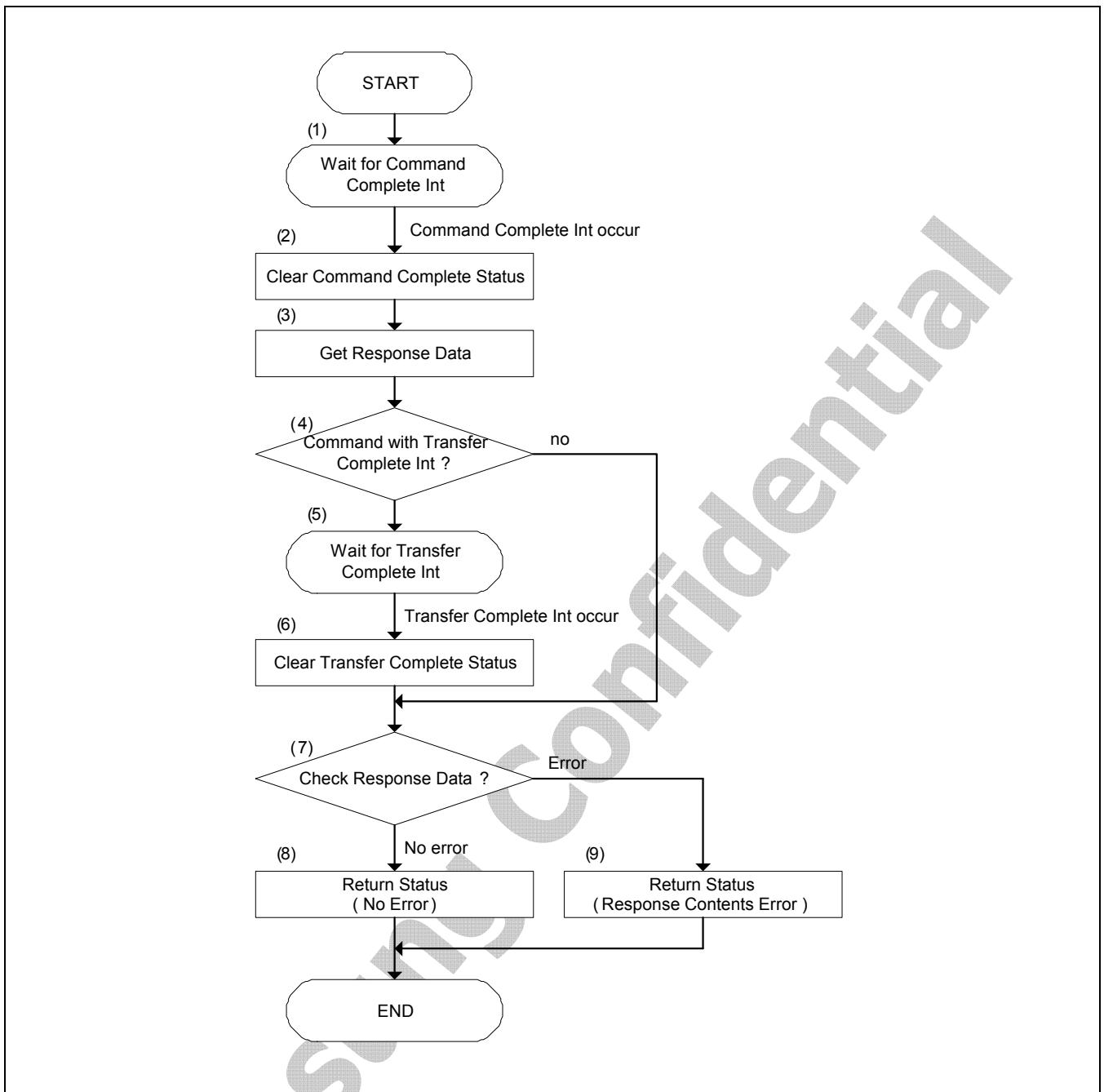


Figure 7-10 Command Complete Sequence

7.4.11 TRANSACTION CONTROL WITH DATA TRANSFER USING DAT LINE

Depending on whether DMA (optional) is used or not, there are two execution methods. The sequence without using DMA is shown in [Figure 7-11](#) and the sequence using DMA is shown in [Figure 7-12](#).

In addition, the sequences for SD transfers are basically classified according to how the number of blocks is specified. The three kinds of classification are as follows:

1. Single Block Transfer:

The number of blocks is specified to the Host Controller before the transfer. The number of blocks specified is always one.

2. Multiple Block Transfer:

The number of blocks is specified to the Host Controller before the transfer. The number of blocks specified is one or more.

3. Infinite Block Transfer:

The number of blocks is not specified to the Host Controller before the transfer. This transfer is continued until an abort transaction is executed. This abort transaction is performed by CMD12 (Stop Command) in the case of a SD memory card and by CMD52 (IO_RW_DIRECT) in the case of a SDIO card.

7.4.12 SEQUENCE WITHOUT USING DMA

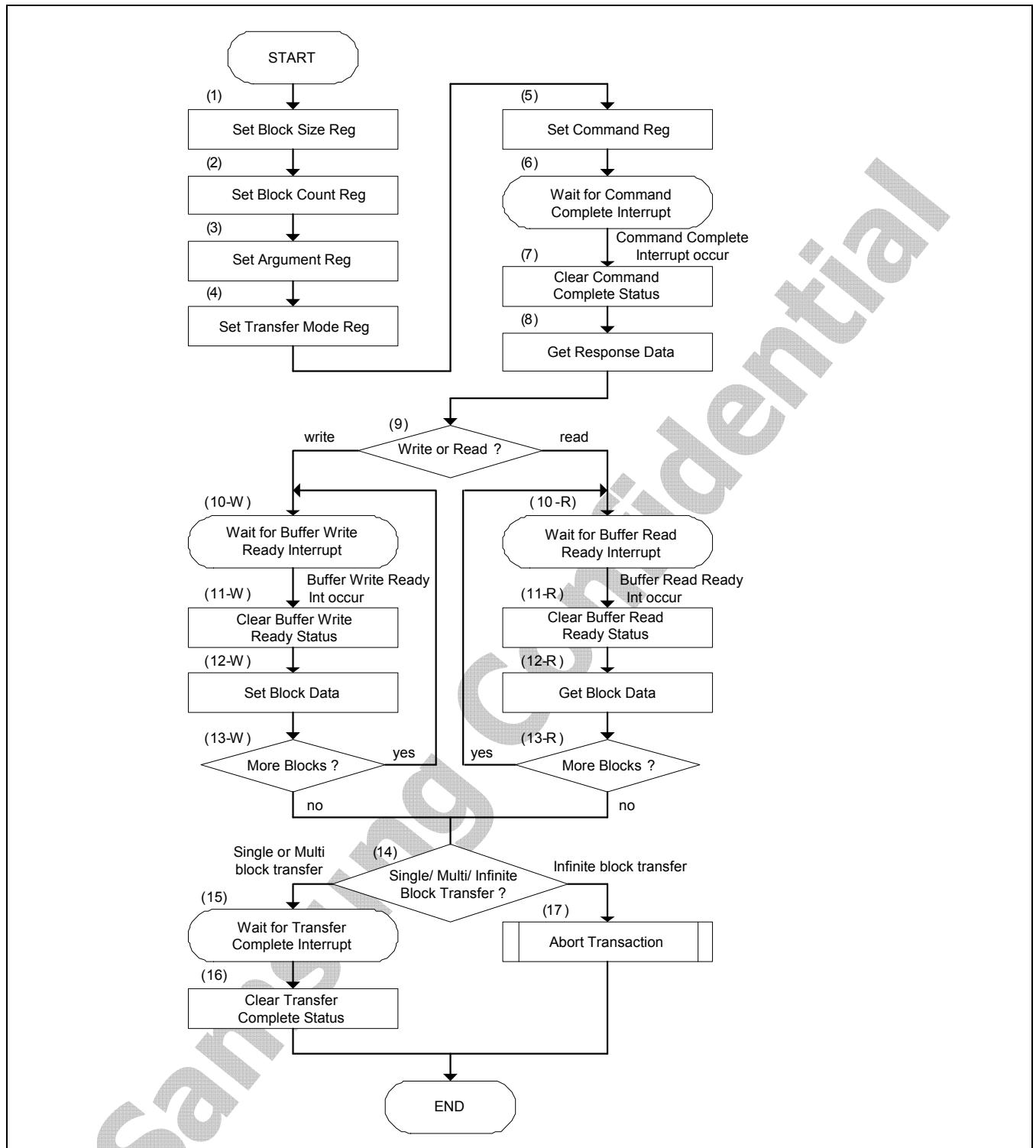


Figure 7-11 Transaction Control with Data Transfer Using DAT Line Sequence (Not using DMA)

1. Set the value corresponding to the executed data byte length of one block to Block Size register.
2. Set the value corresponding to the executed data block count to Block Count register.
3. Set the value corresponding to the issued command to Argument register.
4. Set the value to Multi / Single Block Select and Block Count Enable. At this time, set the value corresponding to the issued command to Data Transfer Direction, Auto CMD12 Enable and DMA Enable.
5. Set the value corresponding to the issued command to Command register.

NOTE: If the upper byte is written in the Command register, it issues a SD command

6. Wait for the Command Complete Interrupt.
7. Write 1 to the Command Complete (STACMDCMPLT) in the Normal Interrupt Status register to clear this bit.
8. Read Response register and get necessary information in accordance with the issued command.
9. If this sequence is for write to a card, proceed to step (10-W). If read from a card, go to step (10-R).
10. (10-W) Wait for Buffer Write Ready Interrupt.
11. (11-W) Write 1 to the Buffer Write Ready (STABUFWTRDY) in the Normal Interrupt Status register to clear this bit.
12. (12-W) Write block data (in according to the number of bytes specified at the step (1)) to Buffer Data Port register.
13. (13-W) Repeat until all blocks are sent and then go to step (14).
14. (10-R) Wait for the Buffer Read Ready Interrupt.
15. (11-R) Write 1 to the Buffer Read Ready (STABUFRDRDY) in the Normal Interrupt Status register to clear this bit.
16. (12-R) Read block data (in according to the number of bytes specified at the step (1)) from the Buffer Data Port register.
17. (13-R) Repeat until all blocks are received and proceed to step (14).
18. (14) If this sequence is for Single or Multiple Block Transfer, proceed to step (15). In case of Infinite Block Transfer, go to step (17).
19. (15) Wait for Transfer Complete Interrupt.
20. (16) Write 1 to the Transfer Complete (STATRANCMPLT) in the Normal Interrupt Status register to clear this bit.
21. (17) Perform the sequence for Abort Transaction.

NOTE: Step (1) and Step (2) can be executed at same time. Step (4) and Step (5) can be executed at same time

7.4.13 SEQUENCE USING DMA

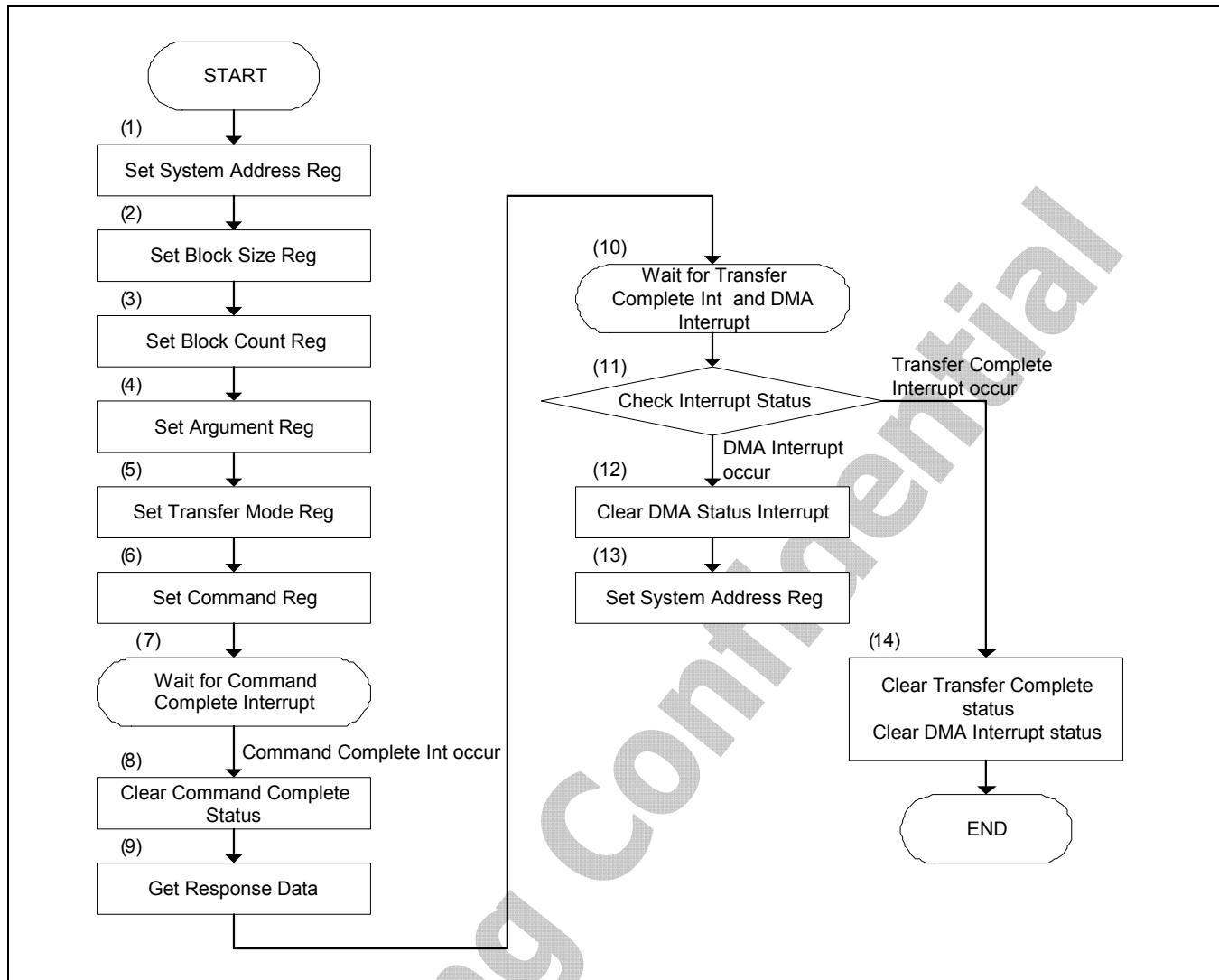


Figure 7-12 Transaction Control with Data Transfer Using DAT Line Sequence (Using DMA)

1. Set the system address for DMA in the System Address register.
2. Set the value corresponding to the executed data byte length of one block in the Block Size register.
3. Set the value corresponding to the executed data block count in the Block Count register (BLKCNT).
4. Set the value corresponding to the issued command in the Argument register (ARGUMENT).
5. Set the values for Multi / Single Block Select and Block Count Enable.

At this time, set the value corresponding to the issued command for Data Transfer Direction, Auto CMD12 Enable and DMA Enable.

6. Set the value corresponding to the issued command in the Command register (CMDREG).

NOTE: If the upper byte is written in the Command register, it issues a SD command and DMA is operated.

7. Wait for the Command Complete Interrupt.
8. Write 1 to the Command Complete (STACMDCMPLT) in the Normal Interrupt Status register to clear this bit.
9. Read Response register and get necessary information in accordance with the issued command.
10. Wait for the Transfer Complete Interrupt and DMA Interrupt.
11. If Transfer Complete (STATRANCMPLT) is set 1, go to Step (14) else if DMA Interrupt is set to 1; proceed to Step (12). Transfer Complete is higher priority than DMA Interrupt.
12. Write 1 to the DMA Interrupt in the Normal Interrupt Status register to clear this bit.
13. Set the next system address of the next data position to the System Address register and go to Step (10).
14. Write 1 to the Transfer Complete and DMA Interrupt in the Normal Interrupt Status register to clear this bit.

NOTE: Step (2) and Step (3) can be executed simultaneously. Step (5) and Step (6) can also be executed simultaneously.

7.5 ABORT TRANSACTION

To perform Abort transaction issue CMD12 (Stop Command) for a SD memory and issue CMD52 for a SDIO card. There are two cases where the Host Driver needs to issue an Abort Transaction. The first case is if the Host Driver stops Infinite Block Transfers. The second case is if the Host Driver stops transfers while a Multiple Block Transfer is executing.

There are two ways to issue an Abort Command, namely, asynchronous abort and synchronous abort.

In an asynchronous abort sequence, the Host Driver issues an Abort Command at anytime unless Command Inhibit (CMD) in the Present State register is set to 1.

In a synchronous abort, the Host Driver issues an Abort Command after the data transfer stopped by using Stop At Block Gap Request in the Block Gap Control register.

7.6 DMA TRANSACTION

DMA allows a peripheral to read and write memory without intervention from the CPU. DMA executes one SD command transaction. Host Controllers that support DMA supports both single block and multiple block transfers.

The System Address register points to the first data address, and data is then accessed sequentially from that address. Host Controller registers remains accessible for issuing non-DAT line commands during a DMA transfer. The result of a DMA transfer is same regardless of the system bus transaction method. DMA does not support infinite transfers.

DMA transfers are stopped and restarted using control bits in the Block Gap Control register. If the Stop At Block Gap Request is set, DMA transfers is suspended. If the Continue Request is set or a Resume Command is issued, DMA continues to execute transfers. Refer to the Block Gap Control register for details. If SD Bus errors occur, SD Bus transfers and DMA transfers are stopped. Setting the Software Reset for DAT Line in the Software Reset register aborts DMA transfers.

7.7 ADMA (ADVANCED DMA)

In the SD Host Controller Standard Specification Version 2.00, new DMA transfer algorithm called ADMA (Advanced DMA) is defined. The DMA algorithm defined in the SD Host Controller Standard Specification Version 1.00 is called SDMA (Single Operation DMA). SDMA had disadvantage that DMA Interrupt generated at every page boundary disturbs CPU to reprogram the new system address. This SDMA algorithm forms a performance bottleneck by interruption at every page boundary. ADMA adopts scatter gather DMA algorithm so that higher data transfer speed is available. The Host Driver can program a list of data transfers between system memory and SD card to the Descriptor Table before executing ADMA. It enables ADMA to operate without interrupting the Host Driver. Furthermore, ADMA can support not only 32-bit system memory addressing but also 64-bit system memory addressing. The 32-bit system memory addressing uses lower 32-bit field of 64-bit address registers. Support of SDMA and ADMA are optional for the Host Controller. ADMA improves the restriction so that data of any location and any size can be transferred in system memory. The format of Descriptor Table is different between them. The Host Controller Specification Ver2.00 defines ADMA as standard ADMA.

7.7.1 BLOCK DIAGRAM OF ADMA

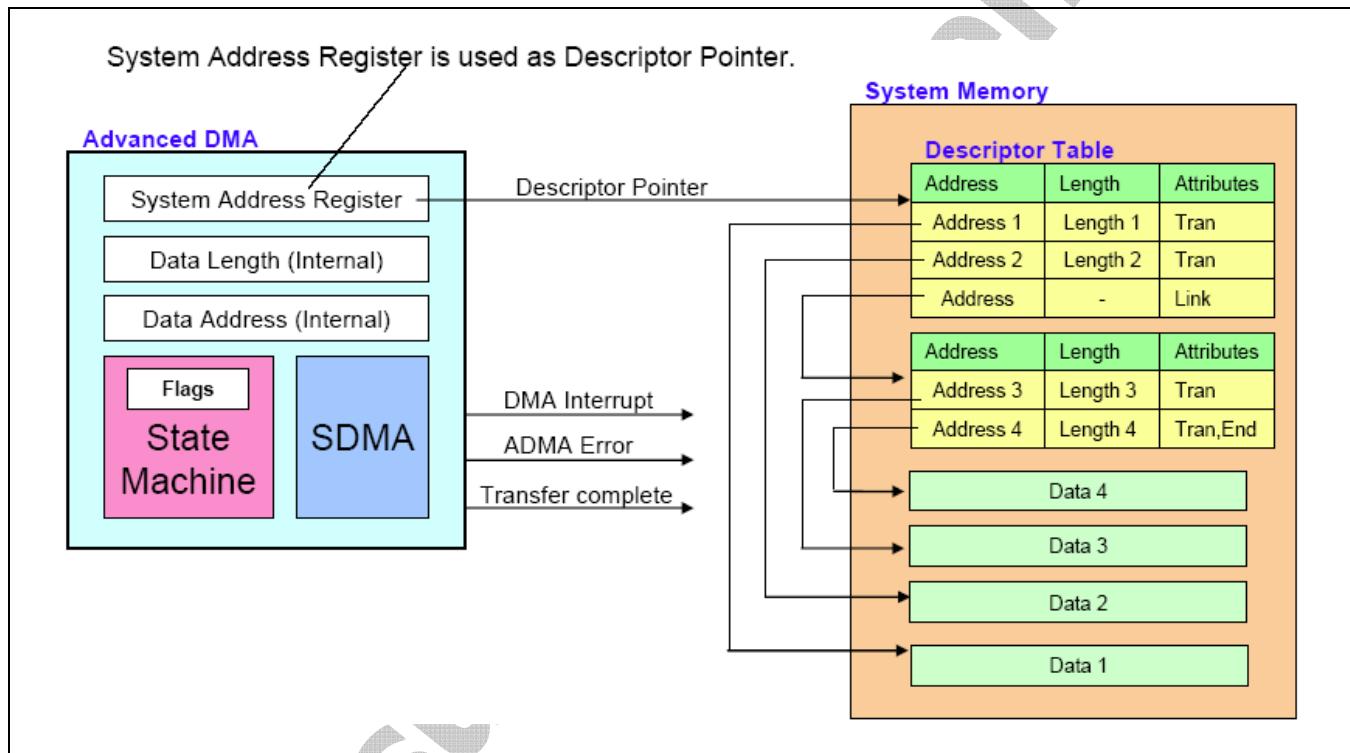


Figure 7-13 Block Diagram of ADMA

[Figure 7-13](#) shows block diagram of ADMA. The Descriptor Table is created in system memory by the Host Driver. 32-bit Address Descriptor Table is used for the system with 32-bit addressing and 64-bit Address Descriptor Table is used for the system with 64-bit addressing. Each descriptor line (one executable unit) consists of address, length and attribute field. The attribute specifies operation of the descriptor line. ADMA includes SDMA, State Machine and Registers circuits. ADMA does not use 32-bit SDMA System Address Register (offset 0) but uses the 64-bit Advanced DMA System Address register (offset 058h) for descriptor pointer. Writing Command register triggers off ADMA transfer. ADMA fetches one descriptor line and execute. This procedure is repeated until end of descriptor is found (End=1 in attribute).

7.7.2 EXAMPLE OF ADMA PROGRAMMING

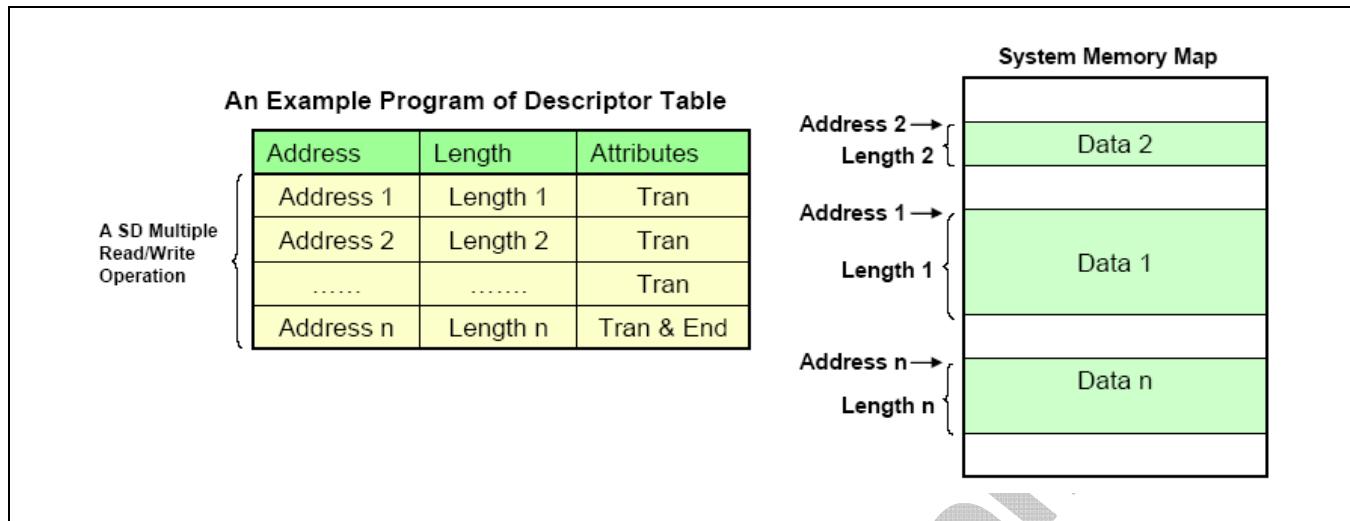


Figure 7-14 Example of ADMA Data Transfer

[Figure 7-14](#) shows a typical ADMA descriptor program. The data area is sliced in various lengths and each slice is placed somewhere in system memory. The Host Driver describes the Descriptor Table with set of address, length and attributes. Each sliced data is transferred in turns as programmed in descriptor.

7.7.3 DATA ADDRESS AND DATA LENGTH REQUIREMENTS

There are 3 requirements to program descriptor. The minimum unit of address is 4byte.

The maximum data length of each descriptor line is less than 64KB.

$$\text{Total Length} = \text{Length 1} + \text{Length 2} + \text{Length 3} + \dots + \text{Length n}$$

= multiple of Block Size If total length of a descriptor were not multiple of block size, ADMA transfer might not be terminated. In this case, the transfer should be aborted by data timeout. Block Count register is defined as 16-bit register and it limits the maximum of 65535 blocks transfer. If ADMA operation is less than or equal 65535 blocks transfer, Block Count register can be used. In this case, total length of Descriptor Table shall be equivalent to multiply block size and block count. If ADMA operation is more than 65535 blocks transfer, Block Count register shall be disabled by setting 0 to Block Count Enable in the Transfer Mode Register. In this case, length of data transfer is not designated by block count but Descriptor Table. Therefore, the timing of detecting the last block on SD bus may be different and it affects the control of Read Transfer Active, Write Transfer Active and DAT line Active in the Present State register. In case of read operation, several blocks may be read more than required. The Host Driver shall ignore out of range error if the read operation is for the last block of memory area.

7.7.4 DESCRIPTOR TABLE

Attribute Column Fields:

Valid	Valid=1 indicates this line of descriptor is effective. If Valid=0 generate ADMA Error Interrupt and stop ADMA to prevent runaway.
End	End=1 indicates to end of descriptor. The Transfer Complete Interrupt is generated when the operation of the descriptor line is completed.
Int	INT=1 generates DMA Interrupt when the operation of the descriptor line is completed.

Operation Column Definitions:

Act2	Act1	Symbol	Comment	Operation
0	0	Nop	No Operation	Do not execute current line and go to next line.
0	1	rsv	reserved	(Same as Nop. Do not execute current line and go to next line.)
1	0	Tran	Transfer Data	Transfer data of one descriptor line
1	1	Link	Link Descriptor	Link to another descriptor

Figure 7-15 32-bit Address Descriptor Table

Figure 7-15 shows the definition of 32-bit Address Descriptor Table. One descriptor line consumes 64-bit (8-byte) memory space. Attribute is used to control descriptor. 3 action symbols are specified. "Nop" operation skips current descriptor line and fetches next one. "Tran" operation transfers data designated by address and length field. "Link" operation is used to connect separated two descriptors. The address field of link points to next Descriptor Table. The combination of Act2=0 and Act1=1 is reserved and defined the same operation as Nop. A future version of controller may use this field and redefine a new operation. 32-bit address is stored in the lower 32-bit of 64-bit address registers. Address field shall be set on 32-bit boundary (Lower 2-bit is always set to 0) for 32-bit address descriptor table. [Table 7-1](#) shows the definition of length field in the Descriptor Table.

Table 7-1 ADMA Length Field

Length Field	Value of Length
0000h	65536 bytes
0001h	1 byte
0002h	2 bytes
.....
FFFFh	65535 bytes

7.7.5 ADMA STATES

[Figure 7-16](#) shows state diagram of ADMA. Four states are defined, namely, Fetch Descriptor state, Change Address state, Transfer Data state, and Stop ADMA state. Operation of each state is explained in [Table 7-2](#).

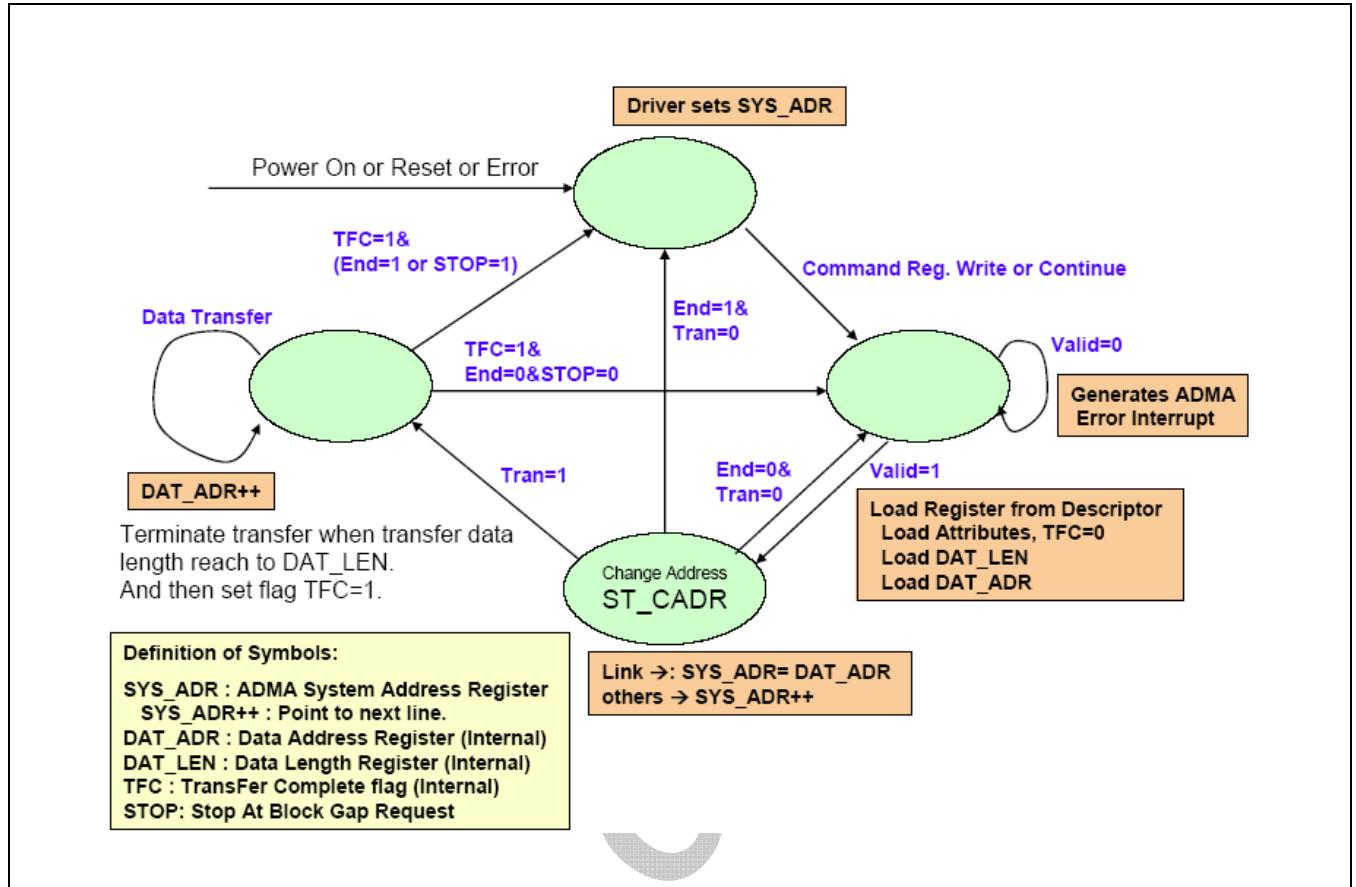


Figure 7-16 State Diagram of the ADMA

Table 7-2 ADMA States

State Name	Operation
ST_FDS (Fetch Descriptor)	ADMA fetches a descriptor line and set parameters in internal registers. Next go to ST_CADR state.
ST_CADR (Change Address)	Link operation loads another Descriptor address to ADMA System Address register. In other operations, ADMA System Address register is incremented to point next descriptor line. If End=0, go to ST_TFR state. ADMA shall not be stopped at this state even if some errors occur.
ST_TFR (Transfer Data)	Data transfer of one descriptor line is executed between system memory and SD card. If data transfer continues (End=0) go to ST_FDS state. If data transfer completes, go to ST_STOP state.
ST_STOP (Stop DMA)	ADMA stays in this state in following cases: (1) After Power on reset or software reset. (2) All descriptor data transfers are completed If a new ADMA operation is started by writing Command register, go to ST_FDS state.

ADMA does not support suspend / resume function but stop and continue are available. When the Stop at Block Gap Request in the Block Gap Control register is set during the ADMA operation, the Block Gap Event Interrupt is generated when ADMA is stopped at block gap. The Host Controller shall stop ADMA read operation by using Read Wait or stopping SD Clock. While stopping ADMA, SD commands cannot be issued. (In case of Host Controller version 1.00, the Stop at Block Gap Request can be set only when the card supports the Read Wait.)

Error occurrence during ADMA transfer can stop ADMA operation and generates ADMA Error Interrupt. The ADMA Error State field in the ADMA Error Status register holds state of ADMA stopped. The host driver can identify error descriptor location by following method.

If ADMA stopped at ST_FDS state, the ADMA System Address Register points the error descriptor line. If ADMA stopped at ST_TFR or ST_STOP state, the ADMA System Address Register points the next location of error descriptor line. By this reason, ADMA2 shall not stop at ST_CADR state.

7.8 I/O DESCRIPTION

Signal	I/O	Description	Pad	Type
SD_0_CLK	OUTPUT	Clock for SDMMC0	Xmmc0CLK	muxed
SD_0_CMD	IN/OUT	Command for SDMMC0	Xmmc0CMD	muxed
SD_0_DATA[0]	IN/OUT	Data for SDMMC0	Xmmc0DATA[0]	muxed
SD_0_DATA[1]	IN/OUT	Data for SDMMC0	Xmmc0DATA[1]	muxed
SD_0_DATA[2]	IN/OUT	Data for SDMMC0	Xmmc0DATA[2]	muxed
SD_0_DATA[3]	IN/OUT	Data for SDMMC0	Xmmc0DATA[3]	muxed
SD_0_DATA[4]	IN/OUT	Data for SDMMC0	Xmmc1DATA[0]	muxed
SD_0_DATA[5]	IN/OUT	Data for SDMMC0	Xmmc1DATA[1]	muxed
SD_0_DATA[6]	IN/OUT	Data for SDMMC0	Xmmc1DATA[2]	muxed
SD_0_DATA[7]	IN/OUT	Data for SDMMC0	Xmmc1DATA[3]	muxed
SD_0_CDn	INPUT	Card Detect for SDMMC0	Xmmc0CDn	muxed
SD_1_CLK	OUTPUT	Clock for SDMMC1	Xmmc1CLK	muxed
SD_1_CMD	IN/OUT	Command for SDMMC1	Xmmc1CMD	muxed
SD_1_DATA[0]	IN/OUT	Data for SDMMC1	Xmmc1DATA[0]	muxed
SD_1_DATA[1]	IN/OUT	Data for SDMMC1	Xmmc1DATA[1]	muxed
SD_1_DATA[2]	IN/OUT	Data for SDMMC1	Xmmc1DATA[2]	muxed
SD_1_DATA[3]	IN/OUT	Data for SDMMC1	Xmmc1DATA[3]	muxed
SD_1_CDn	INPUT	Card Detect for SDMMC1	Xmmc1CDn	muxed
SD_2_CLK	OUTPUT	Clock for SDMMC2	Xmmc2CLK	muxed
SD_2_CMD	IN/OUT	Command for SDMMC2	Xmmc2CMD	muxed
SD_2_DATA[0]	IN/OUT	Data for SDMMC2	Xmmc2DATA[0]	muxed
SD_2_DATA[1]	IN/OUT	Data for SDMMC2	Xmmc2DATA[1]	muxed
SD_2_DATA[2]	IN/OUT	Data for SDMMC2	Xmmc2DATA[2]	muxed
SD_2_DATA[3]	IN/OUT	Data for SDMMC2	Xmmc2DATA[3]	muxed
SD_2_DATA[4]	IN/OUT	Data for SDMMC2	Xmmc3DATA[0]	muxed
SD_2_DATA[5]	IN/OUT	Data for SDMMC2	Xmmc3DATA[1]	muxed
SD_2_DATA[6]	IN/OUT	Data for SDMMC2	Xmmc3DATA[2]	muxed
SD_2_DATA[7]	IN/OUT	Data for SDMMC2	Xmmc3DATA[3]	muxed
SD_2_CDn	INPUT	Card Detect for SDMMC2	Xmmc2CDn	muxed
SD_3_CLK	OUTPUT	Clock for SDMMC3	Xmmc3CLK	muxed
SD_3_CMD	IN/OUT	Command for SDMMC3	Xmmc3CMD	muxed
SD_3_DATA[0]	IN/OUT	Data for SDMMC3	Xmmc3DATA[0]	muxed
SD_3_DATA[1]	IN/OUT	Data for SDMMC3	Xmmc3DATA[1]	muxed
SD_3_DATA[2]	IN/OUT	Data for SDMMC3	Xmmc3DATA[2]	muxed
SD_3_DATA[3]	IN/OUT	Data for SDMMC3	Xmmc3DATA[3]	muxed

Signal	I/O	Description	Pad	Type
SD_3_CDn	INPUT	Card Detect for SDMMC3	Xmmc3CDn	muxed

NOTE: SDMMC external pads are shared with CAMIF or SPI. In order to use these pads for SDMMC, set the GPIO before the SDMMC started. Refer to the GPIO chapter for correct GPIO settings.

Samsung Confidential

7.9 REGISTER DESCRIPTION

7.9.1 REGISTER MAP

Configuration register fields are assigned to one of the attributes described below:

Register	Address	R/W	Description	Reset Value
SDMASYSAD0	0xEB00_0000	R/W	SDMA System Address register (Channel 0)	0x0
BLKSIZE0	0xEB00_0004	R/W	Host DMA Buffer Boundary and Transfer Block Size Register (Channel 0)	0x0
BLKCNT0	0xEB00_0006	R/W	Blocks count for current transfer (channel 0)	0x0
ARGUMENT0	0xEB00_0008	R/W	Command Argument Register (Channel 0)	0x0
TRNMOD0	0xEB00_000C	R/W	Transfer Mode Setting Register (Channel 0)	0x0
CMDREG0	0xEB00_000E	R/W	Command Register (Channel 0)	0x0
RSPREG0_0	0xEB00_0010	ROC	Response Register 0 (Channel 0)	0x0
RSPREG1_0	0xEB00_0014	ROC	Response Register 1 (Channel 0)	0x0
RSPREG2_0	0xEB00_0018	ROC	Response Register 2 (Channel 0)	0x0
RSPREG3_0	0xEB00_001C	ROC	Response Register 3 (Channel 0)	0x0
BDATA0	0xEB00_0020	R/W	Buffer Data Register (Channel 0)	0x0
PRNSTS0	0xEB00_0024	R/ROC	Present State Register (Channel 0)	0x000A0000
HOSTCTL0	0xEB00_0028	R/W	Present State Register (Channel 0)	0x0
PWRCON0	0xEB00_0029	R/W	Present State Register (Channel 0)	0x0
BLKGAP0	0xEB00_002A	R/W	Block Gap Control Register (Channel 0)	0x0
WAKCON0	0xEB00_002B	R/W	Wakeups Control Register (Channel 0)	0x0
CLKCON0	0xEB00_002C	R/W	Command Register (Channel 0)	0x0
TIMEOUTCON0	0xEB00_002E	R/W	Timeout Control Register (Channel 0)	0x0
SWRST0	0xEB00_002F	R/W	Software Reset Register (Channel 0)	0x0
NORINTSTS0	0xEB00_0030	ROC/ RW1C	Normal Interrupt Status Register (Channel 0)	0x0
ERRINTSTS0	0xEB00_0032	ROC/ RW1C	Error Interrupt Status Register (Channel 0)	0x0
NORINTSTSEN0	0xEB00_0034	R/W	Normal Interrupt Status Enable Register (Channel 0)	0x0
ERRINTSTSEN0	0xEB00_0036	R/W	Error Interrupt Status Enable Register (Channel 0)	0x0
NORINTSIGENO	0xEB00_0038	R/W	Normal Interrupt Signal Enable Register (Channel 0)	0x0
ERRINTSIGENO	0xEB00_003A	R/W	Error Interrupt Signal Enable Register (Channel 0)	0x0
ACMD12ERRSTS0	0xEB00_003C	ROC	Auto CMD12 error status register (channel 0)	0x0
CAPAREG0	0xEB00_0040	HWInit	Capabilities Register (Channel 0)	0x05E80080
MAXCURR0	0xEB00_0048	HWInit	Maximum Current Capabilities Register	0x0

Register	Address	R/W	Description	Reset Value
			(Channel 0)	
FEAER0	0xEB00_0050	W	Force Event Auto CMD12 Error Interrupt Register (Channel 0)	0x0000
FEERR0	0xEB00_0052	W	Force Event Error Interrupt Register Error Interrupt (Channel 0)	0x0000
ADMAERRO	0xEB00_0054	R/W	ADMA Error Status Register (Channel 0)	0x00
ADMASYSADDR0	0xEB00_0058	R/W	ADMA System Address Register (Channel 0)	0x00
CONTROL2_0	0xEB00_0080	R/W	Control register 2 (Channel 0)	0x0
CONTROL3_0	0xEB00_0084	R/W	FIFO Interrupt Control (Control Register 3) (Channel 0)	0x7F5F3F1F
CONTROL4_0	0xEB00_008C	R/W	Control register 4 (Channel 0)	0x0
HCVER0	0xEB00_00FE	HWInit	Host Controller Version Register (Channel 0)	0x2401
SDMASYSAD1	0xEB10_0000	R/W	SDMA System Address register (Channel 1)	0x0
BLKSIZE1	0xEB10_0004	R/W	Host DMA Buffer Boundary and Transfer Block Size Register (Channel 1)	0x0
BLKCNT1	0xEB10_0006	R/W	Blocks count for current transfer (channel 1)	0x0
ARGUMENT1	0xEB10_0008	R/W	Command Argument Register (Channel 1)	0x0
TRNMOD1	0xEB10_000C	R/W	Transfer Mode Setting Register (Channel 1)	0x0
CMDREG1	0xEB10_000E	R/W	Command Register (Channel 1)	0x0
RSPREG0_1	0xEB10_0010	ROC	Response Register 0 (Channel 1)	0x0
RSPREG1_1	0xEB10_0014	ROC	Response Register 1 (Channel 1)	0x0
RSPREG2_1	0xEB10_0018	ROC	Response Register 2 (Channel 1)	0x0
RSPREG3_1	0xEB10_001C	ROC	Response Register 3 (Channel 1)	0x0
BDATA1	0xEB10_0020	R/W	Buffer Data Register (Channel 1)	0x0
PRNSTS1	0xEB10_0024	R/ROC	Present State Register (Channel 1)	0x000A0000
HOSTCTL1	0xEB10_0028	R/W	Present State Register (Channel 1)	0x0
PWRCON1	0xEB10_0029	R/W	Present State Register (Channel 1)	0x0
BLKGAP1	0xEB10_002A	R/W	Block Gap Control Register (Channel 1)	0x0
WAKCON1	0xEB10_002B	R/W	Wakeups Control Register (Channel 1)	0x0
CLKCON1	0xEB10_002C	R/W	Command Register (Channel 1)	0x0
TIMEOUTCON1	0xEB10_002E	R/W	Timeout Control Register (Channel 1)	0x0
SWRST1	0xEB10_002F	R/W	Software Reset Register (Channel 1)	0x0
NORINTSTS1	0xEB10_0030	ROC/ RW1C	Normal Interrupt Status Register (Channel 1)	0x0
ERRINTSTS1	0xEB10_0032	ROC/ RW1C	Error Interrupt Status Register (Channel 1)	0x0
NORINTSTSEN1	0xEB10_0034	R/W	Normal Interrupt Status Enable Register (Channel 1)	0x0
ERRINTSTSEN1	0xEB10_0036	R/W	Error Interrupt Status Enable Register (Channel 1)	0x0

Register	Address	R/W	Description	Reset Value
NORINTSIGEN1	0xEB10_0038	R/W	Normal Interrupt Signal Enable Register (Channel 1)	0x0
ERRINTSIGEN1	0xEB10_003A	R/W	Error Interrupt Signal Enable Register (Channel 1)	0x0
ACMD12ERRSTS1	0xEB10_003C	ROC	Auto CMD12 error status register (channel 1)	0x0
CAPAREG1	0xEB10_0040	HWInit	Capabilities Register (Channel 1)	0x05E80080
MAXCURR1	0xEB10_0048	HWInit	Maximum Current Capabilities Register (Channel 1)	0x0
FEAER1	0xEB10_0050	W	Force Event Auto CMD12 Error Interrupt Register (Channel 1)	0x0000
FEERR1	0xEB10_0052	W	Force Event Error Interrupt Register Error Interrupt (Channel 1)	0x0000
ADMAERR1	0xEB10_0054	R/W	ADMA Error Status Register (Channel 1)	0x00
ADMASYSADDR1	0xEB10_0058	R/W	ADMA System Address Register (Channel 1)	0x00
CONTROL2_1	0xEB10_0080	R/W	Control register 2 (Channel 1)	0x0
CONTROL3_1	0xEB10_0084	R/W	FIFO Interrupt Control (Control Register 3) (Channel 1)	0x7F5F3F1F
CONTROL4_1	0xEB10_008C	R/W	Control register 4 (Channel 1)	0x0
HCVER1	0xEB10_00FE	HWInit	Host Controller Version Register (Channel 1)	0x2401
SDMASYSAD2	0xEB20_0000	R/W	SDMA System Address register (Channel 2)	0x0
BLKSIZE2	0xEB20_0004	R/W	Host DMA Buffer Boundary and Transfer Block Size Register (Channel 2)	0x0
BLKCNT2	0xEB20_0006	R/W	Blocks count for current transfer (channel 2)	0x0
ARGUMENT2	0xEB20_0008	R/W	Command Argument Register (Channel 2)	0x0
TRNMOD2	0xEB20_000C	R/W	Transfer Mode Setting Register (Channel 2)	0x0
CMDREG2	0xEB20_000E	R/W	Command Register (Channel 2)	0x0
RSPREG0_2	0xEB20_0010	ROC	Response Register 0 (Channel 2)	0x0
RSPREG1_2	0xEB20_0014	ROC	Response Register 1 (Channel 2)	0x0
RSPREG2_2	0xEB20_0018	ROC	Response Register 2 (Channel 2)	0x0
RSPREG3_2	0xEB20_001C	ROC	Response Register 3 (Channel 2)	0x0
BDATA2	0xEB20_0020	R/W	Buffer Data Register (Channel 2)	0x0
PRNSTS2	0xEB20_0024	R/ROC	Present State Register (Channel 2)	0x000A0000
HOSTCTL2	0xEB20_0028	R/W	Present State Register (Channel 2)	0x0
PWRCON2	0xEB20_0029	R/W	Present State Register (Channel 2)	0x0
BLKGAP2	0xEB20_002A	R/W	Block Gap Control Register (Channel 2)	0x0
WAKCON2	0xEB20_002B	R/W	Wakeup Control Register (Channel 2)	0x0
CLKCON2	0xEB20_002C	R/W	Command Register (Channel 2)	0x0
TIMEOUTCON2	0xEB20_002E	R/W	Timeout Control Register (Channel 2)	0x0
SWRST2	0xEB20_002F	R/W	Software Reset Register (Channel 2)	0x0

Register	Address	R/W	Description	Reset Value
NORINTSTS2	0xEB20_0030	ROC/ RW1C	Normal Interrupt Status Register (Channel 2)	0x0
ERRINTSTS2	0xEB20_0032	ROC/ RW1C	Error Interrupt Status Register (Channel 2)	0x0
NORINTSTSEN2	0xEB20_0034	R/W	Normal Interrupt Status Enable Register (Channel 2)	0x0
ERRINTSTSEN2	0xEB20_0036	R/W	Error Interrupt Status Enable Register (Channel 2)	0x0
NORINTSIGEN2	0xEB20_0038	R/W	Normal Interrupt Signal Enable Register (Channel 2)	0x0
ERRINTSIGEN2	0xEB20_003A	R/W	Error Interrupt Signal Enable Register (Channel 2)	0x0
ACMD12ERRSTS2	0xEB20_003C	ROC	Auto CMD12 error status register (channel 2)	0x0
CAPAREG2	0xEB20_0040	HWInit	Capabilities Register (Channel 2)	0x05E80080
MAXCURR2	0xEB20_0048	HWInit	Maximum Current Capabilities Register (Channel 2)	0x0
FEAER2	0xEB20_0050	W	Force Event Auto CMD12 Error Interrupt Register (Channel 2)	0x0000
FEERR2	0xEB20_0052	W	Force Event Error Interrupt Register Error Interrupt (Channel 2)	0x0000
ADMAERR2	0xEB20_0054	R/W	ADMA Error Status Register (Channel 2)	0x00
ADMASYSADDR2	0xEB20_0058	R/W	ADMA System Address Register (Channel 2)	0x00
CONTROL2_2	0xEB20_0080	R/W	Control register 2 (Channel 2)	0x0
CONTROL3_2	0xEB20_0084	R/W	FIFO Interrupt Control (Control Register 3) (Channel 2)	0x7F5F3F1F
CONTROL4_2	0xEB20_008C	R/W	Control register 4 (Channel 2)	0x0
HCVER2	0xEB20_00FE	HWInit	Host Controller Version Register (Channel 2)	0x2401
SDMASYSAD3	0xEB30_0000	R/W	SDMA System Address register (Channel 3)	0x0
BLKSIZE3	0xEB30_0004	R/W	Host DMA Buffer Boundary and Transfer Block Size Register (Channel 3)	0x0
BLKCNT3	0xEB30_0006	R/W	Blocks count for current transfer (channel 3)	0x0
ARGUMENT3	0xEB30_0008	R/W	Command Argument Register (Channel 3)	0x0
TRNMOD3	0xEB30_000C	R/W	Transfer Mode Setting Register (Channel 3)	0x0
CMDREG3	0xEB30_000E	R/W	Command Register (Channel 3)	0x0
RSPREG0_3	0xEB30_0010	ROC	Response Register 0 (Channel 3)	0x0
RSPREG1_3	0xEB30_0014	ROC	Response Register 1 (Channel 3)	0x0
RSPREG2_3	0xEB30_0018	ROC	Response Register 2 (Channel 3)	0x0
RSPREG3_3	0xEB30_001C	ROC	Response Register 3 (Channel 3)	0x0
BDATA3	0xEB30_0020	R/W	Buffer Data Register (Channel 3)	0x0
PRNSTS3	0xEB30_0024	R/ROC	Present State Register (Channel 3)	0x000A0000

Register	Address	R/W	Description	Reset Value
HOSTCTL3	0xEB30_0028	R/W	Present State Register (Channel 3)	0x0
PWRCON3	0xEB30_0029	R/W	Present State Register (Channel 3)	0x0
BLKGAP3	0xEB30_002A	R/W	Block Gap Control Register (Channel 3)	0x0
WAKCON3	0xEB30_002B	R/W	Wakeup Control Register (Channel 3)	0x0
CLKCON3	0xEB30_002C	R/W	Command Register (Channel 3)	0x0
TIMEOUTCON3	0xEB30_002E	R/W	Timeout Control Register (Channel 3)	0x0
SWRST3	0xEB30_002F	R/W	Software Reset Register (Channel 3)	0x0
NORINTSTS3	0xEB30_0030	ROC/ RW1C	Normal Interrupt Status Register (Channel 3)	0x0
ERRINTSTS3	0xEB30_0032	ROC/ RW1C	Error Interrupt Status Register (Channel 3)	0x0
NORINTSTSEN3	0xEB30_0034	R/W	Normal Interrupt Status Enable Register (Channel 3)	0x0
ERRINTSTSEN3	0xEB30_0036	R/W	Error Interrupt Status Enable Register (Channel 3)	0x0
NORINTSIGEN3	0xEB30_0038	R/W	Normal Interrupt Signal Enable Register (Channel 3)	0x0
ERRINTSIGEN3	0xEB30_003A	R/W	Error Interrupt Signal Enable Register (Channel 3)	0x0
ACMD12ERRSTS3	0xEB30_003C	ROC	Auto CMD12 error status register (channel 3)	0x0
CAPAREG3	0xEB30_0040	HWInit	Capabilities Register (Channel 3)	0x05E80080
MAXCURR3	0xEB30_0048	HWInit	Maximum Current Capabilities Register (Channel 3)	0x0
FEAER3	0xEB30_0050	W	Force Event Auto CMD12 Error Interrupt Register (Channel 3)	0x0000
FEERR3	0xEB30_0052	W	Force Event Error Interrupt Register Error Interrupt (Channel 3)	0x0000
ADMAERR3	0xEB30_0054	R/W	ADMA Error Status Register (Channel 3)	0x00
ADMASYSADDR3	0xEB30_0058	R/W	ADMA System Address Register (Channel 3)	0x00
CONTROL2_3	0xEB30_0080	R/W	Control register 2 (Channel 3)	0x0
CONTROL3_3	0xEB30_0084	R/W	FIFO Interrupt Control (Control Register 3) (Channel 3)	0x7F5F3F1F
CONTROL4_3	0xEB30_008C	R/W	Control register 4 (Channel 3)	0x0
HCVER3	0xEB30_00FE	HWInit	Host Controller Version Register (Channel 3)	0x2401

7.9.2 SDMA SYSTEM ADDRESS REGISTER

7.9.2.1 SDMA System Address Register

- SDMASYSAD0, R/W, Address = 0xEB00_0000
- SDMASYSAD1, R/W, Address = 0xEB10_0000
- SDMASYSAD2, R/W, Address = 0xEB20_0000
- SDMASYSAD3, R/W, Address = 0xEB30_0000

This register contains the physical system memory address used for DMA transfers.

SDMASYSAD	Bit	Description	Initial State
SDMASYSAD	[31:0]	<p>SDMA System Address</p> <p>This register contains the system memory address for a DMA transfer. If the Host Controller stops a DMA transfer, this register points to the system address of the next contiguous data position. It is accessed if no transaction is in-progress (i.e., after a transaction has stopped). Read operations during transfers can return an invalid value.</p> <p>The Host Driver initializes this register before starting a DMA transaction. After DMA has stopped, the next system address of the next contiguous data position is read from this register.</p> <p>The DMA transfer waits at the every boundary specified by the Host SDMA Buffer Boundary in the Block Size register. The Host Controller generates DMA Interrupt to request the Host Driver to update this register. The Host Driver set the next system address of the next data position to this register. If the most upper byte of this register (003h) is written, the Host Controller restarts the DMA transfer. If restarting DMA by the Resume command or by setting Continue Request in the Block Gap Control register, the Host Controller starts at the next contiguous address stored here in the System Address register.</p>	0x00

7.9.3 BLOCK SIZE REGISTER

7.9.3.1 Host DMA Buffer Boundary and Transfer Block Size Register

- BLKSIZE0, R/W, Address = 0xEB00_0004
- BLKSIZE1, R/W, Address = 0xEB10_0004
- BLKSIZE2, R/W, Address = 0xEB20_0004
- BLKSIZE3, R/W, Address = 0xEB30_0004

This register is used to configure the number of bytes in a data block.

BLKSIZE	Bit	Description	Initial State
-	[15]	Reserved	0
BUF BOUND	[14:12]	<p>Host DMA Buffer Boundary</p> <p>The large contiguous memory space may not be available in the virtual memory system. To perform long SDMA transfer, System Address register is updated at every system memory boundary during SDMA transfer. These bits specify the size of contiguous buffer in the system memory. The SDMA transfer waits at the every boundary specified by these fields and the Host Controller generates the DMA Interrupt to request the Host Driver to update the SDMA System Address register. At the end of transfer, the Host Controller issue or not issue DMA Interrupt. In particular, DMA Interrupt is not issued after Transfer Complete Interrupt is issued.</p> <p>If this register is set to 0 (buffer size = 4K bytes), lower 12-bit of byte address points data in the contiguous buffer and the upper 20-bit points the location of the buffer in the system memory. The DMA transfer stops if the Host Controller detects carry out of the address from bit 11 to 12.</p> <p>These bits are supported if the SDMA Support in the Capabilities register is set to 1 and this function is active if DMA Enable in the Transfer Mode register is set to 1.</p> <p>000b = 4K bytes (Detects A11 carry out) 001b = 8K bytes (Detects A12 carry out) 010b = 16K Bytes (Detects A13 carry out) 011b = 32K Bytes (Detects A14 carry out) 100b = 64K bytes (Detects A15 carry out) 101b = 128K Bytes (Detects A16 carry out) 110b = 256K Bytes (Detects A17 carry out) 111b = 512K Bytes (Detects A18 carry out)</p>	0
BLKSIZE	[11:0]	<p>Transfer Block Size</p> <p>This register specifies the block size of data transfers for CMD17, CMD18, CMD24, CMD25, and CMD53. Values ranging from 1 up to maximum buffer size are set. In case of memory, it is set up to 512 bytes. It is accessed only if no transaction is in-progress (i.e., after a transaction has stopped). Read operations during transfers return an invalid value, and write operations are ignored.</p> <p>0200h = 512 Bytes , 01FFh = 511 Bytes</p> <p>.....</p> <p>0004h = 4 Bytes , 0003h = 3 Bytes 0002h = 2 Bytes , 0001h = 1 Byte 0000h = No data transfer</p>	0

7.9.4 BLOCK COUNT REGISTER

7.9.4.1 Blocks Count for Current Transfer

- BLKCNT0, R/W, Address = 0xEB00_0006
- BLKCNT1, R/W, Address = 0xEB10_0006
- BLKCNT2, R/W, Address = 0xEB20_0006
- BLKCNT3, R/W, Address = 0xEB30_0006

This register is used to configure the number of data blocks.

BLKCNT	Bit	Description	Initial State
BLKCNT	[15:0]	<p>Blocks Count For Current Transfer</p> <p>This register is enabled if Block Count Enable in the Transfer Mode register is set to 1 and is valid only for multiple block transfers. The Host Driver sets this register to a value between 1 and the maximum block count. The Host Controller decrements the block count after each block transfer and stops if the count reaches zero. Setting the block count to 0 results in no data blocks being transferred.</p> <p>This register must be accessed if no transaction is in-progress (i.e., after transactions are stopped). During data transfer, read operations on this register returns an invalid value and write operations are ignored. If saving transfer context as a result of a Suspend command, the number of blocks yet to be transferred is determined by reading this register. If restoring transfer context prior to issuing a Resume command, the Host Driver restores the previously saved block count.</p> <p>FFFFh = 65535 blocks ... 0002h = 2 blocks 0001h = 1 block 0000h = Stop Count</p>	0

7.9.5 ARGUMENT REGISTER

7.9.5.1 Command Argument Register

- ARGUMENT0, R/W, Address = 0xEB00_0008
- ARGUMENT1, R/W, Address = 0xEB10_0008
- ARGUMENT2, R/W, Address = 0xEB20_0008
- ARGUMENT3, R/W, Address = 0xEB30_0008

This register contains the SD Command Argument.

ARGUMENT	Bit	Description	Initial State
ARGUMENT	[31:0]	Command Argument The SD Command Argument is specified as bit [39:8] of Command-Format in the SD Memory Card Physical Layer Specification.	0

7.9.6 TRANSFER MODE REGISTER

7.9.6.1 Transfer Mode Register Setting

- TRNMOD0, R/W, Address = 0xEB00_000C
- TRNMOD1, R/W, Address = 0xEB10_000C
- TRNMOD2, R/W, Address = 0xEB20_000C
- TRNMOD3, R/W, Address = 0xEB30_000C

This register is used to control the data transfer operations. The Host Driver sets this register before issuing a command which transfers data (Refer to Data Present Select in the Command register (9.7)), or before issuing a Resume command. The Host Driver saves the value of this register if data transfer is suspended (as a result of a Suspend command) and restore it before issuing a Resume command. To prevent data loss, the Host Controller implements write protection for this register during data transactions. Writes to this register is ignored if the Command Inhibit (DAT) in Present State register is 1.

TRNMOD	Bit	Description	Initial State
Reserved	[15:14]	Reserved	0
BOOTACK	[13]	Boot ACK Receive Enable when Boot mode	0
BOOTCMD	[12]	Boot Command mode Enable Note: In boot mode, Do Not Enable "Auto CMD12 Enable"	0
Reserved	[11:10]	Reserved	0
CCSCON	[9:8]	Command Completion Signal Control '00' = No CCS Operation (Normal operation and No CE-ATA mode) '01' = Read or Write data transfer CCS enable (Only CE-ATA mode) '10' = Without data transfer CCS enable (Only CE-ATA mode) '11' = Abort Completion Signal (ACS) generation (Only CE-ATA mode)	0
Reserved	[7:6]	Reserved	0
MUL1SIN0	[5]	Multi/ Single Block Select This bit enables multiple block DAT line data transfers. For any other commands, this bit is set to 0. If this bit is 0, it is not mandatory to set the Block Count register. (Refer to the Table below "Determination of Transfer Type") 1 = Multiple Block 0 = Single Block	0
RD1WT0	[4]	Data Transfer Direction Select This bit defines the direction of DAT line data transfers. The bit is set to 1 by the Host Driver to transfer data from the SD card to the SD Host Controller and it is set to 0 for all other commands. 1 = Read (Card to Host) 0 = Write (Host to Card)	0
Reserved	[3]	Reserved	0

TRNMOD	Bit	Description	Initial State
ENACMD12	[2]	<p>Auto CMD12 Enable</p> <p>Multiple block transfers for memory require CMD12 to stop the transaction.</p> <p>If this bit is set to 1 and last block transfer is complete, the Host Controller issues CMD12 automatically. The Host Driver does not set this bit to issue commands that do not require CMD12 to stop data transfer.</p> <p>1 = Enable 0 = Disable</p>	0
ENBLKCNT	[1]	<p>Block Count Enable</p> <p>This bit is used to enable the Block Count register, which is only relevant for multiple block transfers. If this bit is 0, the Block Count register is disabled, which is useful in executing an infinite transfer. (Refer to the Table below "Determination of Transfer Type")</p> <p>1 = Enable 0 = Disable</p>	0
ENDMA	[0]	<p>DMA Enable</p> <p>This bit enables DMA functionality. DMA is enabled if it is supported as indicated in the DMA Support in the Capabilities register. If DMA is not supported, this bit is meaningless and always read 0. If this bit is set to 1, a DMA operation begins if the Host Driver writes to the upper byte of Command register (00Fh).</p> <p>1 = Enable 0 = Disable</p>	0

Table below shows the summary of how register settings determine types of data transfer.

Determination of Transfer Type

Multi/Single Block Select	Block Count Enable	Block Count	Function
0	Don't care	Don't care	Single Transfer
1	0	Don't care	Infinite Transfer
1	1	Not Zero	Multiple Transfer
1	1	Zero	Stop Multiple Transfer

NOTE: For CE-ATA access, (Auto) CMD12 must be issued after Command Completion Signal Disable.

Notification for Boot Mode Operation

NOTE:

1. In the boot mode, do not use "Auto Command 12 operation" and set "Response Type Select" field to 2'b00 (No Response).
2. This "Host Controller" do not support alternative boot operation using CMD0 with the argument of 0xFFFFFFFFFA
3. After boot code transfer is done, perform byte-write of address 0xD and set it to 0 so that CMD line goes back to HIGH (Make clear BOOTCMD, BOOTACK field). Then, wait for minimum of 56 SDCLK cycles as shown in the MMC 4.3 SPEC.
4. When BOOTACK is set, Host Controller needs to incur certain error interrupt if ACK pattern is not S-010-E. Currently, simply Data CRC Error occurs and thus, it is difficult to figure out ACK pattern is wrong.

7.9.7 COMMAND REGISTER

7.9.7.1 Command Register

- CMDREG0, R/W, Address = 0xEB00_000E
- CMDREG1, R/W, Address = 0xEB10_000E
- CMDREG2, R/W, Address = 0xEB20_000E
- CMDREG3, R/W, Address = 0xEB30_000E

This register contains the SD Command Argument.

The Host Driver checks the Command Inhibit (DAT) bit and Command Inhibit (CMD) bit in the Present State register before writing to this register. Writing to the upper byte of this register triggers SD command generation. The Host Driver is responsible to write this register because the Host Controller does not protect the writing if Command Inhibit (CMD) is set.

CMDREG	Bit	Description	Initial State
Reserved	[15:14]	Reserved	
CMDIDX	[13:8]	<p>Command Index</p> <p>These bits are set to the command number (CMD0-63, ACMD0-63) that is specified in bits 45-40 of the Command-Format in the SD Memory Card Physical Layer Specification and SDIO Card Specification.</p>	
CMDTYP	[7:6]	<p>Command Type</p> <p>There are three types of special commands: Suspend, Resume and Abort.</p> <ul style="list-style-type: none"> • Suspend Command <p>If the Suspend command succeeds, the Host Controller assumes that the SD Bus has been released and it is possible to issue the next command, which uses the DAT line. The Host Controller de-asserts Read Wait for read transactions and stops checking busy for write transactions. The interrupt cycle starts, in 4-bit mode. If the Suspend command fails, the Host Controller maintains its current state, and the Host Driver restarts the transfer by setting Continue Request in the Block Gap Control register.</p> <ul style="list-style-type: none"> • Resume Command <p>The Host Driver re-starts the data transfer by restoring the registers in the range of 000-00Dh (Refer to Suspend and Resume mechanism). The Host Controller checks for busy before starting write transfers.</p> <ul style="list-style-type: none"> • Abort Command <p>If this command is set when executing a read transfer, the Host Controller stops reads to the buffer. If this command is set while executing a write transfer, the Host Controller stops driving the DAT line. After issuing the Abort command, the Host Driver must issue a software reset (Refer to Abort Transaction (5)).</p> <p>11b = Abort CMD12, CMD52 for writing "I/O Abort" in CCCR 10b = Resume CMD52 for writing "Function Select" in CCCR 01b = Suspend CMD52 for writing "Bus Suspend" in CCCR</p>	

CMDREG	Bit	Description	Initial State
		00b = Normal Other commands	
DATAPRNT	[5]	<p>Data Present Select</p> <p>This bit is set to 1 to indicate that data is present and transferred using the DAT line. It is set to 0 for the following:</p> <ul style="list-style-type: none"> (1) Commands using only CMD line (ex. CMD52). (2) Commands with no data transfer but using busy signal on DAT[0] line (R1b or R5b ex. CMD38) (3) Resume command <p>1 = Data Present 0 = No Data Present</p>	
ENCMDIDX	[4]	<p>Command Index Check Enable</p> <p>If this bit is set to 1, the Host Controller checks the Index field in the response to see if it has the same value as the command index. If it is not, it is reported as a Command Index Error. If this bit is set to 0, the Index field is not checked.</p> <p>1 = Enable 0 = Disable</p>	
ENC MDCRC	[3]	<p>Command CRC Check Enable</p> <p>If this bit is set to 1, the Host Controller checks the CRC field in the response. If an error is detected, it is reported as a Command CRC Error. If this bit is set to 0, the CRC field is not checked. The number of bits checked by the CRC field value changes according to the length of the response.</p> <p>1 = Enable 0 = Disable</p>	
Reserved	[2]	Reserved	
RSPTYP	[1:0]	<p>Response Type Select</p> <p>00 = No Response 01 = Response Length 136 10 = Response Length 48 11 = Response Length 48 check Busy after response</p>	

Relation between Parameters and the Name of Response Type

Response Type	Index Check Enable	CRC Check Enable	Name of Response Type
00	0	0	No Response
01	0	1	R2
10	0	0	R3, R4
10	1	1	R1, R6, R5
11	1	1	R1b, R5b

These bits determine Response types.

NOTE:

1. In the SDIO specification, response type notation of R5b is not defined. R5 includes R5b in the SDIO specification. But R5b is defined in this specification to specify the Host Controller checks busy after receiving response. For example, usually CMD52 is used as R5 but I/O abort command is used as R5b.
2. For CMD52 to read BS after writing "Bus Suspend," Command Type must be "Suspend" as well.

7.10 RESPONSE REGISTER

This register is used to store responses from SD cards.

- Response Register 0 (Channel 0) (RSPREG0_0, ROC, Address = 0xEB00_0010)
- Response Register 1 (Channel 0) (RSPREG1_0, ROC, Address = 0xEB00_0014)
- Response Register 2 (Channel 0) (RSPREG2_0, ROC, Address = 0xEB00_0018)
- Response Register 3 (Channel 0) (RSPREG3_0, ROC, Address = 0xEB00_001C)

- Response Register 0 (Channel 1) (RSPREG0_1, ROC, Address = 0xEB10_0010)
- Response Register 1 (Channel 1) (RSPREG1_1, ROC, Address = 0xEB10_0014)
- Response Register 2 (Channel 1) (RSPREG2_1, ROC, Address = 0xEB10_0018)
- Response Register 3 (Channel 1) (RSPREG3_1, ROC, Address = 0xEB10_001C)

- Response Register 0 (Channel 2) (RSPREG0_2, ROC, Address = 0xEB20_0010)
- Response Register 1 (Channel 2) (RSPREG1_2, ROC, Address = 0xEB20_0014)
- Response Register 2 (Channel 2) (RSPREG2_2, ROC, Address = 0xEB20_0018)
- Response Register 3 (Channel 2) (RSPREG3_2, ROC, Address = 0xEB20_001C)

- Response Register 0 (Channel 3) (RSPREG0_3, ROC, Address = 0xEB30_0010)
- Response Register 1 (Channel 3) (RSPREG1_3, ROC, Address = 0xEB30_0014)
- Response Register 2 (Channel 3) (RSPREG2_3, ROC, Address = 0xEB30_0018)
- Response Register 3 (Channel 3) (RSPREG3_3, ROC, Address = 0xEB30_001C)

RSPREG	Bit	Description	Initial State
CMDRSP	[127:0]	<p>Command Response</p> <p>The Table below describes the mapping of command responses from the SD Bus to this register for each response type. In the table, R[] refers to a bit range within the response data as transmitted on the SD Bus, REP[] refers to a bit range within the Response register.</p> <p>128-bit Response bit order : {RSPREG3, RSPREG2, RSPREG1, RSPREG0}</p>	

7.10.1 RESPONSE BIT DEFINITION FOR EACH RESPONSE TYPE

Kind of Response	Meaning of Response	Response Field	Response Register
R1, R1b (normal response)	Card Status	R [39:8]	REP [31:0]
R1b (Auto CMD12 response)	Card Status for Auto CMD12	R [39:8]	REP [127:96]
R2 (CID, CSD register)	CID or CSD reg. incl.	R [127:8]	REP [119:0]
R3 (OCR register)	OCR register for memory	R [39:8]	REP [31:0]
R4 (OCR register)	OCR register for I/O etc	R [39:8]	REP [31:0]
R5,R5b	SDIO response	R [39:8]	REP [31:0]
R6 (Published RCA response)	New published RCA[31:16] etc	R [39:8]	REP [31:0]

The Response Field indicates bit positions of "Responses" defined in the PHYSICAL LAYER SPECIFICATION Version 1.01. The Table (above) shows that most responses with a length of 48 (R[47:0]) have 32 bits of the response data (R[39:8]) stored in the Response register at REP[31:0]. Responses of type R1b (Auto CMD12 responses) have response data bits R[39:8] stored in the Response register at REP[127:96]. Responses with length 136 (R[135:0]) have 120 bits of the response data (R[127:8]) stored in the Response register at REP[119:0].

To be able to read the response status efficiently, the Host Controller only stores part of the response data in the Response register. This enables the Host Driver to efficiently read 32 bits of response data in one read cycle on a 32-bit bus system. Parts of the response, the Index field and the CRC, are checked by the Host Controller (as specified by the Command Index Check Enable and the Command CRC Check Enable bits in the Command register) and generate an error interrupt if an error is detected. The bit range for the CRC check depends on the response length. If the response length is 48, the Host Controller checks R[47:1], and if the response length is 136 the Host Controller checks R[119:1].

Since the Host Controller may have a multiple block data DAT line transfer executing concurrently with a CMD_wo_DAT command, the Host Controller stores the Auto CMD12 response in the upper bits (REP[127:96]) of the Response register. The CMD_wo_DAT response is stored in REP[31:0]. This allows the Host Controller to avoid overwriting the Auto CMD12 response with the CMD_wo_DAT and vice versa.

If the Host Controller modifies part of the Response register, as shown in the Table above, it shall preserve the unmodified bits.

7.10.2 BUFFER DATA PORT REGISTER

7.10.2.1 Buffer Data Register

- BDATA0, R/W, Address = 0xEB00_0020
- BDATA1, R/W, Address = 0xEB10_0020
- BDATA2, R/W, Address = 0xEB20_0020
- BDATA3, R/W, Address = 0xEB30_0020

32-bit data port register to access internal buffer.

BDATA	Bit	Description	Initial State
BUFDAT	[31:0]	Buffer Data The Host Controller buffer is accessed through this 32-bit single port SRAM memory. Write and Read memories are separated.	Not fixed

NOTE: Detailed documents are to be copied from SD Host Standard Specification.

7.10.3 PRESENT STATE REGISTER

7.10.3.1 Present State Register

- PRNSTS0, R/ROC, Address = 0xEB00_0024
- PRNSTS1, R/ROC, Address = 0xEB10_0024
- PRNSTS2, R/ROC, Address = 0xEB20_0024
- PRNSTS3, R/ROC, Address = 0xEB30_0024

This register contains the SD Command Argument.

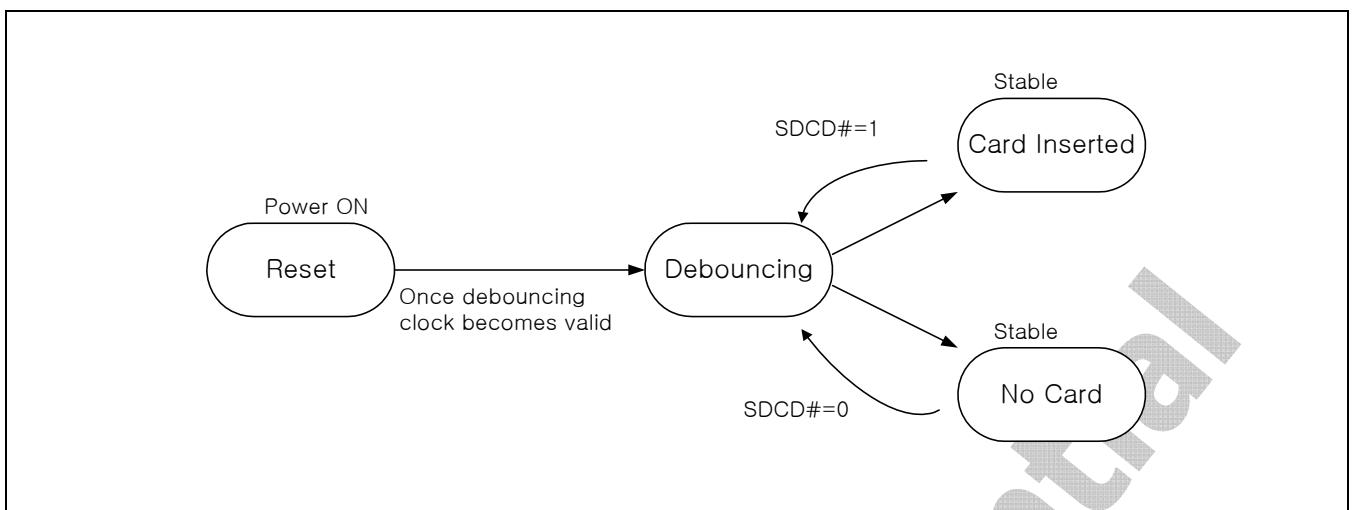
PRNSTS	Bit	Description	Initial State
Reserved	[31:25]	Reserved	0
PRNTCMD	[24]	CMD Line Signal Level (RO) This status is used to check the CMD line level to recover from errors, and for debugging. Note: CMD port is mapped to SD0_CMD pin	0
PRNTDAT	[23:20]	DAT[3:0] Line Signal Level (RO) This status is used to check the DAT line level to recover from errors, and for debugging. This is especially useful in detecting the busy signal level from DAT[0]. D23 : DAT[3] D22 : DAT[2] D21 : DAT[1] D20 : DAT[0] Note: DAT port is mapped to SD0_DAT pin	Line State
Reserved	[19]	Reserved	1
PRNTCD	[18]	Card Detect Pin Level (RO) This bit reflects the inverse value of the SDCD# pin. Debouncing is not performed on this bit. This bit is valid if Card State Stable is set to 1, but it is not guaranteed because of propagation delay. Use of this bit is limited to testing since it must be debounced by software. 1 = Card present (SDCD#=0) 0 = No card present (SDCD#=1) Note: SDCD# port is mapped to SD0_nCD pin, SD2_nCD (Channel 2) port is fixed to LOW.	Line State
STBLCARD	[17]	Card State Stable (RO) This bit is used for testing. If this bit is 0, the Card Detect Pin Level is not stable. If this bit is set to 1, it means the Card Detect Pin Level is stable. No Card state is detected by this bit if set to 1 and Card Inserted is set to 0. The Software Reset For All in the Software Reset register does not affect this bit. 1 = No Card or Inserted 0 = Reset or Debouncing	1 (After Reset)
INSCARD	[16]	Card Inserted (RO) This bit indicates whether a card has been inserted. The Host Controller debounce this signal so that the Host Driver does not require to wait for it to stabilize. Changing from 0 to 1 generates a	0

PRNSTS	Bit	Description	Initial State
		<p>Card Insertion interrupt in the Normal Interrupt Status register and changing from 1 to 0 generates a Card Removal interrupt in the Normal Interrupt Status register. The Software Reset For All in the Software Reset register does not affect this bit. If a card is removed while its power is on and its clock is oscillating, the Host Controller clears SD Bus Power in the Power Control register and SD Clock Enable in the Clock Control register.</p> <p>If this bit is changed from 1 to 0, the Host Controller immediately stops driving CMD and DAT[3:0] (tri-state). In addition, the Host Driver must clear the Host Controller by the Software Reset For All in Software Reset register. The card detect is active regardless of the SD Bus Power.</p> <p>1 = Card Inserted 0 = Reset or Debouncing or No Card</p>	
Reserved	[15:12]	Reserved	
BUFRDRDY	[11]	<p>Buffer Read Enable (ROC)</p> <p>This status is used for non-DMA read transfers. The Host Controller implements multiple buffers to transfer data efficiently. This read only flag indicates that valid data exists in the host side buffer status. If this bit is 1, readable data exists in the buffer. A change of this bit from 1 to 0 occurs if all the block data is read from the buffer. A change of this bit from 0 to 1 occurs if block data is ready in the buffer and generates the Buffer Read Ready interrupt.</p> <p>1 = Enables Read 0 = Disables Read</p>	0
BUFWTRDY	[10]	<p>Buffer Write Enable (ROC)</p> <p>This status is used for non-DMA write transfers. The Host Controller implements multiple buffers to transfer data efficiently. This read only flag indicates if space is available for write data. If this bit is 1, data is written to the buffer. A change of this bit from 1 to 0 occurs if all the block data is written to the buffer. A change of this bit from 0 to 1 occurs if top of block data is written to the buffer and generates the Buffer Write Ready interrupt.</p> <p>1 = Write enable 0 = Write disable</p>	0
RDTRANACT	[9]	<p>Read Transfer Active (ROC)</p> <p>This status is used to detect completion of a read transfer. This bit is set to 1 for either of the following conditions:</p> <ul style="list-style-type: none"> (1) After the end bit of the read command. (2) If writing a 1 to Continue Request in the Block Gap Control register to restart a read transfer. <p>This bit is cleared to 0 for either of the following conditions:</p> <ul style="list-style-type: none"> (1) If the last data block as specified by block length is transferred to the System. (2) If all valid data blocks have been transferred to the System and no current block transfers are being sent as a result of the Stop At Block Gap Request being set to 1. A Transfer Complete interrupt is generated if this bit changes to 0. 	0

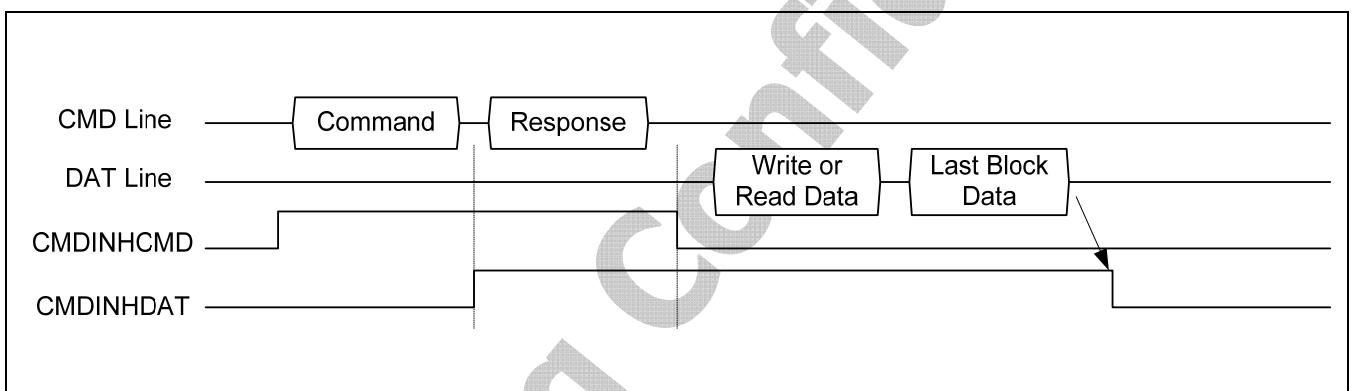
PRNSTS	Bit	Description	Initial State
		1 = Transferring data 0 = No valid data	
WTTRANACT	[8]	<p>Write Transfer Active (ROC)</p> <p>This status indicates that a write transfer is active. If this bit is 0, it means no valid write data exists in the Host Controller.</p> <p>This bit is set in either of the following cases:</p> <ul style="list-style-type: none"> (1) After the end bit of the write command. (2) If 1 is written to Continue Request in the Block Gap Control register to restart a write transfer. <p>This bit is cleared in either of the following cases:</p> <ul style="list-style-type: none"> (1) After getting the CRC status of the last data block as specified by the transfer count (Single and Multiple) (2) After getting the CRC status of any block where data transmission is about to be stopped by a Stop At Block Gap Request. <p>During a write transaction, if this bit is changed to 0 a Block Gap Event interrupt is generated, as result of the Stop At Block Gap Request being set. This status is useful for the Host Driver to determine the right time to issue commands during write busy.</p> <p>1 = Transferring data 0 = No valid data</p>	0
Reserved	[7:3]	Reserved	0
DATLINEACT	[2]	<p>DAT Line Active (ROC)</p> <p>This bit indicates whether one of the DAT line on SD Bus is in use.</p> <p>(a) In the case of read transactions</p> <p>This status indicates if a read transfer is In-progress on the SD Bus. Change in this value from 1 to 0 between data blocks generates a Block Gap Event interrupt in the Normal Interrupt Status register.</p> <p>This bit is set in either of the following cases:</p> <ul style="list-style-type: none"> (1) After the end bit of the read command. (2) If 1 is written to Continue Request in the Block Gap Control register to restart a read transfer. <p>This bit is cleared in either of the following cases:</p> <ul style="list-style-type: none"> (1) If the end bit of the last data block is sent from the SD Bus to the Host Controller. (2) When beginning a wait read transfer at a stop at the block gap initiated by a Stop At Block Gap Request. <p>The Host Controller waits at the next block gap by driving Read Wait at the start of the interrupt cycle. If the Read Wait signal is already driven (data buffer cannot receive data), the Host Controller waits for current block gap by continuing to drive the Read Wait signal. It is necessary to support Read Wait in order to use suspend/ resume function.</p> <p>(b) In the case of write transactions</p> <p>This status indicates that a write transfer is executing on the SD Bus. Change in this value from 1 to 0 generates a Transfer Complete interrupt in the Normal Interrupt Status register.</p>	0

PRNSTS	Bit	Description	Initial State
		<p>This bit is set in either of the following cases:</p> <ul style="list-style-type: none"> (1) After the end bit of the write command. (2) If 1 is written to Continue Request in the Block Gap Control register to continue a write transfer. <p>This bit is cleared in either of the following cases:</p> <ul style="list-style-type: none"> (1) If the SD card releases write busy of the last data block the Host Controller detects if output is not busy. If SD card does not drive busy signal for 8 SD Clocks, the Host Controller considers the card drive "Not Busy". (2) If the SD card releases write busy prior to waiting for write transfer as a result of a Stop At Block Gap Request. <p>1 = DAT Line Active 0 = DAT Line Inactive</p>	
CMDINHDAT	[1]	<p>Command Inhibit (DAT) (ROC)</p> <p>(ROC)</p> <p>This status bit is generated if either the DAT Line Active or the Read Transfer Active is set to 1. If this bit is 0, it indicates the Host Controller can issue the next SD Command. Commands with busy signal belong to Command Inhibit (DAT) (ex. R1b, R5b type).</p> <p>Changing from 1 to 0 generates a Transfer Complete interrupt in the Normal Interrupt Status register.</p> <p>Note: The SD Host Driver saves registers in the range of 000-00Dh for a suspend transaction after this bit has changed from 1 to 0.</p> <p>1 = Cannot issue command which uses the DAT line 0 = Issues command which uses the DAT line</p>	0
CMDINHCMD	[0]	<p>Command Inhibit (CMD) (ROC)</p> <p>If this bit is 0, it indicates the CMD line is not in use and the Host Controller issues a SD Command using the CMD line.</p> <p>This bit is set immediately after the Command register (00Fh) is written. This bit is cleared if the command response is received.</p> <p>Even if the Command Inhibit (DAT) is set to 1, Commands using only the CMD line is issued if this bit is 0. Changing from 1 to 0 generates a Command Complete interrupt in the Normal Interrupt Status register. If the Host Controller cannot issue the command because of a command conflict error (Refer to Command CRC Error) or because of Command Not Issued By Auto CMD12 Error, this bit remains 1 and the Command Complete is not set. Status issuing Auto CMD12 is not read from this bit.</p> <p>1 = Cannot issue command 0 = Issues command using only CMD line</p>	0

NOTE: Buffer Write Enable in Present register must not be asserted for DMA transfers since it generates Buffer Write Ready interrupt.

**Figure 7-17 Card Detect State**

The above [Figure 7-17](#) shows the state definitions of hardware that handles "Debouncing"

**Figure 7-18 Timing of Command Inhibit (DAT) and Command Inhibit (CMD) with Data Transfer**

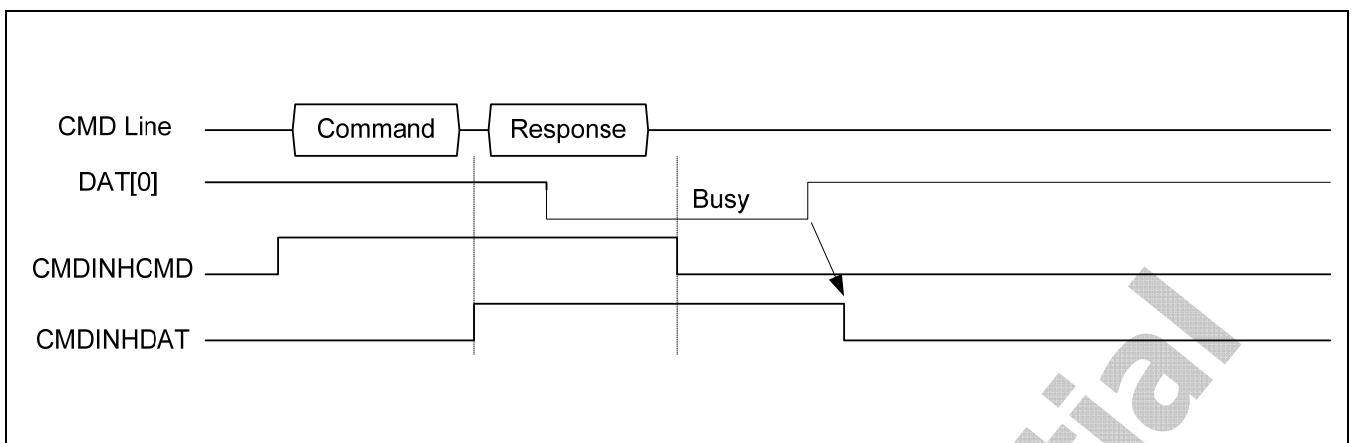


Figure 7-19 Timing of Command Inhibit (DAT) for the Case of Response with Busy

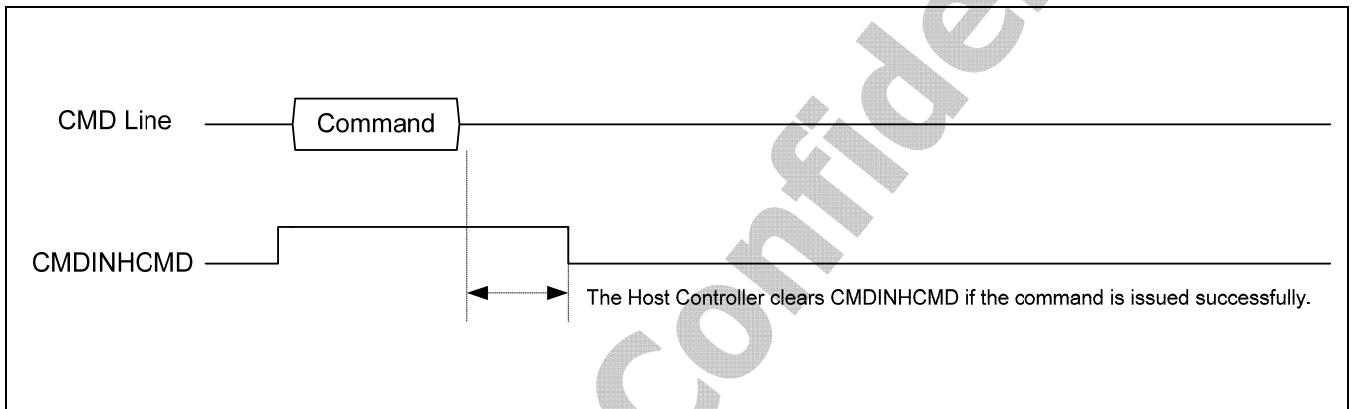


Figure 7-20 Timing of Command Inhibit (CMD) for the Case of No Response Command

7.10.4 HOST CONTROL REGISTER

7.10.4.1 Present State Register

- HOSTCTL0, R/W, Address = 0xEB00_0028
- HOSTCTL1, R/W, Address = 0xEB10_0028
- HOSTCTL2, R/W, Address = 0xEB20_0028
- HOSTCTL3, R/W, Address = 0xEB30_0028

This register contains the SD Command Argument.

HOSTCTL	Bit	Description	Initial State
Reserved	[7]	Reserved This field should be fixed to LOW	0
Reserved	[6]	Reserved This field should be fixed to LOW	0
WIDE8	[5]	Extended Data Transfer Width (It is for MMC 8-bit card.) 1 = 8-bit operation 0 = Bit width is designated by the bit 1 (Data Transfer Width)	0
DMASEL	[4:3]	DMA Select One of supported DMA modes can be selected. The host driver checks support of DMA modes by referring the Capabilities register. Use of selected DMA is determined by DMA Enable of the Transfer Mode register. 00 = Selects SDMA 01 = Reserved 10 = Selects 32-bit Address ADMA2 11 = Selects 64-bit Address ADMA2 (Not supported)	0
OUTEDGEINV	[2]	Output Edge Inversion. If this bit is set to 0 (default), the Host Controller outputs CMD line and DAT lines at the falling edge of the SD Clock. If this bit is set to 1, the Host Controller outputs CMD line and DAT lines at the rising edge of the SD Clock 1 = Rising edge output 0 = Falling edge output	0
WIDE4	[1]	Data Transfer Width This bit selects the data width of the Host Controller. The Host Driver sets it to match the data width of the SD card. 1 = 4-bit mode 0 = 1-bit mode	0
Reserved	[0]	Reserved	0

NOTE: Card Detect Pin Level does not simply reflect SDCD# pin, but selects from SDCD, DAT[3], or CDTestlvl depending on CDSSigSel and SDCDSel values.

7.10.5 POWER CONTROL REGISTER

This function is not implemented in this version.

7.10.5.1 Present State Register

- PWRCON0, R/W, Address = 0xEB00_0029
- PWRCON1, R/W, Address = 0xEB10_0029
- PWRCON2, R/W, Address = 0xEB20_0029
- PWRCON3, R/W, Address = 0xEB30_0029

This register contains the SD Command Argument.

PWRCON	Bit	Description	Initial State
Reserved	[7:4]	Reserved	
SELPWRLVL	[3:1]	<p>SD Bus Voltage Select</p> <p>If these bits are set, the Host Driver selects the voltage level for the SD card. Before setting this register, the Host Driver checks the Voltage Support bits in the Capabilities register. If an unsupported voltage is selected, the Host System does not supply SD Bus voltage.</p> <p>111b = 3.3V (Typ.) 110b = 3.0V (Typ.) 101b = 1.8V (Typ.) 100b – 000b = Reserved</p>	0
PWRON	[0]	<p>SD Bus Power</p> <p>Before setting this bit, the SD Host Driver sets SD Bus Voltage Select. If the Host Controller detects the No Card state, this bit is cleared.</p> <p>If this bit is cleared, the Host Controller immediately stops driving CMD and DAT[3:0] (tri-state) and drive SDCLK to low level.</p> <p>1 = Power on 0 = Power off</p>	0

7.10.6 BLOCK GAP CONTROL REGISTER

7.10.6.1 Block Gap Control Register

- BLKGAP0, R/W, 0xEB00_002A
- BLKGAP1, R/W, 0xEB10_002A
- BLKGAP2, R/W, 0xEB20_002A
- BLKGAP3, R/W, 0xEB30_002A

This register contains the SD Command Argument.

BLKGAP	Bit	Description	Initial State
Reserved	[7:4]	Reserved	0
ENINTBGAP	[3]	<p>Interrupt At Block Gap</p> <p>This bit is valid only in 4-bit mode of the SDIO card and selects a sample point in the interrupt cycle. If set to 1, it enables interrupt detection at the block gap for a multiple block transfer. If set to 0, it disables interrupt detection during a multiple block transfer. If the SD card cannot signal an interrupt during a multiple block transfer, this bit must be set to 0. If the Host Driver detects an SD card insertion, it sets this bit according to the CCCR of the SDIO card. (RW)</p> <p>1 = Enables 0 = Disables</p> <p>Note: it should be fixed to 0.</p>	0
ENRWAIT	[2]	<p>Read Wait Control</p> <p>The read wait function is optional for SDIO cards. If the card supports read wait, set this bit to enable use of the read wait protocol to stop read data using the DAT[2] line. Otherwise the Host Controller has to stop the SD Clock to hold read data, which restricts commands generation. If the Host Driver detects an SD card insertion, it sets this bit according to the CCCR of the SDIO card. If the card does not support read wait, this bit will never be set to 1 otherwise DAT line conflict might occur. If this bit is set to 0, Suspend/ Resume cannot be supported. (RW)</p> <p>1 = Enables Read Wait Control 0 = Disables Read Wait Control</p>	0
CONTREQ	[1]	<p>Continue Request</p> <p>This bit is used to restart a transaction which was stopped using the Stop At Block Gap Request. To cancel stop at the block gap, set Stop At Block Gap Request to 0 and set this bit 1 to restart the transfer.</p> <p>The Host Controller automatically clears this bit in either of the following cases:</p> <p>(1) If a read transaction, the DAT Line Active changes from 0 to 1 as a read transaction restarts. (2) If a write transaction, the Write Transfer Active changes from 0 to 1 as the write transaction restarts.</p> <p>Therefore it is not necessary for Host Driver to set this bit to 0. If</p>	0

BLKGAP	Bit	Description	Initial State
		Stop At Block Gap Request is set to 1, any write to this bit is ignored. (RWAC) 1 = Restart 0 = Not affect	
STOPBGAP	[0]	Stop At Block Gap Request This bit is used to stop executing a transaction at the next block gap for both DMA and non-DMA transfers. Until the Transfer Complete is set to 1, indicating a transfer completion the Host Driver leaves this bit as 1. Clearing both the Stop At Block Gap Request and Continue Request does not restart the transaction. Read Wait stops the read transaction at the block gap. The Host Controller honours Stop At Block Gap Request for write transfers, but for read transfers it requires that the SD card support Read Wait. Therefore the Host Driver does not set this bit during read transfers unless the SD card supports Read Wait and has set Read Wait Control to 1. In the case of write transfers in which the Host Driver writes data to the Buffer Data Port register, the Host Driver sets this bit after all block data is written. If this bit is set to 1, the Host Driver does not write data to Buffer Data Port register. This bit affects Read Transfer Active, Write Transfer Active, DAT Line Active and Command Inhibit (DAT) in the Present State register. Regarding detailed control of bits D01 and D00. (RW) 1 = Stop 0 = Transfer	0

There are three cases to restart the transfer after stop at the block gap. Appropriate case depends on whether the Host Controller issues a Suspend command or the SD card accepts the Suspend command.

Cases are as follows:

1. If the Host Driver does not issue a Suspend command, the Continue Request restarts the transfer.
2. If the Host Driver issues a Suspend command and the SD card accepts it, a Resume command restarts the transfer.
3. If the Host Driver issues a Suspend command and the SD card does not accept it, the Continue Request restarts the transfer.

Any time Stop At Block Gap Request stops the data transfer, the Host Driver waits for Transfer Complete (in the Normal Interrupt Status register) before attempting to restart the transfer. If the data transfer by Continue Request is restarted, the Host Driver clears Stop At Block Gap Request before or simultaneously.

NOTE: After setting Stop At Block Gap Request field, it must not be cleared unless Block Gap Event or Transfer Complete interrupt occurs. Otherwise, the module hangs.

7.10.7 WAKEUP CONTROL REGISTER

7.10.7.1 Wakeup Control Register

- WAKCON0, R/W, Address = 0xEB00_002B
- WAKCON1, R/W, Address = 0xEB10_002B
- WAKCON2, R/W, Address = 0xEB20_002B
- WAKCON3, R/W, Address = 0xEB30_002B

This register is mandatory for the Host Controller, but wakeup functionality depends on the Host Controller system hardware and software. The Host Driver sets SD Bus Power to 1 in the Power Control Register to maintains voltage on the SD Bus, if wakeup event via Card Interrupt is desired.

WAKCON	Bit	Description	Initial State
Reserved	[7:4]	Reserved	0
StaWakeup	[3]	Wakeup Event Status This status is set if the Card Inserted/ Removed or Card Interrupt Stop mode Wakeup Event Occurred. (ROC/RW1C) 1 = Wakeup Interrupt Occurred 0 = Wakeup Interrupt Not occurred or Cleared.	0
ENWKUPREM	[2]	Wakeup Event Enable On SD Card Removal This bit enables wakeup event via Card Removal assertion in the Normal Interrupt Status register. FN_WUS (Wake Up Support) in CIS does not affect this bit. (RW) 1 = Enables 0 = Disables	0
ENWKUPINS	[1]	Wakeup Event Enable On SD Card Insertion This bit enables wakeup event via Card Insertion assertion in the Normal Interrupt Status register. FN_WUS (Wake Up Support) in CIS does not affect this bit. (RW) 1 = Enables 0 = Disable	0
ENWKUPINT	[0]	Wakeup Event Enable On Card Interrupt This bit enables wakeup event via Card Interrupt assertion in the Normal Interrupt Status register. This bit is set to 1 if FN_WUS (Wake Up Support) in CIS is set to 1. (RW) 1 = Enables 0 = Disables	0

7.10.8 CLOCK CONTROL REGISTER

7.10.8.1 Command Register

- CLKCON0, R/W, Address = 0xEB00_002C
- CLKCON1, R/W, Address = 0xEB10_002C
- CLKCON2, R/W, Address = 0xEB20_002C
- CLKCON3, R/W, Address = 0xEB30_002C

At the initialization of the Host Controller, the Host Driver sets the SDCLK Frequency Select according to the Capabilities register.

CLKCON	Bit	Description	Initial State																		
SELFREQ	[15:8]	<p>SDCLK Frequency Select This register is used to select the frequency of SDCLK pin. The frequency is not programmed directly; rather this register holds the divisor of the Base Clock Frequency For SD Clock in the Capabilities register. Only the following settings are allowed.</p> <table border="1"> <tr><td>80h</td><td>base clock divided by 256</td></tr> <tr><td>40h</td><td>base clock divided by 128</td></tr> <tr><td>20h</td><td>base clock divided by 64</td></tr> <tr><td>10h</td><td>base clock divided by 32</td></tr> <tr><td>08h</td><td>base clock divided by 16</td></tr> <tr><td>04h</td><td>base clock divided by 8</td></tr> <tr><td>02h</td><td>base clock divided by 4</td></tr> <tr><td>01h</td><td>base clock divided by 2</td></tr> <tr><td>00h</td><td>base clock</td></tr> </table> <p>Setting 00h specifies the highest frequency of the SD Clock. Setting multiple bits, the most significant bit is used as the divisor. But multiple bits must not be set. The two default divider values are calculated by the frequency that is defined by the Base Clock Frequency For SD Clock in the Capabilities register. (1) 25MHz divider value, (2) 400kHz divider value According to the SD Physical Specification Version 1.01 and the SDIO Card Specification Version 1.0, maximum SD Clock frequency is 25MHz, and never exceeds this limit. The frequency of SDCLK is set by the following formula: Clock Frequency = (Base Clock) / divisor Therefore, select the smallest possible divisor which results in a clock frequency that is less than or equal to the target frequency. For example, if the Base Clock Frequency For SD Clock in the Capabilities register has the value 33MHz, and the target frequency is 25MHz, then selecting the divisor value of 01h yields 16.5MHz, which is the nearest frequency less than or equal to the target. Similarly, to approach a clock value of 400kHz, the divisor value of 40h yields the optimal clock value of</p>	80h	base clock divided by 256	40h	base clock divided by 128	20h	base clock divided by 64	10h	base clock divided by 32	08h	base clock divided by 16	04h	base clock divided by 8	02h	base clock divided by 4	01h	base clock divided by 2	00h	base clock	0
80h	base clock divided by 256																				
40h	base clock divided by 128																				
20h	base clock divided by 64																				
10h	base clock divided by 32																				
08h	base clock divided by 16																				
04h	base clock divided by 8																				
02h	base clock divided by 4																				
01h	base clock divided by 2																				
00h	base clock																				

CLKCON	Bit	Description	Initial State
		258kHz. (Base input clock should not exceed 102MHz.)	
Reserved	[7:4]	Reserved	
STBLEXTCLK	[3]	<p>External Clock Stable</p> <p>This bit is set to 1 if SD Clock output is stable after writing to SD Clock Enable in this register to 1. The SD Host Driver waits to issue command to start until this bit is set to 1. (ROC)</p> <p>1 = Ready 0 = Not Ready</p>	0
ENSDCLK	[2]	<p>SD Clock Enable</p> <p>The Host Controller stops SDCLK if this bit is written to 0. SDCLK Frequency Select changes if this bit is 0. Then, the Host Controller maintains the same clock frequency until SDCLK is stopped (Stop at SDCLK=0). If the Card Inserted in the Present State register is cleared, this clears the bit (RW).</p> <p>1 = Enables 0 = Disables</p>	0
STBLINTCLK	[1]	<p>Internal Clock Stable</p> <p>This bit is set to 1 if SD Clock is stable after writing to Internal Clock Enable in this register to 1. The SD Host Driver waits to set SD Clock Enable until this bit is set to 1.</p> <p>Note: This is useful if PLL is used for a clock oscillator that requires setup time. (ROC)</p> <p>1 = Ready 0 = Not Ready</p>	0
ENINTCLK	[0]	<p>Internal Clock Enable</p> <p>This bit is set to 0 if the Host Driver is not using the Host Controller or the Host Controller awaits a wakeup interrupt. The Host Controller must stop its internal clock to go at very low power state. Still, registers is able to be read and written. Clock starts to oscillate when this bit is set to 1. If clock oscillation is stable, the Host Controller can be set 'Internal Clock Stable (CLKCON)' when this bit as 1. This bit does not affect card detection. (RW)</p> <p>1 = Oscillate 0 = Stop</p>	

7.10.9 TIMEOUT CONTROL REGISTER

7.10.9.1 Timeout Control Register

- TIMEOUTCON0, R/W, Address = 0xEB00_002E
- TIMEOUTCON1, R/W, Address = 0xEB10_002E
- TIMEOUTCON2, R/W, Address = 0xEB20_002E
- TIMEOUTCON3, R/W, Address = 0xEB30_002E

At the initialization of the Host Controller, the Host Driver sets the Data Timeout Counter Value according to the Capabilities register.

TIMEOUTCON	Bit	Description	Initial State
Reserved	[7:4]	Reserved	0
TIMEOUTCON	[3:0]	Data Timeout Counter Value This value determines the interval by which DAT line timeouts are detected. Refer to the Data Timeout Error in the Error Interrupt Status register for information on factors that dictate timeout generation. Timeout clock frequency is generated by dividing the base clock TMCLK value by this value. While setting this register, prevent inadvertent timeout events by clearing the Data Timeout Error Status Enable (in the Error Interrupt status Enable register) 1111b Reserved 1110b TMCLK x 2^27 1101b TMCLK x 2^26 0001b TMCLK x 2^14 0000b TMCLK x 2^13	0

7.10.10 SOFTWARE RESET REGISTER

7.10.10.1 Software Reset Register

- SWRST0, R/W, Address = 0xEB00_002F
- SWRST1, R/W, Address = 0xEB10_002F
- SWRST2, R/W, Address = 0xEB20_002F
- SWRST3, R/W, Address = 0xEB30_002F

A reset pulse is generated when writing 1 to each bit of this register. After completing the reset, the Host Controller clears each bit. Because it takes time to complete software reset, the SD Host Driver confirms that these bits are 0.

SWRST	Bit	Description	Initial State
Reserved	[7:3]	Reserved	0
RSTDAT	[2]	Software Reset For DAT Line Only part of data circuit is reset. DMA circuit is also reset. (RWAC) The following registers and bits are cleared by this bit: Buffer Data Port register Buffer is cleared and initialized. Present State register Buffer Read Enable Buffer Write Enable Read Transfer Active Write Transfer Active DAT Line Active Command Inhibit (DAT) Block Gap Control register Continue Request Stop At Block Gap Request Normal Interrupt Status register Buffer Read Ready Buffer Write Ready DMA Interrupt Block Gap Event Transfer Complete 1 = Reset 0 = Work	0
RSTCMD	[1]	Software Reset For CMD Line Only part of command circuit is reset. (RWAC). The following registers and bits are cleared by this bit: Present State register Command Inhibit (CMD) Normal Interrupt Status register Command Complete 1 = Reset 0 = Work	0
RSTALL	[0]	Software Reset For All This reset affects the entire Host Controller except for the card	0

SWRST	Bit	Description	Initial State
		<p>detection circuit. Register bits of type ROC, RW, RW1C, RWAC, HWInit are cleared to 0. During its initialization, the Host Driver sets this bit to 1 to reset the Host Controller. The Host Controller reset this bit to 0 if capabilities registers are valid and the Host Driver reads them. If this bit is set to 1, the SD card resets itself and must be reinitialized by the Host Driver. (RWAC)</p> <p>1 = Reset 0 = Work</p>	

7.10.11 NORMAL INTERRUPT STATUS REGISTER

7.10.11.1 Normal Interrupt Status Register

- NORINTSTS0, ROC/RW1C, Address = 0xEB00_0030
- NORINTSTS1, ROC/RW1C, Address = 0xEB10_0030
- NORINTSTS2, ROC/RW1C, Address = 0xEB20_0030
- NORINTSTS3, ROC/RW1C, Address = 0xEB30_0030

The Normal Interrupt Status Enable affects reads of this register, but Normal Interrupt Signal Enable does not affect these reads. An interrupt is generated if the Normal Interrupt Signal Enable is enabled and at least one of the status bits is set to 1. For all bits except Card Interrupt and Error Interrupt, writing 1 to a bit clears it; writing 0 keeps the bit unchanged. More than one status is cleared with a single register write. The Card Interrupt is cleared if the card stops asserting the interrupt; that is, if the Card Driver services the interrupt condition.

NORINTSTS	Bit	Description	Initial State
STAERR	[15]	Error Interrupt If any of the bits in the Error Interrupt Status register are set, then this bit is set. Therefore the Host Driver checks this bit first to efficiently tests for an error. This bit is read only. (ROC) 0 = No Error 1 = Error	0
STAFIA3	[14]	FIFO SD Address Pointer Interrupt 3 Status (RW1C) 0 = Occurred 1 = Not Occurred If the FIFO Address of the SD clock side reaches the FIFO Interrupt Address register 3 values, this status bit is asserted.	0
STAFIA2	[13]	FIFO SD Address Pointer Interrupt 2 Status (RW1C) 0 = Occurred 1 = Not Occurred If the FIFO Address of the SD clock side reaches the FIFO Interrupt Address register 2 values, this status bit is asserted.	0
STAFIA1	[12]	FIFO SD Address Pointer Interrupt 1 Status (RW1C) 0 = Occurred 1 = Not Occurred If the FIFO Address of the SD clock side reaches the FIFO Interrupt Address register 1 value, this status bit is asserted.	0
STAFIA0	[11]	FIFO SD Address Pointer Interrupt 0 Status (RW1C) 0 = Occurred 1 = Not Occurred If the FIFO Address of the SD clock side reaches the FIFO Interrupt Address register 0 value, this status bit is asserted.	0
STARWAIT	[10]	Read Wait Interrupt Status (RW1C) 0 = Read Wait Interrupt Not Occurred 1 = Read Wait Interrupt Occurred Note: After checking response for the suspend command, release Read Wait interrupt status manually if BS = 0 (BS means 'Bus Status' field 'Bus Suspend' register in the SDIO card)	0

NORINTSTS	Bit	Description	Initial State
		<p>specification)</p> <p>Note: Read Wait operation procedure is started after 4-SDCLK from the end of the block data read transfer.</p>	
STACCS	[9]	<p>CCS Interrupt Status (RW1C)</p> <p>Command Complete Signal Interrupt Status bit is for CE-ATA interface mode.</p> <p>0 = CCS Interrupt Occurred 1 = CCS Interrupt Not Occurred</p>	0
STACARDINT	[8]	<p>Card Interrupt</p> <p>Writing this bit to 1 does not clear this bit. It is cleared by resetting the SD card interrupt factor. In 1-bit mode, the Host Controller detects the Card Interrupt without SD Clock to support wakeup. In 4-bit mode, the card interrupt signal is sampled during the interrupt cycle, therefore there are some sample delays between the interrupt signal from the SD card and the interrupt to the Host System. It is necessary to define how to handle this delay.</p> <p>If this status is set and the Host Driver needs to start this interrupt service, Card Interrupt Signal Enable in the Normal Interrupt Signal Enable register must be set to 0 in order to clear the card interrupt status latched in the Host Controller and to stop driving the interrupt signal to the Host System. After completion of the card interrupt service (It must reset interrupt factors in the SD card and the interrupt signal may not be asserted), write 1 to clear this register field (RW1C) and set Card Interrupt Signal Enable to 1 to re-start sampling the interrupt signal. The Card Interrupt Status Enable must remain set to high. (RW1C) (2), (3)</p> <p>1 = Generates Card Interrupt 0 = No Card Interrupt</p>	0
STACARDREM	[7]	<p>Card Removal</p> <p>This status is set if the Card Inserted in the Present State register changes from 1 to 0. If the Host Driver writes this bit to 1 to clear this status, the status of the Card Inserted in the Present State register must be confirmed. Because the card detect state may possibly be changed if the Host Driver clear this bit and interrupt event may not be generated. (RW1C)</p> <p>1 = Card removed 0 = Card state stable or Debouncing</p>	0
STACARDINS	[6]	<p>Card Insertion</p> <p>This status is set if the Card Inserted in the Present State register changes from 0 to 1. If the Host Driver writes this bit to 1 to clear this status, the status of the Card Inserted in the Present State register must be confirmed. Because the card detect state may possibly be changed if the Host Driver clear this bit and interrupt event may not be generated. (RW1C)</p> <p>1 = Card inserted 0 = Card state stable or Debouncing</p>	0

NORINTSTS	Bit	Description	Initial State
STABUFRDRDY	[5]	<p>Buffer Read Ready</p> <p>This status is set if the Buffer Read Enable changes from 0 to 1. Refer to the Buffer Read Enable in the Present State register (9.10). (RW1C)</p> <p>1 = Ready to read buffer 0 = Not ready to read buffer</p>	0
STABUFWTRDY	[4]	<p>Buffer Write Ready</p> <p>This status is set if the Buffer Write Enable changes from 0 to 1. Refer to the Buffer Write Enable in the Present State register (9.10). (RW1C)</p> <p>1 = Ready to write buffer 0 = Not ready to write buffer</p>	0
STADMAINT	[3]	<p>DMA Interrupt</p> <p>This status is set if the Host Controller detects the Host SDMA Buffer boundary during transfer. Refer to the Host SDMA Buffer Boundary in the Block Size register (9.3). Other DMA interrupt factors may be added in the future.</p> <p>In case of ADMA, by setting interrupt field in the descriptor table, Host Controller generates this interrupt. If it is used for debugging. This interrupt is not generated after the Transfer Complete. (RW1C)</p> <p>1 = Generates DMA Interrupt 0 = No DMA Interrupt</p>	0
STABLKGAP	[2]	<p>Block Gap Event</p> <p>If the Stop At Block Gap Request in the Block Gap Control register is set, this bit is set if both read/ write transaction is stopped at a block gap. If Stop At Block Gap Request is not set to 1, this bit is not set to 1. (RW1C)</p> <p>(1) In the case of a Read Transaction</p> <p>This bit is set at the falling edge of the DAT Line Active Status (When the transaction is stopped at SD Bus timing. The Read Wait must be supported in order to use this function).</p> <p>(2) Case of Write Transaction</p> <p>This bit is set at the falling edge of Write Transfer Active Status (After getting CRC status at SD Bus timing).</p> <p>1 = Transaction stopped at block gap 0 = No Block Gap Event</p>	0
STATRANCMPLT	[1]	<p>Transfer Complete</p> <p>This bit is set if a read/ write transfer is complete.</p> <p>(1) In the case of a Read Transaction</p> <p>This bit is set at the falling edge of Read Transfer Active Status. There are two cases in which this interrupt is generated. The first is if a data transfer is complete as specified by data length (After the last data has been read to the Host System). The second if data has stopped at the block gap and complete the data transfer by setting the Stop At Block Gap Request in the Block Gap Control register (After valid data has been read to the Host</p>	0

NORINTSTS	Bit	Description	Initial State												
		<p>System).</p> <p>(2) In the case of a Write Transaction</p> <p>This bit is set at the falling edge of the DAT Line Active Status. There are two cases in which this interrupt is generated. The first if the last data is written to the SD card as specified by data length and the busy signal released. The second if data transfers are stopped at the block gap by setting Stop At Block Gap Request in the Block Gap Control register and data transfers complete. (After valid data is written to the SD card and the busy signal released). (RW1C)</p> <p>The table below shows that Transfer Complete has higher priority than Data Timeout Error. If both bits are set to 1, the data transfer is considered complete. Relation between Transfer Complete and Data</p> <table border="1"> <thead> <tr> <th>Transfer Complete</th> <th>Data Timeout Error</th> <th>Meaning of the status</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>Interrupted by another factor</td> </tr> <tr> <td>0</td> <td>1</td> <td>Timeout occur during transfer</td> </tr> <tr> <td>1</td> <td>Don't care</td> <td>Data transfer complete</td> </tr> </tbody> </table> <p>1 = Data Transfer Complete 0 = No Transfer Complete</p>	Transfer Complete	Data Timeout Error	Meaning of the status	0	0	Interrupted by another factor	0	1	Timeout occur during transfer	1	Don't care	Data transfer complete	
Transfer Complete	Data Timeout Error	Meaning of the status													
0	0	Interrupted by another factor													
0	1	Timeout occur during transfer													
1	Don't care	Data transfer complete													
STACMDCMPLT	[0]	<p>Command Complete</p> <p>This bit is set when receive the end bit of the command response. (Except Auto CMD12) Refer to Command Inhibit (CMD) in the Present State register.</p> <p>The table below shows that Command Timeout Error has higher priority than Command Complete. If both bits are set to 1, it is considered that the response was not received correctly. (RW1C)</p> <table border="1"> <thead> <tr> <th>Command Complete</th> <th>Command Timeout Error</th> <th>Meaning of the Status</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>Interrupted by another factor</td> </tr> <tr> <td>Don't care</td> <td>1</td> <td>Response not received within 64 SDCLK cycles.</td> </tr> <tr> <td>1</td> <td>0</td> <td>Response received</td> </tr> </tbody> </table> <p>1 = Command Complete 0 = No command complete</p>	Command Complete	Command Timeout Error	Meaning of the Status	0	0	Interrupted by another factor	Don't care	1	Response not received within 64 SDCLK cycles.	1	0	Response received	0
Command Complete	Command Timeout Error	Meaning of the Status													
0	0	Interrupted by another factor													
Don't care	1	Response not received within 64 SDCLK cycles.													
1	0	Response received													

NOTE:

1. Host Driver checks if interrupt is actually cleared by polling or monitoring the INTREQ port. If HCLK is much faster than SDCLK, it takes long time to be cleared for the bits actually.
2. Card Interrupt status bit keeps previous value until next card interrupt period (level interrupt) and is cleared if write to 1 (RW1C).
3. SD/MMC Controller of the S5PV210 does not support "card interrupt at block gap" used if the multiple block 4-bit operation.

7.10.12 ERROR INTERRUPT STATUS REGISTER

7.10.12.1 Error Interrupt Status Register

- ERRINTSTS0, ROC/RW1C, Address = 0xEB00_0032
- ERRINTSTS1, ROC/RW1C, Address = 0xEB10_0032
- ERRINTSTS2, ROC/RW1C, Address = 0xEB20_0032
- ERRINTSTS3, ROC/RW1C, Address = 0xEB30_0032

Signals defined in this register are enabled by the Error Interrupt Status Enable register, but not by the Error Interrupt Signal Enable register. Generates interrupt if the Error Interrupt Signal Enable is enabled and at least one of the statuses is set to 1. Writing to 1 clears the bit and writing to 0 keeps the bit unchanged. More than one status is cleared on one register write.

ERRINTSTS	Bit	Description	Initial State
Reserved	[15:11]	Reserved	0
STABOOTACKERR	[10]	Boot Ack Error This bit is set when the Host Controller detects Boot Ack receive error during Boot mode.	0
STAADMAERR	[9]	ADMA Error This bit is set if the Host Controller detects errors during ADMA based data transfer. The state of the ADMA at an error occurrence is saved in the ADMA Error Status Register. In addition, the Host Controller generates this Interrupt if it detects invalid descriptor data (Valid=0) at the ST_FDS state. ADMA Error State in the ADMA Error Status indicates that an error occurs in ST_FDS state. The Host Driver may find that Valid bit is not set at the error descriptor. 1 = Error 0 = No Error	0
STAACMDERR	[8]	Auto CMD12 Error Occurs if it detects that one of the bits in Auto CMD12 Error Status register has changed from 0 to 1. This bit is set to 1, if the errors in Auto CMD12 occur and if Auto CMD12 is not executed due to the previous command error. 1 = Error 0 = No Error	0
STACURERR	[7]	Current Limit Error Not implemented in this version. Always 0.	0
STADENDERR	[6]	Data End Bit Error Occurs if it detects 0 at the end bit position of read data which uses the DAT line or at the end bit position of the CRC Status. 1 = Error 0 = No Error	0
STADATCRCERR	[5]	Data CRC Error Occurs if it detects CRC error when transferring read data which uses the DAT line or if it detects the Write CRC status having a value of other than "010".	0

ERRINTSTS	Bit	Description	Initial State
		1 = Error 0 = No Error	
STADATTOUTERR	[4]	Data Timeout Error Occurs if it detects one of following timeout conditions. (1) Busy timeout for R1b, R5b type (2) Busy timeout after Write CRC status (3) Write CRC Status timeout (4) Read Data timeout. 1 = Timeout 0 = No Error	0
STACMDIDXERR	[3]	Command Index Error Occurs if a Command Index error occurs in the command response. 1 = Error 0 = No Error	0
STACMDEBITERR	[2]	Command End Bit Error Occurs if it detects that the end bit of a command response is 0. 1 = End bit Error generated 0 = No Error	
STACMDCRCERR	[1]	Command CRC Error Command CRC Error is generated in two cases. (1) If a response is returned and the Command Timeout Error is set to 0 (indicating no timeout), this bit is set to 1 if it detects a CRC error in the command response. (2) The Host Controller detects a CMD line conflict by monitoring the CMD line if a command is issued. If the Host Controller drives the CMD line to 1 level, but detects 0 levels on the CMD line at the next SDCLK edge, then the Host Controller aborts the command (Stop driving CMD line) and set this bit to 1. The Command Timeout Error also set to 1 to distinguish CMD line conflict. 1 = Generates CRC Error 0 = No Error	0
STACMDTOUTERR	[0]	Command Timeout Error Occurs if no response is returned within 64 SDCLK cycles from the end bit of the command. If the Host Controller detects a CMD line conflict, in which case Command CRC Error also set as shown in Table below, this bit sets without waiting for 64 SDCLK cycles because the Host Controller aborts command. 1 = Timeout 0 = No Error	0

The relation between Command CRC Error and Command Timeout Error is shown in Table below.

The Relation Between Command CRC Error and Command Timeout Error

Command CRC Error	Command Timeout Error	Kinds of Error
0	0	No Error
0	1	Response Timeout Error
1	0	Response CRC Error
1	1	CMD line conflict

7.10.13 NORMAL INTERRUPT STATUS ENABLE REGISTER

7.10.13.1 Normal Interrupt Status Enable Register

- NORINTSTSEN0, R/W, Address = 0xEB00_0034
- NORINTSTSEN1, R/W, Address = 0xEB10_0034
- NORINTSTSEN2, R/W, Address = 0xEB20_0034
- NORINTSTSEN3, R/W, Address = 0xEB30_0034

Setting to 1 enables Interrupt Status.

NORINTSTSEN	Bit	Description	Initial State
-	[15]	Fixed to 0 The Host Driver controls error interrupts using the Error Interrupt Status Enable register. (R)	0
ENSTAFIA3	[14]	FIFO SD Address Pointer Interrupt 3 Status Enable 1 = Enabled 0 = Masked	0
ENSTAFIA2	[13]	FIFO SD Address Pointer Interrupt 2 Status Enable 1 = Enabled 0 = Masked	0
ENSTAFIA1	[12]	FIFO SD Address Pointer Interrupt 1 Status Enable 1 = Enabled 0 = Masked	0
ENSTAFIA0	[11]	FIFO SD Address Pointer Interrupt 0 Status Enable 1 = Enabled 0 = Masked	0
ENSTARWAIT	[10]	Read Wait interrupt status enable 1 = Enabled 0 = Masked	0
ENSTACCS	[9]	CCS Interrupt Status Enable 1 = Enabled 0 = Masked	0
ENSTACARDINT	[8]	Card Interrupt Status Enable If this bit is set to 0, the Host Controller clears interrupt request to the System. The Card Interrupt detection is stopped if this bit is cleared and restarted if this bit is set to 1. The Host Driver must clear the Card Interrupt Status Enable before servicing the Card Interrupt and must set this bit again after all interrupt requests from the card are cleared to prevent inadvertent interrupts. 1 = Enabled 0 = Masked	0
ENSTACARDREM	[7]	Card Removal Status Enable 1 = Enabled 0 = Masked	0

NORINTSTSEN	Bit	Description	Initial State
ENSTACARDNS	[6]	Card Insertion Status Enable 1 = Enabled 0 = Masked	0
ENSTABUFRDRDY	[5]	Buffer Read Ready Status Enable 1 = Enabled 0 = Masked	0
ENSTABUFWTRDY	[4]	Buffer Write Ready Status Enable 1 = Enabled 0 = Masked	0
ENSTADMA	[3]	DMA Interrupt Status Enable 1 = Enabled 0 = Masked	0
ENSTABLKGAP	[2]	Block Gap Event Status Enable 1 = Enabled 0 = Masked	0
ENSTASTANSCMPLT	[1]	Transfer Complete Status Enable 1 = Enabled 0 = Masked	0
ENSTACMDCMPLT	[0]	Command Complete Status Enable 1 = Enabled 0 = Masked	0

7.10.14 ERROR INTERRUPT STATUS ENABLE REGISTER

7.10.14.1 Error Interrupt Status Enable Register

- ERRINTSTSEN0, R/W, 0xEB00_0036
- ERRINTSTSEN1, R/W, 0xEB10_0036
- ERRINTSTSEN2, R/W, 0xEB20_0036
- ERRINTSTSEN3, R/W, 0xEB30_0036

Setting to 1 enables Error Interrupt Status.

ERRINTSTSEN	Bit	Description	Initial State
Reserved	[15:11]	Reserved	0
ENSTABOOTACKERR	[10]	Boot Ack Error Status Enable 1 = Enabled, 0 = Masked	0
ENSTAADMAERR	[9]	ADMA Error Status Enable 1 = Enabled, 0 = Masked	0
ENSTAACMDERR	[8]	Auto CMD12 Error Status Enable 1 = Enabled, 0 = Masked	0
ENSTACURERR	[7]	Current Limit Error Status Enable This function is not implemented in this version. 1 = Enabled 0 = Masked	0
ENSTADENDERR	[6]	Data End Bit Error Status Enable 1 = Enabled 0 = Masked	0
ENSTADATCRCERR	[5]	Data CRC Error Status Enable 1 = Enabled 0 = Masked	0
ENSTADATTOUTERR	[4]	Data Timeout Error Status Enable 1 = Enabled 0 = Masked	0
ENSTACMDIDXERR	[3]	Command Index Error Status Enable 1 = Enabled 0 = Masked	0
ENSTACMDEBITERR	[2]	Command End Bit Error Status Enable 1 = Enabled 0 = Masked	0
ENSTACMDCRCERR	[1]	Command CRC Error Status Enable 1 = Enabled 0 = Masked	0
ENSTACMDTOUTERR	[0]	Command Timeout Error Status Enable 1 = Enabled 0 = Masked	0

7.10.15 NORMAL INTERRUPT SIGNAL ENABLE REGISTER

7.10.15.1 Normal Interrupt Signal Enable Register

- NORINTSIGEN0, R/W, Address = 0xEB00_0038
- NORINTSIGEN1, R/W, Address = 0xEB10_0038
- NORINTSIGEN2, R/W, Address = 0xEB20_0038
- NORINTSIGEN3, R/W, Address = 0xEB30_0038

This register is used to select which interrupt status is indicated to the Host System as the interrupt. These status bits share the same 1 bit interrupt line. To enable interrupt generate set any of this bit to 1.

NORINTSIGEN	Bit	Description	Initial State
	[15]	Fixed to 0 The Host Driver controls error interrupts using the Error Interrupt Signal Enable register.	0
ENSIGFIA3	[14]	FIFO SD Address Pointer Interrupt 3 Signal Enable 1 = Enabled 0 = Masked	0
ENSIGFIA2	[13]	FIFO SD Address Pointer Interrupt 2 Signal Enable 1 = Enabled 0 = Masked	0
ENSIGFIA1	[12]	FIFO SD Address Pointer Interrupt 1 Signal Enable 1 = Enabled 0 = Masked	0
ENSIGFIA0	[11]	FIFO SD Address Pointer Interrupt 0 Signal Enable 1 = Enabled 0 = Masked	0
ENSIGRWAIT	[10]	Read Wait Interrupt Signal Enable 1 = Enabled 0 = Masked	0
ENSIGCCS	[9]	CCS Interrupt Signal Enable Command Complete Signal Interrupt Status bit is for CE-ATA interface mode. 1 = Enabled 0 = Masked	0
ENSIGCARDINT	[8]	Card Interrupt Signal Enable 1 = Enabled 0 = Masked	0
ENSIGCARDREM	[7]	Card Removal Signal Enable 1 = Enabled 0 = Masked	0
ENSIGCARDNS	[6]	Card Insertion Signal Enable 1 = Enabled 0 = Masked	0
ENSIGBUFRDRDY	[5]	Buffer Read Ready Signal Enable	0

NORINTSIGEN	Bit	Description	Initial State
		1 = Enabled 0 = Masked	
ENSIGBUFWTRDY	[4]	Buffer Write Ready Signal Enable 1 = Enabled 0 = Masked	0
ENSIGDMA	[3]	DMA Interrupt Signal Enable 1 = Enabled 0 = Masked	0
ENSIGBLKGAP	[2]	Block Gap Event Signal Enable 1 = Enabled 0 = Masked	0
ENSIGSTANSCMPLT	[1]	Transfer Complete Signal Enable 1 = Enabled 0 = Masked	0
ENSIGCMDCMPLT	[0]	Command Complete Signal Enable 1 = Enabled 0 = Masked	0

7.10.16 ERROR INTERRUPT SIGNAL ENABLE REGISTER

7.10.16.1 Error Interrupt Signal Enable Register

- ERRINTSIGEN0, R/W, Address = 0xEB00_003A
- ERRINTSIGEN1, R/W, Address = 0xEB10_003A
- ERRINTSIGEN2, R/W, Address = 0xEB20_003A
- ERRINTSIGEN3, R/W, Address = 0xEB30_003A

This register is used to select which interrupt status is notified to the Host System as the interrupt. These status bits share the same 1 bit interrupt line. To enable interrupt generate set any of this bit to 1.

ERRINTSIGEN	Bit	Description	Initial State
Reserved	[15:11]	Reserved	0
ENSIGBOOTACKERR	[10]	Boot Ack Error Signal Enable 1 = Enabled 0 = Masked	0
ENSIGADMAERR	[9]	ADMA Error Signal Enable 1 = Enabled 0 = Masked	0
ENSIGACMDERR	[8]	Auto CMD12 Error Signal Enable 1 = Enabled 0 = Masked	0
ENSIGCURERR	[7]	Current Limit Error Signal Enable This function is not implemented in this version. 1 = Enabled 0 = Masked	0
ENSIGDENDERR	[6]	Data End Bit Error Signal Enable 1 = Enabled 0 = Masked	0
ENSIGDATCRCERR	[5]	Data CRC Error Signal Enable 1 = Enabled 0 = Masked	0
ENSIGDATOUTERR	[4]	Data Timeout Error Signal Enable 1 = Enabled 0 = Masked	0
ENSIGCMDIDXERR	[3]	Command Index Error Signal Enable 1 = Enabled 0 = Masked	0
ENSIGCMDEBITERR	[2]	Command End Bit Error Signal Enable 1 = Enabled 0 = Masked	0
ENSIGCMDCRCERR	[1]	Command CRC Error Signal Enable 1 = Enabled 0 = Masked	0
ENSIGCMDTOUTERR	[0]	Command Timeout Error Signal Enable	0

ERRINTSIGEN	Bit	Description	Initial State
		1 = Enabled 0 = Masked	

NOTE: Detailed documents must be copied from SD Host Standard Specification.

Samsung Confidential

7.10.17 AUTOCMD12 ERROR STATUS REGISTER

7.10.17.1 Auto CMD12 Error Status Register

- ACMD12ERRSTS0, ROC, Address = 0xEB00_003C
- ACMD12ERRSTS1, ROC, Address = 0xEB10_003C
- ACMD12ERRSTS2, ROC, Address = 0xEB20_003C
- ACMD12ERRSTS3, ROC, Address = 0xEB30_003C

If Auto CMD12 Error Status is set, the Host Driver checks this register to identify what kind of error Auto CMD12 indicated. This register is valid if the Auto CMD12 Error is set.

ACMD12ERRSTS	Bit	Description	Initial State
Reserved	[15:8]	Reserved	0
STANCMDAER	[7]	Command Not Issued By Auto CMD12 Error If the bit is set to 1, it means CMD_wo_DAT is not executed due to an Auto CMD12 Error (D04-D01) in this register. 1 = Not Issued 0 = No error	0
Reserved	[6:5]	Reserved	0
STACMDIDXERR	[4]	Auto CMD12 Index Error Occurs if the Command Index error occurs in response to a command. 1 = Error 0 = No Error	0
STACMDEBITAER	[3]	Auto CMD12 End Bit Error Occurs if it detects that the end bit of command response is 0. 1 = End Bit Error Generated 0 = No Error	0
STACMDCRCAER	[2]	Auto CMD12 CRC Error Occurs if it detects a CRC error in the command response. 1 = CRC Error Generated 0 = No Error	0
STACMDTOUTAER	[1]	Auto CMD12 Timeout Error Occurs if no response is returned within 64 SDCLK cycles from the end bit of command. If this bit is set to 1, the other error status bits (D04-D02) are meaningless. 1 = Time out 0 = No Error	0
STANACMDAER	[0]	Auto CMD12 Not Executed If memory multiple block data transfer is not started due to command error, this bit is not set because it is not necessary to issue Auto CMD12. If this bit is set to 1, it means the Host Controller cannot issue Auto CMD12 to stop memory multiple block data transfer due to some error. If this bit is set to 1, other error status bits (D04-D01) are meaningless. 1 = Not executed 0 = Executed	0

The relation between Auto CMD12 CRC Error and Auto CMD12 Timeout Error is shown below.

The relation between Command CRC Error and Command Timeout Error

Auto CMD12 CRC Error	Auto CMD12 Timeout Error	Kinds of Error
0	0	No Error
0	1	Response Timeout Error
1	0	Response CRC Error
1	1	CMD line conflict

The timing of changing Auto CMD12 Error Status is classified in three scenarios:

1. If the Host Controller is going to issue Auto CMD12
 - Set D00 to 1 if Auto CMD12 cannot be issued due to an error in the previous command.
 - Set D00 to 0 if Auto CMD12 is issued.
2. At the end bit of an Auto CMD12 response
 - Check received responses by checking the error bits D01, D02, D03 and D04.
 - Set to 1 if error is detected.
 - Set to 0 if error is not detected.
3. Before reading the Auto CMD12 Error Status bit D07
 - Set D07 to 1 if there is a command cannot be issued
 - Set D07 to 0 if there is no command to issue

Timing to generate the Auto CMD12 Error and writing to the Command register are asynchronous. Then D07 are sampled if driver never writes to the Command register. Therefore before reading the Auto CMD12 Error Status register, set the D07 status bit. Generates Auto CMD12 Error Interrupt if one of the error bits D00 to D04 is set to 1. The Command Not Issued by Auto CMD12 Error does not generate an interrupt.

7.10.18 CAPABILITIES REGISTER

7.10.18.1 Capabilities Register

- CAPAREG0, HWInit, Address = 0xEB00_0040
- CAPAREG1, HWInit, Address = 0xEB10_0040
- CAPAREG2, HWInit, Address = 0xEB20_0040
- CAPAREG3, HWInit, Address = 0xEB30_0040

When HWINITFIN bit (CONTROL2 register) as 0, This register can be updated.

This register provides the Host Driver with information specific to the Host Controller implementation. The Host Controller implements these values as fixed or loaded from flash memory during power on initialization. Refer to Software Reset for the Software Reset register for loading from flash memory and completion timing control.

CAPAREG	Bit	Description	Initial State
Reserved	[31:27]	Reserved	
CAPAV18	[26]	Voltage Support 1.8V (HWInit) 1 = 1.8V Supported 0 = 1.8V Not Supported	1
CAPAV30	[25]	Voltage Support 3.0V (HWInit) 1 = 3.0V Supported 0 = 3.0V Not Supported	0
CAPAV33	[24]	Voltage Support 3.3V (HWInit) 1 = 3.3V Supported 0 = 3.3V Not Supported	1
CAPASUSRES	[23]	Suspend/Resume Support (HWInit) This bit indicates whether the Host Controller supports Suspend / Resume functionality. If this bit is 0, the Suspend and Resume mechanism are not supported and the Host Driver does not issue either Suspend or Resume commands. 1 = Supported 0 = Not Supported	1
CAPADMA	[22]	DMA Support (HWInit) This bit indicates whether the Host Controller is capable of using DMA to transfer data between system memory and the Host Controller directly. 1 = DMA Supported 0 = DMA Not Supported	1
CAPAHSPD	[21]	High Speed Support (HWInit) This bit indicates whether the Host Controller and the Host System support High Speed mode and they can supply SD Clock frequency from 25MHz to 50MHz. 1 = High Speed Supported 0 = High Speed Not Supported	1
Reserved	[20]	Reserved	0
CAPAADMA2	[19]	ADMA2 Support	1

CAPAREG	Bit	Description	Initial State
		This bit indicates whether the Host Controller is capable of using ADMA2. 1 = ADMA2 Support 0 = ADMA2 not Support	
Reserved	[28]	Reserved	0
CAPAMAXBLKLEN	[17:16]	Max Block Length (HWInit) This value indicates the maximum block size that the Host Driver can read and write to the buffer in the Host Controller. The buffer transfers this block size without wait cycles. Three sizes are defined as indicated below. 00 = 512-byte 01 = 1024-byte 10 = 2048-byte 11 = Reserved	0
Reserved	[15:14]	Reserved	0
CAPABASECLK	[13:8]	Base Clock Frequency For SD Clock (HWInit) This value indicates the base (maximum) clock frequency for the SD Clock. Unit values are 1MHz. If the real frequency is 16.5MHz, the larger value is set to 01 0001b (17MHz) because the Host Driver uses this value to calculate the clock divider value (Refer to the SDCLK Frequency Select in the Clock Control register.) and it does not exceed upper limit of the SD Clock frequency. The supported clock range is 10MHz to 63MHz. If these bits are all 0, the Host System has to get information via another method. Not 0 = 1MHz to 63MHz 000000b = Get information via another method	0
CAPATOUTUNIT	[7]	Timeout Clock Unit (HWInit) This bit shows the unit of base clock frequency used to detect Data Timeout Error. 0 = kHz 1 = MHz	1
Reserved	[6]	Reserved	0
CAPATOUTCLK	[5:0]	Timeout Clock Frequency (HWInit) This bit shows the base clock frequency used to detect Data Timeout Error. The Timeout Clock Unit defines the unit of this field value. Timeout Clock Unit =0 [kHz] unit: 1kHz to 63kHz Timeout Clock Unit =1 [MHz] unit: 1MHz to 63MHz Not 0 = 1kHz to 63kHz or 1MHz to 63MHz 00 0000b = Get information via another method	0

7.10.19 MAXIMUM CURRENT CAPABILITIES REGISTER

7.10.19.1 Maximum Current Capabilities Register

- MAXCURR0, HWInit, Address = 0xEB00_0048
- MAXCURR1, HWInit, Address = 0xEB10_0048
- MAXCURR2, HWInit, Address = 0xEB20_0048
- MAXCURR3, HWInit, Address = 0xEB30_0048

When HWINITFIN bit (CONTROL2 register) as 0, This register can be updated.

These registers indicate maximum current capability for each voltage. The value is meaningful if Voltage Support is set in the Capabilities register. If this information is supplied by the Host System via another method, all Maximum Current Capabilities register will be 0.

MAXCURR	Bit	Description	Initial State
Reserved	[31:24]	Reserved	
MAXCURR18	[23:16]	Maximum Current for 1.8V (HWInit)	0
MAXCURR30	[15:8]	Maximum Current for 3.0V (HWInit)	0
MAXCURR33	[7:0]	Maximum Current for 3.3V (HWInit)	0

This register measures current in 4mA steps. Each voltage level's current support is described using the Table below.

Maximum Current Value Definition

Register Value	Current Value
0	Get information via another method
1	4mA
2	8mA
3	12mA
...	...
255	1020mA

7.10.20 FORCE EVENT REGISTER FOR AUTO CMD12 ERROR STATUS

7.10.20.1 Force Event Auto CMD12 Error Interrupt Register

- FEAER0, W, Address = 0xEB00_0050
- FEAER1, W, Address = 0xEB10_0050
- FEAER2, W, Address = 0xEB20_0050
- FEAER3, W, Address = 0xEB30_0050

The Force Event Register is not a physically implemented register. Rather, it is an address at which the Auto CMD12 Error Status Register can be written.

Warning: 1: set each bit of the Auto CMD12 Error Status Register
0: no effect

- D15 D12

FEAER	Bit	Description	Initial State
Reserved	[15:8]	Reserved	0x0
FENCMDAER	[7]	Force Event for Command Not Issued By Auto CMD12 Error 1 = Generates Interrupt 0 = No Interrupt	0
Reserved	[6:5]	Reserved	0
FECMDIDXERR	[4]	Force Event for Auto CMD12 Index Error 1 = Interrupt 0 = No Interrupt	0
FECMDEBITAER	[3]	Force Event for Auto CMD12 End Bit Error 1 = Generates Interrupt 0 = No Interrupt	0
FECMDCRCRAER	[2]	Force Event for Auto CMD12 CRC Error 1 = Generates Interrupt 0 = No Interrupt	0
FECMDTOUTAER	[1]	Force Event for Auto CMD12 Timeout Error 1 = Generates Interrupt 0 = No Interrupt	0
FENACMDAER	[0]	Force Event for Auto CMD12 Not Executed 1 = Generates Interrupt 0 = No Interrupt	0

7.10.21 FORCE EVENT REGISTER FOR ERROR INTERRUPT STATUS

7.10.21.1 Force Event Error Interrupt Register Error Interrupt

- FEERR0, W, Address = 0xEB00_0052
- FEERR1, W, Address = 0xEB10_0052
- FEERR2, W, Address = 0xEB20_0052
- FEERR3, W, Address = 0xEB30_0052

The Force Event Register is not a physically implemented register. Rather, it is an address at which the Error Interrupt Status register is written. The effect of a write to this address is reflected in the Error Interrupt Status Register if the corresponding bit of the Error Interrupt Status Enable Register is set.

Warning: 1: set each bit of the Error Interrupt Status Register
0: no effect

NOTE: By setting this register, the Error Interrupt can be set in the Error Interrupt Status register. In order to generate interrupt signal, both the Error Interrupt Status Enable and Error Interrupt Signal Enable must be set.

FEERR	Bit	Description	Initial State
Reserved	[15:11]	Reserved	0x0
FEBOOTACKERR	[10]	Force Event for Boot Ack Error 1 = Interrupt is generated 0 = No Interrupt	0
FEADMAERR	[9]	Force Event for ADMA Error 1 = Generates Interrupt 0 = No Interrupt	0
FEACMDERR	[8]	Force Event for Auto CMD12 Error 1 = Generates Interrupt 0 = No Interrupt	0
Reserved	[7]	Reserved	0
FEDENDERR	[6]	Reserved	0
FEDATCRCERR	[5]	Force Event for Data CRC Error 1 = Generates Interrupt 0 = No Interrupt	0
FEDATTOUTERR	[4]	Force Event for Data Timeout Error 1 = Generates Interrupt 0 = No Interrupt	0
FECMDIDXERR	[3]	Force Event for Command Index Error 1 = Generates Interrupt 0 = No Interrupt	0
FECMDEBITERR	[2]	Force Event for Command End Bit Error 1 = Generates Interrupt 0 = No Interrupt	0

FEERR	Bit	Description	Initial State
FECMDCRCERR	[1]	Force Event for Command CRC Error 1 = Generates Interrupt 0 = No Interrupt	0
FECMDTOUTERR	[0]	Force Event for Command Timeout Error 1 = Generates Interrupt 0 = No Interrupt	0

7.10.22 ADMA ERROR STATUS REGISTER

7.10.22.1 ADMA Error Status Register

- ADMAERR0, R/W, Address = 0xEB00_0054
- ADMAERR1, R/W, Address = 0xEB10_0054
- ADMAERR2, R/W, Address = 0xEB20_0054
- ADMAERR3, R/W, Address = 0xEB30_0054

If ADMA Error Interrupt occurs, the ADMA Error States field in this register holds the ADMA state and the ADMA System Address Register holds the address around the error descriptor. For recovering the error, the Host Driver requires the ADMA state to identify the error descriptor address as follows:

- ST_STOP: Previous location set in the ADMA System Address register is the error descriptor address
- ST_FDS: Current location set in the ADMA System Address register is the error descriptor address
- ST_CADR: This state is never set because do not generate ADMA error in this state.
- ST_TFR: Previous location set in the ADMA System Address register is the error descriptor address

In case of write operation, the Host Driver should use ACMD22 to get the number of written block rather than using this information, since unwritten data may exist in the Host Controller.

The Host Controller generates the ADMA Error Interrupt if it detects invalid descriptor data (Valid = 0) at the ST_FDS state. In this case, ADMA Error State indicates that an error occurs at ST_FDS state. The Host Driver finds that the Valid bit is not set in the error descriptor.

ADMAERR	Bit	Description	Initial State
Reserved	[31:11]	Reserved	0x00
STAADMAFINBLK	[10]	ADMA Final Block Transferred (ROC) In ADMA operation mode, this field is set to High if the Transfer Complete condition and the block are final (no block transfer remains). If this bit is Low when the Transfer Complete condition and Transfer Complete is done due to the Stop at Block Gap, so data to be transferred still remains.	0
ADMACONTREQ	[9]	ADMA Continue Request (WO) If the stop state by ADMA Interrupt, ADMA operation set this bit to HIGH to continue.	0
ADMASTAINT	[8]	ADMA Interrupt Status (RW1C) This bit is set to HIGH if INT attribute in the ADMA Descriptor Table is asserted. This bit is not affected by ADMA error interrupt.	0
	[7:3]	Reserved	0
ADMALENMISERR	[2]	ADMA Length Mismatch Error This error occurs in the following 2 cases. (1) While Block Count Enable being set, the total data length specified by the Descriptor table is different from that specified by the Block Count and Block Length.	00

ADMAERR	Bit	Description	Initial State
		(2) Total data length can not be divided by the block length. 0 = No Error 1 = Error	
ADMAERRST	[1:0]	<p>ADMA Error State</p> <p>This field indicates the state of ADMA when error is occurred during ADMA data transfer. This field never indicates "10" because ADMA never stops in this state.</p> <p>D01 - D00 ADMA Error State when error is occurred</p> <p>Contents of SYS_SDR register</p> <ul style="list-style-type: none"> 00 = ST_STOP (Stop DMA) Points next of the error descriptor 01 = ST_FDS (Fetch Descriptor) Points the error descriptor 10 = Never set this state (Not used) 11 = ST_TFR (Transfer Data) Points the next of the error descriptor 	0

7.10.23 ADMA SYSTEM ADDRESS REGISTER

7.10.23.1 ADMA System Address Register

- ADMASYSADDR0, R/W, Address = 0xEB00_0058
- ADMASYSADDR1, R/W, Address = 0xEB10_0058
- ADMASYSADDR2, R/W, Address = 0xEB20_0058
- ADMASYSADDR3, R/W, Address = 0xEB30_0058

This register contains the physical Descriptor address used for ADMA data transfer.

ADMASYSADDR	Bit	Description	Initial State												
ADMASYSAD	[31:0]	<p>ADMA System Address This register holds byte address of executing command of the Descriptor table. 32-bit Address Descriptor uses lower 32-bit of this register. At the start of ADMA, the Host Driver sets start address of the Descriptor table. The ADMA increments this register address, which points to next line, if having fetched a Descriptor line. If the ADMA Error Interrupt is generated, this register holds valid Descriptor address depending on the ADMA state. The Host Driver programs Descriptor Table on 32-bit boundary and set 32-bit boundary address to this register. ADMA2 ignores lower 2-bit of this register and assumes it to be 00b.</p> <p>32-bit Address ADMA Register Value 32-bit System Address</p> <table> <tbody> <tr> <td>xxxxxxxx 00000000h</td> <td>00000000h</td> </tr> <tr> <td>xxxxxxxx 00000004h</td> <td>00000004h</td> </tr> <tr> <td>xxxxxxxx 00000008h</td> <td>00000008h</td> </tr> <tr> <td>xxxxxxxx 0000000Ch</td> <td>0000000Ch</td> </tr> <tr> <td>.....</td> <td>.....</td> </tr> <tr> <td>xxxxxxxx FFFFFFFFCh</td> <td>FFFFFFFCh</td> </tr> </tbody> </table> <p>Note: The data length of the ADMA Descriptor Table should be the word unit (multiple of the 4-byte).</p>	xxxxxxxx 00000000h	00000000h	xxxxxxxx 00000004h	00000004h	xxxxxxxx 00000008h	00000008h	xxxxxxxx 0000000Ch	0000000Ch	xxxxxxxx FFFFFFFFCh	FFFFFFFCh	00
xxxxxxxx 00000000h	00000000h														
xxxxxxxx 00000004h	00000004h														
xxxxxxxx 00000008h	00000008h														
xxxxxxxx 0000000Ch	0000000Ch														
.....														
xxxxxxxx FFFFFFFFCh	FFFFFFFCh														

7.10.24 CONTROL REGISTER 2

7.10.24.1 Control Register 2

- CONTROL2_0, R/W, Address = 0xEB00_0080
- CONTROL2_1, R/W, Address = 0xEB10_0080
- CONTROL2_2, R/W, Address = 0xEB20_0080
- CONTROL2_3, R/W, Address = 0xEB30_0080

This register contains the SD Command Argument.

CONTROL2	Bit	Description	Initial State
ENSTAASYNCCCLR	[31]	Write Status Clear Async Mode Enable This bit makes async-clear enable about Normal and Error interrupt status bit. During the initialization procedure command operation, this bit should be enabled. 0 = Disable 1 = Enable	0
ENCMDCNFMSK	[30]	Command Conflict Mask Enable This bit can mask enable the Command Conflict Status (bit [1:0] of the "ERROR INTERRUPT STATUS REGISTER") 0 = Mask Disable 1 = Mask Enable Note: If the OUTEDGEINV field in the Host Control Register is set (High Speed data transfer), this field should be enabled to prevent from command conflict status alarm.	0
Reserved	[29]	Reserved (must be 1'b0)	0
SELCARDOUT	[28]	Card Removed Condition Selection 0 = Card Removed condition is "Not Card Insert" State (When the transition from "Card Inserted" state to "Debouncing" state in Figure 7-17) 1 = Card Removed state is "Card Out" State (If the transition from "Debouncing" state to "No Card" state in Figure 7-17)	0
FLTCLKSEL	[27:24]	Filter Clock (iFLTCLK) Selection Filter Clock period = $2^{(FltClkSel + 5)} \times iSDCLK$ period 0000 = 25 x iSDCLK 0001 = 26 x iSDCLK ... 1111 = 220 x iSDCLK	0
LVLDAT	[23:16]	DAT line level Bit[23]=DAT[7], BIT[22]=DAT[6], BIT[21]=DAT[5], BIT[20]=DAT[4], Bit[19]=DAT[3], BIT[18]=DAT[2], BIT[17]=DAT[1], BIT[16]=DAT[0] (Read Only)	Line state
ENFBCLKTX	[15]	Feedback Clock Enable for Tx Data/Command Clock 0 = Disable 1 = Enable	0

CONTROL2	Bit	Description	Initial State
ENFBCLKRX	[14]	Feedback Clock Enable for Rx Data/Command Clock 0 = Disable 1 = Enable	0
Reserved	[13]	Reserved (must be 1'b0)	0
SDOPSIGPC	[12]	SD Output Signal Power Control Support If set this field is enables output CMD and DAT referencing SD Bus Power bit in the "PWRCON register". 0 = CMD and DAT outputs are not controlled by SD Bus Power bit 1 = CMD and DAT outputs are controlled(masked) by SD Bus Power bit Note: This function is not implemented in this version.	0
ENBUSYCHKTXSTART	[11]	CE-ATA I/F mode Busy state check before Tx Data start state 0 = Disable 1 = Enable	0
DFCNT	[10:9]	Debounce Filter Count Debounce Filter Count setting register for Card Detect signal input (SDCD#) 00 = No use debounce filter 01 = 4 x iFLTCLK 10 = 16 x iFLTCLK 11 = 64 x iFLTCLK	0
ENCLKOUTHOLD	[8]	SDCLK Hold Enable Enter and exit of the SDCLK Hold state is done by Host Controller. 0 = Disable 1 = Enable Note: This field should be 1.	0
RWAITMODE	[7]	Read Wait Release Control 0 = Read Wait state is released by the Host Controller (Auto) 1 = Read Wait state is released by the Host Driver (Manual)	0
DISBUFRD	[6]	Buffer Read Disable 0 = Normal mode, user can read buffer(FIFO) data using 0x20 register 1 = User cannot read buffer(FIFO) data using 0x20 register. In this case, the buffer memory is read through memory area (Debug purpose).	0
SELBASE CLK	[5:4]	Base Clock Source Select 00 or 01 = HCLK 10 = SCLK_MMC0~3 (from SYSCON), 11 = Reserved	00

CONTROL2	Bit	Description	Initial State
SDINPSIGPC	[3]	SD Input Signal Power Control Support If set this field enables input CMD and DAT referencing SD Bus Power bit in the "PWRCON register". 0 = No Sync, no switch input enable signal (Command, Data) 1 = Sync, control input enable signal (Command, Data) Note: This function is not implemented in this version.	0
Reserved	[2]	Reserved	0
ENCLKOUTMSKCON	[1]	SDCLK output clock masking when Card Insert cleared If this field is High, it is used not to stop SDCLK if No Card state. 0 = Disable 1 = Enable	0
HWINITFIN	[0]	SD Host Controller Hardware Initialization Finish 0 = Not Finish 1 = Finish	0

NOTE:

1. Ensure to set SDCLK Hold Enable (EnSCHold) if the card does not support Read Wait to guarantee for Receive data not overwritten to the internal FIFO memory.
2. CMD_wo_DAT issue is prohibited during READ transfer if SDCLK Hold Enable is set.

7.10.25 CONTROL REGISTERS 3 REGISTER

7.10.25.1 FIFO Interrupt Control (Control Register 3)

- CONTROL3_0, R/W, Address = 0xEB00_0084
- CONTROL3_1, R/W, Address = 0xEB10_0084
- CONTROL3_2, R/W, Address = 0xEB20_0084
- CONTROL3_3, R/W, Address = 0xEB30_0084

CONTROL3	Bit	Description	Initial State
FCSEL3	[31]	Feedback Clock Select [3] Reference (1)	0x0
FIA3	[30:24]	FIFO Interrupt Address register 3 FIFO (512Byte Buffer memory, word address unit) Initial value (0x7F) generates at 512-byte(128-word) position.	0x7F
FCSEL2	[23]	Feedback Clock Select [2] Reference (1)	0x0
FIA2	[22:16]	FIFO Interrupt Address register 2 FIFO (512Byte Buffer memory, word address unit) Initial value (0x5F) generates at 384-byte(96-word) position.	0x5F
FCSEL1	[15]	Feedback Clock Select [1] Reference (2)	0x0
FIA1	[14:8]	FIFO Interrupt Address register 1 FIFO (512Byte Buffer memory, word address unit) Initial value (0x3F) generates at 256-byte(64-word) position.	0x3F
FCSEL0	[7]	Feedback Clock Select [0] Reference (2)	0x0
FIA0	[6:0]	FIFO Interrupt Address register 0 FIFO (512Byte Buffer memory, word address unit) Initial value (0x1F) generates at 128-byte(32-word) position.	0x1F

NOTE:

1. FCSel[3:2] : Tx Feedback Clock Delay Control : Inverter delay means 10ns delay if SDCLK 50MHz setting '00', '01' = Inverter delay, '10', '11' = basic delay(aroujnd 2ns)
2. FCSel[1:0] : Rx Feedback Clock Delay Control : Inverter delay means10ns delay if SDCLK 50MHz setting '00', '01' = Inverter delay, '10', '11' = basic delay(aroujnd 2ns)
3. Tx Feedback inversion setting (FCSel[3:2] = '00' or '01'), Tx Feedback clock enable (ENFBCLKTX = 0) and Normal Speed mode (OUTEDGEINV = 0) setting make Tx data transfer mismatch (Do not set).

7.10.26 CONTROL REGISTER 4

7.10.26.1 Control Register 4

- CONTROL4_0, R/W, Address = 0xEB00_008C
- CONTROL4_1, R/W, Address = 0xEB10_008C
- CONTROL4_2, R/W, Address = 0xEB20_008C
- CONTROL4_3, R/W, Address = 0xEB30_008C

CONTROL4	Bit	Description	Initial State
Reserved	[31:18]	Reserved	0
SELCLKPADDS	[17:16]	SD Clock Output PAD Drive Strength Select 00 = 2mA 01 = 4mA 10 = 7mA 11 = 9mA Note: This function is not implemented in this version.	0x3
Reserved	[15:2]	Reserved	-
STABLKGAPBUSY	[1]	Status Block Gap Access Busy This bit is "High" when the clock domain crossing (HCLK to SDCLK) operation is processing when the write operation to the BLKGAP register. This bit is status bit and Read Only (RO)	0
STABUSY	[0]	Status Busy This bit is "High" if the clock domain crossing (HCLK to SDCLK) operation is under process. This bit is status bit and Read Only (RO)	0

7.10.27 HOST CONTROLLER VERSION REGISTER

7.10.27.1 Host Controller Version Register

- HCVER0, HWInit, Address = 0xEB00_00FE
- HCVER1, HWInit, Address = 0xEB10_00FE
- HCVER2, HWInit, Address = 0xEB20_00FE
- HCVER3, HWInit, Address = 0xEB30_00FE

This register contains the SD Command Argument.

HCVER	Bit	Description	Initial State
VENVER	[15:8]	Vendor Version Number This status is reserved for the vendor version number. The Host Driver should not use this status. 0x24: SDMMC4.2 Host Controller	0x24
SPECVER	[7:0]	Specification Version Number This status indicates the Host Controller Specification Version. The upper and lower 4-bits indicate the version 00 = SD Host Specification Version 1.0 01 = SD Host Specification Version 2.0 Including the feature of the ADMA and Test Register Others = Reserved	0x01

8

TRANSPORT STREAM INTERFACE

8.1 OVERVIEW OF TRANSPORT STREAM INTERFACE

The Transport Stream Interface (TSI) in S5PV210 receives transport stream data from channel chip, which it writes to a specific address of the output buffer (SDRAM). Depending on the bus bandwidth, TSI has 32 words FIFO.

Using word-aligned address, the TSI sends data streams to the output buffer (SDRAM). One packet size is equal to 47 words. The output buffer size should be equal to a multiple of 47 words (one packet size).

If the data is written in the output buffer, the SDRAM full interrupt occurs.

8.1.1 KEY FEATURES OF TRANSPORT STREAM INTERFACE

- Writes transport stream received from channel chip to output buffer (supports 1-/ 4-/ 8-beat burst, word-aligned)
- Supports TS interface in DVB-H/ DVB-T/ ISDB-T/ T-DMB/ DAB mode
- Supports TS_CLK falling/ rising edge data fetch mode
- Supports active high or active low mode for TS signals (TS_VALID, TS_SYNC, and TS_ERROR)
- Supports MSB to LSB or LSB to MSB data byte order
- Specifies the maximum size of output buffer for store transport stream as 256KBytes
- Supports two sync detecting modes:
 - TS_SYNC signal
 - Sync byte
- Supports PID filter mode with 32 PID filters
- Supports six error cases with SKIP/ STOP mode
- Supports TS_CLK filter.
 - TS_CLK maximum frequency
 - o with TS_CLK filter: ~ 1/2 HCLK
 - o without TS_CLK filter: ~ 1/4 HCLK

8.1.2 BROADCAST MODE

TSI supports DVB-H, DVB-T, ISDB-T, T-DMA, and DAB modes. [Figure 8-1](#) shows an example of the broadcasting support scheme.

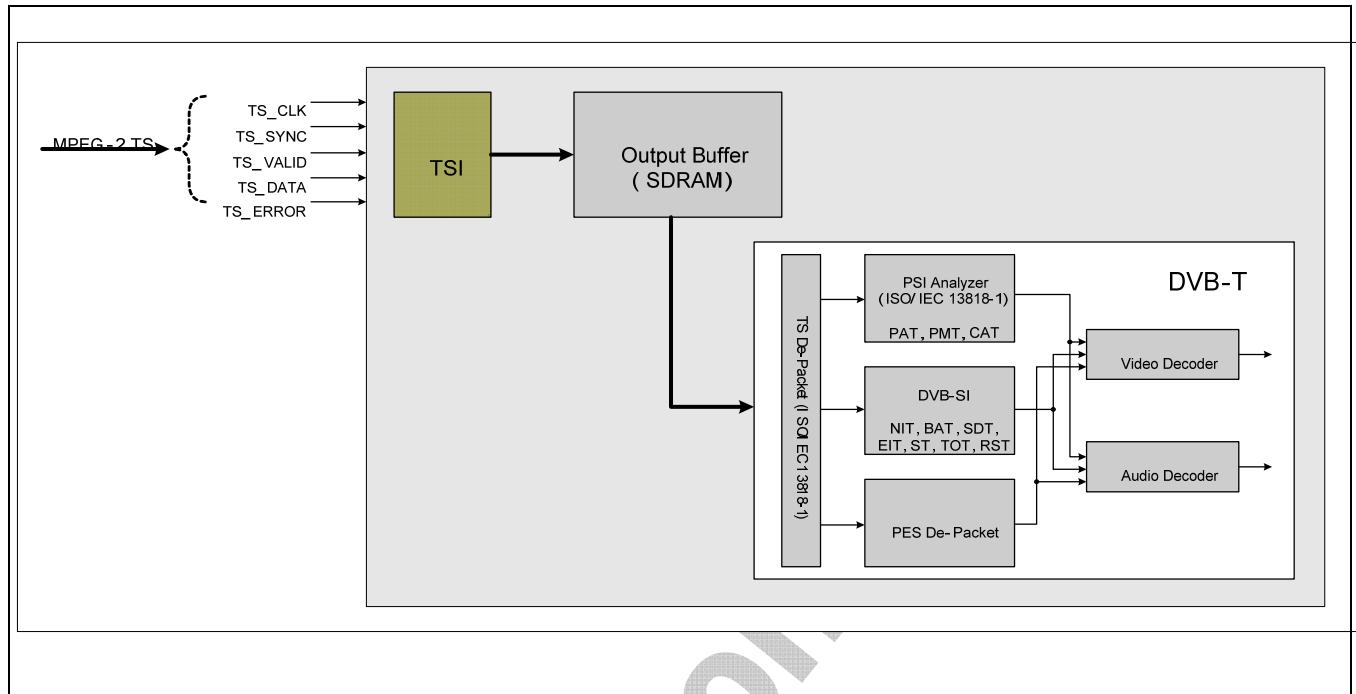


Figure 8-1 Support TSI in the Broadcasting Mode

The MPEG-2 transport stream contains a set of service information for program specific information (PSI). The transport stream is injected from an external channel chip and TSI stores it into output buffer. Then, TS De-Packet, PSI Analyzer, DVB-SI, and PED De-Packet blocks de-multiplex individual PSI from the transport stream and transfer it to audio and video decoders.

PSI comprises of Program Association Table (PAT), Program Map Table (PMT), and Conditional Access Table (CAT), defined in MPEG-2.

- PAT: Transmits the PID information of PMT and NID, and information about various service offering in transmitter.
- PMT: Transmit the PID information of transmit port packet and PCR information transferred with various services.
- CAT: Transmits information for charging broadcasting system used in transmitter.

Table 8-1 shows the characteristics of several mobile-TV standard modes.

Table 8-1 Characteristics of Several Mobile-TV Standard Modes

Mobile TV Standard	Video Codec	Characteristics
DVB-T	MPEG-2	MP@ML 10Mbps, 720*480 @30fps
DVB-H	H.264	Up to Baseline 352*288 @15fps @384kbps
	WMV9	Up to SP @ML 356*288 @15fps @384kbps
T-DMB	H.264	Up to Baseline Profile 352*288 @30fps @768kbps
ISDB-T	H.264	Up to Baseline Profile 320*240 @15fps @384kbps

8.1.3 BLOCK DIAGRAM OF TS INTERFACE

[Figure 8-2](#) shows the overall functional block diagram of TS Interface.

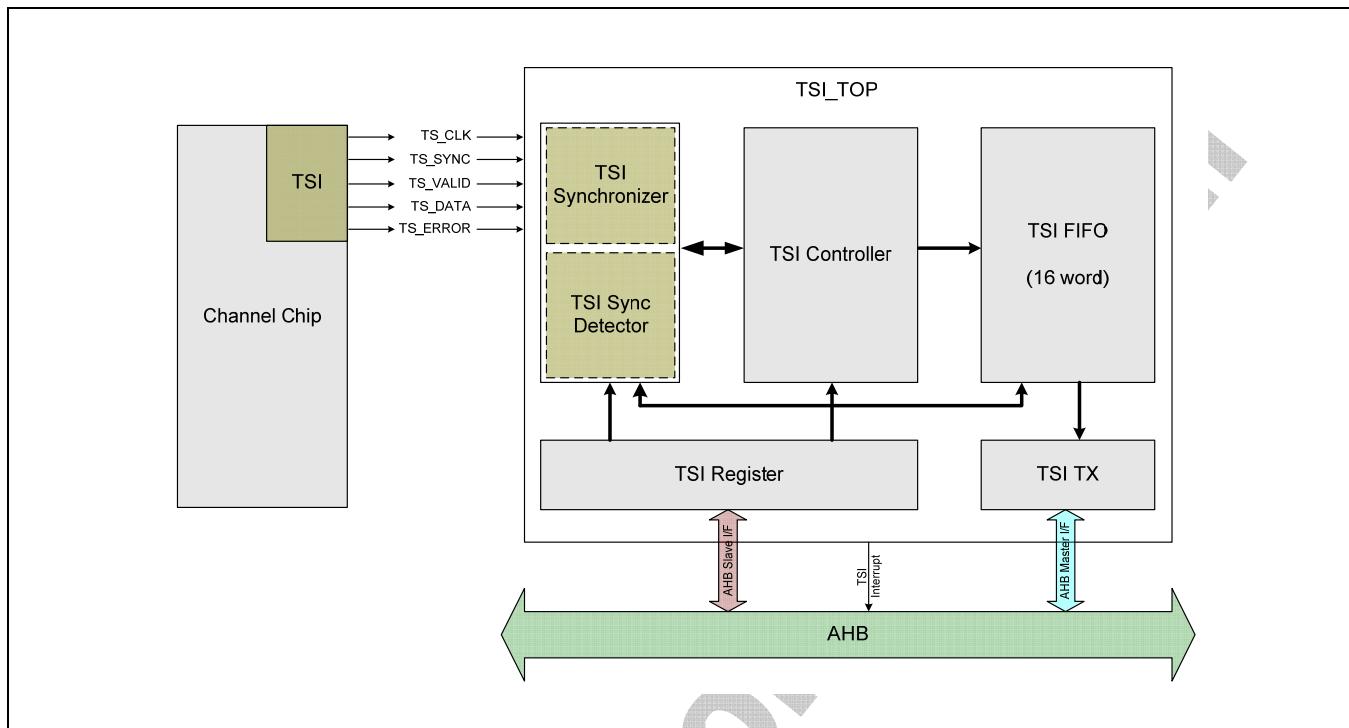


Figure 8-2 TSI Block Diagram

The transport stream data is transferred from an external channel chip via five signals such as TS_CLK, TS_SYNC, TS_VALID, TS_DATA, and TS_ERROR. TSI controller receives transport stream data by capturing the five signals with TSI synchronizer and TSI sync detector. The received transport stream data is stored on TSI FIFO, and TSI TX module transfers the transport stream data into output buffer using AHB master interface. User can control operations of TSI by setting TSI registers via AHB slave interface.

8.1.4 I/O DESCRIPTION OF TSI

Table 8-2 describes the I/O ports of TSI.

Table 8-2 TSI I/O Description

Port Name	I/O	Description (Primary Function)	Source/ Destination
ts_clk	DI	Specifies the TSI system clock (66MHz).	Channel chip/ Buffer
ts_sync	DI	Specifies the TSI synchronization control signal.	Channel chip/ Buffer
ts_val	DI	Specifies the TSI valid signal.	Channel chip/ Buffer
ts_data	DI	Specifies the TSI input data.	Channel chip/ Buffer
ts_error	DI	Specifies the TSI error indicate signal.	Channel chip/ Buffer

, where

- DI - Digital Input Signal

8.1.5 FUNCTIONAL DESCRIPTION

8.1.5.1 TSI Operation

The TS packet has a total of 188 bytes and it consists of the header, adaptation field, and payload. The TS header includes a packet identifier (PID), sync byte, and several control signals (such as transport scrambling, adaptation field, and continuity control). [Figure 8-3](#) shows the transport stream packet data format.

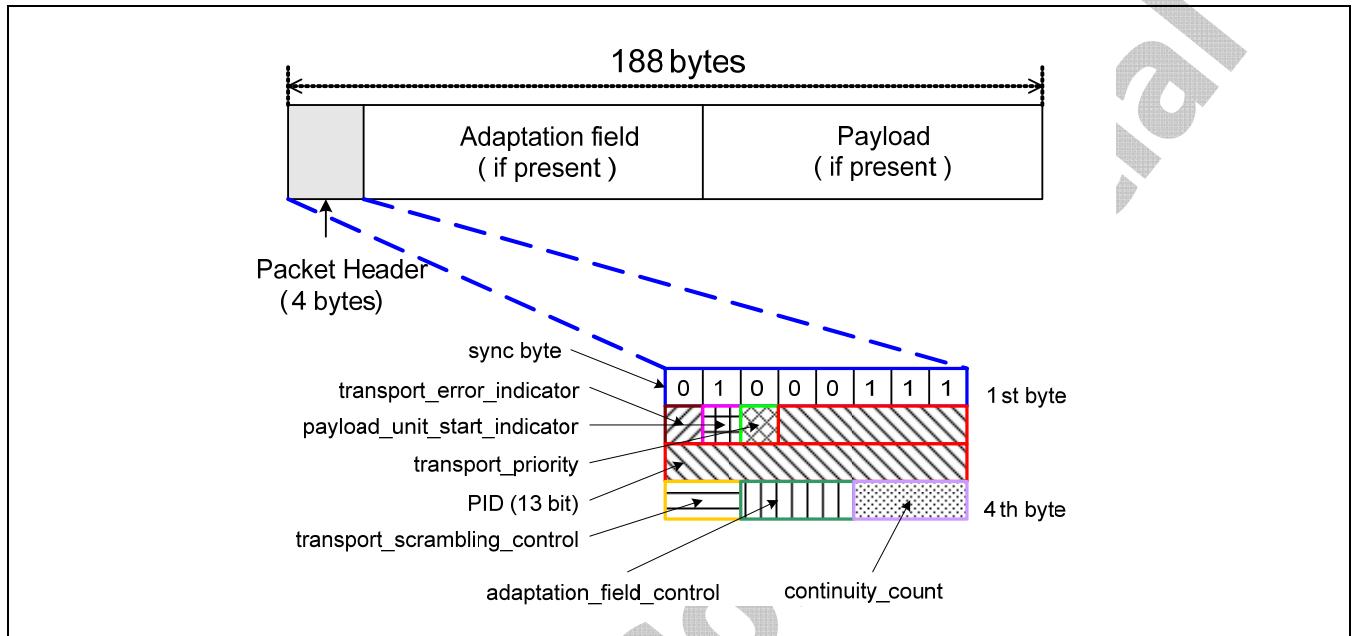


Figure 8-3 Transport Stream Packet Data Format

8.1.5.2 Transport Stream Signals

You can set the transport stream signals (TS_CLK, TS_SYNC, TS_VALID, TS_ERROR, and TS_DATA) as active-high or active-low. The active mode of each signal can be set independently. [Figure 8-4](#) shows the timing diagram of transport stream signals and describes the timing operation of each signal.

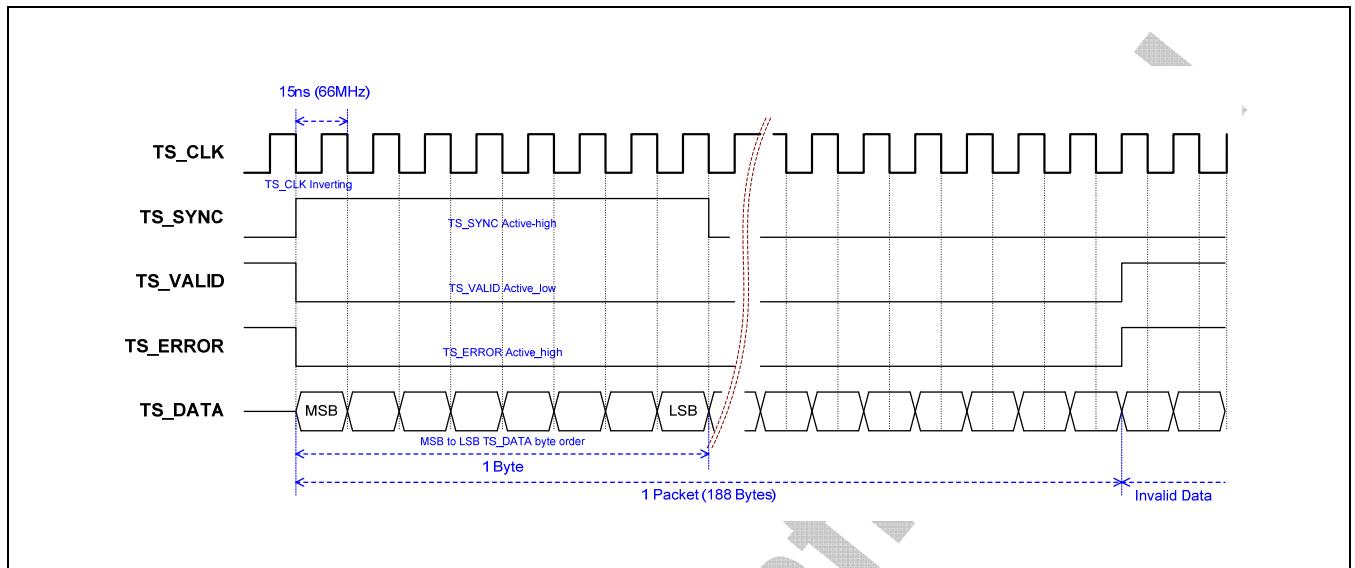


Figure 8-4 Transport Stream Signals

8.1.5.3 Sync Detection

The sync detection of transport stream is done using TS_SYNC signal and sync byte.

8.1.5.3.1 Using TS_SYNC Signal

The sync detection of transport stream using TS_SYNC signal is done by:

- Considering the consecutive 8-bits TS_SYNC signal
- Observing only 1-bit TS_SYNC signal

[Figure 8-5](#) shows the sync detection using TS_SYNC signal.

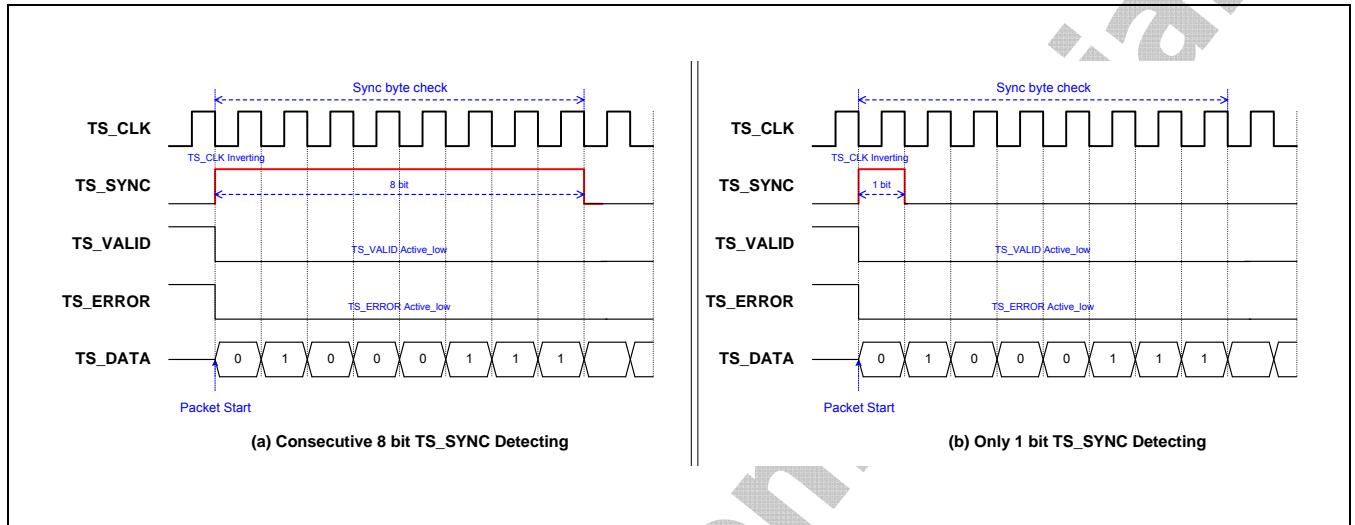


Figure 8-5 Sync Detection using TS_SYNC Signal

8.1.5.3.2 Using Sync Byte

When sync detection using sync byte (0x47) is done, the 0x47 value is transferred in the middle of transport stream. This value can be wrongly recognized as the starting point of the packet. To prevent this situation, when the sync byte is set at TSI sync control register (sync_det_cnt (0~15)), TSI recognizes it as the starting point of the packet.

Three data counters and three current sync detecting counters are used in the TSI module. The current sync detecting counter displays the sync byte that is inputted up to that time. The data detecting counter displays the size of the transferring transport stream after the sync byte is inputted.

Consider that data counter is equal to 187 bytes. If the sync byte is inputted, it initializes the data counter by zero and increases the sync detecting counter by one. On the other hand, if the inputted data is not the sync byte, it disables the data counter and initializes the sync detecting counter by zero.

Consider the enabled total data counter to be less than 187. If the sync byte is inputted, it enables the data counter and increases the related sync detecting counter by one.

If the register value of one among the three sync detecting counters is the same as the value of TSI sync control register (sync_det_cnt), the sync detecting operation is completed.

[Figure 8-6](#) shows the example of the sync detecting operation.

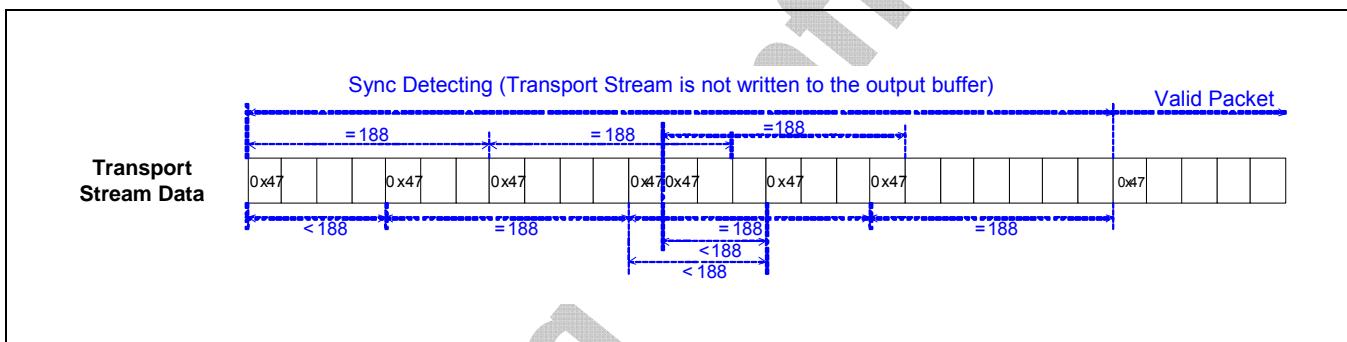


Figure 8-6 Sync Detection using Sync Byte (sync_det_cnt = 3)

Table 8-3 shows the process of the sync detecting operation.

Table 8-3 Sync Detection Process using Sync Byte (sync_det_cnt = 3)

	Condition	Data Count	Sync Detecting Count
-	Sync detecting idle	All data count disabled	All sync detecting count set to 0
1	Sync byte input	Data count1 enabled	csdc1 = 1
2	Sync byte input, data count1 < 187	Data count2 enabled	csdc2 = 1
3	Sync byte input, data count1 == 187	Data count1 set to 0	csdc1 = 2
4	Sync byte input, data count2 == 187	Data count2 set to 0	csdc2 = 2
5	Sync byte input, data count1 < 187, data count2 < 187	Data count3 enabled	csdc3 = 1
6	Sync byte not input, data count1 == 187	Data count1 disabled	csdc1 = 0
7	Sync byte input, data count2 < 187, data count3 < 187	Data count1 enabled	csdc1 = 1
8	Sync byte input, data count2 == 187	Data count2 set to 0	csdc2 = 3
9	Sync byte not input, data count3 == 187	Data count3 disabled	csdc3 = 0
10	Sync byte input, data count2 == 187	Data count2 set to 0	csdc2 == 4. sync detecting complete

8.1.5.4 Error Detection

TSI can generate six errors, as shown in [Table 8-4](#). Each error occurs at the SKIP or STOP mode.

Whenever an error occurs at the SKIP mode, TSI generates an interrupt. Only after fixing the error (Padding or Skip) at the packet where the error occurred, TSI will continue to send data streams to the output buffer.

Whenever an error occurs at the STOP mode, TSI generates an interrupt and stops sending data streams to the output buffer.

In case the error is caused by SKIP or STOP mode in the TSI control register, the interrupt flag in TSI interrupt register is set to 1.

If the TSI mask register is enabled and if an error occurs, the error interrupt signal is sent to the MCU.

Table 8-4 Sync Detection Process using Sync Byte (sync_det_cnt = 3)

#	Error case		Description
1	Sync mismatch (only in TS_SYNC mode)		Sync byte is not received at the start of packet
2	Packet size underflow (only in TS_SYNC mode)		TS_SYNC signal is activated when the packet reception does not end
3	Packet size overflow	TS_SYNC mode	TS_SYNC signal is deactivated at the start of packet
		Sync byte mode	Sync byte is not received at the start of packet
4	TS_ERROR	Bit detecting	TS_ERROR signal is activated when the packet reception is operating.
		Packet detecting	TS_ERROR signal is activated for receiving one packet (Packet data is written to the OUTPUT buffer. Only error flag is generated).
5	TS_CLK timeout (only STOP mode)		TS_CLK is not toggled for n-cycles of HCLK (n is determined by 'TSI CLOCK COUNT register')
6	Internal FIFO full (only STOP mode)		Internal FIFO is filled with received data

[Figure 8-7](#) shows the timing diagram of several TSI error cases. [Figure 8-7](#) -(a) describes the sync mismatch case. Figure 8-7 -(b) describes the packet size underflow case. Figure 8-7 -(c) and Figure 8-7 -(d) describe the packet size overflow case. Figure 8-7 -(e), [Figure 8-7](#) -(f), and [Figure 8-7](#) -(g) describe the ts_error case.

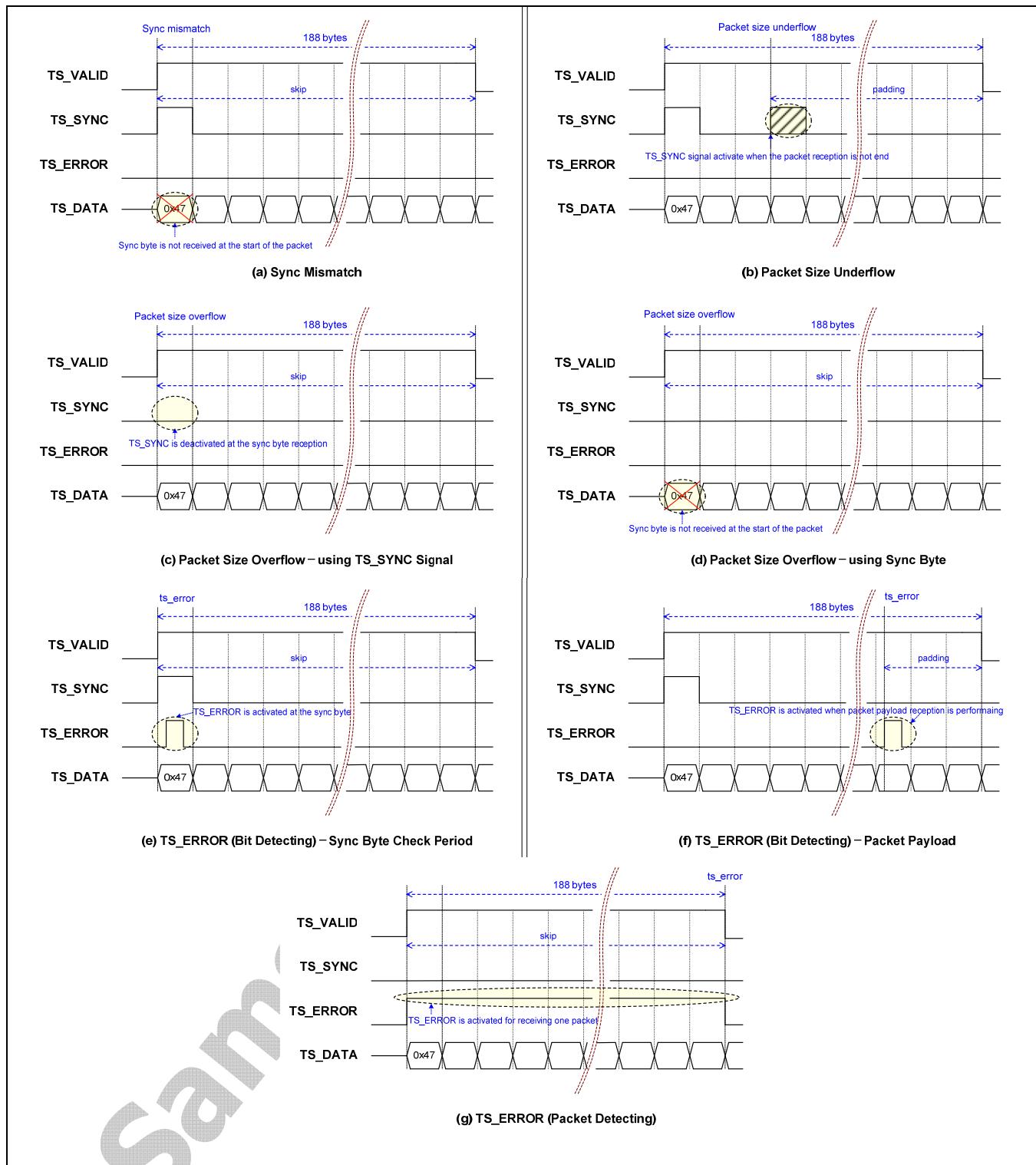


Figure 8-7 TSI Error Cases (with SKIP mode, TS_VALID / TS_SYNC / TS_ERROR is active high)

8.1.5.5 TS_CLK Filter

When HCLK maintains the same level as TS_CLK at the rising edge during two consecutive clock cycles, TS_CLK filter considers TS_CLK as valid.

In case TS_CLK filter is used, a half-period of the TS_CLK should have two or more periods of HCLK. The maximum frequency of TS_CLK will change depending on whether TS_CLK passes two-stage flip-flops (noise filter is disabled) or TS_CLK passes three-stage flip-flops (noise filter is enabled).

If HCLK with an operating frequency of 132MHz is used, the maximum frequency of TS_CLK is limited by 66MHz (without TS_CLK filter) and 33MHz (with TS_CLK filter). [Figure 8-8](#) shows the block diagram of TS_CLK filter.

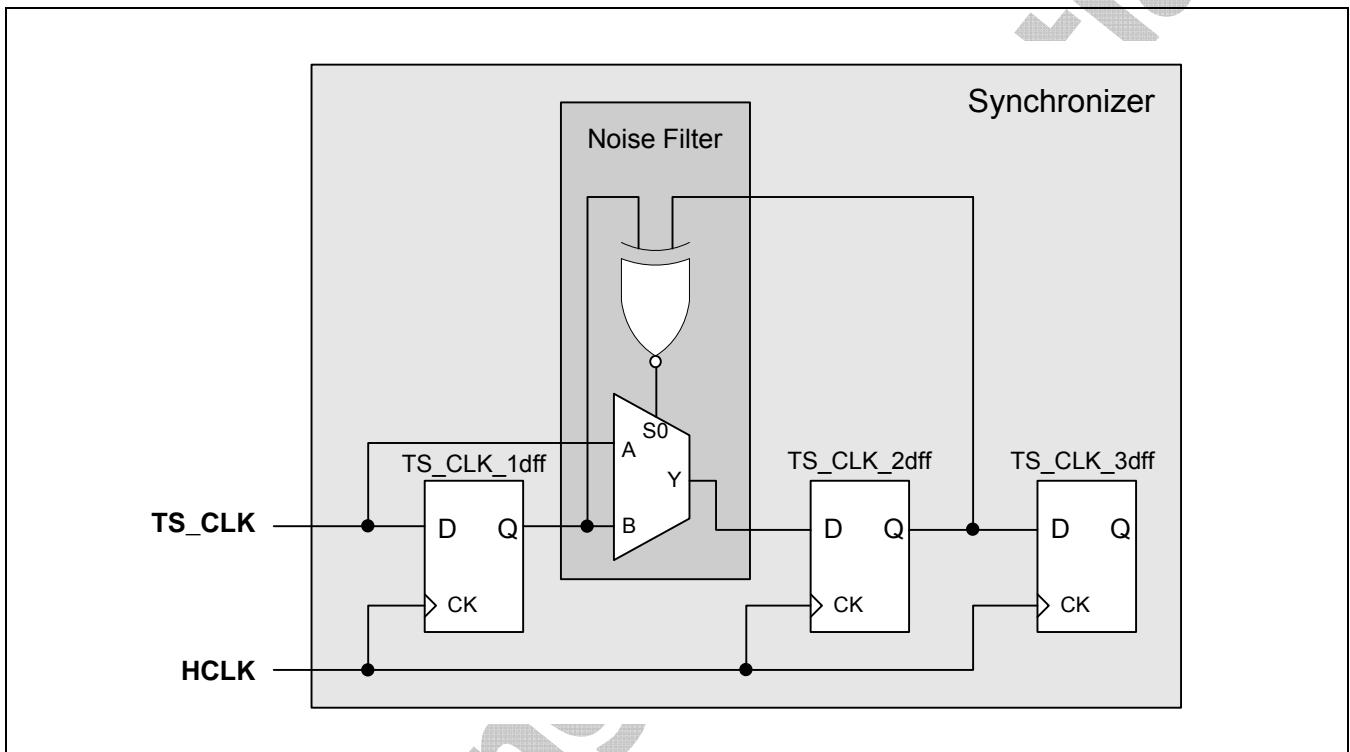


Figure 8-8 Block Diagram of TS_CLK Filter

8.1.5.6 Transport Stream Write

After the transport stream is received from the channel chip, it is stored in internal FIFO (32-word). The TSI sends the transport stream to the output buffer (DRAM) depending on the selected burst length (1-/ 4-/ 8-beat). Using word-aligned address, the TSI sends data streams to the output buffer (SDRAM). One packet size is equal to 47 words. The output buffer size should be equal to a multiple of 47 words (one packet size). If the data is written in the output buffer, the SDRAM full interrupt is generated and the destination address is reloaded in the base address.

8.1.5.7 Program ID filter

TSI supports a Program ID (PID) filter mode that uses 32 PID filters. PID filter can switch on/off independently. Consider the PID filter mode as enabled. In case the PID of transmitted packet header is one of the 32 filters, the transport stream is treated as normal and stored to output buffer. However, if the PID filter mode is disabled, the PID filter value is not checked and all transport streams are recognized as normal.

8.1.5.8 TSI Control FSM

TSI has several operating modes. As shown in [Figure 8-9](#), the TSI can switch from one mode to another according to the condition (state) of control signals.

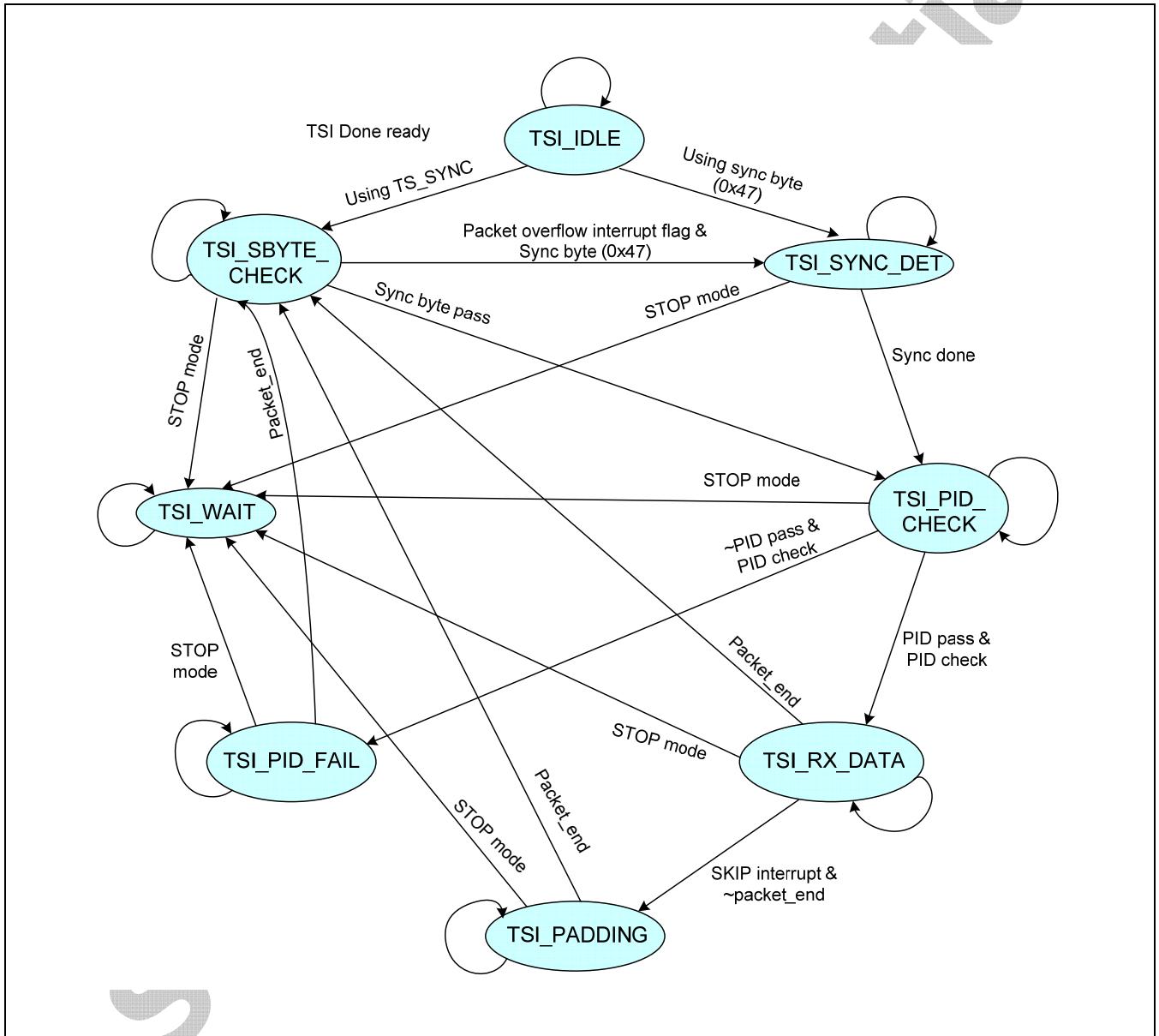


Figure 8-9 TSI Control Finite State Machine (FSM)

8.1.5.9 Shadow Base Address

TSI automatically changes the destination address of output buffer (SDRAM) with base address. When TSI is started (`tsi_on = 1`) or output buffer full interrupt is generated, address value of `TS_BASE` register is set as new destination address to store received packet data.

Example of Shadow Base Address usage:

1. Set the first address in `TS_BASE` register.
2. Start TSI (the first address is set as destination address).
3. Set the second address in `TS_BASE` register right after TSI starts. Output buffer becomes full (the second address is set as new destination address).
4. Set the third address in `TS_BASE` register when output buffer full interrupt is generated. Output buffer becomes full (the third address is set as new destination address).

8.2 REGISTER DESCRIPTION

8.2.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
TS_CLKCON	0xEB40_0000	R/W	Specifies the TSI clock control register.	32'h0000_0002
TS_CON	0xEB40_0004	R/W	Specifies the TSI control register.	32'h1000_0000
TS_SYNC	0xEB40_0008	R/W	Specifies the TSI Sync control register.	32'h0000_00F0
TS_CNT	0xEB40_000C	R/W	Specifies the TSI clock count register.	32'h00FF_FFFF
TS_BASE	0xEB40_0010	R/W	Specifies the TS buffer base address.	32'h0000_0000
TS_SIZE	0xEB40_0014	R/W	Specifies the TS buffer size address.	32'h0000_0000
TS_CADDR	0xEB40_0018	R	Specifies the TS buffer current write address.	32'h0000_0000
TS_INT_MASK	0xEB40_001C	R/W	Specifies the TSI interrupt mask register.	32'h0000_0000
TS_INT	0xEB40_0020	R/W	Specifies the TSI interrupt flag register.	32'h0000_0000
TS_PID0	0xEB40_0024	R/W	Specifies the TSI PID filter0.	32'h0000_0000
TS_PID1	0xEB40_0028	R/W	Specifies the TSI PID filter1.	32'h0000_0000
TS_PID2	0xEB40_002C	R/W	Specifies the TSI PID filter2.	32'h0000_0000
TS_PID3	0xEB40_0030	R/W	Specifies the TSI PID filter3.	32'h0000_0000
TS_PID4	0xEB40_0034	R/W	Specifies the TSI PID filter4.	32'h0000_0000
TS_PID5	0xEB40_0038	R/W	Specifies the TSI PID filter5.	32'h0000_0000
TS_PID6	0xEB40_003C	R/W	Specifies the TSI PID filter6.	32'h0000_0000
TS_PID7	0xEB40_0040	R/W	Specifies the TSI PID filter7.	32'h0000_0000
TS_PID8	0xEB40_0044	R/W	Specifies the TSI PID filter8.	32'h0000_0000
TS_PID9	0xEB40_0048	R/W	Specifies the TSI PID filter9.	32'h0000_0000
TS_PID10	0xEB40_004C	R/W	Specifies the TSI PID filter10.	32'h0000_0000
TS_PID11	0xEB40_0050	R/W	Specifies the TSI PID filter11.	32'h0000_0000
TS_PID12	0xEB40_0054	R/W	Specifies the TSI PID filter12.	32'h0000_0000
TS_PID13	0xEB40_0058	R/W	Specifies the TSI PID filter13.	32'h0000_0000
TS_PID14	0xEB40_005C	R/W	Specifies the TSI PID filter14.	32'h0000_0000
TS_PID15	0xEB40_0060	R/W	Specifies the TSI PID filter15.	32'h0000_0000
TS_PID16	0xEB40_0064	R/W	Specifies the TSI PID filter16.	32'h0000_0000
TS_PID17	0xEB40_0068	R/W	Specifies the TSI PID filter17.	32'h0000_0000
TS_PID18	0xEB40_006C	R/W	Specifies the TSI PID filter18.	32'h0000_0000
TS_PID19	0xEB40_0070	R/W	Specifies the TSI PID filter19.	32'h0000_0000
TS_PID20	0xEB40_0074	R/W	Specifies the TSI PID filter20.	32'h0000_0000
TS_PID21	0xEB40_0078	R/W	Specifies the TSI PID filter21.	32'h0000_0000
TS_PID22	0xEB40_007C	R/W	Specifies the TSI PID filter22.	32'h0000_0000
TS_PID23	0xEB40_0080	R/W	Specifies the TSI PID filter23.	32'h0000_0000
TS_PID24	0xEB40_0084	R/W	Specifies the TSI PID filter24.	32'h0000_0000

Register	Address	R/W	Description	Reset Value
TS_PID25	0xEB40_0088	R/W	Specifies the TSI PID filter25.	32'h0000_0000
TS_PID26	0xEB40_008C	R/W	Specifies the TSI PID filter26.	32'h0000_0000
TS_PID27	0xEB40_0090	R/W	Specifies the TSI PID filter27.	32'h0000_0000
TS_PID28	0xEB40_0094	R/W	Specifies the TSI PID filter28.	32'h0000_0000
TS_PID29	0xEB40_0098	R/W	Specifies the TSI PID filter29.	32'h0000_0000
TS_PID30	0xEB40_009C	R/W	Specifies the TSI PID filter30.	32'h0000_0000
TS_PID31	0xEB40_00A0	R/W	Specifies the TSI PID filter31.	32'h0000_0000
BYTE_SWAP	0xEB40_00BC	R/W	Specifies the TSI tx byte swap enable register for little endian.	32'h0000_0001

8.2.2 TSI REGISTER DESCRIPTION

8.2.2.1 TSI Clock Control Register (TS_CLKCON, R/W, Address = 0xEB40_0000)

TS_CLKCON	Bit	Description	R/W	Initial State
Reserved	[31:2]	-	-	-
TSI clock down ready	[1]	If this field is set to 1, TSI block is ready to be down. 0 = Not ready 1 = Ready	R	1'b1
TSI on	[0]	TSI on/off. 0 = TSI off 1 = TSI on	R/W	1'b0

8.2.2.2 TSI Control Register (TS_CON, R/W, Address = 0xEB40_0004)

TS_CON	Bit	Description	R/W	Initial State
TSI_SWRESET	[31]	Initializes all registers and states of TSI block. 0 = No effect 1 = Reset (equals to H/W reset)	W	-
TS_CLK filter mode	[30]	Specifies the filter mode. 0 = Off 1 = On (use ts_clk filter. max. ts_clk frequency is 1/4 HCLK)	R/W	1'b0
TSI burst length	[29:28]	Sets the TSI burst length. 00 = 1-beat burst 01 = 4-beat burst 10 = 8-beat burst 11 = reserved.	R/W	2'b01
output_buffer_full_int_mode	[27]	Sets the output buffer full interrupt mode. 0 = Disable 1 = Enable	R/W	1'b0
int_fifo_full_int_mode	[26]	Sets the internal FIFO full interrupt mode. 0 = Disable 1 = Enable	R/W	1'b0
sync_mismatch_int_mode	[25:24]	Sets the sync mismatch interrupt mode. 0x = Disable 10 = Enable with skip mode 11 = Enable with stop mode	R/W	2'b00
psuf_int_mode	[23:22]	Sets the packet size underflow interrupt mode. 0x = Disable 10 = Enable with skip mode 11 = Enable with stop mode	R/W	2'b00
psof_int_mode	[21:20]	Sets the packet size overflow interrupt mode. 0x = Disable 10 = Enable with skip mode 11 = Enable with stop mode	R/W	2'b00
ts_clk_timeout_int_mode	[19]	Sets the TS_CLK timeout interrupt mode. 0 = Disable 1 = Enable	R/W	2'b00
ts_error_int_mode	[18:16]	Sets the TS_ERROR interrupt mode. 0xx = Disable 100 = Enable with skip mode (detecting size: bit) 101 = Enable with stop mode (detecting size: bit) 110 = Enable with skip mode (detecting size: packet) 111 = Enable with stop mode (detecting size: packet)	R/W	3'b000
padding_pattern	[15:8]	Specifies the Padding pattern.	R/W	8'h00
pid_filter_mode	[7]	Specifies the PID filtering mode. 0 = Bypass mode 1 = Filtering mode	R/W	1'b0

TS_CON	Bit	Description	R/W	Initial State
ts_error_active	[6]	Specifies the TS_ERROR active mode. 0 = Active high 1 = Active low	R/W	1'b0
data_byte_order	[5]	Specifies the TS_DATA byte ordering. 0 = MSB to LSB 1 = LSB to MSB	R/W	1'b0
ts_valid_active	[4]	Specifies the TS_VALID active mode. 0 = Active high 1 = Active low	R/W	1'b0
ts_sync_active	[3]	Specifies the TS_SYNC active mode. 0 = Active high 1 = Active low	R/W	1'b0
ts_clk_invert	[2]	Specifies the TS_CLK inverting mode. 0 = Non-inverting (falling-edge data fetch) 1 = Inverting (ringing-edge data fetch)	R/W	1'b0
Reserved	[1:0]	-	-	-

8.2.2.3 TSI SYNC Control Register (TS_SYNC, R/W, Address = 0xEB40_0008)

TS_SYNC	Bit	Description	R/W	Initial State
Reserved	[31:20]	-	R	-
sync_csd3	[19:16]	Specifies the Current sync detecting count3.	R	4'h0
sync_csd2	[15:12]	Specifies the Current sync detecting count2.	R	4'h0
sync_csd1	[11:8]	Specifies the Current sync detecting count1.	R/W	4'h0
sync_det_cnt	[7:4]	Specifies the Sync detecting count. If the sync detecting mode uses sync byte, this field indicates the initial detecting count.	-	4'hF
Reserved	[3:2]	-	R/W	-
sync_det_mode	[1:0]	Specifies the Sync detecting mode. 00 = Using TS_SYNC (detecting consecutive 8-bit) 01 = Using TS_SYNC (detecting only 1-bit) 1x = Using sync byte (0x47)		2'b00

8.2.2.4 TSI Clock Count Register (TS_CNT, R/W, Address = 0xEB40_000C)

TS_CNT	Bit	Description	R/W	Initial State
ts_clk_error_cnt	[31:0]	Specifies the TS_CLK timeout period. If the ts_clk does not toggle for n-times of hclk, ts_clk_timeout interrupt is generated. TS_CLK timeout period: HCLK(7.5ns) * n	R/W	32'h00FF_FFFF

8.2.2.5 TS Buffer Base Address Register (TS_BASE, R/W, Address = 0xEB40_0010)

TS_BASE	Bit	Description	R/W	Initial State
ts_base_addr	[31:2]	Specifies the TS buffer base address (word aligned).	R/W	30'h0
-	[1:0]	-	-	-

8.2.2.6 TS Buffer Size Address Register (TS_SIZE, R/W, Address = 0xEB40_0014)

TS_SIZE	Bit	Description	R/W	Initial State
Reserved	[31:16]	-	-	-
ts_buffer_size	[15:0]	This field should be 47-word (188-byte) * n , where n specifies the stream packet word count (0~348 word). If the buffer is full, the buffer write address is cleared to the buffer base address.	R/W	16'h0

8.2.2.7 TS Buffer Current Address Register (TS_CADDR, R, Address = 0xEB40_0018)

TS_CADDR	Bit	Description	R/W	Initial State
ts_caddr	[31:2]	Specifies the TS buffer current write address.	R	30'h0
-	[1:0]	-	-	-

8.2.2.8 TSI Interrupt Mask Register (TS_INT_MASK, R/W, Address = 0xEB40_001C)

TS_INT_MASK	Bit	Description	R/W	Initial State
Reserved	[31:8]	-	-	-
dma_complete_mask	[7]	Specifies the DMA complete interrupt mask. 0 = Masking 1 = Not masking	R/W	1'b0
output_buffer_full_mask	[6]	Specifies the TSI interrupt mask: output buffer full. 0 = Masking 1 = Not masking	R/W	1'b0
int_fifo_full_mask	[5]	Specifies the TSI interrupt mask: internal FIFO full. 0 = Masking 1 = Not masking	R/W	1'b0
sync_mismatch_mask	[4]	Specifies the TSI interrupt mask: sync mismatch. 0 = Masking 1 = Not masking	R/W	1'b0
packet_size_underflow_mask	[3]	Specifies the TSI interrupt mask: packet size underflow. 0 = Masking 1 = Not masking	R/W	1'b0
packet_size_overflow_mask	[2]	Specifies the TSI interrupt mask: packet size overflow 0 = Masking 1 = Not masking	R/W	1'b0
TS_CLK_mask	[1]	Specifies the TSI interrupt mask: TS_CLK 0 = Masking 1= Not masking	R/W	1'b0
TS_ERROR_mask	[0]	Specifies the TSI interrupt mask: TS_ERROR 0 = Masking 1 = Not masking	R/W	1'b0

8.2.2.9 TSI Interrupt Flag Register (TS_INT, R/W, Address = 0xEB40_0020)

TS_INT	Bit	Description	R/W	Initial State
Reserved	[31:8]	-	-	-
dma_complete_flag	[7]	Specifies the Dma transfer complete flag. 0 = Interrupt is not generated 1 = Interrupt is generated (Writing '1' clears this field and writing '0' has no effect.)	R/W	1'b0
output_buffer_full_flag	[6]	Specifies the Output buffer full interrupt flag. 0 = Interrupt is not generated 1 = Interrupt is generated (Writing '1' clears this field and writing '0' has no effect.)	R/W	1'b0
int_fifo_full_flag	[5]	Specifies the Internal FIFO full interrupt flag. 0 = Interrupt is not generated 1 = Interrupt is generated (Writing '1' clears this field and writing '0' has no effect.)	R/W	1'b0
sync_mismatch_flag	[4]	Specifies the Sync mismatch interrupt flag. 0 = Interrupt is not generated 1 = Interrupt is generated (Writing '1' clears this field and writing '0' has no effect.)	R/W	1'b0
psuf_flag	[3]	Specifies the Packet underflow interrupt flag. 0 = Interrupt is not generated 1 = Interrupt is generated (Writing '1' clears this field and writing '0' has no effect.)	R/W	1'b0
psof_flag	[2]	Specifies the Packet overflow interrupt flag. 0 = Interrupt is not generated 1 = Interrupt is generated (Writing '1' clears this field and writing '0' has no effect.)	R/W	1'b0
ts_clk_timeout_flag	[1]	Specifies the TS_CLK timeout interrupt flag. 0 = Interrupt is not generated 1 = Interrupt is generated (Writing '1' clears this field and writing '0' has no effect.)	R/W	1'b0
ts_error_flag	[0]	Specifies the TS_ERROR interrupt flag. 0 = Interrupt is not generated 1 = Interrupt is generated (Writing '1' clears this field and writing '0' has no effect.)	R/W	1'b0

8.2.2.10 TSI PID Filter0 Address Register (TS_PID0, R/W, Address = 0xEB40_0024)

TS_PID0	Bit	Description	R/W	Initial State
Reserved	[31:14]	-	-	-
pid0_en	[13]	Specifies the PID filter0. 0 = Disable 1 = Enable	R/W	1'b0
pid0_value	[12:0]	Specifies the PID0 value. If an input stream's PID is different from this value, then the stream is ignored.	R/W	13'h0

8.2.2.11 TSI PID Filter1 Address Register (TS_PID1, R/W, Address = 0xEB40_0028)

8.2.2.12 TSI PID Filter2 Address Register (TS_PID2, R/W, Address = 0xEB40_002C)

8.2.2.13 TSI PID Filter3 Address Register (TS_PID3, R/W, Address = 0xEB40_0030)

8.2.2.14 TSI PID Filter4 Address Register (TS_PID4, R/W, Address = 0xEB40_0034)

8.2.2.15 TSI PID Filter5 Address Register (TS_PID5, R/W, Address = 0xEB40_0038)

8.2.2.16 TSI PID Filter6 Address Register (TS_PID6, R/W, Address = 0xEB40_003C)

8.2.2.17 TSI PID Filter7 Address Register (TS_PID7, R/W, Address = 0xEB40_0040)

8.2.2.18 TSI PID Filter8 Address Register (TS_PID8, R/W, Address = 0xEB40_0044)

8.2.2.19 TSI PID Filter9 Address Register (TS_PID9, R/W, Address = 0xEB40_0048)

8.2.2.20 TSI PID Filter10 Address Register (TS_PID10, R/W, Address = 0xEB40_004C)

8.2.2.21 TSI PID Filter11 Address Register (TS_PID11, R/W, Address = 0xEB40_0050)

8.2.2.22 TSI PID Filter12 Address Register (TS_PID12, R/W, Address = 0xEB40_0054)

8.2.2.23 TSI PID Filter13 Address Register (TS_PID13, R/W, Address = 0xEB40_0058)

8.2.2.24 TSI PID Filter14 Address Register (TS_PID14, R/W, Address = 0xEB40_005C)

8.2.2.25 TSI PID Filter15 Address Register (TS_PID15, R/W, Address = 0xEB40_0060)

8.2.2.26 TSI PID filter16 Address Register (TS_PID16, R/W, Address = 0xEB40_0064)

8.2.2.27 TSI PID Filter17 Address Register (S_PID17, R/W, Address = 0xEB40_0068)

8.2.2.28 TSI PID Filter18 Address Register (TS_PID18, R/W, Address = 0xEB40_006C)

8.2.2.29 TSI PID Filter19 Address Register (TS_PID19, R/W, Address = 0xEB40_0070)

8.2.2.30 TSI PID Filter20 Address Register (TS_PID20, R/W, Address = 0xEB40_0074)

8.2.2.31 TSI PID Filter21 Address Register (TS_PID21, R/W, Address = 0xEB40_0078)

8.2.2.32 TSI PID Filter22 Address Register (TS_PID22, R/W, Address = 0xEB40_007C)

8.2.2.33 TSI PID Filter23 Address Register (TS_PID23, R/W, Address = 0xEB40_0080)

8.2.2.34 TSI PID Filter24 Address Register (TS_PID24, R/W, Address = 0xEB40_0084)

8.2.2.35 TSI PID Filter25 Address Register (TS_PID25, R/W, Address = 0xEB40_0088)

8.2.2.36 TSI PID Filter26 Address Register (TS_PID26, R/W, Address = 0xEB40_008C)

8.2.2.37 TSI PID Filter27 Address Register (TS_PID27, R/W, Address = 0xEB40_0090)

8.2.2.38 TSI PID Filter28 Address Register (TS_PID28, R/W, Address = 0xEB40_0094)

8.2.2.39 TSI PID Filter29 Address Register (TS_PID29, R/W, Address = 0xEB40_0098)**8.2.2.40 TSI PID Filter30 Address Register (TS_PID30, R/W, Address = 0xEB40_009C)****8.2.2.41 TSI PID Filter31 Address Register (TS_PID31, R/W, Address = 0xEB40_00A0)****8.2.2.42 TSI TX Byte SWAP Register (BYTE_SWAP, R/W, Address = 0xEB40_00BC)**

BYTE_SWAP	Bit	Description	R/W	Initial State
Reserved	[31:0]	-	R/W	-
byte_swap	[0]	Specifies the TSI tx byte swap enable register for little endian. 0 = Disable - big endian 1 = Enable - little endian		1'b1

Section 9

MULTIMEDIA

Table of Contents

1 Display Controller	1-1
1.1 Overview of Display Controller.....	1-1
1.2 Key Features of the Display Controller	1-2
1.3 Functional Description of Display Controller.....	1-4
1.3.1 Brief Description of the Sub-Block.....	1-4
1.3.2 Data Flow.....	1-4
1.3.3 Overview of the Color Data	1-7
1.3.4 Color Space Conversion (CSC).....	1-22
1.3.5 Palette Usage	1-24
1.3.6 Window Blending	1-26
1.3.7 Image Enhancement.....	1-35
1.3.8 VTIME Controller Operation	1-40
1.3.9 Setting of Commands	1-42
1.3.10 Virtual Display.....	1-45
1.3.11 RGB Interface Spec.....	1-46
1.3.12 LCD Indirect i80 System Interface.....	1-53
1.4 Programmer's Model.....	1-57
1.4.1 Overview of Programmer's Model	1-57
1.5 Register Description.....	1-58
1.5.1 Register Map	1-58
1.5.2 Palette Memory (PalRam)	1-66
1.5.3 Gamma LUT Data.....	1-133
2 Camera Interface	2-1
2.1 Overview of Camera Interface	2-1
2.2 Key Features of CAMIF	2-3
2.3 External Interface	2-5
2.4 Input/ Output Description	2-6
2.5 Timing Diagram and Data Alignment of Camera	2-7
2.5.1 Timing Diagram of ITU Camera.....	2-7
2.5.2 MIPI CSI Data Alignment from MIPI Camera.....	2-10
2.6 External Connection Guide	2-11
2.7 Camera Interface Operation	2-12
2.7.1 Input/ Output DMA Ports	2-12
2.7.2 Clock Domain	2-13
2.7.3 Frame Memory Hierarchy	2-14
2.7.4 Memory Storing Method	2-14
2.7.5 Timing Diagram for Register Setting	2-15
2.7.6 Timing Diagram for last IRQ	2-18
2.7.7 Timing Diagram for IRQ (Memory data Scaling mode)	2-19
2.7.8 Input DMA Feature	2-20
2.7.9 Camera Interlace Input Support	2-21
2.8 Register Description.....	2-22
2.8.1 Register Map	2-22
2.8.2 Register Setting Guide for Scaler	2-46

3 MIPI DSIM	3-1
3.1 Architecture of MIPI DSM	3-1
3.1.1 KEY Features of MIPI DSM	3-1
3.1.2 Block Diagram of MIPI DSI System.....	3-2
3.1.3 Interfaces and Protocol.....	3-5
3.1.4 Configuration	3-13
3.1.5 Dual Display Versus Single Display	3-13
3.1.6 PLL	3-13
3.1.7 Buffer	3-13
3.2 I/O Description	3-14
3.3 Register Description.....	3-15
3.3.1 Register Map	3-15
3.4 DPHY PLL Control	3-31
3.4.1 PMS Setting SAMPLE for MIPI PLL	3-31
4 MIPI CSIS.....	4-1
4.1 Overview of MIPI CSIS	4-1
4.2 Block Diagram.....	4-2
4.3 Interface and Protocol.....	4-3
4.4 Data Format	4-4
4.4.1 Data Alignment	4-4
4.4.2 YUV422 8-bit Order	4-4
4.5 I/O Description	4-5
4.6 Register Description.....	4-6
4.6.1 Register Map	4-6
5 G3D	5-1
5.1 Overview of G3D	5-1
5.1.1 Key Features of G3D	5-1
5.1.2 3D Features in G3D	5-2
5.1.3 USSE Features in G3D	5-3
5.1.4 2D Features in G3D	5-4
5.1.5 Block Diagram of SGX540	5-5
5.1.6 Block Diagram of Integration Information	5-10
5.1.7 Register Map	5-11
6 Multi Format Codec	6-1
6.1 Introduction	6-1
6.1.1 Supported Standards.....	6-1
6.1.2 Features.....	6-3
6.1.3 Target Performance and Functions	6-4
6.2 Hardware Overview	6-5
6.2.1 Block Diagram	6-5
6.2.2 Frame Memory	6-7
6.3 Register Description.....	6-10
6.3.1 Register Map	6-10
6.3.2 Control Registers	6-17
6.3.3 Codec Registers	6-35
6.3.4 Encoding Registers.....	6-54
6.4 Shared Memory Interface	6-63

6.4.1 Host Interface	6-63
6.4.2 Shared Memory Structure	6-64
6.5 Metadata Interface	6-78
6.5.1 Shared Memory Interface for Decoders	6-79
6.5.2 Shared Memory Interface for Encoders	6-83
6.6 Appendix	6-84
6.6.1 Summary of Buffer Requirements	6-84
6.6.2 Batch Encoding Interface	6-85
7 TVOUT & Video DAC	7-1
7.1 Overview of Tvout and Video DAC	7-1
7.2 Key Features of Tvout and Video DAC	7-1
7.2.1 I/O and Control	7-1
7.2.2 Video Standard Compliances for CVBS:	7-1
7.2.3 Ancillary Data Insertion	7-1
7.2.4 Post Processing	7-1
7.3 Data Flow	7-2
7.4 Timing Generation (TG Module)	7-3
7.4.1 525/60 Hz	7-3
7.4.2 625/50 Hz	7-3
7.4.3 3.579545 MHz	7-3
7.4.4 4.43361875 MHz	7-3
7.4.5 3.57561149 MHz	7-3
7.4.6 3.58205625 MHz	7-3
7.5 Anti Aliasing Filter (AAF module)	7-8
7.6 Ancillary Data insertion (VBI module)	7-11
7.7 WaveForm Generation and Chroma Modulation (CVBS module)	7-14
7.8 Illegal Color Compensation (CVBS module)	7-17
7.9 Oversampling & DAC Compensation Filter (Osf Module)	7-18
7.10 Register Control (CTRL Module)	7-21
7.11 I/O DesCription	7-21
7.12 Register Description	7-22
7.12.1 Register Map	7-22
7.12.2 Shadow Registers	7-58
7.13 Video DAC	7-60
7.13.1 General Description	7-60
7.13.2 Features	7-60
7.13.3 Core Port Description	7-60
7.13.4 Full Scale Voltage Modification	7-61
7.14 Appendix	7-62
7.14.1 Vertical Bar Pheomenon	7-62
8 Video Processor	8-1
8.1 Overview of Video Processor	8-1
8.1.1 Key Features of Video Processor	8-1
8.2 Block Diagram of Video Processor	8-2
8.3 Function Description of Video Processor	8-3
8.3.1 BOB in Video Processor	8-3
8.3.2 Nterlace to Progressive Conversion	8-4
8.4 Register Description	8-5
8.4.1 Register Map	8-5
8.4.2 The Idea of Poly-phase Filtering in Video Processor	8-45

9 Mixer	9-1
9.1 Overview of Mixer	9-1
9.1.1 Key Features of Mixer	9-1
9.1.2 Block Diagram of Mixer	9-3
9.1.3 Video Clock Relation	9-4
9.2 Register Description	9-5
9.2.1 Register Map	9-5
9.2.2 Shadow Registers (read only).....	9-8
9.3 Layers	9-25
9.3.1 Video Layer.....	9-27
9.3.2 Graphic Layer	9-27
9.3.3 Blank Pixel.....	9-28
9.3.4 Source Data in Memory.....	9-28
9.3.5 Background Layer.....	9-29
10 High-Definition Multimedia Interface	10-1
10.1 Overview of High-Definition Multimedia Interface	10-1
10.1.1 Key Features of HDMI	10-1
10.1.2 Block Diagram of HDMI	10-2
10.1.3 Block Diagram of HDMI SUB-System in S5PV210	10-3
10.1.4 Block Diagram of HDCP Key Management.....	10-4
10.1.5 Video Input Timing Guide for HDMI Timing Generator	10-5
10.1.6 HDMI PHY Configuration.....	10-7
10.1.7 Selected i2C register control	10-10
10.1.8 I/O Description of HDMI PHY	10-11
10.1.9 BLOCK Diagram of Clock Strategy for HDMI TX	10-12
10.2 SPDIF (auxiliary information).....	10-14
10.2.1 Frame Format	10-14
10.3 Registers Description	10-20
10.3.1 Register Map	10-21
10.3.2 Control Register.....	10-31
10.3.3 HDMI Core Register	10-35
10.3.4 SPDIF Register.....	10-75
10.3.5 I2S Register	10-90
10.3.6 Timing Generator Register (TG Configure/Status Register).....	10-100
11 Image Rotator	11-1
11.1 Overview of Image Rotator	11-1
11.2 Key Features of Image Rotator.....	11-1
11.3 Block Diagram of Image Rotator.....	11-2
11.4 Supported Image Rotation Functions	11-3
11.5 Image Rotation with Windows Offset.....	11-4
11.6 Programming Guide	11-5
11.6.1 Register Setting	11-5
11.6.2 Restrictions on the Image Size.....	11-5
11.7 Register Description.....	11-6
11.7.1 Register Map	11-6

12 JPEG.....12-1

12.1 Overview of JPEG Codec	12-1
12.2 Key Features of JPEG Codec.....	12-1
12.3 Block Diagram of JPEG Codec.....	12-2
12.4 Block Diagram In/Out Data Format.....	12-3
12.4.1 Control Circuit and AHB Interface	12-3
12.4.2 DCT/ Quantization	12-3
12.4.3 Huffman Coder and Marker Process	12-3
12.4.4 Quantization Table	12-3
12.4.5 Huffman Table	12-3
12.4.6 Performance	12-3
12.5 Description of Supported Color Format	12-4
12.5.1 In Compression Mode	12-4
12.5.2 In Decompression Mode.....	12-4
12.6 Process	12-6
12.6.1 Register Access.....	12-6
12.6.2 Table Access	12-6
12.6.3 Starting Process	12-7
12.6.4 Process for Image Size	12-8
12.6.5 Process for Input Stream Size.....	12-8
12.6.6 Interrupt Signal	12-9
12.6.7 Interrupt Setting	12-9
12.6.8 S/W Reset.....	12-9
12.6.9 Marker Process.....	12-10
12.6.10 Bitstream of Compressed File	12-10
12.6.11 JPEG Compression Flow	12-11
12.6.12 Jpeg Decompression Flow	12-12
12.7 Register Description.....	12-13
12.7.1 Register Map	12-13
12.7.2 JPEG Huffman and Quantization Register Tables.....	12-26

13 G2D13-1

13.1 Introduction	13-1
13.2 Features	13-1
13.3 Color Format Conversion.....	13-2
13.4 Rendering Pipeline.....	13-3
13.4.1 Primitive Drawing.....	13-3
13.4.2 Rotation and Addressing Direction (Flip).....	13-5
13.4.3 Clipping.....	13-7
13.4.4 Color Key	13-7
13.4.5 Raster Operation	13-8
13.4.6 Alpha Blending.....	13-9
13.5 Register Description.....	13-10
13.5.1 Register Map	13-10
13.5.2 General Registers.....	13-13
13.5.3 Command Registers	13-15
13.5.4 Parameter Setting Registers (Rotation & Direction).....	13-16
13.5.5 Parameter Setting Registers (Source)	13-17
13.5.6 Parameter Setting Registers (Destination).....	13-19
13.5.7 Parameter Setting Registers (Pattern)	13-21
13.5.8 Parameter Setting Registers (Mask)	13-23

13.5.9 Parameter Setting Registers (Clipping Window)	13-24
13.5.10 Parameter Setting Registers (ROP & Alpha Setting)	13-25
13.5.11 Parameter Setting Registers (Color)	13-26
13.5.12 Parameter Setting Registers (Color Key)	13-27

Samsung Confidential

List of Figures

Figure Number	Title	Page Number
Figure 1-1	Block Diagram of Display Controller	1-1
Figure 1-2	Block Diagram of the Data Flow	1-5
Figure 1-3	Block Diagram of the Interface	1-6
Figure 1-4	16BPP (5:6:5) Display Types.....	1-21
Figure 1-5	Blending Equation.....	1-28
Figure 1-6	Blending Diagram	1-30
Figure 1-7	Blending Factor Decision	1-31
Figure 1-8	Color-Key Function Configurations.....	1-32
Figure 1-9	Blending and Color-Key Function	1-33
Figure 1-10	Blending Decision Diagram	1-34
Figure 1-11	Image Enhancement Flow	1-35
Figure 1-12	Image Enhancement Flow	1-36
Figure 1-13	Hue Coefficient Decision	1-37
Figure 1-14	Hue Control Block Diagram	1-38
Figure 1-15	Example1. RGBPSEL==1'b0, RGB_SKIP==1'b1, PIXCOMPEN_DIR==1'b0.....	1-39
Figure 1-16	Example2. RGBPSEL==1'b1, PIXCOMPEN_DIR==1'b0	1-39
Figure 1-17	Example3. RGBPSEL==1'b1, PIXCOMPEN_DIR==1'b1	1-39
Figure 1-18	Sending Command	1-43
Figure 1-19	Example of Scrolling in Virtual Display	1-45
Figure 1-20	LCD RGB Interface Timing	1-46
Figure 1-21	LCD RGB Interface Timing (RGB parallel).....	1-47
Figure 1-22	LCD RGB Interface Timing (RGB skip)	1-48
Figure 1-23	LCD RGB Interface Timing (RGB serial, Dummy disable).....	1-49
Figure 1-24	LCD RGB Interface Timing (RGB serial, Dummy insertion).....	1-50
Figure 1-25	LCD RGB Output Order.....	1-51
Figure 1-26	Delta Structure and LCD RGB Interface Timing	1-52
Figure 1-27	Indirect i80 System Interface WRITE Cycle Timing	1-54
Figure 2-1	Subset of Visual System in S5PV210.....	2-2
Figure 2-2	Camera Interface Overview	2-4
Figure 2-3	ITU-R BT 601 Input Timing Diagram	2-7
Figure 2-4	ITU-R BT 601 Interlace Handling Diagram.....	2-7
Figure 2-5	ITU-R BT 656 Input Timing Diagram	2-8
Figure 2-6	Sync Signal Timing Diagram	2-9
Figure 2-7	JPEG Input Timing Diagram (ITU 601 and Freerun Clock Mode).....	2-9
Figure 2-8	MIPI CSI DATA Alignment.....	2-10
Figure 2-9	IO Connection Guide	2-11
Figure 2-10	Input / Output DMA Ports.....	2-12
Figure 2-11	CAMIF Clock Generation.....	2-13
Figure 2-12	Ping-pong Memory Hierarchy	2-14
Figure 2-13	Memory Storing Style	2-15
Figure 2-14	Timing Diagram for Camera Input Register setting	2-16
Figure 2-15	Timing Diagram for DMA input Register Setting.....	2-17
Figure 2-16	Timing Diagram for Last IRQ (LastIRQEn is Enabled).....	2-18
Figure 2-17	Diagram for Last IRQ (LastIRQEn is Disabled) and Timing Requirement	2-18
Figure 2-18	Timing Diagram for IRQ (Input DMA Path).....	2-19
Figure 2-19	Input DMA or External Camera Interface	2-20

Figure 2-20	Frame Buffer Control	2-21
Figure 2-21	Camera Window Offset Scheme	2-29
Figure 2-22	Interrupt Generation Scheme	2-34
Figure 2-23	Image Mirror and Rotation	2-41
Figure 2-24	YCbCr Plane Memory Storing Style	2-45
Figure 2-25	Scaling Scheme	2-46
Figure 2-26	I/O Timing Diagram for Direct Path.....	2-53
Figure 2-27	Input & Output Modes in CAMIF.....	2-54
Figure 2-28	Capture Frame Control	2-59
Figure 2-29	Image Effect.....	2-62
Figure 2-30	ENVID_M SFR Setting When Input DMA Start to Read Memory Data.....	2-69
Figure 2-31	SFR and Operation (Related Each DMA When Selected Input DMA Path).....	2-70
Figure 2-32	Input DMA Address Change Timing (progressive to progressive)	2-70
Figure 2-33	Input DMA Address Change Timing (progressive to interlace)	2-71
Figure 2-34	Input DMA Address Change Timing (Software Update).....	2-71
Figure 2-35	Input/Ouput DMA pingpong Address Change Scheme	2-72
Figure 2-36	Input DMA Progressive-in to Interlace-out (only interlace_out setting).....	2-72
Figure 2-37	Input DMA Progressive-in to Interlace-out (Weave_in and Interlace_out setting)	2-72
Figure 2-38	Input DMA Offset and Image Size	2-78
Figure 2-39	Output DMA Offset and Image Size	2-78
Figure 3-1	MIPI DSI System Block Diagram	3-2
Figure 3-2	Rx Data Word Alignment	3-4
Figure 3-3	Signal Converting Diagram in Video Mode.....	3-5
Figure 3-4	Block Timing Diagram of HSA Mode (HSA mode reset: DSIM_CONFIG[20] = 0).....	3-6
Figure 3-5	Block Timing Diagram of HSA Mode (HSA mode set: DSIM_CONFIG[20] = 1).....	3-6
Figure 3-6	Block Timing Diagram of HBP Mode (HBP Mode Reset: DSIM_CONFIG[21] = 0)	3-7
Figure 3-7	Block Timing Diagram of HBP Mode (HBP Mode Set: DSIM_CONFIG[21] = 1)	3-7
Figure 3-8	Block Timing Diagram of HFP Mode (HFP Mode Reset: DSIM_CONFIG[22] = 0).....	3-7
Figure 3-9	Block Timing Diagram of HFP Mode (HFP Mode Set: DSIM_CONFIG[22] = 1).....	3-7
Figure 3-10	Block Timing Diagram of HSE Mode (HSE Mode Reset: DSIM_CONFIG[23] = 0)	3-8
Figure 3-11	Block Timing Diagram of HSE Mode (HSE Mode Set: DSIM_CONFIG[23] = 1)	3-8
Figure 3-12	Stable VFP Area Before Command Transfer Allowing Area.....	3-9
Figure 3-13	I80 Interface Timing Diagram	3-10
Figure 3-14	Packetizing for MIPI DSI Command Mode from I80 Interface	3-11
Figure 4-1	MIPI CSI System Block Diagram	4-2
Figure 4-2	Waveform of Output Data	4-3
Figure 4-3	MIPI CSIS Data Alignment	4-4
Figure 5-1	SGX540 Block Diagram	5-5
Figure 5-2	Block Diagram of Integration Information with Related Block	5-10
Figure 6-1	MFC Block Diagram.....	6-6
Figure 6-2	Luma and Chroma Pixel (8 bytes-aligned)	6-7
Figure 6-3	QCIF Image in 16pixel x 16lines (1x1) Tiled Mode.....	6-8
Figure 6-4	QCIF Image in 64pixel x 32lines (4x2) Tiled Mode.....	6-9
Figure 6-5	Shared Memory Input for Decoders	6-80
Figure 6-6	Shared Memory Output for Decoders.....	6-81
Figure 6-7	VC1 Parameters	6-82
Figure 6-8	Shared Memory Input for Encoders.....	6-83
Figure 6-9	Shared Memory Output for Encoders	6-83
Figure 6-10	Shared Memory Output for Encoders	6-85
Figure 6-11	Shared Memory Output for Encoders.....	6-86
Figure 7-1	Data Flow of TVOUT Module.....	7-2
Figure 7-2	Four Field NTSC (M) Sequence and Burst Blanking.....	7-4
Figure 7-3	Field PAL (BGHIDNc) Sequence and Burst Blanking	7-5

Figure 7-4	Eight Field PAL (BGHIDNc) Sequence and Burst Blanking	7-6
Figure 7-5	Horizontal Blanking and Active Video Timing @ 525/60 Hz.....	7-7
Figure 7-6	Horizontal Blanking and Active Video Timing @ 625/50 Hz.....	7-7
Figure 7-7	Magnitude Response of CB and CR Anti Aliasing Filter @ 13.5 MHz Sampling Rate.....	7-10
Figure 7-8	Phase Response of CB and CR Anti Aliasing Filter @ 13.5 MHz Sampling Rate	7-10
Figure 7-9	EIA-608 Closed Caption and Extended Data Service	7-11
Figure 7-10	IEC 61880 Wide Screen Signaling	7-12
Figure 7-11	ITU-R BT.1119 Wide Screen Signaling	7-13
Figure 7-12	Data Flow of CVBS Sub-Module	7-14
Figure 7-13	NTSC (M) Composite Video Signal with 75% Color Bars	7-15
Figure 7-14	PAL (BGHIDNc) Composite Video Signal with 75% Color Bars	7-16
Figure 7-15	Color Cube Comparison	7-17
Figure 7-16	Individual Gain & Offset Control for DAC Channel Balancing.....	7-31
Figure 8-1	Video Data Path.....	8-1
Figure 8-2	Block Diagram of Video Processor.....	8-2
Figure 8-3	Data Type for BOB	8-3
Figure 8-4	Difference Between IPC and X2 Scale-up.....	8-4
Figure 8-5	Examples of Usage Cases	8-15
Figure 8-6	Endian Mode	8-18
Figure 8-7	Video Scaling & Positioning on TV Display	8-21
Figure 8-8	Image Brightness & Contrast Control	8-43
Figure 8-9	4-Tap Vertical Poly-phase Filter	8-45
Figure 8-10	Pixel Repetition at Picture Boundary.....	8-46
Figure 9-1	Block Diagram of Mixer.....	9-3
Figure 9-2	TV Sub-System Block Diagram and Usage Frequency	9-4
Figure 9-3	Mixer Horizontal Scale and Blank-key.....	9-20
Figure 9-4	Mixer Blending	9-26
Figure 9-5	16bpp ARGB Example.....	9-27
Figure 9-6	Example of Expanding	9-27
Figure 9-7	Graphic Data Format in Memory	9-28
Figure 10-1	Block Diagram of HDMI	10-2
Figure 10-2	Block Diagram of HDMI SUB-System in S5PV210	10-3
Figure 10-3	Block Diagram of HDCP Key Management.....	10-4
Figure 10-4	Sequence of I2C Data	10-7
Figure 10-5	Block Diagram of HDMI TX Clock Scheme in S5PV210	10-12
Figure 10-6	Frequency Summary in Use	10-13
Figure 10-7	Frame Format	10-14
Figure 10-8	Sub-frame Format.....	10-15
Figure 10-9	Channel Status Block Extract from SPDIF Stream.....	10-16
Figure 10-10	Channel Status Block	10-17
Figure 10-11	Channel coding	10-18
Figure 10-12	Non-linear PCM Format.....	10-19
Figure 10-13	Structure of Power Down Circuit	10-75
Figure 11-1	Image Rotator Block Diagram.....	11-2
Figure 11-2	Ported Image Rotation Functions	11-3
Figure 11-3	Source Image Example (with window offset function)	11-4
Figure 11-4	Destination Image Example (90 degree rotated with window offset function).....	11-4
Figure 12-1	JPEG	12-2
Figure 12-2	JPEG In/Output Data Format.....	12-3
Figure 12-3	Raw Image Format in Memory (a) YCbCr4:2:2 (b) RGB5:6:5.....	12-4
Figure 12-4	Decimation and 1:2 Interpolation in Vertical Direction.....	12-4
Figure 12-5	YCbCr4:2:2 Color Format	12-5
Figure 12-6	Access Order in Quantizer Table.....	12-6

Figure 12-7	Bitstream of Compressed File	12-10
Figure 13-1	Color Format.....	13-2
Figure 13-2	FIMG-2D Rendering Pipeline.....	13-3
Figure 13-3	Rotation and Flip Example.....	13-6

Samsung Confidential

List of Tables

Table Number	Title	Page Number
Table 1-1	32BPP (8:8:8:8) Palette Data Format	1-24
Table 1-2	25BPP (A: 8:8:8) Palette Data Format.....	1-24
Table 1-3	19BPP (A: 6:6:6) Palette Data Format.....	1-25
Table 1-4	16BPP (A: 5:5:5) Palette Data Format.....	1-25
Table 1-5	Relation 16BPP Between VCLK and CLKVAL (TFT, Frequency of Video Clock Source=60MHz). 1-40	1-40
Table 1-6	i80 Output Mode.....	1-44
Table 1-7	RGB data pin map for RGB I/F	1-55
Table 1-8	RGB data pin map for I80 I/F	1-56
Table 2-1	Maximum Size.....	2-3
Table 2-2	ITU Camera Interface Signal Description	2-6
Table 2-3	Video Timing Reference Codes of ITU-656 8-Bit Format.....	2-8
Table 2-4	Sync Signal Timing Requirements.....	2-9
Table 2-5	DATA Order of YCbCr422 Align	2-10
Table 2-6	Color Space Conversion Equations	2-52
Table 2-7	FIFO Mode Image Format	2-52
Table 3-1	Internal Primary FIFO List.....	3-3
Table 3-2	I80 Interface Address Map	3-10
Table 3-3	Relation Between Input Transactions and DSI Transactions	3-12
Table 3-4	MIPID-PHY Interface Slave Signal	3-14
Table 3-5	PMS and Frequency Constraint.....	3-31
Table 3-6	AFC Code	3-31
Table 3-7	Band Control Setting	3-32
Table 4-1	Timing Diagram of Output Data	4-3
Table 4-2	Data Order of YUV422 Alignment.....	4-4
Table 5-1	Glossary of Terms.....	5-6
Table 5-2	Power Mode Summary About G3D.....	5-10
Table 5-3	G3D Register Summary.....	5-11
Table 6-1	Payload in the Shared Memory.....	6-79
Table 7-1	Filter Coefficients of Anti-aliasing Filters for Luminance Y	7-8
Table 7-2	Filter Coefficients of Anti-aliasing Filters for Chrominance Cb	7-9
Table 7-3	Filter Coefficients of Anti-aliasing Filters for Chrominance Cr	7-9
Table 7-4	Over-sampling Filter Coefficients Configuration	7-19
Table 7-5	Port Description of Video DAC.....	7-60
Table 7-6	Recommended RSET and RO According to Full Scale Voltage	7-61
Table 9-1	Graphic Blending-factor Alpha in Case of Normal Mode	9-18
Table 9-2	Graphic Blending Method	9-18
Table 10-1	HDMI LINK Timing Generator Configuration Guide	10-5
Table 10-2	HDMI PHY Configuration Table for 27MHz OSC_In	10-7
Table 10-3	HDMI PHY Configuration Table for 24MHz OSC_In	10-9
Table 10-4	Register Address Map	10-20
Table 12-1	Registers that Must be Configured Before Compression	12-7
Table 12-2	Relationship between Block Size and Color Format	12-8
Table 12-3	Markers in JPEG Codec	12-10
Table 12-4	JPEG Codec Control Registers	12-13
Table 12-5	Bitwise Expression of COEFxx	12-22
Table 12-6	JPEG Codec Table Assignment	12-26

1

DISPLAY CONTROLLER

1.1 OVERVIEW OF DISPLAY CONTROLLER

The Display controller consists of logic for transferring image data from a local bus of the camera interface controller or a video buffer located in system memory to an external LCD driver interface. The LCD driver interface supports three kinds of interface, namely, RGB-interface, indirect-i80 interface, and YUV interface for writeback. The display controller uses up to five overlay image windows that support various color formats, 256 level alpha blending, color key, x-y position control, soft scrolling, and variable window size, among others.

The display controller supports various color formats such as RGB (1 bpp to 24 bpp) and YCbCr 4:4:4 (only local bus). It is programmed to support the different requirements on screen related to the number of horizontal and vertical pixels, data line width for the data interface, interface timing, and refresh rate.

The display controller transfers the video data and generates the necessary control signals, such as, RGB_VSYNC, RGB_HSYNC, RGB_VCLK, RGB_VDEN and SYS_CS0, SYS_CS1, SYS_WE. In addition to generating control signals, the display controller contains data ports for video data (RGB_VD[23:0], and SYS_VD), as shown in [Figure 1-1](#).

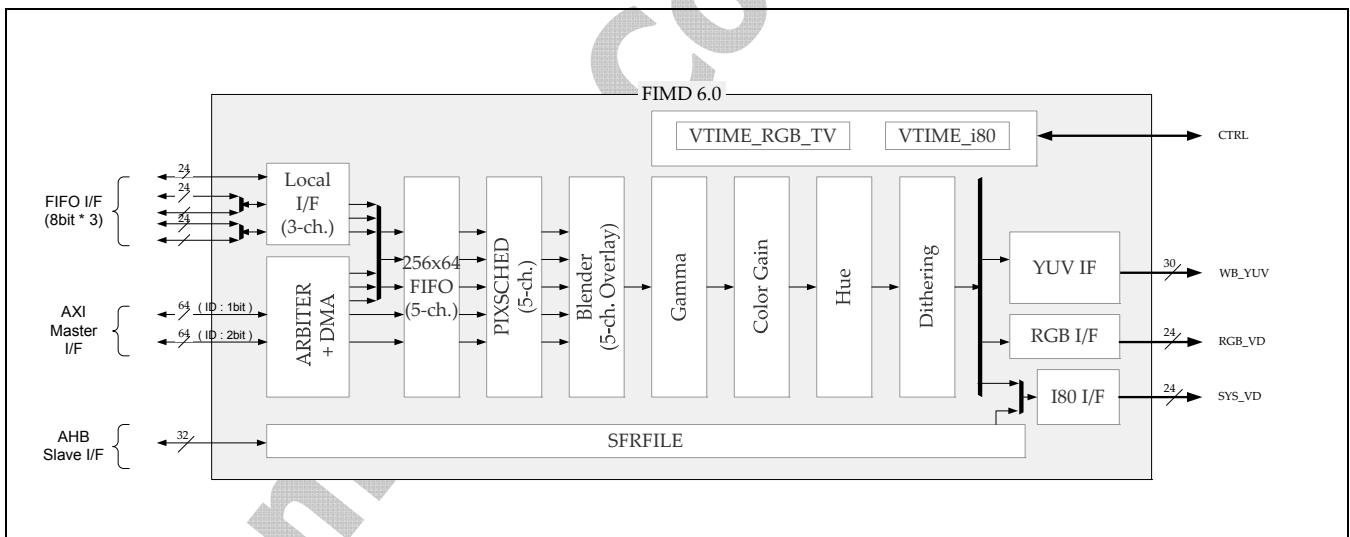


Figure 1-1 Block Diagram of Display Controller

1.2 KEY FEATURES OF THE DISPLAY CONTROLLER

The key features of the display controller include:

Bus Interface	AMBA AXI 64-bit Master/ AHB 32-bit Slave Local Video Bus (YCbCr/ RGB)
Video Output Interface	RGB Interface (24-bit Parallel/ 8-bit Serial) (dummy insertion, color sub-sampling (RGB skip) mode) (general, delta structure) Indirect i80 interface WriteBack interface (YUV444 30-bit)
Dual Output Mode	Supports i80 and WriteBack Supports RGB and WriteBack
PIP (OSD) function	Supports 8-bpp (bit per pixel) palletized color Supports 16-bpp non-palletized color Supports unpacked 18-bpp non-palletized color Supports unpacked 24-bpp non-palletized color Supports X,Y indexed position Supports 8-bit Alpha blending (Plane/ Pixel)
Source format	Window 0 Supports 1, 2, 4, or 8-bpp (bit per pixel) palletized color Supports 16, 18, or 24-bpp non-palletized color Supports RGB (8:8:8) local input from Local Bus (FIMC0) Window 1 Supports 1, 2, 4, or 8-bpp (bit per pixel) palletized color Supports 16, 18, or 24-bpp non-palletized color Supports RGB (8:8:8) local input from Local Bus (FIMC1) Window 2 Supports 1, 2, 4, or 8-bpp (bit per pixel) palletized color Supports 16, 18, or 24-bpp non-palletized color Supports RGB (8:8:8) local input from Local Bus (FIMC2) Window 3/ 4 Supports 1, 2, 4, or 8-bpp (bit per pixel) palletized color Supports 16, 18, or 24-bpp non-palletized color
Configurable Burst Length	Programmable 4/ 8/ 16 Burst DMA
Palette	Window 0/ 1/ 2/ 3/ 4 Supports 256 x 32 bits palette memory (5ea: One palette memory for each window)
Soft Scrolling	Horizontal = 1 Byte resolution Vertical = 1 pixel resolution
Virtual Screen	Virtual image can have up to 16MB image size. Each window can have its own virtual area.
Transparent Overlay	Supports Transparent Overlay

Color Key (Chroma Key)	Supports color key function Supports simultaneously color key and blending function
Partial Display	Supports LCD partial display function through i80 interface
Image Enhancement	Supports Gamma control
	Supports Hue control
	Supports color gain control
	Supports pixel compensation (only for delta structure)

1.3 FUNCTIONAL DESCRIPTION OF DISPLAY CONTROLLER

1.3.1 BRIEF DESCRIPTION OF THE SUB-BLOCK

The display controller consists of a VSFR, VDMA, VPRCS, VTIME, and video clock generator.

To configure the display controller, the VSFR has 121 programmable register sets, one gamma LUT register set (64 registers), one i80 command register set (12 registers), and five 256x32 palette memories.

VDMA is a dedicated display DMA that transfers video data in frame memory to VPRCS. By using this special DMA, you can display video data on screen without CPU intervention.

VPRCS receives video data from VDMA and sends it to display device (LCD) through data ports (RGB_VD, or SYS_VD), after changing the video data into a suitable data format. For example, 8-bit per pixel mode (8 bpp mode) or 16-bit per pixel mode (16 bpp mode).

VTIME consists of programmable logic to support the variable requirement of interface timing and rates commonly found in different LCD drivers. The VTIME block generates RGB_VSYNC, RGB_HSYNC, RGB_VCLK, RGB_VDEN, VEN_VSYNC, VEN_HSYNC, VEN_FIELD, VEN_HREF and SYS_CS0, SYS_CS1, SYS_WE, and so on.

Using the display controller data, you can select one of the above data paths by setting DISPLAY_PATH_SEL[1:0] (0xE010_7008). For more information, refer to Chapter, "Section 02.03. Clock controller".

1.3.2 DATA FLOW

FIFO is in the VDMA. If FIFO is empty or partially empty, the VDMA requests data fetching from frame memory based on burst memory transfer mode. The data transfer rate determines the size of FIFO.

The display controller contains five FIFOs (three local FIFOs and two DMA FIFOs), since it needs to support the overlay window display mode. Use one FIFO for one screen display mode.

VPRCS fetches data from FIFO. It contains the following functions for final image data: blending, image enhancing, and scheduling. It also supports the overlay function. This can overlay any image up to five window images, whose smaller or same size can be blended with the main window image having programmable alpha blending or color (chroma) key function.

[Figure 1-2](#) shows the data flow from system bus to output buffer.

VDMA has five DMA channels (Ch0 ~ Ch4) and three local input interfaces (CAMIF0, CAMIF1, and CAMIF2). The Color Space Conversion (CSC) block changes Hue (YCbCr, local input only) data to RGB data for blending operation. Also, the alpha values written in SFR determine the level of blending. Data from output buffer appears in the Video Data Port.

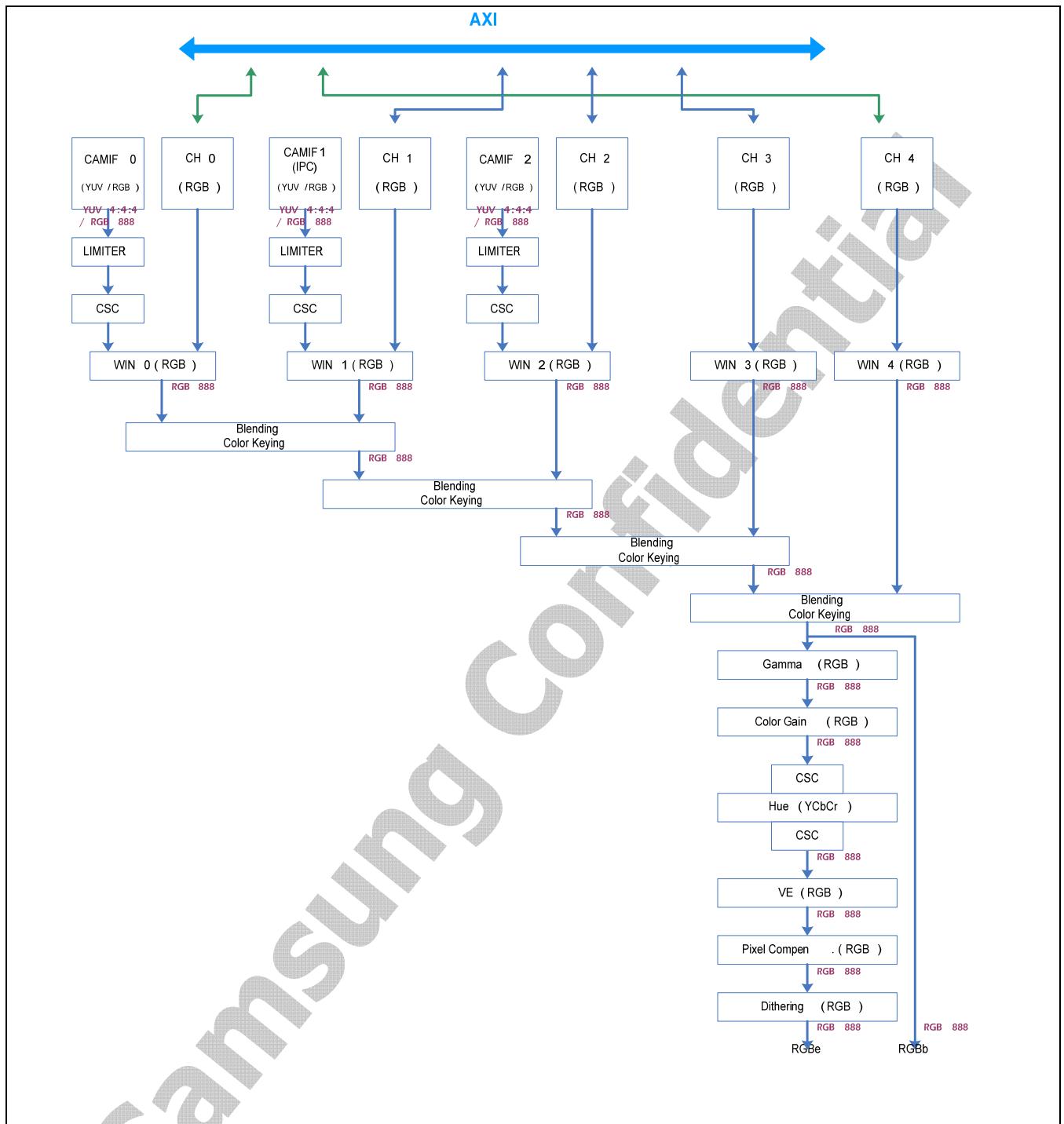


Figure 1-2 Block Diagram of the Data Flow

1.3.2.1 Interface

The display controller supports three types of interfaces:

- The first type is the conventional RGB interface, which uses RGB data, vertical/ horizontal sync, data valid signal, and data sync clock.
- The second type is the indirect i80 Interface, which uses address, data, chip select, read/ write control, and register/ status indicating signal. The LCD driver using i80 Interface contains a frame buffer and can self-refresh, so the display controller updates one still image by writing only one time to the LCD.
- The third type is FIFO interface with CAMIF2 for writeback.

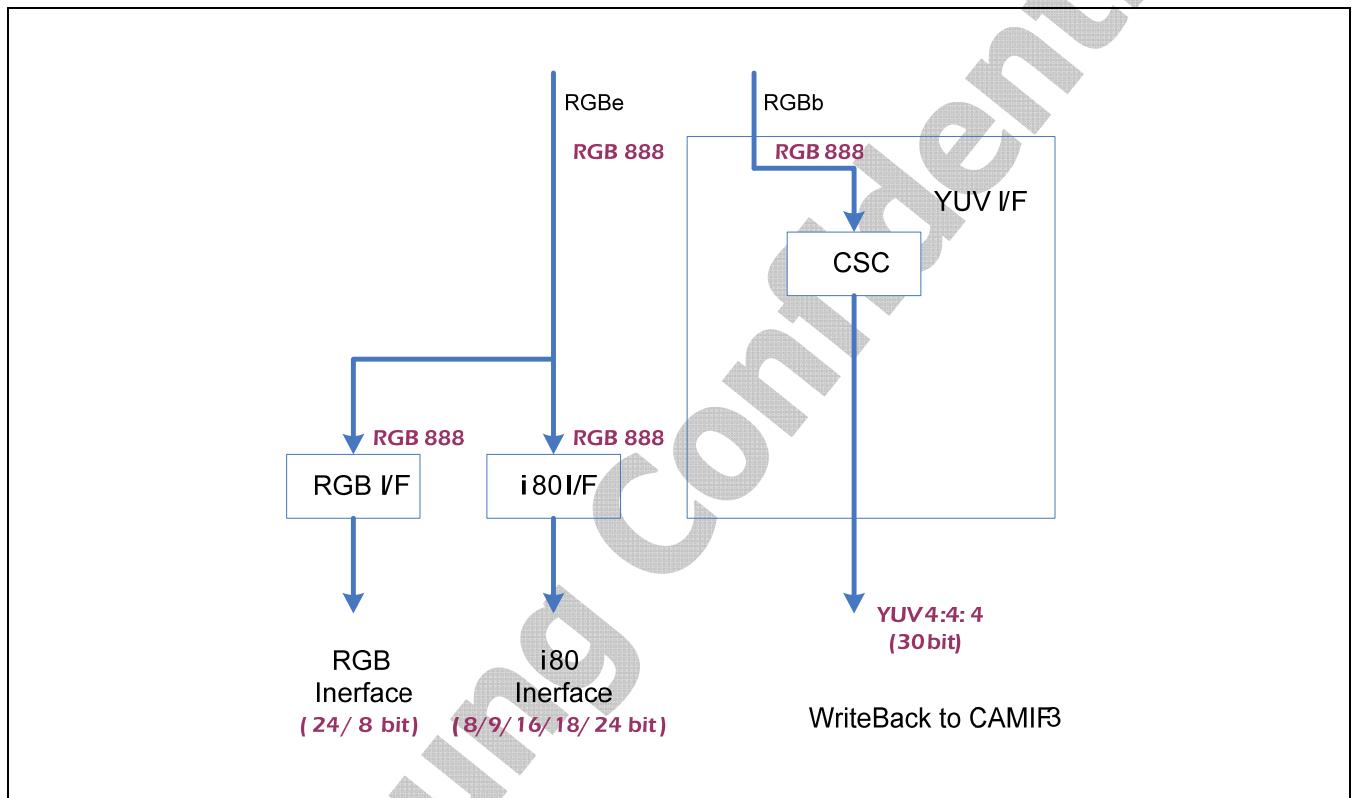


Figure 1-3 Block Diagram of the Interface

1.3.3 OVERVIEW OF THE COLOR DATA

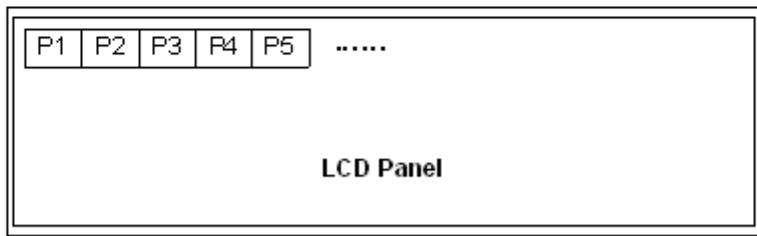
1.3.3.1 RGB Data Format

The display controller requests the specified memory format of frame buffer. The table below shows some examples of each display mode.

1.3.3.2 25BPP Display (A888)

(BSWP=0, HWSWP=0, WSWP=0)

	D[63:57]	D[56]	D[55:32]	D[31:25]	D[24]	D[23:0]
000H	Dummy Bit	AEN	P1	Dummy Bit	AEN	P2
008H	Dummy Bit	AEN	P3	Dummy Bit	AEN	P4
010H	Dummy Bit	AEN	P5	Dummy Bit	AEN	P6
...						



NOTE:

1. AEN: Specifies the transparency value selection bit.
AEN = 0: Selects ALPHA0.
AEN = 1: Selects ALPHA1.
If per-pixel blending is set, then this pixel blends with alpha value selected by AEN.
Alpha value is selected by SFR as ALPHA0_R, ALPHA0_G, ALPHA0_B, ALPHA1_R, ALPHA1_G, and ALPHA1_B.
For more information, refer to the section on "SFR".
2. D [23:16] = Red data, D [15:8] = Green data, and D [7:0] = Blue data.

1.3.3.2.1 32BPP (8888) Mode

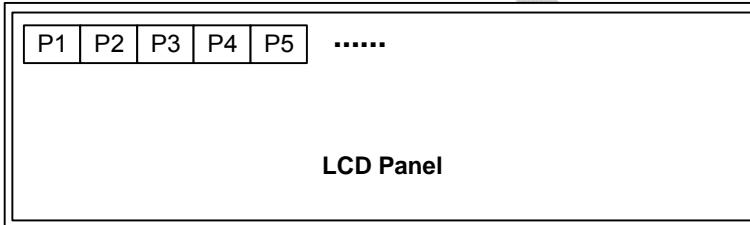
Pixel data contains Alpha value.

(BYSWP=0, HWSWP=0, WSWP=0)

	D[63:56]	D[55:32]	D[31:24]	D[23:0]
000H	ALPHA value	P1	ALPHA value	P2
008H	ALPHA value	P3	ALPHA value	P4
010H	ALPHA value	P5	ALPHA value	P6
...				

(BYSWP=0, HWSWP=0, WSWP=1)

	D[63:56]	D[55:32]	D[31:24]	D[23:0]
000H	ALPHA value	P2	ALPHA value	P1
008H	ALPHA value	P4	ALPHA value	P3
010H	ALPHA value	P6	ALPHA value	P5
...				



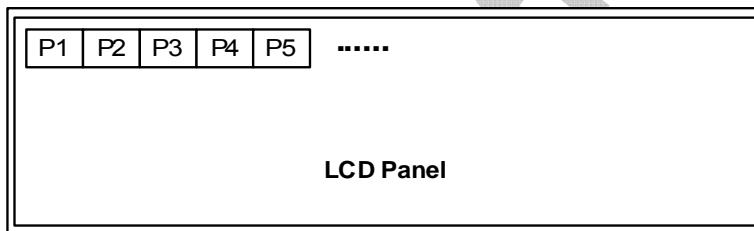
1.3.3.2.2 24BPP Display (A887)

(BSWP=0, HWSWP=0, WSWP=0)

	D[63:56]	D[55]	D[54:32]	D[31:24]	D[23]	D[22: 0]
000H	Dummy Bit	AEN	P1	Dummy Bit	AEN	P2
008H	Dummy Bit	AEN	P3	Dummy Bit	AEN	P4
010H	Dummy Bit	AEN	P5	Dummy Bit	AEN	P6
...						

(BSWP=0, HWSWP=0, WSWP=1)

	D[63:56]	D[55]	D[54:32]	D[31:24]	D[23]	D[22: 0]
000H	Dummy Bit	AEN	P2	Dummy Bit	AEN	P1
008H	Dummy Bit	AEN	P4	Dummy Bit	AEN	P3
010H	Dummy Bit	AEN	P6	Dummy Bit	AEN	P5
...						



NOTE:

1. AEN: Specifies the transparency value selection bit.
AEN = 0: Selects ALPHA0.
AEN = 1: Selects ALPHA1.
If per-pixel blending is set, then this pixel blends with alpha value selected by AEN.
Alpha value is selected by SFR as ALPHA0_R, ALPHA0_G, ALPHA0_B, ALPHA1_R, ALPHA1_G, and ALPHA1_B.
For more information, refer to the section on “SFR”.
2. D [22:15] = Red data, D [14:7] = Green data, and D [6:0] = Blue data.

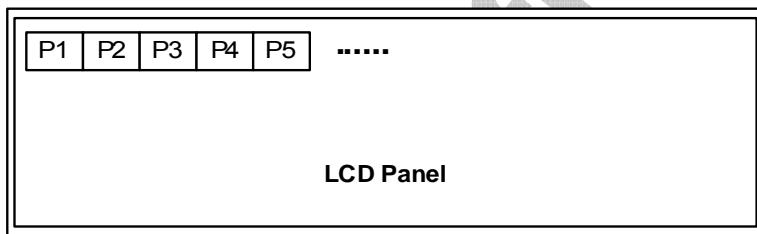
1.3.3.2.3 24BPP Display (888)

(BSWP =0, HWSWP=0, WSWP=0)

	D[63:56]	D[55:32]	D[31:24]	D[23:0]
000H	Dummy Bit	P1	Dummy Bit	P2
008H	Dummy Bit	P3	Dummy Bit	P4
010H	Dummy Bit	P5	Dummy Bit	P6
...				

(BSWP =0, HWSWP=0, WSWP=1)

	D[63:56]	D[55:32]	D[31:24]	D[23:0]
000H	Dummy Bit	P2	Dummy Bit	P1
008H	Dummy Bit	P4	Dummy Bit	P3
010H	Dummy Bit	P6	Dummy Bit	P5
...				



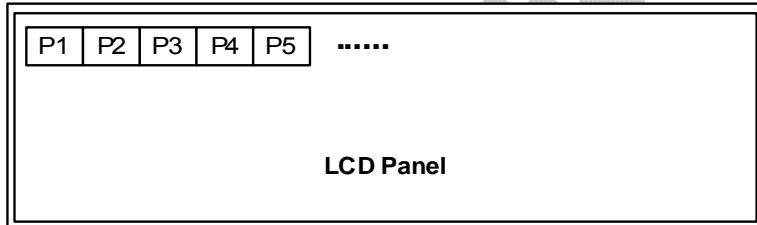
1.3.3.2.4 19BPP Display (A666)

(BSWP=0, HWSWP=0, WSWP=0)

	D[63:51]	D[50]	D[49:32]	D[31:19]	D[18]	D[17:0]
000H	Dummy Bit	AEN	P1	Dummy Bit	AEN	P2
008H	Dummy Bit	AEN	P3	Dummy Bit	AEN	P4
010H	Dummy Bit	AEN	P5	Dummy Bit	AEN	P6
...						

(BSWP=0, HWSWP=0, WSWP=1)

	D[63:51]	D[50]	D[49:32]	D[31:19]	D[18]	D[17:0]
000H	Dummy Bit	AEN	P2	Dummy Bit	AEN	P1
008H	Dummy Bit	AEN	P4	Dummy Bit	AEN	P3
010H	Dummy Bit	AEN	P6	Dummy Bit	AEN	P5
...						



NOTE:

1. AEN: Specifies the transparency value selection bit.

AEN = 0: Selects ALPHA0.

AEN = 1: Selects ALPHA1.

If per-pixel blending is set, then this pixel blends with alpha value selected by AEN.

Alpha value is selected by SFR as ALPHA0_R, ALPHA0_G, ALPHA0_B, ALPHA1_R, ALPHA1_G, and ALPHA1_B.

For more information, refer to the section on “SFR”.

2. D [17:12] = Red data, D [11:6] = Green data, and D [5:0] = Blue data.

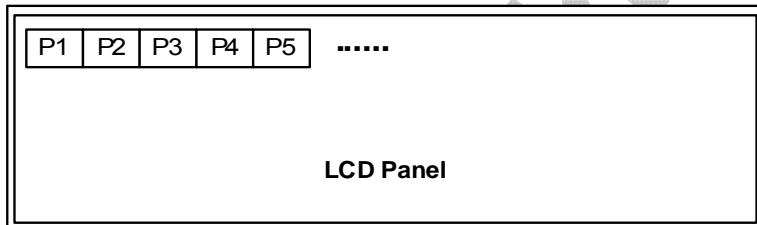
1.3.3.2.5 18BPP Display (666)

(BSWP =0, HWSWP=0, WSWP=0)

	D[63:50]	D[49:32]	D[31:18]	D[17:0]
000H	Dummy Bit	P1	Dummy Bit	P2
008H	Dummy Bit	P3	Dummy Bit	P4
010H	Dummy Bit	P5	Dummy Bit	P6
...				

(BSWP =0, HWSWP=0, WSWP=1)

	D[63:50]	D[49:32]	D[31:18]	D[17:0]
000H	Dummy Bit	P2	Dummy Bit	P1
008H	Dummy Bit	P4	Dummy Bit	P3
010H	Dummy Bit	P6	Dummy Bit	P5
...				



1.3.3.2.6 16BPP Display (A555)

(BSWP=0, HWSWP=0, WSWP=0)

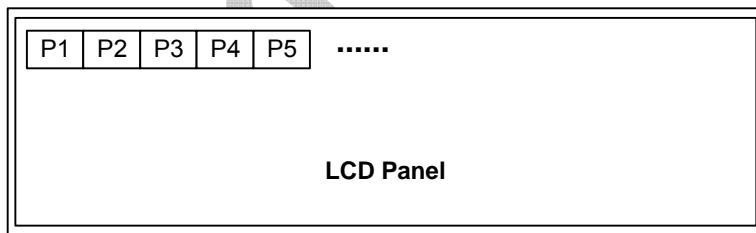
	D[63]	D[62:48]	D[47]	D[46:32]	D[31]	D[30:16]	D[15]	D[14:0]
000H	AEN1	P1	AEN2	P2	AEN3	P3	AEN4	P4
004H	AEN5	P5	AEN6	P6	AEN7	P7	AEN8	P8
008H	AEN9	P9	AEN10	P10	AEN11	P11	AEN12	P12
...								

(BSWP=0, HWSWP=0, WSWP=1)

	D[63]	D[62:48]	D[47]	D[46:32]	D[31]	D[30:16]	D[15]	D[14:0]
000H	AEN3	P3	AEN4	P4	AEN1	P1	AEN2	P2
004H	AEN7	P7	AEN8	P8	AEN5	P5	AEN6	P6
008H	AEN11	P11	AEN12	P12	AEN9	P9	AEN10	P10
...								

(BSWP=0, HWSWP=1, WSWP=0)

	D[63]	D[62:48]	D[47]	D[46:32]	D[31]	D[30:16]	D[15]	D[14:0]
000H	AEN4	P4	AEN3	P3	AEN2	P2	AEN1	P1
004H	AEN8	P8	AEN7	P7	AEN6	P6	AEN5	P5
008H	AEN12	P12	AEN11	P11	AEN10	P10	AEN9	P9
...								



NOTE:

1. AEN: Specifies the transparency value selection bit.
AEN = 0: Selects ALPHA0.
AEN = 1: Selects ALPHA1.
If per-pixel blending is set, then this pixel blends with alpha value selected by AEN.
Alpha value is selected by SFR as ALPHA0_R, ALPHA0_G, ALPHA0_B, ALPHA1_R, ALPHA1_G, and ALPHA1_B.
For more information, refer to the section on “SFR”.
2. D [14:10] = Red data, D [9:5] = Green data, and D [4:0] = Blue data.

1.3.3.2.7 16BPP Display (1555)

(BSWP=0, HWSWP=0, WSWP=0)

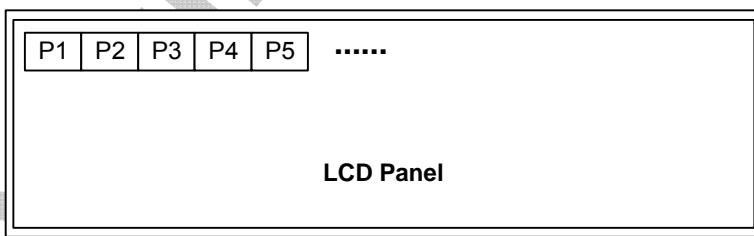
	D[63:48]	D[47:32]	D[31:16]	D[15:0]
000H	P1	P2	P3	P4
008H	P5	P6	P7	P8
010H	P9	P10	P11	P12
...				

(BSWP=0, HWSWP=0, WSWP=1)

	D[63:48]	D[47:32]	D[31:16]	D[15:0]
000H	P3	P4	P1	P2
008H	P7	P8	P5	P6
010H	P11	P12	P9	P10
...				

(BSWP=0, HWSWP=1, WSWP=0)

	D[63:48]	D[47:32]	D[31:16]	D[15:0]
000H	P4	P3	P2	P1
008H	P8	P7	P6	P5
010H	P12	P11	P10	P9
...				



1.3.3.2.8 6BPP Display (565)

(BSWP=0, HWSWP=0, WSWP=0)

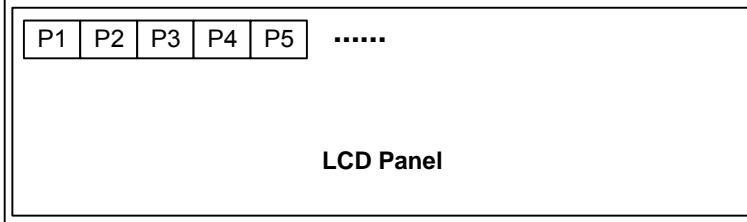
	D[63:48]	D[47:32]	D[31:16]	D[15:0]
000H	P1	P2	P3	P4
008H	P5	P6	P7	P8
010H	P9	P10	P11	P12
...				

(BSWP=0, HWSWP=0, WSWP=1)

	D[63:48]	D[47:32]	D[31:16]	D[15:0]
000H	P3	P4	P1	P2
008H	P7	P8	P5	P6
010H	P11	P12	P9	P10
...				

(BSWP=0, HWSWP=1, WSWP=0)

	D[63:48]	D[47:32]	D[31:16]	D[15:0]
000H	P4	P3	P2	P1
008H	P8	P7	P6	P5
010H	P12	P11	P10	P9
...				



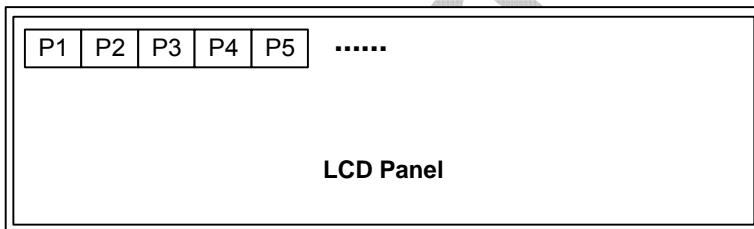
1.3.3.2.9 13BPP Display (A444)

(BYSWP=0, HWSWP=0, WSWP=0)

	D[63:61]	D[60]	D[59:48]	D[47:45]	D[44]	D[43:32]	D[31:29]	D[28]	D[27:16]	D[15:13]	D[12]	D[11:0]
000H	Dummy	AEN1	P1	Dummy	AEN2	P2	Dummy	AEN3	P3	Dummy	AEN4	P4
004H	Dummy	AEN5	P5	Dummy	AEN6	P6	Dummy	AEN7	P7	Dummy	AEN8	P8
008H	Dummy	AEN9	P9	Dummy	AEN10	P10	Dummy	AEN11	P11	Dummy	AEN12	P12
...												

(BYSWP=0, HWSWP=1, WSWP=0)

	D[63:61]	D[60]	D[59:48]	D[47:45]	D[44]	D[43:32]	D[31:29]	D[28]	D[27:16]	D[15:13]	D[12]	D[11:0]
000H	Dummy	AEN4	P4	Dummy	AEN3	P3	Dummy	AEN2	P2	Dummy	AEN1	P1
004H	Dummy	AEN8	P8	Dummy	AEN7	P7	Dummy	AEN6	P6	Dummy	AEN5	P5
008H	Dummy	AEN12	P12	Dummy	AEN11	P11	Dummy	AEN10	P10	Dummy	AEN9	P9
...												



NOTE:

1. AEN: Specifies the transparency value selection bit.
AEN = 0: Selects ALPHA0.
AEN = 1: Selects ALPHA1.
If per-pixel blending is set, then this pixel blends with alpha value selected by AEN.
2. D[11:8] = Red data, D[7:4] = Green data, and D[3:0] = Blue data.
3. 16BPP (4444) mode. (For more information, refer to the section on "SFR") Data has Alpha value.

(BYSWP=0, HWSWP=0, WSWP=0)

	D[63:60]	D[59:48]	D[47:44]	D[43:32]	D[31:28]	D[27:16]	D[15:12]	D[11:0]
000H	ALPHA1	P1	ALPHA2	P2	ALPHA3	P3	ALPHA4	P4
004H	ALPHA5	P5	ALPHA6	P6	ALPHA7	P7	ALPHA8	P8
008H	ALPHA9	P9	ALPHA10	P10	ALPHA11	P11	ALPHA12	P12
...								

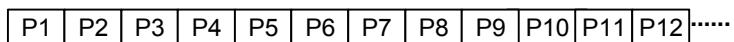
1.3.3.2.10 8BPP Display (A232)

(BYSWP=0, HWSWP=0, WSWP=0)

	D[63]	D[62:56]	D[55]	D[54:48]	D[47]	D[46:40]	D[39]	D[38:32]	D[31]	D[30:24]	D[23]	D[22:16]	D[15]	D[14:8]	D[7]	D[6:0]
000H	AEN	P1	AEN	P2	AEN	P3	AEN	P4	AEN	P5	AEN	P6	AEN	P7	AEN	P8
008H	AEN	P9	AEN	P10	AEN	P11	AEN	P12	AEN	P13	AEN	P14	AEN	P15	AEN	P16
010H	AEN	P17	AEN	P18	AEN	P19	AEN	P20	AEN	P21	AEN	P22	AEN	P23	AEN	P24
...																

(BYSWP=1, HWSWP=0, WSWP=0)

	D[63]	D[62:56]	D[55]	D[54:48]	D[47]	D[46:40]	D[39]	D[38:32]	D[31]	D[30:24]	D[23]	D[22:16]	D[15]	D[14:8]	D[7]	D[6:0]
000H	AEN	P8	AEN	P7	AEN	P6	AEN	P5	AEN	P4	AEN	P3	AEN	P2	AEN	P1
008H	AEN	P16	AEN	P15	AEN	P14	AEN	P13	AEN	P12	AEN	P11	AEN	P10	AEN	P9
010H	AEN	P24	AEN	P23	AEN	P22	AEN	P21	AEN	P20	AEN	P19	AEN	P18	AEN	P17
...																


NOTE:

1. AEN: Specifies the transparency value selection bit.
AEN = 0: Selects ALPHA0.
AEN = 1: Selects ALPHA1.
If per-pixel blending is set, then this pixel blends alpha value selected by AEN.
Alpha value is selected by SFR as ALPHA0_R, ALPHA0_G, ALPHA0_B, ALPHA1_R, ALPHA1_G, and ALPHA1_B.
For more information, refer to the section on "SFR".
2. D [6:5] = Red data, D [4:2] = Green data, and D [1:0] = Blue data.

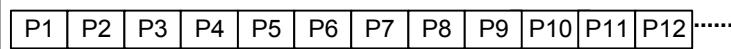
1.3.3.2.11 8BPP Display (Palette)

(BYSWP=0, HWSWP=0, WSWP=0)

	D[63:56]	D[55:48]	D[47:40]	D[39:32]	D[31:24]	D[23:16]	D[15:8]	D[7:0]
000H	P1	P2	P3	P4	P5	P6	P7	P8
008H	P9	P10	P11	P12	P13	P14	P15	P16
010H	P17	P18	P19	P20	P21	P22	P23	P24
...								

(BYSWP=1, HWSWP=0, WSWP=0)

	D[63:56]	D[55:48]	D[47:40]	D[39:32]	D[31:24]	D[23:16]	D[15:8]	D[7:0]
000H	P8	P7	P6	P5	P4	P3	P2	P1
008H	P16	P15	P14	P13	P12	P11	P10	P9
010H	P24	P23	P22	P21	P20	P19	P18	P17
...								



LCD Panel

NOTE: AEN: Specifies the transparency value selection bit (with WPALCON: Palette output format).

AEN = 0: Selects ALPHA0.

AEN = 1: Selects ALPHA1.

If per-pixel blending is set, then this pixel blends with alpha value selected by AEN.

Alpha value is selected by SFR as ALPHA0_R, ALPHA0_G, ALPHA0_B, ALPHA1_R, ALPHA1_G, and ALPHA1_B. For more information, refer to the section on "SFR".

1.3.3.2.12 4BPP Display (Palette)

(BYSWP=0, HWSWP=0, WSWP=0)

	D[63:60]	D[59:56]	D[55:52]	D[51:48]	D[47:44]	D[43:40]	D[39:36]	D[35:32]
000H	P1	P2	P3	P4	P5	P6	P7	P8
008H	P17	P18	P19	P20	P21	P22	P23	P24
...								

	D[31:28]	D[27:24]	D[23:20]	D[19:16]	D[15:12]	D[11:8]	D[7:4]	D[3:0]
000H	P9	P10	P11	P12	P13	P14	P15	P16
008H	P25	P26	P27	P28	P29	P30	P31	P32
...								

(BYSWP=1, HWSWP=0, WSWP=0)

	D[63:60]	D[59:56]	D[55:52]	D[51:48]	D[47:44]	D[43:40]	D[39:36]	D[35:32]
000H	P15	P16	P13	P14	P11	P12	P9	P10
008H	P31	P32	P29	P30	P27	P28	P25	P26
...								

	D[31:28]	D[27:24]	D[23:20]	D[19:16]	D[15:12]	D[11:8]	D[7:4]	D[3:0]
000H	P7	P8	P5	P6	P3	P4	P1	P2
008H	P23	P24	P21	P22	P19	P20	P17	P18
...								

NOTE: AEN: Specifies the transparency value selection bit (with WPALCON: Palette output format)

AEN = 0: Selects ALPHA0.

AEN = 1: Selects ALPHA1.

If per-pixel blending is set, then this pixel blends with alpha value selected by AEN.

Alpha value is selected by SFR as ALPHA0_R, ALPHA0_G, ALPHA0_B, ALPHA1_R, ALPHA1_G, and ALPHA1_B.
For more information, refer to the section on "SFR".

1.3.3.2.13 2BPP Display (Palette)

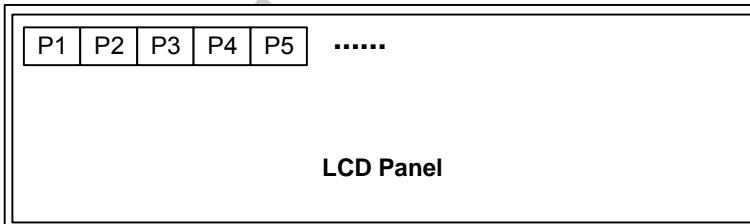
(BYSWP=0, HWSWP=0, WSWP=0)

	D[63:62]	D[61:60]	D[59:58]	D[57:56]	D[55:54]	D[53:52]	D[51:50]	D[49:48]
000H	P1	P2	P3	P4	P5	P6	P7	P8
008H	P33	P34	P35	P36	P37	P38	P39	P40
...								

	D[47:46]	D[45:44]	D[43:42]	D[41:40]	D[39:38]	D[37:36]	D[35:34]	D[33:32]
000H	P9	P10	P11	P12	P13	P14	P15	P16
008H	P41	P42	P43	P44	P45	P46	P47	P48
...								

	D[31:30]	D[29:28]	D[27:26]	D[25:24]	D[23:22]	D[21:20]	D[19:18]	D[17:16]
000H	P17	P18	P19	P20	P21	P22	P23	P24
008H	P49	P50	P51	P52	P53	P54	P55	P56
...								

	D[15:14]	D[13:12]	D[11:10]	D[9:8]	D[7:6]	D[5:4]	D[3:2]	D[1:0]
000H	P25	P26	P27	P28	P29	P30	P31	P32
008H	P57	P58	P59	P60	P61	P62	P63	P64
...								



NOTE: AEN: Specifies the transparency value selection bit (with WPALCON: Palette output format).

AEN = 0: Selects ALPHA0.

AEN = 1: Selects ALPHA1.

If per-pixel blending is set, then this pixel blends with alpha value selected by AEN.

Alpha value is selected by SFR as ALPHA0_R, ALPHA0_G, ALPHA0_B, ALPHA1_R, ALPHA1_G, and ALPHA1_B.
For more information, refer to the section on "SFR".

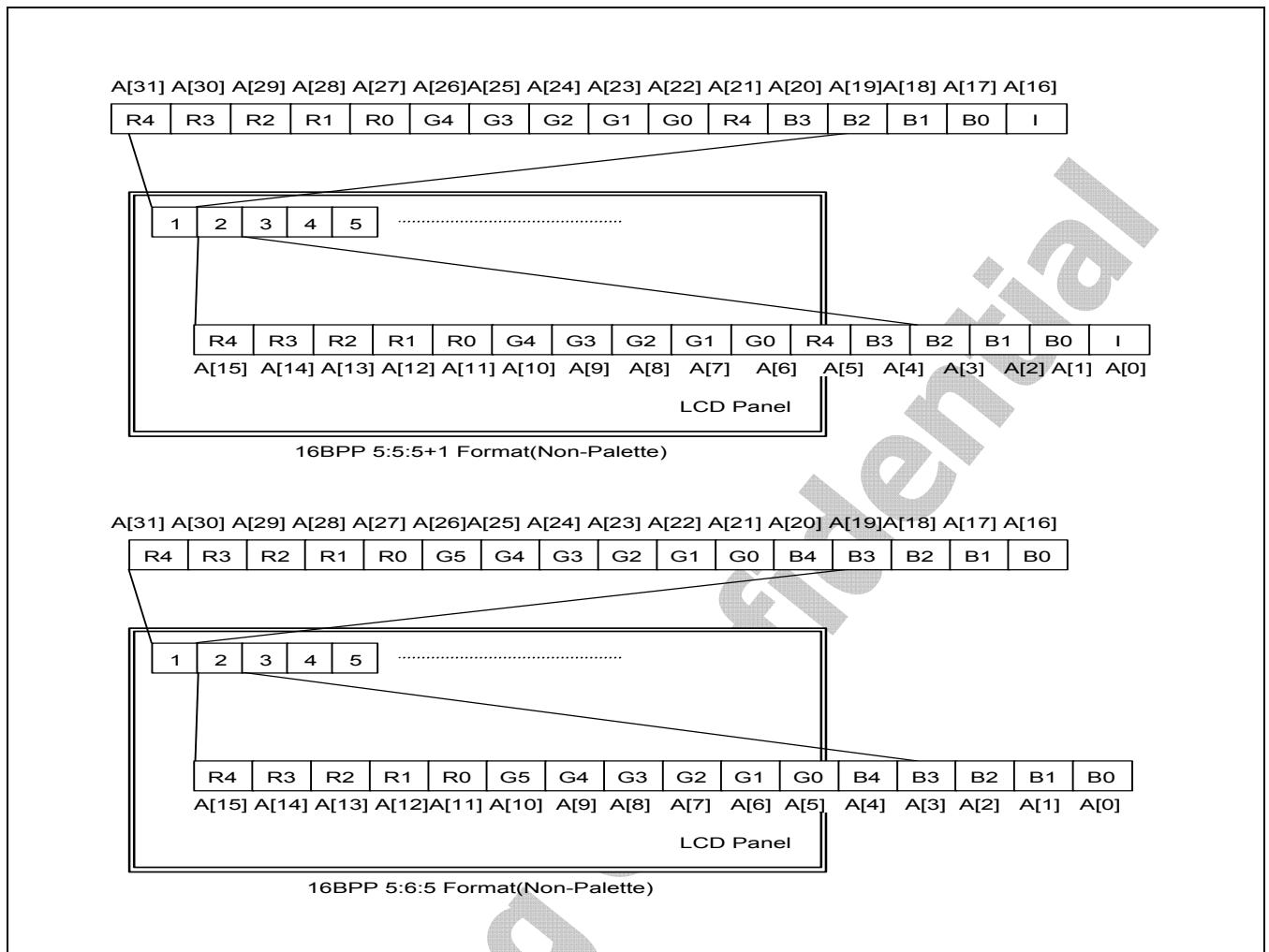


Figure 1-4 16BPP (5:6:5) Display Types

1.3.4 COLOR SPACE CONVERSION (CSC)

1.3.4.1 Color Space Conversion YCbCr to RGB (CSCY2R)

CSCY2R (Color Space Conversion Y to R)			
		601	709
Wide	R =	$Y + 1.371(Cr - 128)$	$Y + 1.540(Cr - 128)$
	G =	$Y - 0.698(Cr - 128) - 0.336(Cb - 128)$	$Y - 0.459(Cr - 128) - 0.183(Cb - 128)$
	B =	$Y + 1.732(Cb - 128)$	$Y + 1.816(Cb - 128)$
Narrow	R =	$1.164(Y - 16) + 1.596(Cr - 128)$	$1.164(Y - 16) + 1.793(Cr - 128)$
	G =	$1.164(Y - 16) - 0.813(Cr - 128) - 0.391(Cb - 128)$	$1.164(Y - 16) - 0.534(Cr - 128) - 0.213(Cb - 128)$
	B =	$1.164(Y - 16) + 2.018(Cb - 128)$	$1.164(Y - 16) + 2.115(Cb - 128)$

NOTE: “Wide” means RGB data has a nominal range from 16 to 235. On the other hand, “Narrow” means RGB data has a nominal range from 0 to 255.

Coefficient approximation.

- $1.164 = (2^7 + 2^4 + 2^2 + 2^0) \gg 7$
- $1.596 = (2^7 + 2^6 + 2^3 + 2^2) \gg 7$
- $0.813 = (2^6 + 2^5 + 2^3) \gg 7$
- $0.391 = (2^5 + 2^4 + 2^1) \gg 7$
- $2.018 = (2^8 + 2^1) \gg 7$
- $1.371 = (2^8 + 2^6 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0) \gg 8$
- $0.698 = (2^7 + 2^5 + 2^4 + 2^1 + 2^0) \gg 8$
- $0.336 = (2^6 + 2^4 + 2^2 + 2^1) \gg 8$
- $1.732 = (2^8 + 2^7 + 2^5 + 2^4 + 2^3 + 2^1 + 2^0) \gg 8$
- $1.540 = (2^8 + 2^7 + 2^3 + 2^1) \gg 8$
- $0.459 = (2^6 + 2^5 + 2^4 + 2^2 + 2^1) \gg 8$
- $0.183 = (2^5 + 2^3 + 2^2 + 2^1 + 2^0) \gg 8$
- $1.816 = (2^8 + 2^7 + 2^6 + 2^4 + 2^0) \gg 8$

1.3.4.2 Color Space Conversion RGB to YCbCr (CSCR2Y)

CSCR2Y (Color Space Conversion R to Y)			
		601	709
Wide	Y =	$0.299R + 0.587G + 0.114B$	$0.213R + 0.715G + 0.072B$
	Cb =	$-0.172R - 0.339G + 0.511B + 128$	$-0.117R - 0.394G + 0.511B + 128$
	Cr =	$0.511R - 0.428G - 0.083B + 128$	$0.511R - 0.464G - 0.047B + 128$
Narrow	Y =	$0.257R + 0.504G + 0.098B + 16$	$0.183R + 0.614G + 0.062B + 16$
	Cb =	$-0.148R - 0.291G + 0.439B + 128$	$-0.101R - 0.338G + 0.439B + 128$
	Cr =	$0.439R - 0.368G - 0.071B + 128$	$0.439R - 0.399G - 0.040B + 128$

NOTE: “Wide” means RGB data has a nominal range from 16 to 235. On the other hand, “Narrow” means RGB data has a nominal range from 0 to 255.

Coefficient approximation.

- $0.257 = (2^6 + 2^1) \gg 8$ $0.183 = (2^5 + 2^3 + 2^2 + 2^1 + 2^0) \gg 8$
- $0.504 = (2^7 + 2^0) \gg 8$ $0.614 = (2^7 + 2^4 + 2^3 + 2^2 + 2^0) \gg 8$
- $0.098 = (2^4 + 2^3 + 2^0) \gg 8$ $0.062 = (2^4) \gg 8$
- $0.148 = (2^5 + 2^2 + 2^1) \gg 8$ $0.101 = (2^4 + 2^3 + 2^1) \gg 8$
- $0.291 = (2^6 + 2^3 + 2^1) \gg 8$ $0.338 = (2^6 + 2^4 + 2^2 + 2^1 + 2^0) \gg 8$
- $0.439 = (2^6 + 2^5 + 2^4) \gg 8$ $0.399 = (2^6 + 2^5 + 2^2 + 2^1) \gg 8$
- $0.368 = (2^7 - 2^5 - 2^1) \gg 8$ $0.040 = (2^3 + 2^1) \gg 8$
- $0.071 = (2^4 + 2^1) \gg 8$

- $0.299 = (2^6 + 2^3 + 2^2 + 2^0) \gg 8$ $0.213 = (2^5 + 2^4 + 2^2 + 2^1 + 2^0) \gg 8$
- $0.587 = (2^7 + 2^4 + 2^2 + 2^1) \gg 8$ $0.715 = (2^7 + 2^5 + 2^4 + 2^2 + 2^1 + 2^0) \gg 8$
- $0.114 = (2^4 + 2^3 + 2^2 + 2^0) \gg 8$ $0.072 = (2^4 + 2^1) \gg 8$
- $0.172 = (2^5 + 2^3 + 2^2) \gg 8$ $0.117 = (2^4 + 2^3 + 2^2 + 2^1) \gg 8$
- $0.339 = (2^6 + 2^4 + 2^3 - 2^0) \gg 8$ $0.394 = (2^6 + 2^5 + 2^2 + 2^0) \gg 8$
- $0.511 = (2^7 + 2^1 + 2^0) \gg 8$
- $0.428 = (2^7 - 2^4 - 2^1) \gg 8$ $0.464 = (2^6 + 2^5 + 2^4 + 2^2 + 2^1 + 2^0) \gg 8$
- $0.083 = (2^4 + 2^2 + 2^0) \gg 8$ $0.047 = (2^3 + 2^2) \gg 8$

1.3.5 PALETTE USAGE

1.3.5.1 Palette Configuration and Format Control

The display controller supports 256-color palette to select color mapping. You can select up to 256 colors from 32-bit colors using these formats.

256 color palette consists of 256 (depth) × 32-bit SPSRAM. Palette supports 8:8:8, 6:6:6, 5:6:5 (R: G: B), and other formats.

For Example:

See A:5:5:5 format, write palette, as shown in [Table 1-2](#).

Connect VD pin to TFT LCD panel (R(5)=VD[23:19], G(5)=VD[15:11], and B(5)=VD[7:3]). AEN bit controls the blending function, enable or disable. Finally, set WPALCON (W1PAL, case window0) register to 0'b101. The 32-bit (8:8:8:8) format has an alpha value directly, without using alpha value register (ALPHA_0/1).

Table 1-1 32BPP (8:8:8:8) Palette Data Format

INDEX / Bit Pos.	3 1	3 0	2 9	2 8	2 7	2 6	2 5	2 4	2 3	2 2	2 1	2 0	1 9	1 8	1 7	1 6	1 5	1 4	1 3	1 2	1 1	1 0	9 8	8 7	7 6	6 5	5 4	4 3	3 2	2 1	0	
00h	ALPHA								R 7	R 6	R 5	R 4	R 3	R 2	R 1	R 0	G 7	G 6	G 5	G 4	G 3	G 2	G 1	G 0	B 7	B 6	B 5	B 4	B 3	B 2	B 1	B 0
01h	ALPHA								R 7	R 6	R 5	R 4	R 3	R 2	R 1	R 0	G 7	G 6	G 5	G 4	G 3	G 2	G 1	G 0	B 7	B 6	B 5	B 4	B 3	B 2	B 1	B 0
.....			
FFh	ALPHA								R 7	R 6	R 5	R 4	R 3	R 2	R 1	R 0	G 7	G 6	G 5	G 4	G 3	G 2	G 1	G 0	B 7	B 6	B 5	B 4	B 3	B 2	B 1	B 0
Number of VD	-	-	-	-	-	-	-	-	2 3	2 2	2 1	2 0	1 9	1 8	1 7	1 6	1 5	1 4	1 3	1 2	1 1	1 0	9 8	8 7	7 6	6 5	5 4	4 3	3 2	2 1	0	

Table 1-2 25BPP (A: 8:8:8) Palette Data Format

INDEX / Bit Pos.	3 1	3 0	2 9	2 8	2 7	2 6	2 5	2 4	2 3	2 2	2 1	2 0	1 9	1 8	1 7	1 6	1 5	1 4	1 3	1 2	1 1	1 0	9 8	8 7	7 6	6 5	5 4	4 3	3 2	2 1	0	
00h	-	-	-	-	-	-	-	A E N	R 7	R 6	R 5	R 4	R 3	R 2	R 1	R 0	G 7	G 6	G 5	G 4	G 3	G 2	G 1	G 0	B 7	B 6	B 5	B 4	B 3	B 2	B 1	B 0
01h	-	-	-	-	-	-	-	A E N	R 7	R 6	R 5	R 4	R 3	R 2	R 1	R 0	G 7	G 6	G 5	G 4	G 3	G 2	G 1	G 0	B 7	B 6	B 5	B 4	B 3	B 2	B 1	B 0
.....			
FFh	-	-	-	-	-	-	-	A E N	R 7	R 6	R 5	R 4	R 3	R 2	R 1	R 0	G 7	G 6	G 5	G 4	G 3	G 2	G 1	G 0	B 7	B 6	B 5	B 4	B 3	B 2	B 1	B 0
Number of VD	-	-	-	-	-	-	-	-	2 3	2 2	2 1	2 0	1 9	1 8	1 7	1 6	1 5	1 4	1 3	1 2	1 1	1 0	9 8	8 7	7 6	6 5	5 4	4 3	3 2	2 1	0	

Table 1-3 19BPP (A: 6:6:6) Palette Data Format

INDEX / Bit Pos.	3 1	3 0	2 9	2 8	2 7	2 6	2 5	2 4	2 3	2 2	2 1	2 0	1 9	1 8	1 7	1 6	1 5	1 4	1 3	1 2	1 1	1 0	9	8	7	6	5	4	3	2	1	0	
00h	-	-	-	-	-	-	-	-	-	-	-	-	-	A E N	R 5 4	R 3	R 2	R 1	R 0	G 5 4	G 3	G 2	G 1	G 0	B 5 4	B 4	B 3	B 2	B 1	B 0			
01h	-	-	-	-	-	-	-	-	-	-	-	-	-	A E N	R 5 4	R 3	R 2	R 1	R 0	G 5 4	G 3	G 2	G 1	G 0	B 5 4	B 4	B 3	B 2	B 1	B 0			
.....			
FFh	-	-	-	-	-	-	-	-	-	-	-	-	-	A E N	R 5 4	R 3	R 2	R 1	R 0	G 5 4	G 3	G 2	G 1	G 0	B 5 4	B 4	B 3	B 2	B 1	B 0			
Number of VD	-	-	-	-	-	-	-	-	-	-	-	-	-	2 3	2 2	2 1	2 0	1 9	1 8	1 5	1 4	1 3	1 2	1 1	1 0	7	6	5	4	3	2	1	0

Table 1-4 16BPP (A: 5:5:5) Palette Data Format

INDEX / Bit Pos.	3 1	3 0	2 9	2 8	2 7	2 6	2 5	2 4	2 3	2 2	2 1	2 0	1 9	1 8	1 7	1 6	1 5	1 4	1 3	1 2	1 1	1 0	9	8	7	6	5	4	3	2	1	0	
00h	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A E N	R 4 3	R 3	R 2	R 1	R 0	G 4 3	G 3	G 2	G 1	G 0	B 4 3	B 3	B 2	B 1	B 0			
01h	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A E N	R 4 3	R 3	R 2	R 1	R 0	G 4 3	G 3	G 2	G 1	G 0	B 4 3	B 3	B 2	B 1	B 0			
.....			
FFh	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A E N	R 4 3	R 3	R 2	R 1	R 0	G 4 3	G 3	G 2	G 1	G 0	B 4 3	B 3	B 2	B 1	B 0			
Number of VD	-	-	-	-	-	-	-	-	-	-	-	-	-	-	2 3	2 2	2 1	2 0	1 9	1 5	1 4	1 3	1 2	1 1	1 0	7	6	5	4	3	2	1	0

1.3.5.2 Palette Read/ Write

You should not access palette memory when the Vertical Status (VSTATUS) register has an ACTIVE status. VSTATUS must be checked to do Read/ Write operation on the palette.

1.3.6 WINDOW BLENDING

1.3.6.1 Overview of Window Blending

The main function of the VPRCS module is window blending. Display controller comprises five window layers (win0 ~ win4).

Example of Application:

The system uses

win0 as OS window, full TV screen window, and so on.

win1 as small (next channel) TV screen with win2 as menu.

win3 as caption.

win 4 as channel information.

win3 and win4 have color limitation while using color index with Color LUT. This feature enhances the system performance by reducing the data rate of total system.

Example of Total Five Windows:

win 0 (base): Local/ (YCbCr, RGB without palette)

win 1 (Overlay1): RGB with palette

win 2 (Overlay2): RGB with palette

win 3 (Caption): RGB (1/2/4) with 16-level Color LUT

win 4 (Cursor): RGB (1/2) with 4-level Color LUT

Overlay Priority

Win 4 > Win 3 > Win 2 > Win 1 > Win 0

Color Key

The register value to ColorKey register must be set by 24-bit RGB format.

Blending Equation

<Data blending>

$\text{Win01(R,G,B)} = \text{Win0(R,G,B)} \times b1 + \text{Win1(R,G,B)} \times a1$
 $\text{Win012(R/G/B)} = \text{Win01(R/G/B)} \times b2 + \text{Win2(R/G/B)} \times a2$
 $\text{Win0123(R/G/B)} = \text{Win012(R/G/B)} \times b3 + \text{Win3(R/G/B)} \times a3$
 $\text{WinOut(R/G/B)} = \text{Win0123(R/G/B)} \times b4 + \text{Win4(R/G/B)} \times a4$

, where,

Win0(R) = Window 0's Red data,
Win0(G) = Window 0's Green data,
Win0(B) = Window 0's Blue data,
Win1(R) = Window 1's Red data,

...

b1 = Background's Data blending equation1 factor,
a1 = Foreground's Data blending equation1 factor,
b2 = Background's Data blending equation2 factor,
a2 = Foreground's Data blending equation2 factor,

<Alpha value blending>

$\text{AR(G,B)01} = \text{AR(G,B)0} \times q1 + \text{AR(G,B)1} \times p1$
 $\text{AR(G,B)012} = \text{AR(G,B)01} \times q2 + \text{AR(G,B)2} \times p2$
 $\text{AR(G,B)0123} = \text{AR(G,B)012} \times q3 + \text{AR(G,B)3} \times p3$

, where,

AR0 = Window 0's Red blending factor,
AG0 = Window 0's Green blending factor,
AB0 = Window 0's Blue blending factor,
AR1 = Window 1's Red blending factor,...
AR01 = Window01's Red blending factor (alpha value blending between AR0 and AR1),
AG01 = Window01's Green blending factor (alpha value blending between AG0 and AG1),
AB01 = Window01's Blue blending factor (alpha value blending between AB0 and AB1),
AR012 = Window012's Red blending factor (alpha value blending between AR01 and AR2),
...
q1 = Background's Alpha value blending equation1 factor,
p1 = Foreground's Alpha value blending equation1 factor,
q2 = Background's Alpha value blending equation2 factor,
p2 = Foreground's Alpha value blending equation2 factor, ...

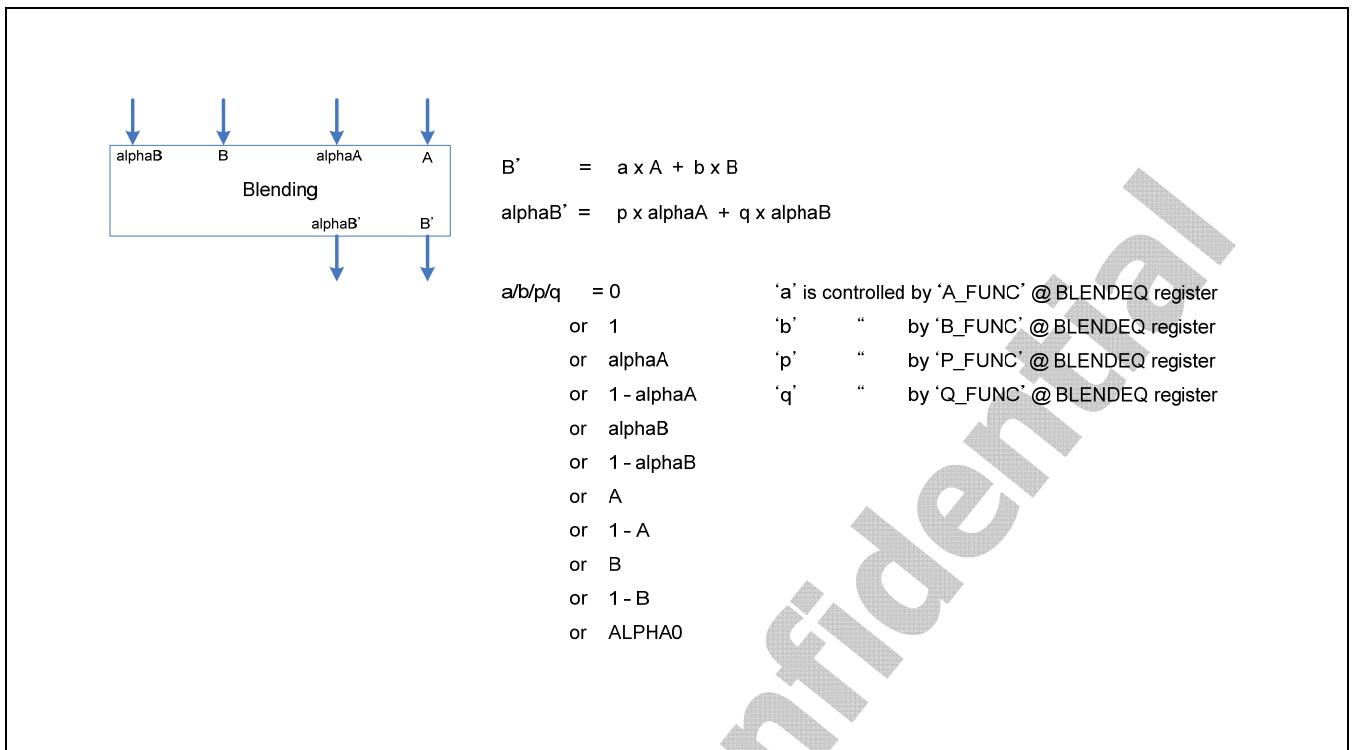


Figure 1-5 Blending Equation

< Default blending equation >

Data blending>

$$B' = B \times (1-\alpha_A) + A \times \alpha_A$$

Alpha value blending>

$$\alpha_B' = 0 (= \alpha_B \times 0 + \alpha_A \times 0)$$

1.3.6.2 Blending Diagram

The display controller can blend five layers for one pixel at the same time. ALPHA0_R, ALPHA0_G, ALPHA0_B, ALPHA1_R, ALPHA1_G, and ALPHA1_B registers control the alpha value (blending factor), which are implemented for each window layer and color (R,G,B).

The example below shows the R (Red) output using ALPHA_R value of each window.

All windows have two kinds of alpha blending value:

- Alpha value for transparency enable (AEN value ==1)
- Alpha value for transparency disable (AEN value == 0).

If WINEN_F and BLD_PIX are enabled and ALPHA_SEL is disabled, then AR is chosen using the following equation:

- $AR = (\text{Pixel}(R)\text{'s AEN value} == 1'b1) ? \text{Reg(ALPHA1_R)} : \text{Reg(ALPHA0_R)};$
- $AG = (\text{Pixel}(G)\text{'s AEN value} == 1'b1) ? \text{Reg(ALPHA1_G)} : \text{Reg(ALPHA0_G)};$
- $AB = (\text{Pixel}(B)\text{'s AEN value} == 1'b1) ? \text{Reg(ALPHA1_B)} : \text{Reg(ALPHA0_B)};$
(where, BLD_PIX == 1, ALPHA_SEL == 0)

If WINEN_F is enabled and BLD_PIX is disabled, then AR is controlled by ALPHA_SEL ALPHA0. AEN bit information is not used anymore.

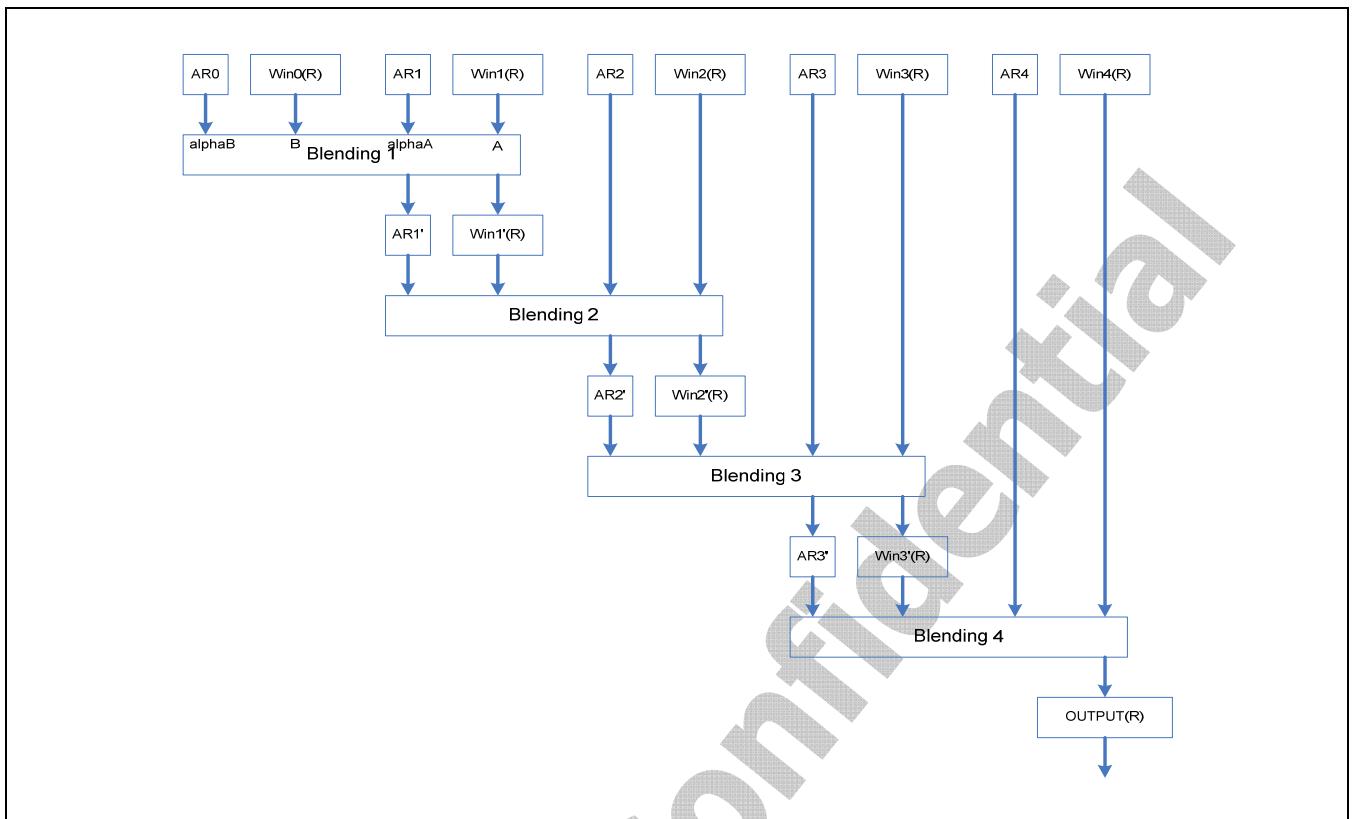


Figure 1-6 Blending Diagram

Example:

Window n's blending factor decision ($n=0, 1, 2, 3, 4$). For more information, refer to the section on "SFR".

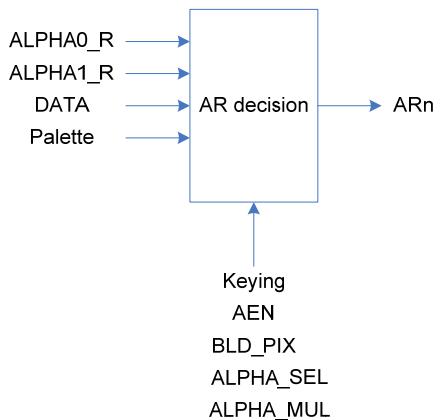


Figure 1-7 Blending Factor Decision

NOTE: If DATA [15:12] (BPPMODE_F = b'1110, ARGB4444 format) is used to blend, alpha value is {DATA [15:12], DATA [15:12]} (4-bit \rightarrow 8-bit expanding).

1.3.6.3 Color-Key Function

The Color-Key function in display controller supports various effects for image mapping. For special functionality, the Color-Key register that specifies the color image of OSD layer is substituted by the background image--either as cursor image or preview image of the camera.

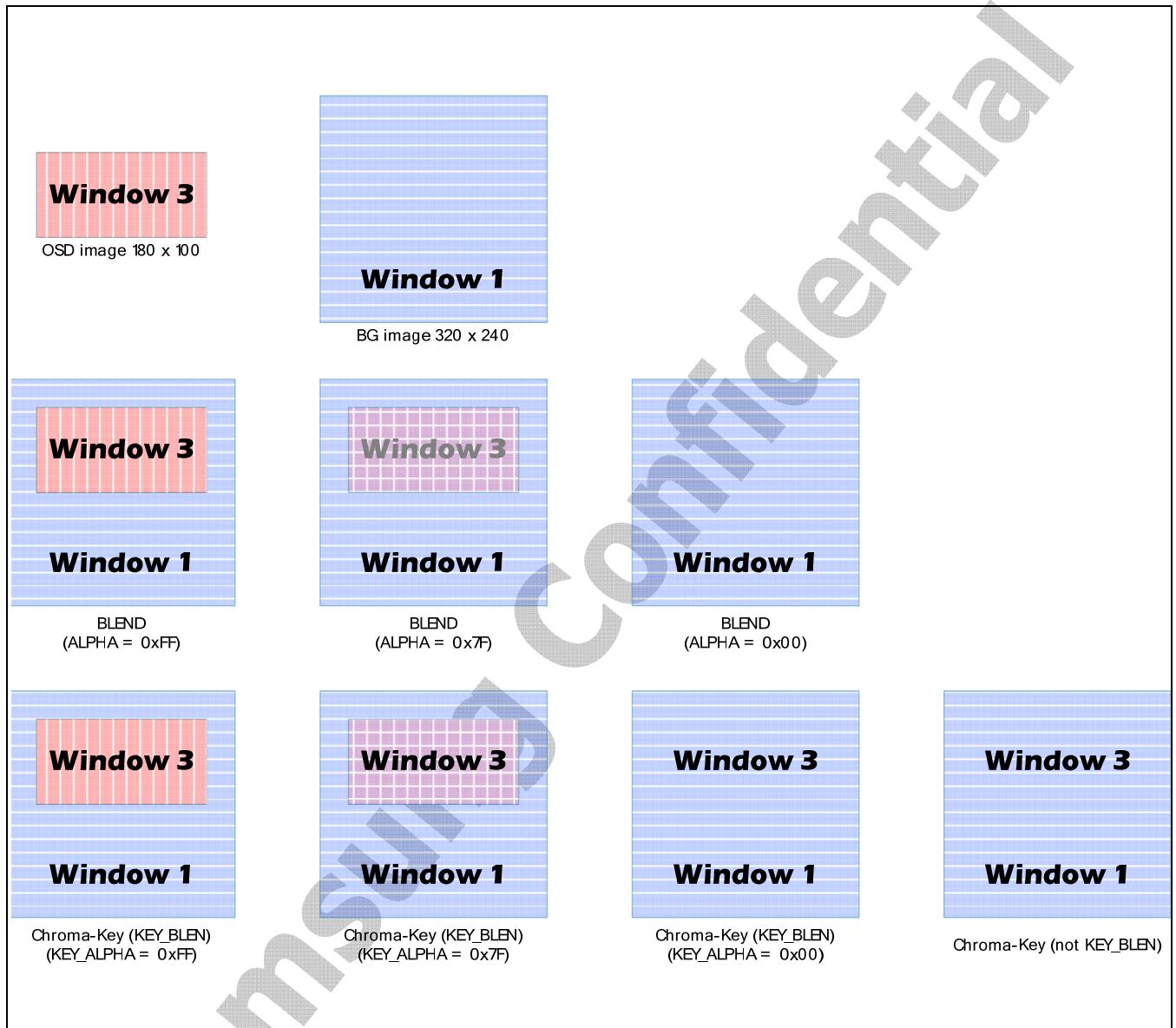


Figure 1-8 Color-Key Function Configurations

1.3.6.4 Blending and COLOR-KEY Function

The display controller supports simultaneous blending function--with two transparency factors and Color-Key function in the same window.



Figure 1-9 Blending and Color-Key Function

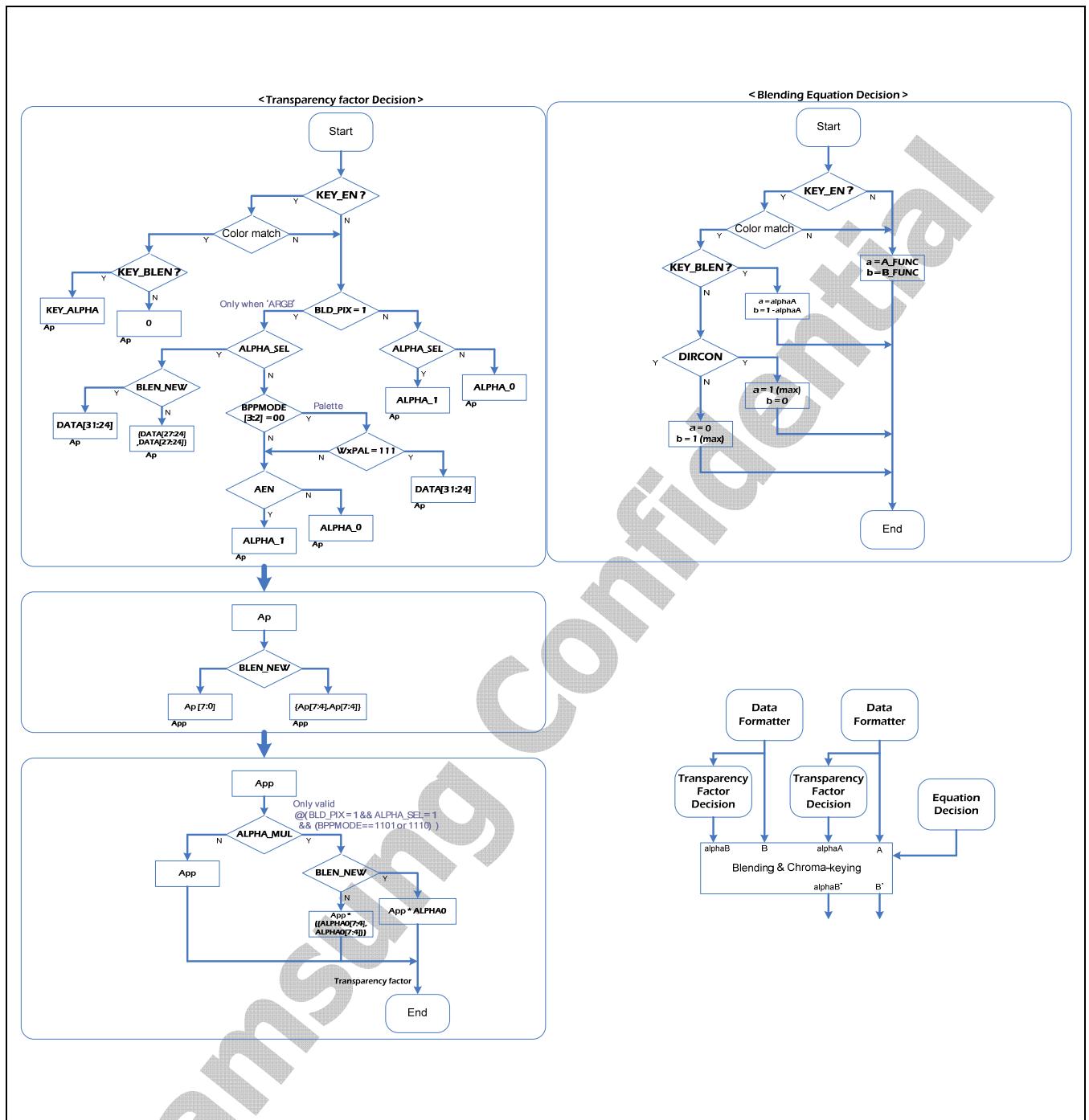


Figure 1-10 Blending Decision Diagram

1.3.7 IMAGE ENHANCEMENT

1.3.7.1 Overview of Image Enhancement

One of the main functions of the VPRCS module is Image Enhancement. The display controller supports Gamma, Hue, Color Gain, and Pixel Compensation Control functions.

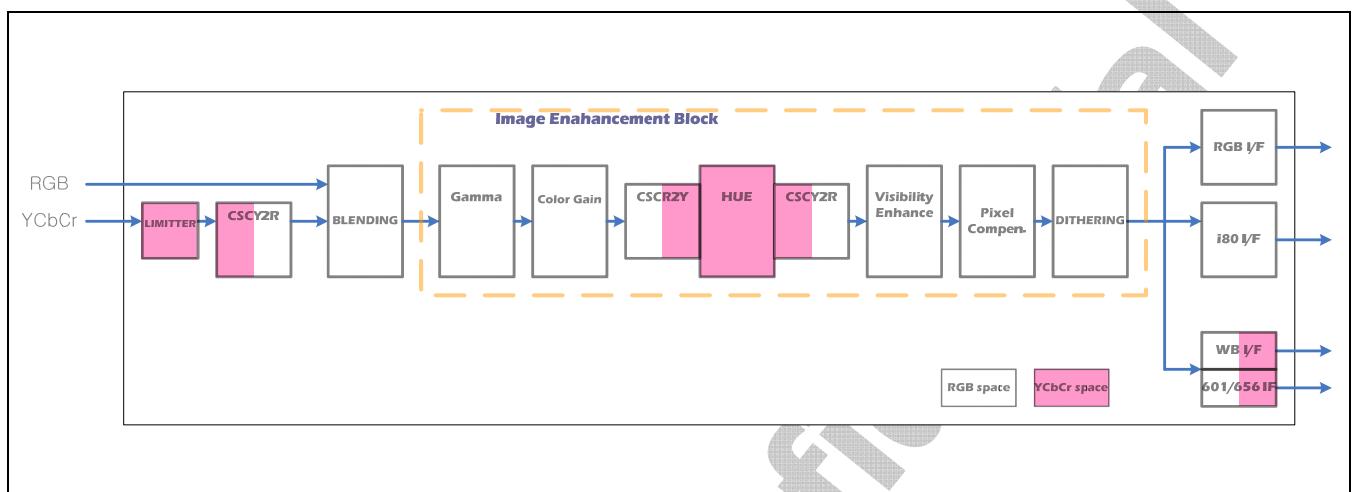


Figure 1-11 Image Enhancement Flow

1.3.7.2 Gamma Control

Gamma control comprises of 65 LUT registers. The output value is determined by piecewise-linear operation between two LUT registers. The output value saturates at 255.

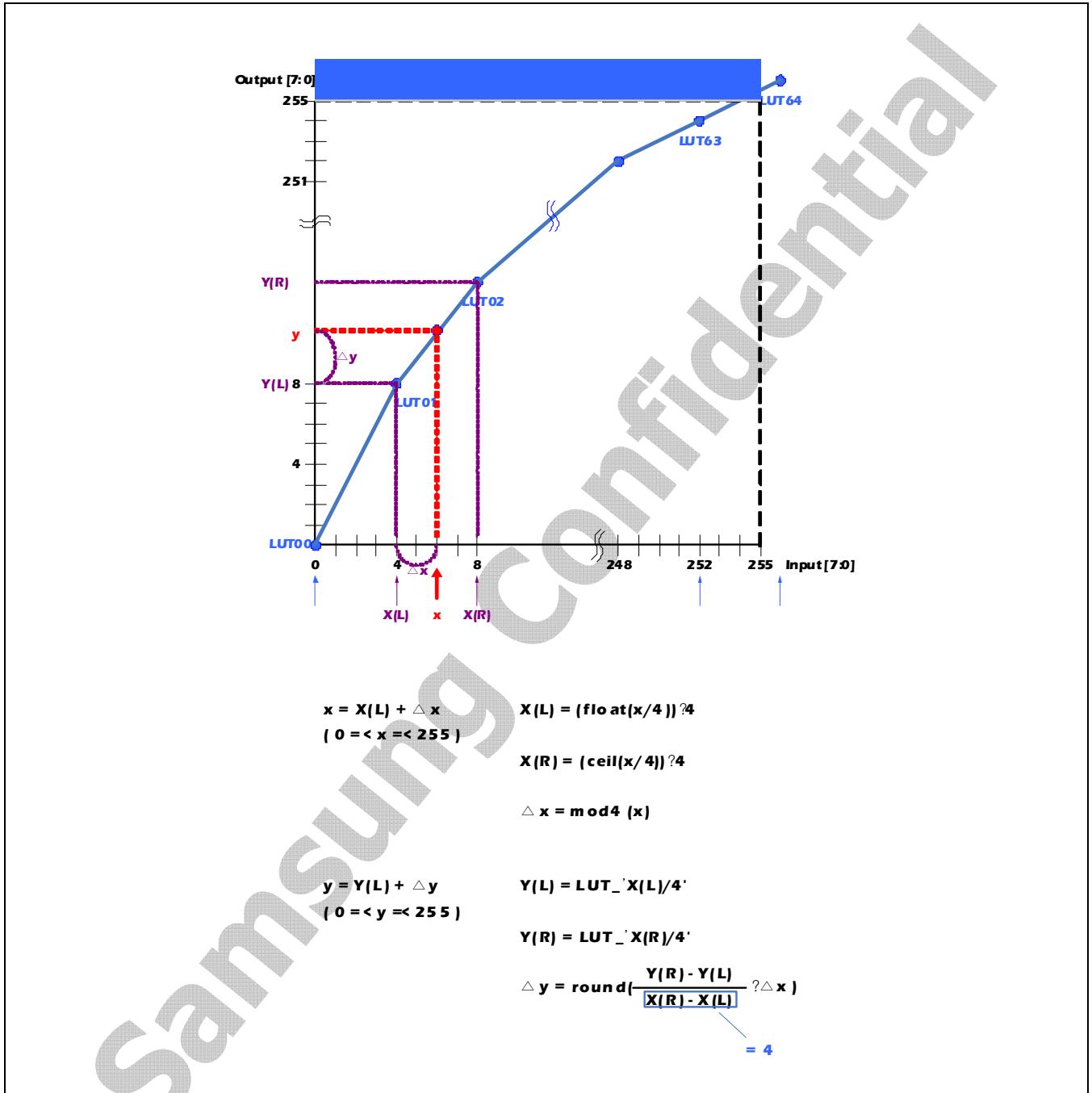


Figure 1-12 Image Enhancement Flow

1.3.7.3 Color Gain Control

The Color Gain Control comprises of three registers for each color: Red, Green, and Blue. The maximum value of Color Gain is 3.99609375 (approximately 4) and it has an 8-bit resolution.

- R (color gain) = $R \cdot CG_RGAIN$
- G (color gain) = $G \cdot CG_GGAIN$
- B (color gain) = $B \cdot CG_BGAIN$

$CG_R(G,B)GAIN$ comprises of 2-bit integer and 8-bit fraction.

(Maximum value is approximately 4 with 8-bit resolution.) The output value saturates at 255.

1.3.7.4 Hue Control

Hue Control comprises of eight registers for coefficients of Hue matrix.

- $Cb<hue> = CBG0 \cdot (Cb + OFFSET_IN) + CBG1 \cdot (Cr + OFFSET_IN) + OFFSET_OUT$
- $Cr<hue> = CRG0 \cdot (Cb + OFFSET_IN) + CRG1 \cdot (Cr + OFFSET_IN) + OFFSET_OUT$
(In general, $OFFSET_IN$ is '-128' and $OFFSET_OUT$ is '+128'.)
- $CBG0 = (Cb + OFFSET_IN) \geq 0 ? CBG0_P : CBG0_N$
- $CBG1 = (Cr + OFFSET_IN) \geq 0 ? CBG1_P : CBG1_N$
- $CRG0 = (Cb + OFFSET_IN) \geq 0 ? CRG0_P : CRG0_N$
- $CRG1 = (Cr + OFFSET_IN) \geq 0 ? CRG1_P : CRG1_N$

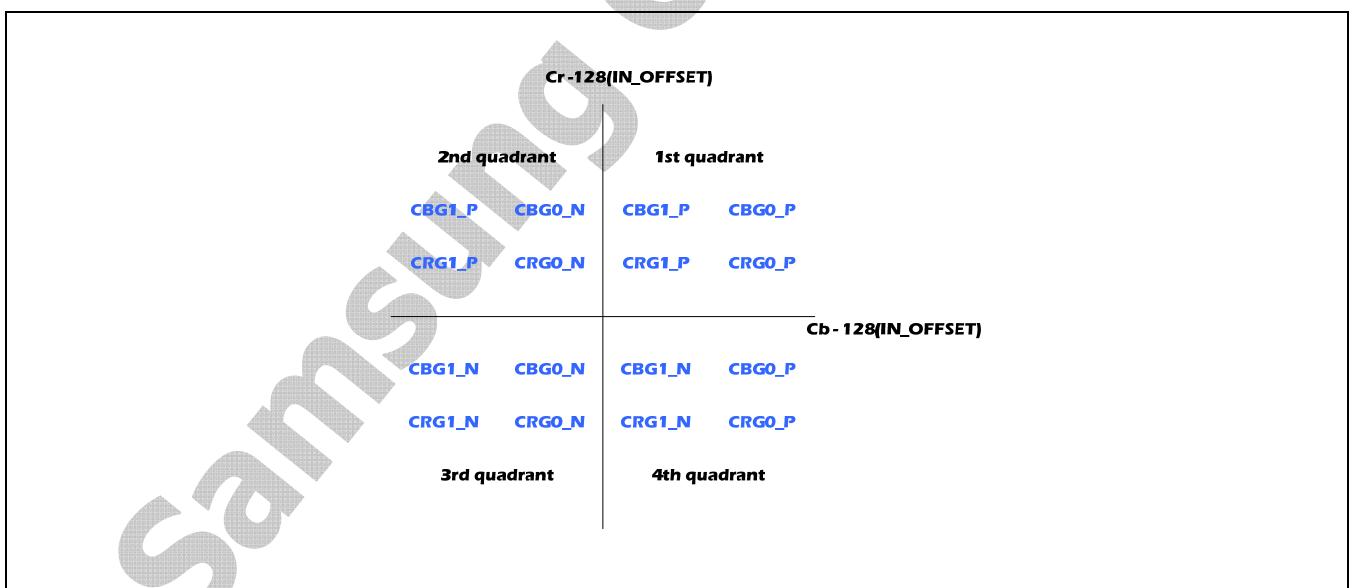


Figure 1-13 Hue Coefficient Decision

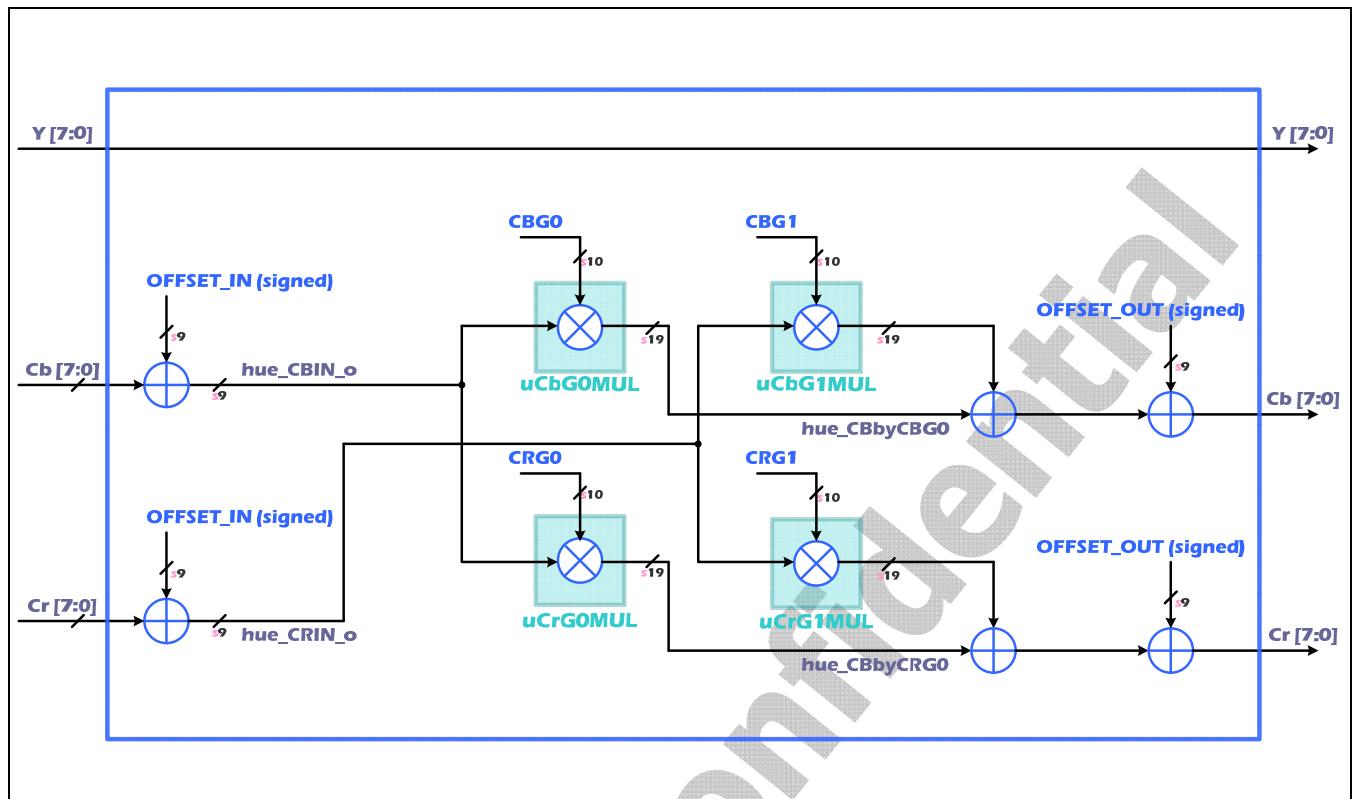


Figure 1-14 Hue Control Block Diagram

1.3.7.5 Pixel Compensation Control

The purpose of Pixel Compensation Control is to compensate data for delta-structure.

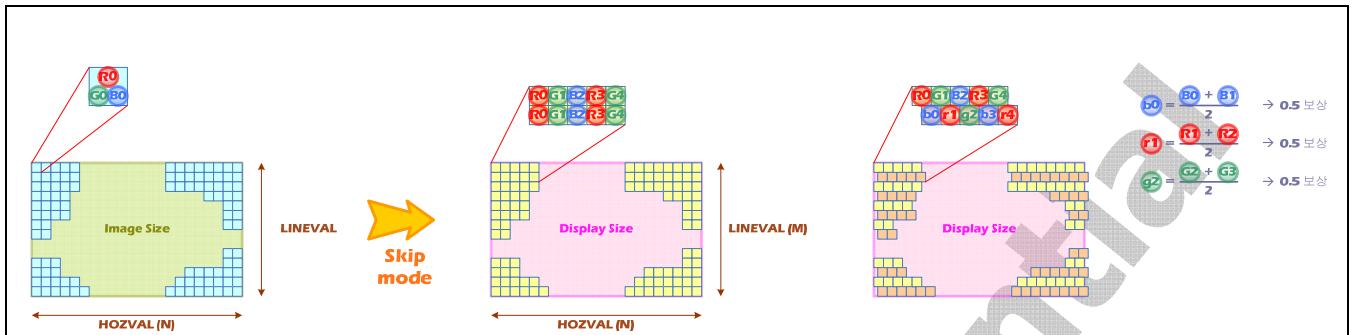


Figure 1-15 Example1. RGBSPSEL==1'b0, RGB_SKIP==1'b1, PIXCOMPEN_DIR==1'b0

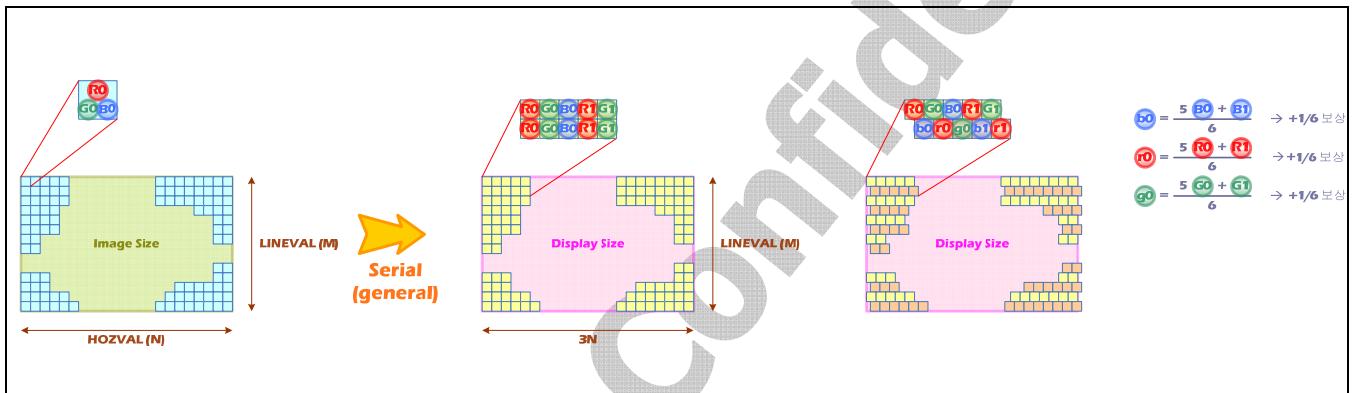


Figure 1-16 Example2. RGBSPSEL==1'b1, PIXCOMPEN_DIR==1'b0

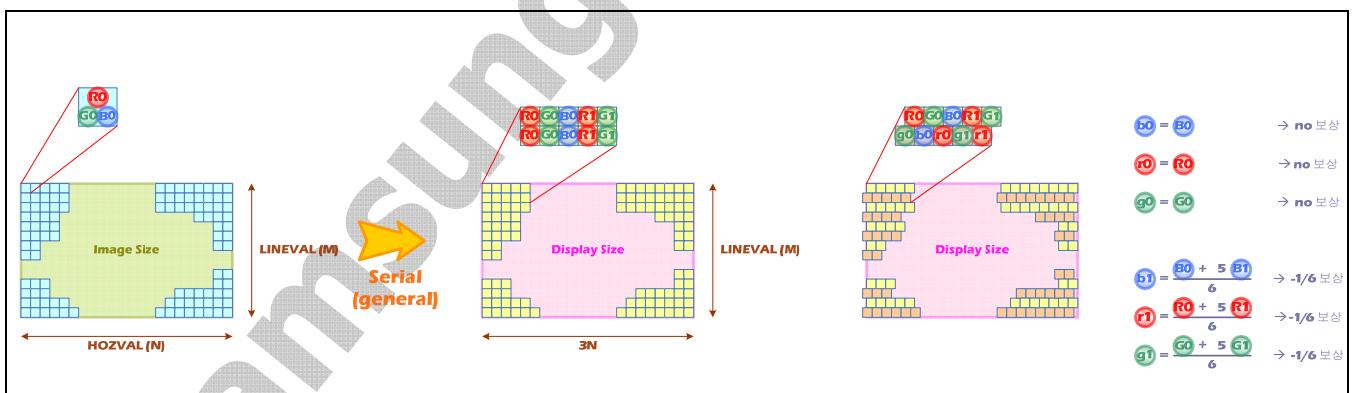


Figure 1-17 Example3. RGBSPSEL==1'b1, PIXCOMPEN_DIR==1'b1

1.3.8 VTIME CONTROLLER OPERATION

VTIME comprises of two blocks, namely:

- VTIME_RGB_TV for RGB timing control
- VTIME_I80 for indirect i80 interface timing control

1.3.8.1 RGB Interface Controller

VTIME generates control signals such as RGB_VSYNC, RGB_HSYNC, RGB_VDEN, and RGB_VCLK signal for the RGB interface. These control signals are used while configuring the VIDTCNO/ 1/2 registers in the VSFR register.

Based on the programmable configurations of display control registers in the VSFR, the VTIME module generates programmable control signals that support different types of display devices.

The RGB_VSYNC signal causes the LCD line pointer to begin at the top of display. The configuration of both HOZVAL field and LINEVAL registers control pulse generation of RGB_VSYNC and RGB_HSYNC. Based on the following equations, the size of the LCD panel determines HOZVAL and LINEVAL:

- HOZVAL = (Horizontal display size) -1
- LINEVAL = (Vertical display size) -1

The CLKVAL field in VIDCON0 register controls the rate of RGB_VCLK signal. [Table 1-5](#) defines the relationship of RGB_VCLK and CLKVAL. The minimum value of CLKVAL is 1.

- $\text{RGB_VCLK (Hz)} = \text{HCLK} / (\text{CLKVAL} + 1)$, where $\text{CLKVAL} \geq 1$

**Table 1-5 Relation 16BPP Between VCLK and CLKVAL
(TFT, Frequency of Video Clock Source=60MHz)**

CLKVAL	60MHz/X	VCLK
2	60 MHz/3	20.0 MHz
3	60 MHz/4	15.0 MHz
:	:	:
63	60 MHz/64	937.5 kHz

VSYNC, VBPD, VFPD, HSYNC, HBPD, HFDPD, HOZVAL, and LINEVAL configure RGB_HSYNC and RGB_VSYNC signal. For more information, refer to Figure 9.1-20.

The frame rate is RGB_VSYNC signal frequency. The frame rate is related to the field of VSYNC, VBPD, VFPD, LINEVAL, HSYNC, HBPD, HFDPD, HOZVAL, and CLKVAL registers. Most LCD drivers need their own adequate frame rate.

To calculate frame rate, use the following equation:

- Frame Rate = $1 / [\{ (VSPW+1) + (VBPD+1) + (LINEVAL + 1) + (VFPD+1) \} \times \{ (HSPW+1) + (HBPD +1) + (HFDPD+1) + (HOZVAL + 1) \} \times (CLKVAL + 1) / (\text{Frequency of Clock source})]$

1.3.8.2 I80 Interface Controller

VTIME_I80 controls display controller for CPU style LDI. It has the following functions:

- Generates I80 Interface Control Signals
- CPU style LDI Command Control
- Timing Control for VDMA and VDPRCS

1.3.8.3 Output Control Signal Generation

VTIME_I80 generates SYS_CS0, SYS_CS1, SYS_WE, and SYS_RS control signals (For Timing Diagram, refer to Figure 9.1-27). Their timing parameters, LCD_CS_SETUP, LCD_WR_SETUP, LCD_WR_ACT, and LCD_WR_HOLD are set through I80IFCONA0 and I80IFCONA1 SFRs.

1.3.8.4 Partial Display Control

Although partial display is the main feature of CPU style LDI, VTIME_I80 does not support this function in hardware logic.

This function is implemented by SFR setting (LINEVAL, HOZVAL, OSD_LeftTopX_F, OSD_LeftTopY_F, OSD_RightBotX_F, OSD_RightBotY_F, PAGEWIDTH, and OFFSIZE).

1.3.8.5 LDI Command Control

LDI receives both command and data. Command specifies an index for selecting the SFR in LDI. In control signal for command and data, only SYS_RS signal has a special function. Usually, SYS_RS has a polarity of '1' for issuing command and vice versa.

Display controller has two kinds of command control:

- Auto command
- Normal command

Auto command is issued automatically, that is, without software control and at a pre-defined rate (rate = 2, 4, 6, ...30). If the rate is equal to 4, it implies that auto commands are send to LDI at the end of every 4 image frames. The software control issues Normal command.

1.3.9 SETTING OF COMMANDS

1.3.9.1 Auto Command

If 0x1 (index), 0x32, 0x2 (index), 0x8f, 0x4 (index), or 0x99 are required to be sent to LDI at every 10 frames, the following steps are recommended:

- LDI_CMD0 ← 0x1, LDI_CMD1 ← 0x32, LDI_CMD2 ← 0x2,
- LDI_CMD3 ← 0x8f, LDI_CMD4 ← 0x4, LDI_CMD5 ← 0x99
- CMD0_EN ← 0x2, CMD1_EN ← 0x2, CMD2_EN ← 0x2,
- CMD3_EN ← 0x2, CMD4_EN ← 0x2, CMD5_EN ← 0x2
- CMD0_RS ← 0x1, CMD1_RS ← 0x0, CMD2_RS ← 0x1,
- CMD3_RS ← 0x0, CMD4_RS ← 0x1, CMD5_RS ← 0x0
- AUTO_CMD_RATE ← 0x5

NOTE:

1. For checking RS polarity, refer to your LDI specification.
2. It is not required to pack LDI_CMD from LDI_CMD0 to LDI_CMD11 contiguously. For example, it is only possible to use LDI_CMD0, LDI_CMD3, and LDI_CMD11.
3. Maximum 12 auto commands are available.

1.3.9.2 Normal Command

To execute Normal command, follow these steps:

Put commands into LDI_CMD0 ~ 11 (maximum 12 commands).

Set CMDx_EN in LDI_CMDCON0 to enable normal command x (For example, if you want to enable command 4, you have to set CMD4_EN to 0x01).

Set NORMAL_CMD_ST in I80IFCONB0/1.

The display controller has the following characteristics for command operations:

- Auto/ Normal/ Auto and Normal command mode is possible for each of the 12 commands.
- Sends 12 maximum commands between frames in its normal operation (Normal operation means ENVID=1 and video data is displayed in LCD panel).
- Issues commands in the order of CMD0 → CMD1 → CMD2 → CMD3 → ... → CMD10 → CMD11.
- Skips disabled commands (CMDx_EN = 0x0).
 - Sends over 12 commands (Possible in Normal command and system initialization).
 - Set 12 LDI_CMDx, CMDx_EN, and CMDx_RS.
 - Set NORMAL_CMD_ST.
 - Read NORMAL_CMD_ST with polling. If 0, go to NORMAL_CMD_ST setting.

1.2.7.2.1 Command Setting Example

- ** CMD0_EN = 2'b10, CMD1_EN = 2'b11, CMD2_EN = 2'b01, CMD3_EN = 2'b11, and CMD4_EN = 2'b01
(Auto Command: CMD0, CMD1, CMD3, Normal Command: CMD1, CMD2, CMD3, and CMD4)
- ** AUTO_COMMAND_RATE = 4'b0010 (per 4 frames)
- ** CMD0_RS = 1, CMD1_RS = 1, CMD2_RS = 0, CMD3_RS = 1, and CMD4_RS = 0.
- ** RSPOL = 0

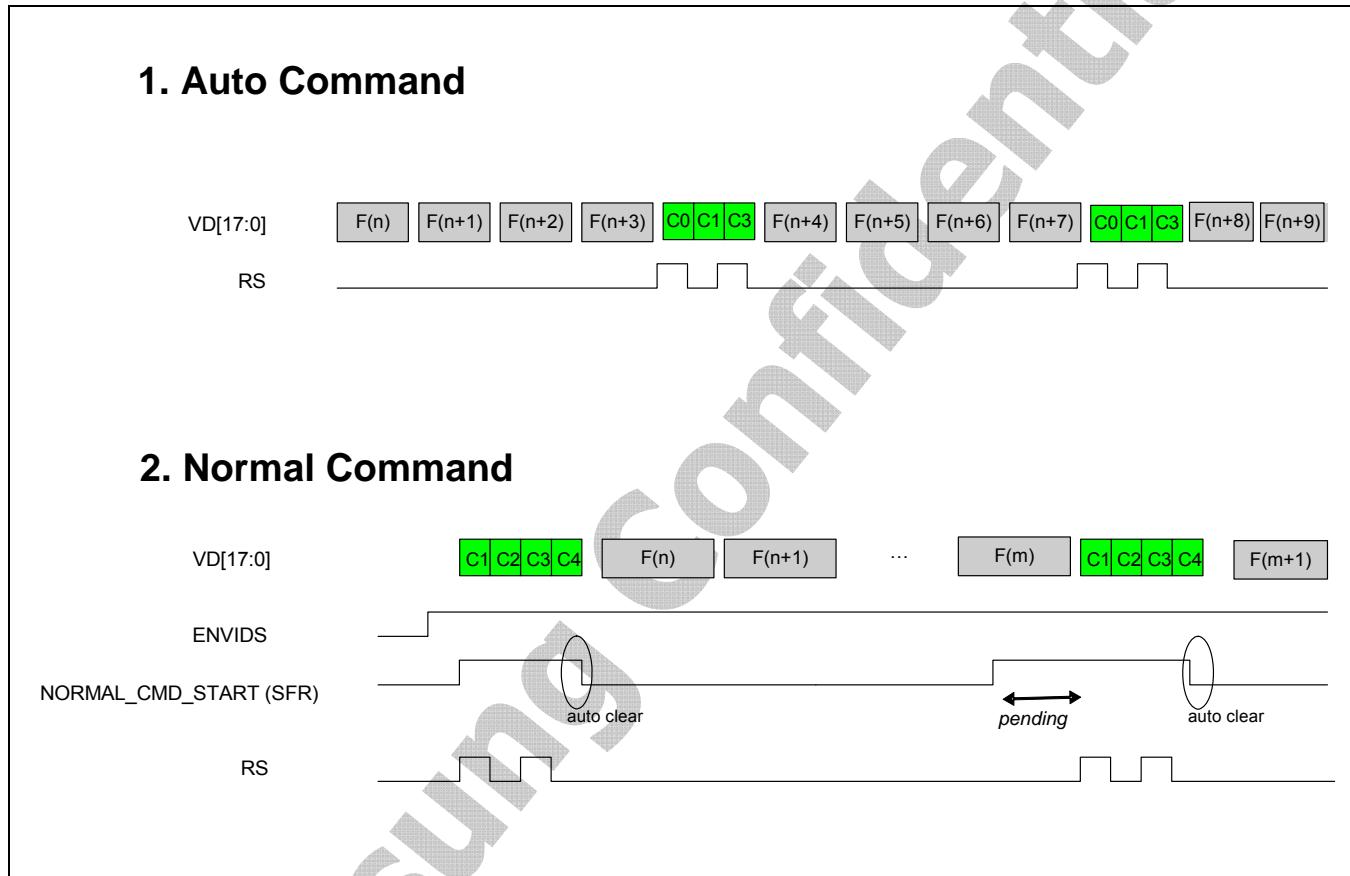


Figure 1-18 Sending Command

1.2.7.2.2 Indirect I80 Interface Trigger

VTIME_I80 starts its operation when a software trigger occurs. There are two kinds of triggers. However, software trigger is generated by setting TRGCON SFR.

1.3.9.3 Interrupt

Completion of one frame generates Frame Done Interrupt.

1.3.9.3.1 Indirect I80 Interface Output Mode

The following table shows the output mode of Indirect i80 interface based on mode@VIDCON0.

Table 1-6 i80 Output Mode

VIDCON0 Register	Value	BPP	Bus Width	Split	DATA	Command
DSI_EN	1	24	24	X	{ R[7:0],G[7:0],B[7:0] }	CMD [23:0]
L0/1_DATA	000	16	16	X	{ R[7:3],G[7:2],B[7:3] }	CMD [15:0]
	001	18	16	O (1st) (2nd)	{ R[7:2],G[7:2],B[7:4] } { 14'b0,B[3:2] }	CMD [15:0] -
	010	18	9	O (1st) (2nd)	{ R[7:2],G[7:5] } { G[4:2],B[7:2] }	CMD [17:9] CMD [8:0]
	011	24	16	O (1st) (2nd)	{ R[7:0],G[7:0] } { B[7:0], 8'b0 }	- -
	100	18	18	X	{ R[7:2],G[7:2],B[7:2] }	CMD [17:0]
	101	16	8	O (1st) (2nd)	{ R[7:3],G[7:5] } { G[4:2],B[7:3] }	CMD [15:8] CMD [7:0]

1.3.10 VIRTUAL DISPLAY

The display controller supports hardware horizontal or vertical scrolling. If the screen scrolls, change the fields of LCDBASEU and LCDBASEL (refer to [Figure 1-19](#)), but not the values of PAGEWIDTH and OFFSIZE. The size of video buffer in which the image is stored should be larger than the LCD panel screen size.

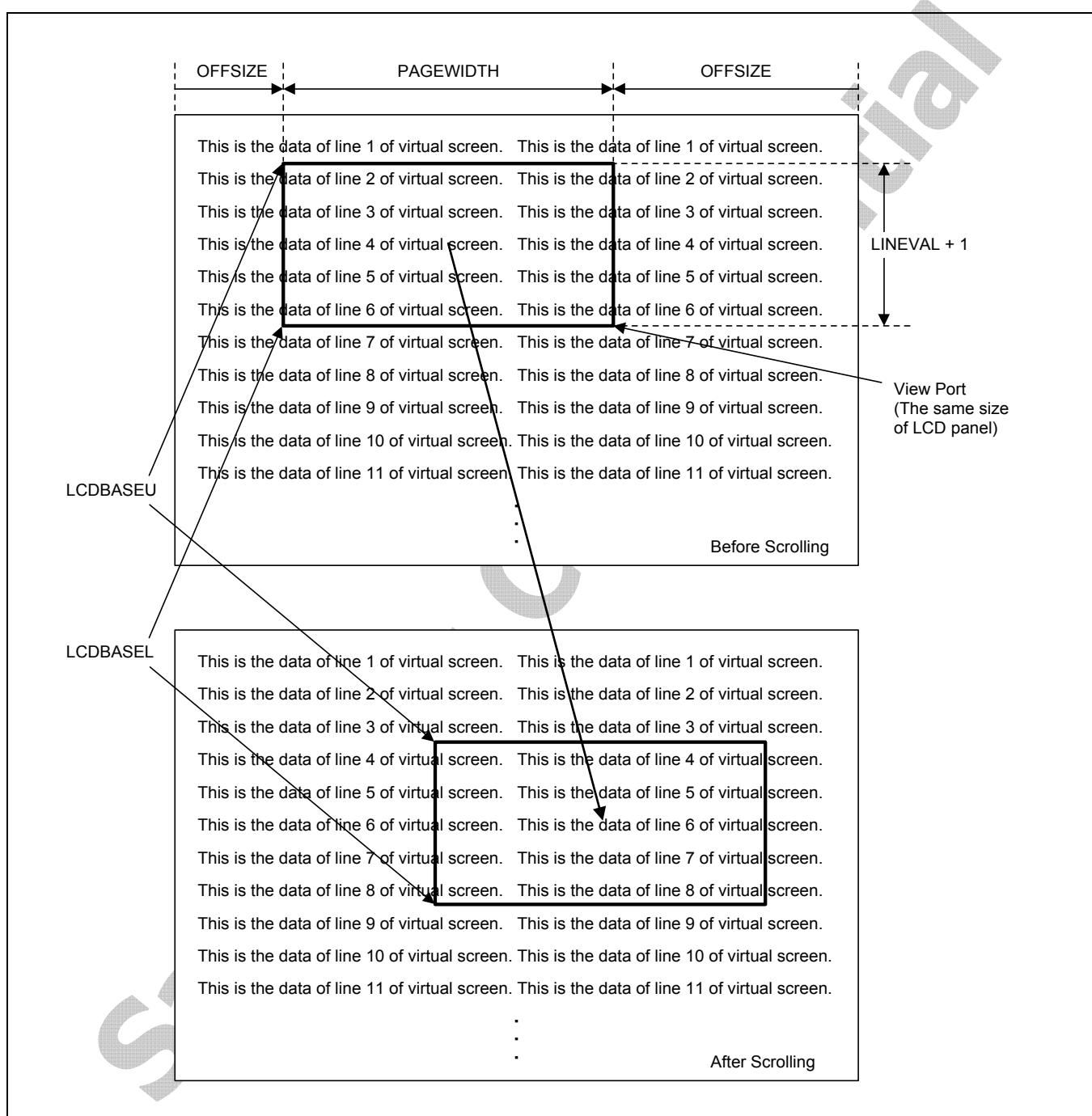


Figure 1-19 Example of Scrolling in Virtual Display

1.3.11 RGB INTERFACE SPEC

1.3.11.1 Signals

Signal	Input/Output	Description	PAD	Type
LCD_HSYNC	Output	Horizontal Sync. Signal	XvHsync	Muxed
LCD_VSYNC	Output	Vertical Sync. Signal	XvVsync	Muxed
LCD_VCLK	Output	LCD Video Clock	XvVclk	Muxed
LCD_VDEN	Output	Data Enable	XvVden	Muxed
LCD_VD[23:0]	Output	YCbCr data output	XvVd[23:0]	Muxed

NOTE: Type field indicates whether pads are dedicated to signal or pads are connected to multiplexed signals.

1.3.11.2 LCD RGB Interface Timing

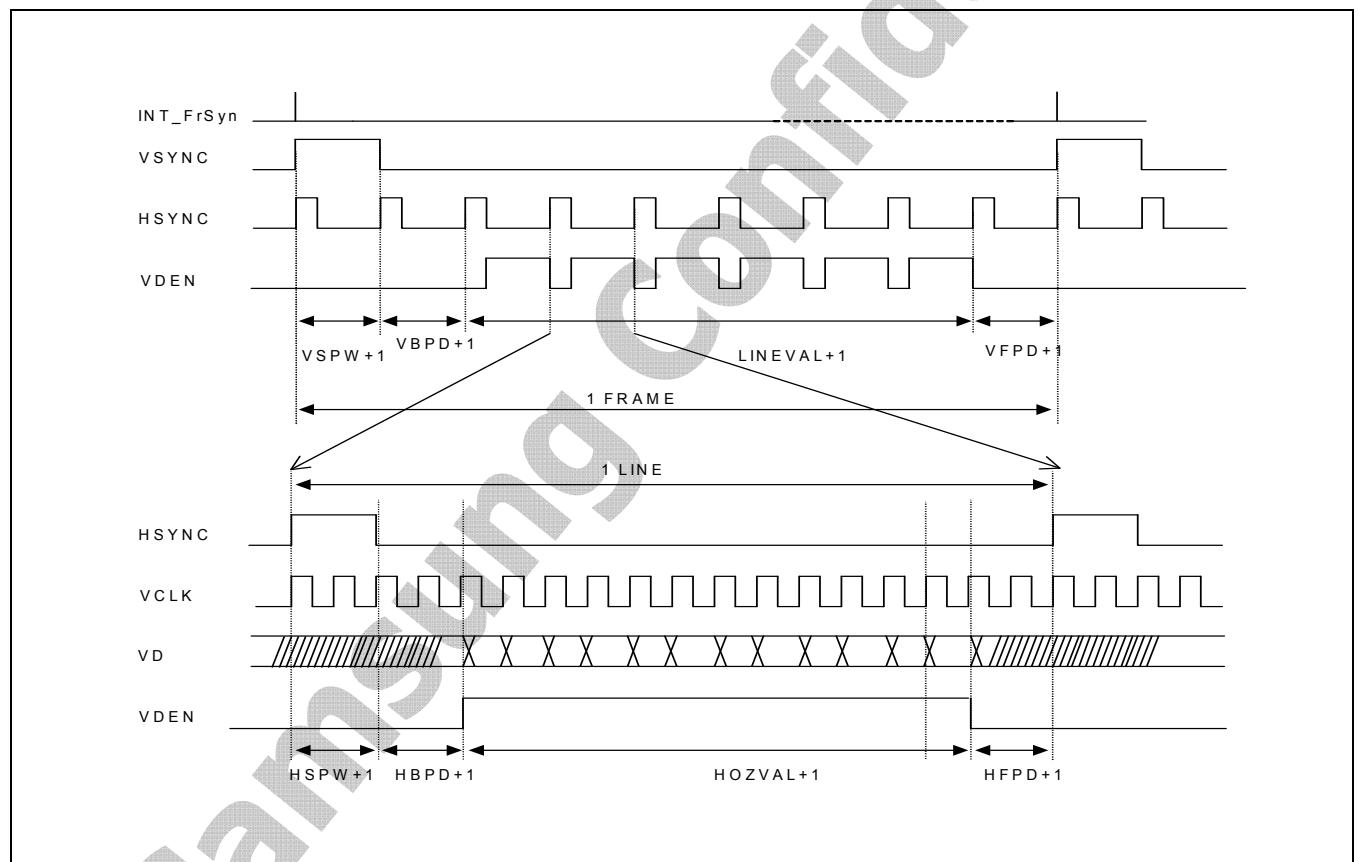


Figure 1-20 LCD RGB Interface Timing

1.3.11.3 Parallel Output

1.3.11.3.1 General 24-bit Output ($RGBSPSEL = 0$, $RGB_SKIP_EN = 0$)

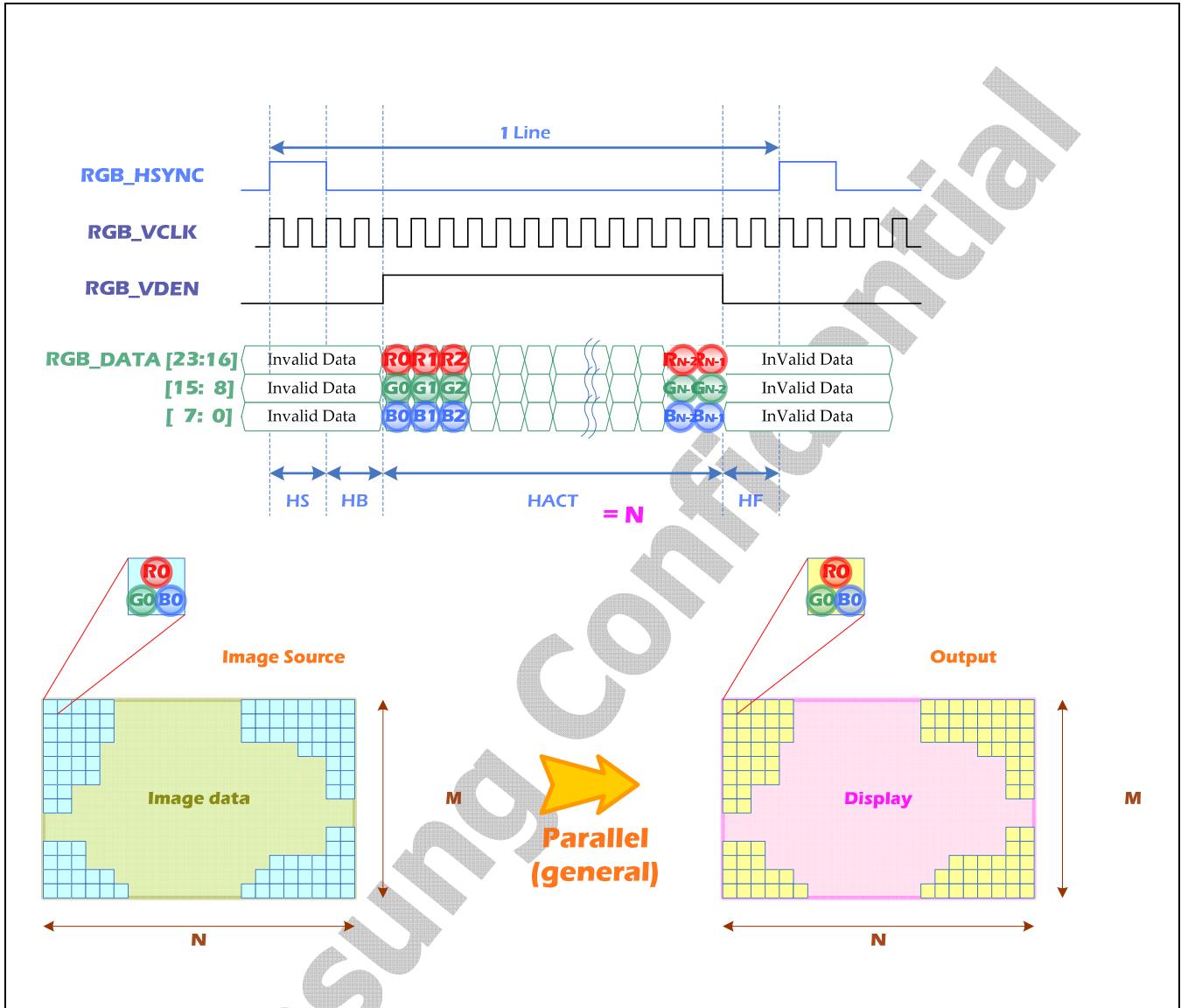


Figure 1-21 LCD RGB Interface Timing (RGB parallel)

1.3.11.3.2 RGB SKIP 8-bit output (Color sub sampling) ($RGBSPSEL = 0$, $RGB_SKIP_EN = 1$)

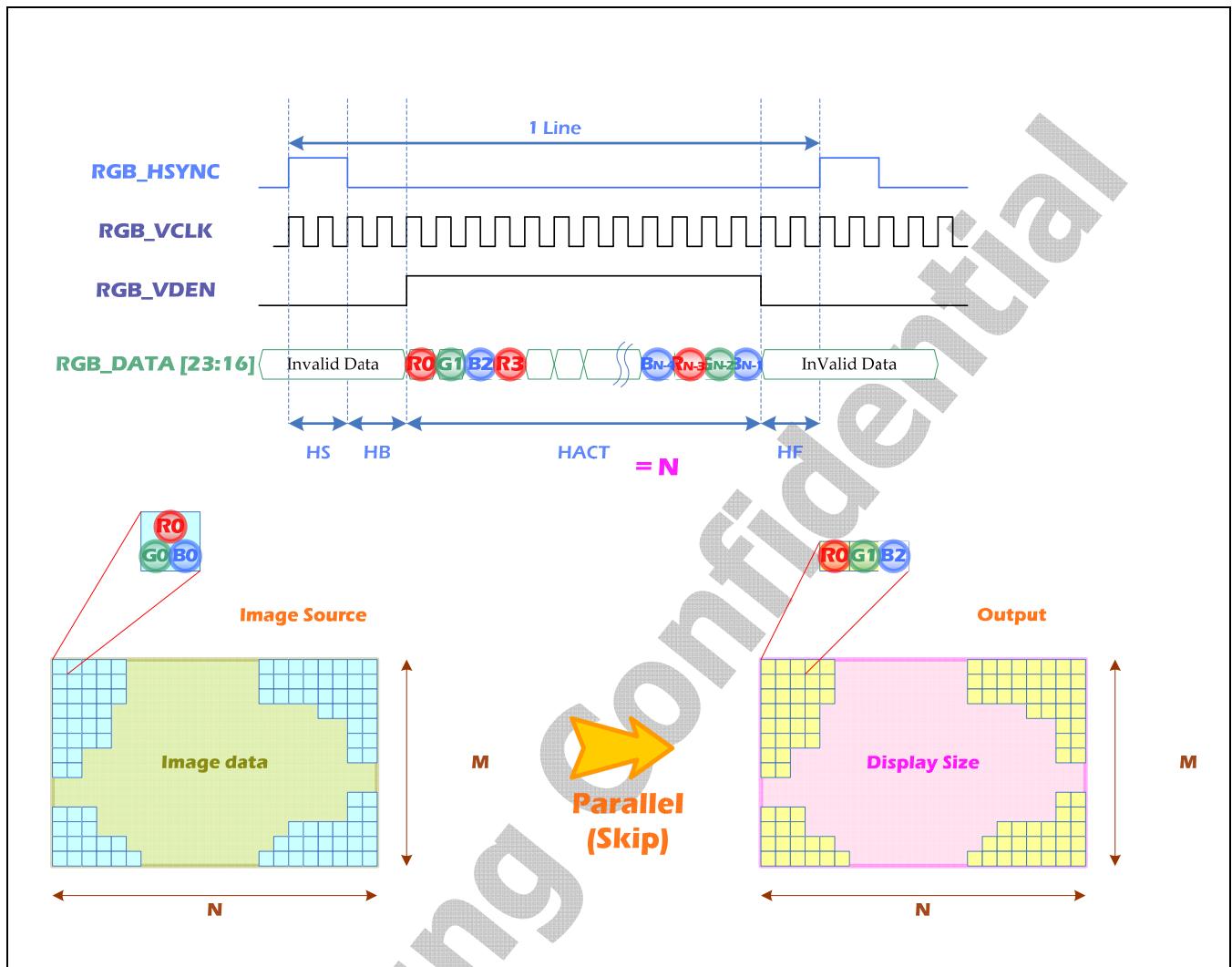


Figure 1-22 LCD RGB Interface Timing (RGB skip)

1.3.11.4 Serial 8-bit Output

1.3.11.4.1 General 8-bit output ($RGBSPSEL = 1$, $RGB_DUMMY_EN = 0$)

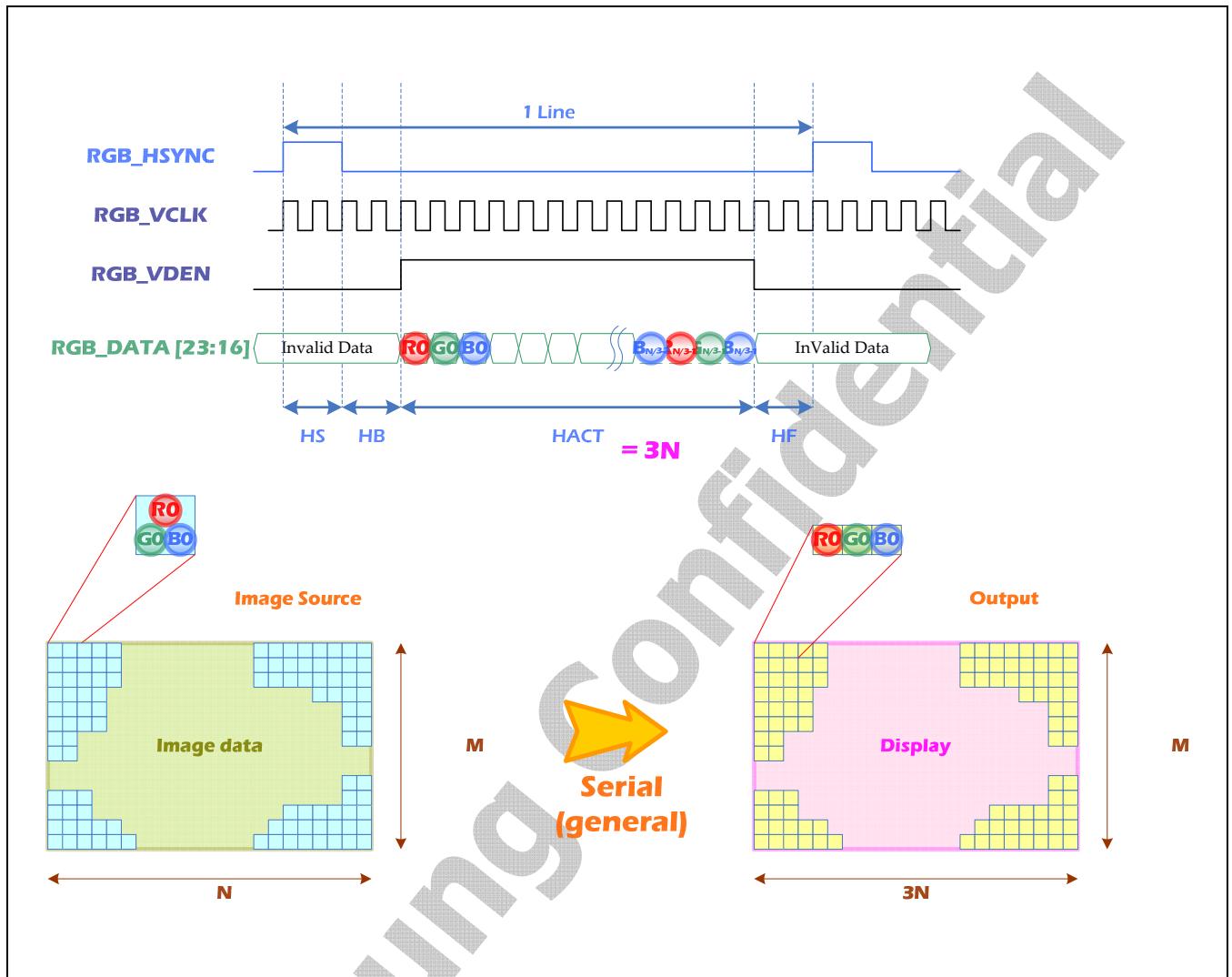


Figure 1-23 LCD RGB Interface Timing (RGB serial, Dummy disable)

1.3.11.4.2 Dummy Insertion Output ($RGBSPSEL = 1$, $RGB_DUMMY_EN = 1$)

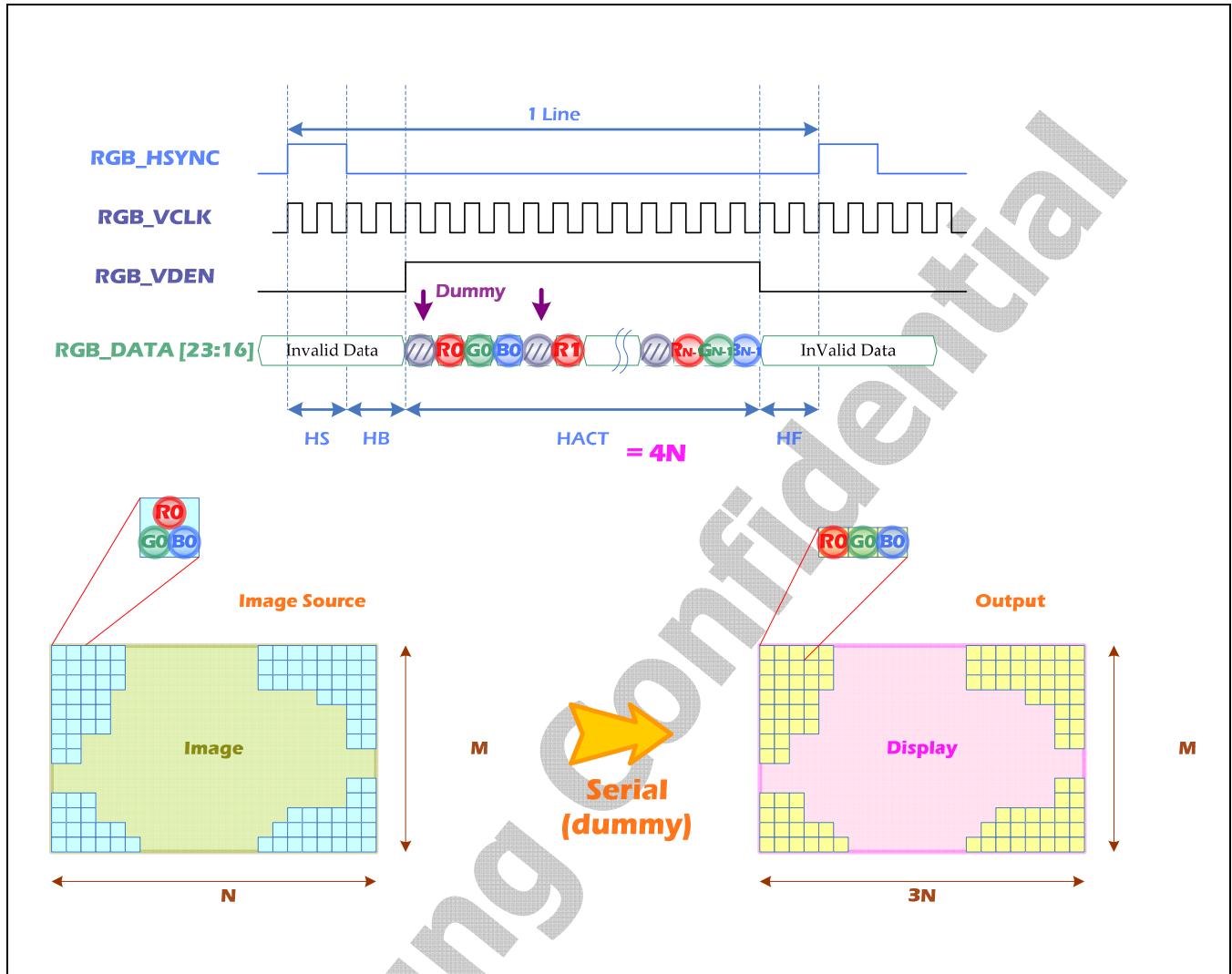


Figure 1-24 LCD RGB Interface Timing (RGB serial, Dummy insertion)

1.3.11.5 Output Configuration Structure

1.3.11.5.1 Color Order Control

'RGB_ORDER_O' controls odd line color structure. On the other hand, 'RGB_ORDER_E' @VIDCON2 controls even line color structure.

RGB_ORDER(O/E)	Output width	
	24bit	8 bit
000	[23 : 0] R0 G0 B0	1st → 2nd → 3rd R0 → G0 → B0
001	G0 B0 R0	G0 → B0 → R0
010	B0 R0 G0	B0 → R0 → G0
100	B0 G0 R0	B0 → G0 → R0
101	R0 B0 G0	R0 → B0 → G0
110	G0 R0 B0	G0 → R0 → B0

Figure 1-25 LCD RGB Output Order

1.3.11.5.2 Example of Delta Structure

For more information, refer to register 'RGB_ORDER_O,E'.

Example:

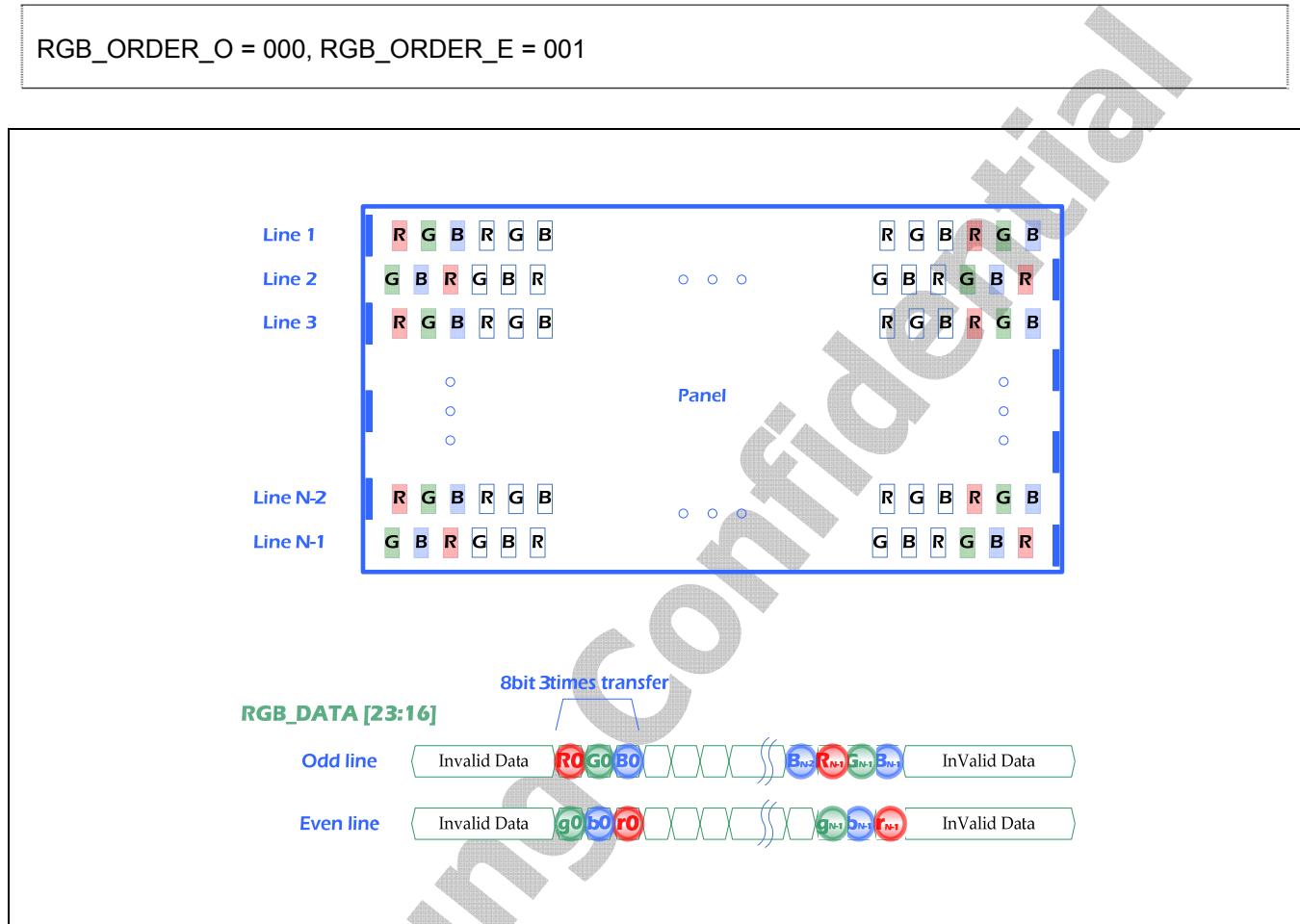


Figure 1-26 Delta Structure and LCD RGB Interface Timing

1.3.12 LCD INDIRECT I80 SYSTEM INTERFACE

1.3.12.1 Signals

Signal	Input/Output	Description	PAD	Type
SYS_VD[17:0]	Input/Output	Video Data	XvVD[17:0]	Muxed
SYS_CS0	Output	Chip select for LCD0	XvHSYNC	Muxed
SYS_CS1	Output	Chip select for LCD1	XvVSYNC	Muxed
SYS_WE	Output	Write enable	XvVCLK	Muxed
SYS_OE	Output	Output enable	XvVD[23]	Muxed
SYS_RS/SYS_ADD[0]	Output	Address Output SYS_ADD[0] is Register/ State select	XvVDEN	Muxed

NOTE: Type field indicates whether pads are dedicated to the signal, or pads are connected to the multiplexed signals.

* MIPI DSI mode (when VIDCON0 [30] =1)

SYS_ADD [1] = SYS_ST: 0 when VDOUT is from Frame

SYS_ADD [1] = SYS_ST: 1 when VDOUT is from Command

1.3.12.2 Indirect i80 System Interface WRITE Cycle Timing

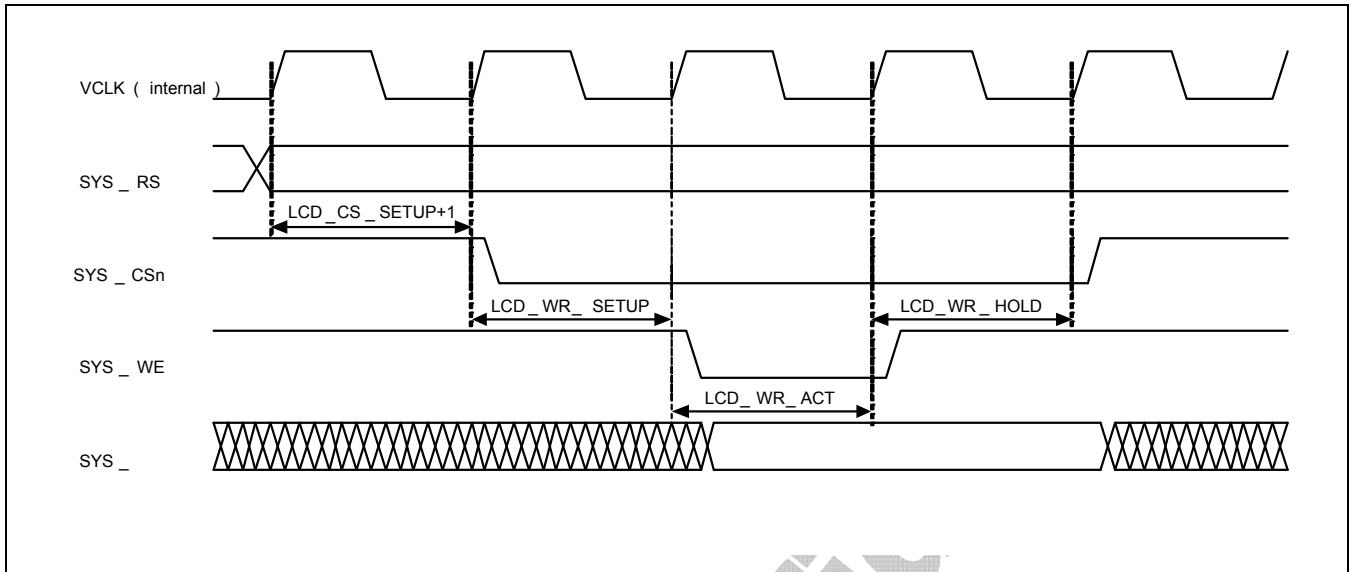


Figure 1-27 Indirect i80 System Interface WRITE Cycle Timing

Table 1-7 RGB data pin map for RGB I/F

	Parallel RGB			Serial RGB	
	24BPP (888)	18BPP (666)	16BPP (565)	24BPP (888)	18BPP (666)
VD[23]	R[7]	R[5]	R[4]	D[7]	D[5]
VD[22]	R[6]	R[4]	R[3]	D[6]	D[4]
VD[21]	R[5]	R[3]	R[2]	D[5]	D[3]
VD[20]	R[4]	R[2]	R[1]	D[4]	D[2]
VD[19]	R[3]	R[1]	R[0]	D[3]	D[1]
VD[18]	R[2]	R[0]	-	D[2]	D[0]
VD[17]	R[1]	-	-	D[1]	-
VD[16]	R[0]	-	-	D[0]	-
VD[15]	G[7]	G[5]	G[5]	-	-
VD[14]	G[6]	G[4]	G[4]	-	-
VD[13]	G[5]	G[3]	G[3]	-	-
VD[12]	G[4]	G[2]	G[2]	-	-
VD[11]	G[3]	G[1]	G[1]	-	-
VD[10]	G[2]	G[0]	G[0]	-	-
VD[9]	G[1]	-	-	-	-
VD[8]	G[0]	-	-	-	-
VD[7]	B[7]	B[5]	B[4]	-	-
VD[6]	B[6]	B[4]	B[3]	-	-
VD[5]	B[5]	B[3]	B[2]	-	-
VD[4]	B[4]	B[2]	B[1]	-	-
VD[3]	B[3]	B[1]	B[0]	-	-
VD[2]	B[2]	B[0]	-	-	-
VD[1]	B[1]	-	-	-	-
VD[0]	B[0]	-	-	-	-

Table 1-8 RGB data pin map for I80 I/F

	I80 CPU Interface (Parallel)									
	16BPP(565)	18BPP(666)	18BPP(666)	24BPP (888)	18BPP(666)	16BPP(565)	1st	2nd	1st	2nd
Lx_DATA16	000	001	010	011	100	101				
		1st	2nd	1st	2nd	1st	2nd		1st	2nd
VD[23]	-	-	-	-	-	-	-	-	-	-
VD[22]	-	-	-	-	-	-	-	-	-	-
VD[21]	-	-	-	-	-	-	-	-	-	-
VD[20]	-	-	-	-	-	-	-	-	-	-
VD[19]	-	-	-	-	-	-	-	-	-	-
VD[18]	-	-	-	-	-	-	-	-	-	-
VD[17]	-	-	-	-	-	-	-	R[5]	-	-
VD[16]	-	-	-	-	-	-	-	R[4]	-	-
VD[15]	R[4]	R[5]	-	-	R[7]	B[7]	R[3]	-	-	-
VD[14]	R[3]	R[4]	-	-	R[6]	B[6]	R[2]	-	-	-
VD[13]	R[2]	R[3]	-	-	R[5]	B[5]	R[1]	-	-	-
VD[12]	R[1]	R[2]	-	-	R[4]	B[4]	R[0]	-	-	-
VD[11]	R[0]	R[1]	-	-	R[3]	B[3]	G[5]	-	-	-
VD[10]	G[5]	R[0]	-	-	R[2]	B[2]	G[4]	-	-	-
VD[9]	G[4]	G[5]	-	-	R[1]	B[1]	G[3]	-	-	-
VD[8]	G[3]	G[4]	-	R[5]	G[2]	R[0]	B[0]	G[2]	-	-
VD[7]	G[2]	G[3]	-	R[4]	G[1]	G[7]	-	G[1]	R[4]	G[2]
VD[6]	G[1]	G[2]	-	R[3]	G[0]	G[6]	-	G[0]	R[3]	G[1]
VD[5]	G[0]	G[1]	-	R[2]	B[5]	G[5]	-	B[5]	R[2]	G[0]
VD[4]	B[4]	G[0]	-	R[1]	B[4]	G[4]	-	B[4]	R[1]	B[4]
VD[3]	B[3]	B[5]	-	R[0]	B[3]	G[3]	-	B[3]	R[0]	B[3]
VD[2]	B[2]	B[4]	-	G[5]	B[2]	G[2]	-	B[2]	G[5]	B[2]
VD[1]	B[1]	B[3]	B[1]	G[4]	B[1]	G[1]	-	B[1]	G[4]	B[1]
VD[0]	B[0]	B[2]	B[0]	G[3]	B[0]	G[0]	-	B[0]	G[3]	B[0]

1.4 PROGRAMMER'S MODEL

1.4.1 OVERVIEW OF PROGRAMMER'S MODEL

Use the following registers to configure display controller:

1. VIDCON0: Configures video output format and displays enable/disable.
2. VIDCON1: Specifies RGB I/F control signal.
3. VIDCON2: Specifies output data format control.
4. VIDCON3: Specifies image enhancement control.
5. I80IFCONx: Specifies CPU interface control signal.
6. VIDTCONx: Configures video output timing and determines the size of display.
7. WINCONx: Specifies each window feature setting.
8. VIDOSDxA, VIDOSDxB: Specifies window position setting.
9. VIDOSDxC,D: Specifies OSD size setting.
10. VIDWxALPHA0/1: Specifies alpha value setting.
11. BLEND EQx: Specifies blending equation setting.
12. VIDWxxADDx: Specifies source image address setting.
13. WxKEYCONx: Specifies color key setting register.
14. WxKEYALPHA: Specifies color key alpha value setting.
15. WINxMAP: Specifies window color control.
16. GAMMALUT_xx: Specifies gamma value setting.
17. COLOR GAIN CON: Specifies color gain value setting.
18. HUExx: Specifies Hue coefficient and offset value setting.
19. WPALCON: Specifies palette control register.
20. WxRTQOSCON: Specifies RTQoS control register.
21. WxPDATAx: Specifies Window Palette Data of each Index.
22. SHADOWCON: Specifies Shadow control register.
23. WxRTQOSCON: Specifies QoS control register.

1.5 REGISTER DESCRIPTION

1.5.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
VIDCON0	0xF800_0000	R/W	Specifies video control 0 register.	0x0000_0000
VIDCON1	0xF800_0004	R/W	Specifies video control 1 register.	0x0000_0000
VIDCON2	0xF800_0008	R/W	Specifies video control 2 register.	0x0000_0000
VIDCON3	0xF800_000C	R/W	Specifies video control 3 register.	0x0000_0000
VIDTCN0	0xF800_0010	R/W	Specifies video time control 0 register.	0x0000_0000
VIDTCN1	0xF800_0014	R/W	Specifies video time control 1 register.	0x0000_0000
VIDTCN2	0xF800_0018	R/W	Specifies video time control 2 register.	0x0000_0000
VIDTCN3	0xF800_001C	R/W	Specifies video time control 3 register.	0x0000_0000
WINCON0	0xF800_0020	R/W	Specifies window control 0 register.	0x0000_0000
WINCON1	0xF800_0024	R/W	Specifies window control 1 register.	0x0000_0000
WINCON2	0xF800_0028	R/W	Specifies window control 2 register.	0x0000_0000
WINCON3	0xF800_002C	R/W	Specifies window control 3 register.	0x0000_0000
WINCON4	0xF800_0030	R/W	Specifies window control 4 register.	0x0000_0000
SHADOWCON	0xF800_0034	R/W	Specifies window shadow control register.	0x0000_0000
VIDOSD0A	0xF800_0040	R/W	Specifies video window 0's position control register.	0x0000_0000
VIDOSD0B	0xF800_0044	R/W	Specifies video window 0's position control register.	0x0000_0000
VIDOSD0C	0xF800_0048	R/W	Specifies video window 0's size control register.	0x0000_0000
VIDOSD1A	0xF800_0050	R/W	Specifies video window 1's position control register.	0x0000_0000
VIDOSD1B	0xF800_0054	R/W	Specifies video window 1's position control register	0x0000_0000
VIDOSD1C	0xF800_0058	R/W	Specifies video window 1's alpha control register.	0x0000_0000
VIDOSD1D	0xF800_005C	R/W	Specifies video window 1's size control register.	0x0000_0000
VIDOSD2A	0xF800_0060	R/W	Specifies video window 2's position control register.	0x0000_0000
VIDOSD2B	0xF800_0064	R/W	Specifies video window 2's position control register.	0x0000_0000
VIDOSD2C	0xF800_0068	R/W	Specifies video window 2's alpha control register.	0x0000_0000
VIDOSD2D	0xF800_006C	R/W	Specifies video window 2's size control register.	0x0000_0000
VIDOSD3A	0xF800_0070	R/W	Specifies video window 3's position control register.	0x0000_0000

Register	Address	R/W	Description	Reset Value
VIDOSD3B	0xF800_0074	R/W	Specifies video window 3's position control register.	0x0000_0000
VIDOSD3C	0xF800_0078	R/W	Specifies video window 3's alpha control register.	0x0000_0000
VIDOSD4A	0xF800_0080	R/W	Specifies video window 4's position control register.	0x0000_0000
VIDOSD4B	0xF800_0084	R/W	Specifies video window 4's position control register.	0x0000_0000
VIDOSD4C	0xF800_0088	R/W	Specifies video window 4's alpha control register.	0x0000_0000
VIDW00ADD0B0	0xF800_00A0	R/W	Specifies window 0's buffer start address register, buffer 0.	0x0000_0000
VIDW00ADD0B1	0xF800_00A4	R/W	Specifies window 0's buffer start address register, buffer 1.	0x0000_0000
VIDW00ADD0B2	0xF800_20A0	R/W	Specifies window 0's buffer start address register, buffer 2.	0x0000_0000
VIDW01ADD0B0	0xF800_00A8	R/W	Specifies window 1's buffer start address register, buffer 0.	0x0000_0000
VIDW01ADD0B1	0xF800_00AC	R/W	Specifies Window 1's buffer start address register, buffer 1.	0x0000_0000
VIDW01ADD0B2	0xF800_20A8	R/W	Specifies Window 1's buffer start address register, buffer 2.	0x0000_0000
VIDW02ADD0B0	0xF800_00B0	R/W	Specifies Window 2's buffer start address register, buffer 0.	0x0000_0000
VIDW02ADD0B1	0xF800_00B4	R/W	Specifies Window 2's buffer start address register, buffer 1.	0x0000_0000
VIDW02ADD0B2	0xF800_20B0	R/W	Specifies window 2's buffer start address register, buffer 2.	0x0000_0000
VIDW03ADD0B0	0xF800_00B8	R/W	Specifies window 3's buffer start address register, buffer 0.	0x0000_0000
VIDW03ADD0B1	0xF800_00BC	R/W	Specifies window 3's buffer start address register, buffer 1.	0x0000_0000
VIDW03ADD0B2	0xF800_20B8	R/W	Specifies window 3's buffer start address register, buffer 2.	0x0000_0000
VIDW04ADD0B0	0xF800_00C0	R/W	Specifies window 4's buffer start address register, buffer 0.	0x0000_0000
VIDW04ADD0B1	0xF800_00C4	R/W	Specifies window 4's buffer start address register, buffer 1.	0x0000_0000
VIDW04ADD0B2	0xF800_20C0	R/W	Specifies window 4's buffer start address register, buffer 2.	0x0000_0000
VIDW00ADD1B0	0xF800_00D0	R/W	Specifies window 0's buffer end address register, buffer 0.	0x0000_0000

Register	Address	R/W	Description	Reset Value
VIDW00ADD1B1	0xF800_00D4	R/W	Specifies window 0's buffer end address register, buffer 1.	0x0000_0000
VIDW00ADD1B2	0xF800_20D0	R/W	Specifies window 0's buffer end address register, buffer 2.	0x0000_0000
VIDW01ADD1B0	0xF800_00D8	R/W	Specifies window 1's buffer end address register, buffer 0.	0x0000_0000
VIDW01ADD1B1	0xF800_00DC	R/W	Specifies window 1's buffer end address register, buffer 1.	0x0000_0000
VIDW01ADD1B2	0xF800_20D8	R/W	Specifies window 1's buffer end address register, buffer 2.	0x0000_0000
VIDW02ADD1B0	0xF800_00E0	R/W	Specifies window 2's buffer end address register, buffer 0.	0x0000_0000
VIDW02ADD1B1	0xF800_00E4	R/W	Specifies window 2's buffer end address register, buffer 1.	0x0000_0000
VIDW02ADD1B2	0xF800_20E0	R/W	Specifies window 2's buffer end address register, buffer 2.	0x0000_0000
VIDW03ADD1B0	0xF800_00E8	R/W	Specifies window 3's buffer end address register, buffer 0.	0x0000_0000
VIDW03ADD1B1	0xF800_00EC	R/W	Specifies window 3's buffer end address register, buffer 1.	0x0000_0000
VIDW03ADD1B2	0xF800_20E8	R/W	Specifies window 3's buffer end address register, buffer 2.	0x0000_0000
VIDW04ADD1B0	0xF800_00F0	R/W	Specifies window 4's buffer end address register, buffer 0.	0x0000_0000
VIDW04ADD1B1	0xF800_00F4	R/W	Specifies window 4's buffer end address register, buffer 1.	0x0000_0000
VIDW04ADD1B2	0xF800_20F0	R/W	Specifies window 4's buffer end address register, buffer 2.	0x0000_0000
VIDW00ADD2	0xF800_0100	R/W	Specifies window 0's buffer size register.	0x0000_0000
VIDW01ADD2	0xF800_0104	R/W	Specifies window 1's buffer size register.	0x0000_0000
VIDW02ADD2	0xF800_0108	R/W	Specifies window 2's buffer size register.	0x0000_0000
VIDW03ADD2	0xF800_010C	R/W	Specifies window 3's buffer size register.	0x0000_0000
VIDW04ADD2	0xF800_0110	R/W	Specifies window 4's buffer size register.	0x0000_0000
VIDINTCON0	0xF800_0130	R/W	Specifies video interrupt control register.	0x0000_0000
VIDINTCON1	0xF800_0134	R/W	Specifies video interrupt pending register.	0x0000_0000
W1KEYCON0	0xF800_0140	R/W	Specifies color key control register.	0x0000_0000
W1KEYCON1	0xF800_0144	R/W	Specifies color key value (transparent value) register.	0x0000_0000
W2KEYCON0	0xF800_0148	R/W	Specifies color key control register.	0x0000_0000
W2KEYCON1	0xF800_014C	R/W	Specifies color key value (transparent value) register.	0x0000_0000

Register	Address	R/W	Description	Reset Value
W3KEYCON0	0xF800_0150	R/W	Specifies color key control register.	0x0000_0000
W3KEYCON1	0xF800_0154	R/W	Specifies color key value (transparent value) register.	0x0000_0000
W4KEYCON0	0xF800_0158	R/W	Specifies color key control register.	0x0000_0000
W4KEYCON1	0xF800_015C	R/W	Specifies color key value (transparent value) register.	0x0000_0000
W1KEYALPHA	0xF800_0160	R/W	Specifies color key alpha value register.	0x0000_0000
W2KEYALPHA	0xF800_0164	R/W	Specifies color key alpha value register.	0x0000_0000
W3KEYALPHA	0xF800_0168	R/W	Specifies color key alpha value register.	0x0000_0000
W4KEYALPHA	0xF800_016C	R/W	Specifies color key alpha value register.	0x0000_0000
DITHMODE	0xF800_0170	R/W	Specifies dithering mode register.	0x0000_0000
WIN0MAP	0xF800_0180	R/W	Specifies window 0's color control.	0x0000_0000
WIN1MAP	0xF800_0184	R/W	Specifies window 1's color control.	0x0000_0000
WIN2MAP	0xF800_0188	R/W	Specifies window 2's color control.	0x0000_0000
WIN3MAP	0xF800_018C	R/W	Specifies window 3's color control.	0x0000_0000
WIN4MAP	0xF800_0190	R/W	Specifies window 4's color control.	0x0000_0000
WPALCON_H	0xF800_019c	R/W	Specifies window palette control register.	0x0000_0000
WPALCON_L	0xF800_01A0	R/W	Specifies window palette control register.	0x0000_0000
TRIGCON	0xF800_01A4	R/W	Specifies i80/ RGB trigger control register.	0x0000_0000
I80IFCONA0	0xF800_01B0	R/W	Specifies i80 interface control 0 for main LDI.	0x0000_0000
I80IFCONA1	0xF800_01B4	R/W	Specifies i80 interface control 0 for sub LDI.	0x0000_0000
I80IFCONB0	0xF800_01B8	R/W	Specifies i80 interface control 1 for main LDI.	0x0000_0000
I80IFCONB1	0xF800_01BC	R/W	Specifies i80 interface control 1 for sub LDI.	0x0000_0000
COLORGAINCON	0xF800_01C0	R/W	Specifies color gain control register.	0x1004_0100
LDI_CMDCON0	0xF800_01D0	R/W	Specifies i80 interface LDI command control 0.	0x0000_0000
LDI_CMDCON1	0xF800_01D4	R/W	Specifies i80 interface LDI command control 1.	0x0000_0000
SIFCCON0	0xF800_01E0	R/W	Specifies LCD i80 system interface command control 0.	0x0000_0000
SIFCCON1	0xF800_01E4	R/W	Specifies LCD i80 system interface command control 1.	0x0000_0000
SIFCCON2	0xF800_01E8	R	Specifies LCD i80 system interface command control 2.	0x????_????
HUECOEF00	0xF800_01EC	R/W	Specifies Hue coefficient control register.	0x0100_0100
HUECOEF01	0xF800_01F0	R/W	Specifies Hue coefficient control register.	0x0000_0000

Register	Address	R/W	Description	Reset Value
HUECOEF10	0xF800_01F4	R/W	Specifies Hue coefficient control register.	0x0000_0000
HUECOEF11	0xF800_01F8	R/W	Specifies Hue coefficient control register.	0x0100_0100
HUEOFFSET	0xF800_01FC	R/W	Specifies Hue offset control register.	0x0180_0080
VIDW0ALPHA0	0xF800_0200	R/W	Specifies window 0's alpha value 0 register.	0x0000_0000
VIDW0ALPHA1	0xF800_0204	R/W	Specifies window 0's alpha value 1 register.	0x0000_0000
VIDW1ALPHA0	0xF800_0208	R/W	Specifies window 1's alpha value 0 register.	0x0000_0000
VIDW1ALPHA1	0xF800_020c	R/W	Specifies window 1's alpha value 1 register.	0x0000_0000
VIDW2ALPHA0	0xF800_0210	R/W	Specifies window 2's alpha value 0 register.	0x0000_0000
VIDW2ALPHA1	0xF800_0214	R/W	Specifies window 2's alpha value 1 register.	0x0000_0000
VIDW3ALPHA0	0xF800_0218	R/W	Specifies window 3's alpha value 0 register.	0x0000_0000
VIDW3ALPHA1	0xF800_021c	R/W	Specifies window 3's alpha value 1 register.	0x0000_0000
VIDW4ALPHA0	0xF800_0220	R/W	Specifies window 4's alpha value 0 register.	0x0000_0000
VIDW4ALPHA1	0xF800_0224	R/W	Specifies window 4's alpha value 1 register.	0x0000_0000
BLENDEQ1	0xF800_0244	R/W	Specifies window 1's blending equation control register.	0x0000_00c2
BLENDEQ2	0xF800_0248	R/W	Specifies window 2's blending equation control register.	0x0000_00c2
BLENDEQ3	0xF800_024c	R/W	Specifies window 3's blending equation control register.	0x0000_00c2
BLENDEQ4	0xF800_0250	R/W	Specifies window 4's blending equation control register.	0x0000_00c2
BLENDCON	0xF800_0260	R/W	Specifies blending control register.	0x0000_0000
W0RTQOSCON	0xF800_0264	R/W	Specifies window 0's RTQOS control register.	0x0000_0000
W1RTQOSCON	0xF800_0268	R/W	Specifies window 1's RTQOS control register.	0x0000_0000
W2RTQOSCON	0xF800_026C	R/W	Specifies window 2's RTQOS control register.	0x0000_0000
W3RTQOSCON	0xF800_0270	R/W	Specifies window 3's RTQOS control register.	0x0000_0000
W4RTQOSCON	0xF800_0274	R/W	Specifies window 4's RTQOS control register.	0x0000_0000
LDI_CMD0	0xF800_0280	R/W	Specifies i80 interface LDI command 0.	0x0000_0000
LDI_CMD1	0xF800_0284	R/W	Specifies i80 interface LDI command 1.	0x0000_0000
LDI_CMD2	0xF800_0288	R/W	Specifies i80 interface LDI command 2.	0x0000_0000
LDI_CMD3	0xF800_028C	R/W	Specifies i80 interface LDI command 3.	0x0000_0000
LDI_CMD4	0xF800_0290	R/W	Specifies i80 interface LDI command 4.	0x0000_0000
LDI_CMD5	0xF800_0294	R/W	Specifies i80 interface LDI command 5.	0x0000_0000

Register	Address	R/W	Description	Reset Value
LDI_CMD6	0xF800_0298	R/W	Specifies i80 interface LDI command 6.	0x0000_0000
LDI_CMD7	0xF800_029C	R/W	Specifies i80 interface LDI command 7.	0x0000_0000
LDI_CMD8	0xF800_02A0	R/W	Specifies i80 interface LDI command 8.	0x0000_0000
LDI_CMD9	0xF800_02A4	R/W	Specifies i80 interface LDI command 9.	0x0000_0000
LDI_CMD10	0xF800_02A8	R/W	Specifies i80 interface LDI command 10.	0x0000_0000
LDI_CMD11	0xF800_02AC	R/W	Specifies i80 interface LDI command 11.	0x0000_0000
GAMMALUT_01_00	0xF800_037C	R/W	Specifies Gamma LUT data of the index 0, 1.	0x0010_0000
GAMMALUT_03_02	0xF800_0380	R/W	Specifies Gamma LUT data of the index 2, 3.	0x0030_0020
GAMMALUT_05_04	0xF800_0384	R/W	Specifies Gamma LUT data of the index 4, 5.	0x0050_0040
GAMMALUT_07_06	0xF800_0388	R/W	Specifies Gamma LUT data of the index 6, 7.	0x0070_0060
GAMMALUT_09_08	0xF800_038c	R/W	Specifies Gamma LUT data of the index 8, 9.	0x0090_0080
GAMMALUT_11_10	0xF800_0390	R/W	Specifies Gamma LUT data of the index 10, 11.	0x00b0_00a0
GAMMALUT_13_12	0xF800_0394	R/W	Specifies Gamma LUT data of the index 12, 13.	0x00d0_00c0
GAMMALUT_15_14	0xF800_0398	R/W	Specifies Gamma LUT data of the index 14, 15.	0x00f0_00e0
GAMMALUT_17_16	0xF800_039C	R/W	Specifies Gamma LUT data of the index 16, 17.	0x0110_0100
GAMMALUT_19_18	0xF800_03a0	R/W	Specifies Gamma LUT data of the index 18, 19.	0x0130_0120
GAMMALUT_21_20	0xF800_03a4	R/W	Specifies Gamma LUT data of the index 20, 21.	0x0150_0140
GAMMALUT_23_22	0xF800_03a8	R/W	Specifies Gamma LUT data of the index 22, 23.	0x0170_0160
GAMMALUT_25_24	0xF800_03ac	R/W	Specifies Gamma LUT data of the index 24, 25.	0x0190_0180
GAMMALUT_27_26	0xF800_03b0	R/W	Specifies Gamma LUT data of the index 26, 27.	0x01b0_01a0
GAMMALUT_29_28	0xF800_03b4	R/W	Specifies Gamma LUT data of the index 28, 29.	0x01d0_01c0
GAMMALUT_31_30	0xF800_03b8	R/W	Specifies Gamma LUT data of the index 30, 31.	0x01f0_01e0
GAMMALUT_33_32	0xF800_03bc	R/W	Specifies Gamma LUT data of the index 32, 33.	0x0210_0200

Register	Address	R/W	Description	Reset Value
GAMMALUT_35_34	0xF800_03c0	R/W	Specifies Gamma LUT data of the index 34, 35.	0x0230_0220
GAMMALUT_37_36	0xF800_03c4	R/W	Specifies Gamma LUT data of the index 36, 37.	0x0250_0240
GAMMALUT_39_38	0xF800_03c8	R/W	Specifies Gamma LUT data of the index 38, 39.	0x0270_0260
GAMMALUT_41_40	0xF800_03cc	R/W	Specifies Gamma LUT data of the index 40, 41.	0x0290_0280
GAMMALUT_43_42	0xF800_03d0	R/W	Specifies Gamma LUT data of the index 42, 43.	0x02b0_02a0
GAMMALUT_45_44	0xF800_03d4	R/W	Specifies Gamma LUT data of the index 44, 45.	0x02d0_02c0
GAMMALUT_47_46	0xF800_03d8	R/W	Specifies Gamma LUT data of the index 46, 47.	0x02f0_02e0
GAMMALUT_49_48	0xF800_03dc	R/W	Specifies Gamma LUT data of the index 48, 49.	0x0310_0300
GAMMALUT_51_50	0xF800_03e0	R/W	Specifies Gamma LUT data of the index 50, 51.	0x0330_0320
GAMMALUT_53_52	0xF800_03e4	R/W	Specifies Gamma LUT data of the index 52, 53.	0x0350_0340
GAMMALUT_55_54	0xF800_03e8	R/W	Specifies Gamma LUT data of the index 54, 55.	0x0370_0360
GAMMALUT_57_56	0xF800_03ec	R/W	Specifies Gamma LUT data of the index 56, 57.	0x0390_0380
GAMMALUT_59_58	0xF800_03f0	R/W	Specifies Gamma LUT data of the index 58, 59.	0x03b0_03a0
GAMMALUT_61_60	0xF800_03f4	R/W	Specifies Gamma LUT data of the index 60, 61.	0x03d0_03c0
GAMMALUT_63_62	0xF800_03f8	R/W	Specifies Gamma LUT data of the index 62, 63.	0x03f0_03e0
GAMMALUT_xx_64	0xF800_03fc	R/W	Specifies Gamma LUT data of the index 64.	0x0000_0400
SHD_VIDW00AD D0	0xF800_40A0	R	Specifies window 0's buffer start address register (shadow).	0x0000_0000
SHD_VIDW01AD D0	0xF800_40A8	R	Specifies window 1's buffer start address register (shadow).	0x0000_0000
SHD_VIDW02AD D0	0xF800_40B0	R	Specifies window 2's buffer start address register (shadow).	0x0000_0000
SHD_VIDW03AD D0	0xF800_40B8	R	Specifies window 3's buffer start address register (shadow).	0x0000_0000
SHD_VIDW04AD D0	0xF800_40C0	R	Specifies window 4's buffer start address register (shadow).	0x0000_0000

Register	Address	R/W	Description	Reset Value
SHD_VIDW00AD D1	0xF800_40D0	R	Specifies window 0's buffer end address register (shadow)	0x0000_0000
SHD_VIDW01AD D1	0xF800_40D8	R	Specifies window 1's buffer end address register (shadow)	0x0000_0000
SHD_VIDW02AD D1	0xF800_40E0	R	Specifies window 2's buffer end address register (shadow).	0x0000_0000
SHD_VIDW03AD D1	0xF800_40E8	R	Specifies window 3's buffer end address register (shadow).	0x0000_0000
SHD_VIDW04AD D1	0xF800_40F0	R	Specifies window 4's buffer end address register (shadow).	0x0000_0000
SHD_VIDW00AD D2	0xF800_4100	R	Specifies window 0's buffer size register (shadow).	0x0000_0000
SHD_VIDW01AD D2	0xF800_4104	R	Specifies window 1's buffer size register (shadow).	0x0000_0000
SHD_VIDW02AD D2	0xF800_4108	R	Specifies window 2's buffer size register (shadow).	0x0000_0000
SHD_VIDW03AD D2	0xF800_410C	R	Specifies window 3's buffer size register (shadow).	0x0000_0000
SHD_VIDW04AD D2	0xF800_4110	R	Specifies window 4's buffer size register (shadow).	0x0000_0000

1.5.2 PALETTE MEMORY (PALRAM)

	Start Address	End Address	R/W	Description	Reset Value
Win0 PalRam	0xF800_2400 (0xF800_0400)	0xF800_27FC (0xF800_07FC)	R/W	Specifies 0~255 entry palette data.	Undefined
Win1 PalRam	0xF800_2800 (0xF800_0800)	0xF800_2BFC (0xF800_0BFC)	R/W	Specifies 0~255 entry palette data.	Undefined
Win2 PalRam	0xF800_2C00	0xF800_2FFC	R/W	Specifies 0~255 entry palette data.	Undefined
Win3 PalRam	0xF800_3000	0xF800_33FC	R/W	Specifies 0~255 entry palette data.	Undefined
Win4 PalRam	0xF800_3400	0xF800_37FC	R/W	Specifies 0~255 entry palette data.	Undefined

1.5.2.1 Video Main Control 0 Register (VIDCON0, R/W, Address = 0xF800_0000)

VIDCON0	Bit	Description	Initial State
Reserved	[31]	Reserved (should be 0).	0
DSI_EN	[30]	Enables MIPI DSI. 0 = Disables 1 = Enables (i80 24bit data interface, SYS_ADD[1])	0
Reserved	[29]	Reserved (should be 0)	0
VIDOUT	[28:26]	Determines the output format of Video Controller. 000 = RGB interface 001 = Reserved 010 = Indirect I80 interface for LDI0 011 = Indirect I80 interface for LDI1 100 = WB interface and RGB interface 101 = Reserved 110 = WB Interface and i80 interface for LDI0 111 = WB Interface and i80 interface for LDI1	000
L1_DATA16	[25:23]	Selects output data format mode of indirect i80 interface (LDI1). (VIDOUT[1:0] == 2'b11) 000 = 16-bit mode (16 bpp) 001 = 16 + 2-bit mode (18 bpp) 010 = 9 + 9-bit mode (18 bpp) 011 = 16 + 8-bit mode (24 bpp) 100 = 18-bit mode (18bpp) 101 = 8 + 8-bit mode (16bpp)	000
L0_DATA16	[22:20]	Selects output data format mode of indirect i80 interface (LDI0). (VIDOUT[1:0] == 2'b10) 000 = 16-bit mode (16 bpp) 001 = 16 + 2-bit mode (18 bpp) 010 = 9 + 9-bit mode (18 bpp) 011 = 16 + 8-bit mode (24 bpp) 100 = 18-bit mode (18bpp) 101 = 8 + 8-bit mode (16bpp)	000
Reserved	[19]	Reserved (should be 0).	0
RGSPSEL	[18]	Selects display mode (VIDOUT[1:0] == 2'b00). 0 = RGB parallel format 1 = RGB serial format Selects the display mode (VIDOUT[1:0] != 2'b00). 0 = RGB parallel format	0
PNRMODE	[17]	Controls inverting RGB_ORDER (@VIDCON3). 0 = Normal: RGBORDER[2] @VIDCON3 1 = Invert: ~RGBORDER[2] @VIDCON3 Note: This bit is used for the previous version of FIMD. You do not have to use this bit if you use RGB_ORDER@VIDCON3 register.	00
CLKVALUP	[16]	Selects CLKVAL_F update timing control. 0 = Always 1 = Start of a frame (only once per frame)	0

VIDCON0	Bit	Description	Initial State
Reserved	[15:14]	Reserved.	0
CLKVAL_F	[13:6]	Determines the rates of VCLK and CLKVAL[7:0]. VCLK = HCLK / (CLKVAL+1), where CLKVAL >= 1 Notes. 1. The maximum frequency of VCLK is 100Mhz(pad:50pf). 2. CLKSEL_F register selects Video Clock Source.	0
VCLKFREE	[5]	Controls VCLK Free Run (Only valid at RGB IF mode). 0 = Normal mode (controls using ENVID) 1 = Free-run mode	0
CLKDIR	[4]	Selects the clock source as direct or divide using CLKVAL_F register. 0 = Direct clock (frequency of VCLK = frequency of Clock source) 1 = Divided by CLKVAL_F	0x00
Reserved	[3]	Should be 0.	0x0
CLKSEL_F	[2]	Selects the video clock source. 0 = HCLK 1 = SCLK_FIMD HCLK is the bus clock, whereas SCLK_FIMD is the special clock for display controller. For more information, refer to Chapter, "02.03 CLOCK CONTROLLER".	0
ENVID	[1]	Enables/ disables video output and logic immediately. 0 = Disables the video output and display control signal. 1 = Enables the video output and display control signal.	0
ENVID_F	[0]	Enables/ disables video output and logic at current frame end. 0 = Disables the video output and display control signal. 1 = Enables the video output and display control signal. * If this bit is set to "on" and "off", then "H" is read and video controller is enabled until the end of current frame.	0

NOTE: Display On: ENVID and ENVID_F are set to "1".

Direct Off: ENVID and ENVID_F are set to "0" simultaneously.

Per Frame Off: ENVID_F is set to "0" and ENVID is set to "1".

Caution: 1: If VIDCON0 is set for Per Frame Off in interlace mode, the value of INTERLACE_F should be set to "0" in the same time.
2: If display controller is off using direct off, it is impossible to turn on the display controller without reset.

1.5.2.2 Video Main Control 1 Register (VIDCON1, R/W, Address = 0xF800_0004)

VIDCON1	Bit	Description	Initial State
LINECNT (read only)	[26:16]	Provides the status of the line counter (read only). Up count from 0 to LINEVAL.	0
FSTATUS	[15]	Specifies the Field Status (read only). 0 = ODD Field 1 = EVEN Field	0
VSTATUS	[14:13]	Specifies the Vertical Status (read only). 00 = VSYNC 01 = BACK Porch 10 = ACTIVE 11 = FRONT Porch	0
Reserved	[12:11]	Reserved	0
FIXVCLK	[10:9]	Specifies the VCLK hold scheme at data under-flow. 00 = VCLK hold 01 = VCLK running 11 = VCLK running and VDEN disable	0
Reserved	[8]	Reserved	0
IVCLK	[7]	Controls the polarity of the VCLK active edge. 0 = Video data is fetched at VCLK falling edge 1 = Video data is fetched at VCLK rising edge	0
IHSYNC	[6]	Specifies the HSYNC pulse polarity. 0 = Normal 1 = Inverted	0
IVSYNC	[5]	Specifies the VSYNC pulse polarity. 0 = Normal 1 = Inverted	0
IVDEN	[4]	Specifies the VDEN signal polarity. 0 = Normal 1 = Inverted	0
Reserved	[3:0]	Reserved	0x0

1.5.2.3 Video Main Control 2 Register (VIDCON2, R/W, Address = 0xF800_0008)

VIDCON2	Bit	Description	Initial State
Reserved	[31:28]	Reserved	0
RGB_SKIP_EN	[27]	Enables the RGB skip mode (only where RGBSPSEL == 1'b0). 0 = Disables 1 = Enables	0
Reserved	[26]	Reserved	0
RGB_DUMMY_LOC	[25]	Controls RGB dummy insertion location (only where RGBSPSEL == 1'b1 and RGB_DUMMY_EN == 1'b1) 0 = Last (4th) position 1 = 1st position	0
RGB_DUMMY_EN	[24]	Enables RGB dummy insertion mode (only where RGBSPSEL == 1'b1) 0 = Disables 1 = Enables	0
Reserved	[23:22]	Reserved (should be 0)	0
RGB_ORDER_E	[21:19]	Controls RGB interface output order (Even line, line # 2, 4, 6, 8.) , where, RGBSPSEL== 1'b0 000 = RGB 001 = GBR 010 = BRG 100 = BGR 101 = RBG 110 = GRB , where, (RGBSPSEL == 1'b1) or (RGBSPSEL == 1'b0 and RGB_SKIP_EN = 1'b1) 000 = R→G→B 001 = G→B→R 010 = B→R→G 100 = B→G→R 101 = R→B→G 110 = G→R→B Note: PNR0[0] @VIDCON0 should be 0, when you use RGB_ORDER_O[2:0] @VIDCON3 register.	0
RGB_ORDER_O	[18:16]	Controls RGB interface output order (Odd Line, line #1, 3, 5, 7.) , where, RGBSPSEL== 1'b0 000 = RGB 001 = GBR 010 = BRG 100 = BGR 101 = RBG 110 = GRB , where, (RGBSPSEL == 1'b1) or (RGBSPSEL == 1'b0 and RGB_SKIP_EN = 1'b1) 000 = R→G→B 001 = G→B→R	0

VIDCON2	Bit	Description	Initial State
		010 = B→R→G 100 = B→G→R 101 = R→B→G 110 = G→R→B Note: PNR0[0] @VIDCON0 should be 0, when you use RGB_ORDER_E[2:0]@VIDCON3 register.	
Reserved	[15:14]	Reserved. Note: This bit should be 1.	0
TVFORMATSEL	[13:12]	Specifies the output format of YUV data. 00 = Reserved 01 = YUV422 1x = YUV444	0
Reserved	[11:9]	Reserved	0
OrgYCbCr	[8]	Specifies the order of YUV data. 0 = Y - CbCr 1 = CbCr - Y	0
YUVOrd	[7]	Specifies the order of Chroma data. 0 = Cb - Cr 1 = Cr - Cb	0
Reserved	[6:5]	Reserved	0
WB_FRAME_SKIP	[4:0]	Controls the WB frame skip rate. The maximum rate is up to 1:30 [only where (VIDOUT[2:0] == 3'b001 or 3'b100 TV encoder interface), (INTERLACE_F==1'b0) and (TV422 or TVRGB output)]. 00000 = no skip (1 : 1) 00001 = skip rate = 1 : 2 00010 = skip rate = 1 : 3 ... 11101 = skip rate = 1 : 30 1111x = reserved	0

1.5.2.4 Video Main Control 3 Register (vidcon3, R/W, ADDRESS = 0xF800_000C)

VIDCON3	Bit	Description	Initial State
Reserved	[31:21]	Reserved (should be 0)	0
Reserved	[20:19]	Reserved	0
CG_ON	[18]	Enables Control Color Gain. 0 = Disables (bypass) 1 = Enables	0
Reserved	[17]	Reserved	0
GM_ON	[16]	Enables Control Gamma. 0 = Disables (bypass) 1 = Enables	0
Reserved	[15]	Reserved	0
HUE_CSC_F_Narrow	[14]	Controls HUE CSC_F Narrow/ Wide. 0 = Wide 1 = Narrow	0
HUE_CSC_F_EQ709	[13]	Controls HUE_CSC_F parameter. 0 = Eq. 601 1 = Eq.709	0
HUE_CSC_F_ON	[12]	Enables HUE_CSC_F. 0 = Disables 1 = Enables (when HUE_ON == 1'b1)	0
Reserved	[11]	Reserved	0
HUE_CSC_B_Narrow	[10]	Controls HUE CSC_B Narrow/ Wide. 0 = Wide 1 = Narrow	0
HUE_CSC_B_EQ709	[9]	Controls HUE_CSC_B parameter. 0 = Eq. 601 1 = Eq.709	0
HUE_CSC_B_ON	[8]	Enables HUE_CSC_B. 0 = Disables 1 = Enables (when HUE_ON == 1'b1)	0
HUE_ON	[7]	Enables Control Hue. 0 = Disables (bypass) 1 = Enables	0
Resvrd	[6:2]	Reserved (should be 0)	0
PC_DIR	[1]	Controls Pixel Compensation direction. 0 = + 0.5(pos.) 1 = - 0.5(neg.)	0
PC_ON	[3:0]	Enables Pixel Compensation. 0 = Disable 1 = Enable Note: TV output data is compensated by PC_ON == 1'b1.	0x0

1.5.2.5 Video Time Control 0 Register (VIDTCON0, R/W, Address = 0xF800_0010)

VIDTCON0	Bit	Description	Initial State
VBPDE	[31:24]	Vertical back porch specifies the number of inactive lines at the start of a frame after vertical synchronization period. (Only for even field of YVU interface)	0x00
VBDP	[23:16]	Vertical back porch specifies the number of inactive lines at the start of a frame after vertical synchronization period.	0x00
VFPD	[15:8]	Vertical front porch specifies the number of inactive lines at the end of a frame before vertical synchronization period.	0x00
VSPW	[7:0]	Vertical sync pulse width determines the high-level width of VSYNC pulse by counting the number of inactive lines.	0x00

1.5.2.6 Video Time Control 1 Register (VIDTCON1, R/W, Address = 0xF800_0014)

VIDTCON1	Bit	Description	Initial State
VFPDE	[31:24]	Vertical front porch specifies the number of inactive lines at the end of a frame before vertical synchronization period. (Only for the even field of YVU interface).	0
HBPD	[23:16]	Horizontal back porch specifies the number of VCLK periods between the falling edge of HSYNC and start of active data.	0x00
HFPD	[15:8]	Horizontal front porch specifies the number of VCLK periods between the end of active data and rising edge of HSYNC.	0x00
HSPW	[7:0]	Horizontal sync pulse width determines the high-level width of HSYNC pulse by counting the number of VCLK.	0x00

1.5.2.7 Video Time Control 2 Register (VIDTCON2, R/W, Address = 0xF800_0018)

VIDTCON2	Bit	Description	Initial State
LINEVAL	[21:11]	Determines the vertical size of display. In the Interlace mode, (LINEVAL + 1) should be even.	0
HOZVAL	[10:0]	Determines the horizontal size of display.	0

NOTE: HOZVAL = (Horizontal display size) -1 and LINEVAL = (Vertical display size) -1.

1.5.2.8 Video Time Control3 Register (VIDTCON3, R/W, Address = 0xF800_001C)

VIDTCON3	Bit	Description	Initial State
VSYNCEN	[31]	Enables VSYNC Signal Output. 0 = Disables 1 = Enables VBPD(VFPD, VSPW) + 1 < LINEVAL (when VSYNCEN =1)	0
Reserved	[30]	Reserved (should be 0).	0
FRMEN	[29]	Enables the FRM signal output. 0 = Disables 1 = Enables	0
INVFRM	[28]	Controls the polarity of FRM pulse. 0 = Active HIGH 1 = Active LOW	0
FRMV RATE	[27:24]	Controls the FRM issue rate (Maximum rate up to 1:16)	0x00
Reserved	[23:16]	Reserved	0x00
FRMV FPD	[15:8]	Specifies the number of line between data active and FRM signal.	0x00
FRMV SPW	[7:0]	Specifies the number of line of FRM signal width. (FRMV FPD + 1) + (FRMV SPW + 1) < LINEVAL + 1 (in RGB)	0x00

1.5.2.9 Window 0 Control Register (WINCON0, R/W, Address = 0xF800_0020)

WINCON0	Bit	Description	Initial State
BUFSTATUS_H	[31]	<p>Specifies the Buffer Status (read only).</p> <p>Note: BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L}</p> <p>00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2</p>	0
BUFSEL_H	[30]	<p>Selects the Buffer set.</p> <p>Note: BUFSEL = {BUFSEL_H, BUFSEL_L}</p> <p>00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2 (only available where BUF_MODE == 1'b1)</p>	0
LIMIT_ON	[29]	<p>Enables CSC source limiter (for clamping xvYCC source).</p> <p>0 = Disables 1 = Enables (when local SRC data has xvYCC color space, InRGB=1)</p>	0
EQ709	[28]	<p>Controls CSC parameter.</p> <p>0 = Eq. 601 1 = Eq. 709 (when local SRC data has HD (709) color gamut)</p>	0
nWide/Narrow	[27:26]	<p>Chooses the color space conversion equation from YCbCr to RGB according to input value range (2'00 for YCbCr Wide range and 2'11 for YCbCr Narrow range)</p> <ul style="list-style-type: none"> - Wide Range: Y/ Cb/ Cr: 255-0 - Narrow Range: Y: 235-16, Cb/ Cr: 240-16 	00
TRGSTATUS	[25]	<p>Specifies the Trigger Status (read only).</p> <p>0 = No trigger is issued 1 = Trigger is issued</p>	0
Reserved	[24:23]	Reserved	00
ENLOCAL_F	[22]	<p>Selects the Data access method.</p> <p>0 = Dedicated DMA 1 = Local Path</p>	0
BUFSTATUS_L	[21]	<p>Specifies the Buffer Status (read only).</p> <p>Note: BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L}</p>	0
BUFSEL_L	[20]	<p>Selects the Buffer set.</p> <p>Note: BUFSEL = {BUFSEL_H, BUFSEL_L}</p>	0
BUFAUTOEN	[19]	<p>Specifies the Double Buffer Auto control bit.</p> <p>0 = Fixed by BUFSEL 1 = Auto Changed by Trigger Input</p>	0
BITSWP_F	[18]	<p>Specifies the Bit swap control bit.</p> <p>0 = Swap Disable 1 = Swap Enable</p> <p>Note: It should be 0 when ENLOCAL is 1.</p>	0
BYTSWP_F	[17]	<p>Specifies the Byte swaps control bit.</p> <p>0 = Swap Disable</p>	0

WINCON0	Bit	Description	Initial State
		1 = Swap Enable Note: It should be 0 when ENLOCAL is 1.	
HAWSWP_F	[16]	Specifies the Half-Word swap control bit. 0 = Swap Disable 1 = Swap Enable Note: It should be 0 when ENLOCAL is 1.	0
WSWP_F	[15]	Specifies the Word swap control bit. 0 = Swap Disable 1 = Swap Enable Note: It should be 0 when ENLOCAL is 1.	0
BUF_MODE	[14]	Selects the auto-buffering mode. 0 = Double 1 = Triple	0
InRGB	[13]	Specifies the input color space of source image (only for 'ENLOCAL' enable). 0 = RGB 1 = YCbCr	0
-	[12:11]	Reserved (Should be 0)	0
BURSTLEN	[10:9]	Selects the DMA Burst Maximum Length. 00 = 16 word-burst 01 = 8 word-burst 10 = 4 word-burst	0
Reserved	[8:7]	-	0
BLD_PIX_F	[6]	Selects the blending category (In case of window0, this is only required for deciding window 0's blending factor.) 0 = Per plane blending 1 = Per pixel blending	0
BPPMODE_F	[5:2]	Selects the Bits Per Pixel (BPP) mode for Window image. 0000 = 1 bpp 0001 = 2 bpp 0010 = 4 bpp 0011 = 8 bpp (palletized) 0100 = 8 bpp (non-palletized, A: 1-R:2-G:3-B:2) 0101 = 16 bpp (non-palletized, R:5-G:6-B:5) 0110 = 16 bpp (non-palletized, A:1-R:5-G:5-B:5) 0111 = 16 bpp (non-palletized, I :1-R:5-G:5-B:5) 1000 = Unpacked 18 bpp (non-palletized, R:6-G:6-B:6) 1001 = Unpacked 18 bpp (non-palletized, A:1-R:6-G:6-B:5) 1010 = Unpacked 19 bpp (non-palletized, A:1-R:6-G:6-B:6) 1011 = Unpacked 24 bpp (non-palletized R:8-G:8-B:8) 1100 = Unpacked 24 bpp (non-palletized A:1-R:8-G:8-B:7) *1101 = Unpacked 25 bpp (non-palletized A:1-R:8-G:8-B:8) *1110 = Unpacked 13 bpp (non-palletized A:1-R:4-G:4-B:4) 1111 = Unpacked 15 bpp (non-palletized R:5-G:5-B:5)	0

WINCON0	Bit	Description	Initial State
		<p>Note: *1101 = Supports unpacked 32 bpp (non-palletized A:8-R:8-G:8-B:8) for per pixel blending.</p> <p>*1110 = Supports 16 bpp (non-palletized A:4-R:4-G:4-B:4) for per pixel blending. (16 level blending)</p>	
ALPHA_SEL_F	[1]	<p>Selects the Alpha value.</p> <p>When per plane blending case (BLD_PIX ==0): 0 = Using ALPHA0_R/G/B values 1 = Using ALPHA1_R/G/B values</p> <p>When per pixel blending (BLD_PIX ==1) 0 = Selected by AEN (A value) 1 = Using DATA[31:24] data in word boundary (only when BPPMODE_F = 4'b1101) DATA[31:28], [15:12] data in word boundary (only when BPPMODE_F = 4'b1110)</p>	0
ENWIN_F	[0]	<p>Enables/ disables video output and logic immediately.</p> <p>0 = Disables the video output and video control signal. 1 = Enables the video output and video control signal.</p>	0

1.5.2.10 Window 1 Control Register (WINCON1, R/W, Address = 0xF800_0024)

WINCON1	Bit	Description	Initial State
BUFSTATUS_H	[31]	<p>Specifies the Buffer Status (read only).</p> <p>Note: BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L}</p> <p>00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2</p>	0
BUFSEL_H	[30]	<p>Select the Buffer set.</p> <p>Note: BUFSEL = {BUFSEL_H, BUFSEL_L}</p> <p>00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2 (only available when BUF_MODE == 1'b1)</p>	0
LIMIT_ON	[29]	<p>Enables Control CSC source limiter (for clamping xvYCC source).</p> <p>0 = Disables 1 = Enables (when local SRC data has xvYCC color space, InRGB=1)</p>	0
EQ709	[28]	<p>Controls the CSC parameter.</p> <p>0 = Eq.601 1 = Eq.709 (when local SRC data has HD(709) color gamut)</p>	0
nWide/Narrow	[27:26]	<p>Chooses the color space conversion equation from YCbCr to RGB based on input value range (2'00 for YCbCr Wide range and 2'11 for YCbCr Narrow range).</p> <ul style="list-style-type: none"> - Wide Range: Y/ Cb/ Cr: 255-0 - Narrow Range: Y: 235-16, Cb/ Cr: 240-16 	00
TRGSTATUS	[25]	<p>Specifies Window 0 Software Trigger Update Status (read only).</p> <p>0 = Update 1 = Not Update</p> <p>If the Software Trigger in window 1 occurs, this bit is automatically set to '1'. This value is cleared only after updating the shadow register sets.</p>	0
Reserved	[24:23]	Reserved. (should be 0)	0
ENLOCAL_F	[22]	<p>Selects the Data access method.</p> <p>0 = Dedicated DMA 1 = Local Path</p>	0
BUFSTATUS_L	[21]	<p>Specifies the Buffer Status (read only).</p> <p>Note: BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L}</p>	0
BUFSEL_L	[20]	<p>Selects the Buffer set.</p> <p>Note: BUFSEL = {BUFSEL_H, BUFSEL_L}</p>	0
BUFAUTOEN	[19]	<p>Specifies the Double Buffer Auto control bit.</p> <p>0 = Fixed by BUFSEL 1 = Auto changed by Trigger Input</p>	0
BITSWP_F	[18]	<p>Specifies the Bit swap control bit.</p> <p>0 = Swap Disable 1 = Swap Enable</p> <p>Note: It should be 0 when ENLOCAL is 1.</p>	0

WINCON1	Bit	Description	Initial State
BYTSPW_F	[17]	Specifies the Byte swaps control bit. 0 = Swap Disable 1 = Swap Enable Note: It should be 0 when ENLOCAL is 1.	0
HAWSPW_F	[16]	Specifies the Half-Word swap control bit. 0 = Swap Disable 1 = Swap Enable Note: It should be 0 when ENLOCAL is 1.	0
WSWP_F	[15]	Specifies the Word swap control bit. 0 = Swap Disable 1 = Swap Enable Note: It should be 0 when ENLOCAL is 1.	0
BUF_MODE	[14]	Selects the auto-buffering mode. 0 = Double 1 = Triple	0
InRGB	[13]	Indicates the input color space of source image (only for 'EnLcal' enable). 0 = RGB 1 = YCbCr	0
-	[12:11]	Reserved (should be 0)	0
BURSTLEN	[10:9]	Specifies the DMA's Burst Maximum Length selection. 00 = 16 word- burst 01 = 8 word- burst 10 = 4 word- burst	0
-	[8]	Reserved (should be 0)	0
ALPHA_MUL_F	[7]	Specifies the Multiplied Alpha value mode. 0 = Disables 1 = Enables multiplied mode When ALPHA_MUL is 1, set BLD_PIX =1, ALPHA_SEL=1, and BPPMODE_F[5:2] = 4'b1101 or 4'b1110. Note: Alpha value = alpha_pixel (from data) * ALPHA0_R/G/B	0
BLD_PIX_F	[6]	Selects the blending category. 0 = Per plane blending 1 = Per pixel blending	0
BPPMODE_F	[5:2]	Selects the Bits Per Pixel (BPP) mode in Window image. 0000 = 1 bpp 0001 = 2 bpp 0010 = 4 bpp 0011 = 8 bpp (palletized) 0100 = 8 bpp (non-palletized, A: 1-R:2-G:3-B:2) 0101 = 16 bpp (non-palletized, R:5-G:6-B:5) 0110 = 16 bpp (non-palletized, A:1-R:5-G:5-B:5) 0111 = 16 bpp (non-palletized, I :1-R:5-G:5-B:5) 1000 = Unpacked 18 bpp (non-palletized, R:6-G:6-B:6) 1001 = Unpacked 18 bpp (non-palletized, A:1-R:6-G:6-B:5)	0

WINCON1	Bit	Description	Initial State
		<p>1010 = Unpacked 19 bpp (non-palletized, A:1-R:6-G:6-B:6) 1011 = Unpacked 24 bpp (non-palletized R:8-G:8-B:8) 1100 = Unpacked 24 bpp (non-palletized A:1-R:8-G:8-B:7) *1101 = Unpacked 25 bpp (non-palletized A:1-R:8-G:8-B:8) *1110 = Unpacked 13 bpp (non-palletized A:1-R:4-G:4-B:4) 1111 = Unpacked 15 bpp (non-palletized R:5-G:5-B:5)</p> <p>Note: *1101 = supports unpacked 32 bpp (non-palletized A:8-R:8-G:8-B:8) for per pixel blending. *1110 = supports 16 bpp (non-palletized A:4-R:4-G:4-B:4) for per pixel blending. (16 level blending)</p>	
ALPHA_SEL_F	[1]	<p>Selects the Alpha value.</p> <p>When Per plane blending case (BLD_PIX ==0) 0 = Using ALPHA0_R/G/B values 1 = Using ALPHA1_R/G/B values</p> <p>When Per pixel blending (BLD_PIX ==1) 0 = Selected by AEN (A value) 1 = Using DATA[31:24] data in word boundary (only when BPPMODE_F = 4'b1101) DATA[31:28], [15:12] data in word boundary (only when BPPMODE_F = 4'b1110)</p>	0
ENWIN_F	[0]	<p>Enables/ disables video output and logic immediately.</p> <p>0 = Disables the video output and video control signal. 1 = Enables the video output and video control signal.</p>	0

1.5.2.11 Window 2 Control Register (WINCON2, R/W, Address = 0xF800_0028)

WINCON2	Bit	Description	Initial State
BUFSTATUS_H	[31]	<p>Specifies the Buffer Status (read only).</p> <p>Note. BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L}</p> <p>00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2</p>	0
BUFSEL_H	[30]	<p>Selects the Buffer set.</p> <p>Note. BUFSEL = {BUFSEL_H, BUFSEL_L}</p> <p>00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2 (only available when BUF_MODE == 1'b1)</p>	0
LIMIT_ON	[29]	<p>Enables CSC source limiter (for clamping xvYCC source).</p> <p>0 = Disables 1 = Enables (when local SRC data has xvYCC color space, InRGB=1)</p>	0
EQ709	[28]	<p>Controls CSC parameter.</p> <p>0 = Eq.601 1 = Eq.709 (when local SRC data has HD (709) color gamut)</p>	0
nWide/Narrow	[27:26]	<p>Chooses color space conversion equation from YCbCr to RGB based on the input value range (2'00 for YCbCr Wide range and 2'11 for YCbCr Narrow range).</p> <ul style="list-style-type: none"> - Wide Range: Y/ Cb/ Cr: 255-0 - Narrow Range: Y: 235-16, Cb/ Cr: 240-16 	00
Reserved	[25:24]	Reserved	00
Reserved	[23]	Should be '0'.	0
ENLOCAL_F	[22]	<p>Selects the Data access method.</p> <p>0 = Dedicated DMA 1 = Local Path</p>	0
BUFSTATUS_L	[21]	<p>Specifies the Buffer Status (read only).</p> <p>Note: BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L}</p>	
BUFSEL_L	[20]	<p>Selects the Buffer set.</p> <p>Note: BUFSEL = {BUFSEL_H, BUFSEL_L}</p>	0
BUFAUTOEN	[19]	<p>Specifies the Double Buffer Auto control bit.</p> <p>0 = Fixed by BUFSEL 1 = Auto changed by Trigger Input</p>	0
BITSWP_F	[18]	<p>Specifies the Bit swap control bit.</p> <p>0 = Swap Disable 1 = Swap Enable</p> <p>Note: It should be 0 when ENLOCAL is 1.</p>	0
BYTSPW_F	[17]	<p>Specifies the Byte swaps control bit.</p> <p>0 = Swap Disable 1 = Swap Enable</p> <p>Note: It should be 0 when ENLOCAL is 1.</p>	0

WINCON2	Bit	Description	Initial State
HAWSWP_F	[16]	Specifies the Half-Word swap control bit. 0 = Swap Disable 1 = Swap Enable Note: It should be 0 when ENLOCAL is 1.	0
WSWP_F	[15]	Specifies the Word swap control bit. 0 = Swap Disable 1 = Swap Enable Note: It should be 0 when ENLOCAL is 1.	0
BUF_MODE	[14]	Selects the auto-buffering mode. 0 = Double 1 = Triple	0
InRGB	[13]	Specifies the input color space of source image (only for 'EnLcal' enable). 0 = RGB 1 = YCbCr	0
-	[12:11]	Reserved (should be 0).	0
BURSTLEN	[10:9]	Selects the DMA's Burst Maximum Length. 00 = 16 word- burst 01 = 8 word- burst 10 = 4 word- burst	0
-	[8]	Reserved (should be 0).	0
ALPHA_MUL_F	[7]	Specifies the Multiplied Alpha value mode. 0 = Disables 1 = Enables multiplied mode When ALPHA_MUL is 1, set BLD_PIX =1, ALPHA_SEL=1, and BPPMODE_F[5:2] = 4'b1101 or 4'b1110. Note: Alpha value = alpha_pixel (from data) * ALPHA0_R/G/B	0
BLD_PIX_F	[6]	Selects the blending category. 0 = Per plane blending 1 = Per pixel blending	0
BPPMODE_F	[5:2]	Selects the Bits Per Pixel (BPP) mode in Window image. 0000 = 1 bpp 0001 = 2 bpp 0010 = 4 bpp 0011 = 8 bpp (palletized) 0100 = 8 bpp (non-palletized, A: 1-R:2-G:3-B:2) 0101 = 16 bpp (non-palletized, R:5-G:6-B:5) 0110 = 16 bpp (non-palletized, A:1-R:5-G:5-B:5) 0111 = 16 bpp (non-palletized, I :1-R:5-G:5-B:5) 1000 = Unpacked 18 bpp (non-palletized, R:6-G:6-B:6) 1001 = Unpacked 18 bpp (non-palletized, A:1-R:6-G:6-B:5) 1010 = Unpacked 19 bpp (non-palletized, A:1-R:6-G:6-B:6) 1011 = Unpacked 24 bpp (non-palletized R:8-G:8-B:8) 1100 = Unpacked 24 bpp (non-palletized A:1-R:8-G:8-B:7) *1101 = Unpacked 25 bpp (non-palletized A:1-R:8-G:8-B:8) *1110 = Unpacked 13 bpp (non-palletized A:1-R:4-G:4-B:4)	0

WINCON2	Bit	Description	Initial State
		<p>1111 = Unpacked 15 bpp (non-palletized R:5-G:5-B:5) Note: *1101 = Supports unpacked 32 bpp (non-palletized A:8-R:8-G:8-B:8) for per pixel blending. *1110 = Supports 16 bpp (non-palletized A:4-R:4-G:4-B:4) for per pixel blending. (16 level blending)</p>	
ALPHA_SEL_F	[1]	<p>Selects the Alpha value. When Per plane blending case BLD_PIX ==0: 0 = Using ALPHA0_R/G/B values 1 = Using ALPHA1_R/G/B values When Per pixel blending BLD_PIX ==1: 0 = Selected by AEN (A value) 1 = Using DATA[31:24] data in word boundary (only when BPPMODE_F = 4'b1101) DATA[31:28], [15:12] data in word boundary (only when BPPMODE_F = 4'b1110)</p>	0
ENWIN_F	[0]	<p>Enables/ disables the video output and logic immediately. 0 = Disables the video output and video control signal. 1 = Enables the video output and video control signal.</p>	0

1.5.2.12 Window 3 Control Register (WINCON3, R/W, Address = 0xF800_002C)

WINCON3	Bit	Description	Initial State
BUFSTATUS_H	[31]	<p>Specifies the Buffer Status (read only).</p> <p>Note: BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L}</p> <p>00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2</p>	
BUFSEL_H	[30]	<p>Selects the Buffer set</p> <p>Note: BUFSEL = {BUFSEL_H, BUFSEL_L}</p> <p>00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2 (only available where BUF_MODE == 1'b1)</p>	
-	[29:26]	Reserved (should be 0).	
TRIGSTATUS	[25]	<p>Specifies the Trigger Status (read only)</p> <p>0 = No trigger is issued 1 = Trigger is issued</p>	
-	[24:22]	Reserved (should be 0).	
BUFSTATUS_L	[21]	<p>Specifies the Buffer Status (read only).</p> <p>Note: BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L}</p>	
BUFSEL_L	[20]	<p>Selects the Buffer set.</p> <p>Note: BUFSEL = {BUFSEL_H, BUFSEL_L}</p>	
BUFAUTOEN	[19]	<p>Specifies the Double Buffer Auto control bit.</p> <p>0 = Fixed by BUFSEL 1 = Auto changed by Trigger Input</p>	
BITSWP_F	[18]	<p>Specifies the Bit swap control bit.</p> <p>0 = Swap Disable 1 = Swap Enable</p>	0
BYTSWP_F	[17]	<p>Specifies the Byte swaps control bit.</p> <p>0 = Swap Disable 1 = Swap Enable</p>	0
HAWSWP_F	[16]	<p>Specifies the Half-Word swap control bit.</p> <p>0 = Swap Disable 1 = Swap Enable</p>	0
WSWP_F	[15]	<p>Specifies the Word swap control bit.</p> <p>0 = Swap Disable 1 = Swap Enable</p>	
BUF_MODE	[14]	<p>Selects the auto-buffering mode.</p> <p>0 = Double 1 = Triple</p>	0
-	[13:11]	Reserved (should be 0).	0
BURSTLEN	[10:9]	<p>Selects the DMA Burst Maximum Length.</p> <p>00 = 16 word- burst 01 = 8 word- burst 10 = 4 word- burst</p>	0

WINCON3	Bit	Description	Initial State
-	[8]	Reserved (should be 0).	0
ALPHA_MUL_F	[7]	<p>Specifies the Multiplied Alpha value mode.</p> <p>0 = Disables 1 = Enables multiplied mode</p> <p>When ALPHA_MUL is 1, set BLD_PIX =1, ALPHA_SEL=1, and BPPMODE_F[5:2] = 4'b1101 or 4'b1110.</p> <p>Note. Alpha value = alpha_pixel (from data) * ALPHA0_R/G/B</p>	0
BLD_PIX_F	[6]	<p>Selects the blending category.</p> <p>0 = Per plane blending 1 = Per pixel blending</p>	
BPPMODE_F	[5:2]	<p>Selects the Bits Per Pixel (BPP) mode in Window image.</p> <p>0000 = 1 bpp 0001 = 2 bpp 0010 = 4 bpp 0011 = 8 bpp (palletized) 0100 = 8 bpp (non-palletized, A: 1-R:2-G:3-B:2) 0101 = 16 bpp (non-palletized, R:5-G:6-B:5) 0110 = 16 bpp (non-palletized, A:1-R:5-G:5-B:5) 0111 = 16 bpp (non-palletized, I:1-R:5-G:5-B:5) 1000 = Unpacked 18 bpp (non-palletized, R:6-G:6-B:6) 1001 = Unpacked 18 bpp (non-palletized, A:1-R:6-G:6-B:5) 1010 = Unpacked 19 bpp (non-palletized, A:1-R:6-G:6-B:6) 1011 = Unpacked 24 bpp (non-palletized R:8-G:8-B:8) 1100 = Unpacked 24 bpp (non-palletized A:1-R:8-G:8-B:7) *1101 = Unpacked 25 bpp (non-palletized A:1-R:8-G:8-B:8) *1110 = Unpacked 13 bpp (non-palletized A:1-R:4-G:4-B:4) 1111 = Unpacked 15 bpp (non-palletized R:5-G:5-B:5)</p> <p>Note: *1101 = Supports unpacked 32 bpp (non-palletized A:8-R:8-G:8-B:8) for per pixel blending. *1110 = Supports 16 bpp (non-palletized A:4-R:4-G:4-B:4) for per pixel blending. (16 level blending)</p>	0
ALPHA_SEL_F	[1]	<p>Selects the Alpha value.</p> <p>When Per plane blending case BLD_PIX ==0: 0 = Using ALPHA0_R/G/B values 1 = Using ALPHA1_R/G/B values</p> <p>When Per pixel blending BLD_PIX ==1: 0 = Selected by AEN (A value) 1 = Using DATA[31:24] data in word boundary (only when BPPMODE_F = 4'b1101) DATA[31:28], [15:12] data in word boundary (only when BPPMODE_F = 4'b1110)</p>	0
ENWIN_F	[0]	Enables/ disables video output and logic immediately.	0
		0 = Disables the video output and video control signal. 1 = Enables the video output and video control signal.	

1.5.2.13 Window 4 Control Register (WINCON4, R/W, Address = 0xF800_0030)

WINCON4	Bit	Description	Initial State
BUFSTATUS_H	[31]	<p>Specifies the Buffer Status (read only).</p> <p>Note: BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L}</p> <p>00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2</p>	0
BUFSEL_H	[30]	<p>Selects the Buffer set.</p> <p>Note: BUFSEL = {BUFSEL_H, BUFSEL_L}</p> <p>00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2 (only available where BUF_MODE == 1'b1)</p>	0
-	[29:26]	Reserved (should be 0).	0
TRIGSTATUS	[25]	<p>Specifies the Trigger Status (read only).</p> <p>0 = No trigger is issued 1 = Trigger is issued</p>	0
-	[24:22]	Reserved (should be 00).	0
BUFSTATUS_L	[21]	<p>Specifies the Buffer Status (read only).</p> <p>Note: BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L}</p>	0
BUFSEL_L	[20]	<p>Selects the Buffer set.</p> <p>Note: BUFSEL = {BUFSEL_H, BUFSEL_L}</p>	0
BUFAUTOEN	[19]	<p>Specifies the Double Buffer Auto control bit.</p> <p>0 = Fixed by BUFSEL 1 = Auto changed by Trigger Input</p>	0
BITSWP_F	[18]	<p>Specifies the Bit swap control bit.</p> <p>0 = Swap Disable 1 = Swap Enable</p>	0
BYTSWP_F	[17]	<p>Specifies the Byte swap control bit.</p> <p>0 = Swap Disable 1 = Swap Enable</p>	0
HAWSWP_F	[16]	<p>Specifies the Half-Word swap control bit.</p> <p>0 = Swap Disable 1 = Swap Enable</p>	0
WSWP_F	[15]	<p>Specifies the Word swap control bit.</p> <p>0 = Swap Disable 1 = Swap Enable</p>	0
BUF_MODE	[14]	<p>Selects the auto-buffering mode.</p> <p>0 = Double 1 = Triple</p>	0
-	[13:11]	Reserved (should be 0).	0
BURSTLEN	[10:9]	<p>Selects the DMA Burst Maximum Length.</p> <p>00 = 16 word- burst 01 = 8 word- burst 10 = 4 word- burst</p>	0

WINCON4	Bit	Description	Initial State
-	[8]	Reserved (should be 0)	0
ALPHA_MUL_F	[7]	<p>Specifies the Multiplied Alpha value mode.</p> <p>0 = Disables 1 = Enables multiplied mode</p> <p>When ALPHA_MUL is 1, set BLD_PIX =1, ALPHA_SEL=1, and BPPMODE_F[5:2] = 4'b1101 or 4'b1110.</p> <p>Note: Alpha value = alpha_pixel (from data) * ALPHA0_R/G/B</p>	0
BLD_PIX_F	[6]	<p>Selects the blending category.</p> <p>0 = Per plane blending 1 = Per pixel blending</p>	0
BPPMODE_F	[5:2]	<p>Selects the Bits Per Pixel (BPP) mode in Window image.</p> <p>0000 = 1 bpp 0001 = 2 bpp 0010 = 4 bpp 0011 = 8 bpp (palletized) 0100 = 8 bpp (non-palletized, A: 1-R:2-G:3-B:2) 0101 = 16 bpp (non-palletized, R:5-G:6-B:5) 0110 = 16 bpp (non-palletized, A:1-R:5-G:5-B:5) 0111 = 16 bpp (non-palletized, I:1-R:5-G:5-B:5) 1000 = Unpacked 18 bpp (non-palletized, R:6-G:6-B:6) 1001 = Unpacked 18 bpp (non-palletized, A:1-R:6-G:6-B:5) 1010 = Unpacked 19 bpp (non-palletized, A:1-R:6-G:6-B:6) 1011 = Unpacked 24 bpp (non-palletized, R:8-G:8-B:8) 1100 = Unpacked 24 bpp (non-palletized, A:1-R:8-G:8-B:7) *1101 = Unpacked 25 bpp (non-palletized, A:1-R:8-G:8-B:8) *1110 = Unpacked 13 bpp (non-palletized, A:1-R:4-G:4-B:4) 1111 = Unpacked 15 bpp (non-palletized, R:5-G:5-B:5)</p> <p>Note: *1101 = Support unpacked 32 bpp (non-palletized, A:8-R:8-G:8-B:8) for per pixel blending.</p> <p>*1110 = Support 16 bpp (non-palletized A:4-R:4-G:4-B:4) for per pixel blending. (16 level blending)</p>	0
ALPHA_SEL_F	[1]	<p>Selects the Alpha value.</p> <p>When Per plane blending case BLD_PIX ==0: 0 = Using ALPHA0_R/G/B values 1 = Using ALPHA1_R/G/B values</p> <p>When Per pixel blending BLD_PIX ==1: 0 = Selected by AEN (A value) 1 = Using DATA[31:24] data in word boundary (only when BPPMODE_F = 4'b1101) DATA[31:28], [15:12] data in word boundary (only when BPPMODE_F = 4'b1110)</p>	0
ENWIN_F	[0]	Enables/ disables video output and logic immediately.	0
		0 = Disables the video output and video control signal. 1 = Enables the video output and video control signal.	

1.5.2.14 Window Shadow Control Register (VIDOSD0C, R/W, Address = 0xF800_0034)

SHODOWCON	Bit	Description	Initial State
-	[31:15]	Reserved (should be 0).	0
W4_SHADOW _PROTECT	[14]	Protects to update window 4's shadow register (xxx_F). 0 = Updates shadow register per frame 1 = Protects to update (updates shadow register at next frame after 'SHADOW_PROTECT' turns to be 1'b0)	0
W3_SHADOW _PROTECT	[13]	Protects to update window 3's shadow register (xxx_F) 0 = Updates shadow register per frame 1 = Protects to update (updates shadow register at next frame after 'SHADOW_PROTECT' turns to be 1'b0)	0
W2_SHADOW _PROTECT	[12]	Protects to update window 2's shadow register (xxx_F) 0 = Updates shadow register per frame 1 = Protects to update (updates shadow register at next frame after 'SHADOW_PROTECT' turns to be 1'b0)	0
W1_SHADOW _PROTECT	[11]	Protects to update window 1's shadow register (xxx_F) 0 = Updates shadow register per frame 1 = Protects to update (update shadow register at next frame after 'SHADOW_PROTECT' turns to be 1'b0)	0
W0_SHADOW _PROTECT	[10]	Protects to update window 0's shadow register (xxx_F) 0 = Updates shadow register per frame 1 = Protects to update (update shadow register at next frame after 'SHADOW_PROTECT' turns to be 1'b0)	0
-	[9:8]	Reserved	0
C2_ENLOCAL_F	7	Enables Channel 2 Local Path. 0 = Disables 1 = Enables	0
C1_ENLOCAL_F	6	Enables Channel 1 Local Path. 0 = Disables 1 = Enables	0
C0_ENLOCAL_F	5	Enables Channel 0 Local Path. 0 = Disables 1 = Enables	0
C4_EN_F	4	Enables Channel 4. 0 = Disables 1 = Enables	0
C3_EN_F	3	Enables Channel 3. 0 = Disables 1 = Enables	0
C2_EN_F	2	Enables Channel 2. 0 = Disables 1 = Enables	0
C1_EN_F	1	Enables Channel 1. 0 = Disables 1 = Enables	0
C0_EN_F	0	Enables Channel 0. 0 = Disables 1 = Enables	0

1.5.2.15 Channel Mapping Control Register2 (WINCHMAP2, R/W, Address = 0xEE00_003C)

WINCHMAP2	Bit	Description	Initial State
CH4FISEL	[30:28]	Selects Channel 4's channel. 001 = Window 0 010 = Window 1 101 = Window 2 110 = Window 3 111 = Window 4	111
CH3FISEL	[27:25]	Selects Channel 3's channel. 001 = Window 0 010 = Window 1 101 = Window 2 110 = Window 3 111 = Window 4	110
CH2FISEL	[24:22]	Selects Channel 2's channel. 001 = Window 0 010 = Window 1 101 = Window 2 110 = Window 3 111 = Window 4	101
CH1FISEL	[21:19]	Selects Channel 1's channel. 001 = Window 0 010 = Window 1 101 = Window 2 110 = Window 3 111 = Window 4	010
CH0FISEL	[18:16]	Selects Channel 0's channel. 001 = Window 0 010 = Window 1 101 = Window 2 110 = Window 3 111 = Window 4	001
W4FISEL	[14:12]	Selects Window 4's channel. 001 = Channel 0 010 = Channel 1 101 = Channel 2 110 = Channel 3 111 = Channel 4	111
W3FISEL	[11:9]	Selects Window 3's channel. 001 = Channel 0 010 = Channel 1 101 = Channel 2 110 = Channel 3 111 = Channel 4	110
W2FISEL	[8:6]	Selects Window 2's channel. 001 = Channel 0 010 = Channel 1 101 = Channel 2	101

WINCHMAP2	Bit	Description	Initial State
		110 = Channel 3 111 = Channel 4	
W1FISEL	[5:3]	Selects Window 1's channel. 001 = Channel 0 010 = Channel 1 101 = Channel 2 110 = Channel 3 111 = Channel 4	010
W0FISEL	[2:0]	Selects Window 0's channel. 001 = Channel 0 010 = Channel 1 101 = Channel 2 110 = Channel 3 111 = Channel 4	001

1.5.2.16 Window 0 Position Control A Register (VIDOSD0A, R/W, Address = 0xF800_0040)

VIDOSD0A	Bit	Description	Initial State
OSD_LeftTopX_F	[21:11]	Specifies the horizontal screen coordinate for left top pixel of OSD image.	0
OSD_LeftTopY_F	[10:0]	Specifies the vertical screen coordinate for left top pixel of OSD image (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate must be even.)	0

1.5.2.17 Window 0 Position Control B Register (VIDOSD0B, R/W, Address = 0xF800_0044)

VIDOSD0B	Bit	Description	Initial State
OSD_RightBotX_F	[21:11]	Specifies the horizontal screen coordinate for right bottom pixel of OSD image.	0
OSD_RightBotY_F	[10:0]	Specifies the vertical screen coordinate for right bottom pixel of OSD image. (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate must be odd value.)	0

NOTE: Registers must have word boundary X position. Therefore, 24 BPP mode must have X position by 1 pixel. (For example, X = 0, 1, 2, 3....)

16 BPP mode must have X position by 2 pixel. (For example, X = 0, 2, 4, 6....)

8 BPP mode must have X position by 4 pixel. (For example, X = 0, 4, 8, 12....)

1.5.2.18 Window 0 Position Control C Register (VIDOSD0C, R/W, Address = 0xF800_0048)

VIDOSD0C	Bit	Description	Initial State
Reserved	[25:24]	Reserved (should be 0)	0
OSDSIZE	[23:0]	Specifies the Window Size For example, Height * Width (Number of Word)	0

1.5.2.19 Window 1 Position Control A Register (VIDOSD0C, R/W, Address = 0xF800_0050)

VIDOSD1A	Bit	Description	Initial State
OSD_LeftTopX_F	[21:11]	Specifies the Horizontal screen coordinate for left top pixel of OSD image.	0
OSD_LeftTopY_F	[10:0]	Specifies the Vertical screen coordinate for left top pixel of OSD image. (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate must be even.)	0

1.5.2.20 Window 1 Position Control B Register (VIDOSD1B, R/W, Address = 0xF800_0054)

VIDOSD1B	Bit	Description	Initial State
OSD_RightBotX_F	[21:11]	Specifies the horizontal screen coordinate for right bottom pixel of OSD image.	0
OSD_RightBotY_F	[10:0]	Specifies the vertical screen coordinate for right bottom pixel of OSD image. (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate must be odd value.)	0

NOTE: Registers must have word boundary X position.

Therefore, 24 BPP mode must have X position by 1 pixel. (For example, X = 0, 1, 2, 3....)

16 BPP mode must have X position by 2 pixel. (For example, X = 0, 2, 4, 6....)

8 BPP mode must have X position by 4 pixel. (For example, X = 0, 4, 8, 12....)

1.5.2.21 Window 1 Position Control C Register (VIDOSD1C, R/W, Address = 0xF800_0058)

VIDOSD1C	Bit	Description	Initial State
Reserved	[24]	Reserved	0
ALPHA0_R_H_F	[23:20]	Specifies Red Alpha upper value (case AEN == 0)	0
ALPHA0_G_H_F	[19:16]	Specifies Green Alpha upper value (case AEN == 0)	0
ALPHA0_B_H_F	[15:12]	Specifies Blue Alpha upper value (case AEN == 0)	0
ALPHA1_R_H_F	[11:8]	Specifies Red Alpha upper value (case AEN == 1)	0
ALPHA1_G_H_F	[7:4]	Specifies Green Alpha upper value (case AEN == 1)	0
ALPHA1_B_H_F	[3:0]	Specifies Blue Alpha upper value (case AEN == 1)	0

NOTE: Refer to VIDW1ALPHA0,1 register

1.5.2.22 Window 1 Position Control D Register (VIDOSD1D, R/W, Address = 0xF800_005C)

VIDOSD1D	Bit	Description	Initial State
Reserved	[25:24]	Reserved (should be 0)	0
OSDSIZE	[23:0]	Specifies Window Size. For example, Height * Width(Number of Word)	0

1.5.2.23 Window 2 Position Control A Register (VIDOSD2A, R/W, Address = 0xF800_0060)

VIDOSD2A	Bit	Description	Initial State
OSD_LeftTopX_F	[21:11]	Specifies the horizontal screen coordinate for left top pixel of OSD image.	0
OSD_LeftTopY_F	[10:0]	Specifies the vertical screen coordinate for left top pixel of OSD image. (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate must be even value.)	0

1.5.2.24 Window 2 Position Control B Register (VIDOSD2B, R/W, Address = 0xF800_0064)

VIDOSD2B	Bit	Description	Initial State
OSD_RightBotX_F	[21:11]	Specifies the horizontal screen coordinate for right bottom pixel of OSD image.	0
OSD_RightBotY_F	[10:0]	Specifies the vertical screen coordinate for right bottom pixel of OSD image. (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate must be odd value.)	0

NOTE: Registers must have word boundary X position.

Therefore, 24 BPP mode must have X position by 1 pixel. (For example, X = 0, 1, 2, 3....)

16 BPP mode must have X position by 2 pixel. (For example, X = 0, 2, 4, 6....)

8 BPP mode must have X position by 4 pixel. (For example, X = 0, 4, 8, 12....)

1.5.2.25 Window 2 Position Control C Register (VIDOSD2C, R/W, Address = 0xF800_0068)

VIDOSD2C	Bit	Description	Initial State
Reserved	[24]	Reserved	0
ALPHA0_R_H_F	[23:20]	Specifies the Red Alpha upper value (case AEN == 0).	0
ALPHA0_G_H_F	[19:16]	Specifies the Green Alpha upper value (case AEN == 0).	0
ALPHA0_B_H_F	[15:12]	Specifies the Blue Alpha upper value (case AEN == 0).	0
ALPHA1_R_H_F	[11:8]	Specifies the Red Alpha upper value (case AEN == 1).	0
ALPHA1_G_H_F	[7:4]	Specifies the Green Alpha upper value (case AEN == 1).	0
ALPHA1_B_H_F	[3:0]	Specifies the Blue Alpha upper value (case AEN == 1).	0

NOTE: For more information, refer to VIDW2ALPHA0,1 register.

1.5.2.26 Window 2 Position Control D Register (VIDOSD2D, R/W, Address = 0xF800_006C)

VIDOSD2D	Bit	Description	Initial State
Reserved	[25:24]	Reserved (should be 0)	0
OSDSIZE	[23:0]	Specifies the Window Size For example, Height * Width(Number of Word)	0

1.5.2.27 Window 3 Position Control A Register (VIDOSD3A, R/W, Address = 0xF800_0070)

VIDOSD3A	Bit	Description	Initial State
OSD_LeftTopX_F	[21:11]	Specifies the Horizontal screen coordinate for left top pixel of OSD image	0
OSD_LeftTopY_F	[10:0]	Specifies the Vertical screen coordinate for left top pixel of OSD image. (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate must be even value.)	0

1.5.2.28 Window 3 Position Control B Register (VIDOSD3B, R/W, Address = 0xF800_0074)

VIDOSD3B	Bit	Description	Initial State
OSD_RightBotX_F	[21:11]	Specifies the Horizontal screen coordinate for right bottom pixel of OSD image.	0
OSD_RightBotY_F	[10:0]	Specifies the Vertical screen coordinate for right bottom pixel of OSD image. (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate must be odd value.)	0

NOTE: Registers must have word boundary X position.

Therefore, 24 BPP mode must have X position by 1 pixel. (For example, X = 0, 1, 2, 3....)

16 BPP mode must have X position by 2 pixel. (For example, X = 0, 2, 4, 6....)

8 BPP mode must have X position by 4 pixel. (For example, X = 0, 4, 8, 12....)

1.5.2.29 Window 3 Position Control C Register (VIDOSD3C, R/W, Address = 0xF800_0078)

VIDOSD3C	Bit	Description	Initial State
Reserved	[24]	Reserved	0
ALPHA0_R_H_F	[23:20]	Specifies the Red Alpha upper value (case AEN == 0).	0
ALPHA0_G_H_F	[19:16]	Specifies the Green Alpha upper value (case AEN == 0).	0
ALPHA0_B_H_F	[15:12]	Specifies the Blue Alpha upper value (case AEN == 0).	0
ALPHA1_R_H_F	[11:8]	Specifies the Red Alpha upper value (case AEN == 1).	0
ALPHA1_G_H_F	[7:4]	Specifies the Green Alpha upper value (case AEN == 1).	0
ALPHA1_B_H_F	[3:0]	Specifies the Blue Alpha upper value (case AEN == 1).	0

NOTE: For more information, refer to VIDW3ALPHA0, 1 register.

1.5.2.30 Window 4 Position Control A Register (VIDOSD4A, R/W, Address = 0xF800_0080)

VIDOSD4A	Bit	Description	Initial State
OSD_LeftTopX_F	[21:11]	Specifies the Horizontal screen coordinate for left top pixel of OSD image.	0
OSD_LeftTopY_F	[10:0]	Specifies the Vertical screen coordinate for left top pixel of OSD image. (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate MUST be even value.)	0

1.5.2.31 Window 4 Position Control B Register (VIDOSD4B, R/W, Address = 0xF800_0084)

VIDOSD4B	Bit	Description	Initial State
OSD_RightBotX_F	[21:11]	Specifies the Horizontal screen coordinate for right bottom pixel of OSD image.	0
OSD_RightBotY_F	[10:0]	Specifies the Vertical screen coordinate for right bottom pixel of OSD image. (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate MUST be odd value.)	0

NOTE: Registers must have word boundary X position.

Therefore, 24 BPP mode must have X position by 1 pixel. (For example, X = 0, 1, 2, 3....)

16 BPP mode must have X position by 2 pixel. (For example, X = 0, 2, 4, 6....)

8 BPP mode must have X position by 4 pixel. (For example, X = 0, 4, 8, 12....)

1.5.2.32 Window 4 Position Control C Register (VIDOSD4C, R/W, Address = 0xF800_0088)

VIDOSD4C	Bit	Description	Initial State
Reserved	[24]	Reserved	0
ALPHA0_R_H_F	[23:20]	Specifies the Red Alpha upper value (case AEN == 0).	0
ALPHA0_G_H_F	[19:16]	Specifies the Green Alpha upper value (case AEN == 0).	0
ALPHA0_B_H_F	[15:12]	Specifies the Blue Alpha upper value (case AEN == 0).	0
ALPHA1_R_H_F	[11:8]	Specifies the Red Alpha upper value (case AEN == 1).	0
ALPHA1_G_H_F	[7:4]	Specifies the Green Alpha upper value (case AEN == 1).	0
ALPHA1_B_H_F	[3:0]	Specifies the Blue Alpha upper value (case AEN == 1).	0

NOTE: For more information, refer to VIDW4ALPHA0,1 register.

1.5.2.33 Frame Buffer Address 0 Register (VIDW)

- VIDW00ADD0B0, R/W, Address = 0xF800_00A0
- VIDW00ADD0B1, R/W, Address = 0xF800_00A4
- VIDW00ADD0B2, R/W, Address = 0xF800_20A0
- VIDW01ADD0B0, R/W, Address = 0xF800_00A8
- VIDW01ADD0B1, R/W, Address = 0xF800_00AC
- VIDW01ADD0B2, R/W, Address = 0xF800_20A8
- VIDW02ADD0B0, R/W, Address = 0xF800_00B0
- VIDW02ADD0B1, R/W, Address = 0xF800_00B4
- VIDW02ADD0B2, R/W, Address = 0xF800_20B0
- VIDW03ADD0B0, R/W, Address = 0xF800_00B8
- VIDW03ADD0B1, R/W, Address = 0xF800_00BC
- VIDW03ADD0B2, R/W, Address = 0xF800_20B8
- VIDW04ADD0B0, R/W, Address = 0xF800_00C0
- VIDW04ADD0B1, R/W, Address = 0xF800_00C4
- VIDW04ADD0B2, R/W, Address = 0xF800_20C0

VIDWxxADD0	Bit	Description	Initial State
VBASEU_F	[31:0]	Specifies A [31:0] of the start address for Video frame buffer.	0

1.5.2.34 Frame Buffer Address 1 Register (VIDW)

- VIDW00ADD1B0, R/W, Address = 0xF800_00D0
- VIDW00ADD1B1, R/W, Address = 0xF800_00D4
- VIDW00ADD1B2, R/W, Address = 0xF800_20D0
- VIDW01ADD1B0, R/W, Address = 0xF800_0D8
- VIDW01ADD1B1, R/W, Address = 0xF800_00DC
- VIDW01ADD1B2, R/W, Address = 0xF800_20D8
- VIDW02ADD1B0, R/W, Address = 0xF800_00E0
- VIDW02ADD1B1, R/W, Address = 0xF800_00E4
- VIDW02ADD1B2, R/W, Address = 0xF800_20E0
- VIDW03ADD1B0, R/W, Address = 0xF800_00E8
- VIDW03ADD1B1, R/W, Address = 0xF800_00EC
- VIDW03ADD1B2, R/W, Address = 0xF800_20E8
- VIDW04ADD1B0, R/W, Address = 0xF800_00F0
- VIDW04ADD1B1, R/W, Address = 0xF800_00F4
- VIDW04ADD1B2, R/W, Address = 0xF800_20F0

VIDWxxADD1	Bit	Description	Initial State
VBASEL_F	[31:0]	Specifies A[31:0] of the end address for Video frame buffer. VBASEL = VBASEU + (PAGEWIDTH+OFFSIZE) x (LINEVAL+1)	0x0

1.5.2.35 Rame Buffer Address 2 Register (VIDW)

- VIDW00ADD2, R/W, Address = 0xF800_0100
- VIDW01ADD2, R/W, Address = 0xF800_0104
- VIDW02ADD2, R/W, Address = 0xF800_0108
- VIDW03ADD2, R/W, Address = 0xF800_010C
- VIDW04ADD2, R/W, Address = 0xF800_0110

VIDWxxADD2	Bit	Description	Initial State
OFFSIZE_F	[25:13]	Specifies the Virtual screen offset size (number of byte). This value defines the difference between address of last byte displayed on the previous Video line and address of first byte to be displayed in the new Video line. OFFSIZE_F must have value that is multiple of 4-byte size or 0.	0
PAGEWIDTH_F	[12:0]	Specifies the Virtual screen page width (number of byte). This value defines the width of view port in the frame. PAGEWIDTH must have bigger value than the burst size and the size must be aligned word boundary.	0

NOTE: 'PAGEWIDTH + OFFSET' should be aligned double-word aligned (8-byte).

1.5.2.36 Video Interrupt Control 0 Register (VIDINTCON0, R/W, Address = 0xF800_0130)

VIDINTCON0	Bit	Description	Initial State
Reserved	[31:26]	Reserved	0
FIFOINTERVAL	[25:20]	Controls the interval of the FIFO interrupt.	0
SYSMAINCON	[19]	Sends complete interrupt enable bit to Main LCD 0 = Disables Interrupt 1 = Enables Interrupt Note: This bit is meaningful if both INTEN and I80IFDONE are high.	0
SYSSUBCON	[18]	Sends complete interrupt enable bit to Sub LCD 0 = Disables Interrupt. 1 = Enables Interrupt. Note: This bit is meaningful if both INTEN and I80IFDONE are high.	0
I80IFDONE	[17]	Enables the I80 Interface Interrupt (only for I80 Interface mode). 0 = Disables Interrupt. 1 = Enables Interrupt. Note: This bit is meaningful if INTEN is high.	0
FRAMESEL0	[16:15]	Specifies the Video Frame Interrupt 0 at start of: 00 = BACK Porch 01 = VSYNC 10 = ACTIVE	0

VIDINTCON0	Bit	Description	Initial State
		11 = FRONT Porch	
FRAMESEL1	[14:13]	Specifies the Video Frame Interrupt 1 at start of: 00 = None 01 = BACK Porch 10 = VSYNC 11 = FRONT Porch	0
INTFRMEN	[12]	Specifies the Video Frame Interrupt Enable Control Bit. 0 = Disables Video Frame Interrupt 1 = Enables Video Frame Interrupt Note: This bit is meaningful when INTEN is high.	0
FIFOSEL	[11:5]	Specifies the FIFO Interrupt control bit. Each bit has a special significance: [11] Window 4 control (0 = disables, 1 = enables) [10] Window 3 control (0 = disables, 1 = enables) [9] Window 2 control (0 = disables, 1 = enables) [8] Reserved [7] Reserved [6] Window 1 control (0 = disables, 1 = enables) [5] Window 0 control (0 = disables, 1 = enables) Note: This bit is meaningful if both INTEN and INTIFOEN are high	0
FIFOLEVEL	[4:2]	Selects the Video FIFO Interrupt Level. 000 = 0 ~ 25% 001 = 0 ~ 50% 010 = 0 ~ 75% 011 = 0% (empty) 100 = 100% (full)	0
INTIFOEN	[1]	Specifies the Video FIFO Interrupt Enable Control Bit. 0 = Disables Video FIFO Level Interrupt 1 = Enables Video FIFO Level Interrupt Note: This bit is meaningful if INTEN is high.	0
INTEN	[0]	Specifies the Video Interrupt Enable Control Bit. 0 = Disables Video Interrupt 1 = Enables Video Interrupt	0

NOTE:

1. If video frame interrupt occurs, you can select maximum two points by setting FRAMESEL0 and FRAMESEL1. For example, in case of FRAMESEL0=00 and FRAMESEL1=11, video frame interrupt is triggered both at the start of back porch and front porch.
2. S5PV210 interrupt controller has three interrupt sources related to display controller, namely, LCD[0], LCD[1] and LCD[2]. (For more information, refer to Chapter 4.1, “Vectored Interrupt Controller”). LCD[0] specifies FIFO Level interrupt, LCD[1] specifies video frame sync interrupt and LCD[2] specifies i80 done interface interrupt.

1.5.2.37 Video Interrupt Control 1 Register (VIDINTCON1, R/W, Address = 0xF800_0134)

VIDINTCON1	Bit	Description	Initial State
Reserved	[31:5]	Reserved	0
Reserved	[4:3]	Reserved (should be 0).	0
INTI80PEND	[2]	Specifies the i80 Done interrupt. Writes “1” to clear this bit. 0 = Interrupt has not been requested 1 = I80 Done status has asserted the interrupt request	0
INTFRMPEND	[1]	Specifies the Frame sync interrupt. Writes “1” to clear this bit. 0 = Interrupt has not been requested 1 = Frame sync status has asserted the interrupt request	0
INTFIFOPEND	[0]	Specifies the FIFO Level interrupt. Writes “1” to clear this bit. 0 = Interrupt has not been requested 1 = FIFO empty status has asserted the interrupt request	0

1.5.2.38 Win1 Color Key 0 Register (W1KEYCON0, R/W, Address = 0xF800_0140)

W1KEYCON0	Bit	Description	Initial State
KEYBLEN_F	[26]	Enables blending. 0 = Disables blending 1 = Enables blending using original Alpha for non-key area and KEY_ALPHA for key area	0
KEYEN_F	[25]	Enables/Disables Color Key (Chroma key). 0 = Disables color key 1 = Enables color key	0
DIRCON_F	[24]	Controls color key (Chroma key) direction. 0 = If the pixel value matches foreground image with COLVAL, the pixel from background image is displayed (only in OSD area) 1 = If the pixel value matches background image with COLVAL, the pixel from foreground image is displayed (only in OSD area)	0
COMPKEY_F	[23:0]	Each bit corresponds to COLVAL [23:0]. If some position bit is set, then it disables the position bit of COLVAL.	0

NOTE: Set BLD_PIX = 1, ALPHA_SEL =0, A_FUNC = 0x2, and B_FUNC = 0x3 to enable alpha blending using color key.

1.5.2.39 Win 1 Color key 1 Register (W1KEYCON1, R/W, Address = 0xF800_0144)

W1KEYCON1	Bit	Description	Initial State
COLVAL_F	[23:0]	Specifies the color key value for transparent pixel effect.	0

1.5.2.40 Win2 Color Key 0 Register (W2KEYCON0, R/W, Address = 0xF800_0148)

W2KEYCON0	Bit	Description	Initial State
KEYBLEN_F	[26]	Enables blending. 0 = Disables blending 1 = Enables blending using original Alpha for non-key area and KEY_ALPHA for key area	0
KEYEN_F	[25]	Enables color key (Chroma key). 0 = Disables color key 1 = Enables color key	0
DIRCON_F	[24]	Controls color key (Chroma key) direction. 0 = If the pixel value matches foreground image with COLVAL, the pixel from background image is displayed (only in OSD area) 1 = If the pixel value matches background image with COLVAL, the pixel from foreground image is displayed (only in OSD area)	0
COMPKEY_F	[23:0]	Each bit corresponds to COLVAL [23:0]. If some position bit is set, then it disables the position bit of COLVAL.	0

NOTE: Set BLD_PIX = 1, ALPHA_SEL =0, A_FUNC = 0x2, and B_FUNC = 0x3 to enable alpha blending using color key.

1.5.2.41 Win2 Color key 1 Register (W2KEYCON1, R/W, Address = 0xF800_014C)

W2KEYCON1	Bit	Description	Initial State
COLVAL_F	[23:0]	Specifies the color key value for transparent pixel effect.	0

1.5.2.42 Win3 Color Key 0 Register (W3KEYCON0, R/W, Address = 0xF800_0150)

W3KEYCON0	Bit	Description	Initial State
KEYBLEN_F	[26]	Enables blending. 0 = Disables blending 1 = Enables blending using original Alpha for non-key area and KEY_ALPHA for key area	0
KEYEN_F	[25]	Enables Color Key (Chroma key). 0 = Disables color key 1 = Enables color key	0
DIRCON_F	[24]	Controls Color key (Chroma key) direction. 0 = If the pixel value matches foreground image with COLVAL, the pixel from background image is displayed (only in OSD area) 1 = If the pixel value matches background image with COLVAL, the pixel from foreground image is displayed (only in OSD area)	0
COMPKEY_F	[23:0]	Each bit corresponds to COLVAL [23:0]. If some position bit is set, then it disables the position bit of COLVAL.	0

NOTE: Set BLD_PIX = 1, ALPHA_SEL =0, A_FUNC = 0x2, and B_FUNC = 0x3 to enable alpha blending using color key.

1.5.2.43 Win3 Color key 1 Register (W3KEYCON1, R/W, Address = 0xF800_0154)

W3KEYCON1	Bit	Description	Initial State
COLVAL_F	[23:0]	Specifies the color key value for transparent pixel effect.	0

1.5.2.44 Win4 Color Key 0 Register (W4KEYCON0, R/W, Address = 0xF800_0158)

W4KEYCON0	Bit	Description	Initial State
KEYBLEN_F	[26]	Enables blending. 0 = Disables blending 1 = Enables blending using original Alpha for non-key area and KEY_ALPHA for key area	0
KEYEN_F	[25]	Enables color Key (Chroma key). 0 = Disables color key 1 = Enables color key	0
DIRCON_F	[24]	Controls color key (Chroma key) direction. 0 = If the pixel value matches foreground image with COLVAL, the pixel from background image is displayed (only in OSD area) 1 = If the pixel value matches background image with COLVAL, the pixel from foreground image is displayed (only in OSD area)	0
COMPKEY_F	[23:0]	Each bit corresponds to COLVAL [23:0]. If some position bit is set, then it disables the COLVAL position bit.	0

NOTE: Set BLD_PIX = 1, ALPHA_SEL =0, A_FUNC = 0x2, and B_FUNC = 0x3 to enable alpha blending using color key.

1.5.2.45 Win4 Color key 1 Register (W4KEYCON1, R/W, Address = (W4KEYCON1, R/W, Address = 0xF800_015C)

W4KEYCON1	Bit	Description	Initial State
COLVAL_F	[23:0]	Specifies the color key value for transparent pixel effect.	0

NOTE: Both COLVAL and COMPKEY use 24-bit color data in all BPP modes.

@ BPP24 mode: 24-bit color value is valid.

- A. COLVAL
 - Red: COLVAL [23:17]
 - Green: COLVAL [15: 8]
 - Blue: COLVAL [7:0]
- B. COMPKEY
 - Red: COMPKEY [23:17]
 - Green: COMPKEY [15: 8]
 - Blue: COMPKEY [7:0]

@ BPP16 (5:6:5) mode: 16-bit color value is valid.

- A. COLVAL
 - Red: COLVAL [23:19]
 - Green: COLVAL [15: 10]
 - Blue: COLVAL [7:3]
- B. COMPKEY
 - Red: COMPKEY [23:19]
 - Green: COMPKEY [15: 10]
 - Blue: COMPKEY [7:3]
 - COMPKEY [18:16] must be 0x7.
 - COMPKEY [9: 8] must be 0x3.
 - COMPKEY [2:0] must be 0x7.

NOTE: COMPKEY register must be set properly for each BPP mode.

1.5.2.46 Win1 Color Key ALPHA Control Register (W1KEYALPHA, R/W, Address = 0xF800_0160)

W1KEYALPHA	Bit	Description	Initial State
-	[31:14]	Reserved.	0
KEYALPHA_R_F	[23:0]	Specifies the Key alpha R value.	0
KEYALPHA_G_F	[15:8]	Specifies the Key alpha G value.	0
KEYALPHA_B_F	[7:0]	Specifies the Key alpha B value.	0

1.5.2.47 Win2 Color Key ALPHA Control Register (W2KEYALPHA, R/W, Address = 0xF800_0164)

W2KEYALPHA	Bit	Description	Initial State
-	[31:14]	Reserved.	0
KEYALPHA_R_F	[23:0]	Specifies the Key alpha R value.	0
KEYALPHA_G_F	[15:8]	Specifies the Key alpha G value.	0
KEYALPHA_B_F	[7:0]	Specifies the Key alpha B value.	0

1.5.2.48 Win3 Color Key ALPHA Control Register (W3KEYALPHA, R/W, Address = 0xF800_0168)

W3KEYALPHA	Bit	Description	Initial State
-	[31:14]	Reserved.	0
KEYALPHA_R_F	[23:0]	Specifies the Key alpha R value.	0
KEYALPHA_G_F	[15:8]	Specifies the Key alpha G value.	0
KEYALPHA_B_F	[7:0]	Specifies the Key alpha B value.	0

1.5.2.49 Win4 Color Key ALPHA Control Register (W4KEYALPHA, R/W, Address = 0xF800_016C)

W4KEYALPHA	Bit	Description	Initial State
-	[31:14]	Reserved.	0
KEYALPHA_R_F	[23:0]	Specifies the Key alpha R value.	0
KEYALPHA_G_F	[15:8]	Specifies the Key alpha G value.	0
KEYALPHA_B_F	[7:0]	Specifies the Key alpha B value.	0

1.5.2.50 Dithering Control 1 Register (DITHMODE, R/W, Address = 0xF800_0170)

DITHMODE	Bit	Description	Initial State
	[7]	Not used for normal access (Writing non-zero values to these registers results in abnormal behavior.)	0
RDithPos	[6:5]	Controls Red Dither bit. 00 = 8-bit 01 = 6-bit 10 = 5-bit	0
GDithPos	[4:3]	Controls Green Dither bit. 00 = 8-bit 01 = 6-bit 10 = 5-bit	0
BDithPos	[2:1]	Controls Blue Dither bit. 00 = 8-bit 01 = 6-bit 10 = 5-bit	0
DITHEN_F	[0]	Enables Dithering bit. 0 = Disables dithering 1 = Enables dithering	0

1.5.2.51 Win0 Color MAP (WIN0MAP, R/W, Address = 0xF800_0180)

WIN0MAP	Bit	Description	Initial State
MAPCOLEN_F	[24]	Specifies window's color mapping control bit. If this bit is enabled, then Video DMA stops and MAPCOLOR appears on background image instead of original image. 0 = Disables 1 = Enables	0
MAPCOLOR	[23:0]	Specifies the color value.	0

1.5.2.52 Win1 Color MAP (WIN1MAP, R/W, Address = 0xF800_0184)

WIN1MAP	Bit	Description	Initial State
MAPCOLEN_F	[24]	Specifies the window's color mapping control bit. If this bit is enabled, then Video DMA stops and MAPCOLOR appears on background image instead of original image. 0 = Disables 1 = Enables	0
MAPCOLOR	[23:0]	Specifies the color value.	0

1.5.2.53 Win2 Color MAP (WIN2MAP, R/W, Address = 0xF800_0188)

WIN2MAP	Bit	Description	Initial State
MAPCOLEN_F	[24]	Specifies the window's color mapping control bit. If this bit is enabled, then Video DMA stops and MAPCOLOR appears on background image instead of original image. 0 = Disables 1 = Enables	0
MAPCOLOR	[23:0]	Specifies the color value.	0

1.5.2.54 Win3 Color MAP (WIN3MAP, R/W, Address = 0xF800_018C)

WIN3MAP	Bit	Description	Initial State
MAPCOLEN_F	[24]	Specifies the window's color mapping control bit. If this bit is enabled, then Video DMA stops and MAPCOLOR appears on background image instead of original image. 0 = Disables 1 = Enables	0
MAPCOLOR	[23:0]	Specifies the color value.	0

1.5.2.55 Win4 Color MAP (WIN4MAP, R/W, Address = 0xF800_0190)

WIN4MAP	Bit	Description	Initial State
MAPCOLEN_F	[24]	Specifies the window's color mapping control bit. If this bit is enabled, then Video DMA stops and MAPCOLOR appears on background image instead of original image. 0 = Disables 1 = Enables	0
MAPCOLOR	[23:0]	Specifies the color value.	0

1.5.2.56 Window Palette control Register (WPALCON_H, R/W, Address = 0xF800_019C)

WPALCON_H	Bit	Description	Initial State
Reserved	[31:19]	Reserved	0
W4PAL_H	[18:17]	W4PAL[2:1]	0
Reserved	[16:15]	Reserved	0
W3PAL_H	[14:13]	W3PAL[2:1]	0
Reserved	[12:11]	Reserved	0
W2PAL_H	[10: 9]	W2PAL[2:1]	0
Reserved	[8: 0]	Reserved	0

1.5.2.57 Window Palette control Register (WPALCON_L, R/W, Address = 0xF800_01A0)

WPALCON_L	Bit	Description	Initial State
Reserved	[31:23]	Reserved	0
PALUPDATEEN	[9]	0 = Normal Mode 1 = Enable (Palette Update)	0
W4PAL_L	[8]	W4PAL[0]	0
W3PAL_L	[7]	W3PAL[0]	0
W2PAL_L	[6]	W2PAL[0]	0
W1PAL_L	[5: 3]	W1PAL[2:0]	0
W0PAL_L	[2: 0]	W0PAL[2:0]	0

NOTE: WPALCON = {WPALCON_H,WPALCON_L}

WPALCON	Description	Initial State
PALUPDATEEN	0 = Normal Mode 1 = Enable (Palette Update)	0
W4PAL[3:0]	Specifies the size of palette data format of Window 4. 000 = 16-bit (5:6:5) 001 = 16-bit (A:5:5:5) 010 = 18-bit (6:6:6) 011 = 18-bit (A:6:6:5) 100 = 19-bit (A:6:6:6) 101 = 24-bit (8:8:8) 110 = 25-bit (A:8:8:8) 111 = 32-bit (8:8:8:8) (A: 8-bit)	0
W3PAL[2:0]	Specifies the size of palette data format of Window 3. 000 = 16-bit (5:6:5) 001 = 16-bit (A:5:5:5) 010 = 18-bit (6:6:6) 011 = 18-bit (A:6:6:5) 100 = 19-bit (A:6:6:6) 101 = 24-bit (8:8:8) 110 = 25-bit (A:8:8:8) 111 = 32-bit (8:8:8:8) (A: 8-bit)	0
W2PAL[2:0]	Specifies the size of palette data format of Window 2. 000 = 16-bit (5:6:5) 001 = 16-bit (A:5:5:5) 010 = 18-bit (6:6:6) 011 = 18-bit (A:6:6:5) 100 = 19-bit (A:6:6:6) 101 = 24-bit (8:8:8) 110 = 25-bit (A:8:8:8) 111 = 32-bit (8:8:8:8) (A: 8-bit)	0

WPALCON	Description	Initial State
W1PAL[2:0]	Specifies the size of palette data format of Window 1. 000 = 25-bit (A:8:8:8) 001 = 24-bit (8:8:8) 010 = 19-bit (A:6:6:6) 011 = 18-bit (A:6:6:5) 100 = 18-bit (6:6:6) 101 = 16-bit (A:5:5:5) 110 = 16-bit (5:6:5) 111 = 32-bit (8:8:8:8) (A: 8-bit)	0
W0PAL[2:0]	Specifies the size of palette data format of Window 0. 000 = 25-bit (A:8:8:8) 001 = 24-bit (8:8:8) 010 = 19-bit (A:6:6:6) 011 = 18-bit (A:6:6:5) 100 = 18-bit (6:6:6) 101 = 16-bit (A:5:5:5) 110 = 16-bit (5:6:5) 111 = 32-bit (8:8:8:8) (A: 8-bit)	0

NOTE: The bit map for W0/ W1 is different from W2/ W3/ W4.

1.5.2.58 I80 / RGB Trigger Control Register (TRIGCON, R/W, Address = 0xF800_01A4)

TRIGCON	Bit	Description	Initial State
-	[31:27]	Reserved	0
SWTRGCMW4BUF	[26]	Specifies Window 4 double buffer trigger. 1 = Enables Software Trigger Command (write only) * Only when TRGMODE_W4BUF is '1'	0
TRGMODE_W4BUF	[25]	Specifies Window 4 double buffer trigger. 0 = Disables Trigger 1 = Enables Trigger	0
Reserved	[24:22]	Reserved	0
SWTRGCMW3BUF	[21]	Specifies Window 3 double buffer trigger. 1 = Enables Software Trigger Command (write only) * Only when TRGMODE_W3BUF is '1'	0
TRGMODE_W3BUF	[20]	Specifies Window 3 double buffer trigger. 0 = Disables Trigger 1 = Enables Trigger	0
Reserved	[19:17]	Reserved	0
SWTRGCMW2BUF	[16]	Specifies Window 2 double buffer trigger. 1 = Enables Software Trigger Command (write only) * Only when TRGMODE_W2BUF is '1'	0
TRGMODE_W2BUF	[15]	Specifies Window 2 double buffer trigger. 0 = Disables Trigger 1 = Enables Trigger	0
Reserved	[14:12]	Reserved	0
SWTRGCMW1BUF	[11]	Specifies Window 1 double buffer trigger. 1 = Enables Software Trigger Command (write only) * Only when TRGMODE_W1BUF is '1'	0
TRGMODE_W1BUF	[10]	Specifies Window 1 double buffer trigger. 0 = Disables Trigger 1 = Enables Trigger	0
Reserved	[9:7]	Reserved	0
SWTRGCMW0BUF	[6]	Specifies Window 0 double buffer trigger. 1 = Enables Software Trigger Command (write only) * Only when TRGMODE_W0BUF is '1'	0
TRGMODE_W0BUF	[5]	Specifies Window 0 double buffer trigger. 0 = Disables Trigger 1 = Enables Trigger	0
Reserved	[4:3]	Reserved	0

TRIGCON	Bit	Description	Initial State
SWFRSTATUS_I80	[2]	Specifies Frame Done Status (read only; I80 start trigger) 0 = Not Requested 1 = Requested * Clear Condition: Read or New Frame Start * Only when TRGMODE is '1'	0
SWTRGCMOD_I80	[1]	Enables I80 start trigger. 1 = Software Triggering Command (write only) * Only when TRGMODE is '1'	0
TRGMODE_I80	[0]	Enables I80 start trigger. 0 = Disables i80 Software Trigger 1 = Enables i80 Software Trigger	0

NOTE: Two continuous software trigger inputs generated in some video clocks (VCLK) are recognized as one.

1.5.2.59 LCD I80 Interface Control 0 (I80IFCONAx)

- I80IFCONA0, R/W, Address = 0xF800_01B0
- I80IFCONA1, R/W, Address = 0xF800_01B4

I80IFCONAx	Bit	Description	Initial State
Reserved	[22:20]	Reserved	0
LCD_CS_SETUP	[19:16]	Specifies the numbers of clock cycles for the active period of address signal enable to chip select enable.	0
LCD_WR_SETUP	[15:12]	Specifies the numbers of clock cycles for the active period of CS signal enable to write signal enable.	0
LCD_WR_ACT	[11:8]	Specifies the numbers of clock cycles for the active period of chip select enable.	0
LCD_WR_HOLD	[7:4]	Specifies the numbers of clock cycles for the active period of chip select disable to write signal disable.	0
Reserved	[3]	Reserved	
RSPOL	[2]	Specifies the polarity of RS Signal 0 = Low 1 = High	0
Reserved	[1]	Reserved	0
I80IFEN	[0]	Controls the LCD I80 interface. 0 = Disables 1 = Enables	0

1.5.2.60 LCD I80 Interface Control 1 (I80IFCONBx)

- I80IFCONB0, R/W, Address = 0xF800_01B8
- I80IFCONB1, R/W, Address = 0xF800_01BC

I80IFCONBx	Bit	Description	Initial State
Reserved	[11:10]	Reserved	0
NORMAL_CMD_ST	[9]	1 = Normal Command Start * Auto clears after sending out one set of commands	0
Reserved	[8:7]	Reserved	
FRAME_SKIP	[6:5]	Specifies the I80 Interface Output Frame Decimation Factor. 00 = 1 (No Skip) 01 = 2 10 = 3	00
Reserved	[4]	Reserved	0
AUTO_CMD_RATE	[3:0]	0000 = Disables auto command (If you don't use any auto-command, then you should set AUTO_CMD_RATE as "0000".) 0001 = per 2 Frames 0010 = per 4 Frames 0011 = per 6 Frames ... 1111 = per 30 Frames	0000

1.5.2.61 Color GAIN Control Register (COLORGAINCON, R/W, Address = 0xF800_01C0)

COLORGAINCON	Bit	Description	Initial State
Reserved	[31:30]	Reserved	0
CG_RGAIN	[29:20]	<p>Specifies the color gain value of R data (maximum 4, 8-bit resolution).</p> <p>0h000 = 0 0h001 = 0.00390625 (1/256) 0h002 = 0.0078125 (2/256) ... 0hOFF = 0.99609375 (255/256) 0h100 = 1.0 ... 0x3FF = 3.99609375 (max)</p>	0x100
CG_GGAIN	[19:10]	<p>Specifies the color gain value of G data (maximum 4, 8-bit resolution).</p> <p>0h000 = 0 0h001 = 0.00390625 (1/256) 0h002 = 0.0078125 (2/256) ... 0hOFF = 0.99609375 (255/256) 0h100 = 1.0 ... 0x3FF = 3.99609375 (max)</p>	0x100
CG_BGAIN	[9:0]	<p>Specifies the color gain value of B data (maximum 4, 8-bit resolution).</p> <p>0h000 = 0 0h001 = 0.00390625 (1/256) 0h002 = 0.0078125 (2/256) ... 0hOFF = 0.99609375 (255/256) 0h100 = 1.0 ... 0x3FF = 3.99609375 (max)</p>	0x100

Samsung Confidential

1.5.2.62 LCD i80 Interface Command Control 0 (LDI_CMDCON0, R/W, Address = 0xF800_01D0)

LDI_CMDCON0	Bit	Description	Initial State
Reserved	[31:24]	Reserved	
CMD11_EN	[23:22]	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00
CMD10_EN	[21:20]	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00
CMD9_EN	[19:18]	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00
CMD8_EN	[17:16]	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00
CMD7_EN	[15:14]	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00
CMD6_EN	[13:12]	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00
CMD5_EN	[11:10]	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00
CMD4_EN	[9:8]	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00
CMD3_EN	[7:6]	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00
CMD2_EN	[5:4]	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Normal and Auto Command Enable	00
CMD1_EN	[3:2]	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00

LDI_CMDCON0	Bit	Description	Initial State
CMD0_EN	[1:0]	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00

1.5.2.63 LCD i80 Interface Command Control 1 (LDI_CMDCON1, R/W, Address = 0xF800_01D4)

LDI_CMDCON1	Bit	Description	Initial State
Reserved	[31:10]	Reserved	0
CMD11_RS	[11]	Controls Command 11 RS	0
CMD10_RS	[10]	Controls Command 10 RS	0
CMD9_RS	[9]	Controls Command 9 RS	0
CMD8_RS	[8]	Controls Command 8 RS	0
CMD7_RS	[7]	Controls Command 7 RS	0
CMD6_RS	[6]	Controls Command 6 RS	0
CMD5_RS	[5]	Controls Command 5 RS	0
CMD4_RS	[4]	Controls Command 4 RS	0
CMD3_RS	[3]	Controls Command 3 RS	0
CMD2_RS	[2]	Controls Command 2 RS	0
CMD1_RS	[1]	Controls Command 1 RS	0
CMD0_RS	[0]	Controls Command 0 RS	0

1.5.2.64 I80 System Interface Manual Command Control 0 (SIFCCON0, R/W, Address = 0xF800_01E0)

SIFCCON0	Bit	Description	Initial State
Reserved	[7]	Reserved (should be 0)	0
SYS_ST_CON	[6]	Controls LCD i80 System Interface ST Signal. 0 = Low 1 = High	0
SYS_RS_CON	[5]	Controls LCD i80 System Interface RS Signal. 0 = Low 1 = High	0
SYS_nCS0_CON	[4]	Controls LCD i80 System Interface nCS0 (main) Signal. 0 = Disables (High) 1 = Enables (Low)	0
SYS_nCS1_CON	[3]	Controls LCD i80 System Interface nCS1 (sub) Signal. 0 = Disables (High) 1 = Enables (Low)	0
SYS_nOE_CON	[2]	Controls LCD i80 System Interface nOE Signal. 0 = Disables (High) 1 = Enables (Low)	0
SYS_nWE_CON	[1]	Controls LCD i80 System Interface nWE Signal. 0 = Disables (High) 1 = Enables (Low)	0
SCOMEN	[0]	Enables LCD i80 System Interface Command Mode. 0 = Disables (Normal Mode) 1= Enables (Manual Command Mode)	

1.5.2.65 I80 System Interface Manual Command Control 1 (SIFCCON1, R/W, Address = 0xF800_01E4)

SIFCCON1	Bit	Description	Initial State
SYS_WDATA	[23:0]	Controls the LCD i80 System Interface Write Data.	0

1.5.2.66 I80 System Interface Manual Command Control 2 (SIFCCON2, R/W, Address = 0xF800_01E8)

SIFCCON2	Bit	Description	Initial State
SYS_RDATA	[23:0]	Controls the LCD i80 System Interface Read Data.	0

1.5.2.67 Hue Control Registers (HUECOEF00, R/W, Address = 0xF800_0200)

- HUECOEF00, R/W, Address = 0xF800_01EC
- HUECOEF01, R/W, Address = 0xF800_01F0
- HUECOEF02, R/W, Address = 0xF800_01F4
- HUECOEF03, R/W, Address = 0xF800_01F8

HUECOEF0x	Bit	Description	Initial State
-	[31:26]	Reserved	0
CBGx_P	[25:16]	<p>Specifies the Hue matrix coefficient 00 (when 'cb + In_offset' is positive). (Signed)</p> <p>0h000 = 0 0h001 = 0.00390625 (1/256) 0h002 = 0.0078125 (2/256) ... 0h0FF = 0.99609375 (255/256) 0h100 = 1.0 (256/256) 0h300 = -1.0 (-256/256) 0h301 = - 0.99609375 (-255/256) ... 0h3FF = -0.00390625 (-1/256)</p> <p>0h101 ~ 2FF = Reserved (don't use)</p>	0x100
-	[15:10]	Reserved	0
CBGx_N	[9:0]	<p>Specifies the Hue matrix coefficient 00 (when 'cb + In_offset' is negative). (Signed)</p> <p>0h000 = 0 0h001 = 0.00390625 (1/256) 0h002 = 0.0078125 (2/256) ... 0h0FF = 0.99609375 (255/256) 0h100 = 1.0 (256/256) 0h300 = -1.0 (-256/256) 0h301 = - 0.99609375 (-255/256) ... 0h3FF = -0.00390625 (-1/256)</p> <p>0h101 ~ 2FF = Reserved (don't use)</p>	0x100

1.5.2.68 Hue Offset Control Register (HUEOFFSET, R/W, Address = 0xF800_01fC)

HUEOFFSET	Bit	Description	Initial State
-	[31:25]	Reserved	0
OFFSET_IN	[24:16]	Specifies the Hue matrix input offset (signed). 0h000 = + 0 0h001 = + 1 0h002 = + 2 ... 0h0FF = + 255 0h100 = - 256 ... 0x1FF = - 1	0x180 (-128)
-	[15:9]	Reserved	0
OFFSET_OUT	[8:0]	Specifies the Hue matrix output offset (signed). 0h000 = + 0 0h001 = + 1 0h002 = + 2 ... 0h0FF = + 255 0h100 = - 256 ... 0x1FF = - 1	0x080 (+128)

NOTE: Generally, HUE_OFFSET_IN = -128 and HUE_OFFSET_OUT = +128

Hue Equation:

$$\begin{aligned} Cb < \text{hue} > &= CBG0 \cdot (Cb + \text{OFFSET_IN}) + CBG1 \cdot (Cr + \text{OFFSET_IN}) + \text{OFFSET_OUT} \\ Cr < \text{hue} > &= CRG0 \cdot (Cb + \text{OFFSET_IN}) + CRG1 \cdot (Cr + \text{OFFSET_IN}) + \text{OFFSET_OUT} \end{aligned}$$

Coefficient Decision:

$$\begin{aligned} CBG0 &= (Cb - 128) \geq 0 ? CBG0_P : CBG0_N \\ CBG1 &= (Cr - 128) \geq 0 ? CBG1_P : CBG1_N \\ CRG0 &= (Cb - 128) \geq 0 ? CRG0_P : CRG0_N \\ CRG1 &= (Cr - 128) \geq 0 ? CRG1_P : CRG1_N \end{aligned}$$

1.5.2.69 Window 0 Alpha0 Control Register (VIDW0ALPHA0, R/W, Address = 0xF800_0200)

VIDW0ALPHA0	Bit	Description	Initial State
Reserved	[24]	Reserved	0
ALPHA0_R_F	[23:16]	Specifies the Red Alpha value (case AEN == 0).	0
ALPHA0_G_F	[15:8]	Specifies the Green Alpha value (case AEN == 0).	0
ALPHA0_B_F	[7:0]	Specifies the Blue Alpha value (case AEN == 0).	0

1.5.2.70 Window 0 Alpha1 control Register (VIDW0 ALPHA1, R/W, Address = 0xF800_0204)

VIDW0ALPHA1	Bit	Description	Initial State
Reserved	[24]	Reserved	0
ALPHA1_R_F	[23:16]	Specifies the Red Alpha value (case AEN == 1).	0
ALPHA1_G_F	[15:8]	Specifies the Green Alpha value (case AEN == 1).	0
ALPHA1_B_F	[7:0]	Specifies the Blue Alpha value (case AEN == 1).	0

1.5.2.71 Window 1 Alpha0 control Register (VIDW1ALPHA0, R/W, Address = 0xF800_0208)

VIDW1ALPHA0	Bit	Description	Initial State
Reserved	[24]	Reserved	0
Reserved	[23:20]	Reserved	0
ALPHA0_R_L_F	[19:16]	Specifies the Red Alpha lower value (case AEN == 0).	0
-	[15:12]	Reserved	0
ALPHA0_G_L_F	[11: 8]	Specifies the Green Alpha lower value (case AEN == 0).	0
-	[7: 4]	Reserved	0
ALPHA0_B_L_F	[3: 0]	Specifies the Blue Alpha lower value (case AEN == 0).	0

NOTE: ALPHA0_R (G, B) [7:4] = ALPHA0_R(G,B)_H[3:0]@VIDOSD1C
 ALPHA0_R(G,B) [3:0] = ALPHA0_R(G,B)_L[3:0]@VIDW1ALPHA0

1.5.2.72 Window 1 Alpha1 Control Register (VIDW0ALPHA1, R/W, Address = 0xF800_020C)

VIDW0ALPHA1	Bit	Description	Initial State
Reserved	[24]	Reserved	0
Reserved	[23:20]	Reserved	0
ALPHA1_R_L_F	[19:16]	Specifies the Red Alpha lower value (case AEN == 1).	0
-	[15:12]	Reserved	0
ALPHA1_G_L_F	[11: 8]	Specifies the Green Alpha lower value (case AEN == 1).	0
-	[7: 4]	Reserved	0
ALPHA1_B_L_F	[3: 0]	Specifies the Blue Alpha lower value (case AEN == 1).	0

NOTE: ALPHA1_R(G,B) [7:4] = ALPHA1_R(G,B)_H[3:0]@VIDOSD1C
 ALPHA1_R(G,B) [3:0] = ALPHA1_R(G,B)_L[3:0]@VIDW1ALPHA1

1.5.2.73 Window 2 Alpha0 Control Register (VIDW0ALPHA0, R/W, Address = 0xF800_0210)

VIDW0ALPHA0	Bit	Description	Initial State
Reserved	[24]	Reserved	0
Reserved	[23:20]	Reserved	0
ALPHA0_R_L_F	[19:16]	Specifies the Red Alpha lower value (case AEN == 0).	0
-	[15:12]	Reserved	0
ALPHA0_G_L_F	[11: 8]	Specifies the Green Alpha lower value (case AEN == 0).	0
-	[7: 4]	Reserved	0
ALPHA0_B_L_F	[3: 0]	Specifies the Blue Alpha lower value (case AEN == 0).	0

NOTE: ALPHA0_R (G, B) [7:4] = ALPHA0_R (G, B)_H[3:0]@VIDOSD2C
 ALPHA0_R (G, B) [3:0] = ALPHA0_R (G, B)_L[3:0]@VIDW2ALPHA0

1.5.2.74 Window 2 Alpha1 control Register (VIDW2ALPHA1, R/W, Address = 0xF800_0214)

VIDW2ALPHA1	Bit	Description	Initial State
Reserved	[24]	Reserved	0
Reserved	[23:20]	Reserved	0
ALPHA1_R_L_F	[19:16]	Specifies the Red Alpha lower value (case AEN == 1).	0
-	[15:12]	Reserved	0
ALPHA1_G_L_F	[11: 8]	Specifies the Green Alpha lower value (case AEN == 1).	0
-	[7: 4]	Reserved	0
ALPHA1_B_L_F	[3: 0]	Specifies the Blue Alpha lower value (case AEN == 1).	0

NOTE: ALPHA1_R (G, B) [7:4] = ALPHA1_R (G, B)_H[3:0]@VIDOSD2C
 ALPHA1_R (G, B) [3:0] = ALPHA1_R (G, B)_L[3:0]@VIDW2ALPHA1

1.5.2.75 Window 3 Alpha0 Control Register (VIDW0ALPHA0, R/W, Address = 0xF800_0218)

VIDW0ALPHA0	Bit	Description	Initial State
Reserved	[24]	Reserved	0
Reserved	[23:20]	Reserved	0
ALPHA0_R_L_F	[19:16]	Specifies the Red Alpha lower value (case AEN == 0).	0
-	[15:12]	Reserved	0
ALPHA0_G_L_F	[11: 8]	Specifies the Green Alpha lower value (case AEN == 0).	0
-	[7: 4]	Reserved	0
ALPHA0_B_L_F	[3: 0]	Specifies the Blue Alpha lower value (case AEN == 0).	0

NOTE: ALPHA0_R(G,B) [7:4] = ALPHA0_R(G,B)_H[3:0]@VIDOSD3C
 ALPHA0_R(G,B) [3:0] = ALPHA0_R(G,B)_L[3:0]@VIDW3ALPHA0

1.5.2.76 Window 3 Alpha1 Control Register (VIDW3ALPHA1, R/W, Address = 0xF800_021C)

VIDW3ALPHA1	Bit	Description	Initial State
Reserved	[24]	Reserved	0
Reserved	[23:16]	Reserved	0
ALPHA1_R_L_F	[19:16]	Specifies the Red Alpha lower value (case AEN == 1).	0
-	[15:12]	Reserved	0
ALPHA1_G_L_F	[11: 8]	Specifies the Green Alpha lower value (case AEN == 1).	0
-	[7: 4]	Reserved	0
ALPHA1_B_L_F	[3: 0]	Specifies the Blue Alpha lower value (case AEN == 1).	0

NOTE: ALPHA1_R(G,B) [7:4] = ALPHA1_R(G,B)_H[3:0]@VIDOSD3C
 ALPHA1_R(G,B) [3:0] = ALPHA1_R(G,B)_L[3:0]@VIDW3ALPHA1

1.5.2.77 Window 4 Alpha0 Control Register (VIDW4_ALPHA0, R/W, Address = 0xF800_0220)

VIDW4ALPHA0	Bit	Description	Initial State
Reserved	[24]	Reserved	0
Reserved	[23:20]	Reserved	0
ALPHA0_R_L_F	[19:16]	Specifies the Red Alpha lower value (case AEN == 0).	0
-	[15:12]	Reserved	0
ALPHA0_G_L_F	[11: 8]	Specifies the Green Alpha lower value (case AEN == 0).	0
-	[7: 4]	Reserved	0
ALPHA0_B_L_F	[3: 0]	Specifies the Blue Alpha lower value (case AEN == 0).	0

NOTE: ALPHA0_R(G,B) [7:4] = ALPHA0_R(G,B)_H[3:0]@VIDOSD4C
 ALPHA0_R(G,B) [3:0] = ALPHA0_R(G,B)_L[3:0]@VIDW4ALPHA0

1.5.2.78 Window 4 Alpha1 Control Register (VIDW0ALPHA1, R/W, Address = 0xF800_0224)

VIDW0ALPHA1	Bit	Description	Initial State
Reserved	[24]	Reserved	0
Reserved	[23:20]	Reserved	0
ALPHA1_R_L_F	[19:16]	Specifies the Red Alpha lower value (case AEN == 1).	0
-	[15:12]	Reserved	0
ALPHA1_G_L_F	[11: 8]	Specifies the Green Alpha lower value (case AEN == 1).	0
-	[7: 4]	Reserved	0
ALPHA1_B_L_F	[3: 0]	Specifies the Blue Alpha lower value (case AEN == 1).	0

NOTE: ALPHA1_R(G,B) [7:4] = ALPHA1_R(G,B)_H[3:0]@VIDOSD4C
ALPHA1_R(G,B) [3:0] = ALPHA1_R(G,B)_L[3:0]@VIDW4ALPHA1

1.5.2.79 Window 1 Blending Equation Control Register (BLENDEQ1, R/W, Address = 0xF800_0244)

BLENDEQ1	Bit	Description	Initial State
Reserved	[31:22]	Reserved	0x000
Q_FUNC_F	[21:18]	Specifies the constant used in alphaB (alpha value of *background) 0000 = 0 (zero) 0001 = 1 (maximum) 0010 = **alphaA (alpha value of *foreground) 0011 = 1 - alphaA 0100 = alphaB 0101 = 1 - alphaB 0110 = ALPHA0 0111 = Reserved 100x = Reserved 1010 = A (foreground color data) 1011 = 1 - A 1100 = B (background color data) 1101 = 1 - B 111x = Reserved	0x0
-	[17:16]	Reserved	00
P_FUNC_F	[15:12]	Specifies the constant used in alpha. Same as above (see COEF_Q).	0x0
-	[11:10]	Reserved	00
B_FUNC_F	[9:6]	Specifies the constant used in B. Same as above (see COEF_Q).	0x3
-	[5:4]	Reserved	00
A_FUNC_F	[3:0]	Specifies the constant used in A. Same as above (see COEF_Q).	0x2

NOTE: For more information, refer to Figure 1-5, "Blending equation".

background = Window 0, foreground = Window 1 (in Blend Equation 1)

alphaA and alphaB are decided by BPPMODE_F, BLD_PIX, ALPHA_SEL @WINCONx, and WxPAL @WPALCON.

1.5.2.80 Window 2 Blending Equation Control Register (BLENDEQ2, R/W, Address = 0xF800_0248)

BLENDEQ2	Bit	Description	Initial State
Reserved	[31:22]	Reserved	0x000
Q_FUNC_F	[21:18]	Specifies the constant used in alphaB (alpha value of *background). 0000 = 0 (zero) 0001 = 1 (maximum) 0010 = **alphaA (alpha value of *foreground) 0011 = 1 – alphaA 0100 = alphaB 0101 = 1 – alphaB 0110 = ALPHA0 0111 = Reserved 100x = Reserved 1010 = A (foreground color data) 1011 = 1 – A 1100 = B (background color data) 1101 = 1 – B 111x = Reserved	0x0
-	[17:16]	Reserved	00
P_FUNC_F	[15:12]	Specifies the constant used in alpha. Same as above (see COEF_Q)	0x0
-	[11:10]	Reserved	00
B_FUNC_F	[9:6]	Specifies the constant used in B. Same as above (see COEF_Q)	0x3
-	[5:4]	Reserved	00
A_FUNC_F	[3:0]	Specifies the constant used in A. Same as above (see COEF_Q)	0x2

NOTE: For more information, refer to Figure 1-5, “Blending equation”.

background = Window 01, foreground = Window 2 (in Blend Equation 2)

alphaA and alphaB are decided by BPPMODE_F, BLD_PIX, ALPHA_SEL @WINCONx, and WxPAL @WPALCON.

1.5.2.81 Window 3 Blending Equation Control Register (BLENDEQ3, R/W, Address = 0xF800_024C)

BLENDEQ3	Bit	Description	Initial State
Reserved	[31:22]	Reserved	0x000
Q_FUNC_F	[21:18]	Specifies the constant used in alphaB (alpha value of *background) 0000 = 0 (zero) 0001 = 1 (maximum) 0010 = **alphaA (alpha value of *foreground) 0011 = 1 – alphaA 0100 = alphaB 0101 = 1 – alphaB 0110 = ALPHA0 0111 = Reserved 100x = Reserved 1010 = A (foreground color data) 1011 = 1 – A 1100 = B (background color data) 1101 = 1 – B 111x = Reserved	0x0
-	[17:16]	Reserved	00
P_FUNC_F	[15:12]	Specifies the constant used in alpha. Same as above (see COEF_Q).	0x0
-	[11:10]	Reserved	00
B_FUNC_F	[9:6]	Specifies the constant used in B. Same as above (see COEF_Q).	0x3
-	[5:4]	Reserved	00
A_FUNC_F	[3:0]	Specifies the constant used in A. Same as above (see COEF_Q).	0x2

NOTE: For more information, refer to Figure 1-5, “Blending equation”.

background = Window 012, foreground = Window 3 (in Blend Equation 3)

alphaA and alphaB are decided by BPPMODE_F, BLD_PIX, ALPHA_SEL @WINCONx, and WxPAL @WPALCON.

1.5.2.82 Window 4 Blending Equation Control Register (BLENDEQ4, R/W, Address = 0xF800_0250)

BLENDEQ4	Bit	Description	Initial State
Reserved	[31:22]	Reserved	0x000
Q_FUNC_F	[21:18]	Specifies the constant used in alphaB (alpha value of *background) 0000 = 0 (zero) 0001 = 1 (maximum) 0010 = **alphaA (alpha value of *foreground) 0011 = 1 – alphaA 0100 = alphaB 0101 = 1 – alphaB 0110 = ALPHA0 0111 = Reserved 100x = Reserved 1010 = A (foreground color data) 1011 = 1 – A 1100 = B (background color data) 1101 = 1 – B 111x = Reserved	0x0
-	[17:16]	Reserved	00
P_FUNC_F	[15:12]	Specifies the constant used in alpha. Same as above (see COEF_Q).	0x0
-	[11:10]	Reserved	00
B_FUNC_F	[9:6]	Specifies the constant used in B. Same as above (see COEF_Q).	0x3
-	[5:4]	Reserved	00
A_FUNC_F	[3:0]	Specifies the constant used in A. Same as above (see COEF_Q).	0x2

NOTE: For more information, refer to Figure 1-5, “Blending equation”.

background = Window 0123, foreground = Window 4 (in Blend Equation 4)

alphaA and alphaB are decided by BPPMODE_F, BLD_PIX, ALPHA_SEL @WINCONx, and WxPAL @WPALCON.

1.5.2.83 Blending Equation Control Register (BLENDCON, R/W, Address = 0xF800_0260)

BLENDCON	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0x000
BLEND_NEW	[0]	Specifies the Alpha value width. 0 = 4-bit alpha value 1 = 8-bit alpha value	0x0

1.5.2.84 Window RTQOS Control Registers

- W0RTQOSCON, R/W, Address = 0xF800_0264
- W1RTQOSCON, R/W, Address = 0xF800_0268
- W2RTQOSCON, R/W, Address = 0xF800_026C
- W3RTQOSCON, R/W, Address = 0xF800_0270
- W4RTQOSCON, R/W, Address = 0xF800_0274

HUECOEF0x	Bit	Description	Initial State
-	[31:12]	Reserved (should be 0)	0
FIFOLEVEL	[11:4]	Specifies the real-time QoS FIFO level. If FIFO depth is less than FIFOLEVEL[7:0], then RTQoS output is 1.	0
-	[3:2]	Reserved (should be 0)	0
QOS_GATE_DIS	[1]	Disables the RTQoS output signal gate. 0 = Gated 1 = Not gated	0
-	[0]	Reserved (should be 0).	0

1.5.2.85 Write-Back Control Registers

WRITEBACK	Bit	Description	Initial State
WRITEBACK_EN	[31]	Write-Back enable. 0 : disable 1 : enable	0

1.5.2.86 LCD I80 Interface Command (I80IFCONx)

- LDI_CMD0, R/W, Address = 0xF800_0280
- LDI_CMD1, R/W, Address = 0xF800_0284
- LDI_CMD2, R/W, Address = 0xF800_0288
- LDI_CMD3, R/W, Address = 0xF800_028C
- LDI_CMD4, R/W, Address = 0xF800_0290
- LDI_CMD5, R/W, Address = 0xF800_0294
- LDI_CMD6, R/W, Address = 0xF800_0298
- LDI_CMD7, R/W, Address = 0xF800_029C
- LDI_CMD8, R/W, Address = 0xF800_02A0
- LDI_CMD9, R/W, Address = 0xF800_02A4
- LDI_CMD10, R/W, Address = 0xF800_02A8
- LDI_CMD11, R/W, Address = 0xF800_02AC

I80IFCONx	Bit	Description	Initial State
LDI_CMD	[23:0]	Specifies the LDI command.	0

1.5.2.87 Win0 Palette Ram Access Address (not SFR)

Register	Address	R/W	Description	Initial State
00	0xF800_2400 (0xF800_0400)	R/W	Specifies the Window 0 Palette entry 0 address.	undefined
01	0xF800_2404 (0xF800_0404)	R/W	Specifies the Window 0 Palette entry 1 address.	undefined
-	-	-	-	-
FF	0xF800_27FC (0xF800_07FC)	R/W	Specifies the Window 0 Palette entry 255 address.	undefined

1.5.2.88 Win1 Palette Ram Access Address (not SFR)

Register	Address	R/W	Description	Initial State
00	0xF800_2800 (0xF800_0800)	R/W	Specifies the Window 1 Palette entry 0 address.	undefined
01	0xF800_2804 (0xF800_0804)	R/W	Specifies the Window 1 Palette entry 1 address.	undefined
-	-	-	-	-
FF	0xF800_2BFC (0xF800_0BFC)	R/W	Specifies the Window 1 Palette entry 255 address.	undefined

1.5.2.89 Win2 Palette Ram Access Address (not SFR)

Register	Address	R/W	Description	Initial State
00	0xF800_2C00	R/W	Specifies the Window 2 Palette entry 0 address.	undefined
01	0xF800_2C04	R/W	Specifies the Window 2 Palette entry 1 address.	undefined
-	-	-	-	-
FF	0xF800_2FFC	R/W	Specifies the Window 2 Palette entry 255 address.	undefined

1.5.2.90 Win3 Palette Ram Access Address (not SFR)

Register	Address	R/W	Description	Initial State
00	0xF800_3000	R/W	Specifies the Window 3 Palette entry 0 address.	undefined
01	0xF800_3004	R/W	Specifies the Window 3 Palette entry 1 address.	undefined
-	-	-	-	-
FF	0xF800_33FC	R/W	Specifies the Window 3 Palette entry 255 address.	undefined

1.5.2.91 Win4 Palette Ram Access Address (not SFR)

Register	Address	R/W	Description	Reset Value
00	0xF800_3400	R/W	Specifies the Window 4 Palette entry 0 address.	undefined
01	0xF800_3404	R/W	Specifies the Window 4 Palette entry 1 address.	undefined
-	-	-	-	-
FF	0xF800_37FC	R/W	Specifies the Window 4 Palette entry 255 address.	undefined

1.5.2.92 Window RTQOS Control Registers

- W0RTQOSCON, R/W, Address = 0xF800_0264
- W1RTQOSCON, R/W, Address = 0xF800_0268
- W2RTQOSCON, R/W, Address = 0xF800_026C
- W3RTQOSCON, R/W, Address = 0xF800_0270
- W4RTQOSCON, R/W, Address = 0xF800_0274

HUECOEF0x	Bit	Description	Initial State
-	[31:12]	Reserved (should be 0).	0
FIFOLEVEL	[11:4]	Specifies the real-time QoS FIFO level. If FIFO depth is less than FIFOLEVEL[7:0], the RTQoS output is 1.	0
-	[3:2]	Reserved (should be 0).	0
QOS_GATE_DIS	[1]	Disables the RTQoS output signal gate. 0 = Gated 1 = Not gated	0
-	[0]	Reserved (should be 0).	0

1.5.3 GAMMA LUT DATA

Register	Address	R/W	Description	Reset Value
GAMMALUT_1_0	0X0037C	R/W	Specifies the Gamma LUT data of the index 0, 1.	0X0010_0000
GAMMALUT_3_2	0X00380	R/W	Specifies the Gamma LUT data of the index 2, 3.	0X0030_0020
GAMMALUT_5_4	0X00384	R/W	Specifies the Gamma LUT data of the index 4, 5.	0X0050_0040
GAMMALUT_7_6	0X00388	R/W	Specifies the Gamma LUT data of the index 6, 7.	0X0070_0060
GAMMALUT_9_8	0X0038C	R/W	Specifies the Gamma LUT data of the index 8, 9.	0X0090_0080
GAMMALUT_11_10	0X00390	R/W	Specifies the Gamma LUT data of the index 10, 11.	0X00B0_00A0
GAMMALUT_13_12	0X00394	R/W	Specifies the Gamma LUT data of the index 12, 13.	0X00D0_00C0
GAMMALUT_15_14	0X00398	R/W	Specifies the Gamma LUT data of the index 14, 15.	0X00F0_00E0
GAMMALUT_17_16	0X0039C	R/W	Specifies the Gamma LUT data of the index 16, 17.	0X0110_0100
GAMMALUT_19_18	0X003A0	R/W	Specifies the Gamma LUT data of the index 18, 19.	0X0130_0120
GAMMALUT_21_20	0X003A4	R/W	Specifies the Gamma LUT data of the index 20, 21.	0X0150_0140
GAMMALUT_23_22	0X003A8	R/W	Specifies the Gamma LUT data of the index 22, 23.	0X0170_0160
GAMMALUT_25_24	0X003AC	R/W	Specifies the Gamma LUT data of the index 24, 25.	0X0190_0180
GAMMALUT_27_26	0X003B0	R/W	Specifies the Gamma LUT data of the index 26, 27.	0X01B0_01A0
GAMMALUT_29_28	0X003B4	R/W	Specifies the Gamma LUT data of the index 28, 29.	0X01D0_01C0
GAMMALUT_31_30	0X003B8	R/W	Specifies the Gamma LUT data of the index 30, 31.	0X01F0_01E0
GAMMALUT_33_32	0X003BC	R/W	Specifies the Gamma LUT data of the index 32, 33.	0X0210_0200
GAMMALUT_35_34	0X003C0	R/W	Specifies the Gamma LUT data of the index 34, 35.	0X0230_0220
GAMMALUT_37_36	0X003C4	R/W	Specifies the Gamma LUT data of the index 36, 37.	0X0250_0240
GAMMALUT_39_38	0X003C8	R/W	Specifies the Gamma LUT data of the index 38, 39.	0X0270_0260
GAMMALUT_41_40	0X003CC	R/W	Specifies the Gamma LUT data of the index 40, 41.	0X0290_0280
GAMMALUT_43_42	0X003D0	R/W	Specifies the Gamma LUT data of the index 42, 43.	0X02B0_02A0
GAMMALUT_45_44	0X003D4	R/W	Specifies the Gamma LUT data of the index 44, 45.	0X02D0_02C0
GAMMALUT_47_46	0X003D8	R/W	Specifies the Gamma LUT data of the index 46, 47.	0X02F0_02E0
GAMMALUT_49_48	0X003DC	R/W	Specifies the Gamma LUT data of the index 48, 49.	0X0310_0300
GAMMALUT_51_50	0X003E0	R/W	Specifies the Gamma LUT data of the index 50, 51.	0X0330_0320
GAMMALUT_53_52	0X003E4	R/W	Specifies the Gamma LUT data of the index 52, 53.	0X0350_0340
GAMMALUT_55_54	0X003E8	R/W	Specifies the Gamma LUT data of the index 54, 55.	0X0370_0360
GAMMALUT_57_56	0X003EC	R/W	Specifies the Gamma LUT data of the index 56, 57.	0X0390_0380
GAMMALUT_59_58	0X003F0	R/W	Specifies the Gamma LUT data of the index 58, 59.	0X03B0_03A0
GAMMALUT_61_60	0X003F4	R/W	Specifies the Gamma LUT data of the index 60, 61.	0X03D0_03C0
GAMMALUT_63_62	0X003F8	R/W	Specifies the Gamma LUT data of the index 62, 63.	0X03F0_03E0
GAMMALUT_xx_64	0X003FC	R/W	Specifies the Gamma LUT data of the index 64.	0X0000_0400

GAMMALUT_x_y	Bit	Description	Initial State
GM_LUT_x	[26:18]	Specifies the Gamma LUT value register of index x.	
GM_LUT_y	[10: 2]	Specifies the Gamma LUT value register of index y.	

1.5.3.1 Frame Buffer Address 0 Shadow Registers

- SHD_VIDW00ADD0, R/W, Address = 0xF800_40A0
- SHD_VIDW01ADD0, R/W, Address = 0xF800_40A8
- SHD_VIDW02ADD0, R/W, Address = 0xF800_40B0
- SHD_VIDW03ADD0, R/W, Address = 0xF800_40B8
- SHD_VIDW04ADD0, R/W, Address = 0xF800_40C0

SHD_VIDWxxADD0	Bit	Description	Initial State
VBANK_F	[31:24]	Specifies A[31:24] of the bank location for video buffer in the system memory (Shadow).	0
VBASEU_F	[23:0]	Specifies A[23:0] of the start address for video frame buffer (Shadow).	0

1.5.3.2 Frame Buffer Address 1 Shadow Registers

- SHD_VIDW00ADD1, R/W, Address = 0xF800_40D0
- SHD_VIDW01ADD1, R/W, Address = 0xF800_40D8
- SHD_VIDW02ADD1, R/W, Address = 0xF800_40D0
- SHD_VIDW03ADD1, R/W, Address = 0xF800_40D8
- SHD_VIDW04ADD1, R/W, Address = 0xF800_40D0

SHD_VIDWxxADD1	Bit	Description	Initial State
VBASEL_F	[23:0]	Specifies A[23:0] of the end address for video buffer (Shadow).	0x0

1.5.3.3 Frame Buffer Address 2 Shadow Registers

- SHD_VIDW00ADD2, R/W, Address = 0xF800_40A0
- SHD_VIDW01ADD2, R/W, Address = 0xF800_40A8
- SHD_VIDW02ADD2, R/W, Address = 0xF800_40B0
- SHD_VIDW03ADD2, R/W, Address = 0xF800_40B8
- SHD_VIDW04ADD2, R/W, Address = 0xF800_40C0

SHD_VIDWxxADD2	Bit	Description	Initial State
OFFSIZE_F	[25:13]	Specifies the Virtual screen offset size that is the number of byte (Shadow).	0
PAGEWIDTH_F	[12:0]	Specifies the Virtual screen page width (number of byte). This value defines the width of view port in the frame (Shadow).	0

2 CAMERA INTERFACE

2.1 OVERVIEW OF CAMERA INTERFACE

The camera interface (CAMIF) in S5PV210 is a fully interactive mobile camera interface. CAMIF supports ITU R BT-601/656 standard, AXI interface and MIPI (CSI). The maximum input image size of CAMIF is 8192 x 8192 pixels.

S5PV210 has three CAMIF units, namely, CAMIF0, CAMIF1 and CAMIF2, as shown in [Figure 2-1](#).

Each of these units is designed to perform different functions, as shown in [Figure 2-2](#).

- T_PatternMux generates test pattern to calibrate input sync signals as HREF and VSYNC.
- Capture specifies the capturing signal and window cut. Use the register settings to invert video sync signals and pixel clock polarity in CAMIF.
- Scaler generates various sizes for an image.
- Input DMA (read only) reads from the memory image data.
- Output DMA (write only) writes image data to memory.
- CAMIF supports image rotation (90 degrees clockwise) and image effect functions.

The key application of these features is in a folder-type cellular phone.

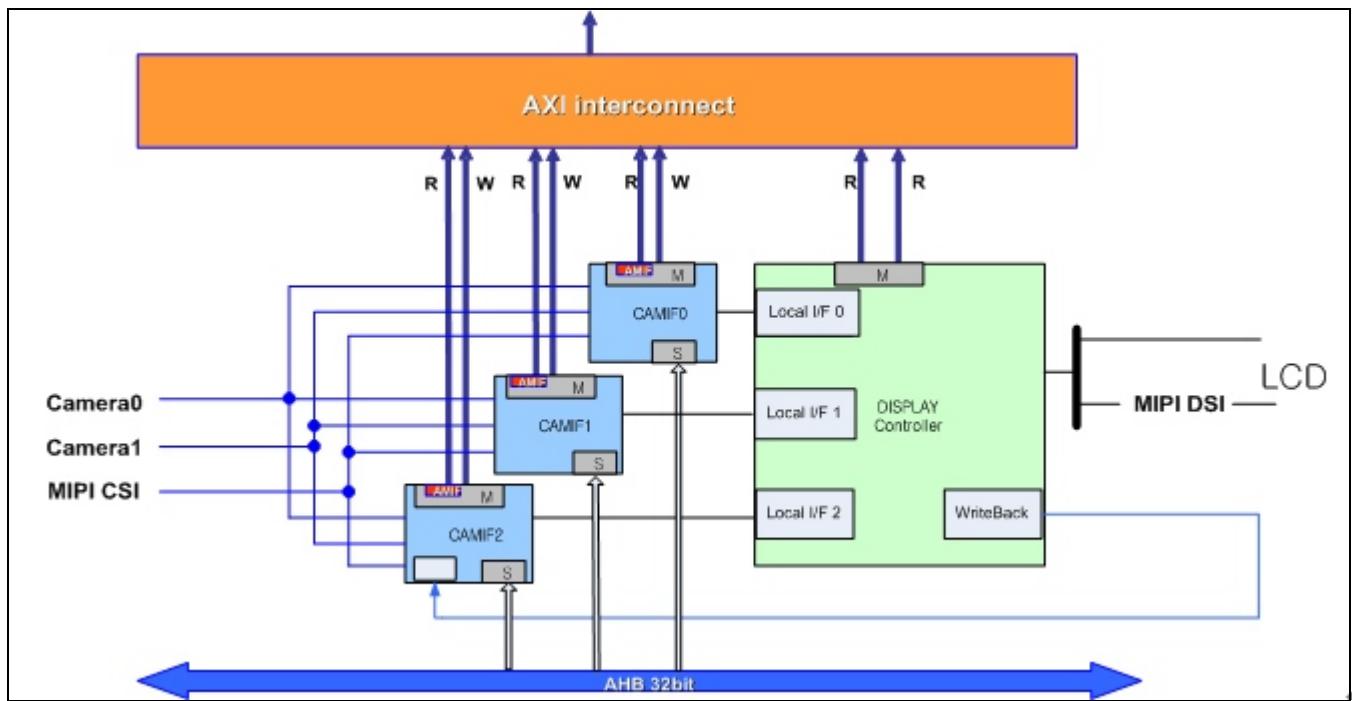


Figure 2-1 Subset of Visual System in S5PV210

2.2 KEY FEATURES OF CAMIF

The key features of CAMIF include:

- Multiple input support
 - ITU-R BT 601/ 656/ 709 mode
 - DMA (AXI 64-bit interface) mode
 - MIPI (CSI) mode
 - Direct FIFO (PlayBack) mode
- Multiple output support
 - DMA (AXI 64-bit interface) mode
 - Direct FIFO mode
- Digital Zoom In (DZI) capability
- Multiple camera input
- Programmable polarity of video sync signals
- Supports maximum 8192 x 8192 pixels input (Refer to [Table 2-1](#))
- Image mirror and rotation (X-axis mirror, Y-axis mirror, 90°, 180°, and 270° rotation)
- Generates various image formats
- Supports capture frame control
- Supports image effect

Table 2-1 Maximum Size

	Item	Maximum Size		
		CAMIF0	CAMIF1	CAMIF2
Scaler	Scaler input Hsize (=PreDstWidth)	4224 pixels	4224 pixels	1920 pixels
	Scaler bypass mode	8192 pixels	8192 pixels	8192 pixels
Output Rotator	TargetHsize (without output rotation)	4224 pixels	4224 pixels	1920 pixels
	TargetHsize (with output rotation)	1920 pixels	1920 pixels	1280 pixels
Input Rotator	REAL_WIDTH (without input rotation)	8192 pixels	8192 pixels	8192 pixels
	REAL_HEIGHT (with input rotation)	1920 pixels	1920 pixels	1280 pixels

NOTE: The maximum size of Scaler and Rotator depends on the line buffer size.

The maximum size of the Output Rotator and Input Rotator is less than the maximum size of Scaler.

Note: Minimum input size : 32 x 32

Note: Minimum output size : 32 x 32 (normal) , 128 x 128 (output rotation and interlace out are enable)

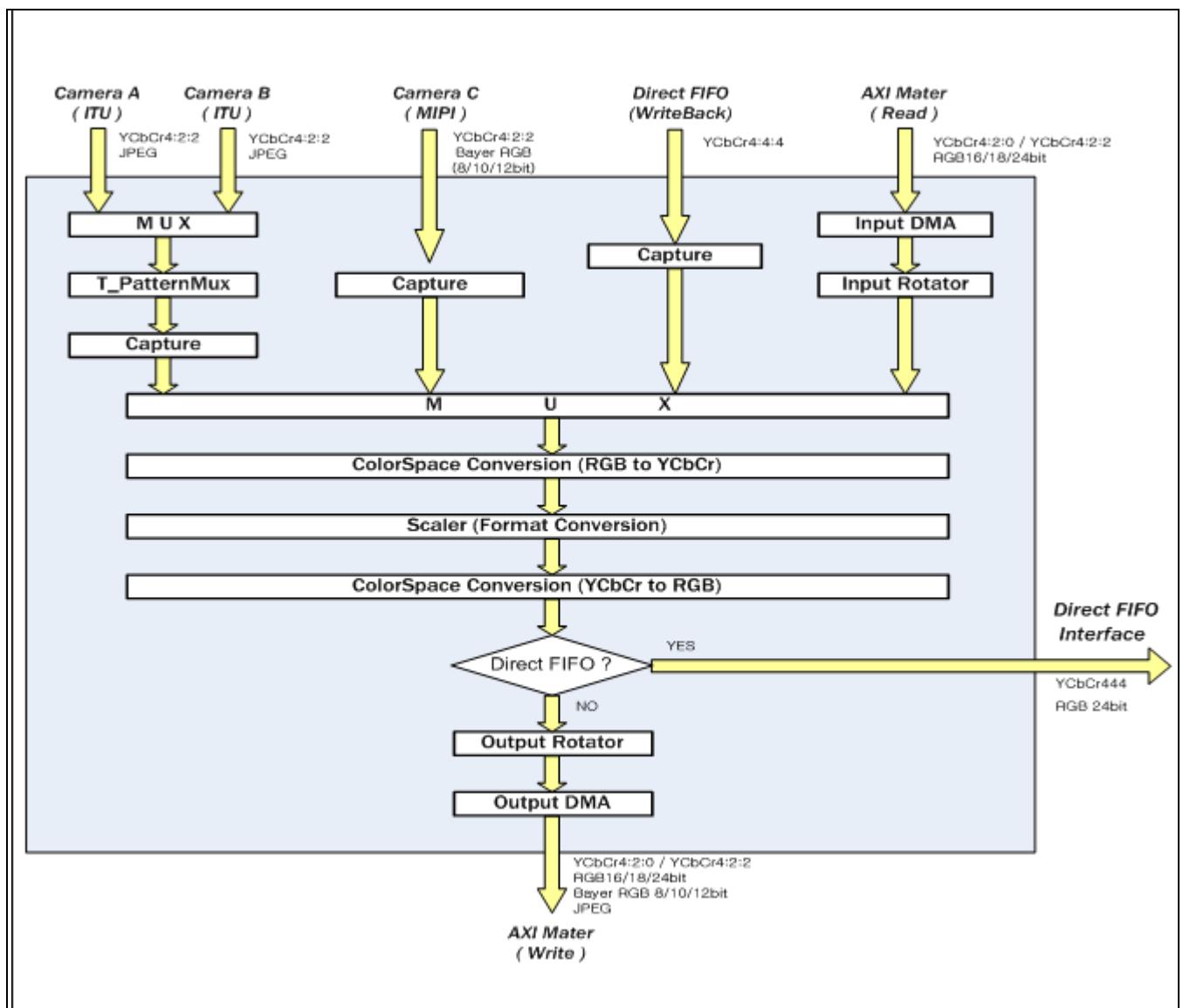


Figure 2-2 Camera Interface Overview

NOTE: In case of Direct FIFO (WriteBack) input mode, CAMIF does not support cropping, capture frame control, test pattern, and scaler bypass function.

2.3 EXTERNAL INTERFACE

CAMIF supports three video standards, namely:

- ITU-R BT 601 YCbCr 8-bit mode
- ITU-R BT 656 YCbCr 8-bit mode
- MIPI mode

Samsung Confidential

2.4 INPUT/ OUTPUT DESCRIPTION

Table 2-2 ITU Camera Interface Signal Description

Signal	I/O	Description	PAD	Type
External ITU Camera Processor A Interface Signal				
PCLK_A	I	Specifies the Pixel Clock driven by external Camera processor A.	XciPCLK	Muxed
VSYNC_A	I	Specifies the Frame Sync driven by external Camera processor A.	XciVSYNC	Muxed
HREF_A	I	Specifies the Horizontal Sync driven by external Camera processor A.	XciHREF	Muxed
DATA_A[7:0]	I	Specifies the Pixel Data driven by external Camera processor A.	XciDATA [7:0]	Muxed
FIELD_A	I	Specifies the Field signal driven by external Camera processor A.	XciFIELD	Muxed
CAM_MCLK_A	O	Specifies the Clock for external Camera processor A.	XciCLKenb	Muxed
External ITU Camera Processor B Interface Signal				
PCLK_B	I	Specifies the Pixel Clock driven by external Camera processor B.	XmsmADDR[8]	Muxed
VSYNC_B	I	Specifies the Frame Sync driven by external Camera processor B.	XmsmADDR[9]	Muxed
HREF_B	I	Specifies the Horizontal Sync driven by external Camera processor B.	XmsmADDR[10]	Muxed
DATA_B[7:0]	I	Specifies the Pixel Data driven by external Camera processor B.	XmsmADDR[7:0]	Muxed
FIELD_B	I	Specifies the Field signal driven by external Camera processor B.	XmsmADDR[11]	Muxed
CAM_MCLK_B	O	Specifies the Clock for external Camera processor B.	XmsmADDR[12]	Muxed

NOTE: I/O direction. I: input, O: output, and B: bi-direction.

Type field indicates whether pads are dedicated to the signal or pads are connected to the multiplexed signals.

2.5 TIMING DIAGRAM AND DATA ALIGNMENT OF CAMERA

2.5.1 TIMING DIAGRAM OF ITU CAMERA

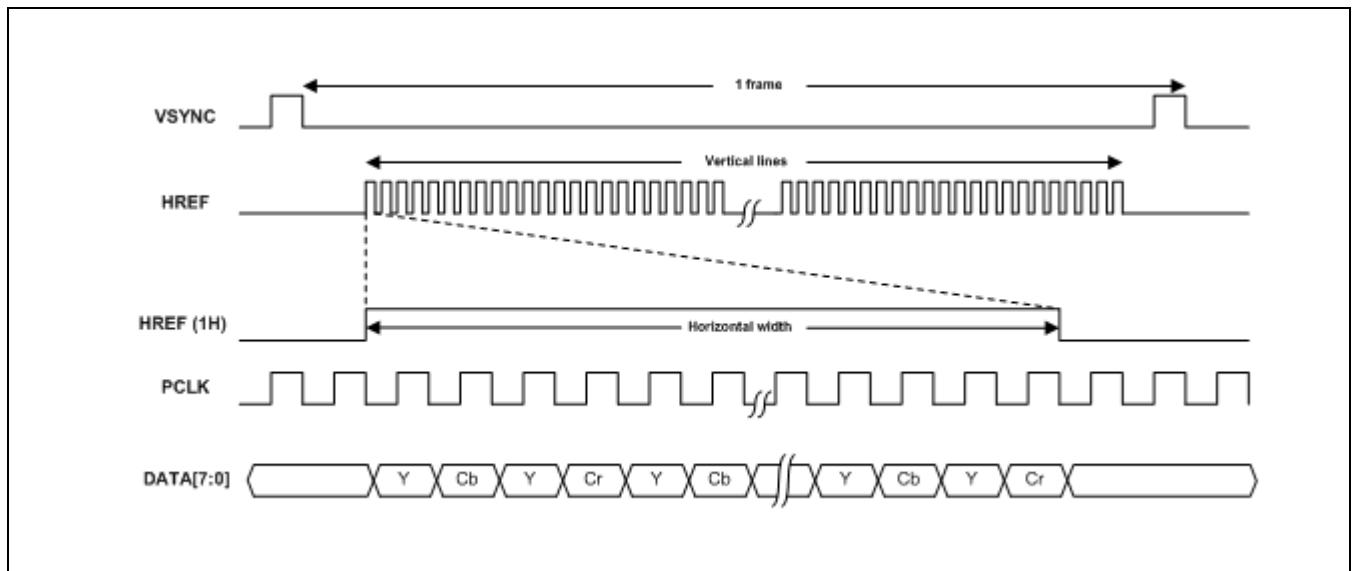


Figure 2-3 ITU-R BT 601 Input Timing Diagram

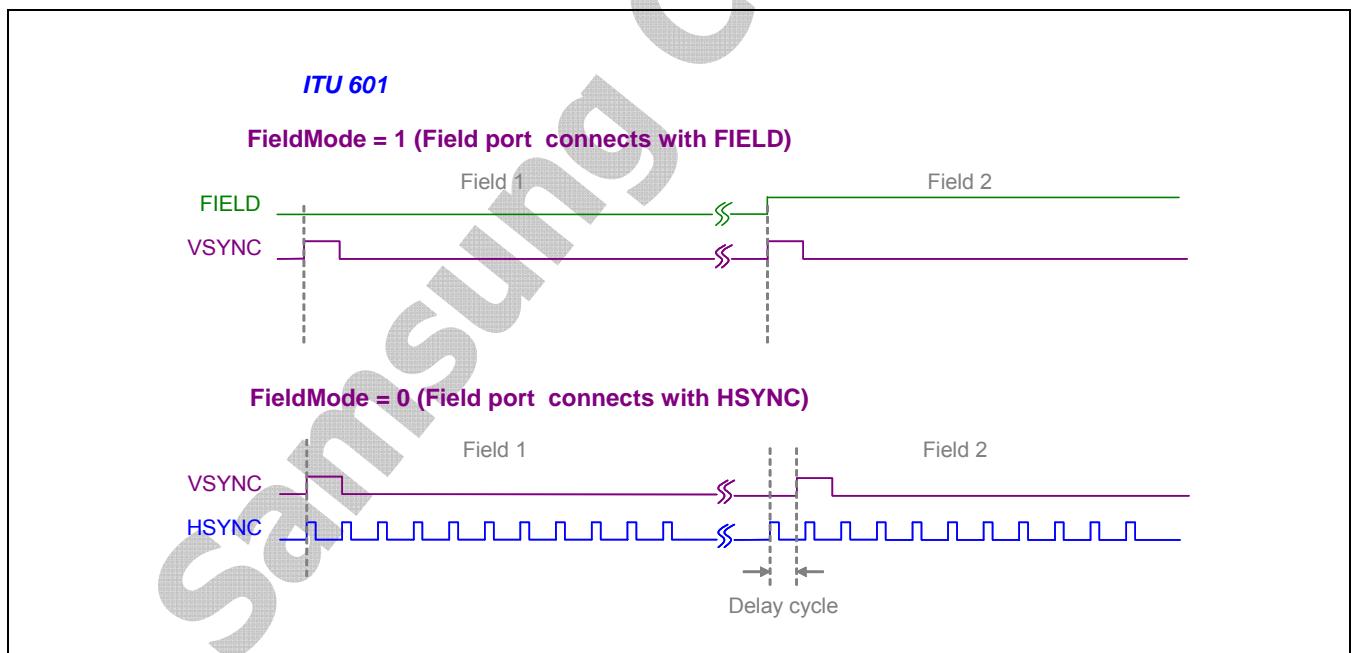


Figure 2-4 ITU-R BT 601 Interlace Handling Diagram

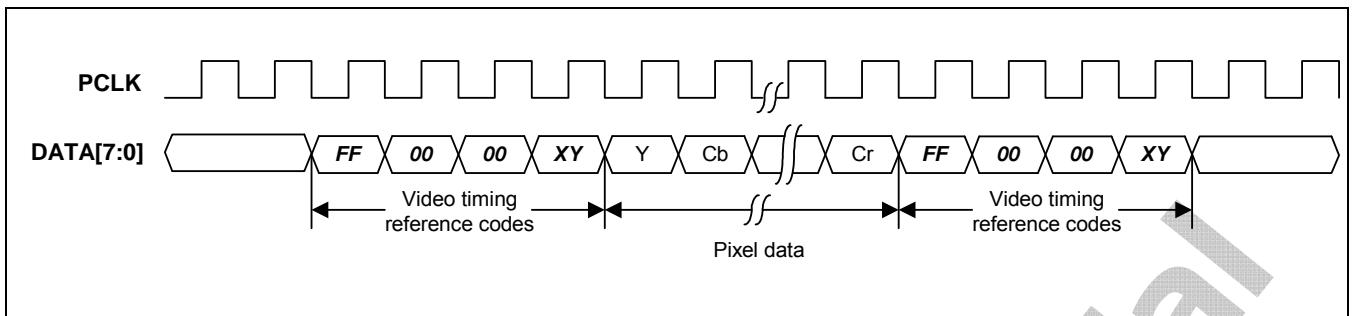


Figure 2-5 ITU-R BT 656 Input Timing Diagram

There are two timing reference signals in ITU-R BT 656 format: one at the beginning of each video data block (start of active video, SAV) and other at the end of each video data block (end of active video, EAV), as shown in [Figure 2-5](#) and [Table 2-3](#).

Table 2-3 Video Timing Reference Codes of ITU-656 8-Bit Format

Data Bit Number	First Word	Second Word	Third Word	Fourth Word
7 (MSB)	1	0	0	1
6	1	0	0	F
5	1	0	0	V
4	1	0	0	H
3	1	0	0	P3
2	1	0	0	P2
1	1	0	0	P1
0	1	0	0	P0

NOTE: F = 0 (during field 1), 1 (during field 2)

V = 0 (elsewhere), 1 (during field blanking)

H = 0 (in SAV: Start of Active Video), 1 (in EAV: End of Active Video)

P0, P1, P2, P3 = Protection Bit

The camera interface logic catches video sync bits like H (SAV, EAV) and V (Frame Sync) after reserving data as "FF-00-00".

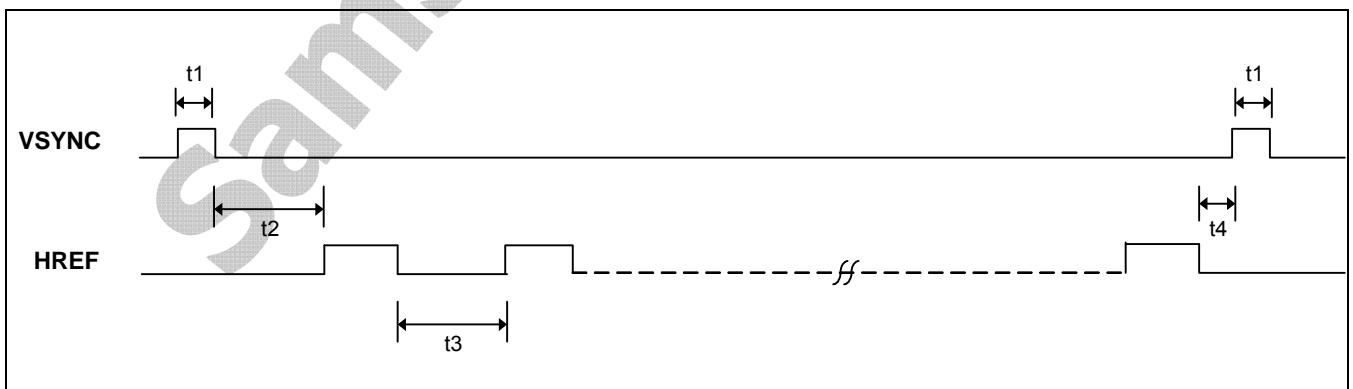
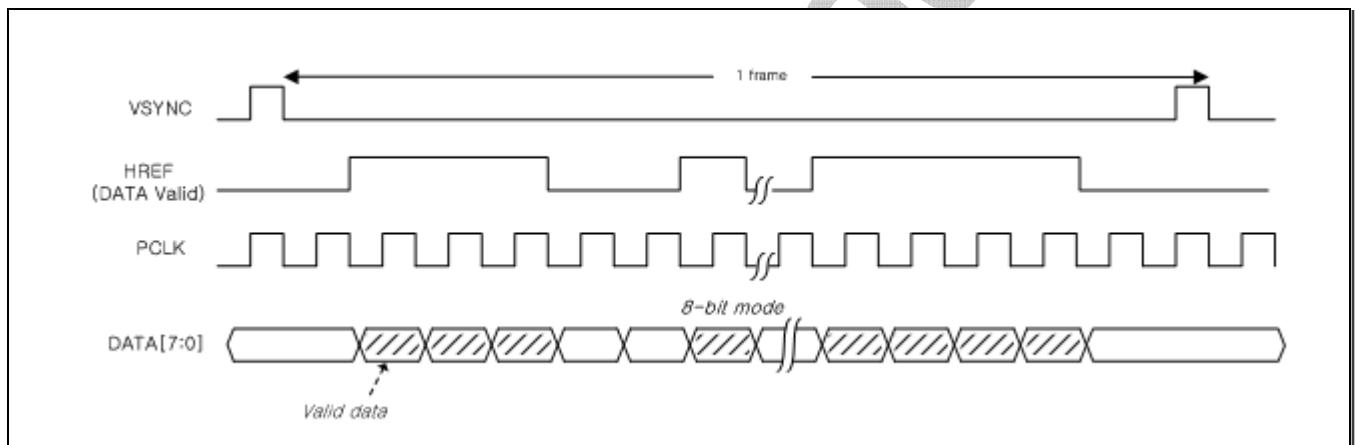


Figure 2-6 Sync Signal Timing Diagram**Table 2-4 Sync Signal Timing Requirements**

	Minimum	Maximum
t1	2 Horizontal Line	-
t2	2 Cycles of Pixel Clock + 5 Cycles of System Bus Clock	-
t3	2 Cycles of Pixel Clock	-
t4	12 Cycles of Pixel Clock	-

NOTE: If rotator is enabled, then ($t_4 + t_1$) should be sufficient to finish DMA transactions, since DMA transactions for rotator line buffer are delayed by 4 or 8 horizontal lines.

**Figure 2-7 JPEG Input Timing Diagram (ITU 601 and Freerun Clock Mode)**

2.5.2 MIPI CSI DATA ALIGNMENT FROM MIPI CAMERA

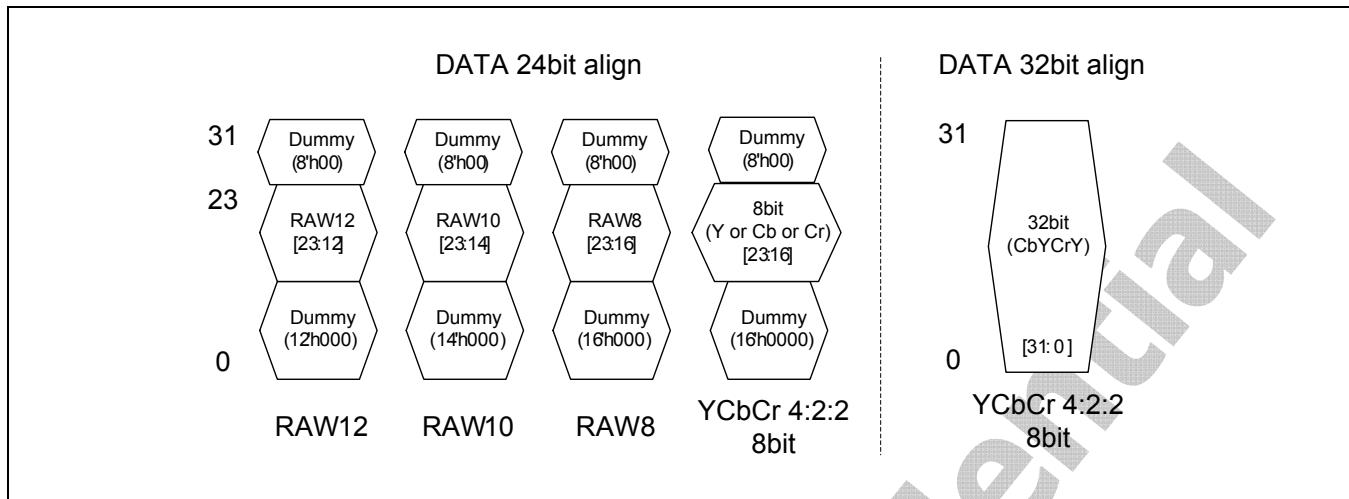


Figure 2-8 MIPI CSI DATA Alignment

Table 2-5 DATA Order of YCbCr422 Align

Format	Stream Order of Content	DATA – 24-bit Align	DATA – 32-bit Align
YCbCr422	Cb1→Y1→Cr1→Y2→ ...	DATA1[23:16] = Cb1 DATA2[23:16] = Y1 DATA3[23:16] = Cr1 DATA4[23:16] = Y2	DATA1[31:24] = Cb1 DATA1[23:16] = Y1 DATA1[15:8] = Cr1 DATA1[7:0] = Y2

2.6 EXTERNAL CONNECTION GUIDE

The CAMIF input signals must not result in inter-skewing of the pixel clock line. Therefore, it is recommended to use next pin location and routing.

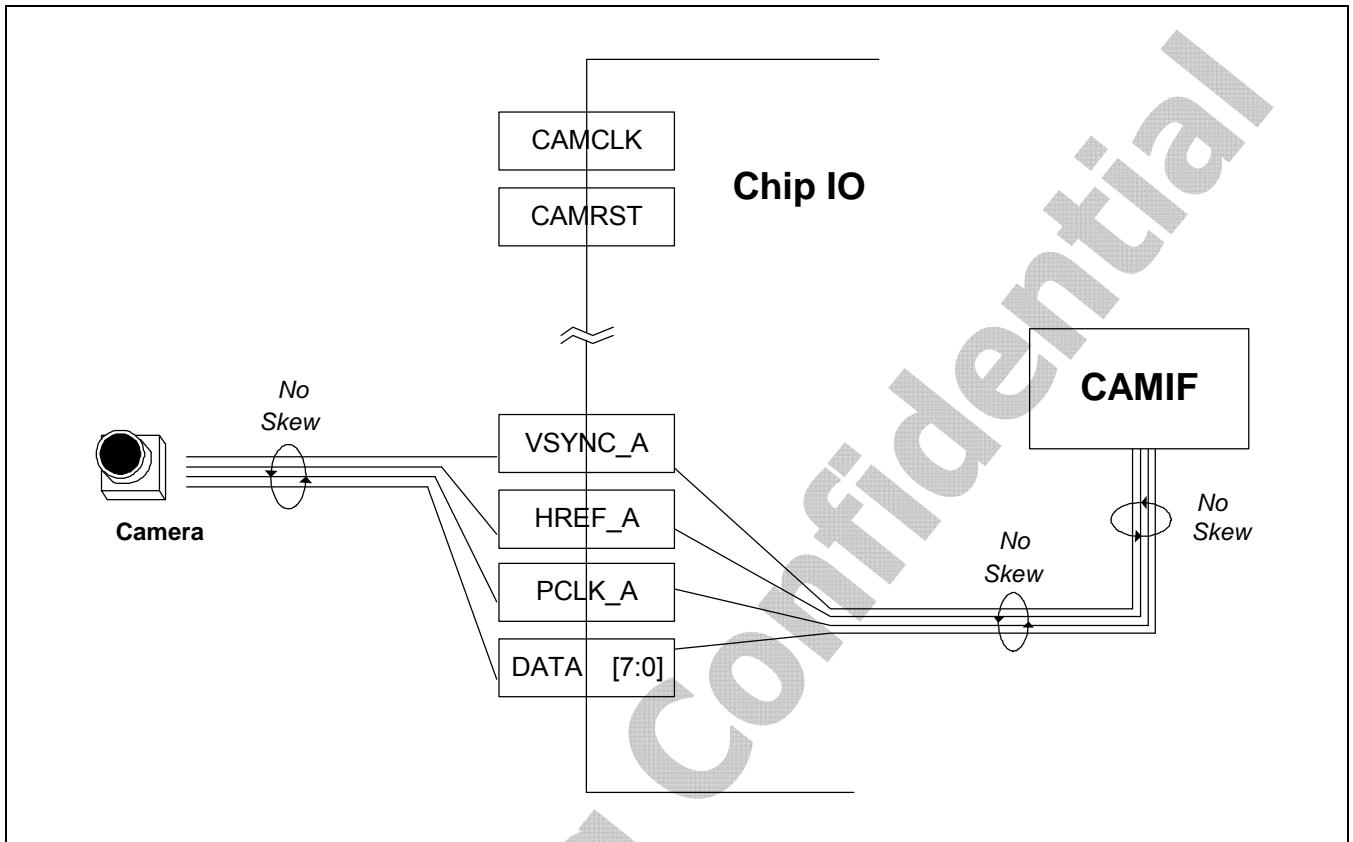


Figure 2-9 IO Connection Guide

2.7 CAMERA INTERFACE OPERATION

2.7.1 INPUT/ OUTPUT DMA PORTS

Each CAMIF consists of two DMA ports, namely, Input DMA Port and Output DMA Port. From the view of system bus, both the ports are independent. The Input DMA port reads the image data from memory. On the other hand, the Output DMA port stores the image data into memory. These two master ports support various digital applications such as Digital Still Camera (DSC), MPEG-4 video conference, video recording, and so on.

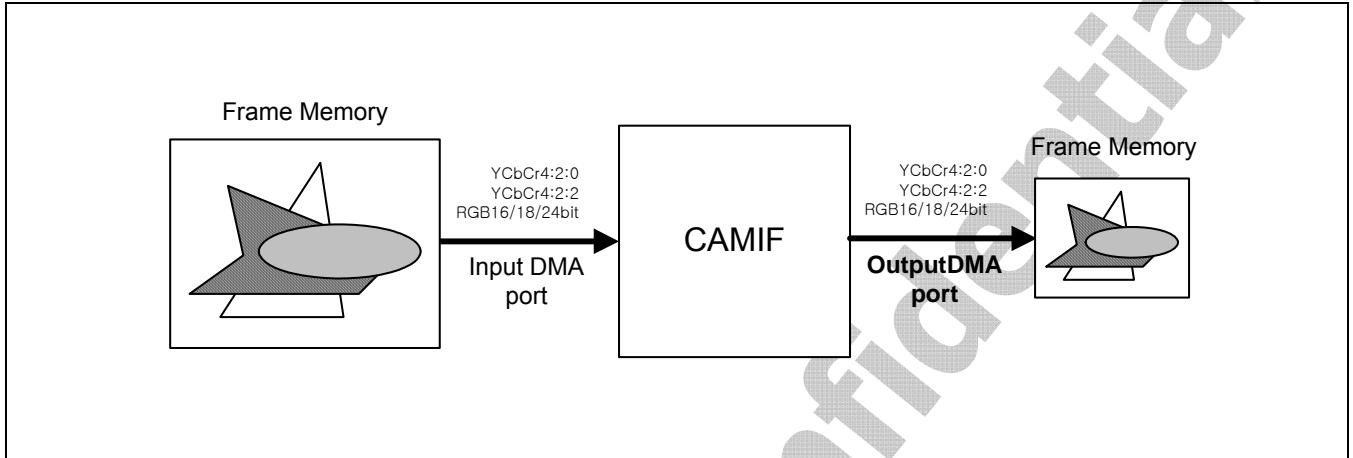


Figure 2-10 Input / Output DMA Ports

2.7.2 CLOCK DOMAIN

Each CAMIF consists of three clock domains. The first clock domain is the system bus clock. The second clock domain is the camera pixel clock, PCLK. The third clock domain is the internal core clock. The system bus clock must be faster than the camera pixel clock.

As shown in [Figure 2-11](#), CAM_MCLK must be separated from the fixed frequency like PLL clock. If external clock oscillator is used, CAM_MCLK should be floated. It is not necessary for the three clock domains to synchronize. Other signals like PCLK should similarly be connected to Schmitt-triggered level shifter.

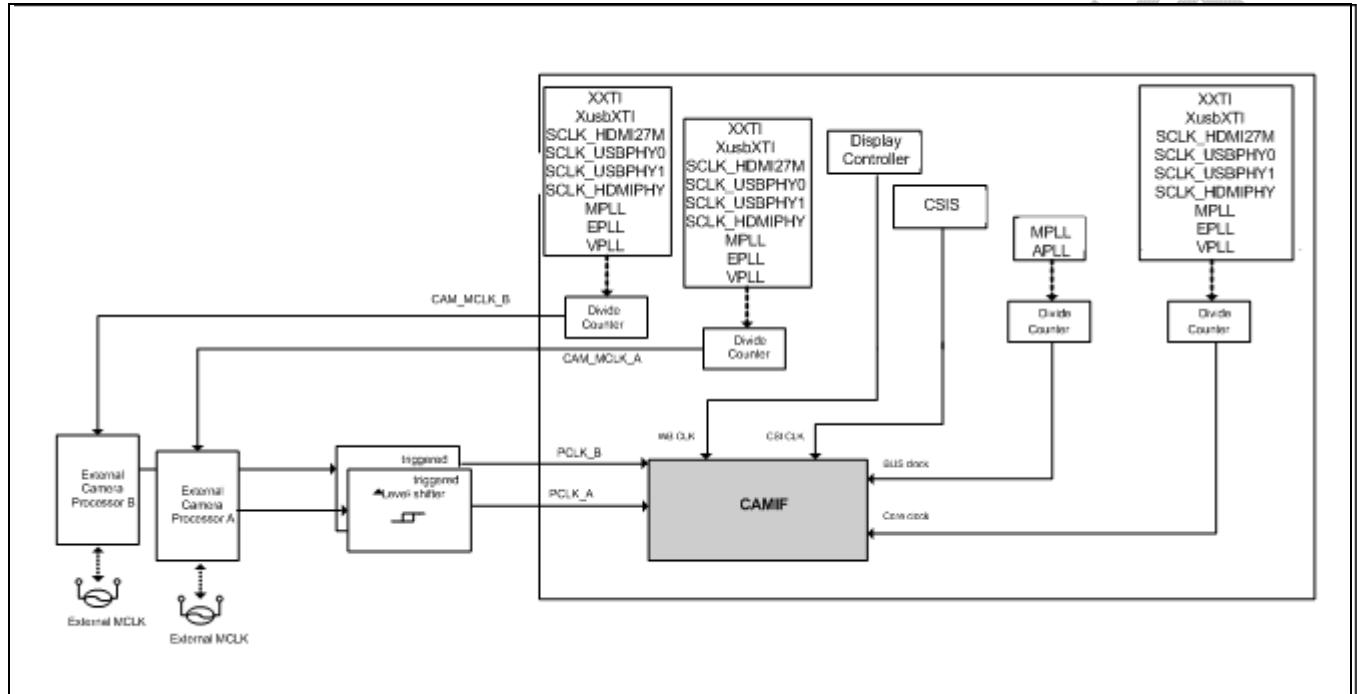


Figure 2-11 CAMIF Clock Generation

NOTE: The maximum frequency of core clock depends on whether user uses a local path between CAMIF and display controller or not. When it comes to the local path, the maximum frequency of core clock is 133MHz(It is recommended to use MPPLL as the source of core clock). For other operation modes except for the local path, the maximum frequency of core clock is 166MHz(BUS clock should be used).

The maximum frequency of both CAM_MCLK_A and CAM_MCLK_B is 100MHz. And the maximum frequency of both PCLK_A and PCLK_B is 83MHz.

2.7.3 FRAME MEMORY HIERARCHY

Frame memory consists of four ping-pong memories for output DMA ports. Also, the ping-pong memory consists of three element memories, namely, luminance Y, chrominance Cb, and chrominance Cr. It is recommended that the arbitration priority of CAMIF must be higher than any other masters (except for LCD controller). It is highly recommended to set the CAMIF priorities as fixed priorities, not rotation priorities.

In case of multi-AHB bus, the priority of system bus (including CAMIF) must be higher than others. If the bus traffic is so heavy that a DMA operation cannot complete for one horizontal period plus blank, it might result in malfunctioning. Therefore, the priority of CAMIF must be changed to other round robin or circular arbitration priorities. It is recommended that the bus including CAMIF should have a higher priority than any other buses in the memory matrix system. The CAMIF should not be the default master of AMBA system.

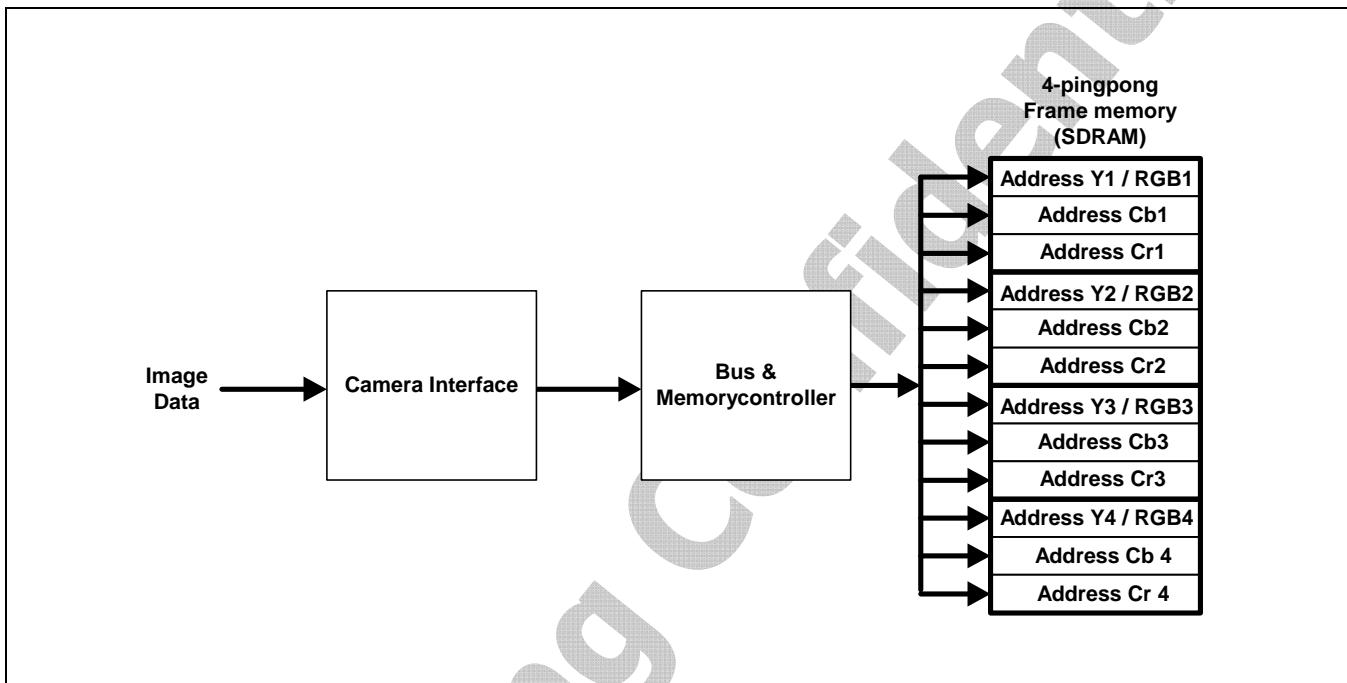


Figure 2-12 Ping-pong Memory Hierarchy

2.7.4 MEMORY STORING METHOD

The storing order of frame memory is little-endian. The first entering pixel is stored in the LSB sides, while the last entering pixel is stored in the MSB sides. The carried data by AXI bus is 64-bit. Therefore, CAMIF makes each Y-Cb-Cr words using little endian style. End-of-horizontal line not aligned with 64-bit is padded with zero for RAW8, RAW10 and RAW12 format. For more information, refer to [Figure 2-13](#).

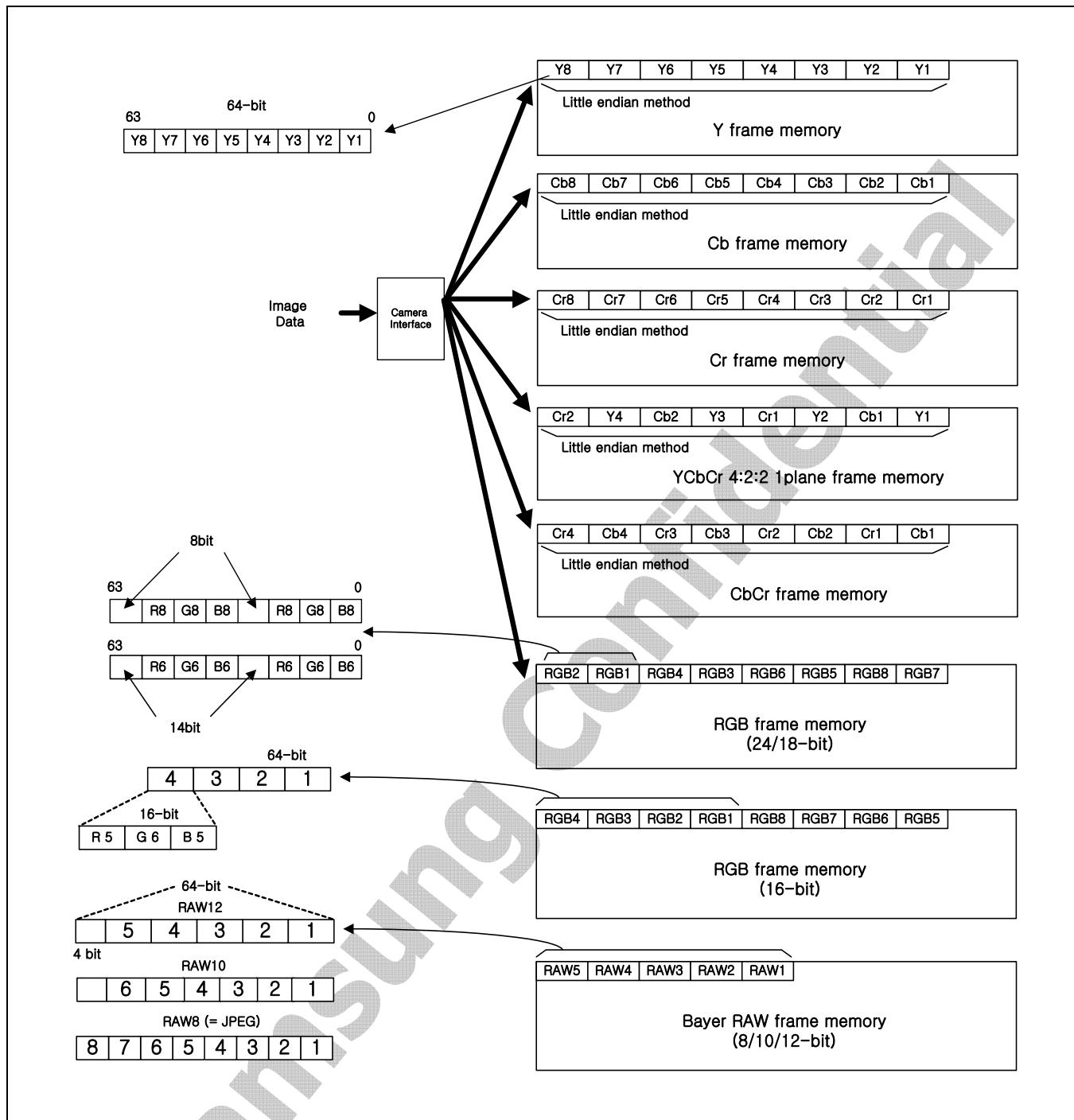


Figure 2-13 Memory Storing Style

2.7.5 TIMING DIAGRAM FOR REGISTER SETTING

The first register setting for frame capture command can happen anytime in the frame period. It is recommended to first set the VSYNC "L" state, input DMA start "L" state, and then VVALID "H" state. VSYNC and VVALID

information can be read from status SFR. For more information, refer to [Figure 2-14](#).

All commands (including ImgCptEn) are valid at VSYNC falling edge or VVALID rising edge. Except for the first SFR setting, all commands should be programmed in the Interrupt Service Routine (ISR). Size, image mirror or rotation, windowing, and zoom in settings are allowed to change during capturing. In case of DMA input mode, all commands should be programmed after the InputDMA and OutputDMA operation end, as shown in [Figure 2-15](#).

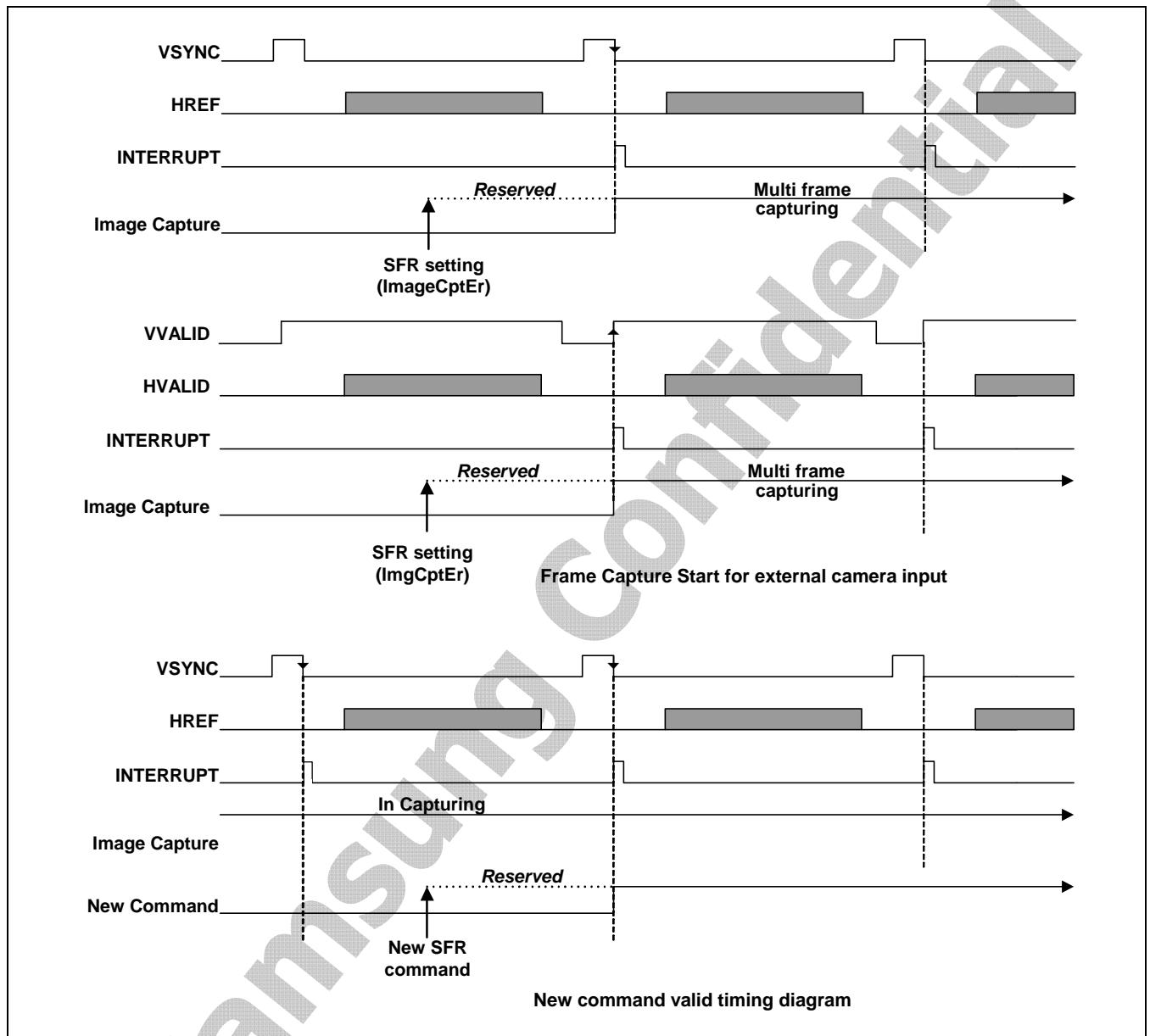


Figure 2-14 Timing Diagram for Camera Input Register setting

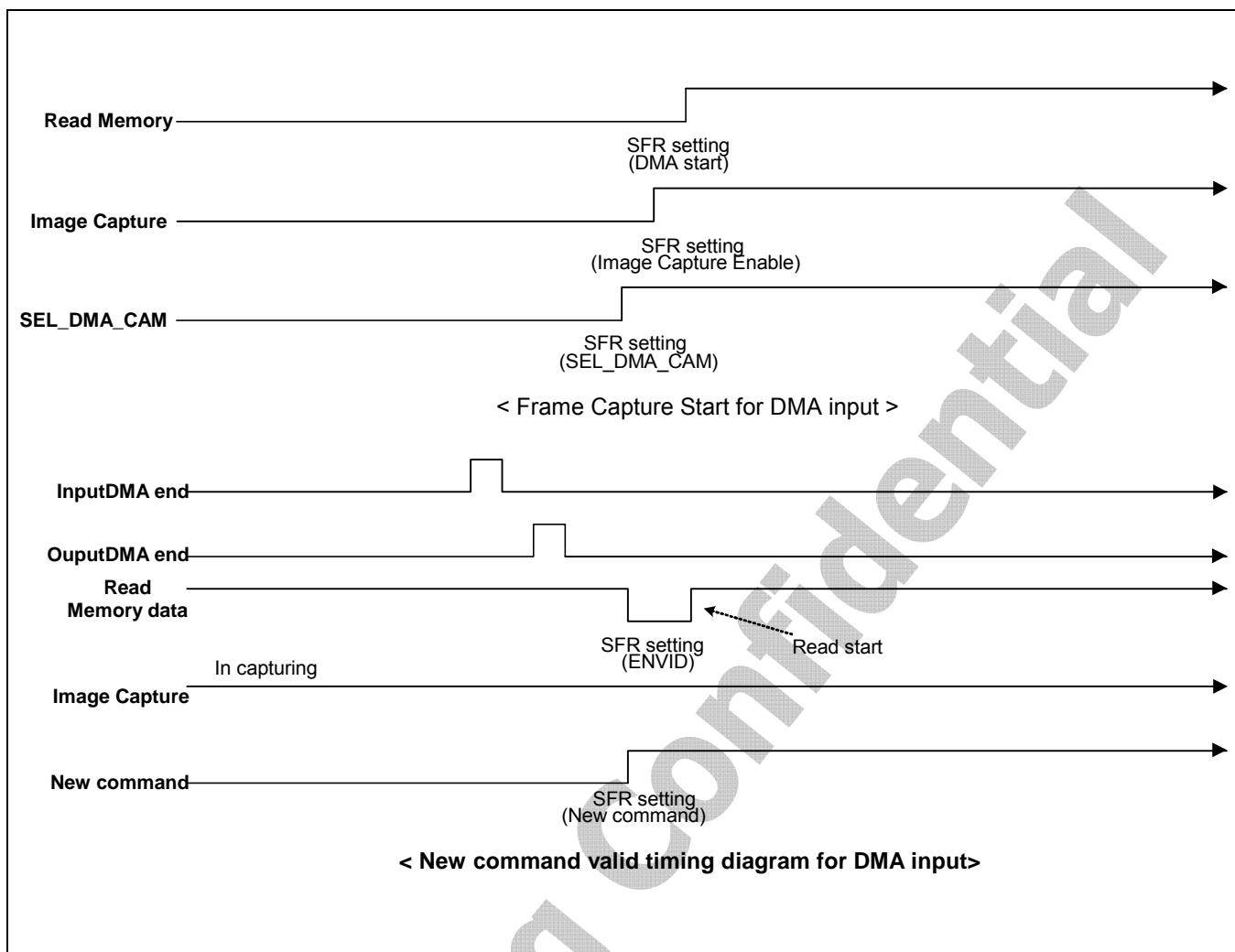


Figure 2-15 Timing Diagram for DMA input Register Setting

2.7.6 TIMING DIAGRAM FOR LAST IRQ

The IRQ (except LastIRQ) is generated before image capture. Last IRQ specifies the end of camera signal capture. It can be set by the timing diagram shown in [Figure 2-16](#). LastIRQEn specifies the ISR setting for next frame command. Therefore, for proper Last IRQ, you should follow the next sequence between LastIRQEn and ImgCptEn/ ImgCptEn_Sc.

It is recommended that ImgCptEn/ ImgCptEn_SC is set at the same time and at the end of SFR setting in ISR. FrameCnt specifies the next frame count. It is read in ISR.

As shown in [Figure 2-17](#), the last captured frame count is “1”, that is, Frame 1 specifies the last captured frame among frame 0~3. FrameCnt is increased by 1 at IRQ rising. The DMA input is selected by setting SFR. In this case, IRQ is generated after Output DMA operation is completed per frame. The SFR setting (ENVID_M ‘0’ → ‘1’) makes this mode aware of the starting point, and therefore, this mode does not require IRQ of starting point and LastIRQ. FrameCnt is increased by 1 at ENVID_M (InputDMA start) low to rising (‘0’ → ‘1’) and ImgCptEn_SC ‘1’.

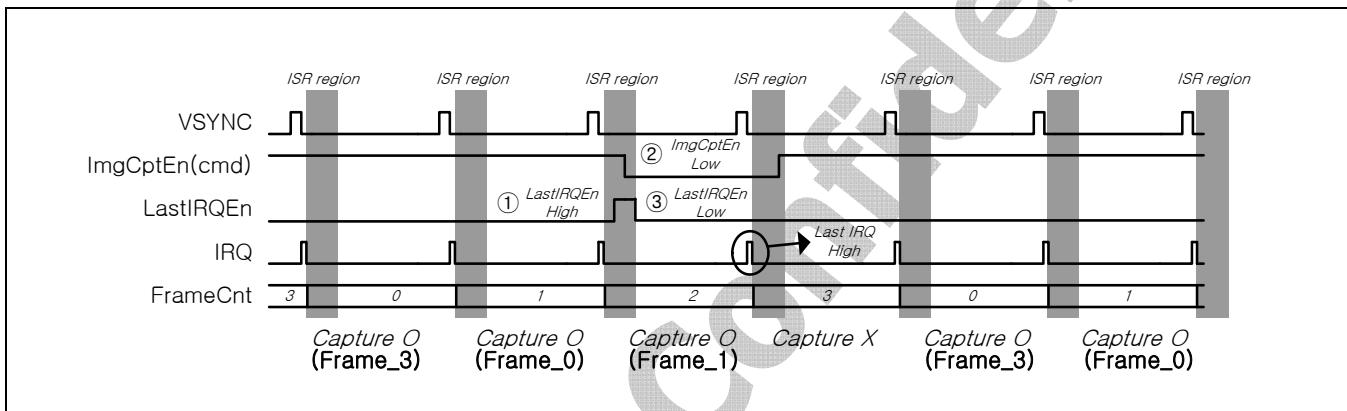


Figure 2-16 Timing Diagram for Last IRQ (LastIRQEn is Enabled)

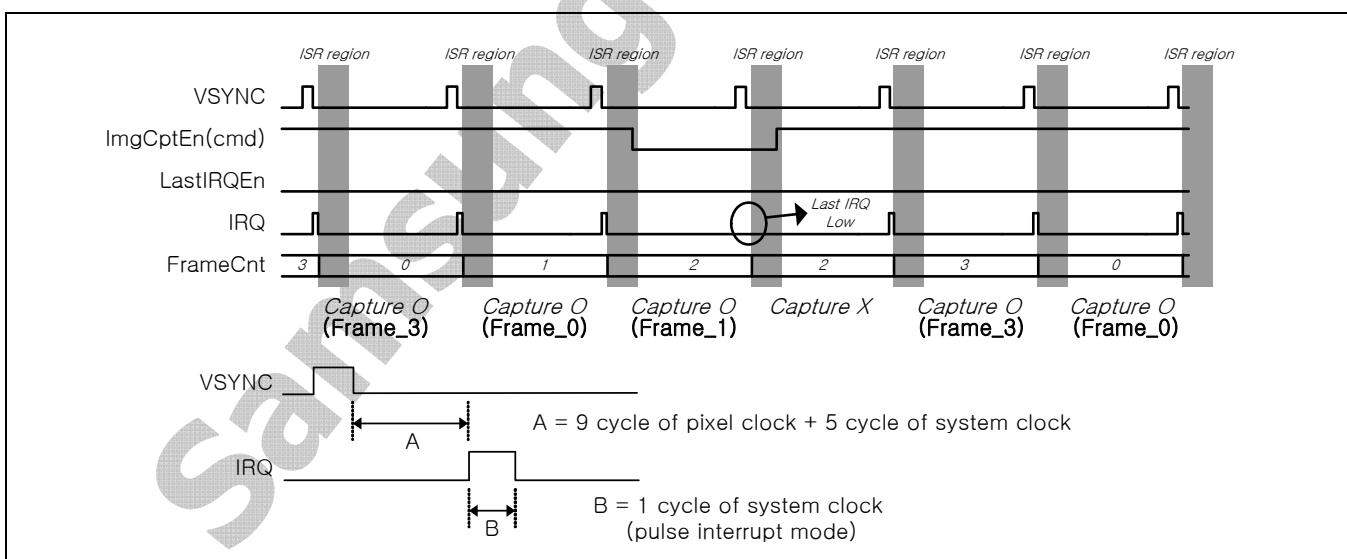


Figure 2-17 Diagram for Last IRQ (LastIRQEn is Disabled) and Timing Requirement

2.7.7 TIMING DIAGRAM FOR IRQ (MEMORY DATA SCALING MODE)

You can select the Input DMA by setting SFR. After the DMA operation is completed for each frame, IRQ is generated. The SFR setting (ENVID_M '0' → '1') makes this mode aware of the starting point, and therefore, this mode does not require IRQ of starting point and LastIRQ. FrameCnt is increased by 1 at ENVID_M low to rising ('0' → '1') and ImgCptEn_SC '1'.

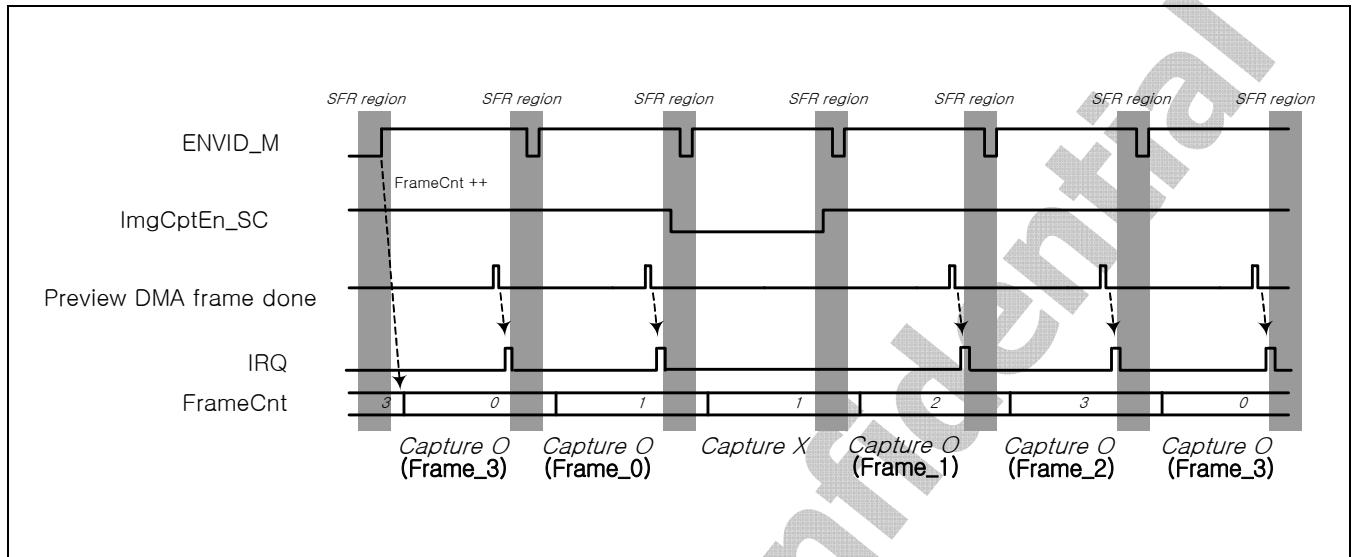


Figure 2-18 Timing Diagram for IRQ (Input DMA Path)

2.7.8 INPUT DMA FEATURE

Input DMA supports memory data scaling.

Two different image data are required for Picture-in-Picture (PIP) operation. The first image is saved in memory by codecs like H.264, MPEG4, JPEG, and so on, while the second image is saved in memory via input DMA path.

The input DMA path comprises of YCbCr/RGB output format through scaler/ DMA path. The LCD controller displays and controls the two images.

If input DMA (reading the memory data) is used in the path, SEL_DMA_CAM (MSCTRL bit [3]) signal must be set to '1'. This input path is called Memory Scaling DMA path. The window zoom function is disabled in Memory Scaling DMA path.

NOTE: The memory image format for input DMA input includes:

- YCbCr 4:2:0 (non-interleave)
- YCbCr 4:2:2 (non-interleave)
- YCbCr 4:2:2 (Interleave)
- RGB

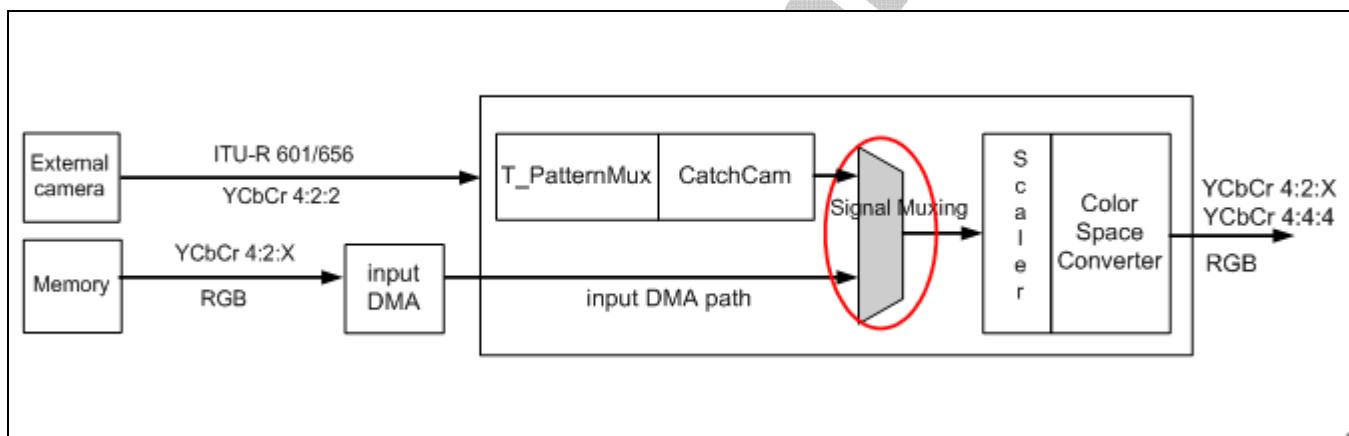


Figure 2-19 Input DMA or External Camera Interface

2.7.9 CAMERA INTERLACE INPUT SUPPORT

To get data from the external camera, S5PV210 provides two modes, namely, ITU-R BT 601 YCbCr 8-bit mode and ITU-R BT 656 YCbCr 8-bit mode. It supports progressive input and interlaced input in both the modes.

2.7.9.1 Progressive Input

In progressive mode, all the input data is stored in four buffers (that is in ping-pong memory designated by SFR) sequentially by the unit of frame. For more information, refer to [Figure 2-20](#).

2.7.9.2 Interlaced Input

In interlaced mode, all the input data is stored in four buffers (that is in ping-pong memory designated by SFR). However, in this mode, both field frame data and odd field frame data are stored successively. Therefore, even field frame data is stored in first and third ping-pong memories, while odd field frame data is stored second and fourth ping-pong memories. In case of image capture, start frame is always even field frame.

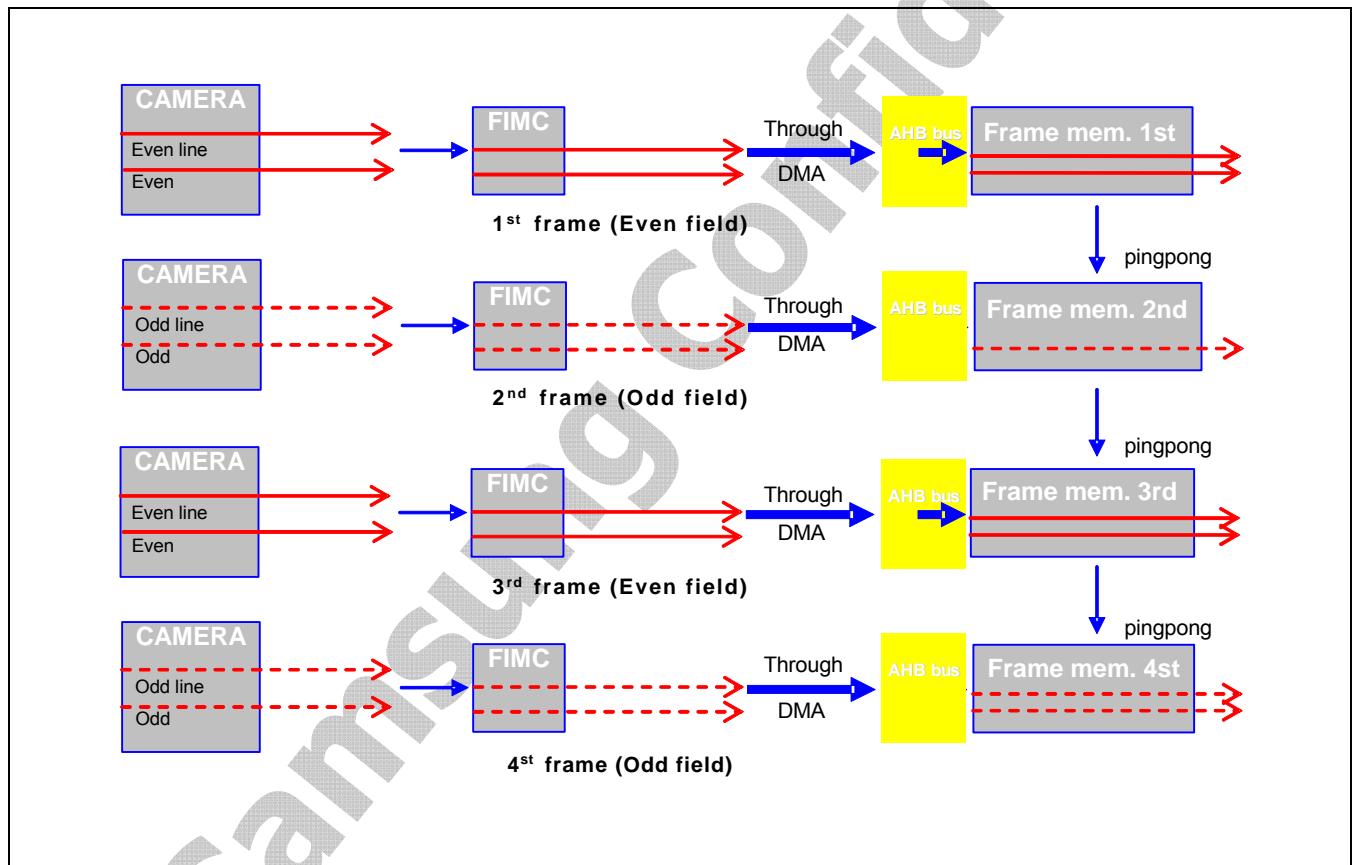


Figure 2-20 Frame Buffer Control

2.8 REGISTER DESCRIPTION

2.8.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
CISRCFMT0	0xFB20_0000	R/W	Specifies input source format.	0x0000_0000
CIWDOFST0	0xFB20_0004	R/W	Specifies window offset register.	0x0000_0000
CIGCTRL0	0xFB20_0008	R/W	Specifies global control register.	0x2001_0080
CIWDOFST20	0xFB20_0014	R/W	Specifies window offset register 2.	0x0000_0000
CIOYSA10	0xFB20_0018	R/W	Specifies Y 1st frame start address for output DMA.	0x0000_0000
CIOYSA20	0xFB20_001C	R/W	Specifies Y 2nd frame start address for output DMA.	0x0000_0000
CIOYSA30	0xFB20_0020	R/W	Specifies Y 3rd frame start address for output DMA.	0x0000_0000
CIOYSA40	0xFB20_0024	R/W	Specifies Y 4th frame start address for output DMA.	0x0000_0000
CIOCBSA10	0xFB20_0028	R/W	Specifies Cb 1st frame start address for output DMA.	0x0000_0000
CIOCBSA20	0xFB20_002C	R/W	Specifies Cb 2nd frame start address for output DMA.	0x0000_0000
CIOCBSA30	0xFB20_0030	R/W	Specifies Cb 3rd frame start address for output DMA.	0x0000_0000
CIOCBSA40	0xFB20_0034	R/W	Specifies Cb 4th frame start address for output DMA.	0x0000_0000
CIOCRSA10	0xFB20_0038	R/W	Specifies Cr 1st frame start address for output DMA.	0x0000_0000
CIOCRSA20	0xFB20_003C	R/W	Specifies Cr 2nd frame start address for output DMA.	0x0000_0000
CIOCRSA30	0xFB20_0040	R/W	Specifies Cr 3rd frame start address for output DMA.	0x0000_0000
CIOCRSA40	0xFB20_0044	R/W	Specifies Cr 4th frame start address for output DMA.	0x0000_0000
CITRGFMT0	0xFB20_0048	R/W	Specifies target image format.	0x0000_0000
CIOCTRL0	0xFB20_004C	R/W	Specifies control-related Output DMA.	0x0000_0000
CISCPRERATIO0	0xFB20_0050	R/W	Specifies pre-scaler control 1.	0x0000_0000
CISCPREDST0	0xFB20_0054	R/W	Specifies pre-scaler control 2.	0x0000_0000
CISCCTRL0	0xFB20_0058	R/W	Specifies main-scaler control.	0x1800_0000
CITAREA0	0xFB20_005C	R/W	Specifies target area.	0x0000_0000
CIOLINESKIP0	0xFB20_0060	R/W	Specifies output DMA line skip.	0x0000_0000
CISTATUS0	0xFB20_0064	R/W	Specifies status register.	0x0000_0000
CIIMGCPT0	0xFB20_00C0	R/W	Specifies image capture enable command.	0x0000_0000

Register	Address	R/W	Description	Reset Value
CICPTSEQ0	0xFB20_00C4	R/W	Specifies sequence-related capture.	0xFFFF_FFFF
CITHOLD0	0xFB20_00C8	R/W	Specifies QoS threshold.	0x0000_0000
CIIMGEFF0	0xFB20_00D0	R/W	Specifies related image effects.	0x0010_0080
CIIYSA00	0xFB20_00D4	R/W	Specifies Y frame start address for Input DMA.	0x0000_0000
CIICBSA00	0xFB20_00D8	R/W	Specifies Cb frame start address for Input DMA.	0x0000_0000
CIICRSA00	0xFB20_00DC	R/W	Specifies Cr frame start address for Input DMA.	0x0000_0000
CIILINESKIP_Y0	0xFB20_00EC	R/W	Specifies input DMA Y line skip.	0x0000_0000
CIILINESKIP_Cb0	0xFB20_00F0	R/W	Specifies input DMA Cb line skip.	0x0000_0000
CIILINESKIP_Cr0	0xFB20_00F4	R/W	Specifies input DMA Cr line skip.	0x0000_0000
CIREAL_ISIZE0	0xFB20_00F8	R/W	Specifies real input DMA image size.	0x0000_0000
MSCTRL0	0xFB20_00FC	R/W	Specifies input DMA control register.	0x0400_0000
CIIYSA10	0xFB20_0144	R/W	Specifies Y frame start address 1 for Input DMA.	0x0000_0000
CIICBSA10	0xFB20_0148	R/W	Specifies Cb frame start address 1 for Input DMA.	0x0000_0000
CIICRSA10	0xFB20_014C	R/W	Specifies Cr frame start address 1 for Input DMA.	0x0000_0000
CIOYOFF0	0xFB20_0168	R/W	Specifies output DMA Y offset.	0x0000_0000
CIOCBOFF0	0xFB20_016C	R/W	Specifies output DMA Cb offset.	0x0000_0000
CIOCROFF0	0xFB20_0170	R/W	Specifies output DMA Cr offset.	0x0000_0000
CIIYOFF0	0xFB20_0174	R/W	Specifies input DMA Y offset.	0x0000_0000
CIICBOFF0	0xFB20_0178	R/W	Specifies input DMA Cb offset.	0x0000_0000
CIICROFF0	0xFB20_017C	R/W	Specifies input DMA Cr offset.	0x0000_0000
ORGISIZE0	0xFB20_0180	R/W	Specifies input DMA original image size.	0x0000_0000
ORGOSIZE0	0xFB20_0184	R/W	Specifies output DMA original image size.	0x0000_0000
CIEXTEN0	0xFB20_0188	R/W	Specifies real output DMA image size.	0x0000_0000
CIDMAPARAM0	0xFB20_018C	R/W	Specifies DMA parameter register.	0x0000_0000
CSIIMGFMT0	0xFB20_0194	R/W	Specifies MIPI CSI image format register.	0x0000_001E
CMISC0	0xFB20_0198	R/W	Specifies miscellaneous control	0x0000_0000
CIKEY0	0xFB20_019C	R/W	Specifies key detect register.	0x0000_0000

Register	Address	R/W	Description	Reset Value

Register	Address	R/W	Description	Reset Value
CISRCFMT1	0xFB30_0000	R/W	Specifies input source format.	0x0000_0000
CIWDOFST1	0xFB30_0004	R/W	Specifies window offset register.	0x0000_0000
CIGCTRL1	0xFB30_0008	R/W	Specifies global control register.	0x2001_0080
CIWDOFST21	0xFB30_0014	R/W	Specifies window offset register 2.	0x0000_0000
CIOYSA11	0xFB30_0018	R/W	Specifies Y 1st frame start address for output DMA.	0x0000_0000
CIOYSA21	0xFB30_001C	R/W	Specifies Y 2nd frame start address for output DMA.	0x0000_0000
CIOYSA31	0xFB30_0020	R/W	Specifies Y 3rd frame start address for output DMA.	0x0000_0000
CIOYSA41	0xFB30_0024	R/W	Specifies Y 4th frame start address for output DMA.	0x0000_0000
CIOCBSA11	0xFB30_0028	R/W	Specifies Cb 1st frame start address for output DMA.	0x0000_0000
CIOCBSA21	0xFB30_002C	R/W	Specifies Cb 2nd frame start address for output DMA.	0x0000_0000
CIOCBSA31	0xFB30_0030	R/W	Specifies Cb 3rd frame start address for output DMA.	0x0000_0000
CIOCBSA41	0xFB30_0034	R/W	Specifies Cb 4th frame start address for output DMA.	0x0000_0000
CIOCRSA11	0xFB30_0038	R/W	Specifies Cr 1st frame start address for output DMA.	0x0000_0000
CIOCRSA21	0xFB30_003C	R/W	Specifies Cr 2nd frame start address for output DMA.	0x0000_0000
CIOCRSA31	0xFB30_0040	R/W	Specifies Cr 3rd frame start address for output DMA.	0x0000_0000
CIOCRSA41	0xFB30_0044	R/W	Specifies Cr 4th frame start address for output DMA.	0x0000_0000
CITRGFMT1	0xFB30_0048	R/W	Specifies target image format.	0x0000_0000
CIOCTRL1	0xFB30_004C	R/W	Specifies control-related Output DMA.	0x0000_0000
CISCPRERATIO1	0xFB30_0050	R/W	Specifies pre-scaler control 1.	0x0000_0000
CISCPREDST1	0xFB30_0054	R/W	Specifies pre-scaler control 2.	0x0000_0000
CISCCTRL1	0xFB30_0058	R/W	Specifies main-scaler control.	0x1800_0000
CITAREA1	0xFB30_005C	R/W	Specifies target area.	0x0000_0000
CIOLINESKIP1	0xFB30_0060	R/W	Specifies output DMA line skip.	0x0000_0000
CISTATUS1	0xFB30_0064	R/W	Specifies status register.	0x0000_0000
CIIMGCPT1	0xFB30_00C0	R/W	Specifies image capture enable command.	0x0000_0000
CICPTSEQ1	0xFB30_00C4	R/W	Specifies sequence-related capture.	0xFFFF_FFFF
CITHOLD1	0xFB30_00C8	R/W	Specifies QoS threshold.	0x0000_0000
CIIMAGEFF1	0xFB30_00D0	R/W	Specifies related image effects.	0x0010_0080
CIIYSA01	0xFB30_00D4	R/W	Specifies Y frame start address for Input	0x0000_0000

Register	Address	R/W	Description	Reset Value
			DMA.	
CIICBSA01	0xFB30_00D8	R/W	Specifies Cb frame start address for Input DMA.	0x0000_0000
CIICRSA01	0xFB30_00DC	R/W	Specifies Cr frame start address for Input DMA.	0x0000_0000
CIILINESKIP_Y1	0xFB30_00EC	R/W	Specifies input DMA Y line skip.	0x0000_0000
CIILINESKIP_Cb1	0xFB30_00F0	R/W	Specifies input DMA Cb line skip.	0x0000_0000
CIILINESKIP_Cr1	0xFB30_00F4	R/W	Specifies input DMA Cr line skip.	0x0000_0000
CIREAL_ISIZE1	0xFB30_00F8	R/W	Specifies real input DMA image size.	0x0000_0000
MSCTRL1	0xFB30_00FC	R/W	Specifies input DMA control register.	0x0400_0000
CIIYSA11	0xFB30_0144	R/W	Specifies Y frame start address 1 for Input DMA.	0x0000_0000
CIICBSA11	0xFB30_0148	R/W	Specifies Cb frame start address 1 for Input DMA.	0x0000_0000
CIICRSA11	0xFB30_014C	R/W	Specifies Cr frame start address 1 for Input DMA.	0x0000_0000
CIOYOFF1	0xFB30_0168	R/W	Specifies output DMA Y offset.	0x0000_0000
CIOCBOFF1	0xFB30_016C	R/W	Specifies output DMA Cb offset.	0x0000_0000
CIOCROFF1	0xFB30_0170	R/W	Specifies output DMA Cr offset.	0x0000_0000
CIIYOFF1	0xFB30_0174	R/W	Specifies input DMA Y offset.	0x0000_0000
CIICBOFF1	0xFB30_0178	R/W	Specifies input DMA Cb offset.	0x0000_0000
CIICROFF1	0xFB30_017C	R/W	Specifies input DMA Cr offset.	0x0000_0000
ORGISIZE1	0xFB30_0180	R/W	Specifies input DMA original image size.	0x0000_0000
ORGOSIZE1	0xFB30_0184	R/W	Specifies output DMA original image size.	0x0000_0000
CIEXTEN1	0xFB30_0188	R/W	Specifies real output DMA image size.	0x0000_0000
CIDMAPARAM1	0xFB30_018C	R/W	Specifies DMA parameter register.	0x0000_0000
CSIIMGFMT1	0xFB30_0194	R/W	Specifies MIPI CSI image format register.	0x0000_001E
CMISC1	0xFB30_0198	R/W	Specifies miscellaneous control	0x0000_0000
CIKEY1	0xFB30_019C	R/W	Specifies key detect register.	0x0000_0000

Register	Address	R/W	Description	Reset Value
CISRCFMT2	0xFB40_0000	R/W	Specifies input source format.	0x0000_0000
CIWDOFST2	0xFB40_0004	R/W	Specifies window offset register.	0x0000_0000
CIGCTRL2	0xFB40_0008	R/W	Specifies global control register.	0x2001_0080
CIWDOFST22	0xFB40_0014	R/W	Specifies window offset register 2.	0x0000_0000
CIOYSA12	0xFB40_0018	R/W	Specifies Y 1st frame start address for output DMA.	0x0000_0000
CIOYSA22	0xFB40_001C	R/W	Specifies Y 2nd frame start address for output DMA.	0x0000_0000
CIOYSA32	0xFB40_0020	R/W	Specifies Y 3rd frame start address for output DMA.	0x0000_0000
CIOYSA42	0xFB40_0024	R/W	Specifies Y 4th frame start address for output DMA.	0x0000_0000
CIOCBSA12	0xFB40_0028	R/W	Specifies Cb 1st frame start address for output DMA.	0x0000_0000
CIOCBSA22	0xFB40_002C	R/W	Specifies Cb 2nd frame start address for output DMA.	0x0000_0000
CIOCBSA32	0xFB40_0030	R/W	Specifies Cb 3rd frame start address for output DMA.	0x0000_0000
CIOCBSA42	0xFB40_0034	R/W	Specifies Cb 4th frame start address for output DMA.	0x0000_0000
CIOCRSA12	0xFB40_0038	R/W	Specifies Cr 1st frame start address for output DMA.	0x0000_0000
CIOCRSA22	0xFB40_003C	R/W	Specifies Cr 2nd frame start address for output DMA.	0x0000_0000
CIOCRSA32	0xFB40_0040	R/W	Specifies Cr 3rd frame start address for output DMA.	0x0000_0000
CIOCRSA42	0xFB40_0044	R/W	Specifies Cr 4th frame start address for output DMA.	0x0000_0000
CITRGFMT2	0xFB40_0048	R/W	Specifies target image format.	0x0000_0000
CIOCTRL2	0xFB40_004C	R/W	Specifies control-related Output DMA.	0x0000_0000
CISCPRERATIO2	0xFB40_0050	R/W	Specifies pre-scaler control 1.	0x0000_0000
CISCCTRL2	0xFB40_0054	R/W	Specifies pre-scaler control 2.	0x0000_0000
CISCCTRL2	0xFB40_0058	R/W	Specifies main-scaler control.	0x1800_0000
CITAREA2	0xFB40_005C	R/W	Specifies target area.	0x0000_0000
CIOLINESKIP2	0xFB40_0060	R/W	Specifies output DMA line skip.	0x0000_0000
CISTATUS2	0xFB40_0064	R/W	Specifies status register.	0x0000_0000
CIIMGCPT2	0xFB40_00C0	R/W	Specifies image capture enable command.	0x0000_0000
CICPTSEQ2	0xFB40_00C4	R/W	Specifies sequence-related capture.	0xFFFF_FFFF
CITHOLD2	0xFB40_00C8	R/W	Specifies QoS threshold.	0x0000_0000
CIIMGEFF2	0xFB40_00D0	R/W	Specifies related image effects.	0x0010_0080

Register	Address	R/W	Description	Reset Value
CIIYSA02	0xFB40_00D4	R/W	Specifies Y frame start address for Input DMA.	0x0000_0000
CIICBSA02	0xFB40_00D8	R/W	Specifies Cb frame start address for Input DMA.	0x0000_0000
CIICRSA02	0xFB40_00DC	R/W	Specifies Cr frame start address for Input DMA.	0x0000_0000
CIILINESKIP_Y2	0xFB40_00EC	R/W	Specifies input DMA Y line skip.	0x0000_0000
CIILINESKIP_Cb2	0xFB40_00F0	R/W	Specifies input DMA Cb line skip.	0x0000_0000
CIILINESKIP_Cr2	0xFB40_00F4	R/W	Specifies input DMA Cr line skip.	0x0000_0000
CIREAL_ISIZE2	0xFB40_00F8	R/W	Specifies real input DMA image size.	0x0000_0000
MSCTRL2	0xFB40_00FC	R/W	Specifies input DMA control register.	0x0400_0000
CIIYSA12	0xFB40_0144	R/W	Specifies Y frame start address 1 for Input DMA.	0x0000_0000
CIICBSA12	0xFB40_0148	R/W	Specifies Cb frame start address 1 for Input DMA.	0x0000_0000
CIICRSA12	0xFB40_014C	R/W	Specifies Cr frame start address 1 for Input DMA.	0x0000_0000
CIOYOFF2	0xFB40_0168	R/W	Specifies output DMA Y offset.	0x0000_0000
CIOCBOFF2	0xFB40_016C	R/W	Specifies output DMA Cb offset.	0x0000_0000
CIOCROFF2	0xFB40_0170	R/W	Specifies output DMA Cr offset.	0x0000_0000
CIIYOFF2	0xFB40_0174	R/W	Specifies input DMA Y offset.	0x0000_0000
CIICBOFF2	0xFB40_0178	R/W	Specifies input DMA Cb offset.	0x0000_0000
CIICROFF2	0xFB40_017C	R/W	Specifies input DMA Cr offset.	0x0000_0000
ORGISIZE2	0xFB40_0180	R/W	Specifies input DMA original image size.	0x0000_0000
ORGOSIZE2	0xFB40_0184	R/W	Specifies output DMA original image size.	0x0000_0000
CIEXTEN2	0xFB40_0188	R/W	Specifies real output DMA image size.	0x0000_0000
CIDMAPARAM2	0xFB40_018C	R/W	Specifies DMA parameter register.	0x0000_0000
CSIIMGFMT2	0xFB40_0194	R/W	Specifies MIPI CSI image format register.	0x0000_001E
CMISC2	0xFB40_0198	R/W	Specifies miscellaneous control	0x0000_0000
CIKEY2	0xFB40_019C	R/W	Specifies key detect register.	0x0000_0000

NOTE:

NOTE: 'L' means that SFR can change at vsync edge during camera capture. (O: possible change, X: impossible change). Also, 'M' means that SFRs have relationship capturing result while using input DMA path. (O: relationship, X: no relationship).

2.8.1.1 Camera Source Format Register (CISRCFMTn)

- CISRCFMT0, R/W, Address = 0xFB20_0000
- CISRCFMT1, R/W, Address = 0xFB30_0000
- CISRCFMT2, R/W, Address = 0xFB40_0000

CISRCFMTn	Bit	Description	Initial State						
ITU601_656n	[31]	1 = ITU-R BT.601 YCbCr 8-bit mode enable 0 = ITU-R BT.656 YCbCr 8-bit mode enable (ML=XX)	0						
UVOffset	[30]	Controls Cb,Cr value offset. 1 = Cb=Cb+128 , Cr=Cr+128 0 = +0 (normally used) (ML=XX)	0						
Reserved	[29]	Should be '0'.	0						
SrcHsize_CAM	[28:16]	Specifies the source horizontal pixel number (camera or FIFO input). For more information, refer to gathering extension register (SrcHsize_CAM_ext). Note) 16's multiple. Must be 4's multiple of PreHorRatio if WinOfsEn is 0 (ML=XO)	0						
Order422_CAM	[15:14]	Specifies the camera input YCbCr order for 8-bit mode. <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>8-bit mode</td></tr> <tr><td>Data Flow →</td></tr> <tr><td>00 = Y0Cb0Y1Cr0...</td></tr> <tr><td>01 = Y0Cr0Y1Cb0...</td></tr> <tr><td>10 = Cb0Y0Cr0Y1...</td></tr> <tr><td>11 = Cr0Y0Cb0Y1...</td></tr> </table> (ML=XX)	8-bit mode	Data Flow →	00 = Y0Cb0Y1Cr0...	01 = Y0Cr0Y1Cb0...	10 = Cb0Y0Cr0Y1...	11 = Cr0Y0Cb0Y1...	0
8-bit mode									
Data Flow →									
00 = Y0Cb0Y1Cr0...									
01 = Y0Cr0Y1Cb0...									
10 = Cb0Y0Cr0Y1...									
11 = Cr0Y0Cb0Y1...									
SrcVsize_CAM	[13:0]	Specifies the source vertical pixel number (Camera or FIFO input). Note) It must be a multiple of PreVerRatio if V scale down or WinOfsEn is 0. It should be 2's multiple in case YCbCr 422 input and cam interlace mode. (ML=XO)	0						

2.8.1.2 Camera Window Offset Register (CIWDOFSTn)

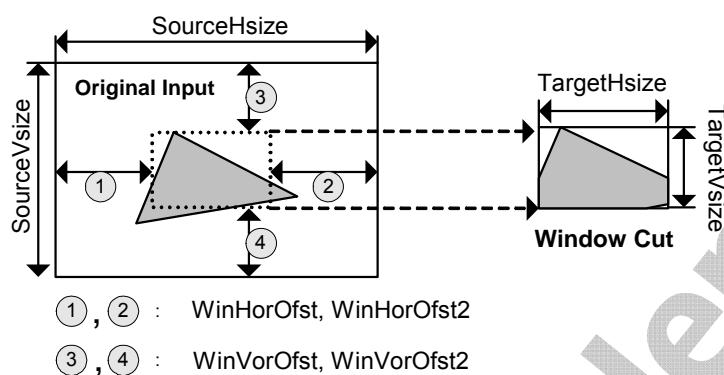


Figure 2-21 Camera Window Offset Scheme

WinHorOfst2 and WinVerOfst2 are assigned in the CIWDOFST2n registers.

- CIWDOFST0, R/W, Address = 0xFB20_0004
- CIWDOFST1, R/W, Address = 0xFB30_0004
- CIWDOFST2, R/W, Address = 0xFB40_0004

CIWDOFSTn	Bit	Description	Initial State
WinOfsEn	[31]	1 = Enables window offset 0 = No offset Note) If input format is either RAW or WB(Write Back), this function is not valid. (ML=XO)	0
ClrOvFiY	[30]	1 = Clears the overflow indication flag of input FIFO Y 0 = Normal (ML=XX)	0
ClrOvRLB	[29]	Clears the overflow indication flag of Line Buffer for Rotation. (ML=XX)	0
Reserved	[28:27]	Reserved	0

WinHorOfst	[26:16]	Specifies window horizontal offset by pixel unit. It should be multiple of 2. Note) CAMIF0 & CAMIF2 : SourceHsize-WinHorOfst- WinHorOfst2 should be a multiple of 16. For more information, refer to gathering extension register (WinHorOfst_ext). (ML=XO)	0
ClrOvFiCb	[15]	1 = Clears the overflow indication flag of input FIFO Cb 0 = Normal (ML=XX)	0
ClrOvFiCr	[14]	1 = Clears the overflow indication flag of input FIFO Cr 0 = Normal (ML=XX)	0
Reserved	[13:12]	Reserved	0
WinVerOfst	[11:0]	Specifies window vertical offset by pixel unit. In case of interlaced input, this value should be 2's multiple. (ML=XO)	0

NOTE: Clear bits should be set to zero after clearing the flags.

Below constraints of Crop HSIZE and Crop Vsize are only for CAMIF0 & CAMIF1 & CAMIF2.

Crop Hsize (= SourceHsize - WinHorOfst - WinHorOfst2) must be 16's multiple. Also, It should be 4's multiple of PreHorRatio.

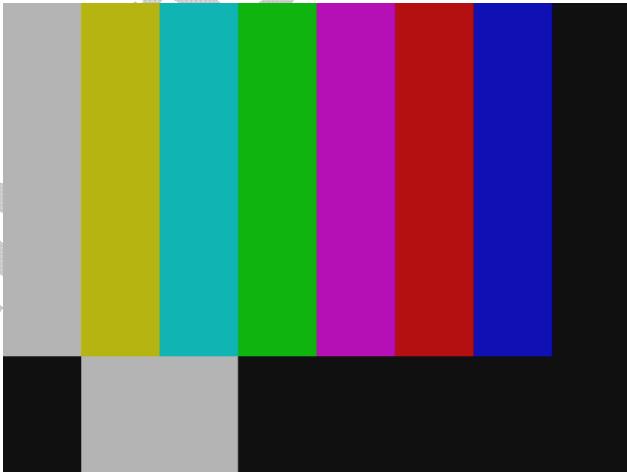
Crop Vsize (= SourceVsize - WinVerOfst - WinVerOfst2) must be multiple of PreVerRatio when V scale down.
Must be an even number and minimum 8 if the output image format is YCbCr 4:2:0.

Example:

Crop Hsize	Permitted Prescale_ratio	PreDstWidth_xx
8n	2	4n
16n	2 or 4	4n
32n	2, 4 or 8	4n

2.8.1.3 Global Control Register (CIGCTRLn)

- CIGCTRL0, R/W, Address = 0xFB20_0008
- CIGCTRL1, R/W, Address = 0xFB30_0008
- CIGCTRL2, R/W, Address = 0xFB40_0008

CIGCTRLn	Bit	Description	Initial State
SwRst	[31]	<p>Specifies the camera interface software reset. Before you set this bit, set the ITU601_656n bit of CISRCFMT as "1" temporarily at first SFR setting. Next sequence is recommended.</p> <p>[ITU601 case: ITU601_656n "1" → SwRst "1" → SwRst "0" for first SFR setting],</p> <p>[ITU656 case: ITU601_656n "1" → SwRst "1" → SwRst "0" → ITU601_656n "0" for first SFR setting]</p> <p>Note)</p> <p>1) User should not use SwRst function in the middle of transferring data out by DMA.</p> <p>2) User should disable 'ImgCptEn' and 'IRQ_Enable' bit before using this function.</p> <p>(ML=XX)</p>	0
Reserved	[30]	Should be '0'.	0
SelCam_ITU	[29]	<p>Selects external multiple ITU camera.</p> <p>1 = Selects ITU Camera A 0 = Selects ITU Camera B</p> <p>(ML=XX)</p>	1
TestPattern	[28:27]	<p>This register should be set at ITU-T 601 8-bit mode only, not at ITU-T 656 mode. Source CAM size should be 640 x 480 size and Order422_CAM register should be '00'.</p> <p>00 = External camera processor input (normal) 01 = Color bar test pattern</p>  <p>10 = Reserved 11 = Reserved</p> <p>(ML=XX)</p>	0

CIGCTRLn	Bit	Description	Initial State
InvPolPCLK	[26]	1 = Inverse the polarity of PCLK 0 = Normal (ML=XX)	0
InvPolVSYNC	[25]	1 = Inverse the polarity of VSYNC 0 = Normal (ML=XX)	0
InvPolHREF	[24]	1 = Inverse the polarity of HREF 0 = Normal (ML=XX)	0
Reserved	[23]	Should be '0'.	0
IRQ_Ovfen	[22]	1 = Enables Overflow interrupt (Interrupt is generated during overflow occurrence) 0 = Disables Overflow interrupt (normal) (ML=XX)	0
Href_mask	[21]	1 = Mask out Href during Vsync blank 0 = No mask (ML=XX)	0
Reserved	[20]	Should be '1'.	0
IRQ_CLR	[19]	Writes IRQ_CLR to '1' to clear Interrupt. This bit Auto-clear. (ML=XX)	0
IRQ_EndDisable	[18]	This bit is related to Camera or Local FIFO(WB) input only. 1 = Interrupt enables at Frame end point 0 = Interrupt disables at Frame end point (default) (ML=XX)	0
IRQ_StartEnable	[17]	This bit is related to Camera or Local FIFO(WB) input only. 1 = Interrupt disables at Frame start point 0 = Interrupt enables at Frame start point (default) (ML=XX)	0
IRQ_Enable	[16]	1 = Enables Interrupt (default) 0 = Disables Interrupt Note: If the interrupt enables, then the bit[20]@CIGCTRLn should be set to '1'. (ML=XX)	1
Reserved	[15:14]	Reserved	0
SwUpdate	[13]	Updates shadow register by software setting (Both DMA input and Camera input are applied) 1 = Shadow register updates immediately (Autoclears 1 to 0) 0 = Shadow register updates at Hardware frame start pulse (ML=XX)	0
ShadowDisable	[12]	Shadow register cannot be updated by Hardware frame start (input path should be local path). At the start of first frame, it is necessary to set ShadowDisable as '0'. 1 = Disables (update is impossible)	0

CIGCTRLn	Bit	Description	Initial State
		0 = Enables (update is possible) (ML=XX)	
Reserved	[11:9]	-	0
CAM_JPEG	[8]	Specifies the camera input in 8-bit JPEG format. If the image format is selected as JPEG format, the image format conversion is not possible. It should be set as scaler bypass mode and ITU601 8-bit mode. 1 = JPEG format 0 = Non-JPEG format (ML=XX)	0
Reserved	[7]	Should be '1'.	1
SelWB_CAMIF	[6]	Specifies the WriteBack input select signal. 1 = WriteBack input select (YCbCr4:4:4 only) 0 = Camera input select (ML=XX)	0
CSC_601_709	[5]	Selects ColorSpaceConversion equation. 1 = ITU709 equation select (HD size target method) 0 = ITU601 equation select (SD size target method) (ML=XO)	0
InvPolHSYNC	[4]	1 = Inverses the polarity of HSYNC (this bit is useful only when delay count interlace mode and FIELD port is connected to HSYNC) 0 = Normal (ML=XX)	0
SelCam_CAMIF	[3]	Selects the External camera. 1 = Selects MIPI Camera 0 = Selects ITU Camera (ML=XX)	0
FIELDMODE	[2]	Specifies the ITU601 interlace field mode (Do not care this bit in ITU656 mode). 1 = Uses the FIELD port mode (FIELD port = FIELD signal) 0 = Uses the Edge delay count mode (FIELD port = HSYNC signal) Note: Check the FIELD port connection. (ML=XX)	0
InvPolFIELD	[1]	1 = Inverses the polarity of FIELD 0 = Normal (ML=XX)	0
Cam_Interlace	[0]	Specifies the External Camera scan method. 1 = Interlace 0 = Progressive If this mode is enable, control signals cannot change under operation except ImgCptEn,ImgCptEnSC and ScalerStart (ML=XX)	0

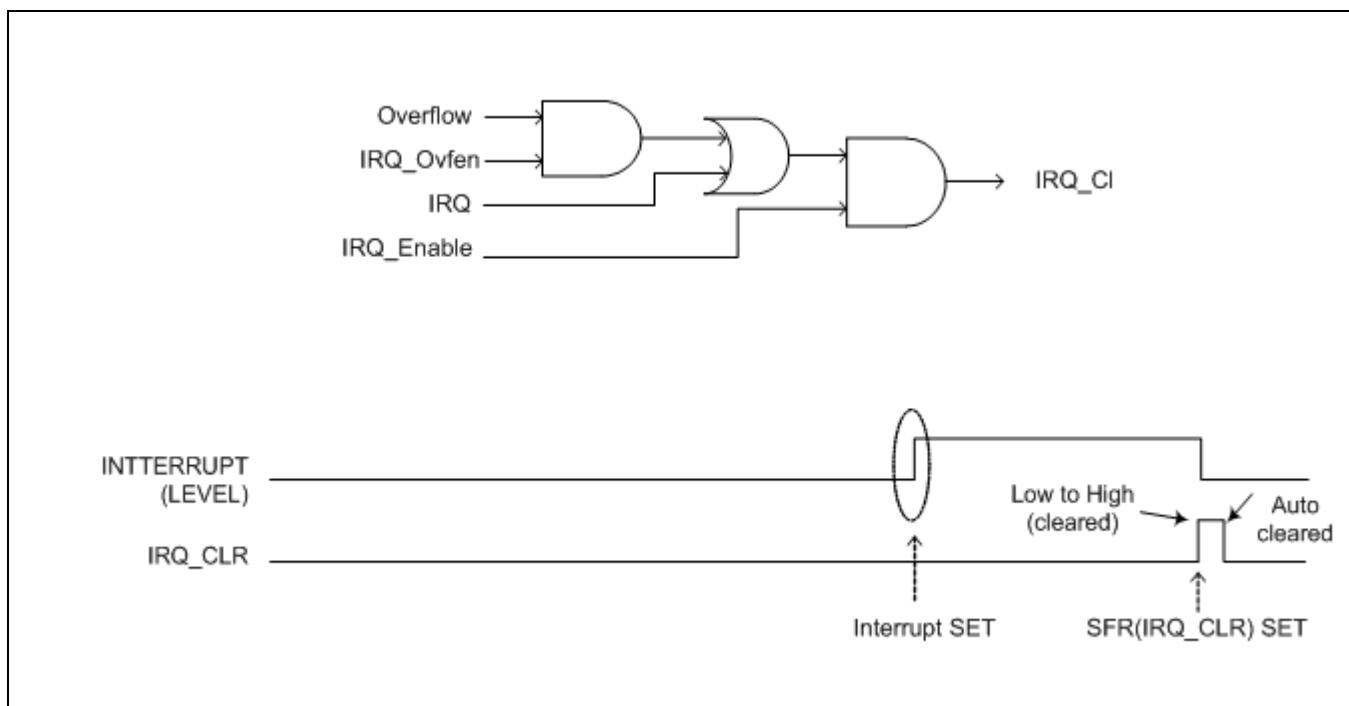


Figure 2-22 Interrupt Generation Scheme

2.8.1.4 Window Offset Register 2 (CIWDOFST2n)

- CIWDOFST20, R/W, Address = 0xFB20_0014
- CIWDOFST21, R/W, Address = 0xFB30_0014
- CIWDOFST22, R/W, Address = 0xFB40_0014

CIWDOFST2n	Bit	Description	Initial State
Reserved	[31:28]	Reserved	0
WinHorOfst2	[27:16]	<p>Specifies the window horizontal offset2 by pixel unit. It should be multiple of 2.</p> <p>Note) CAMIF0 & CAMIF2 : SourceHsize-WinHorOfst- WinHorOfst2 should be multiple of 16. (ML=XO)</p>	0
Reserved	[15:12]	Reserved	0

WinVerOfst2	[11:0]	Specifies the window vertical offset2 by pixel unit. In case of interlaced input, this value should be 2's multiple. (ML=XO)	0
-------------	--------	--	---

2.8.1.5 Output DMA Y1 Start Address Register (CIOYSA1n)

- CIOYSA10, R/W, Address = 0xFB20_0018
- CIOYSA11, R/W, Address = 0xFB30_0018
- CIOYSA12, R/W, Address = 0xFB40_0018

CIOYSA1n	Bit	Description	Initial State
CIOYSA1	[31:0]	Output format: YCbCr 2/3 plane → Y 1st frame start address Output format: YCbCr 1 plane → YCbCr 1st frame start address Output format: RGB → RGB 1st frame start address Note: In tile mode, this value should be aligned 4Kbytes. That is to say, CIOYSA1[11:0] should be 0x000. (ML=OX)	0

2.8.1.6 Output DMA Y2 Start Address Register (CIOYSA2n)

- CIOYSA20, R/W, Address = 0xFB20_001C
- CIOYSA21, R/W, Address = 0xFB30_001C
- CIOYSA22, R/W, Address = 0xFB40_001C

CIOYSA2n	Bit	Description	Initial State
CIOYSA2	[31:0]	Output format: YCbCr 2/3 plane → Y 2nd frame start address Output format: YCbCr 1 plane → YCbCr 2nd frame start address Output format: RGB → RGB 2nd frame start address Note: In tile mode, this value should be aligned 4Kbytes. That is to say, CIOYSA1[11:0] should be 0x000. (ML=OX)	0

2.8.1.7 Output DMA Y3 Start Address Register (CIOYSA3n)

- CIOYSA30, R/W, Address = 0xFB20_0020
- CIOYSA31, R/W, Address = 0xFB30_0020
- CIOYSA32, R/W, Address = 0xFB40_0020

CIOYSA3n	Bit	Description	Initial State
CIOYSA3	[31:0]	Output format: YCbCr 2/3 plane → Y 3rd frame start address Output format: YCbCr 1 plane → YCbCr 3rd frame start address Output format: RGB → RGB 3rd frame start address Note: In tile mode, this value should be aligned 4Kbytes. That is to say, CIOYSA1[11:0] should be 0x000. (ML=OX)	0

2.8.1.8 Output DMA Y4 Start Address Register (CIOYSA4n)

- CIOYSA40, R/W, Address = 0xFB20_0024
- CIOYSA41, R/W, Address = 0xFB30_0024
- CIOYSA42, R/W, Address = 0xFB40_0024

CIOYSA4n	Bit	Description	Initial State
CIOYSA4	[31:0]	Output format: YCbCr 2/3 plane → Y 4th frame start address Output format: YCbCr 1 plane → YCbCr 4th frame start address Output format: RGB → RGB 4th frame start address Note: In tile mode, this value should be aligned 4Kbytes. That is to say, CIOYSA1[11:0] should be 0x000. (ML=OX)	0

2.8.1.9 Output DMA Cb1 Start Address Register (CIOCBSA1n)

- CIOCBSA10, R/W, Address = 0xFB20_0028
- CIOCBSA11, R/W, Address = 0xFB30_0028
- CIOCBSA12, R/W, Address = 0xFB40_0028

CIOCBSA1n	Bit	Description	Initial State
CIOCBSA1	[31:0]	Output format: YCbCr 3 plane → Cb 1st frame start address Output format: YCbCr 2 plane → CbCr 1st frame start address Note: In tile mode, this value should be aligned 4Kbytes. That is to say, CIOYSA1[11:0] should be 0x000. (ML=0X)	0

2.8.1.10 Output DMA Cb2 Start Address Register (CIOCBSA2n)

- CIOCBSA20, R/W, Address = 0xFB20_002C
- CIOCBSA21, R/W, Address = 0xFB30_002C
- CIOCBSA22, R/W, Address = 0xFB40_002C

CIOCBSA2n	Bit	Description	Initial State
CIOCBSA2	[31:0]	Output format: YCbCr 3 plane → Cb 2nd frame start address Output format: YCbCr 2 plane → CbCr 2nd frame start address Note: In tile mode, this value should be aligned 4Kbytes. That is to say, CIOYSA1[11:0] should be 0x000. (ML=OX)	0

2.8.1.11 Output DMA Cb3 Start Address Register (CIOCBSA3n)

- CIOCBSA30, R/W, Address = 0xFB20_0030
- CIOCBSA31, R/W, Address = 0xFB30_0030
- CIOCBSA32, R/W, Address = 0xFB40_0030

CIOCBSA3n	Bit	Description	Initial State
CIOCBSA3	[31:0]	Output format: YCbCr 3 plane → Cb 3rd frame start address Output format: YCbCr 2 plane → CbCr 3rd frame start address Note: In tile mode, this value should be aligned 4Kbytes. That is to say, CIOYSA1[11:0] should be 0x000. (ML=OX)	0

2.8.1.12 Output DMA Cb4 Start Address Register (CIOCBSA4n)

- CIOCBSA40, R/W, Address = 0xFB20_0034
- CIOCBSA41, R/W, Address = 0xFB30_0034
- CIOCBSA42, R/W, Address = 0xFB40_0034

CIOCBSA4n	Bit	Description	Initial State
CIOCBSA4	[31:0]	Output format: YCbCr 3 plane → Cb 4th frame start address Output format: YCbCr 2 plane → CbCr 4th frame start address Note: In tile mode, this value should be aligned 4Kbytes. That is to say, CIOYSA1[11:0] should be 0x000. (ML=OX)	0

2.8.1.13 Output DMA Cr1 Start Address Register (CIOCRSA1n)

- CIOCRSA10, R/W, Address = 0xFB20_0038

- CIOCRSA11, R/W, Address = 0xFB30_0038
- CIOCRSA12, R/W, Address = 0xFB40_0038

CIOCRSA1n	Bit	Description	Initial State
CIOCRSA1	[31:0]	Output format: YCbCr 3 plane → Cr 1st frame start address Note: In tile mode, this value should be aligned 4Kbytes. That is to say, CIOYSA1[11:0] should be 0x000. (ML=OX)	0

2.8.1.14 Output DMA Cr2 Start Address Register (CIOCRSA2n)

- CIOCRSA20, R/W, Address = 0xFB20_003C
- CIOCRSA21, R/W, Address = 0xFB30_003C
- CIOCRSA22, R/W, Address = 0xFB40_003C

CIOCRSA2n	Bit	Description	Initial State
CIOCRSA2	[31:0]	Output format: YCbCr 3 plane → Cr 2nd frame start address Note: In tile mode, this value should be aligned 4Kbytes. That is to say, CIOYSA1[11:0] should be 0x000. (ML=OX)	0

2.8.1.15 Output DMA Cr3 Start Address Register (CIOCRSA3n)

- CIOCRSA30, R/W, Address = 0xFB20_0040
- CIOCRSA31, R/W, Address = 0xFB30_0040
- CIOCRSA32, R/W, Address = 0xFB40_0040

CIOCRSA3n	Bit	Description	Initial State
CIOCRSA3	[31:0]	Output format: YCbCr 3 plane → Cr 3rd frame start address Note: In tile mode, this value should be aligned 4Kbytes. That is to say, CIOYSA1[11:0] should be 0x000. (ML=OX)	0

2.8.1.16 Output DMA Cr4 Start Address Register (CIOCRSA4n)

- CIOCRSA40, R/W, Address = 0xFB20_0044
- CIOCRSA41, R/W, Address = 0xFB30_0044
- CIOCRSA42, R/W, Address = 0xFB40_0044

CIOCRSA4n	Bit	Description	Initial State
CIOCRSA4	[31:0]	Output format: YCbCr 3 plane → Cr 4th frame start address Note: In tile mode, this value should be aligned 4Kbytes. That is to say, CIOYSA1[11:0] should be 0x000. (ML=OX)	0

2.8.1.17 Target Format Register (CITRGFMTn)

- CITRGFMT0, R/W, Address = 0xFB20_0048
- CITRGFMT1, R/W, Address = 0xFB30_0048
- CITRGFMT2, R/W, Address = 0xFB40_0048

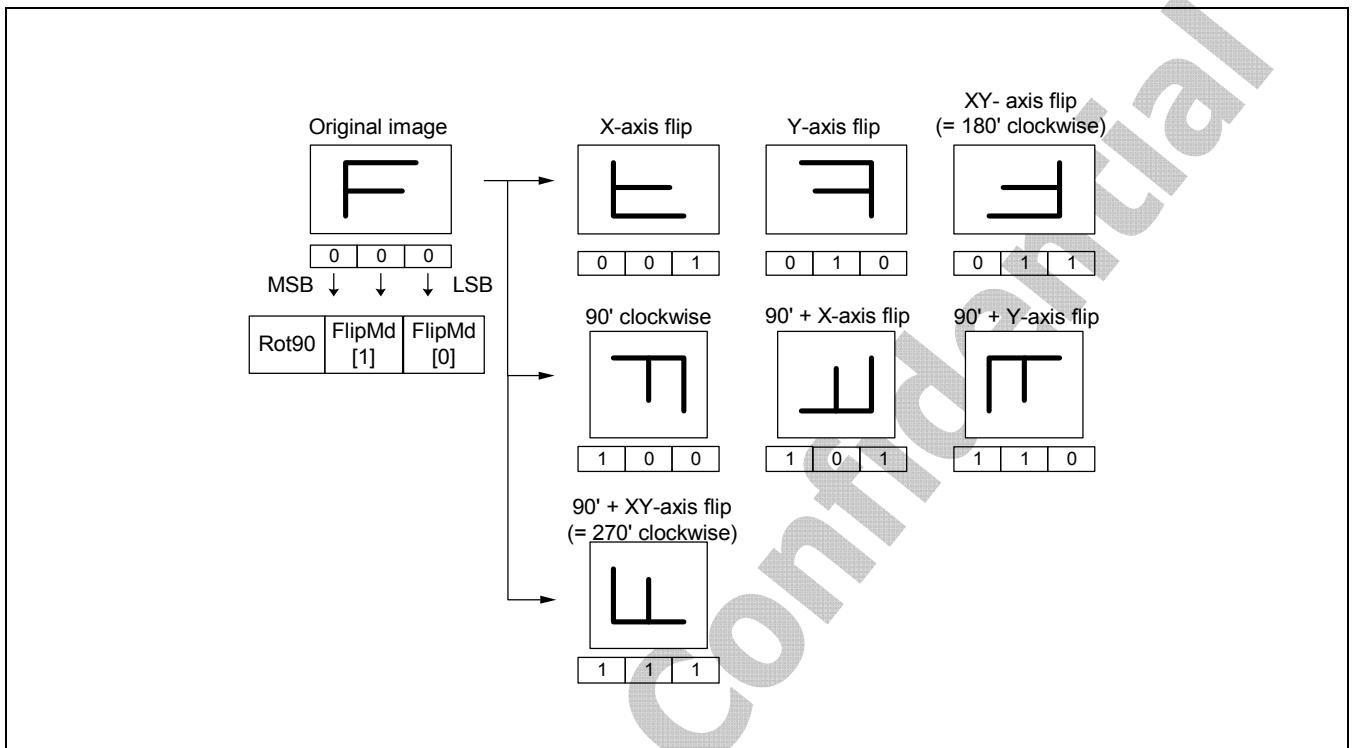


Figure 2-23 Image Mirror and Rotation

CITRGFMTn	Bit	Description	Initial State
InRot90	[31]	1 = Rotate clockwise 90° (Using the Input Rotator only). Input Rotator and Output Rotator do not work at the same time. If InRot90 mode is enabled, then output data path should be LCD FIFO path. 0 = Input Rotator bypass (ML=OO)	0
OutFormat	[30:29]	00 = YCbCr 4:2:0 output image format (2 or 3 plane) 01 = YCbCr 4:2:2 output image format (2 or 3 plane) (ref. 2 or 3 plane format register → C_INT_OUT) 10 = YCbCr 4:2:2 output image format (1 plane) 11 = RGB output image format (Ref. RGB format register → OutRGB_FMT). Note: Refer to gathering extension register for format YCbCr444. (ML=OO)	0
TargetHsize	[28:16]	Specifies the horizontal pixel number of target image. Refer to gathering extension register (TargetHsize_ext). Note.) CAMIF0 and CAMIF2 : In case of output DMA, TargetHsize should be multiple of 16. CAMIF1 : In case of interlaced output DMA and 90-degree-rotation, TargetVsize should be more than 16 (ML=OO)	0
OutFlipMd	[15:14]	Specifies image mirror and rotation for output DMA. 00 = Normal 01 = X-axis mirror 10 = Y-axis mirror 11 = 180° rotation Note) User cannot use this function, If input format is among CAM_JPEG or MIPI RAW or MIPI User-defined format. (ML=OO)	0
OutRot90	[13]	1 = Rotate clockwise 90° (Using the Output Rotator) 0 = Output Rotator bypass Note) User cannot use this function, If input format is among CAM_JPEG or MIPI RAW or MIPI User-defined format. (ML=OO)	0

CITRGFMTn	Bit	Description	Initial State
TargetVsize	[12:0]	<p>Specifies vertical pixel number of target image. The minimum number is 4. Refer to gathering extension register (TargetVsize_ext).</p> <p>CAMIF0 and CAMIF2 :</p> <p>In case of output DMA and 90-degree-rotation, TargetVsize should be multiple of 16.</p> <p>In case of interlaced output DMA and 90-degree-rotation, TargetVsize should be multiple of 32.</p> <p>CAMIF1</p> <p>In case of interlaced output DMA and 90-degree-rotation, TargetVsize should be more than 32.</p> <p>(ML=OO)</p>	0

TargetHsize and TargetVsize should not be larger than Camera SourceHsize and Camera SourceVsize. InputDMA source size does not care.

Caution: Only input rotator supports InputDMA image data. The output rotator supports Camera or InputDMA image data. Input and output rotators should not work at the same time because input and output rotator memories are shared for saving the memory size.

NOTE: If the TargetVsize value is set to an odd number (N) when output format is YCbCr 4:2:0, the odd numbers (N) of Y lines and (N-1)/2 of Cb, Cr lines are generated. Also, X-flip or XY-flip is not allowed. Thus, YCbCr 4:2:0 output format should use an even TargetVsize number.

NOTE: If TargetVsize value cannot be divided by 4 ($4n+1, 4n+2, 4n+3$) when output format is YCbCr 4:2:0 and Interlaced out, The odd number(N) of Y lines and the (N-1)/2 of Cb, Cr lines are generated. Also, X-flip or XY-flip are not allowed. Thus YCbCr 4:2:0 ouput format and Interlaced out should use 4's multiple TargetVsize number.

2.8.1.18 Output DMA Control Register (CIOCTRLn)

- CIOCTRL0, R/W, Address = 0xFB20_004C
- CIOCTRL1, R/W, Address = 0xFB30_004C
- CIOCTRL2, R/W, Address = 0xFB40_004C

CIOCTRLn	Bit	Description	Initial State															
Weave_out	[31]	<p>Even and Odd fields can be weaved together and combined to form a complete progressive frame by hardware. This field is useful for interlace DMA output mode (Interlace_out or CAM_INTERLACE). Even field address (1st frame start address) is used weave address. Odd fields address (2nd frame start address) is ignored.</p> <p>1 = Weave 2 = Normal (ML=0X)</p>	0															
Reserved	[30:26]	Reserved	0															
Order2p_out	[25:24]	<p>Specifies YCbCr 4:2:0 or 4:2:2 2plane output Chroma memory storing style order (should be C_INT_OUT = 1).</p> <table border="1"> <tr> <th>bit</th><th>MSB</th><th>LSB</th></tr> <tr> <td>00</td><td>Cr3Cb3Cr2Cb2Cr1Cb1Cr0Cb0</td><td></td></tr> <tr> <td>01</td><td>Cb3Cr3Cb2Cr2Cb1Cr1Cb0Cr0</td><td></td></tr> <tr> <td>10</td><td>Reserved</td><td></td></tr> <tr> <td>11</td><td>Reserved</td><td></td></tr> </table> <p>(ML=OO)</p>	bit	MSB	LSB	00	Cr3Cb3Cr2Cb2Cr1Cb1Cr0Cb0		01	Cb3Cr3Cb2Cr2Cb1Cr1Cb0Cr0		10	Reserved		11	Reserved		0
bit	MSB	LSB																
00	Cr3Cb3Cr2Cb2Cr1Cb1Cr0Cb0																	
01	Cb3Cr3Cb2Cr2Cb1Cr1Cb0Cr0																	
10	Reserved																	
11	Reserved																	
Reserved	[23:4]	Reserved	0															
C_INT_OUT	[3]	<p>1 = YCbCr 4:2:0 or 4:2:2 2plane output format 0 = YCbCr 4:2:0 or 4:2:2 3plane output format (ML=OO)</p>	0															
LastIRQEn	[2]	<p>1 = enables last IRQ at the end of frame capture (It is recommended to check the done signal of capturing image for JPEG) 0 = normal (ML=XX)</p>	0															
Order422_out	[1:0]	<p>Specifies YCbCr 4:2:2 1plane output memory storing style order.</p> <table border="1"> <tr> <th>bit</th><th>MSB</th><th>LSB</th></tr> <tr> <td>00</td><td>Cr1Y3Cb1Y2Cr0Y1Cb0Y0</td><td></td></tr> <tr> <td>01</td><td>Cb1Y3Cr1Y2Cb0Y1Cr0Y0</td><td></td></tr> <tr> <td>10</td><td>Y3Cr1Y2Cb1Y1Cr0Y0Cb0</td><td></td></tr> <tr> <td>11</td><td>Y3Cb1Y2Cr1Y1Cb0Y0Cr0</td><td></td></tr> </table> <p>(ML=OO)</p>	bit	MSB	LSB	00	Cr1Y3Cb1Y2Cr0Y1Cb0Y0		01	Cb1Y3Cr1Y2Cb0Y1Cr0Y0		10	Y3Cr1Y2Cb1Y1Cr0Y0Cb0		11	Y3Cb1Y2Cr1Y1Cb0Y0Cr0		0
bit	MSB	LSB																
00	Cr1Y3Cb1Y2Cr0Y1Cb0Y0																	
01	Cb1Y3Cr1Y2Cb0Y1Cr0Y0																	
10	Y3Cr1Y2Cb1Y1Cr0Y0Cb0																	
11	Y3Cb1Y2Cr1Y1Cb0Y0Cr0																	

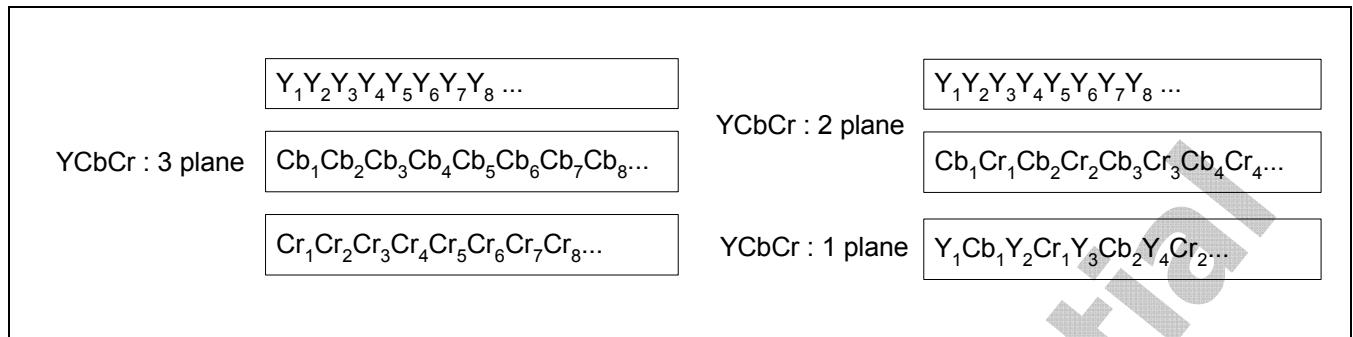


Figure 2-24 YCbCr Plane Memory Storing Style

2.8.2 REGISTER SETTING GUIDE FOR SCALER

SRC_Width and DST_Width satisfy the double word (8-bytes) boundary constraints such that the number of horizontal pixel represents kn, where n = 1, 2, 3 ... and k = 1/ 2/ 8 for 24bpp RGB/ 16bpp RGB/ YCbCr420 image.

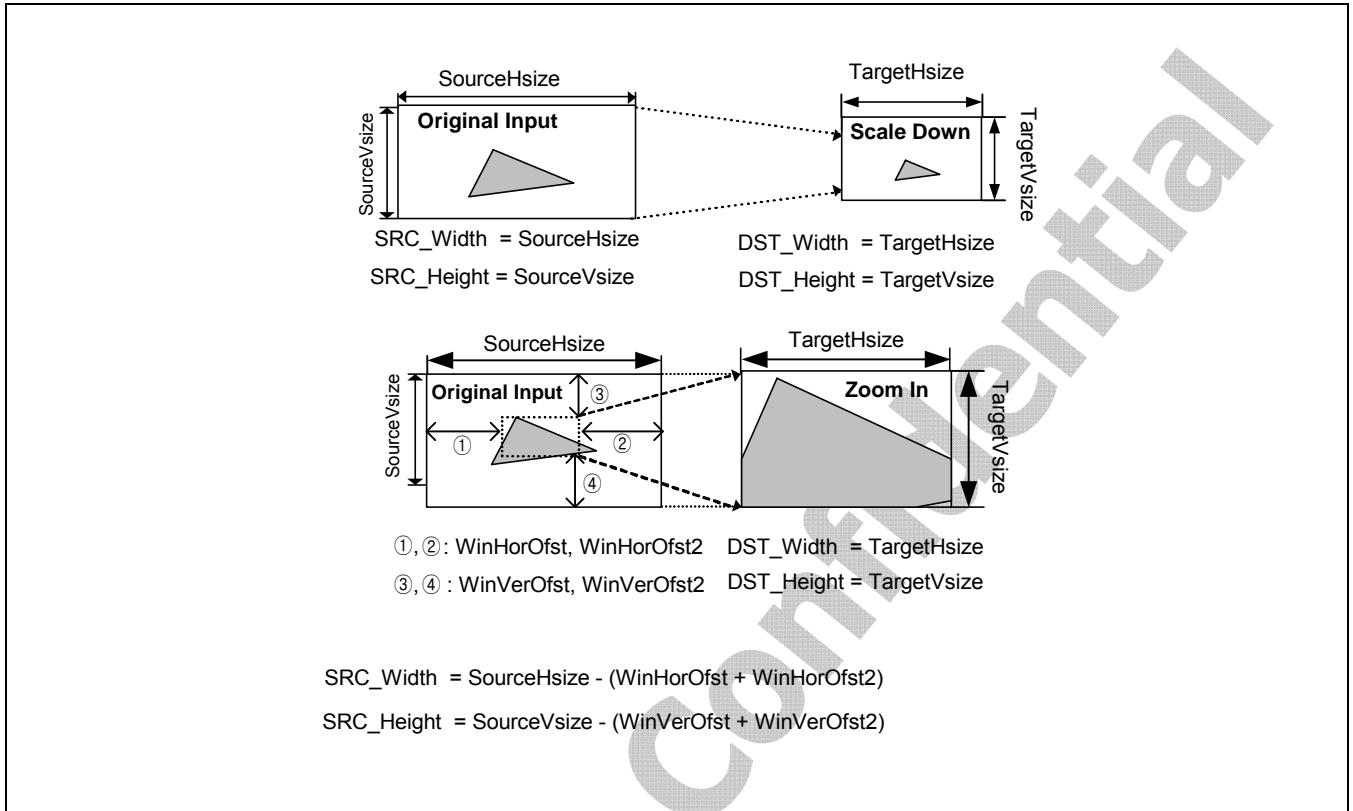


Figure 2-25 Scaling Scheme

The other control registers of pre-scaled image size, pre-scale ratio, pre-scale shift ratio, and main scale ratio are defined according to the following equations:

- If (SRC_Width >= 64 × DST_Width) { Exit(-1); /* Out Of Horizontal Scale Range */ }
- else if (SRC_Width >= 32 × DST_Width) { PreHorRatio = 32; H_Shift = 5 }
- else if (SRC_Width >= 16 × DST_Width) { PreHorRatio = 16; H_Shift = 4 }
- else if (SRC_Width >= 8 × DST_Width) { PreHorRatio = 8; H_Shift = 3 }
- else if (SRC_Width >= 4 × DST_Width) { PreHorRatio = 4; H_Shift = 2 }
- else if (SRC_Width >= 2 × DST_Width) { PreHorRatio = 2; H_Shift = 1 }
- else { PreHorRatio = 1; H_Shift = 0 }

$$\text{PreDstWidth} = \text{SRC_Width} / \text{PreHorRatio}$$

CAMIF0 & CAMIF2	$\text{MainHorRatio} = (\text{SRC_Width} \ll 8) / (\text{DST_Width} \ll \text{H_Shift})$
CAMIF1	$\text{MainHorRatio} = (\text{SRC_Width} \ll 14) / (\text{DST_Width} \ll \text{H_Shift})$

- If (SRC_Height >= 64 × DST_Height) { Exit(-1); /* Out Of Vertical Scale Range */ }
- else if (SRC_Height >= 32 × DST_Height) { PreVerRatio = 32; V_Shift = 5 }
- else if (SRC_Height >= 16 × DST_Height) { PreVerRatio = 16; V_Shift = 4 }
- else if (SRC_Height >= 8 × DST_Height) { PreVerRatio = 8; V_Shift = 3 }
- else if (SRC_Height >= 4 × DST_Height) { PreVerRatio = 4; V_Shift = 2 }
- else if (SRC_Height >= 2 × DST_Height) { PreVerRatio = 2; V_Shift = 1 }
- else { PreVerRatio = 1; V_Shift = 0 }

$$\text{PreDstHeight} = \text{SRC_Height} / \text{PreVerRatio}$$

CAMIF0 & CAMIF2	$\text{MainVerRatio} = (\text{SRC_Height} \ll 8) / (\text{DST_Height} \ll \text{V_Shift})$
CAMIF1	$\text{MainVerRatio} = (\text{SRC_Height} \ll 14) / (\text{DST_Height} \ll \text{V_Shift})$

$$\text{SHfactor} = 10 - (\text{H_Shift} + \text{V_Shift})$$

Caution: In case of Zoom-in, you should check the next equation (CAM-In case).
 $((\text{SourceHsize} - (\text{WinHorOfst} + \text{WinHorOfst2})) / \text{PreHorRatio}) \leq \text{Max. scaler line buffer size width}$

2.8.2.1 Pre-Scaler Control Register 1 (CISCPRERATIOn)

- CISCPRERATIO0, R/W, Address = 0xFB20_0050
- CISCPRERATIO1, R/W, Address = 0xFB30_0050
- CISCPRERATIO2, R/W, Address = 0xFB40_0050

CISCPRERATIOn	Bit	Description	Initial State
SHfactor	[31:28]	Specifies the shift factor for pre-scaler. (ML=OO)	0
Reserved	[27:23]	Reserved	0
PreHorRatio	[22:16]	Specifies the horizontal ratio of pre-scaler. (ML=OO)	0
Reserved	[15:7]	Reserved	0
PreVerRatio	[6:0]	Specifies the vertical ratio of pre-scaler. (ML=OO)	0

2.8.2.2 Pre-Scaler Control Register 2 (CISCPREDSTn)

- CISCPREDST0, R/W, Address = 0xFB20_0054
- CISCPREDST1, R/W, Address = 0xFB30_0054
- CISCPREDST2, R/W, Address = 0xFB40_0054

CISCPREDSTn	Bit	Description	Initial State
Reserved	[31:30]	Reserved	0
PreDstWidth	[29:16]	Specifies the destination width for pre-scaler. (ML=OO)	0
Reserved	[15:14]	Reserved	0
PreDstHeight	[13:0]	Specifies the destination height for pre-scaler. (ML=OO)	0

2.8.2.3 Main-Scaler Control Register (CISCCTRLn)

- CISCCTRL0, R/W, Address = 0xFB20_0058
- CISCCTRL1, R/W, Address = 0xFB30_0058
- CISCCTRL2, R/W, Address = 0xFB40_0058

CISCCTRLn	Bit	Description	Initial State
ScalerBypass	[31]	<p>Specifies scaler bypass. In this case, ImgCptEn_SC should be 0, but ImgCptEn should be 1.</p> <p>Generally, this mode is used to handle a large image whose size is greater than the maximum size of scaler. (This mode is intended to capture the JPEG input image for DSC application).</p> <p>In this case, the input pixel buffering depends on only the input FIFOs; therefore, the system bus should not be busy in this mode.</p> <p>ScalerBypass has certain restrictions. For instance, size scaling, color space conversion, input DMA mode, Write Back mode and RGB format are not allowed. If input format is YCbCr4:2:2, the output format should also be YCbCr4:2:2 or YCbCr4:2:0.</p> <p>(ML=XX)</p>	0
ScaleUp_H	[30]	<p>Specifies horizontal scale up/ down flag for scaler (In 1:1 scale ratio, this bit should be “1”.)</p> <p>1 = Up 0 = Down (ML=OO)</p>	0
ScaleUp_V	[29]	<p>Specifies vertical scale up/down flag for scaler (In 1:1 scale ratio, this bit should be “1”).</p> <p>1 = Up 0 = Down (ML=OO)</p>	0
CSCR2Y	[28]	<p>Selects YCbCr data dynamic range for color space conversion from RGB to YCbCr.</p> <p>1 = Wide => Y/Cb/Cr (0 ~ 255): Wide default 0 = Narrow => Y (16 ~ 235), Cb/Cr (16 ~ 240)</p> <p>* Recommended CSC range setting</p> <p>CSCR2Y= CSCY2R (Wide=Wide or Narrow=Narrow) (ML=OO)</p>	1
CSCY2R	[27]	<p>Specifies YCbCr data dynamic range selection for the color space conversion from YCbCr to RGB.</p> <p>1 = Wide => Y/Cb/Cr (0 ~ 255): Wide default 0 = Narrow => Y (16 ~ 235), Cb/Cr (16 ~ 240) (ML=OO)</p>	1
LCDPathEn	[26]	<p>Enables FIFO mode.</p> <p>1 = Enables for FIFO output 0 = Enables for DMA output.</p>	0

CISCCTRLn	Bit	Description	Initial State
		If FIFO mode is enabled, then the input mode should be DMA input and WSWP bit at WINCON0(0xF800_0020) in LCD controller should be '0'. The FIFO mode output format is YCbCr4;4:4 3 Plane or RGB 24-bit. Its selection depends on OutFormat register. OutFormat = RGB → RGB24bit. Other setting means YCbCr4:4:4. If interlace out end DMA input want to set together, Output mode should be FIFO output (ML=OO)	
Interlace	[25]	Output scan method selection register. (RAW and JPEG input formats are not available) 1 : Interlace scan out (Input data should be progressive mode) 0 : progressive scan out Note) If this bit is configured by 0 for interlaced input, the output is also interlaced format not converted into progressive one. (ML=OX)	0
MainHorRatio	[24:16]	Specifies horizontal scale ratio for main-scaler. Note) CAMIF1 : Refer to the gathering extension register(MainHorRatio_ext). (ML=OO)	0
ScalerStart	[15]	Specifies the Scaler start. 1 = Scaler start 0 = Scaler stop or scaler bypass (ML=OO)	0
InRGB_FMT	[14:13]	Specifies input DMA RGB format. 00 = RGB565, 01 = RGB666 10 = RGB888, 11 = Reserved (ML=OX)	0
OutRGB_FMT	[12:11]	Specifies output DMA RGB format. 00 = RGB565 01 = RGB666 10 = RGB888 11 = Reserved (ML=OO)	0
Ext_RGB	[10]	Specifies input RGB data extension enable bit for conversion of RGB565/666 mode to RGB888 mode. 1 = Extension 0 = normal i) Input R = 5-bit in RGB565 mode 10100 -> 10100101 (Extension): [7]=[2], [6]=[1], [5]=[0] 10100 -> 10100000 (Normal) ii) Input R = 6-bit in RGB666 mode	0

CISCTRLn	Bit	Description	Initial State
		101100 -> 10110010 (Extension): [7]=[1], [6]=[0] 101100 -> 10110000 (Normal) (ML=OO)	
One2One	[9]	Scaler does not run interpolation, but runs repetition for upsampling. Sometimes other IP needs this method if input format are YCbCr4:2:0 and YCbCr4:2:2. Scalerbypass should be set to '0' and don't care plane. Caution!: One2One should be used if input/ output format and size are same. One2One function has size constraints, as described in Table 9.2-1. For example: input YCbCr4:2:0 2plane -> output YCbCr4:2:0 3plane (O.K) (ML=OO)	0
MainVerRatio	[8:0]	Specifies vertical scale ratio for main-scaler. Note) CAMIF1 : Refer to the gathering extension register(MainVerRatio_ext) (ML=OO)	0

Table 2-6 Color Space Conversion Equations

	Wide	Narrow
CSCY2R (601)	$R = Y + 1.371(Cr-128)$ $G = Y - 0.698(Cr-128) - 0.336(Cb-128)$ $B = Y + 1.732(Cb-128)$	$R = 1.164(Y-16) + 1.596(Cr-128)$ $G = 1.164(Y-16) - 0.813(Cr-128) - 0.391(Cb-128)$ $B = 1.164(Y-16) + 2.018(Cb-128)$
CSCY2R (709)	$R = Y + 1.540(Cr-128)$ $G = Y - 0.459(Cr-128) - 0.183(Cb-128)$ $B = Y + 1.816(Cb-128)$	$R = 1.164(Y-16) + 1.793(Cr-128)$ $G = 1.164(Y-16) - 0.534(Cr-128) - 0.213(Cb-128)$ $B = 1.164(Y-16) + 2.115(Cb-128)$
CSCR2Y (601)	$Y = 0.299R + 0.587G + 0.114B$ $Cb = -0.172R - 0.339G + 0.511B + 128$ $Cr = 0.511R - 0.428G - 0.083B + 128$	$Y = 0.257R + 0.504G + 0.098B + 16$ $Cb = -0.148R - 0.291G + 0.439B + 128$ $Cr = 0.439R - 0.368G - 0.071B + 128$
CSCR2Y (709)	$Y = 0.213R + 0.715G + 0.072B$ $Cb = -0.117R - 0.394G + 0.511B + 128$ $Cr = 0.511R - 0.464G - 0.047B + 128$	$Y = 0.183R + 0.614G + 0.062B + 16$ $Cb = -0.101R - 0.338G + 0.439B + 128$ $Cr = 0.439R - 0.399G - 0.040B + 128$

DMA Mode Operation (Normal mode): DMA Input → DMA Output

The source image format is in one of the following formats: YCbCr420, YCbCr422, YCbCr444 and RGB16-/ 18-/ 24-bit. On the other hand, the destination image format is in one of the following ones: YCbCr420, YCbCr422, YCbCr444 and RGB 16-/18-/24-bit. (Input and output format are possible for both Progressive and Interlace format).

All source and destination image data need to be stored in memory system aligned with double word boundary and should support DMA operation. Therefore, the width of source and destination image should satisfy the double word boundary condition.

FIFO Mode Operation: DMA Input → FIFO Output

In FIFO Mode, (LCDPathEn =1), two types of color space conversion are available such as RGB2YCbCr and YCbCr2RGB. This is similar to DMA mode operation. The destination image is transferred to FIFO in display controller (or some other IP with FIFO interface) without additional memory bandwidth such as CAMIF-to-Memory and Memory-to-Display Controller. Output data format is determined only by Output format register: OutFormat = RGB format (24-bit RGB) or OutFormat=YCbCr format(YCbCr444). The source image format and destination image format restrictions are described in the table below.

Table 2-7 FIFO Mode Image Format

DMA input (Progressive / Interlace)		FIFO output (Progressive / Interlace)
YCbCr	YCbCr420: 3/2 plane	YCbCr 444 3 plane or RGB 24-bit
	YCbCr422: 3/2/1 plane	
	YCbCr444: 3/2 plane	
RGB	RGB 16-/ 18-/ 24-bit	

In FIFO mode (LCDPathEnable =1), you can either select progressive or interlace scan mode based on the

“interlace” control register, Register Files Lists.

The “interlace” control bit is available if LCDPathEn=1, otherwise its value is unaffected by DMA mode operation, which supports only progressive scan mode. Even if an interlaced scan mode is enabled (LCDPathEn = 1 and Interlace = 1), per frame management, which consists of even field and odd filed, is automatic. This means that user interruption is unnecessary to interfield switching in the same frame. Therefore, the frame management scheme is identical for both progressive and interlaced scan modes. Interlace is not supported if camera processor selects the input data

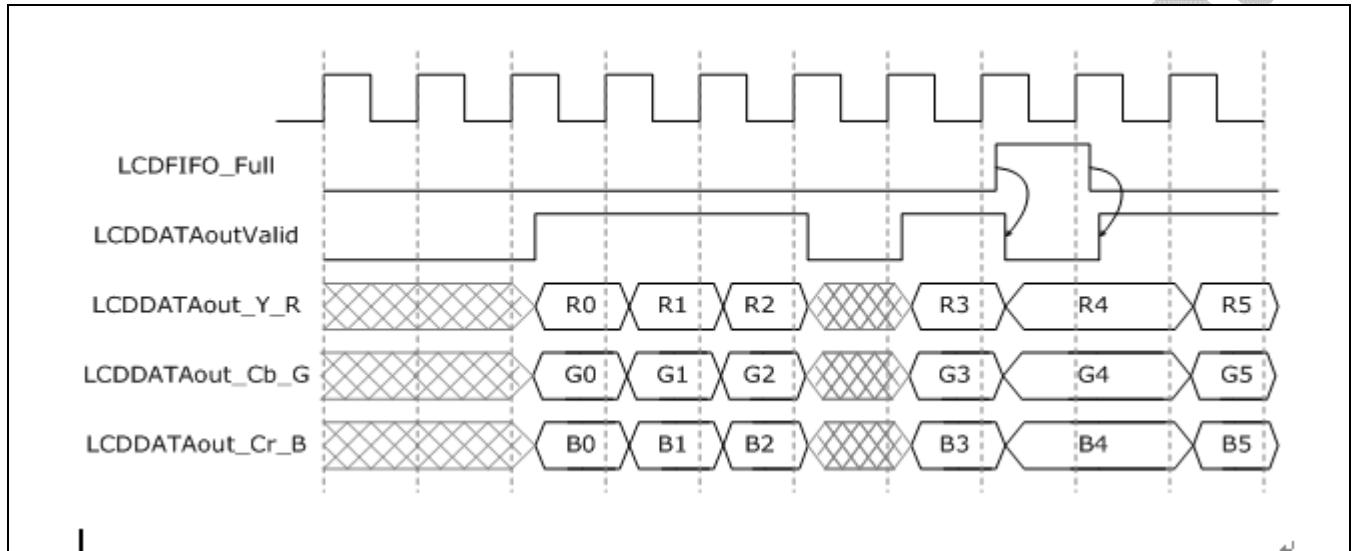


Figure 2-26 I/O Timing Diagram for Direct Path

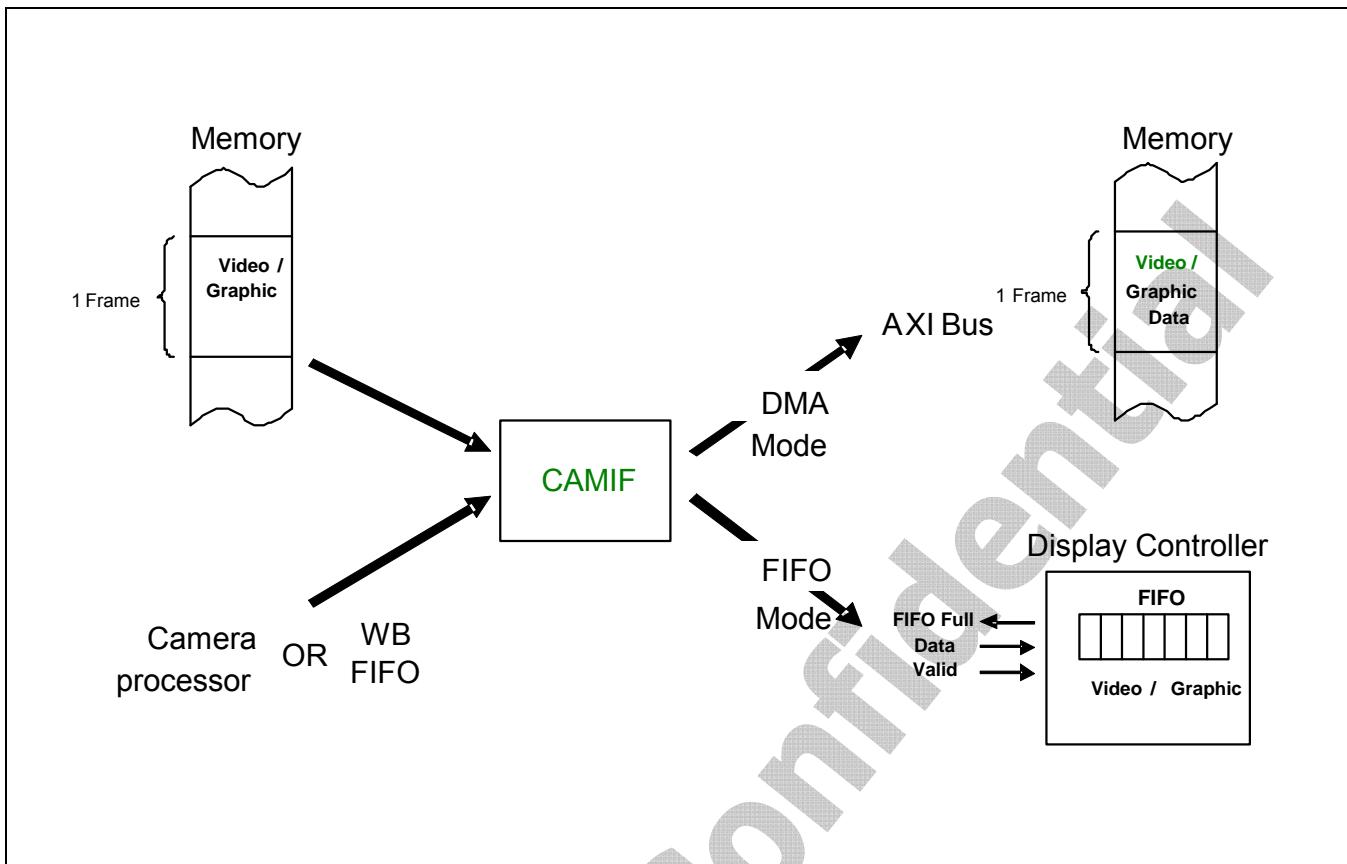


Figure 2-27 Input & Output Modes in CAMIF

2.8.2.4 Output DMA Target Area Register (CITAREAn)

- CITAREA0, R/W, Address = 0xFB20_005C
- CITAREA1, R/W, Address = 0xFB30_005C
- CITAREA2, R/W, Address = 0xFB40_005C

CITAREAn	Bit	Description	Initial State
Reserved	[31:28]	Reserved	0
CITAREA	[27:0]	Specifies target area for output DMA = Target H size x Target V size. (ML=OO)	0

2.8.2.5 Output DMA Line Skip Register (CIOLINESKIPn)

- CIOLINESKIP0, R/W, Address = 0xFB20_0060
- CIOLINESKIP1, R/W, Address = 0xFB30_0060
- CIOLINESKIP2, R/W, Address = 0xFB40_0060

CIOLINESKIPn	Bit	Description	Initial State
Reserved	[31:24]	Reserved	0
OLINESKIP_Cr	[23:20]	Specifies Cr Line skip for output DMA. If OLINESKIP_Cr is k, Cr Line is stored in every k+1 line. Note) Maximum value can be 8 (ML=OO)	0
Reserved	[19:14]	Reserved	0
OLINESKIP_Cb	[13:10]	Specifies Cb Line skip for output DMA. If OLINESKIP_Cb is k, Cb Line is stored in every k+1 line. Note) Maximum value can be 8 (ML=OO)	0
Reserved	[9:4]	Reserved	0
OLINESKIP_Y	[3:0]	Specifies Y Line skip for output DMA. If OLINESKIP_Y is k, Y Line is stored in every k+1 line. Note) Maximum value can be 8 (ML=OO)	0

2.8.2.6 Status Register (CISTATUSn)

- CISTATUS0, Address = 0xFB20_0064
- CISTATUS1, Address = 0xFB30_0064
- CISTATUS2, Address = 0xFB40_0064

CISTATUSn	Bit	Description	RW	Initial State
OvFiY	[31]	Specifies overflow state of FIFO Y. (ML=XX)	R	0
OvFiCb	[30]	Specifies overflow state of FIFO Cb. (ML=XX)	R	0
OvFiCr	[29]	Specifies overflow state of FIFO Cr. (ML=XX)	R	0
VSYNC	[28]	Specifies camera VSYNC (CPU refers this bit for first SFR setting after external camera muxing. It is seen in the ITU-R BT 656 mode). (ML=XX)	R	0
FrameCnt	[27:26]	Specifies frame count of output DMA. This counter value specifies the next frame number. (ML=XX)	R	0
WinOfstEn	[25]	Specifies window offset enable status. (ML=XX)	R	0
FlipMd	[24:23]	Specifies flip mode of output DMA. (ML=XX)	R	0
ImgCptEn	[22]	Specifies image capture enable of global camera interface. (ML=XX)	R	0
ImgCptEn_SC	[21]	Specifies image capture enable of scaler path. (ML=XX)	R	0
VSYNC_A	[20]	Specifies external camera A VSYNC. Polarity inversion is not adopted. (ML=XX)	R	X
VSYNC_B	[19]	Specifies external camera B VSYNC. Polarity inversion is not adopted. (ML=XX)	R	X
OvRLB	[18]	Specifies overflow status of line buffer for rotation. (ML=XX)	R	0
FrameEnd	[17]	If the frame operation finishes, then FrameEnd is generated and FrameEnd is cleared by setting '0'. (ML=XX)	R/W	0
LastCaptureEnd	[16]	Specifies last frame capture status. LastCaptureEnd is cleared by setting '0'. This signal is applied only by camera input mode. (ML=XX)	R/W	0

CISTATUSn	Bit	Description	RW	Initial State
VVALID_A	[15]	Specifies external camera A VVALID. (ML=XX)	R	X
VVALID_B	[14]	Specifies external camera B VVALID. (ML=XX)	R	X
IRQ_CAM	[13]	Specifies interrupt status for camera input mode. (ML=XX)	R	0
IRQ_DMAend	[12]	Specifies interrupt status for DMA frame end in DMA input mode. (ML=XX)	R	0
FrameCptStatus	[11]	Specifies capture frame control status. 1 = Enables present capture 0 = Disables present capture (ML=XX)	R	0
FrameFieldStatus	[10]	Specifies ITU camera field status and internal value after inverse polarity. 1 = Present frame Field1 0 = Present frame Field0 (ML=XX)	R	0
LCD_ENSTATUS	[9]	Specifies LCD controller enable status. 1 = Enables 0 = Disables (ML=XX)	R	0
ENVID_STATUS	[8]	Specifies Input DMA enable internal status. Sometimes this status is used to check whether the software completely clears ENVID bit or not. 1 = Enables Input DMA operation remain 0 = Disables Input DMA operation	R	
Reserved	[7:0]	Reserved	-	0

2.8.2.7 Image Capture Enable Register (CIIMGCPTn)

- CIIMGCPT0, R/W, Address = 0xFB20_00C0
- CIIMGCPT1, R/W, Address = 0xFB30_00C0
- CIIMGCPT2, R/W, Address = 0xFB40_00C0

CIIMGCPTn	Bit	Description	Initial State
ImgCptEn	[31]	Enables camera interface global capture. (ML=XO)	0
ImgCptEn_Sc	[30]	Enables capture for scaler. This bit must be zero in scaler-bypass mode. (ML=OO)	0
Reserved	[29:26]	Reserved	0
Cpt_FrEn	[25]	Controls capture frame (only camera progressive input is applied). 1 = Enables (Step-by-Step frame one shot mode) 0 = Disables (FreeRun mode) Note) User should configure 0 for user-defined packet or CAM_JPEG mode. User should not change this bit under capture enable status. (ML=XO)	0
Reserved	[24]	Reserved	0
Cpt_FrPtr	[23:19]	Captures sequence turnaround pointer. (ML=XX)	0
Cpt_FrMod	[18]	Captures frame control mode. 1 = Applies Cpt_FrCnt mode (Captures Cpt_FrCnt frames along Cpt_FrSeq, after enabling capture DMA frame control. If Cpt_FrCnt = 0, then capture ends.) 0 = Apply Cpt_FrEn mode (Captures frames along Cpt_FrSeq when Cpt_FrEn is high. This sequence repeats until capture frame control is disabled.) (ML=XX)	0
Cpt_FrCnt	[17:10]	Specifies number of frames to be captured. If register reads, then you can see the value of a shadow register, which is downcounted if a frame is captured. In other words, Cpt_FrCnt has an initially loaded value after a frame is captured. Note) User have to disable and enable Cpt_FrEn register before starting CAMIF to use this (capture frame count) funciton (ML=XX)	0
Reserved	[9:0]	Reserved	0

2.8.2.8 Capture Sequence Register (CICPTSEQn)

- CICPTSEQ0, R/W, Address = 0xFB20_00C4
- CICPTSEQ1, R/W, Address = 0xFB30_00C4
- CICPTSEQ2, R/W, Address = 0xFB40_00C4

CICPTSEQn	Bit	Description	Initial State
Cpt_FrSeq	[31:0]	Specifies capture sequence pattern. This register is valid if Cpt_FrEn has a high value. (ML=XX)	FFFF_FFFF

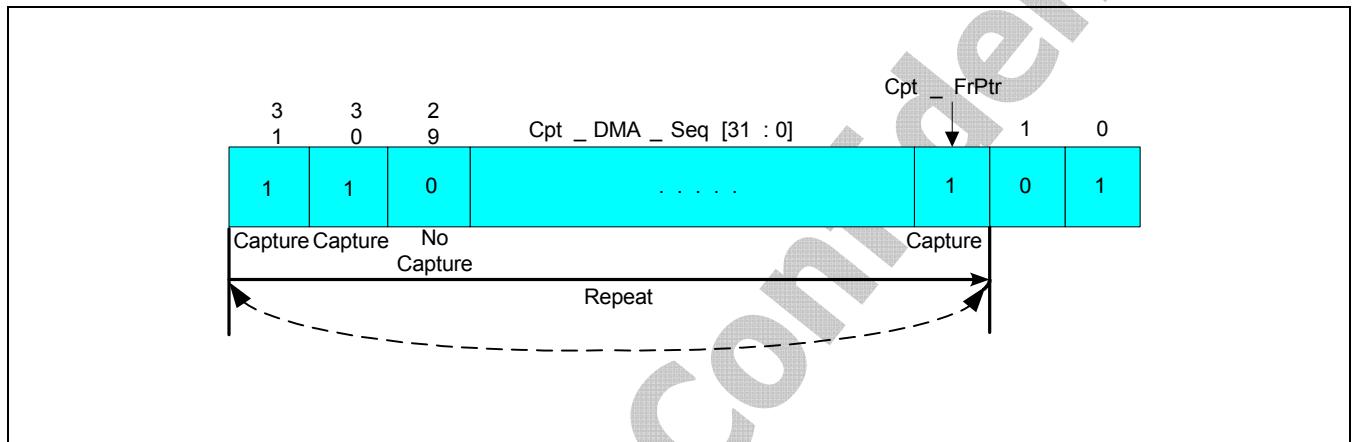


Figure 2-28 Capture Frame Control

※ For skipped frames, IRQ is not generated and FrameCnt is not increased.

2.8.2.9 QoS Threshold Register (CITHOLDn)

- CITHOLD0, R/W, Address = 0xFB20_00C8
- CITHOLD1, R/W, Address = 0xFB30_00C8
- CITHOLD2, R/W, Address = 0xFB40_00C8

CITHOLDn	Bit	Description	Initial State
R_QoS_EN	[31]	1 : QoS Enable for Read DMA channel buffer 0 : QoS Disable for Read DMA channel buffer Read DMA QoS Enable and Write DMA QoS Enable cannot set simultaneously. Only one QoS Enable is possible. (ML=XO)	0
W_QoS_EN	[30]	1 : QoS Enable for Write DMA channel buffer 0 : QoS Disable for Write DMA channel buffer Read DMA QoS Enable and Write DMA QoS Enable cannot set simultaneously. Only one QoS Enable is possible. (ML=XO)	0
Reserved	[29:23]		0
RTh_QoS	[22:16]	Read buffer threshold register.(related Input DMA) If RTh_QoS >= Buffer write count, Read channel is generated no margin signal for bus high performance (ML=XO)	0
Reserved	[15:7]		0
WTh_QoS	[6:0]	Write buffer threshold register.(related Output DMA) If WTh_QoS < Buffer write count, Write channel is generated no margin signal for bus high performance (ML=XO)	0

2.8.2.10 Image Effects Register (CIIMGEFFn)

- CIIMGEFF0, R/W, Address = 0xFB20_00D0
- CIIMGEFF1, R/W, Address = 0xFB30_00D0
- CIIMGEFF2, R/W, Address = 0xFB40_00D0

CIIMGEFFn	Bit	Description	Initial State
Reserved	[31]	Reserved	0
IE_ON	[30]	0 = Disables image effect function 1 = Enables image effect function (ML=OO)	0
IE_AFTER_SC	[29]	Specifies image effect location 1 = After scaling (camera, write back mode and input DMA image are applied except scaler bypass mode) 0 = Before scaling (only ITU camera image are applied) It applies image effect, even though it is in scaler bypass mode. (ML=OO)	0
FIN	[28:26]	Specifies image effect selection. 3'd0 = Bypass 3'd1 = Arbitrary Cb/Cr 3'd2 = Negative 3'd3 = Art Freeze 3'd4 = Embossing 3'd5 = Silhouette (ML=OO)	0
Reserved	[25:21]	Reserved	0
PAT_Cb	[20:13]	Used only for FIN is Arbitrary Cb/Cr (PAT_Cb/Cr == 8'd128 for Grayscale) Wide CSC Range: 0 < PAT_Cb < 255 Narrow CSC Range: 16 ≤ PAT_Cb ≤ 240 (ML=OO)	8'd128
Reserved	[12:8]	Reserved	0
PAT_Cr	[7:0]	Used only for FIN is Arbitrary Cb/Cr (PAT_Cb/Cr == 8'd128 for Grayscale) Wide CSC Range: 0 < PAT_Cr < 255 Narrow CSC Range: 16 ≤ PAT_Cr ≤ 240 (ML=OO)	8'd128

Cf) sepia: PAT_Cb = 8'd115, PAT_Cr = 8'd145

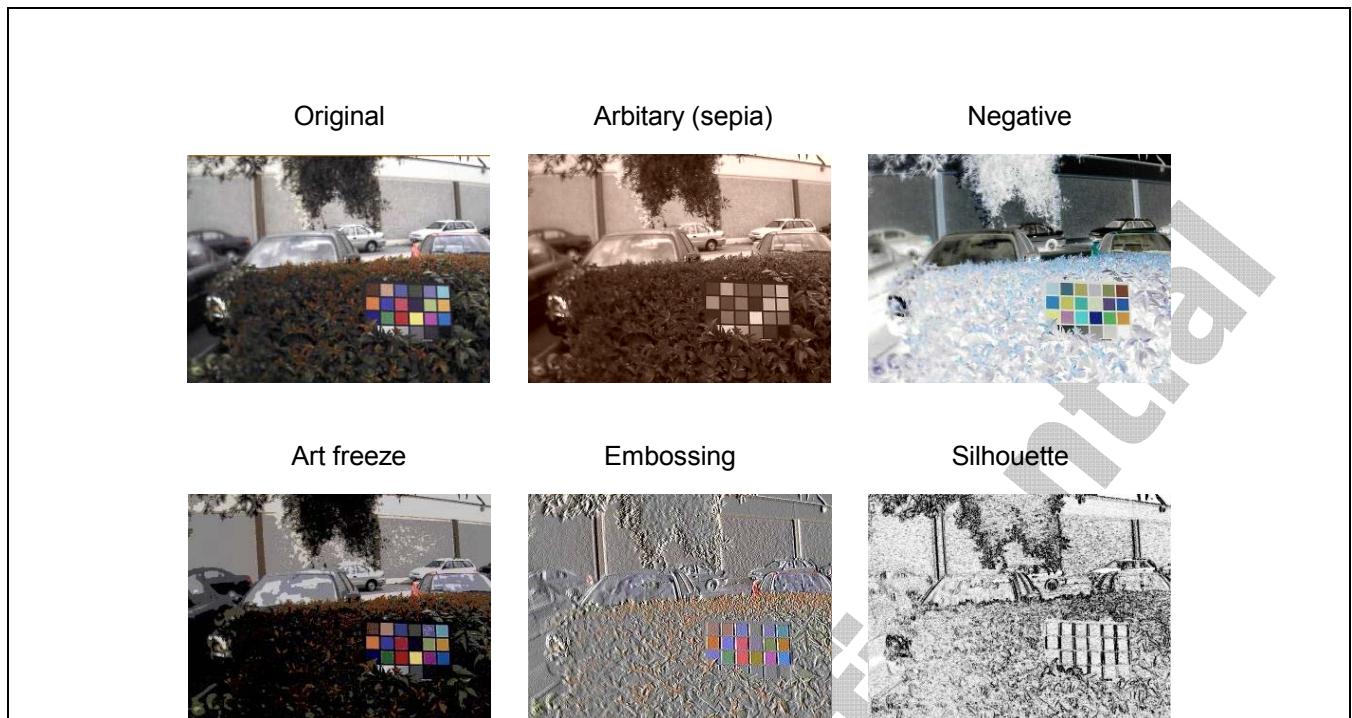


Figure 2-29 Image Effect

2.8.2.11 Input DMA Y0 Start Register (CIYSA0n)

- CIYSA0, R/W, Address = 0xFB20_00D4
- CIYSA1, R/W, Address = 0xFB30_00D4
- CIYSA2, R/W, Address = 0xFB40_00D4

CIYSA0n	Bit	Description	Initial State
CIYSA0	[31:0]	Input format: YCbCr 2/3 plane → Y frame start address Input format: YCbCr 1 plane → YCbCr frame start address Input format: RGB → RGB frame start address Note: In tile mode, this value should be aligned 4Kbytes. That is to say, CIOYSA1[11:0] should be 0x000. (ML=OX)	0

2.8.2.12 Input DMA Cb0 Start Register (CIICBSA0n)

- CIICBSA00, R/W, Address = 0xFB20_00D8
- CIICBSA01, R/W, Address = 0xFB30_00D8
- CIICBSA02, R/W, Address = 0xFB40_00D8

CIICBSA0n	Bit	Description	Initial State
CIICBSA0	[31:0]	Input format: YCbCr 3 plane → Cb frame start address Input format: YCbCr 2 plane → CbCr frame start address Note: In tile mode, this value should be aligned 4Kbytes. That is to say, CIOYSA1[11:0] should be 0x000. (ML=OX)	0

2.8.2.13 Input DMA Cr0 Start Register (CIICRSA0n)

- CIICRSA00, R/W, Address = 0xFB20_00DC
- CIICRSA01, R/W, Address = 0xFB30_00DC
- CIICRSA02, R/W, Address = 0xFB40_00DC

CIICRSA0n	Bit	Description	Initial State
CIICRSA0	[31:0]	Input format: YCbCr 3 plane → Cr frame start address Note: In tile mode, this value should be aligned 4Kbytes. That is to say, CIOYSA1[11:0] should be 0x000. (ML=OX)	0

2.8.2.14 Input DMA Y Line Skip Register (CIILINESKIP_Yn)

- CIILINESKIP_Y0, R/W, Address = 0xFB20_00EC
- CIILINESKIP_Y1, R/W, Address = 0xFB30_00EC
- CIILINESKIP_Y2, R/W, Address = 0xFB40_00EC

CIILINESKIP_Yn	Bit	Description	Initial State
Reserved	[31:28]	Reserved	0
ILINESKIP_Y	[27:24]	Specifies Y Line Skip for Input DMA Note) Maximum value can be 8 (ML=OX)	0
Reserved	[23:0]	Reserved	0

2.8.2.15 Input DMA Cb Line Skip Register (CIILINESKIP_CBn)

- CIILINESKIP_CB0, R/W, Address = 0xFB20_00F0
- CIILINESKIP_CB1, R/W, Address = 0xFB30_00F0
- CIILINESKIP_CB2, R/W, Address = 0xFB40_00F0

CIILINESKIP_CBn	Bit	Description	Initial State
Reserved	[31:28]	Reserved	0
ILINESKIP_Cb	[27:24]	Specifies Cb Line Skip for Input DMA Note) Maximum value can be 8 (ML=OX)	0
Reserved	[23:0]	Reserved	0

2.8.2.16 Input DMA Cr Line Skip Register (CIILINESKIP_CRn)

- CIILINESKIP_CR0, R/W, Address = 0xFB20_00F4
- CIILINESKIP_CR1, R/W, Address = 0xFB30_00F4
- CIILINESKIP_CR2, R/W, Address = 0xFB40_00F4

CIILINESKIP_CRn	Bit	Description	Initial State
Reserved	[31:28]	Reserved	0
ILINESKIP_Cr	[27:24]	Specifies Cr Line Skip for Input DMA Note) Maximum value can be 8 (ML=OX)	0
Reserved	[23:0]	Reserved	0

2.8.2.17 Real Input DMA Size Register (CIREAL_ISIZEn)

- CIREAL_ISIZE0, R/W, Address = 0xFB20_00F8
- CIREAL_ISIZE1, R/W, Address = 0xFB30_00F8
- CIREAL_ISIZE2, R/W, Address = 0xFB40_00F8

CIREAL_ISIZEn	Bit	Description	Initial State
AutoLoadEnable	[31]	<p>Restarts input DMA automatically (only software trigger mode). At the start of first frame, it is required to set ENVID_M start. After the first frame, the next frame does not need ENVID_M setting. If autoload function is running, size and format value should be fixed.</p> <p>0 = Disables AutoLoad 1 = Enables AutoLoad (ML=OX)</p>	0
ADDR_CH_DIS	[30]	<p>Disables input DMA address change (only software trigger mode)</p> <p>At the start of first frame, ADDR_CH_DIS should be equal to '0'.</p> <p>0 = Enables address change 1 = Disables address change (ML=OX)</p>	0
REAL_HEIGHT	[29:16]	<p>Specifies input DMA real image vertical pixel size (Minimum 8).</p> <p>Note)</p> <p>1) 2's multiple : YCbCr 420 2) 4's multiple : Weave-in mode and YCbCr 420 input 3) 2's multiple : Weave-in mode except YCbCr 420 input 4) 2's multiple : Input rotator ON except RGB, YCbCr 444 input</p> <p>Note)</p> <p>Must be multiple of PreVerRatio. If InRot90 = 1, then it must be 16's multiple. Must be 4's multiple of PreHorRatio. Minimum 16. (ML=OX)</p>	0
Reserved	[15:14]	Reserved	0
REAL_WIDTH	[13:0]	<p>Specifies input DMA real image horizontal pixel size.</p> <p>Note)</p> <p>Must be 16's multiple. Must be 4's multiple of PreHorRatio. Minimum 16. If InRot90 = 1, then it must be minimum 8. Must be multiple of PreVerRatio. (ML=OX)</p>	0

2.8.2.18 Input DMA Control Register (MSCTRLn)

- MSCTRL0, Address = 0xFB20_00FC
- MSCTRL1, Address = 0xFB30_00FC
- MSCTRL2, Address = 0xFB40_00FC

MSCTRLn	Bit	Description	R/W	Initial State															
Weave_in	[31]	<p>Even and Odd fields can be read separately from a complete progressive frame. The 1st frame reads even field data and the 2nd frame reads odd field data. When 1st and 2nd frame operation finish, InputDMA is disabled. Both Weave_in and Interlace_out should be set in simultaneous frames. Also, it is recommended that pingpong address should not be changed at Interlace even/ odd field (BC_SEL field should 0).</p> <p>0 = Weave 1 = Normal</p> <p>Note) When using input rotator in Weave_in mode, output horizontal size should be even value. Because vertical data will be converted into horizontal one after rotating. (ML=XX)</p>	R/W	0															
Reserved	[30:28]	Reserved	R/W	0															
Successive_cnt	[27:24]	Specifies input DMA burst successive count (Default is 4 but 3, 2, or 1 are also possible). This value should not be '0'. (ML=OX)	R/W	4'd4															
Reserved	[23:20]	Reserved	R/W	0															
InBuf_Mode	[19]	Specifies input DMA buffer address mode. 1 = Ping-Pong buffer mode (Address 0 and 1 are valid) 0 = Single buffer mode (Only address 0 is valid) (ML=OX)	R/W	0															
Reserved	[18]	Reserved	R/W	0															
Order2p_in	[17:16]	<p>Specifies YCbCr 4:2:0 or 4:2:2 2plane memory reading style order in source input DMA image.</p> <table border="1"> <tr> <th>Bit</th> <th>MSB</th> <th>LSB</th> </tr> <tr> <td>00</td> <td>Cr3Cb3Cr2Cb2Cr1Cb1Cr0Cb0</td> <td></td> </tr> <tr> <td>01</td> <td>Cb3Cr3Cb2Cr2Cb1Cr1Cb0Cr0</td> <td></td> </tr> <tr> <td>10</td> <td>Reserved</td> <td></td> </tr> <tr> <td>11</td> <td>Reserved</td> <td></td> </tr> </table> <p>(ML=OX)</p>	Bit	MSB	LSB	00	Cr3Cb3Cr2Cb2Cr1Cb1Cr0Cb0		01	Cb3Cr3Cb2Cr2Cb1Cr1Cb0Cr0		10	Reserved		11	Reserved		R/W	0
Bit	MSB	LSB																	
00	Cr3Cb3Cr2Cb2Cr1Cb1Cr0Cb0																		
01	Cb3Cr3Cb2Cr2Cb1Cr1Cb0Cr0																		
10	Reserved																		
11	Reserved																		
C_INT_IN	[15]	1 = YCbCr 4:2:0 or 4:2:2 2plane input format 0 = YCbCr 4:2:0 or 4:2:2 3plane input format (ML=OX)	R/W	0															

MSCTRLn	Bit	Description	R/W	Initial State															
InFlipMd	[14:13]	<p>Specifies image mirror and rotation for input DMA.</p> <p>00 = Normal 01 = X-axis mirror 10 = Y-axis mirror 11 = 180° rotation (XY-axis mirror)</p> <p>Note) [Memory to memory path] If input/output image format are all YCbCr 3plane, 'InFlipMd' must be used instead of OutFlipMd. (ML=OX)</p>	R/W	0															
FIFO_CTRL	[12]	<p>Specifies a basis FIFO control of input DMA or Input Rotator.</p> <p>0 = FIFO Full (Next burst transaction is possible except Full FIFO) 1 = FIFO Empty (Next burst transaction is possible when FIFO is empty) (ML=XX)</p>	R/W	0															
Reserved	[11]	Reserved	R/W	0															
BC_SEL	[10]	<p>Selects the input DMA buffer change.</p> <p>0 = Ping-Pong address is changed at interlace even/odd field end. 1 = Ping-Pong address is changed at the frame operation end. (ML=OX)</p>	R/W	0															
Reserved	[9]	Reserved	RW	0															
Buffer_Ptr	[8]	<p>Specifies the input DMA buffer address selection pointer.</p> <p>This register initializes to set the first frame address before starting input DMA.</p> <p>This register should not be written under frame operation.</p> <p>0 = Buffer address 0 1 = Buffer address 1 (ML=OX)</p>	RW	0															
Reserved	[7]	Reserved	RW	0															
EOF_M	[6]	If Input DMA operation is complete, it generates end of frame. (ML=OX)	R	0															
Order422_M	[5:4]	<p>If source input DMA image is 1plane YCbCr 4:2:2, then 1plane YCbCr 4:2:2 inputs memory reading order style.</p> <table border="1"> <tr> <td>bit</td> <td>MSB</td> <td>LSB</td> </tr> <tr> <td>00</td> <td>Y3Cb1Y2Cr1Y1Cb0Y0Cr0</td> <td></td> </tr> <tr> <td>01</td> <td>Cb1Y3Cr1Y2Cb0Y1Cr0Y0</td> <td></td> </tr> <tr> <td>10</td> <td>Y3Cr1Y2Cb1Y1Cr0Y0Cb0</td> <td></td> </tr> <tr> <td>11</td> <td>Cr1Y3Cb1Y2Cr0Y1Cb0Y0</td> <td></td> </tr> </table> <p>(ML=OX)</p>	bit	MSB	LSB	00	Y3Cb1Y2Cr1Y1Cb0Y0Cr0		01	Cb1Y3Cr1Y2Cb0Y1Cr0Y0		10	Y3Cr1Y2Cb1Y1Cr0Y0Cb0		11	Cr1Y3Cb1Y2Cr0Y1Cb0Y0		R/W	0
bit	MSB	LSB																	
00	Y3Cb1Y2Cr1Y1Cb0Y0Cr0																		
01	Cb1Y3Cr1Y2Cb0Y1Cr0Y0																		
10	Y3Cr1Y2Cb1Y1Cr0Y0Cb0																		
11	Cr1Y3Cb1Y2Cr0Y1Cb0Y0																		

MSCTRLn	Bit	Description	R/W	Initial State
SEL_DMA_CAM	[3]	Selects input data selection. 0 = External camera input path 1 = Memory data input path (Input DMA) (ML=OX)	R/W	0
InFormat_M	[2:1]	Specifies the source image format for input DMA. 00 = YCbCr 4:2:0 input image format. (2 or 3 plane) 01 = YCbCr 4:2:2 input image format. (2 or 3 plane) (ref. 2 or 3 plane format register → C_INT_IN) 10 = YCbCr 4:2:2 input image format. (1 plane) 11 = RGB input image format. (ref. RGB format register → InRGB_FMT) Note) Refer to the gathering extension register. YCbCr444_IN (ML=OX)	R/W	0
ENVID_M	[0]	Starts input DMA operation (Software setting triggers low to high). The hardware clears automatically. If data flows from input DMA to local direct FIFO, the software can clear this bit when LCD_ENSTATUS is '0'. 1) SEL_DMA_CAM = '0', ENVID_M don't care (using external camera signal) 2) SEL_DMA_CAM = '1', ENVID_M is set (0→1), then Input DMA operation starts (ML=OX)	R/W	0

NOTE: ENVID_M SFR must be set at the end. Starting order for using DMA input path.

SEL_DMA_CAM (others SFR setting) → Image Capture Enable and Scaler start SFR setting → ENVID_M SFR setting.

- Cf.) Image Capture Enable SFR must be set at the end. Starting order for using Direct FIFO input path.
SEL_DMA_CAM → SelWB_CAMIF (others SFR setting) → Image Capture Enable and Scaler start SFR setting
- Cf.) Image Capture Enable SFR be set at last. Starting order for using camera input path.
SEL_DMA_CAM → SelWB_CAMIF → SelCam_CAMIF (others SFR setting) → Image Capture Enable and Scaler start SFR setting

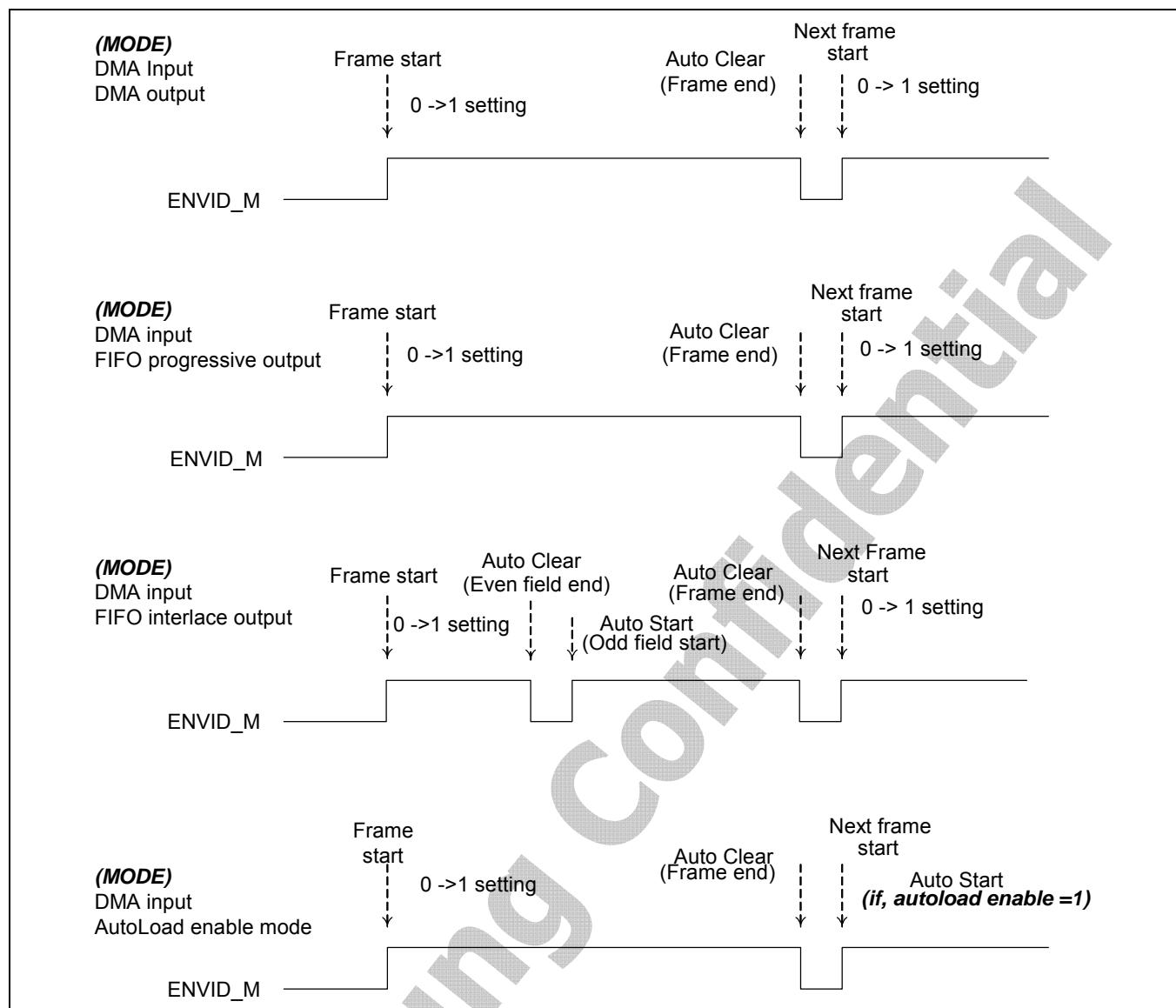


Figure 2-30 ENVID_M SFR Setting When Input DMA Start to Read Memory Data

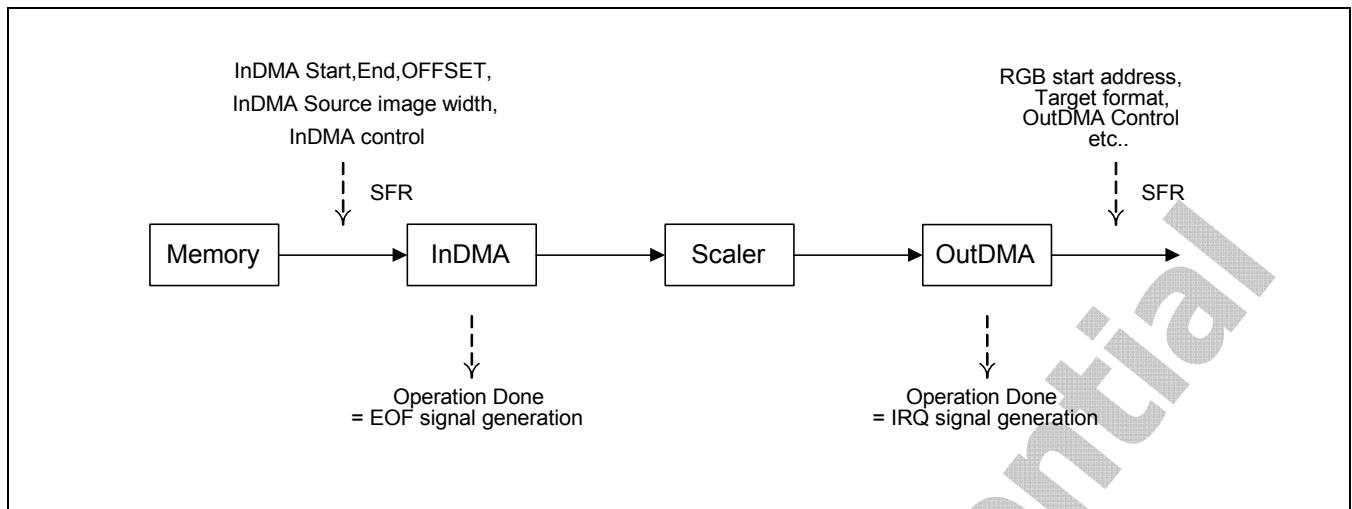


Figure 2-31 SFR and Operation (Related Each DMA When Selected Input DMA Path)

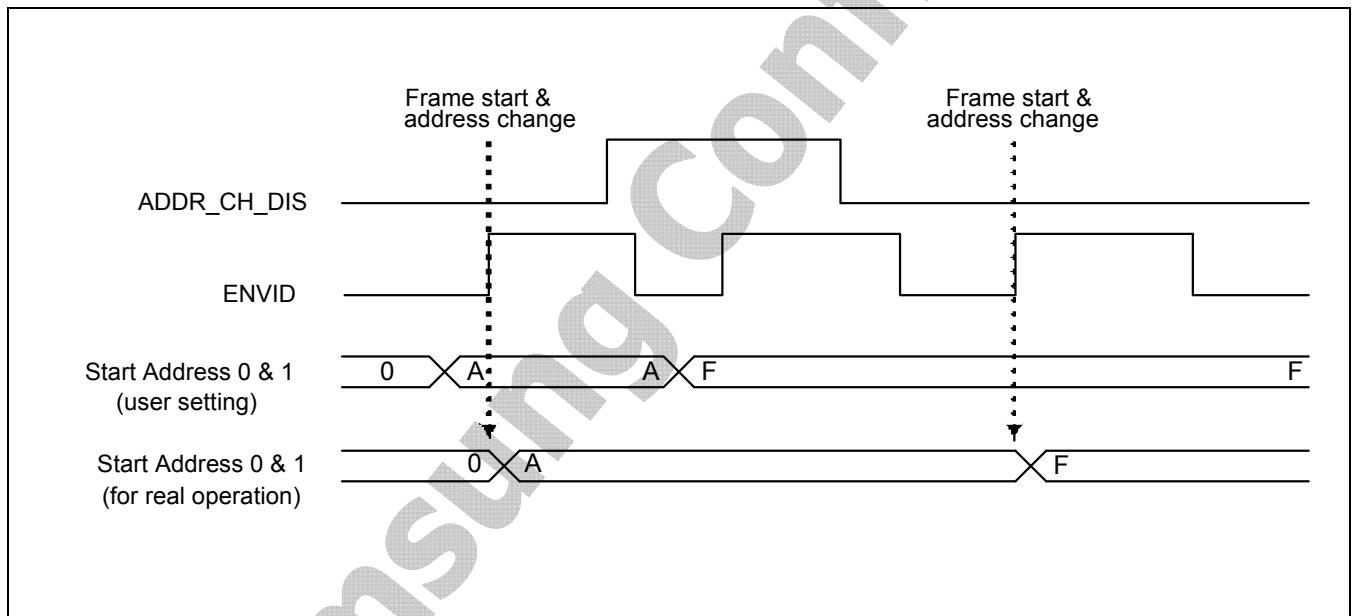


Figure 2-32 Input DMA Address Change Timing (progressive to progressive)

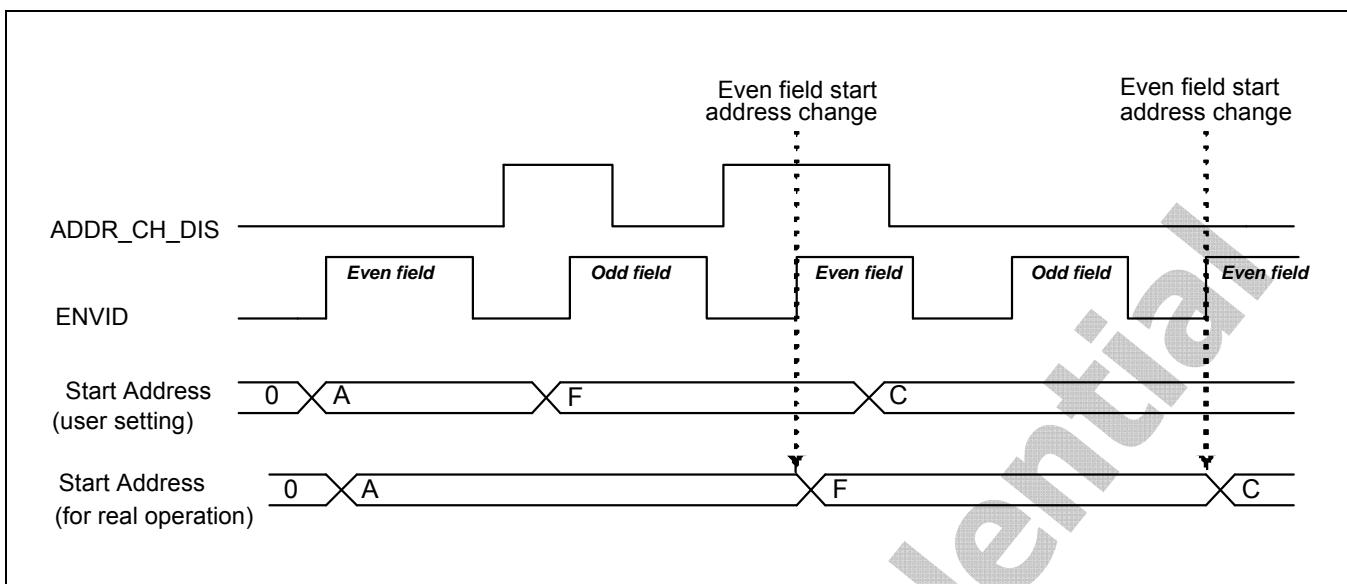


Figure 2-33 Input DMA Address Change Timing (progressive to interlace)

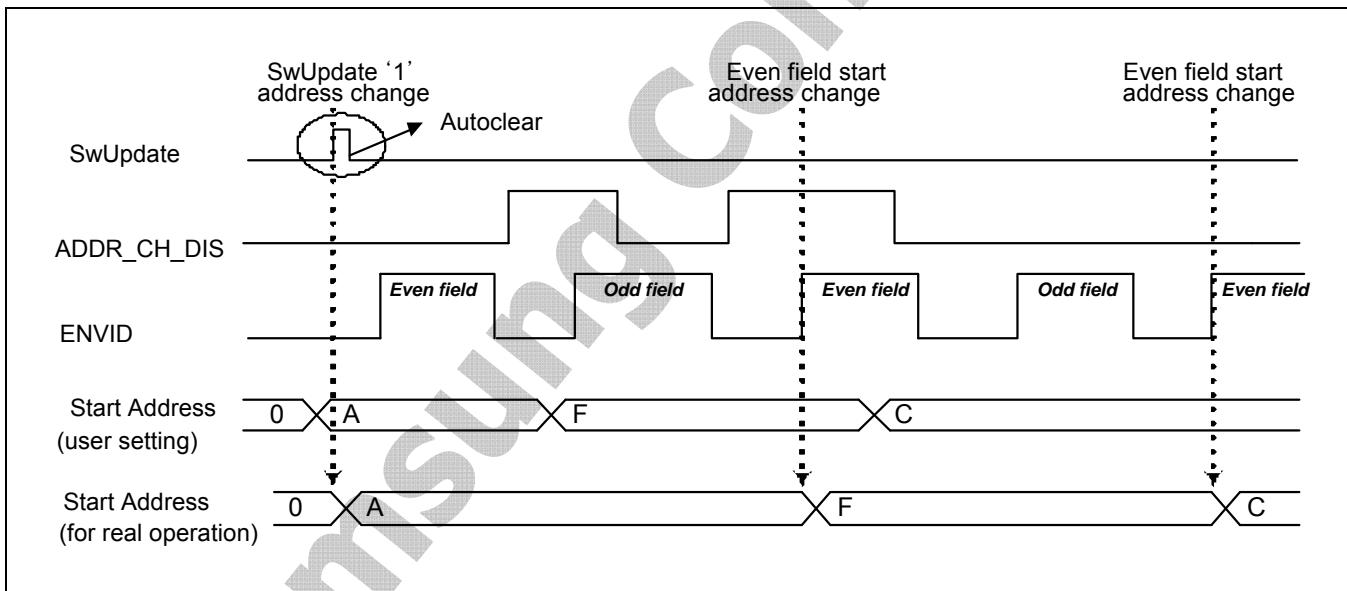


Figure 2-34 Input DMA Address Change Timing (Software Update)

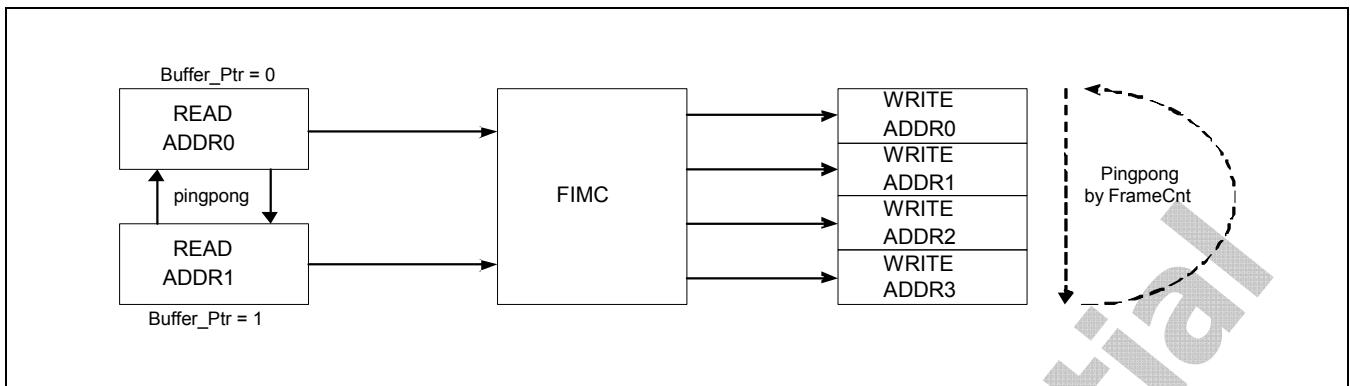


Figure 2-35 Input/Ouput DMA pingpong Address Change Scheme

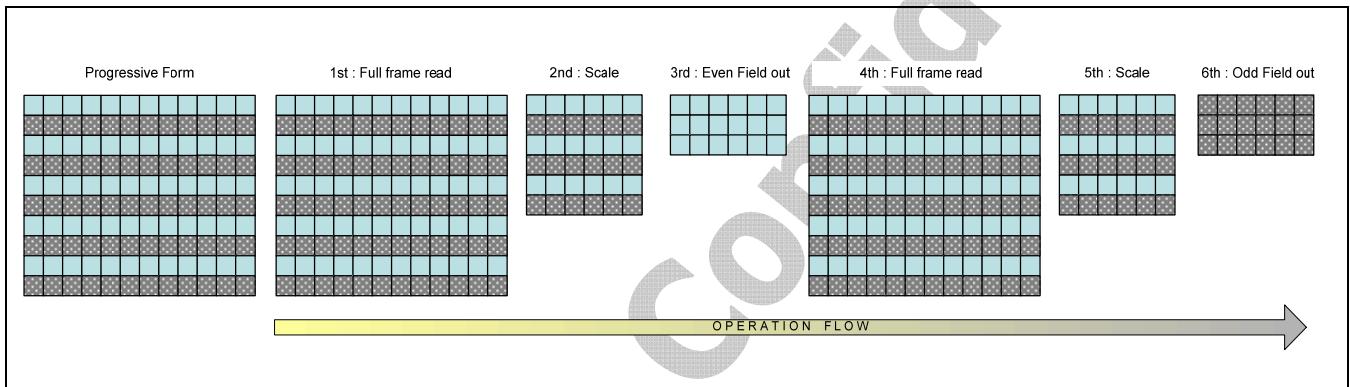


Figure 2-36 Input DMA Progressive-in to Interlace-out (only interlace_out setting)

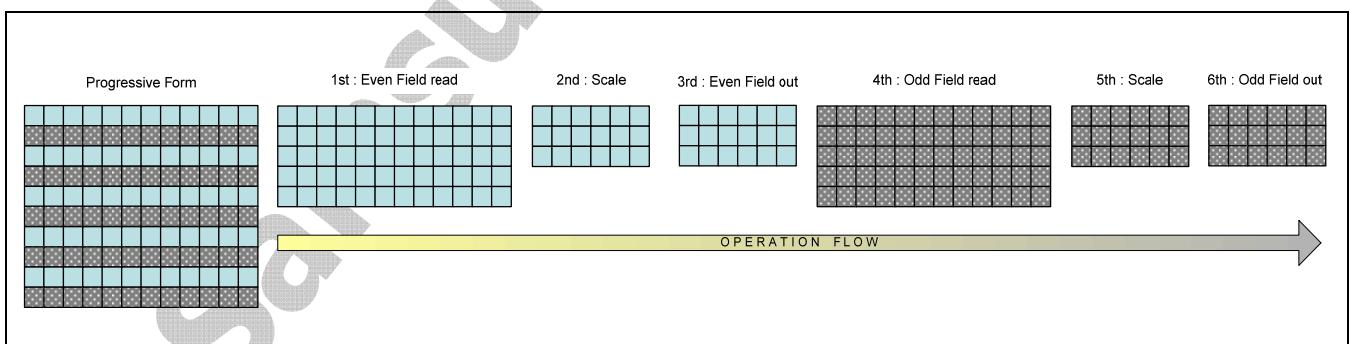


Figure 2-37 Input DMA Progressive-in to Interlace-out (Weave_in and Interlace_out setting)

2.8.2.19 Input DMA Y1 Start Register (CIIYSA1n)

- CIIYSA10, R/W, Address = 0xFB20_0144
- CIIYSA11, R/W, Address = 0xFB30_0144
- CIIYSA12, R/W, Address = 0xFB40_0144

CIIYSA1n	Bit	Description	Initial State
CIIYSA1	[31:0]	Input format: YCbCr 2/3 plane → Y frame start address Input format: YCbCr 1 plane → YCbCr frame start address Input format: RGB → RGB frame start address Note: In tile mode, this value should be aligned 4Kbytes. That is to say, CIOYSA1[11:0] should be 0x000. (ML=OX)	0

2.8.2.20 Input DMA CB1 Start Register (CIICBSA1n)

- CIICBSA10, R/W, Address = 0xFB20_0148
- CIICBSA11, R/W, Address = 0xFB30_0148
- CIICBSA12, R/W, Address = 0xFB40_0148

CIICBSA1n	Bit	Description	Initial State
CIICBSA1	[31:0]	Input format: YCbCr 3 plane → Cb frame start address Input format: YCbCr 2 plane → CbCr frame start address Note: In tile mode, this value should be aligned 4Kbytes. That is to say, CIOYSA1[11:0] should be 0x000. (ML=OX)	0

2.8.2.21 Input DMA CR1 Start Register (CIICRSA1n)

- CIICRSA10, R/W, Address = 0xFB20_014C
- CIICRSA11, R/W, Address = 0xFB30_014C
- CIICRSA12, R/W, Address = 0xFB40_014C

CIICRSA1n	Bit	Description	Initial State
CIICRSA1	[31:0]	Input format: YCbCr 3 plane → Cr frame start address Note: In tile mode, this value should be aligned 4Kbytes. That is to say, CIOYSA1[11:0] should be 0x000. (ML=OX)	0

2.8.2.22 Output DMA Y Offset Register (CIOYOFFn)

- CIOYOFF0, R/W, Address = 0xFB20_0168
- CIOYOFF1, R/W, Address = 0xFB30_0168
- CIOYOFF2, R/W, Address = 0xFB40_0168

CIOYOFFn	Bit	Description	Initial State
Reserved	[31:30]	Reserved	0
OYOFF_V	[29:16]	Output DMA vertical offset for Y component Output format: YCbCr 2/3 plane → Y height offset Output format: YCbCr 1 plane → YCbCr height offset Output format: RGB → RGB height offset Note) Offset value is based on line unit (ML=OO)	0
Reserved	[15:14]	Reserved	0
OYOFF_H	[13:0]	Output DMA horizontal offset for Y component Output format: YCbCr 2/3 plane → Y width offset Output format: YCbCr 1 plane → YCbCr width offset Output format: RGB → RGB width offset Note) Offset value is based on pixel unit (ML=OO)	0

2.8.2.23 Output DMA Cb Offset Register (CIOCBOFFn)

- CIOCBOFF0, R/W, Address = 0xFB20_016C
- CIOCBOFF1, R/W, Address = 0xFB30_016C
- CIOCBOFF2, R/W, Address = 0xFB40_016C

CIOCBOFFn	Bit	Description	Initial State
Reserved	[31:30]	Reserved	0
OCBOFF_V	[29:16]	Output DMA vertical offset for Cb component Output format: YCbCr 3 plane → Cb height offset Output format: YCbCr 2 plane → CbCr height offset Note) Offset value is based on line unit (ML=OO)	0
Reserved	[15:14]	Reserved	0

OCBOFF_H	[13:0]	Output DMA horizontal offset for Cb component Output format: YCbCr 3 plane → Cb width offset Output format: YCbCr 2 plane → CbCr width offset Note) Offset value is based on pixel unit (ML=OO)	0
----------	--------	--	---

Samsung Confidential

2.8.2.24 Output DMA Cr Offset Register

- CIOCROFF0, R/W, Address = 0xFB20_0170
- CIOCROFF1, R/W, Address = 0xFB30_0170
- CIOCROFF2, R/W, Address = 0xFB40_0170

CIOCROFFn	Bit	Description	Initial State
Reserved	[31:30]	Reserved	0
OCROFF_V	[29:16]	Specifies the output DMA vertical offset for Cr component. Output format: YCbCr 3 plane → Cr height offset Note) Offset value is based on line unit (ML=OO)	0
Reserved	[15:14]	Reserved	0
OCROFF_H	[13:0]	Specifies the output DMA horizontal offset for Cr component. Output format: YCbCr 3 plane → Cr width offset Note) Offset value is based on pixel unit (ML=OO)	0

2.8.2.25 Input DMA Y Offset Register (CIIYOFFn)

- CIIYOFF0, R/W, Address = 0xFB20_0174
- CIIYOFF1, R/W, Address = 0xFB30_0174
- CIIYOFF2, R/W, Address = 0xFB40_0174

CIIYOFFn	Bit	Description	Initial State
Reserved	[31:30]	Reserved	0
IYOFF_V	[29:16]	Specifies the input DMA vertical offset for Y component. Input format: YCbCr 2/3 plane → Y height offset Input format: YCbCr 1 plane → YCbCr height offset Input format: RGB → RGB height offset Note) Offset value is based on line unit (ML=OX)	0
Reserved	[15:14]	Reserved	0
IYOFF_H	[13:0]	Specifies the input DMA horizontal offset for Y component. Input format: YCbCr 2/3 plane → Y width offset Input format: YCbCr 1 plane → YCbCr width offset Input format: RGB → RGB width offset Note) Offset value is based on pixel unit (ML=OX)	0

2.8.2.26 Input DMA Cb Offset Register (CIICBOFFn)

- CIICBOFF0, R/W, Address = 0xFB20_0178
- CIICBOFF1, R/W, Address = 0xFB30_0178
- CIICBOFF2, R/W, Address = 0xFB40_0178

CIICBOFFn	Bit	Description	Initial State
Reserved	[31:30]	Reserved	0
ICBOFF_V	[29:16]	Specifies the input DMA vertical offset for Cb component. Input format: YCbCr 3 plane → Cb height offset Input format: YCbCr 2 plane → CbCr height offset Note) Offset value is based on line unit (ML=OX)	0
Reserved	[15:14]	Reserved	0
ICBOFF_H	[13:0]	Specifies the input DMA horizontal offset for Cb component. Input format: YCbCr 3 plane → Cb width offset Input format: YCbCr 2 plane → CbCr width offset Note) Offset value is based on pixel unit (ML=OX)	0

2.8.2.27 Input DMA Cr Offset Register (CIICROFFn)

- CIICRFF0, R/W, Address = 0xFB20_017C
- CIICRFF1, R/W, Address = 0xFB30_017C
- CIICRFF2, R/W, Address = 0xFB40_017C

CIICROFFn	Bit	Description	Initial State
Reserved	[31:30]	Reserved	0
ICROFF_V	[29:16]	Specifies the input DMA vertical offset for Cr component. Input format: YCbCr 3 plane → Cr height offset Note) Offset value is based on line unit (ML=OX)	0
Reserved	[15:14]	Reserved	0
ICROFF_H	[13:0]	Specifies the input DMA horizontal offset for Cr component. Input format: YCbCr 3 plane → Cr width offset Note) Offset value is based on pixel unit (ML=OX)	0

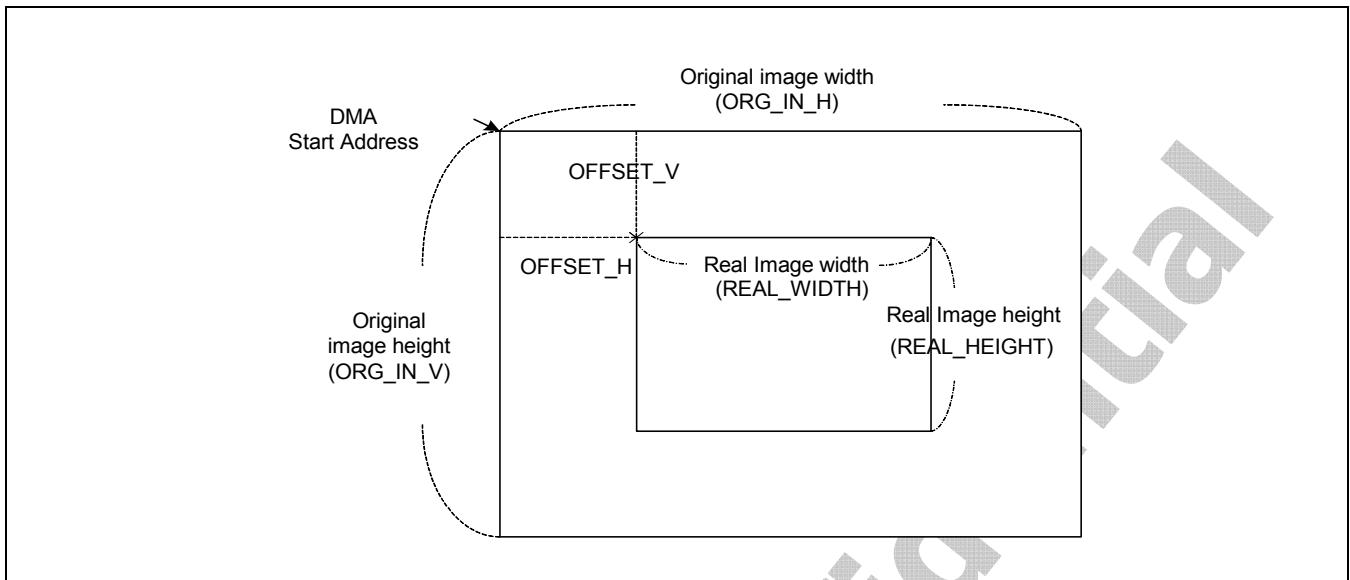


Figure 2-38 Input DMA Offset and Image Size

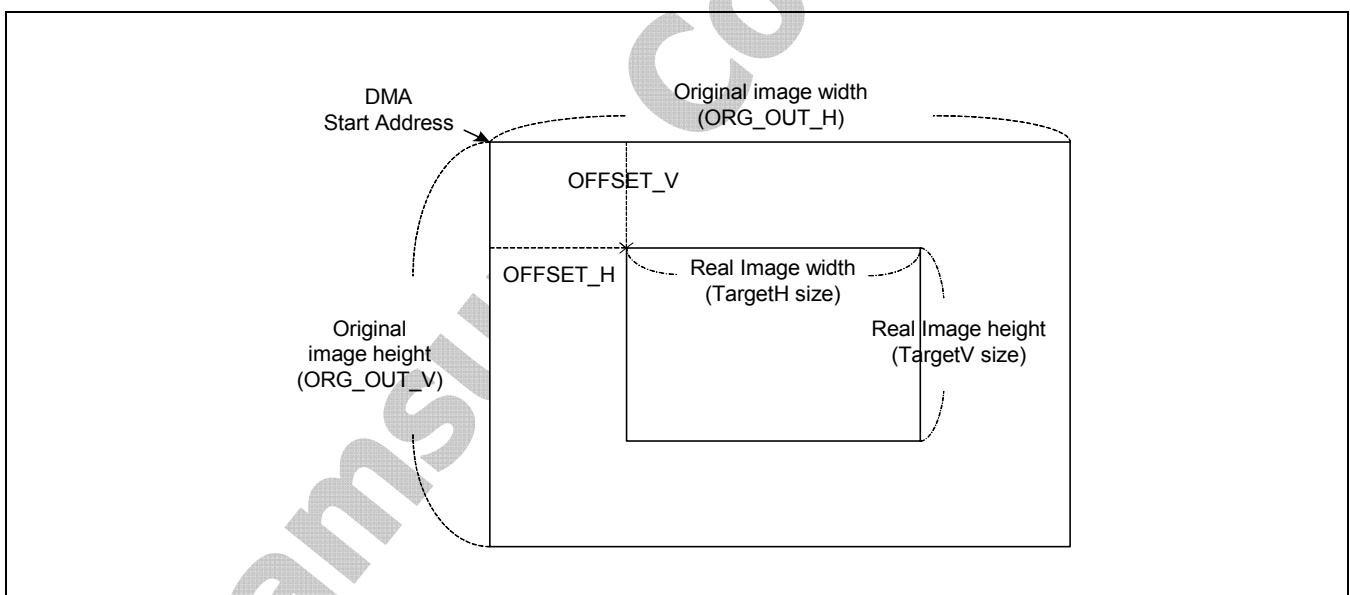


Figure 2-39 Output DMA Offset and Image Size

- **DMA Start Address**

Start address of ADDRStart_Y/Cb/Cr/RGB points to the first address, where the corresponding component of Y/Cb/Cr/RGB is read or written. ADDRStart_Cb is only valid for 2 or 3 planes YCbCr420, 422, 444 source image formats. ADDRStart_Cr is only valid for 3 planes YCbCr420, 422, 444 source image.

Only for CAMIF0 & CAMIF2 : Each of these should be aligned with double word boundary (that is ADDRStart_X[2:0] = 3'b000).

- **DMA OFFSET**

- Offset_H_Y = Y offset per a horizontal line (8's multiple only for CAMIF0,2)
= Number of pixel (or sample) in horizontal offset
- Offset_H_Cb = Cb offset per a horizontal line (8's multiple only for CAMIF0,2)
= Number of pixel (or sample) in horizontal offset
- Offset_H_Cr = Cr offset per a horizontal line (8's multiple only for CAMIF0,2)
= Number of pixel (or sample) in horizontal offset
- Offset_V_Y = Number of vertical Y offset
- Offset_V_Cb = Number of vertical Cb offset
- Offset_V_Cr = Number of vertical Cr offset

2.8.2.28 Original Input DMA Image Size (ORGISIZEn)

- ORGISIZE0, R/W, Address = 0xFB20_0180
- ORGISIZE1, R/W, Address = 0xFB30_0180
- ORGISIZE2, R/W, Address = 0xFB40_0180

ORGISIZEn	Bit	Description	Initial State
Reserved	[31:30]	Reserved	0
ORG_IN_V	[29:16]	Specifies the input DMA original image vertical pixel size. (minimum is 8). This size should not be less than REAL_HEIGHT register. Note) When Input rotator is enabled : this value should be multiple of 8 (ML=0X)	0
Reserved	[15:14]	Reserved	0

ORG_IN_H	[13:0]	Specifies the input DMA source image horizontal pixel size. Must be multiple of 16. This size should not be less than REAL_WIDTH register. (ML=0X)	0
----------	--------	---	---

Note : Memory region of input DMA should follow below equation for input rotator is enabled.

$$\text{ORG_IN_V} + [8 - \{\text{ORG_IN_V} - \text{OFFSET_V}\} \% 8]$$

2.8.2.29 Original Output DMA Image Size (ORGOSIZE_n)

- ORGOSIZE0, R/W, Address = 0xFB20_0184
- ORGOSIZE1, R/W, Address = 0xFB30_0184
- ORGOSIZE2, R/W, Address = 0xFB40_0184

ORGOSIZE _n	Bit	Description	Initial State
Reserved	[31:30]	Reserved	0
ORG_OUT_V	[29:16]	Specifies the output DMA original image vertical pixel size (minimum is 8). This size should not be less than TargetVsize register. If output rotator is running, this size should not be less than TargetHsize register. Note) If output format is YCbCr 420, this value should be even number. (ML=OO)	0
Reserved	[15:14]	Reserved	0
ORG_OUT_H	[13:0]	Specifies the output DMA source image horizontal pixel size. Must be multiple of 16. This size should not be less than TargetHsize register. If output rotator is running, this size should not be less than TargetVsize register. (ML=OO)	0

2.8.2.30 Gathering Extension Register (CIEXTENn)

- CIEXTEN0, R/W, Address = 0xFB20_0188
- CIEXTEN1, R/W, Address = 0xFB30_0188
- CIEXTEN2, R/W, Address = 0xFB40_0188

CIEXTENN	Bit	Description	Initial State
Reserved	[31]	Reserved	0
SrcHsize_CAM_ext	[30]	Specifies the bit value [13] of camera source horizontal pixel number register. {SrcHsize_CAM_ext,SrcHsize_CAM} = {[13], [12:0]}. Thus, total camera source horizontal size = [13:0] (ML=OO)	0
Reserved	[29]	Reserved	0
WinHorOfst_ext	[28]	Specifies the bit value [11] of window horizontal offset register. {WinHorOfst_ext,WinHorOfst} = {[11],[10:0]}. Thus, total window horizontal offset size = [11:0] (ML=OO)	0
Reserved	[27]	Reserved	
TargetHsize_ext	[26]	Specifies the bit value [13] of target image horizontal pixel number register. {TargetHsize_ext,TargetHsize} = {[13],[12:0]} Thus, total target image horizontal size = [13:0] (ML=OO)	0
Reserved	[25]	Reserved	0
TargetVsize_ext	[24]	Specifies the bit value [13] of target image vertical number register. {TargetVsize_ext,TargetVsize} = {[13],[12:0]} Thus, total target image vertical size = [13:0] (ML=OO)	0
Reserved	[23]	Reserved	0
YCbCr444_OUT	[22]	If this bit is 1, output format is YCbCr 444. This register priority is higher than OutFormat in CITRGFMTn (described in Chapter 8.17). (ML=OO)	0
Reserved	[21]	Reserved	0
YCbCr444_IN	[20]	Input DMA format YCbCr4:4:4 (this register priority is higher than InFormat_M register) (ML=OX)	0
Reserved	[19:16]	Reserved	0
MainHorRatio_ext	[15:10]	CAMIF1: Bit value [5:0] of the Mainscale horizontal ratio register. {MainHorRatio,MainHorRatio_ext} = {[14:6],[5:0]} Thus, total Mainscale horizontal ratio register range = [14:0] (ML=OO)	0
Reserved	[9:6]	Reserved	0

MainVerRatio_ext	[5:0]	CAMIF1: Bit value [5:0] of the Mainscale vertical ratio register. {MainVerRatio,MainVerRatio_ext} = {[14:6],[5:0]} Thus, total Mainscale vertical ratio register range = [14:0] (ML=OO)	0
------------------	-------	---	---

Samsung Confidential!

2.8.2.31 DMA Parameter Register (CIDMAPARAMn)

- CIDMAPARAM0, R/W, Address = 0xFB20_018C
- CIDMAPARAM1, R/W, Address = 0xFB30_018C
- CIDAMPARAM2, R/W, Address = 0xFB40_018C

CIDMAPARAMn	Bit	Description	Initial State
Reserved	[31]	Reserved	0
MODE_R	[30:29]	Specifies the INPUT DMA address access style. 0 = Linear 1 = Reserved 2 = Reserved 3 = 64x32 tile (ML=OX)	0
Reserved	[28:15]	Reserved	0
MODE_W	[14:13]	Specifies the OUTPUT DMA address access style. 0 = Linear 1 = Reserved 2 = Reserved 3 = 64x32 tile Note) If input format is either CAM_JPEG or MIPI RAW, User can not use 64x32 tile mode. (ML=XX)	0
Reserved	[12:4]	Reserved	0
Reserved	[3:0]	Reserved	0

NOTE: Refer to Chapter 9.7, "MFC" for Tile Mode description.

2.8.2.32 Mipi Input Format Register (CSIIMGFMTn)

- CSIIMGFMT0, R/W, Address = 0xFB20_0194
- CSIIMGFMT1, R/W, Address = 0xFB30_0194
- CSIIMGFMT2, R/W, Address = 0xFB40_0194

CSIIMGFMTn	Bit	Description	Initial State
Reserved	[31:10]	Reserved	0
DATAB_port	[9:8]	Specifies the MIPI CSIS data align. 0 = 24-bit align 1 = 32-bit align 2 = Reserved 3 = Reserved (ML=XX)	0
Reserved	[7:6]	Reserved	0
ImgFormOfCh0	[5:0]	Specifies the image format of MIPI Channel 0. If the RAW format is image format, image format conversion is not possible. Set scaler bypass mode. 0x1E = YUV422 8-bit 0x2A = RAW8 0x2B = RAW10 0x2C = RAW12 (ML=XX)	0x1E

NOTE: Frame End Address calculation method (useful only for TILE 64x32 access mode)

Note: When tile mode is enable, lower 13bits of Base_address[31:0] have Zero value. So, SFRs related to Base_address should be zero their lower 13bits

When condition is YCbCr4:2:0 3plane & In Rotator 90' & X,XY-flip & Horizontal size \leq 32 & OFFSET_X_Cr \neq 0, Minimum input size is 64x64.

Example: Image pixel size: 720p (1280 x 720), Format: YCbCr4:2:0 2plane (NV12)

```
* hor_img_size (width) = 1280byte, ver_img_size (height) = 720
if (hor_img_size % 16 == 0) hor_img_offset = 0
else hor_img_offset = 16 - (hor_img_size % 16)
if (ver_img_size % 16 == 0) ver_img_offset = 0
elsever_img_offset = 16 - (ver_img_size % 16)
if (Luma) { // Y plane
pixel_x = hor_img_size+hor_img_offset = 1280
pixel_y = ver_img_size+ver_img_offset = 720
}
else if (Chroma) { // Cb/Cr plane
```

```
pixel_x = hor_img_size+hor_img_offset = 1280  
pixel_y = (ver_img_size+ver_img_offset) / 2 = 360  
}
```

1) Luma case

```
pixel_x_minus = pixel_x - 1 = 1279  
pixel_y_minus = pixel_y - 1 = 719 = 1011001111 (binary)  
roundup_x = INT (INT ((pixel_x - 1)/16)/8) + 1 = 10  
roundup_y = INT(INT((pixel_y - 1)/16)/4) + 1 = 12  
  
if (pixel_y_minus[5] == 0) // pixel_y_minus[5:0]='b 001111 , pixel_y_minus[5]=0  
pic_range = pixel_y_minus[14:6] * roundup_x + pixel_x_minus[14:8] + 1 = 11 * 10 + 4 + 1 = 115  
else  
pic_range = roundup_x * roundup_y
```

2) Chroma case

```
pixel_x_minus = pixel_x - 1 = 1279  
pixel_y_minus = pixel_y - 1 = 359 = 101100111 (binary)  
roundup_x = INT (INT ((pixel_x - 1)/16)/8) + 1 = 10  
roundup_y = INT(INT((pixel_y - 1)/16)/4) + 1 = 6  
  
if (pixel_y_minus[5] == 0) // pixel_y_minus[5:0]='b 100111 , pixel_y_minus[5]=1  
pic_range = pixel_y_minus[14:6] * roundup_x + pixel_x_minus[14:8] + 1  
else  
pic_range = roundup_x * roundup_y = 10 * 6 = 60
```

Thus, each plane frame end address = Base_address[31:0] + {pic_range, 2'b0, 11'b0}

2.8.2.33 Miscellaneous Register (CMISCn)

- CMISC0, R/W, Address = 0xFB20_0198
- CMISC1, R/W, Address = 0xFB30_0198
- CMISC2, R/W, Address = 0xFB40_0198

CMISCn	Bit	Description	Initial State
Reserved	[31:2]	-	0
SEL_CORE_CLOCK	[1]	Specifies Clock source selection for CAMIF core block. 0 : Bus clock 1 : Special clock for local path operation between CAMIF and Display Controller. (ML=OO)	0
Reserved	[0]	-	0

2.8.2.34 Key Detect Register (CIKEYn)

- CIKEY0, R/W, Address = 0xFB20_019C
- CIKEY1, R/W, Address = 0xFB30_019C
- CIKEY2, R/W, Address = 0xFB40_019C

CIKEYn	Bit	Description	Initial State
KEY_DETECT	[31]	Specifies KEY detect for graphic layer scaling. 1 = KEY detect ON 0 = OFF (Normal) (ML=OO)	0
Reserved	[30:28]	-	0
R_KEY	[27:20]	Specifies 'R' of the RGB region key value. (ML=OO)	0
Reserved	[19:18]	-	0
G_KEY	[17:10]	Specifies 'G' of the RGB region key value. (ML=OO)	0
Reserved	[9:8]	-	0
B_KEY	[7:0]	Specifies 'B' of the RGB region key value. (ML=OO)	0

3 MIPI DSIM

3.1 ARCHITECTURE OF MIPI DSM

3.1.1 KEY FEATURES OF MIPI DSM

The key features of MIPI DSIM include:

- Complies to MIPI DSI Standard Specification V1.01r11
 - Maximum resolution ranges up to XGA (1028x768)
 - Supports 1, 2, 3, or 4 data lanes
 - Supports pixel format: 16bpp, 18bpp packed, 18bpp loosely packed (3 byte format), and 24bpp
- Interfaces
 - Complies with Protocol-to-PHY Interface (PPI) in MIPI D-PHY Specification V0.90
 - Supports RGB Interface for Video Image Input from display controller
 - Supports I₂O Interface for Command Mode Image input from display controller
 - Supports PMS control interface for PLL to configure byte clock frequency
 - Supports Prescaler to generate escape clock from byte clock

3.1.2 BLOCK DIAGRAM OF MIPI DSI SYSTEM

3.1.2.1 Total System Block Diagram

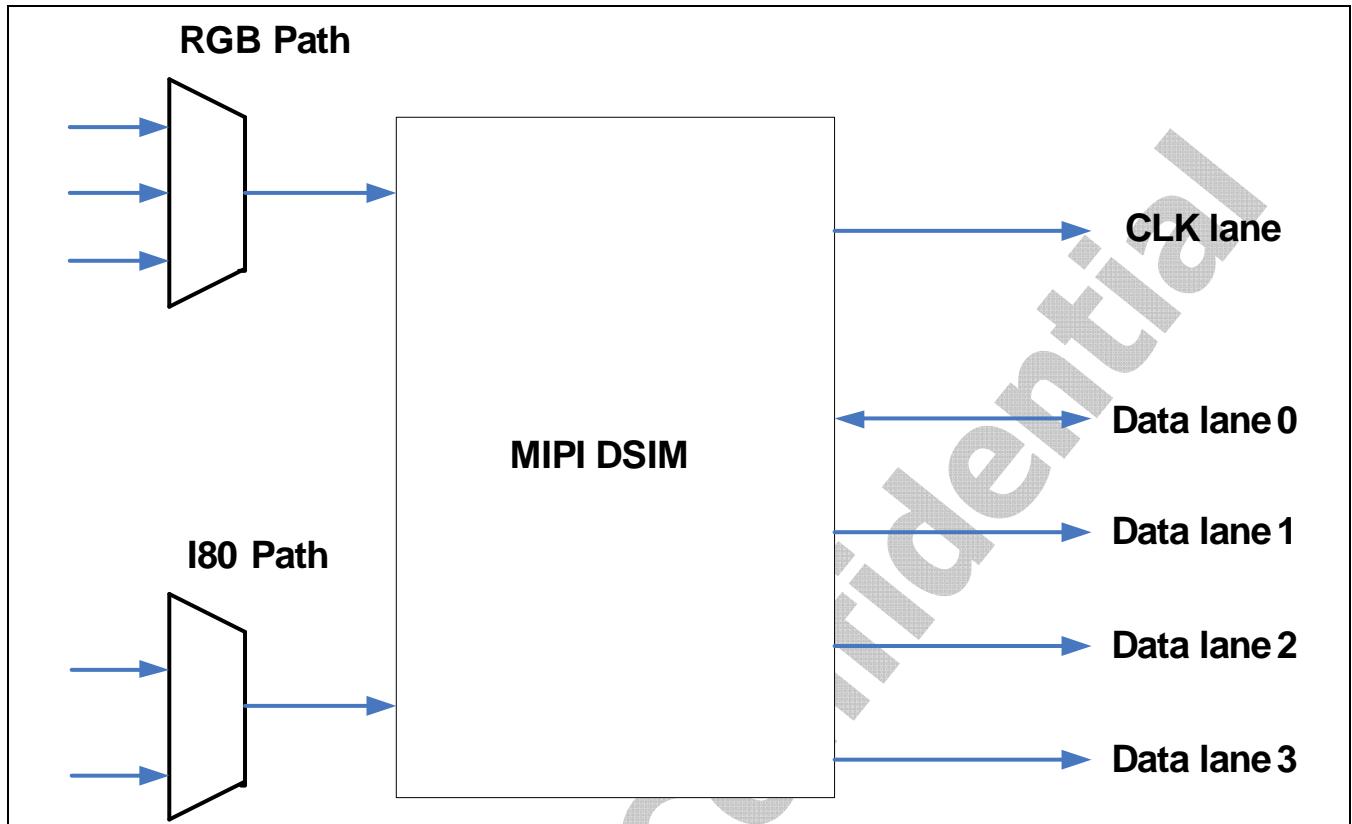


Figure 3-1 MIPI DSI System Block Diagram

NOTE:

1. DSIM gets data from the different IPs.
2. You can select one of above data paths by setting DISPLAY_PATH_SEL [1:0] (0xE010_7008).

For more information, refer to Section, “2-3 S5PV210_CMU”.

3.1.2.2 Internal Primary FIFOs

[Table 3-1](#) describes configurable-sized primary FIFOs.

Table 3-1 Internal Primary FIFO List

Port	FIFO Type	Size	Description
Main display	Packet Header FIFO	3byte X 64 depth	Specifies the packet header FIFO for main display.
	Payload FIFO	4byte X 1024 depth	Specifies the payload FIFO for main display image.
Sub display for I80 INTERFACE image data	Packet Header FIFO	3byte X 4 depth	Specifies the packet header FIFO for I80 INTERFACE sub display.
	Payload FIFO	4byte X 512 depth	Specifies the payload FIFO for I80 INTERFACE sub display image.
Command for I80 INTERFACE command	Packet Header FIFO	3byte X 16 depth	Specifies the packet header FIFO for I80 INTERFACE command packet.
	Payload FIFO	4byte X 16 depth	Specifies the payload FIFO for I80 INTERFACE command long packet payload.
SFR for general packets	Packet Header FIFO	3byte X 16 depth	Specifies the packet header FIFO for general packet.
	Payload FIFO	4byte X 512 depth	Specifies the payload FIFO for general long packet.
RX FIFO	Packet header and Payload FIFO	4byte X 64 depth	Specifies Rx FIFO for LPDR. This FIFO is common for packet header and payload.

3.1.2.3 Packet Header Arbitration

There are four-packet headers FIFOs for Tx, namely, main display, sub display, I80 INTERFACE command, and SFR FIFO. The main and sub display FIFO packet headers contain the image data, while the I80 INTERFACE command FIFO packet header contains the command packets. On the other hand, the SFR FIFO packet header contains command packets, sub display image data (in Video mode), and so on.

The packet header arbiter has a “Fixed priority” algorithm. Priority order is fixed as main display, sub display, I80 INTERFACE command, and SFR FIFO packet header.

In the Video mode, sub display and I80 INTERFACE command FIFO are not used. The SFR FIFO packet header checks if the main display FIFO is empty (no request) in not-active image region and then sends its request.

3.1.2.4 RxFIFO Structure

To read the packets received via low power data receiving mode, RxFIFO acts like an SFR. RxFIFO is an asynchronous FIFO with ByteClk and PCLK domains as input clock and output clock domains respectively. The Rx data is synchronized to RxClk. RXBUF has four Rx Byte buffers for aligning byte to word.

The packet headers of all the packets stored in RXFIFO are word-aligned, that is, the first byte of a packet is always stored in LSB. For example, if a long packet has 7-byte payload, the last byte is filled with dummy byte and the next packet is stored in the next word, as shown in [Figure 3-2](#).

NOTE: CRC data is not stored in RXFIFO.

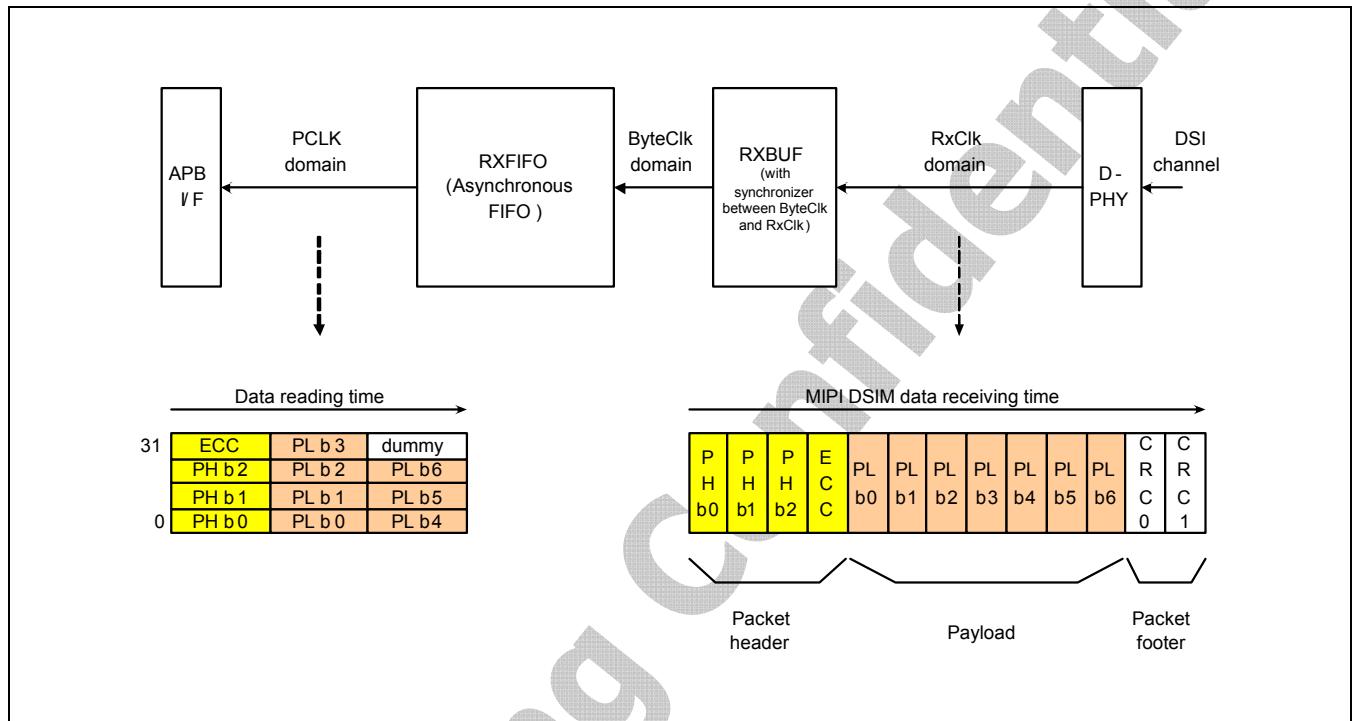


Figure 3-2 Rx Data Word Alignment

3.1.3 INTERFACES AND PROTOCOL

Display Controller-to-DSI Conceptual Signal Converting Diagram in Video Mode

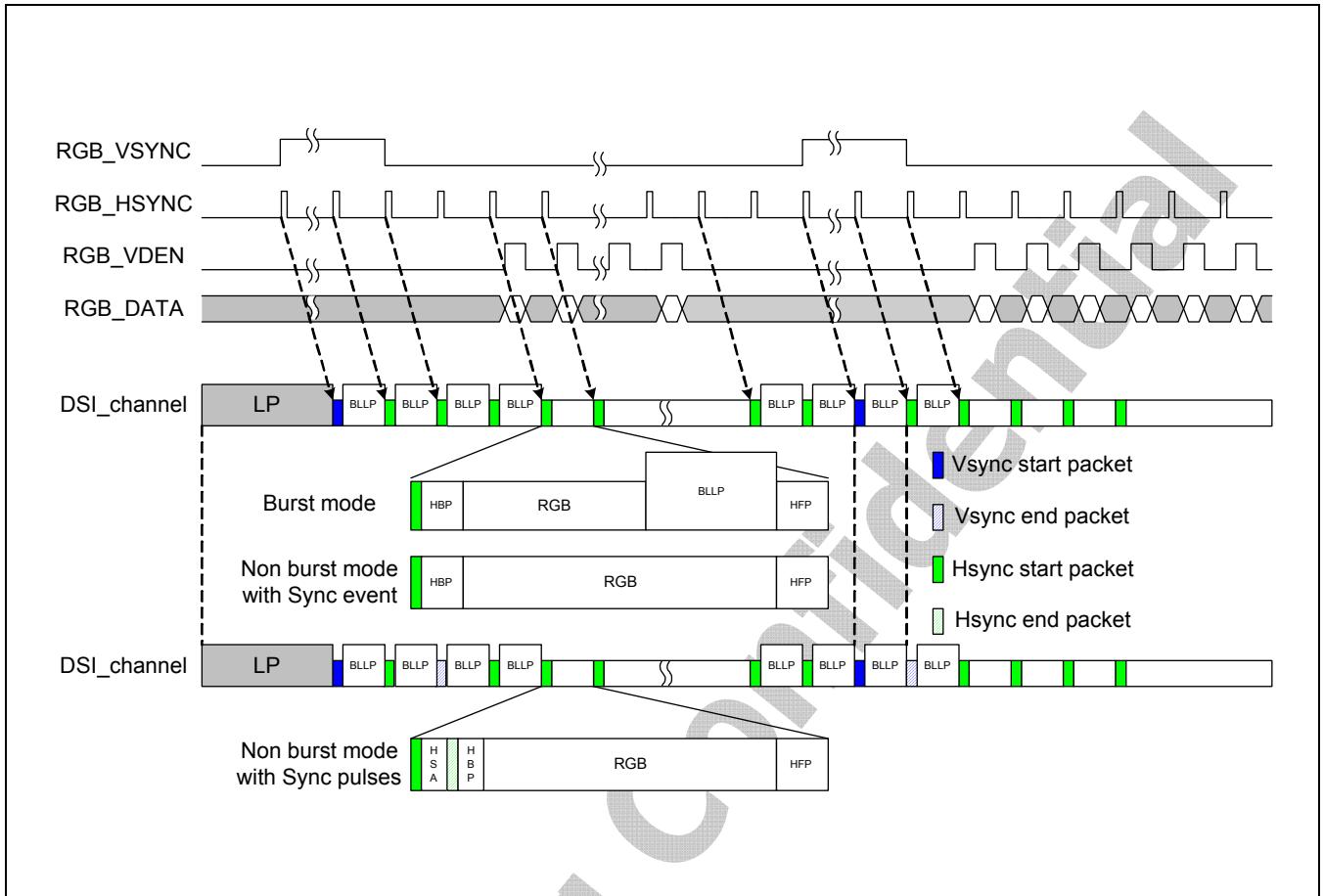


Figure 3-3 Signal Converting Diagram in Video Mode

3.1.3.1 Interface Timing and Protocol

3.1.3.1.1 Display Controller Interface

MIPI DSI Master has two-display controller interfaces, namely, RGB INTERFACE for main display and CPU INTERFACE (I80 INTERFACE) for main/ sub display. The Video mode uses RGB INTERFACE while the Command mode uses CPU INTERFACE.

The RGB image data is loaded on the data bus of RGB INTERFACE and I80 INTERFACE with the same order: RGB_VD[23:0] or SYS_VDOUT[23:0] is {R[7:0],G[7:0],B[7:0]}. Each byte aligns to the most significant bit. For instance, in the 12-bit mode, only three 4-bit values are valid as R, G, and B each, that is, data[23:20], data[15:12], and data[7:4]. The DSIM ignores rest of the bits.

3.1.3.1.2 RGB Interface

Vsync, Hsync, and VDEN are active high signals. Among the three signals, Vsync and Hsync are pulse types that spend several video clocks. RGB_VD[23:0] is {R[7:0],G[7:0],B[7:0]}. All sync signals are synchronized to the rising edge of RGB_VCLK. The display controller sends minimum one horizontal line length of Vsync pulse, V back porch, and V front porch. Hsync pulse width should be longer than 1-byte clock cycle.

3.1.3.1.3 HSA Mode

HSA mode specifies the Horizontal Sync Pulse area disable mode.

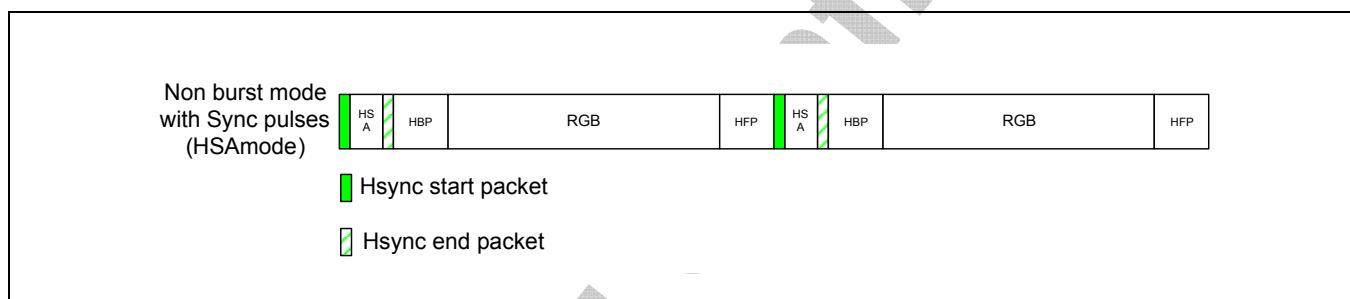


Figure 3-4 Block Timing Diagram of HSA Mode (HSA mode reset: DSIM_CONFIG[20] = 0)

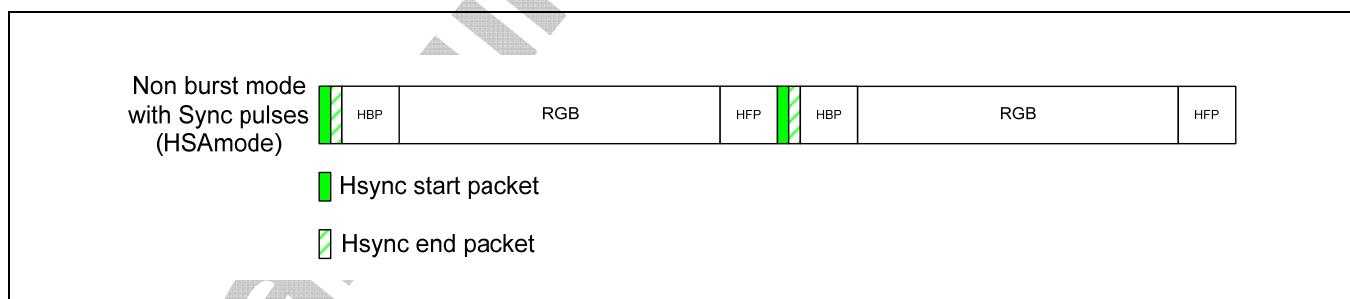


Figure 3-5 Block Timing Diagram of HSA Mode (HSA mode set: DSIM_CONFIG[20] = 1)

HBP mode HBP mode specifies the Horizontal Back Porch disable mode.

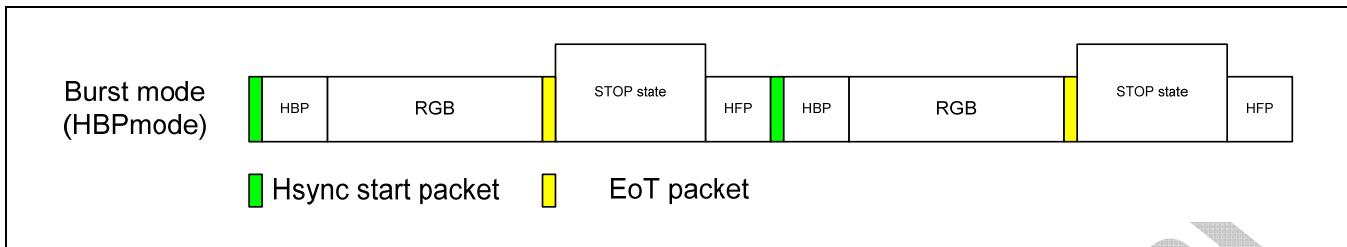


Figure 3-6 Block Timing Diagram of HBP Mode (HBP Mode Reset: DSIM_CONFIG[21] = 0)

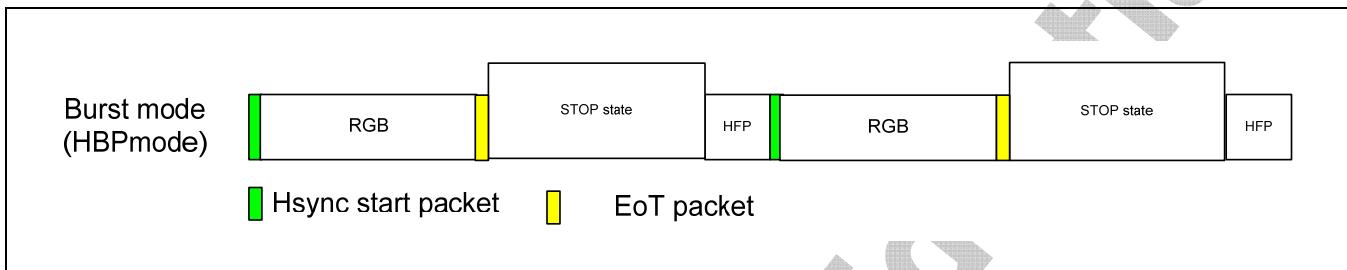


Figure 3-7 Block Timing Diagram of HBP Mode (HBP Mode Set: DSIM_CONFIG[21] = 1)

HFP mode HFP mode specifies the Horizontal Front Porch disable mode.

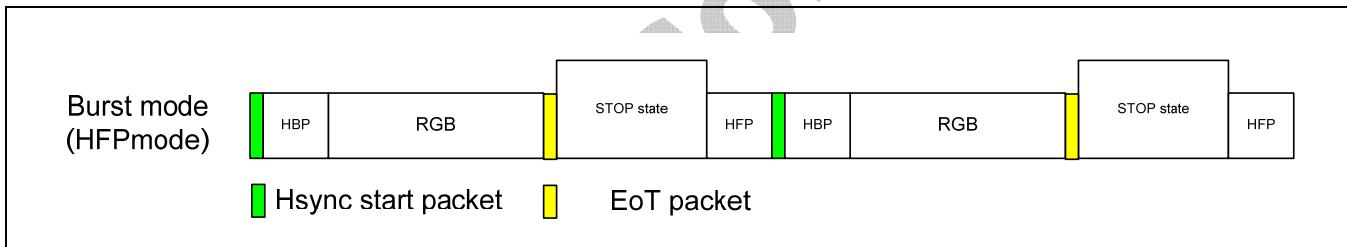


Figure 3-8 Block Timing Diagram of HFP Mode (HFP Mode Reset: DSIM_CONFIG[22] = 0)

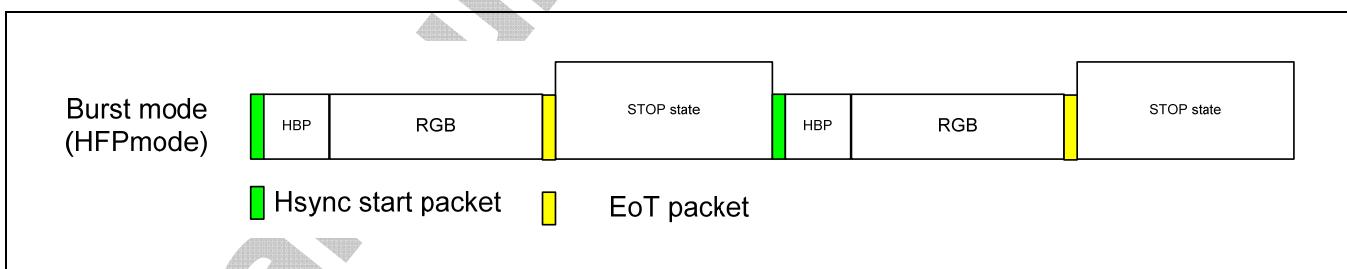


Figure 3-9 Block Timing Diagram of HFP Mode (HFP Mode Set: DSIM_CONFIG[22] = 1)

3.1.3.1.4 HSE Mode

HSE mode specifies the Horizontal Sync End Packet Enable mode in Vsync pulse or Vporch area.

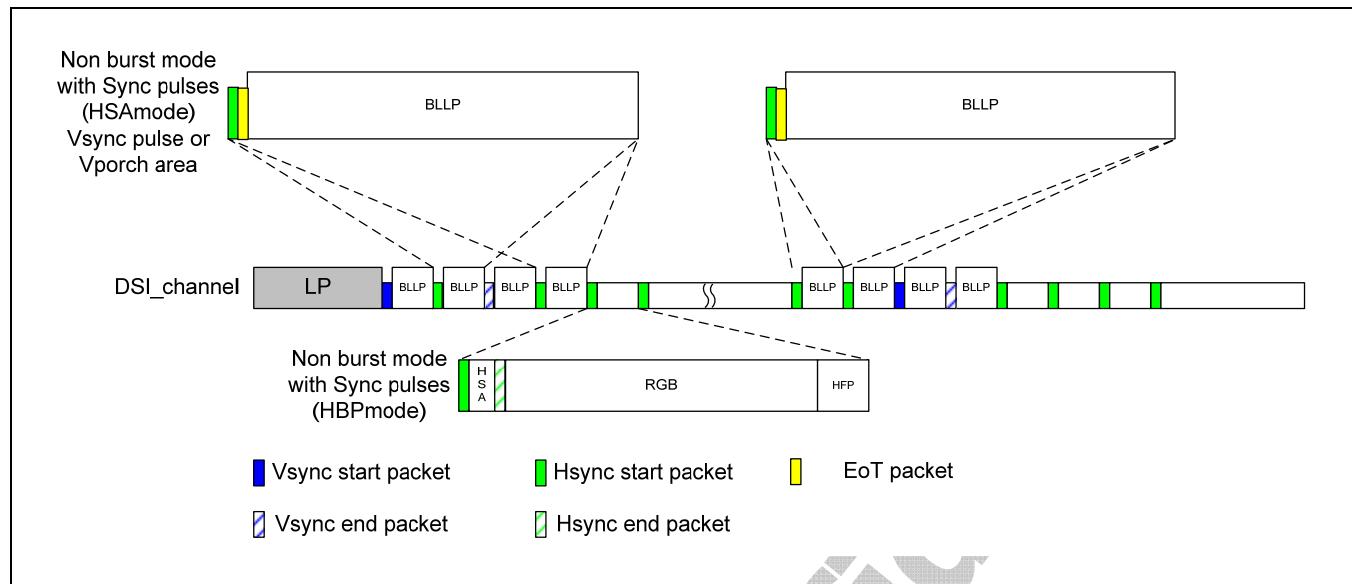


Figure 3-10 Block Timing Diagram of HSE Mode (HSE Mode Reset: DSIM_CONFIG[23] = 0)

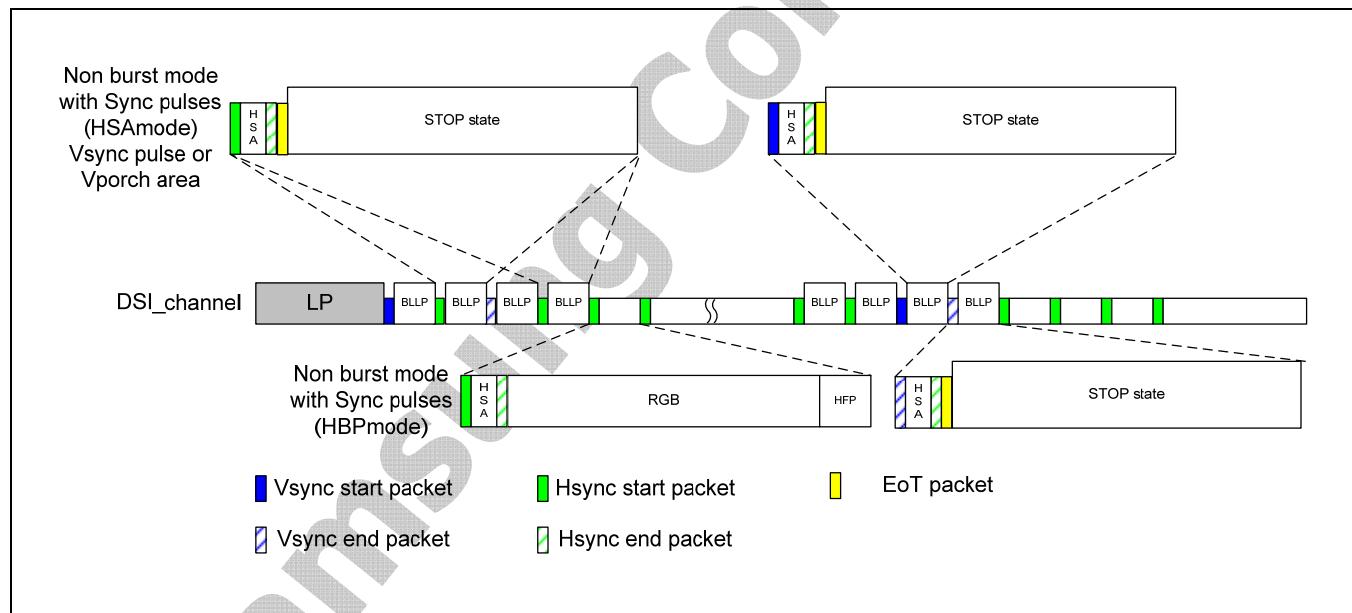


Figure 3-11 Block Timing Diagram of HSE Mode (HSE Mode Set: DSIM_CONFIG[23] = 1)

3.1.3.1.5 Transfer General Data in Video Mode

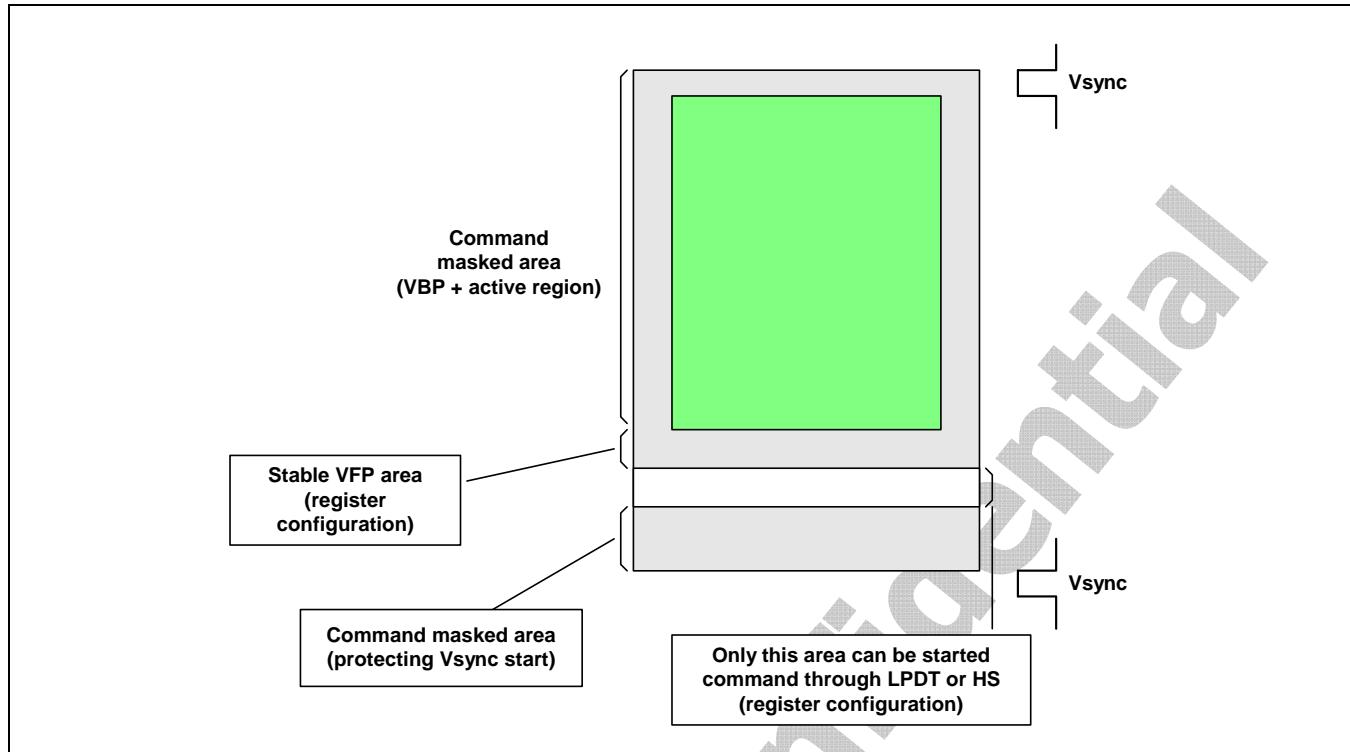


Figure 3-12 Stable VFP Area Before Command Transfer Allowing Area

3.1.3.1.6 MIPI DSIM Converts RGB Interface to Video Mode

Vsync and Hsync packets are extremely important to protect image in Video mode. MIPI DSIM allows several lines in VFP area to transfer general data transfer. As shown in [Figure 3-12](#), the vertical front porch is divided into three areas, namely, stable VFP area, command allowed area, and command masked area.

The register configures stable VFP area. Configuration boundary is 11'h000 ~ 11'h7FFF in DSIM_MVPORCH.

The register also configures the command allowed area. Configuration boundary is 4'h0 ~ 4'hF in DSIM_MVPORCH. Only this area is allowed to start "command transfer" through HS mode or LPDT. In LPDT, data transferring takes a long time to complete (approximately hundreds of microseconds or more). In this time, Hsync packet does not arrive due to LPDT long packet. MIPI DSIM comprises of big size FIFO for lost Hsync packet. After LPDT, MIPI DSIM transfers these Hsync packets immediately through HS mode.

To protect Vsync, command masked area is masked area. This area is calculated using LPDT bandwidth. For example, if EscClk is 10MHz, the maximum long packet payload size is 1KB and LPDT, LPDT transferring time is 824us (packet size: 1030byte, LPDT maximum bandwidth: 10Mbps). If one line time is 20us, the line timing violation occurs in 42 lines. Therefore, command masked area is larger than $42 + \alpha$. This ' α ' is transferring time of the violated Hsync packets.

Display controller should be configured in such a way that VFP lines are sum of stable vfp, command allowed area, and command masked area.

3.1.3.1.7 I80 Interface

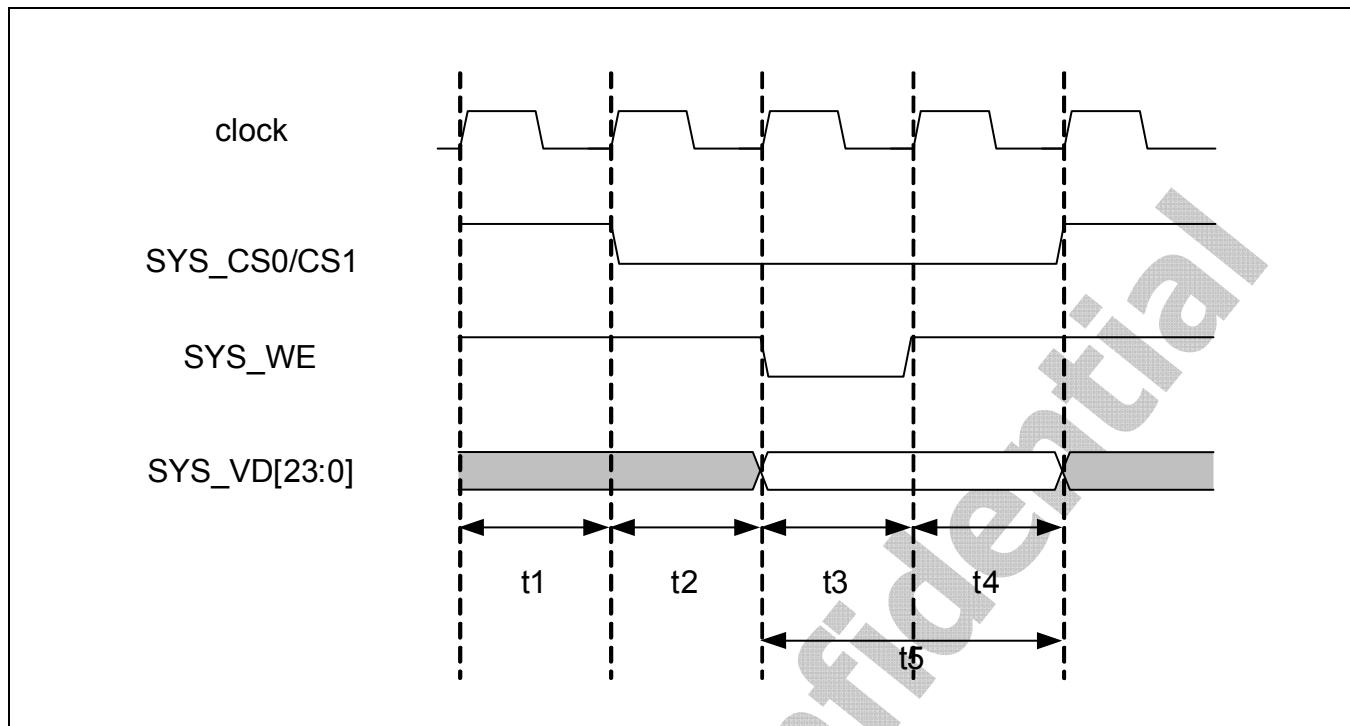


Figure 3-13 I80 Interface Timing Diagram

- $T_1 \geq 1$ clock cycle.
- $T_2 \geq 0$ clock cycle.
- $T_3 \geq 1$ clock cycle.
- $T_4 \geq 0$ clock cycle.
- $T_5 \geq 1$ clock cycle.
- $T_2+T_3+T_4 > 1$ cycle of byte clock

A display controller generates these signals with its internal clock: **SYS_CS0/CS1**, **SYS_WE**, and **SYS_VD**. MIPI DSIM master decodes the **SYS_ADDR**. [Table 3-2](#) describes the I80 INTERFACE address map.

Table 3-2 I80 Interface Address Map

SYS_ADDR[1:0]	Description
2'b00	Specifies the image data.
2'b01	Reserved
2'b10	Specifies the payload data.
2'b11	Specifies the packet header.

[Figure 3-14](#) shows how MIPI DSI Master makes packet from the image data stream via I80 INTERFACE in Command mode. MIPI DSI master makes packet from the first line with DCS command “write_memory_start” and the other lines with DCS command “write_memory_continue”.

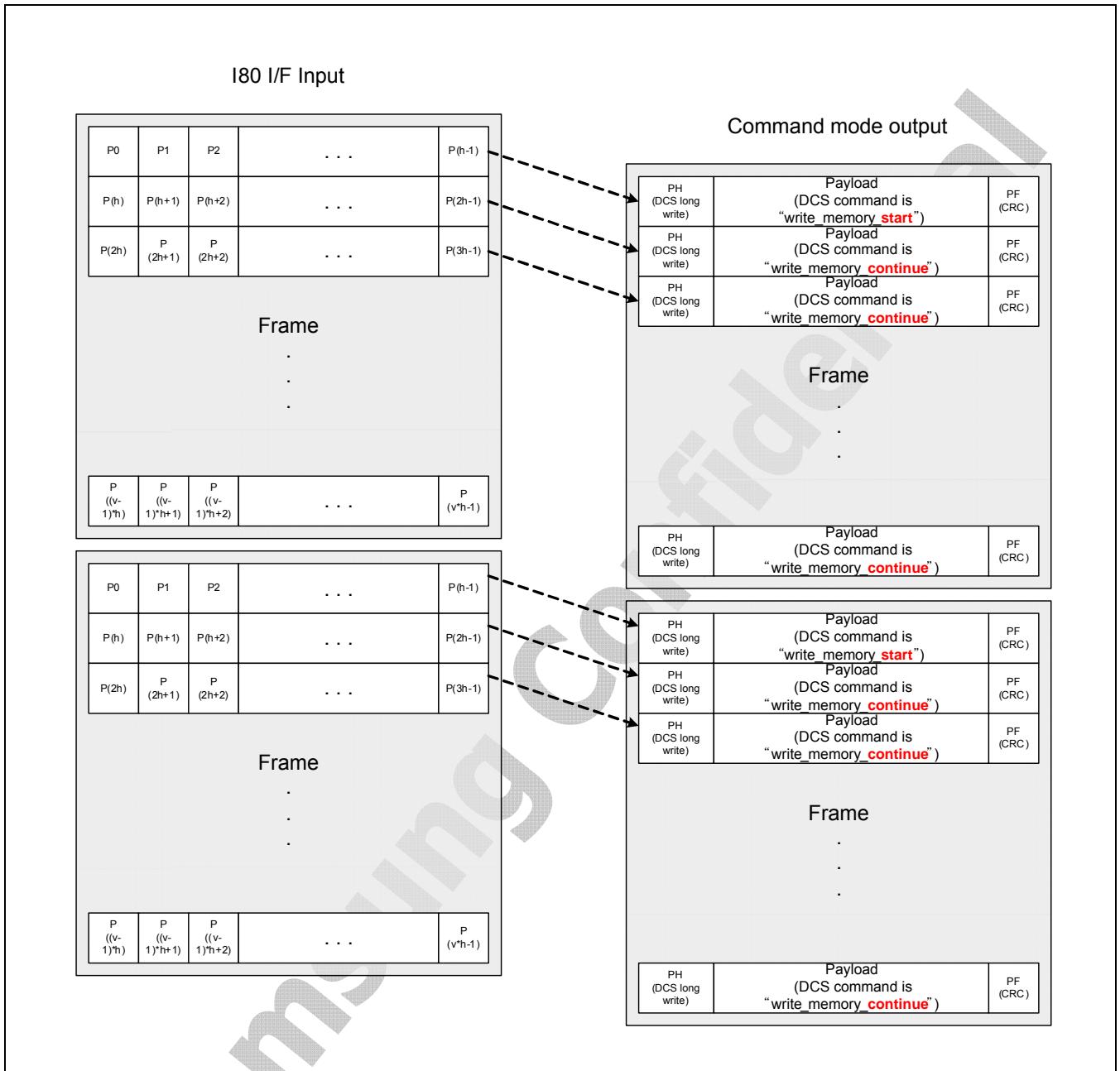


Figure 3-14 Packetizing for MIPI DSI Command Mode from I80 Interface

3.1.3.1.8 Relation Between Input Transactions and DSI Transactions**Table 3-3 Relation Between Input Transactions and DSI Transactions**

Input Interface	Input transaction	DSI Transaction
RGB	RGB transaction	Specifies the RGB Packet. 888, 666, 666 (loosely packed), and 565 should be specified via register configuration.
I80	I80 Image Transaction	Specifies the Data type, that is, “DCS Long Write packet”. (DCS command is “memory write start/continue”.)
I80	I80 Command Transaction	Specifies any DSI packet. Bytes in I80 transaction should be the same bytes in DSI packets.
SFR	Header and Payload FIFO access	Specifies any DSI Packets. Bytes in APB transaction should be the same bytes in DSI packets.

3.1.4 CONFIGURATION

3.1.4.1 Video Mode Versus Command Mode

MIPI DSI Master Block supports two modes, namely, Video mode and Command mode.

3.1.5 DUAL DISPLAY VERSUS SINGLE DISPLAY

3.1.5.1 Dual Display

MIPI DSI Master supports dual display configuration in Command mode only, that is, both main and sub display image should be transmitted via i80 interface.

3.1.5.2 Single Display

For single display configuration, use video mode or command mode.

3.1.6 PLL

To transmit Image data, MIPI DSI Master Block needs high frequency clock (80MHz ~ 1GHz) generated by PLL.

To configure PLL, MIPI DSI Master comprises of SFRs and corresponding interface signals. PLL is embedded in PHY module. You should use other PLL in SoC if it meets the timing specification.

3.1.7 BUFFER

In MIPI DSI standard specification, DSI Master sends image stream in burst mode. The image stream transmits in high-speed and bit-clock frequency. This mode allows the device to stay in stop state longer to reduce power consumption. For this mode, MIPI DSI Master has a dual line buffer to store one complete line and send it faster at the next line time.

3.2 I/O DESCRIPTION

Table 3-4 MIPI-DPHY Interface Slave Signal

Signal	I/O	Description	Pad	Type
MIPI_DP_0	B	Specifies the DP signal for MIPI-DPHY Master data-lane 0.	XmipiDP[0]	Dedicated
MIPI_DN_0	B	Specifies the DN signal for MIPI-DPHY Master data-lane 0.	XmipiDN[0]	Dedicated
MIPI_DP_1	O	Specifies the DP signal for MIPI-DPHY Master data-lane 1.	XmipiDP[1]	Dedicated
MIPI_DN_1	O	Specifies the DN signal for MIPI-DPHY Master data-lane 1.	XmipiDN[1]	Dedicated
MIPI_DP_2	O	Specifies the DP signal for MIPI-DPHY Master data-lane 2.	XmipiDP[2]	Dedicated
MIPI_DN_2	O	Specifies the DN signal for MIPI-DPHY Master data-lane 2.	XmipiDN[2]	Dedicated
MIPI_DP_3	O	Specifies the DP signal for MIPI-DPHY Master data-lane 3.	XmipiDP[3]	Dedicated
MIPI_DN_3	O	Specifies the DN signal for MIPI-DPHY Master data-lane 3.	XmipiDN[3]	Dedicated
MIPI_CLK_TX_P	O	Specifies the DP signal for MIPI-DPHY Master clock-lane.	XmipiCLK_TX_P	Dedicated
MIPI_CLK_TX_N	O	Specifies the DN signal for MIPI-DPHY Master clock-lane.	XmipiCLK_TX_N	Dedicated

NOTE:

1. I/O direction. I: input, O: output, and B: bi-direction.
2. Type field indicates whether pads are dedicated to the signal or pads are connected to the multiplexed signals.

3.3 REGISTER DESCRIPTION

3.3.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
DSIM_STATUS	0xFA50_0000	R	Specifies the status register.	0x0010_010F
DSIM_SWRST	0xFA50_0004	R/W	Specifies the software reset register.	0x0000_0000
DSIM_CLKCTRL	0xFA50_0008	R/W	Specifies the clock control register.	0x0000_FFFF
DSIM_TIMEOUT	0xFA50_000C	R/W	Specifies the time out register.	0x00FF_FFFF
DSIM_CONFIG	0xFA50_0010	R/W	Specifies the configuration register.	0x0200_0000
DSIM_ESCMODE	0xFA50_0014	R/W	Specifies the escape mode register.	0x0000_0000
DSIM_MDRESOL	0xFA50_0018	R/W	Specifies the main display image resolution register.	0x0300_0400
DSIM_MVPORCH	0xFA50_001C	R/W	Specifies the main display Vporch register.	0xF000_0000
DSIM_MHPORCH	0xFA50_0020	R/W	Specifies the main display Hporch register.	0x0000_0000
DSIM_MSYNC	0xFA50_0024	R/W	Specifies the main display Sync Area register.	0x0000_0000
DSIM_SDRESOL	0xFA50_0028	R/W	Specifies the sub display image resolution register.	0x0300_0400
DSIM_INTSRC	0xFA50_002C	R/W	Specifies the interrupt source register.	0x0000_0000
DSIM_INTMSK	0xFA50_0030	R/W	Specifies the interrupt mask register.	0xB337_FFFF
DSIM_PKTHDR	0xFA50_0034	W	Specifies the packet header FIFO register.	0x0000_0000
DSIM_PAYLOAD	0xFA50_0038	W	Specifies the payload FIFO register.	0x0000_0000
DSIM_RXFIFO	0xFA50_003C	R	Specifies the read FIFO register.	0xFFFF_FFFF
DSIM_FIFOTHLD	0xFA50_0040	R/W	Specifies the FIFO threshold level register.	0x0000_01FF
DSIM_FIFOCTRL	0xFA50_0044	R	Specifies the FIFO status and control register.	0x0155_551F
DSIM_MEMACCHR	0xFA50_0048	R/W	Specifies the FIFO memory AC characteristic register.	0x0000_4040
DSIM_PLLCTRL	0xFA50_004C	R/W	Specifies the PLL control register.	0x0000_0000
DSIM_PLLTMR	0xFA50_0050	R/W	Specifies the PLL timer register.	0xFFFF_FFFF
DSIM_PHYACCHR	0xFA50_0054	R/W	Specifies the D-PHY AC characteristic register.	0x0000_0000
DSIM_PHYACCHR1	0xFA50_0058	R/W	Specifies the D-PHY AC characteristic register 1.	0x0000_0000

NOTE: M_RESETN at MIPI_PHY_CON0 (0xE010_7200) should be '1' before enabling DSIM.

3.3.1.1 Status Register (DSIM_STATUS, R, Address = 0xFA50_0000)

This register reads and checks internal and interface status. It also checks FSM status, Line buffer status, current image line number, and so on.

DSIM_STATUS	Bit	Description	Initial State
PLLStable	[31]	D-phy pll generates stable byteclk.	0
Reserved	[30:21]	Reserved	0
SwRstRls	[20]	Specifies the software reset status. 0 = Reset state 1 = Release state	0
Reserved	[19:17]	Reserved	0
Direction	[16]	Specifies the data direction indicator. 0 = Forward direction 1 = Backward direction	1
Reserved	[15:11]	Reserved	0
TxReadyHsClk	[10]	Specifies the HS clock ready at clock lane. 0 = Not ready for transmitting HS data at clock lane 1 = Ready for transmitting HS data at clock lane	0
UlpsClk	[9]	Specifies the ULPS indicator at clock lane. 0 = No ULPS in clock lane 1 = ULSP in clock lane	1
StopstateClk	[8]	Specifies the stop state indicator at clock lane. 0 = No stop state in clock lane 1 = Stop state in clock lane	0
UlpsDat[3:0]	[7:4]	Specifies the ULPS indicator at data lanes. UlpsDat[0]: Data lane 0 UlpsDat[1]: Data lane 1 UlpsDat[2]: Data lane 2 UlpsDat[3]: Data lane 3 0 = No ULPS in each data lane 1 = ULPS in each data lane	F
StopstateDat[3:0]	[3:0]	Specifies the stop state indicator at data lane. StopstateDat[0]: Data lane 0 StopstateDat[1]: Data lane 1 StopstateDat[2]: Data lane 2 StopstateDat[3]: Data lane 3 0 = No stop state in each data lane 1 = Stop state in each data lane	0

3.3.1.2 Software Reset Register (DSIM_SWRST, R/W, Address = 0xFA50_0004)

DSIM_SWRST	Bit	Description	Initial State
Reserved	[31:17]	Reserved	-
FuncRst	[16]	<p>Specifies the software reset (High active). “Software reset” resets all FF in MIPI DSIM (except SFRs: STATUS, SWRST, CLKCTRL, TIMEOUT, CONFIG, ESCMODE*, MDRESOL, MDVPORCH, MHPORCH, MSYNC, INTMSK, SDRESOL, FIFOTHLD, FIFOCTRL**, MEMACCHR, PLLCTRL, PLLTMR, PHYACCHR, and VERINFORM).</p> <p>0 = Standby 1 = Reset</p> <p>*: ForceStopstate, CmdLpd, TxLpd, **: nInitRx, nInitSfr, nInitI80, nInitSub, nInitMD</p>	0
Reserved	[15:1]	Reserved	-
SwRst	[0]	<p>Specifies the software reset (High active). “Software reset” resets all FF in MIPI DSIM (except some SFRs: STATUS, SWRST, CLKCTRL, PLLCTRL, PLLTMR, and PHYTUNE).</p> <p>0 = Standby 1 = Reset</p>	0

3.3.1.3 Clock Control Register (DSIM_CLKCTRL, R/W, Address = 0xFA50_0008)

DSIM_CLKCTRL	Bit	Description	Initial State
TxRequestHsClk	[31]	Specifies the HS clock request for HS transfer at clock lane (Turn on HS clock)	0
Reserved	[30:29]	Reserved	-
EscClkEn	[28]	Enables the escape clock generating prescaler. 0 = Disables 1 = Enables	0
PLLbypass	[27]	Sets the PLLBypass signal connected to D-PHY module input for selecting clock source bit. For more information, refer to MIPI D-PHY specification. 0 = PLL output 1 = External Serial clock	0
ByteClkSrc	[26:25]	Selects byte clock source. (It must be 00.) 00 = D-PHY PLL (default). PLL_out clock is used to generate ByteClk by dividing 8.	0
ByteClkEn	[24]	Enables byte clock. 1 = Disables 0 = Enables	0
LaneEscClkEn	[23:19]	Enables escape clock for D-phy lane. LaneEscClkEn[0] = Clock lane LaneEscClkEn[1] = Data lane 0 LaneEscClkEn[2] = Data lane 1 LaneEscClkEn[3] = Data lane 2 LaneEscClkEn[4] = Data lane 3 0 = Disables 1 = Enables	0
Reserved	[18:16]	Reserved	-
EscPrescaler	[15:0]	Specifies the escape clock prescaler value. The escape clock frequency range varies up to 20MHz. Note: The requirement for BTA is that the Host Escclk frequency should range between 66.7 ~ 150% of the peripheral escape clock frequency. EscClk = ByteClk / (EscPrescaler)	0xFFFF

3.3.1.4 Time Out register (DSIM_TIMEOUT, R/W, Address = 0xFA50_000C)

DSIM_TIMEOUT	Bit	Description	Initial State
Reserved	[31:24]	Reserved	-
BtaTout	[23:16]	Specifies the timer for BTA. This register specifies time out from BTA request to change the direction with respect to Tx escape clock.	0xFF
LpdrTout	[15:0]	Specifies the timer for LP Rx mode timeout. This register specifies time out on how long RxValid deasserts, after RxLpd asserts with respect to Tx escape clock. RxValid specifies Rx data valid indicator. RxLpd specifies an indicator that D-phy is under RxLpd mode. RxValid and RxLpd specifies signal from D-phy.	0xFFFF

3.3.1.5 Configuration Register (DSIM_CONFIG, R/W, 0xFA50_0010)

This register configures MIPI DSI master such as data lane number, input interface, porch area, frame rate, BTA, LPDT, ULPS, and so on.

DSIM_CONFIG	Bit	Description	Initial State
Reserved	[31:30]	Reserved	-
Mflush_VS	[29]	Auto flush of MD FIFO using Vsync pulse. It needs that Main display FIFO should be flushed for deleting garbage data. 0 = Enable (defalut) 1 = Disable	1
EoT_r03	[28]	Disables EoT packet in HS mode. 0 = Enables EoT packet generation for V1.01r11 1 = Disables EoT packet generation for V1.01r03	0
Synclinform	[27]	Selects Sync Pulse or Event mode in Video mode. 0 = Event mode (non burst, burst) 1 = Pulse mode (non burst only) In command mode, this bit is ignored.	0
BurstMode	[26]	Selects Burst mode in Video mode In Non-burst mode, RGB data area is filled with RGB data and Null packets, according to input bandwidth of RGB interface. In Burst mode, RGB data area is filled with RGB data only. 0 = Non-burst mode 1 = Burst mode In command mode, this bit is ignored.	0
VideoMode	[25]	Specifies display configuration. 0 = Command mode 1 = Video mode	1

DSIM_CONFIG	Bit	Description	Initial State
AutoMode	[24]	<p>Specifies auto vertical count mode.</p> <p>In Video mode, the vertical line transition uses line counter configured by VSA, VBP, and Vertical resolution. If this bit is set to '1', the line counter does not use VSA and VBP registers.</p> <p>0 = Configuration mode 1 = Auto mode</p> <p>In command mode, this bit is ignored.</p>	0
HseMode	[23]	<p>In Vsync pulse and Vporch area, MIPI DSI master transfers only Hsync start packet to MIPI DSI slave at MIPI DSI spec 1.1r02. This bit transfers Hsync end packet in Vsync pulse and Vporch area (optional).</p> <p>0 = Disables transfer 1 = Enables transfer</p> <p>In command mode, this bit is ignored.</p>	0
HfpMode	[22]	<p>Specifies HFP disable mode. If this bit set, DSI master ignores HFP area in Video mode.</p> <p>0 = Enables 1 = Disables</p> <p>In command mode, this bit is ignored.</p>	0
HbpMode	[21]	<p>Specifies HBP disable mode. If this bit set, DSI master ignores HBP area in Video mode.</p> <p>0 = Enables 1 = Disables</p> <p>In command mode, this bit is ignored.</p>	0
HsaMode	[20]	<p>Specifies HSA disable mode. If this bit set, DSI master ignores HSA area in Video mode.</p> <p>0 = Enables 1 = Disables</p> <p>In command mode, this bit is ignored.</p>	0
MainVc	[19:18]	Specifies virtual channel number for main display.	0
SubVc	[17:16]	Specifies virtual channel number for sub display.	0
Reserved	[15]	Reserved	-
MainPixelFormat	[14:12]	<p>Specifies pixel stream format for main display.</p> <p>000 = 3bpp (for Command mode only) 001 = 8bpp (for Command mode only) 010 = 12bpp (for Command mode only) 011 = 16bpp (for Command mode only) 100 = 16-bit RGB (565) (for Video mode only) 101 = 18-bit RGB (666: packed pixel stream) (for Video mode only) 110 = 18-bit RGB (666: loosely packed pixel stream) for Common 111 = 24-bit RGB (888) for common</p>	0

DSIM_CONFIG	Bit	Description	Initial State
Reserved	[11]	Reserved	-
SubPixelFormat	[10:8]	<p>Specifies pixel stream format for sub display.</p> <p>000 = 3bpp (for Command mode only) 001 = 8bpp (for Command mode only) 010 = 12bpp (for Command mode only) 011 = 16bpp (for Command mode only) 100 = 16-bit RGB (565) (for Video mode only) 101 = 18-bit RGB (666: packed pixel stream) for Video mode only 110 = 18-bit RGB (666: loosely packed pixel stream) for common 111 = 24-bit RGB (888) (for Common)</p>	0
Reserved	[7]	Reserved	-
NumOfDatLane	[6:5]	<p>Sets the data lane number.</p> <p>00 = Data lane 0 (1 data lane) 01 = Data lane 0 ~ 1 (2 data lanes) 10 = Data lane 0 ~ 2 (3 data lanes) 11 = Data lane 0 ~ 3 (4 data lanes)</p>	0
LaneEn[4:0]	[4:0]	<p>Enables the lane. If Lane_EN is disabled, the lane ignores input and drives initial value through output port.</p> <p>0 = Lane is off. 1 = Lane is on.</p> <p>LaneEn[0] = Clock lane enabler LaneEn[1] = Data lane 0 enabler LaneEn[2] = Data lane 1 enabler LaneEn[3] = Data lane 2 enabler LaneEn[4] = Data lane 3 enabler</p>	0

3.3.1.6 Escape Mode Register (DSIM_ESCMODE, R/W, Address = 0xFA50_0014)

This register configures MIPI DSI master.

DSIM_ESCMODE	Bit	Description	Initial State
STOPstate_Cnt	[31:21]	After transmitting read packet or write “set_tear_on” command, BTA requests to D-phy automatically. This counter value specifies the interval value between transmitting read packet (or write “set_tear_on” command) and BTA request. 11'h000 = 2 EscClk 11'h001 = 2 EscClk + 1 EscClk ~ 11'h3FF = 2 EscClk + 1023 EscClk	0
ForceStopstate	[20]	Forces Stopstate for D-PHY.	0
Reserved	[19:17]	Reserved	-
ForceBta	[16]	Forces Bus Turn Around. 1 = Sends the protocol layer request to D-PHY. MIPI DSI peripheral becomes master after BTA sequence. This bit clears automatically after receiving BTA acknowledge from MIPI DSI peripheral.	0
Reserved	[15:8]	Reserved	-
CmdLpd़t	[7]	Specifies LPDT transfers command in SFR FIFO. 0 = HS Mode 1 = LP Mode	0
TxLpd़t	[6]	Specifies data transmission in LP mode (all data transfer in LPDT). 0 = HS Mode 1 = LP Mode	0
Reserved	[5]	Reserved	-
TxTriggerRst	[4]	Specifies remote reset trigger function. After trigger operation, these bits will be cleared automatically.	0
TxUlpsDat	[3]	Specifies ULPS request for data lane. Manually clears after ULPS exit.	0
TxUlpsExit	[2]	Specifies ULPS exit request for data lane. Manually clears after ULPS exit.	0
TxUlpsClk	[1]	Specifies ULPS request for clock lane. Manually clears after ULPS exit.	0
TxUlpsClkExit	[0]	Specifies ULPS exit request for clock lane. Manually clears after ULPS exit.	0

3.3.1.7 Main Display Image Resolution Register (DSIM_MDRESOL, R/W, Address = 0xFA50_0018)

DSIM_MDRESOL	Bit	Description	Initial State
MainStandby	[31]	Specifies standby for receiving DISPCON output in Command mode after setting all configuration. 0 = Not ready 1 = Stand by Standby should be set after configuration (resolution, reqtype, pixelform, and so on) is set for command mode. In Video mode, if this bit value is 0, data is not transferred.	0
Reserved	[30:27]	Reserved	-
MainVResol[10:0]	[26:16]	Specifies Vertical resolution (1 ~ 768).	0x300
Reserved	[15:11]	Reserved	-
MainHResol[10:0]	[10:0]	Specifies Horizontal resolution (1 ~ 1024).	0x400

3.3.1.8 Main Display VPORCH Register (DSIM_MVPORCH, R/W, Address = 0xFA50_001C)

DSIM_MVPORCH	Bit	Description	Initial State
CmdAllow	[31:28]	Specifies the number of horizontal lines, where command packet transmission is allowed after Stable VFP period. For more information, see Figure 3-12	0xF
Reserved	[27]	Reserved	-
StableVfp[10:0]	[26:16]	Specifies the number of horizontal lines, where command packet transmission is not allowed after end of active region. For more information, see Figure 3-12 *Note: In Command mode, these bits are ignored.	0
Reserved	[15:11]	Reserved	-
MainVbp[10:0]	[10:0]	Specifies vertical back porch width for Video mode (line count). In Command mode, these bits are ignored.	0

NOTE: * Transfers command packets after Stable VFP area. Display controller VFP lines should be set based on sum of these values: Stable VFP, command allowing area and command masked area. See the section for transferring general data in Video mode.

3.3.1.9 Main Display HPORCH Register (DSIM_MHPORCH, R/W, Address = 0xFA50_0020)

DSIM_MHPORCH	Bit	Description	Initial State
MainHfp[15:0]	[31:16]	Specifies the horizontal front porch width for Video mode. HFP is specified using blank packet. These bits specify the word counts for blank packet in HFP. In Command mode, these bits are ignored.	0
MainHbp[15:0]	[15:0]	Specifies the horizontal back porch width for Video mode. HBP is specified using blank packet. These bits specify the word counts for blank packet in HBP. In Command mode, these bits are ignored.	0

3.3.1.10 Main Display Sync Area Register (DSIM_MSYNC, R/W, Address = 0xFA50_0024)

DSIM_MSYNC	Bit	Description	Initial State
MainVsa[9:0]	[31:22]	Specifies the vertical sync pulse width for Video mode (Line count). In command mode, these bits are ignored.	0
Reserved	[21:16]	Reserved	-
MianHsa[15:0]	[15:0]	Specifies the horizontal sync pulse width for Video mode. HSA is specified using blank packet. These bits specify word counts for blank packet in HSA. In command mode, these bits are ignored.	0

3.3.1.11 Sub Display Image Resolution Register (DSIM_SDRESOL, R/W, Address = 0xFA50_0028)

DSIM_SDRESOL	Bit	Description	Initial State
SubStandby	[31]	Specifies standby for receiving DISPCON output in Command mode after setting all configuration. 0 = Not ready 1 = Standby Standby should be set after configuration (resolution, reqtype, pixelform, and so on) is set for command mode. In Video mode, this bit is ignored.	0
Reserved	[30:27]	Reserved	-
SubVResol[10:0]	[26:16]	Specifies the Vertical resolution (1 ~ 768).	0x300
Reserved	[15:11]	Reserved	-
SubHResol[10:0]	[10:0]	Specifies the Horizontal resolution (1 ~ 1024).	0x400

3.3.1.12 Interrupt Source Register (DSIM_INTSRC, R/W, Address = 0xFA50_002C)

This register identifies interrupt sources.

Internal block error, data transmit interrupt, inter-layer (D_PHY) error, etc.

The bits are set even if they are masked off by DSIM_INTMSK_REG.

Write '1' to clear the Interrupt.

DSIM_INTSRC	Bit	Description	Initial State
PLLStable	[31]	Indicates that D-phy PLL is stable.	0
SwRstRelease	[30]	Releases the software reset.	0
SFRFifoEmpty	[29]	Specifies the SFR payload FIFO empty.	0
Reserved	[28:26]	Reserved	-
BusTurnOver	[25]	Indicates when bus grant turns over from DSI slave to DSI master.	0
FrameDone	[24]	Indicates when MIPI DSIM transfers the whole image frame. Note: If Hsync is not received during two line times, internal timer is timed out and this bit is flagged.	0
Reserved	[23:22]	Reserved	-
LpdrTout	[21]	Specifies the LP Rx timeout. See time out register (0x10).	0
TaTout	[20]	Turns around Acknowledge Timeout. See time out register (0x10).	0
Reserved	[19]	Reserved	-
RxDatDone	[18]	Completes receiving data.	0
RxTE	[17]	Receives TE Rx trigger.	0
RxAck	[16]	Receives ACK Rx trigger.	0
ErrRxECC	[15]	Specifies the ECC multi bit error in LPDR.	0
ErrRxCRC	[14]	Specifies the CRC error in LPDR.	0
ErrEsc3	[13]	Specifies the escape mode entry error lane 3. For more information, refer to standard D-PHY specification.	0
ErrEsc2	[12]	Specifies the escape mode entry error lane 2. For more information, refer to standard D-PHY specification.	0
ErrEsc1	[11]	Specifies the escape mode entry error lane 1. For more information, refer to standard D-PHY specification.	0
ErrEsc0	[10]	Specifies the escape mode entry error lane 0. For more information, refer to standard D-PHY specification.	0
ErrSync2	[9]	Specifies the LPDT sync error lane 3. For more information, refer to standard D-PHY specification.	0
ErrSync2	[8]	Specifies the LPDT Sync Error lane2. For more information, refer to standard D-PHY specification.	0
ErrSync1	[7]	Specifies the LPDT Sync Error lane1. For more information, refer to standard D-PHY specification.	0
ErrSync0	[6]	Specifies the LPDT Sync Error lane0. For more information, refer to standard D-PHY specification.	0

DSIM_INTSRC	Bit	Description	Initial State
ErrControl2	[5]	Controls Error lane3. For more information, refer to standard D-PHY specification.	0
ErrControl2	[4]	Controls Error lane2. For more information, refer to standard D-PHY specification.	0
ErrControl1	[3]	Controls Error lane1. For more information, refer to standard D-PHY specification.	0
ErrControl0	[2]	Controls Error lane0. For more information, refer to standard D-PHY specification.	0
ErrContentLP0	[1]	Specifies the LP0 Contention Error (only lane0, because BTA occurs at lane0 only). For more information, refer to standard D-PHY specification.	0
ErrContentLP1	[0]	Specifies the LP1 Contention Error (only lane0, because BTA occurs at lane0 only). For more information, refer to standard D-PHY specification.	0

3.3.1.13 Interrupt Mask Register (DSIM_INTMSK, R/W, Address = 0xFA50_0030)

This register masks interrupt sources.

DSIM_INTMSK	Bit	Description	Initial State
MskPIIStable	[31]	Indicates that D-PHY PLL is stable.	-
MskSwRstRelease	[30]	Releases software reset.	0
MskSFRFifoEmpty	[29]	Empties SFR payload FIFO.	1
Reserved	[28:26]	Reserved	-
MskBusTurnOver	[25]	Indicates when bus grant turns over from DSI slave to DSI master.	1
MskFrameDone	[24]	Indicates when MIPI DSIM transfers whole image frame.	1
Reserved	[23:22]	Reserved	-
MskLpdrTout	[21]	Specifies LP Rx timeout. See time out register (0x10).	1
MskTaTout	[20]	Specifies turnaround acknowledge timeout. See time out register (0x10)	1
Reserved	[19]	Reserved	-
MskRxDatDone	[18]	Specifies completion of data receiving.	1
MskRxTE	[17]	Specifies receipt of TE Rx trigger.	1
MskRxAck	[16]	Specifies receipt of ACK Rx trigger.	1
MskRxECC	[15]	Specifies ECC multibit error in LPDR.	1
MskRxCRC	[14]	Specifies CRC error in LPDR.	1
MskEsc3	[13]	Specifies escape mode entry error in lane3. For more information, refer to standard D-PHY specification.	1
MskEsc2	[12]	Specifies escape mode entry error in lane2. For more information, refer to standard D-PHY specification.	1
MskEsc1	[11]	Specifies escape mode entry error in lane1. For more information, refer to standard D-PHY specification.	1
MskEsc0	[10]	Specifies escape mode entry error in lane0. For more information, refer to standard D-PHY specification.	1
MskSync3	[9]	Specifies LPDT sync error in lane3. For more information, refer to standard D-PHY specification.	1
MskSync2	[8]	Specifies LPDT sync error in lane2. For more information, refer to standard D-PHY specification.	1
MskSync1	[7]	Specifies LPDT sync error in lane1. For more information, refer to standard D-PHY specification.	1
MskSync0	[6]	Specifies LPDT sync error in lane0. For more information, refer to standard D-PHY specification.	1
MskControl3	[4]	Controls error in lane3. For more information, refer to standard D-PHY specification.	1
MskControl2	[4]	Controls error in lane2. For more information, refer to standard D-PHY specification.	1
MskControl1	[3]	Controls error in lane1. For more information, refer to standard D-PHY specification.	1

DSIM_INTMSK	Bit	Description	Initial State
		standard D-PHY specification.	
MskControl0	[2]	Controls error in lane0. For more information, refer to standard D-PHY specification.	1
MskContentLP0	[1]	Specifies LP0 contention error. For more information, refer to standard D-PHY specification.	1
MskContentLP1	[0]	Specifies LP1 contention error. For more information, refer to standard D-PHY specification.	1

3.3.1.14 Packet Header FIFO Register (DSIM_PKTHDR, W, Address = 0xFA50_0034)

This register is the FIFO for packet header to send DSI packets.

DSIM_PKTHDR	Bit	Description	Initial State
Reserved	[31:24]	Reserved	-
PacketHeader	[23:0]	Writes the packet header of Tx packet. [7:0] = DI [15:8] = Dat0 (Word Count lower byte for long packet) [23:16] = Dat1 (Word Count upper byte for long packet)	0

3.3.1.15 Payload FIFO Register (DSIM_PAYLOAD, W, Address = 0xFA50_0038)

This register specifies the FIFO for payload to send DSI packets.

DSIM_PAYLOAD	Bit	Description	Initial State
Payload	[31:0]	Writes the Payload of Tx packet.	0

3.3.1.16 Read FIFO Register (DSIM_RXFIFO, R, Address = 0xFA50_003C)

This register is the gate of FIFO read

DSIM_RXFIFO	Bit	Description	Initial State
RxDat	[31:0]	In the Rx mode, you can read Rx data through this register. Note that the CRC in packet is not stored in RxFIFO.	Unknown

3.3.1.17 PLL Control Register (DSIM_PLLCTRL, R/W, Address = 0xFA50_004C)

This register configures PLL control, D-PHY, clock range indication, and so on.

DSIM_PLLCTRL	Bit	Description	Initial State
Reserved	[31:28]	Should be 0.	-
FreqBand[3:0]	[27:24]	Indicates Bitclk frequency band for D-PHY global timing. For more information, refer to Table 3-7 .	0
PllEn	[23]	Enables PLL.	0
Reserved	[22:20]	Should be 0.	-
PMS[19:1]	[19:1]	Specifies the PLL PMS value.	0
Reserved	[0]	Reserved	0

3.3.1.18 PLL Timer Register (DSIM_PLLTMR, R/W, Address = 0xFA50_0050)

DSIM_PLLTMR	Bit	Description	Initial State
PllTimer	[31:0]	Specifies the PLL Timer for stability of the generated clock (System clock cycle base). If the timer value goes to 0x00000000, the clock stable bit of status and interrupt register is set.	0xFFFFFFFF

3.3.1.19 DSIM D-PHY AC Characteristic Register (DSIM_PHYACCHR, R/W, Address = 0xFA50_0054)

DSIM_PHYACCHR	Bit	Description	Initial State
Reserved	[31:15]	Reserved. Should be 0.	0
AFC_EN	[14]	Enables AFC. 0 = Disables 1 = Enables	0
Reserved	[13:8]	Reserved. Should be 0.	0
AFC_CTL	[7:5]	Specifies the AFC control value for MIPI DPHY. This value is meaningful when AFC_EN is 1. Refer to Table 8.7-6 for more information.	0
Reserved	[4:0]	Reserved. Should be 0.	0

3.3.1.20 DSIM D-PHY AC Characteristic Register 1 (DSIM_PHYACCCHR1, R/W, Address = 0xFA50_0058)

DSIM_PHYACCCHR	Bit	Description	Initial State
Reserved	[31:7]	Reserved. Should be 0.	0
Reserved	[6:4]	Reserved	0
Reserved	[3:2]	Reserved	0
DpDnSwap_CLK	[1]	Swaps Dp/Dn channel of clock lane. If this bit is set, Dp and Dn channel swap each other.	0
DpDnSwap_DAT	[0]	Swaps Dp/Dn channel of Data lanes. If this bit is set, Dp and Dn channel swap each other.	0

3.4 DPHY PLL CONTROL

3.4.1 PMS SETTING SAMPLE FOR MIPI PLL

3.4.1.1 PMS Setting

Writes a value to PMS field in DSIM_PLLCTRL (0xFA50_004C) to set PMS value for DPHY PLL. Each P, M, and S resides in PMS [18:13], PMS[12:4], and PMS[3:1] .

Before setting the PMS value, follow these instructions:

1. Do not set the P or M value as zero, since setting the P(00000) or M(0000000000) causes malfunction of the PLL.
2. The selected M value should be chosen within the range of 41~125 for PLL stability. The VCO output frequency range of MIPI_PLL varies from 500MHz to 1000MHz
3. Keep to range of Fin_pll varies from 6 MHz to 12 MHz, using P code setting.

Table 3-5 PMS and Frequency Constraint

	Function	Value	Description
Fin	Fin	6~200MHz	Specifies PLL input frequency.
Fin_pll	Fin/P	6 ~ 12 MHz	Specifies PFD input frequency.
VCO_out	(M*Fin)/P	500 ~ 1000 MHz	Specifies VCO output frequency.
Fout	(M*Fin)/(P*2^S)	15.625 ~ 1000 MHz	Specifies PLL output frequency.
P[5:0]	P	1 ~ 63	Specifies PMS[18:13].
M[8:0]	M	20 ~ 511	Specifies PMS[12:4].
S[2:0]	2^S	1, 2, 4, 8, 16, 32	Specifies PMS[3:1].

Table 3-6 AFC Code

Serial Clock	AFC_CTL[2:0]
6 ~ 6.99 MHz	001
7 ~ 7.99 MHz	000
8 ~ 8.99 MHz	011
9 ~ 9.99 MHz	010
10 ~ 10.99 MHz	101
11 ~ 12 MHz	100

Table 3-7 Band Control Setting

Serial Clock (= ByteClk X 8)	FreqBand[3:0]
~ 99.99 MHz	0000
100 ~ 119.99 MHz	0001
120 ~ 159.99 MHz	0010
160 ~ 199.99 MHz	0011
200 ~ 239.99 MHz	0100
140 ~ 319.99 MHz	0101
320 ~ 389.99 MHz	0110
390 ~ 449.99 MHz	0111
450 ~ 509.99 MHz	1000
510 ~ 559.99 MHz	1001
560 ~ 639.99 MHz	1010
640 ~ 689.99 MHz	1011
690 ~ 769.99 MHz	1100
770 ~ 869.99 MHz	1101
870 ~ 949.99 MHz	1110
950 ~ 1000 MHz	1111

3.4.1.2 Sample for Fout 80 MHz

To set PMS value for Fout 80 MHz, refer to the following table.

	Case 1	Case 2
Fin	24MHz	30MHz
Fin_pll	8MHz	10MHz
P[5:0]	3	3
M[8:0]	80	64
S[2:0]	8	8
VCO_out	640MHz	640MHz
Fout	80MHz	80MHz

3.4.1.3 Sample for Fout 1000 MHz

To set PMS value for Fout 1000 MHz, refer to the following table.

	Case 1	Case 2
Fin	24MHz	30MHz
Fin_pll	8MHz	10MHz
P[5:0]	3	3
M[9:0]	125	100
S[2:0]	1	1
VCO_out	1000MHz	1000MHz
Fout	1000MHz	1000MHz

3.4.1.4 Sample for Fout 999 MHz

To set PMS value for Fout 999 MHz, refer to the following table.

	Case 1	Case 2
Fin	27MHz	9MHz
Fin_pll	9MHz	9MHz
P[5:0]	3	1
M[9:0]	111	111
S[2:0]	1	1
VCO_out	999MHz	999MHz
Fout	999MHz	999MHz

4 MIPI CSIS

4.1 OVERVIEW OF MIPI CSIS

The key features of MIPI CSIS include:

- Compliance to MIPI CSI2 Standard Specification Version 1.0
 - Supports 1, 2, 3, or 4 data lanes
 - Supports 1 channel
 - Supports RAW8, RAW10, RAW12, and YUV422 8-bit
 - All of User defined Byte-based Data packet
- Interfaces
 - Compatible with PPI (Protocol-to-PHY Interface) in MIPI D-PHY Specification Version 0.90

4.2 BLOCK DIAGRAM

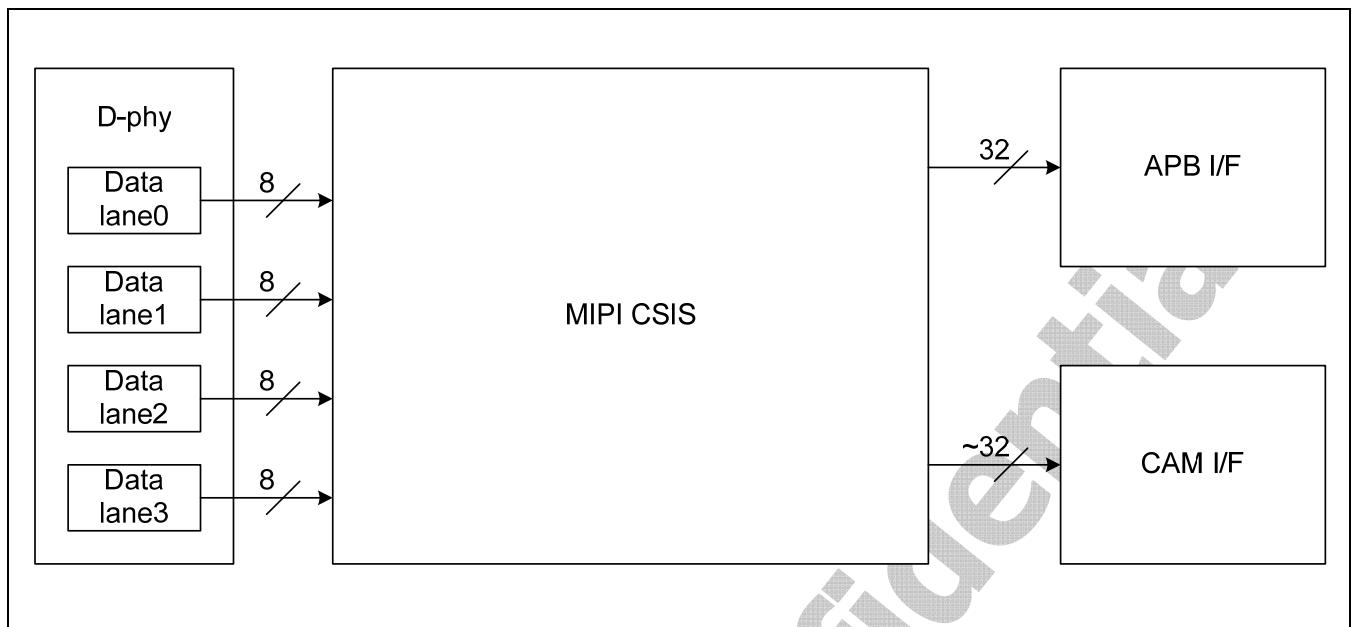


Figure 4-1 MIPI CSI System Block Diagram

4.3 INTERFACE AND PROTOCOL

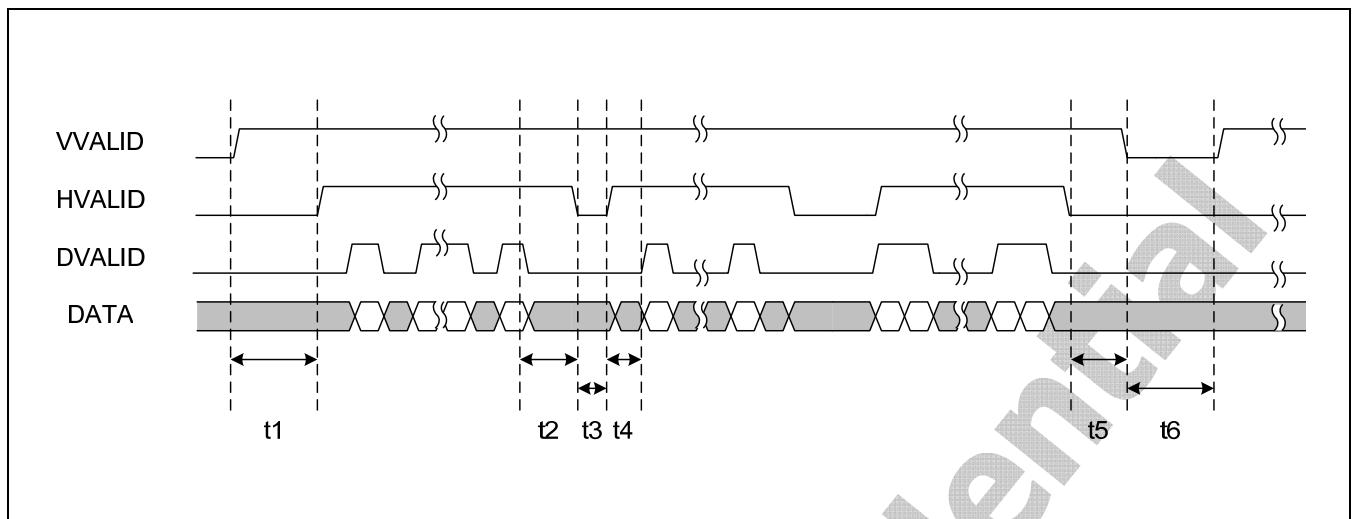


Figure 4-2 Waveform of Output Data

Table 4-1 Timing Diagram of Output Data

	Description	Minimum Cycle of Pixel Clock
t1	Specifies the interval between rising of VVALID and first rising of HVALID.	Vsync_SIntv + 1 (1 ~ 64)
t2	Specifies the interval between last falling of DVALID and falling of HVALID.	Hsync_LIntv + 2 (2 ~ 66)
t3	Specifies the interval between falling of HVALID and rising of next HVALID.	1
t4	Specifies the interval between rising of HVALID and first rising of DVALID.	0
t5	Specifies the interval between last falling of HVALID and falling of VVALID.	Vsync_EIntv (0 ~ 4095)
t6	Specifies the interval between falling of VVALID and rising of next VVALID.	1

4.4 DATA FORMAT

4.4.1 DATA ALIGNMENT

CSIS supports two type of data alignment, as illustrated in [Figure 4-3](#).

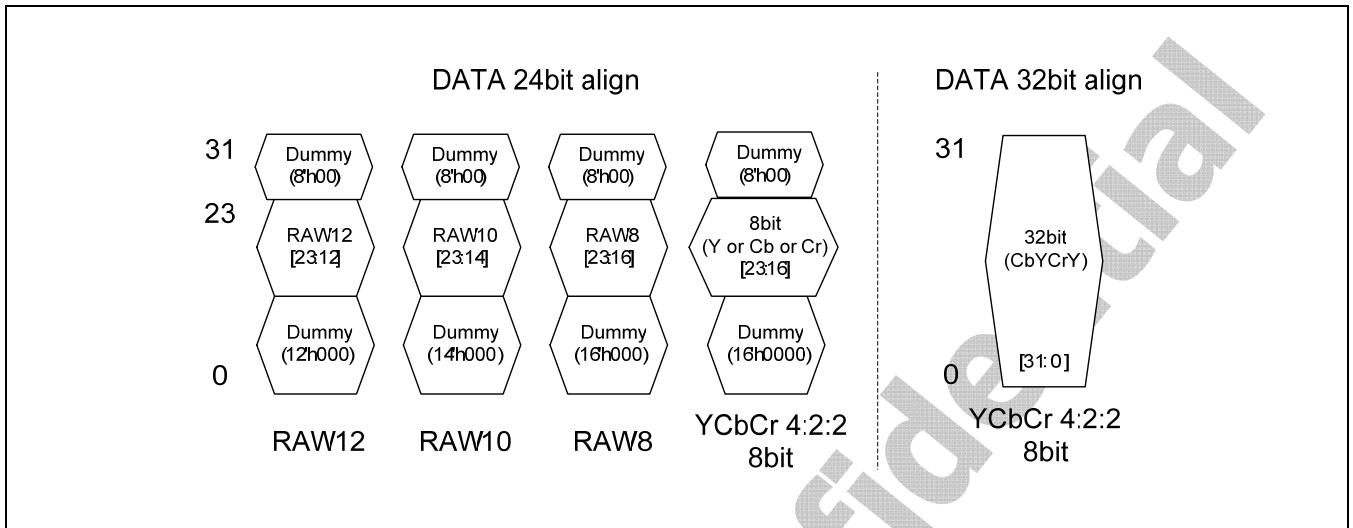


Figure 4-3 MIPI CSIS Data Alignment

4.4.2 YUV422 8-BIT ORDER

YUV422 8-bit format data is stored as a UYVY sequence, as specified in [Table 4-2](#).

Table 4-2 Data Order of YUV422 Alignment

Format	Stream Order of content	24-bit Alignment	32-bit Alignment
YUV422 8-bit	U1→Y1→V1→Y2→ ...	DATA1[23:16] = U1	DATA1[31:24] = U1
		DATA2[23:16] = Y1	DATA1[23:16] = Y1
		DATA3[23:16] = V1	DATA1[15:8] = V1
		DATA4[23:16] = Y2	DATA1[7:0] = Y2

4.5 I/O DESCRIPTION

Signal	I/O	Description	Pad	Type
DPDATA0	B	Specifies the DP signal for MIPI-DPHY slave data-lane 0.	XmipiSDP0	Dedicated
DNDATA0	B	Specifies the DN signal for MIPI-DPHY slave data-lane 0.	XmipiSDN0	Dedicated
DPDATA1	B	Specifies the DP signal for MIPI-DPHY slave data-lane 1.	XmipiSDP1	Dedicated
DNDATA1	B	Specifies the DN signal for MIPI-DPHY slave data-lane 1.	XmipiSDN1	Dedicated
DPDATA2	B	Specifies the DP signal for MIPI-DPHY slave data-lane 2.	XmipiSDP2	Dedicated
DNDATA2	B	Specifies the DN signal for MIPI-DPHY slave data-lane 2.	XmipiSDN2	Dedicated
DPDATA3	B	Specifies the DP signal for MIPI-DPHY slave data-lane 3.	XmipiSDP3	Dedicated
DNDATA3	B	Specifies the DN signal for MIPI-DPHY slave data-lane 3.	XmipiSDN3	Dedicated
DPCLK	B	Specifies the DP signal for MIPI-DPHY slave clock-lane.	XmipiSDPCLK	Dedicated
DNCLK	B	Specifies the DN signal for MIPI-DPHY slave clock-lane.	XmipiSDNCLK	Dedicated

NOTE:

1. I/O direction. I: input, O: output, and B: bi-direction.
2. Type field indicates whether pads are dedicated to signal or connected to the multiplexed signals.

4.6 REGISTER DESCRIPTION

4.6.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
CSIS_CONTROL	0xFA60_0000	R/W	Specifies the control register.	0x0010_0000
CSIS_DPHYCTRL	0xFA60_0004	R/W	Specifies the D-PHY control register.	0x0000_0000
CSIS_CONFIG	0xFA60_0008	R/W	Specifies the configuration register.	0x0000_0000
CSIS_DPHYSTS	0xFA60_000C	R	Specifies the D-PHY stop state register.	0x0000_00F1
CSIS_INTMSK	0xFA60_0010	R/W	Specifies the interrupt mask register.	0x0000_0000
CSIS_INTSRC	0xFA60_0014	R/W	Specifies the interrupt status register.	0x0000_0000
CSIS_RESOL	0xFA60_002C	R/W	Specifies the image resolution register.	0x8000_8000
SDW_CONFIG	0xFA60_0038	R/W	Specifies the shadow register of configuration.	0x0000_0000
SDW_RESOL	0xFA60_003C	R	Specifies the shadow register of resolution.	0x8000_8000
CSIS_PKTDATA	0xFA60_2000 ~ 0xFA60_3FFC	R	Specifies the memory area for storing non-image data. Odd frame: 0x2000 ~ 0x2FFC Even frame: 0x3000 ~ 0x3FFC	0xFFFF_FFFF

NOTE: S_RESETN at MIPI_PHY_CON0 (0xE010_E814) should be '1' before enabling CSIS.

4.6.1.1 Control Register (CSIS_CONTROL, R/W, Address = 0xFA60_0000)

CSIS_CONTROL	Bit	Description	Initial State
S_DpDn_Swap_Clk	[31]	Swaps Dp channel and Dn channel of clock lanes. 0 = Default 1 = Swaps	0
S_DpDn_Swap_Dat	[30]	Swaps Dp channel and Dn channel of data lanes. 0 = Default 1 = Swaps	0
Reserved	[29:21]	Should be 0.	0
Parallel	[20]	Specifies data alignment size. Refer to 4.4 "Data Format" . 0 = 24-bit data alignment 1 = 32-bit data alignment	1
Reserved	[19:17]	Should be 0.	0
Update_Shadow	[16]	Updates the shadow registers. 0 = Default 1 = Updates the shadow registers After configuration, set this bit for updating shadow registers. This bit is cleared automatically after updating shadow registers.	0
Reserved	[15:9]	Should be 0.	0
WCLK_Src	[8]	Specifies wrapper clock source. 0 = PCLK 1 = EXTCLK This bit determines the source of pixel clock, which transfers image data to CAMIF.	0
Reserved	[7:5]	Should be 0.	0
SwRst	[4]	Specifies software reset. 0 = No reset 1 = Reset All writable registers in CSIS return to their reset value. After this bit is active for three cycles, this bit is de-asserted automatically. Note: Almost all MIPI CSIS blocks use “ByteClk” from D-PHY. “ByteClk” is not a continuous clock. You must assert software reset if the camera module is turned off.	0
Reserved	[3:1]	Reserved	0
Enable	[0]	Specifies the CSIS system on/ off. 0 = Off 1 = On If this bit is low even though the CSIS clock is alive, then any request from CSIS is not serviced and kept waiting. Once the main host disables CSIS, it should be reset by software or hardware before the main host enables CSIS again.	0

4.6.1.2 D-PHY Control Register (CSIS_DPHYCTRL, R/W, Address = 0xFA60_0004)

This register controls D-PHY.

CSIS_DPHYCTRL	Bit	Description	Initial State
Reserved	[31:4]	Should not change the value.	0
DPHYOn	[4:0]	Enables D-PHY clock and data lane. [4]: Data lane 3 [3]: Data lane 2 [2]: Data lane 1 [1]: Data lane 0 [0]: Clock lane 0 = Disables 1 = Enables	0

4.6.1.3 Configuration Register (CSIS_CONFIG, R/W, Address = 0xFA60_0008)

CSIS_CONFIG	Bit	Description	Initial State
Hsync_LIntv	[31:26]	Specifies the interval between Hsync falling and Hsync rising (Line interval). As shown in Figure 4-2 , t2 specifies this interval. 6'h00 ~ 6'h3F cycle of Pixel clock.	0
Vsync_SIntv	[25:20]	Specifies the interval between Vsync rising and first Hsync rising. As shown in Figure 4-2 , t1 specifies this interval. 6'h00 ~ 6'h3F cycle of Pixel clock	0
Vsync_EIntv	[19:8]	Specifies the interval between last Hsync falling and Vsync falling. As shown in Figure 4-2 , t5 specifies this interval. 12'h000 ~ 12'hFFF cycle of Pixel clock	0
DataFormat[5:0]	[7:2]	Specifies the image data format. 0x1E = YUV422 (8-bit) 0x2A = RAW8 0x2B = RAW10 0x2C = RAW12 0x30 = user defined 1 0x31 = user defined 2 0x32 = user defined 3 0x33 = user defined 4 Others = Reserved	0
NumOfDatLane	[1:0]	Specifies the number of data lanes. 00 = 1 Data Lane 01 = 2 Data Lane 10 = 3 Data Lane 11 = 4 Data Lane	0

4.6.1.4 DPHY State Register (CSIS_DPHYSTS, R, Address = 0xFA60_000C)

CSIS_DPHYSTS	Bit	Description	Initial State
Reserved	[31:12]	Reserved	0
UlpsDat	[11:8]	Determines whether the data lane [3:0] is in ULPS. [7]: Data lane 3 [6]: Data lane 2 [5]: Data lane 1 [4]: Data lane 0 0 = Not ULPS 1 = ULPS	0
StopStateDat	[7:4]	Determines whether the data lane [3:0] is in Stop state. [7]: Data lane 3 [6]: Data lane 2 [5]: Data lane 1 [4]: Data lane 0 0 = Not Stop state 1 = Stop state	F
Reserved	[3:2]	Reserved	0
UlpsClk	[1]	Determines whether the clock lane is in ULPS. 0 = Not ULPS 1 = ULPS	0
StopStateClk	[0]	Determines whether the clock lane is in Stop state. 0 = Not Stop state 1 = Stop state	1

4.6.1.5 Interrupt Mask Register (CSIS_INTMSK, R/W, Address = 0xFA60_0010)

This register masks the interrupt sources.

CSIS_INTMSK	Bit	Description	Initial State
MSK_EvenBefore	[31]	Receives non-image data at even frame and before image. 0 = Disables Interrupt 1 = Enables Interrupt	0
MSK_EvenAfter	[30]	Receives non-image data at even frame and after image. 0 = Disables Interrupt 1 = Enables Interrupt	0
MSK_OddBefore	[29]	Receives non-image data at odd frame and before image. 0 = Disables Interrupt 1 = Enables Interrupt	0
MSK_OddAfter	[28]	Receives non-image data at odd frame and after image. 0 = Disables Interrupt 1 = Enables Interrupt	0
Reserved	[27:13]	Reserved	0
MSK_ERR_SOT_HS	[12]	Specifies start of transmission error. 0 = Disables Interrupt 1 = Enables Interrupt	0
Reserved	[11:6]	Reserved	0
MSK_ERR_LOST_FS	[5]	Lost of Frame Start packet 0 = Disables Interrupt 1 = Enables Interrupt	0
MSK_ERR_LOST_FE	[4]	Lost of Frame End packet 0 = Disables Interrupt 1 = Enables Interrupt	0
MSK_ERR_OVER	[3]	Controls error. 0 = Disables Interrupt 1 = Enables Interrupt	0
MSK_ERR_ECC	[2]	Specifies ECC error. 0 = Disables Interrupt 1 = Enables Interrupt	0
MSK_ERR_CRC	[1]	Specifies CRC error. 0 = Disables Interrupt 1 = Enables Interrupt	0
MSK_ERR_ID	[0]	Specifies unknown ID error. 0 = Disables Interrupt 1 = Enables Interrupt	0

4.6.1.6 Interrupt Source Register (CSIS_INTSRC, R/W, Address = 0xFA60_0014)

This register identifies interrupt sources.

CSIS_INTSRC	Bit	Description	Initial State
EvenBefore	[31]	Receives non-image data at even frame and before image. Write 1 = Clears the status bit Write 0 = No effect	0
EvenAfter	[30]	Receives non-image data at even frame and after image. Write 1 = Clears the status bit Write 0 = No effect	0
OddBefore	[29]	Receives non-image data at odd frame and before image. Write 1 = Clears the status bit Write 0 = No effect	0
OddAfter	[28]	Receives non-image data at odd frame and after image. Write 1 = Clears status bit Write 0 = No effect	0
Reserved	[27:16]	Reserved	0
ERR_SOT_HS	[15:12]	Specifies start of transmission error.	0
Reserved	[11:6]	Reserved	0
ERR_LOST_FS	[5]	Indicates the lost of Frame Start packet	0
ERR_LOST_FE	[4]	Indicates the lost of Frame End packet	0
ERR_OVER	[3]	Specifies overflow caused in image FIFO. The outer bandwidth has to be faster than the input bandwidth. However, image FIFO can overflow due to user fault. There are two ways to prevent overflow: Tune output pixel clock faster than current: WCLK_Src in CSIS_CTRL register should be set to 1. Then assign a faster clock. Tune input byte clock slower than current: Set register in camera module through I2C channel. When this interrupt is generated, Turn the camera off. Assert software reset. If you do not assert software reset, MIPI CSIS will not receive any data. Tune the clock frequency and re-configure all the related registers. MIPI CSIS module is now ready for operation. Write 1 = Clears the status bit. Write 0 = Has no effect.	0
ERR_ECC	[2]	Specifies ECC error. Write 1 = Clears status bit Write 0 = No effect	0
ERR_CRC	[1]	Specifies CRC error. Write 1 = Clears status bit Write 0 = No effect	0

CSIS_INTSRC	Bit	Description	Initial State
ERR_ID	[0]	Specifies unknown ID error. Write 1 = Clears status bit Write 0 = No effect	0

4.6.1.7 Resolution Register (CSIS_RESOL, R/W, Address = 0xFA60_002C)

CSIS_RESOL	Bit	Description	Initial State
HResol	[31:16]	Specifies horizontal image resolution. Input boundary of each image format is as follows: YUV422 (8-bit): 0x0001 ~ 0xFFFF RAW8: 0x0001 ~ 0xFFFF RAW10: 4n (where n is 1, 2, 3, ...) RAW12: 2n (where n is 1, 2, 3, ...)	0x8080
VResol	[15:0]	Specifies vertical image resolution. Input boundary: 0x0001 ~ 0xFFFF	0x8080

4.6.1.8 Shadow Configuration Register (CSIS_sdw_config, R, Address = 0xFA60_0038)

CSIS_SDW_CONFIG	Bit	Description	Initial State
Hsync_LIntv	[31:26]	Specifies current interval between Hsync falling and Hsync rising (Line interval).	0
Vsync_SIntv	[25:20]	Specifies current interval between Vsync rising and first Hsync rising.	0
Vsync_EIntv	[19:8]	Specifies current interval between last Hsync falling and Vsync falling.	0
DataFormat[5:0]	[7:2]	Specifies current image data format.	0
NumOfDatLane[1:0]	[1:0]	Specifies current number of data lanes. These bits are always the same as the number of data lanes in CSIS_CONFIG register because these bits are static signals that do not change in operation.	0

4.6.1.9 Shadow Resolution Register (CSIS_SDW_resol, R, Address = 0xFA60_003C)

CSIS_PKTDATA	Bit	Description	Initial State
HResol	[31:16]	Specifies current horizontal image resolution.	0
VResol	[15:0]	Specifies current vertical image resolution.	0

4.6.1.10 Packet Data Register (CSIS_PKTDATA, R, Address = 0xFA60_2000 ~ 0xFA60_3FFC)

CSIS_PKTDATA	Bit	Description	Initial State
PktData	[31:0]	Specifies packet data.	Unknown

5 G3D

5.1 OVERVIEW OF G3D

The G3D block is based on the SGX540 core from Imagination Technologies. SGX represents a new generation of programmable PowerVR graphics IP cores. The PowerVR SGX architecture is scalable and can target all market segments--from mainstream mobile devices to high-end desktop graphics. The PowerVR SGX540 core is designed for feature phones, PDA, and handheld gaming applications.

PowerVR SGX processes a number of differing multimedia data types concurrently, such as:

- Pixel Data
- Vertex Data
- General Purpose Processing

5.1.1 KEY FEATURES OF G3D

The key features of G3D include:

- Supports 3D and vector graphics on common hardware
- Uses Tile-based architecture
- Incorporates Universal Scalable Shader engine, which is a multi-threaded engine with Pixel and Vertex Shader functionality
- Supports industry standard APIs such as OGL-ES 1.1 and 2.0, OpenVG 1.0
- Supports multiple operating systems such as Symbian, Linux, WinCE, and future versions of these systems
- Enables fine-grained task switching, load balancing, and power management
- Supports advanced geometry DMA driven operation for minimum CPU interaction
- Supports programmable high-quality image anti-aliasing
- Enables fully virtualized memory addressing for smooth functioning of operating system in a unified memory architecture
- Supports standard master and slave AXI bus interfaces

5.1.2 3D FEATURES IN G3D

The 3D features in G3D include:

- Supports deferred pixel shading
- Uses on-chip tile floating point depth buffer
- Supports 8-bit stencil with on-chip tile stencil buffer
- Provides eight parallel depth/ stencil tests per clock
- Supports scissor test
- Supports textures such as:
 - Cube Map
 - Projected
 - Non-square
- Supports texture formats such as:
 - RGBA: 8888, 4444, 565, 1555, and 1565
 - Monochromatic: 8, 16, 16f, 32f, and 32int
 - Dual channel: 8:8, 16:16, and 16f:16f
 - Compressed textures: PVR-TC1, PVR-TC2, and ETC1
 - All YUV formats (programmable)
- Supports the same resolution for both frame buffer maximum size and texture maximum size
 - Frame buffer maximum size = 2048 x 2048
 - Texture maximum size = 2048 x 2048
- Controls texture filtering
 - Bilinear, Trilinear, and Anisotropic
 - Independent minimum and mag control
- Supports anti-aliasing
 - 4x Multisampling
 - Programmable sample positions
- Supports indexed primitive list
 - Bus mastered
- Provides programmable vertex DMA
- Supports “render to texture” including twiddled formats
 - Generates auto MipMap generation
- Multiple on-chip render targets (MRT) is dependent on the availability of on-chip memory used as intermediate data stores (not included in the SGX540 core)

5.1.3 USSE FEATURES IN G3D

USSE represents the engine at centre of PowerVR SGX540 architecture. It supports a broad range of instructions. The key features of USSE in G3D include:

- Supports single programming model
 - Enables multi-threading with 16 simultaneous execution threads and up to 64 simultaneous data instances
 - Provides zero-cost swapping in and swapping out of threads
 - Supports cached program execution model with maximum program size of 4096 instructions
 - Contains dedicated pixel processing and vertex processing instructions
 - Supports 2048 32-bit registers
- Supports SIMD execution unit related operations in
 - 32-bit IEEE float
 - 2-way 16-bit fixed point
 - 4-way 8-bit integer
 - 32-bit bitwise (logical only)
- Controls static and dynamic flow in
 - Subroutine calls
 - Loops
 - Conditional branches
 - Zero-cost instruction predication
- Supports procedural geometry
 - Allows generation of primitives
 - Enables effective geometry compression
 - Supports high order surface
- Enables external data access
 - Reads from main memory via cache
 - Writes to main memory
 - Supports data fence facility
- Reads dependent texture

5.1.4 2D FEATURES IN G3D

A dedicated processing pipeline processes 2D graphics within the SGX540 core. The OpenVG API uses these advanced techniques. This API is fully supported.

The 2D features in G3D include:

- Supports x2 clock frequency, that is, twice the frequency of the SGX540 core (optional)
- Supports ROP 2, 3, and 4 (including AA text)
- Supports source, pattern (brush), destination, and mask surface
- Enables alpha blending (per-pixel and global)
- Supports color key
- Supports these input formats: 1-, 2-, 4- and 8-bit palletized; and 4- and 8-bit alpha
- Supports these input and output formats: RGB(A) -- 3:3:2, 4:4:4:4, 5:5:5, 5:5:5:1, 5:6:5, 8:8:8:8, and 8:8:8:0
- Strides up to 2048 pixels
- Provides throughput (Note: All performance figures are affected by memory bandwidth. The basic assumption is that 2D pipeline operates at x2 SGX540 core frequency).
 - All ROPs (including color fill) – Two per clock
 - Source Copy – Two per clock
 - Alpha Blends – 2 per clock
 - Rotated Blits – 2 per clock
- Supports other 2D features such as:
 - Clipping, scissoring, and masking
 - Paint generation and image interpolation
 - Geometry generation
 - Translucency
 - Gradients
 - Complex pixel filters
 - Perspective texturing
 - Transformations
 - Strokes
 - Multi segment lines, wide lines, and multi segment wide lines
 - Mitred line intersections
 - Multi vertex polygons
 - Arbitrary polygon fills using gradients, blends, and translucency
 - Smooth arbitrary rotation

5.1.5 BLOCK DIAGRAM OF SGX540

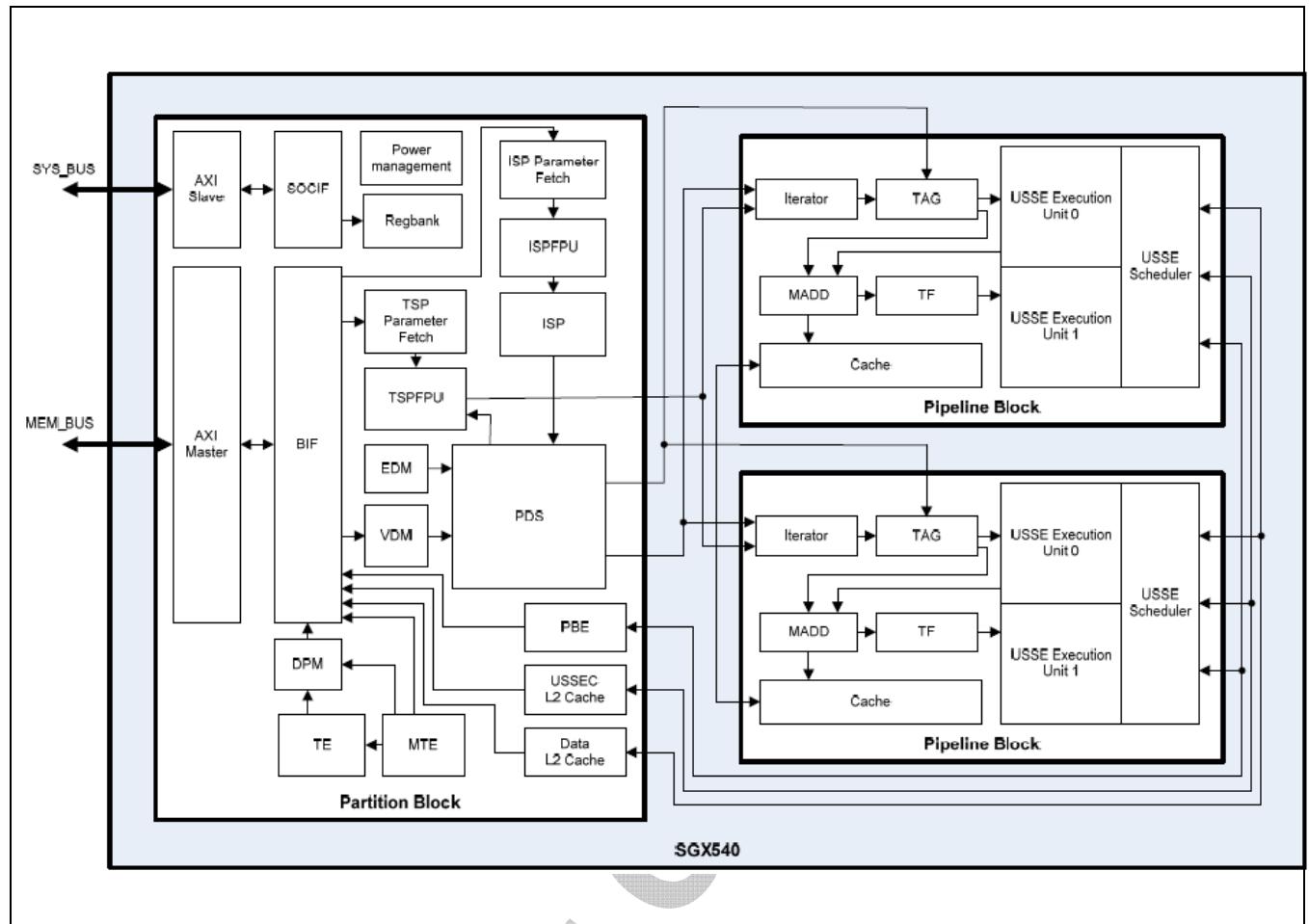


Figure 5-1 SGX540 Block Diagram

[Figure 5-1](#) describes the terms used in the block diagram above.

Table 5-1 Glossary of Terms

Term	Description	Term	Description
VS	Vertex Shader	USSE	Universal Scalable Shader Engine
PS	Pixel Shader	TF	Texture Filter
OGL	Open GL Application Programming Interface	MADD	Mux/ Arb/ Demux/ Decompress – Cache Management Module
DMS	Data Master Selector	MTE	Macro Tiling Engine
PP	Pixel Presenter	ROP	Collective Term For 2D Raster Operations
VDM	Vertex Data Master	MRT	Multiple Render Target
GPDM	General Purpose Data Master	TE	Tiling Engine
PDM	Pixel Data Master	DPM	Dynamic Parameter Management
ISP	Image Synthesis Processor – Hidden Surface Removal Engine	TSP FPU	Texturing and Shading Floating Point Setup Unit
PDS	Programmable Data Sequencer	TAG	Texture Address Generator

The SGX540 block is based on the PowerVR SGX core from Imagination Technologies.

5.1.5.1 Coarse Grain Scheduler

The Coarse Grain Scheduler (CGS) specifies the main system controller for the PowerVR SGX540 architecture. It consists of two stages, namely, the Data Master Selector (DMS) and the Programmable Data Sequencer (PDS). The DMS processes requests from data masters and determines which tasks will be executed based on the resource requirements. The PDS then controls the loading and processing of data on the USSE.

5.1.5.2 Data Master Selector

The DMS processes request from the data masters and selects a task to be executed by the USSE and PDS by tracking task resource requirements. The selected data master then has its resource allocated and source data from the individual data master, which is passed to the PDS sequencing engine.

There are three data masters in the SGX540 core, namely:

- Pixel Data Master (PDM)
- Vertex Data Master (VDM)
- General Purpose Data Master (GPDM)

5.1.5.3 Pixel Data Master

The PDM initiates “rasterize processing” within the system. It consists of the ISP and Pixel Presenter (PP) modules. Each pixel pipeline processes pixels for a different half of a given tile, which allows for optimum efficiency within each pipe due to locality of data.

5.1.5.4 VDM Data Master

The VDM initiates “transform and lighting processing” within the system. The VDM module reads an input control stream, which contains triangle index data and state data. The state data indicates the PDS program, size of the vertices, and amount of USSE output buffer resource available to the VDM. The triangle data is parsed to determine unique indices that must be processed by the USSE. These are grouped together according to the configuration provided by the driver and presented to the DMS.

5.1.5.5 General Purpose Data Master

The GPDM responds to events within the system. Each event causes an interrupt to the host or synchronized execution of a program on the PDS. The program may or may not cause a subsequent task to be executed on the USSE.

5.1.5.6 PDS

The DMS and PDS controls whether vertices, pixels, or imaging data operations are processed by the USSE. It controls the order, location, and size of these operations. It also controls two operations: fetching essential data for the USSE and allocating resources.

5.1.5.7 USSE

USSE is a user programmable processing unit. Although general in nature, its instructions and features are optimized for three types of tasks, namely, processing vertices (vertex shading), processing pixels (pixel shading), and imaging processing.

5.1.5.8 Multi-Level Cache

The multi-level cache is a level cache consisting of two modules, namely, the Main Cache and the Mux, Arbiter, Demux, De-Compression Unit (MADD).

MADD is a wrapper around the main cache module that manages and formats requests to and from the cache. It also provides Level 0 caching for texture and USSE requests.

5.1.5.9 Vertex Processing

There are three main processes within the PowerVR architecture that must be performed to generate 3D graphics.

- To create screen space representation, triangle information in the form of vertices must be transformed and be lit.
- To create display lists in memory, these transformed and lit vertices are passed through a tiling engine.
- To create the final image in the Pixel Processing pipeline, the display list in memory is rasterized on a tile-by-tile basis. The transform and light and tiling operation together can be regarded as the vertex-processing pipeline.

5.1.5.10 Transform and Lighting

A 3D object is expressed in terms of triangles, each of which is made up of three vertices with a minimum of X, Y, and Z coordinates. The basic steps to transform a typical 3D application are explained below, along with a brief description of lighting models. The transform and lighting (TNL) process within SGX540 is performed by data moving through VDM, PDS, and USSE respectively.

5.1.5.11 Macro Tiling Engine

The Macro Tiling Engine (MTE) takes in vertex and index data from the USSE and PDS, and generates a macro-tiled block of vertex index data. This data is written to memory after removing the redundant data. In addition to this, the MTE generates a set of primitive blocks for the tiling engine. A primitive block is a list of primitives, where each primitive consists of its indices and fixed point x,y of the vertices.

5.1.5.12 Tiling Engine

The Tiling Engine (TE) accepts blocks of primitive data from the MTE, and performs two incremental tiling algorithms, namely, the bounding box and perfect tiling algorithms. These algorithms produce a minimal list of tiles containing the primitives. Information about the primitives contained within the tiles is written as a control stream (display list) to memory, which is dynamically allocated by the Dynamic Parameter Management (DPM) block.

5.1.5.13 Dynamic Parameter Management

DPM ensures that SGX540 is able to render arbitrarily complex scenes. During tiling, the DPM allocates memory from a parameter memory pool, and after rasterization releases it. SGX540 breaks down the display list into groups of tiles (macro tiles), and each macro tile is rendered separately (“Partial Rendering”). As each macro tile is rendered, the results are merged with the render results from a previous macro tile to produce the correct final image.

5.1.5.14 Pixel Processing

The Pixel Processing pipeline takes the result of vertex processing process from the memory and performs a number of processes to generate the final rasterized pixels. This can be divided into three main stages:

- Hidden Surface Removal (ISP)
- Texturing and Shading (TSP, Iterators, TAG, TF, and USSE)
- Pixel Formatting (Pixel Co-Processor)

5.1.5.15 Image Synthesis Processor

To determine the visible pixels for each triangle in a given tile before being textured and shaded, the Image Synthesis Processor (ISP) specifies the first stage of pixel-processing pipeline that performs hidden surface removal. This is a key feature of the PowerVR architecture that is referred to as deferred texturing and shading. It performs a pixel accurate occlusion detection operation ahead of the computationally intensive pixel shading operations.

5.1.5.16 TSP

The TSP parameter fetches requests, and parses position and TSP vertex data from internal 3D display list--for visible primitives produced by the hidden surface removal engine (ISP).

To set up triangle for the TSP, the TSP FPU uses vertex data sourced from the TSP parameter fetch. Multiple plane equations are produced that define how colors and texture coordinate sets are interpolated across primitives.

5.1.5.17 Texture Address Generator

The Texture Address Generator (TAG) receives a set of coordinates from iterators or the USSE, along with their corresponding state information. From this, it calculates a set of addresses to perform the required texture lookup. It also generates a set of coefficients to be used by the return Texture Filter (TF) module.

5.1.5.18 Texture Filter

The Texture Filter (TF) receives data from the cache, following requests submitted by the TAG module, and filters the resultant data as required. It computes bilinear, trilinear, and anisotropic filtering results. These results are then passed into the USSE for combining with the complex pixel shader calculated colors that are written to the Pixel Co-processor module.

5.1.6 BLOCK DIAGRAM OF INTEGRATION INFORMATION

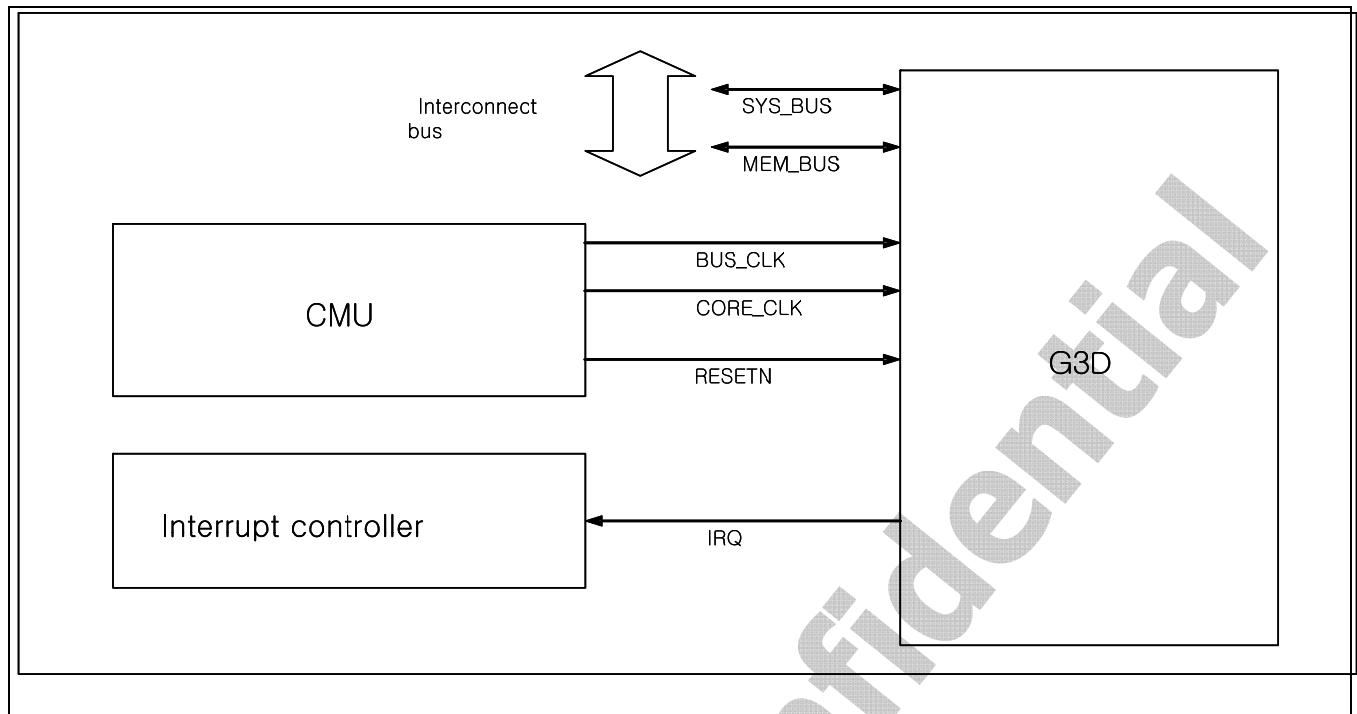


Figure 5-2 Block Diagram of Integration Information with Related Block

The Clock Management Unit (CMU) comprises of two blocks: BUS_CLK (for bus interface part) and CORE_CLK (for G3D internal function part). Both clocks are supported in 200Mhz clock domain, but you can set the two clocks with different clocks. BUS_CLK and CORE_CLK can be fully asynchronous.

G3D block has six controllable clock domains partitioned internally to functional areas of the design. It automatically controls clock gating if some blocks are not used at that time.

G3D block has its own power domain. If you do not use G3D block, then you can turn off the G3D block thoroughly by setting PMU. The detail power states are summarized in [Table 5-2](#). You can see the detailed explanation of power mode in Chapter, “PMU”.

Table 5-2 Power Mode Summary About G3D

Power Mode	NORMAL	IDLE	DEEP-IDLE	STOP	DEEP-STOP	SLEEP
G3D	Power on/ Clock gating/ Power gating	KEEP power state in NORMAL mode(NOTE)	KEEP power state in NORMAL mode/ Power gating	Clock gating/ Power gating	Power gating	Power off

NOTE: KEEP power state in NORMAL mode means power-on G3D in NORMAL mode is still power-on, clock-gated G3D in NORMAL mode is still clock-gated, and power-gated G3D in NORMAL mode is still power-gated.

5.1.7 REGISTER MAP

G3D contains the same control registers, as shown in [Table 5-3](#). Contact us for more information about registers, descriptions, operating systems support, and 3D libraries.

Refer to <http://www.khronos.org> about Open APIs like OGL-ES and OpenVG.

Table 5-3 G3D Register Summary

Module Name	Start Address	End Address	Size
G3D	0xF300_0000	0xF3FF_FFFF	16M bytes

6 MULTI FORMAT CODEC

6.1 INTRODUCTION

The multi-format video codec is a synthesizable core which can perform encoding and decoding of multiple streams at 30fps up to 1080p resolution image (1920x1080).

The MFC can handle a real-time coding up to 16 multi-channels which supports H.264, MPEG4, MPEG2, VC-1 and H.263. Detailed features are described in later sections. Note that MFC can support up to 16 channels, but the actual number of channels is dependent on use cases and limited by many system specific factors, such as the system memory, the clock speed provided to MFC, and so on.

6.1.1 SUPPORTED STANDARDS

- ITU-T H.264, ISO/IEC 14496-10
 - Decoding : High Profile Level 4.0 1920x1080 @ 30fps 20Mbps
 - o Baseline Profile Level 4.0
 - Except FMO (Flexible Macroblock Ordering), ASO (Arbitrary Slice Ordering) and RS (Redundant Slice)
 - o Main Profile Level 4.0
 - o High Profile Level 4.0
 - Encoding : High Profile Level 4.0 1920x1080 @ 30fps 20Mbps
 - o Baseline / Main / High Profile
 - o - Except FMO (Flexible Macroblock Ordering), ASO (Arbitrary Slice Ordering) and RS (Redundant Slice)
 - o Support 8x8 transform in high profile
 - o Support cyclic intra macroblock refresh
- ITU-T H.263 Profile 3
 - Decoding : Profile 3 Level 70 D1 @30fps 8Mbps
 - o Profile 3, restricted up to SD resolution 30fps
 - o Support H.263 Annexes
 - Annex I: Advanced Intra Coding
 - Annex J: De-blocking (in-loop) filter
 - Annex K: Slice Structured Mode without RS & ASO
 - Annex T: Modified Quantization
 - Annex D: Unrestricted Motion Vector Mode
 - Annex F: Advanced Prediction Mode except overlapped motion compensation for luminance
 - Encoding : Base Profile @ 30fps 8Mbps
 - o Baseline Profile
 - o Support custom size up to 1920x1088

- ISO/IEC 14496-2 MPEG4
 - Decoding : Advanced Simple Profile Level 5 D1 @ 30fps 8Mbps
 - o MPEG4 Simple Profile
 - o MPEG4 Advanced Simple Profile Level 5
 - o Xvid
 - o De-blocking filter for post-processing
 - Error resilience tools: re-sync marker, data-partitioning with reversible VLC
 - o Data-partitioning supports up to SD resolution
 - o Support quarter pixel motion compensation
 - o GMC (Global Motion Compensation) is restricted to 1 warp point
 - 2 and 3-warping point supported with quality degradation, i.e. continuing decoding with corrupted image
 - o Support only one rectangular visual object
 - o Only forward reversible VLC (RVLC) is supported
 - o Support error resilience tool
 - o Support post-processing by re-using H.263 in-loop filter.
 - Encoding : Advanced Simple Profile Level 5 D1 @ 30fps 8Mbps
 - o Support MPEG4 Simple / Advanced Simple Profile
 - Except data partitioning, RVLC
 - o Support only one rectangular visual object
 - o Support only DC prediction
- ISO/IEC 13818-2 MPEG2
 - Decoding : Main Profile High Level 1920x1080 @ 30fps 40Mbps
 - o Support Main Profile High Level
 - o Support MPEG1 except D-picture
- SMPTE 421M VC-1
 - Decoding : Advanced Profile Level 3 1920x1080 @ 30fps 45Mbps
 - o Support Simple Profile Medium Level
 - o Support Main Profile High Level
 - o Support Advanced Profile Level 3
 - o Multi-resolution is not processed inside video decoder
 - o Provide Range Mapping information for post-processing

6.1.2 FEATURES

- Image features
 - Max size: 1920x1088 @ progressive mode
1920x544 @ interlaced mode
 - Min size : 32x32 @ progressive mode
32x16 @ interlaced mode
 - Input image size for encoding
 - Arbitrary input image size for encoding (no constraints such as multiple of 8 or 16)
 - Chrominance interleaved in external memory
 - 4:2:0 for encoding
 - 4:2:0 for decoding
 - Support monochrome for H.264 decoding
 - 8 bits per sample
 - Non paired field mode is not supported
- Slice
 - Minimum size: 16x16
- Picture coding structure at encoding
 - Progressive mode
 - Field mode (only H.264)
- Inter prediction at encoding
 - Number of reference frames : Max 2 (P frame: 1 or 2, B frame: 2)
 - Search range: Horizontal +/- 64, Vertical +/- 32
 - Motion estimation resolution: 1/4 pel for H264, 1/2 pel for MPEG4
 - Number of B frames: 1 or 2
 - Supported modes
 - H264: 16x16, 16x8, 8x16, 8x8, spatial direct mode
 - MPEG4: 4MV & UMV
- Intra macroblock for encoding
 - Support cyclic intra macroblock refresh
 - Intra prediction: Support 4x4 (9 modes), 16x16 (4 modes), 8x8 (9 modes) at H.264
- Rate control at encoding
 - CBR (Constant Bit-Rate) and VBR (Variable Bit-Rate)
 - Frame level (H.264/MPEG4/H.263) and macroblock level (H.264) rate control can be enabled or disabled selectively
- Stream
 - Time-multiplexed multi-stream encoding/decoding up to 16 channels
 - Start code must be included at every position of frame or slice in the stream at decoding

6.1.3 TARGET PERFORMANCE AND FUNCTIONS

6.1.3.1 Video Decoding Capability

- Decoding up to 1080p@30fps at 200MHz core clock frequency and 200MHz bus clock frequency
- The maximum resolution can be limited by the video standards, even though MFC has a capability to handle up to 1080p.

6.1.3.2 Video Encoding Capability

- Encoding up to 1080p@30fps at 200MHz core clock frequency and 200MHz bus clock frequency
- The performance numbers are verified under the condition that the number of reference frames for a P frame is 1.
- The maximum resolution can be limited by the video standards, even though MFC has a capability to handle up to 1080p.

6.1.3.3 Error Detection

- H264
 - Header error detection
 - Checking if firmware parsing is out of range
 - H/W
 - VLD error detection: When there is no stream to decode before stream decoder is complete.
 - Macroblock error detection: When MB type is out of range (i.e., 0~25 @ I slice / 0~30 @ P slice / 0~48 @ B slice)
 - Sub MB type error detection: When sub_MB_type or intra prediction mode is out of range (i.e., Intra prediction mode 0~7, sub_mb_type 0~2 @ P slice / 0~11 @ B slice)
 - IDF error detection: When ref_idx value is greater than num_ref_idx_lx_active_minus1 (PPS)
 - Delta Q error detection: When mb_qp_delta is out of range (i.e., -26~25)
 - Timeout interrupt: When decoding time is greater than the firmware setting
- MPEG2/ MPEG4/ H.263/ VC-1
 - Header error detection: When the result of firmware parsing is out of range
 - H/W
 - VLD error: When VLD Result is out of table.
 - Coeff error: When AC/DC coefficient of 1 block is more than 64
 - Time out interrupt: When decoding time is greater than firmware setting

6.2 HARDWARE OVERVIEW

6.2.1 BLOCK DIAGRAM

Top level of the MFC in [Figure 6-1](#) contains the hardware modules, including an OpenRISC with 8KB I-cache and 4KB D-cache. The optimum partition of the codec functions into software and hardware has ensured that the small sized hardware supports multiple standards. The hardware operates encoding and decoding at the slice level. On the other hand, the firmware on RISC performs other processing, such as slice header parsing and/or generation. Settings by the host processor can be changed at the frame boundary through the host interface.

[Figure 6-1](#) presents a block diagram of MFC which is composed of RISC, MFC core, RG, bus interface, host interface, and stream interface. MFC core includes many codec accelerators. RG stands for register group which can be accessed by RISC and HOST. Host and RISC can communicate through registers in RG and risc2host interrupt generated by register in RG. If RISC gets some interrupt or information from HW, RISC set the registers to let host know the status of MFC. Host clears the interrupt signal by resetting the MFC_RISC_HOST_INT register.

There are two AXI master interfaces in which both Port_A and Port_B are used for full performance. As per the AXI standard, MFC masters take care of read/write hazard issues. Before read access of written data by MFC, internal masters in MFC always check the response of write access.

The search SRAM in the diagram contains reference image for motion estimation and motion compensation.

The shared SRAM is for sharing current image for encoding.

To reduce bandwidth for reference image loading, there is a pixel cache in the MFC core. The size is 2KB for luma and 1KB for chroma. In encoding case, only chroma reference will be loaded by pixel cache, because luma reference is already loaded in search SRAM for motion estimation. To use the pixel cache in decoding, reconstruction image have to be placed in Port_A memory area.

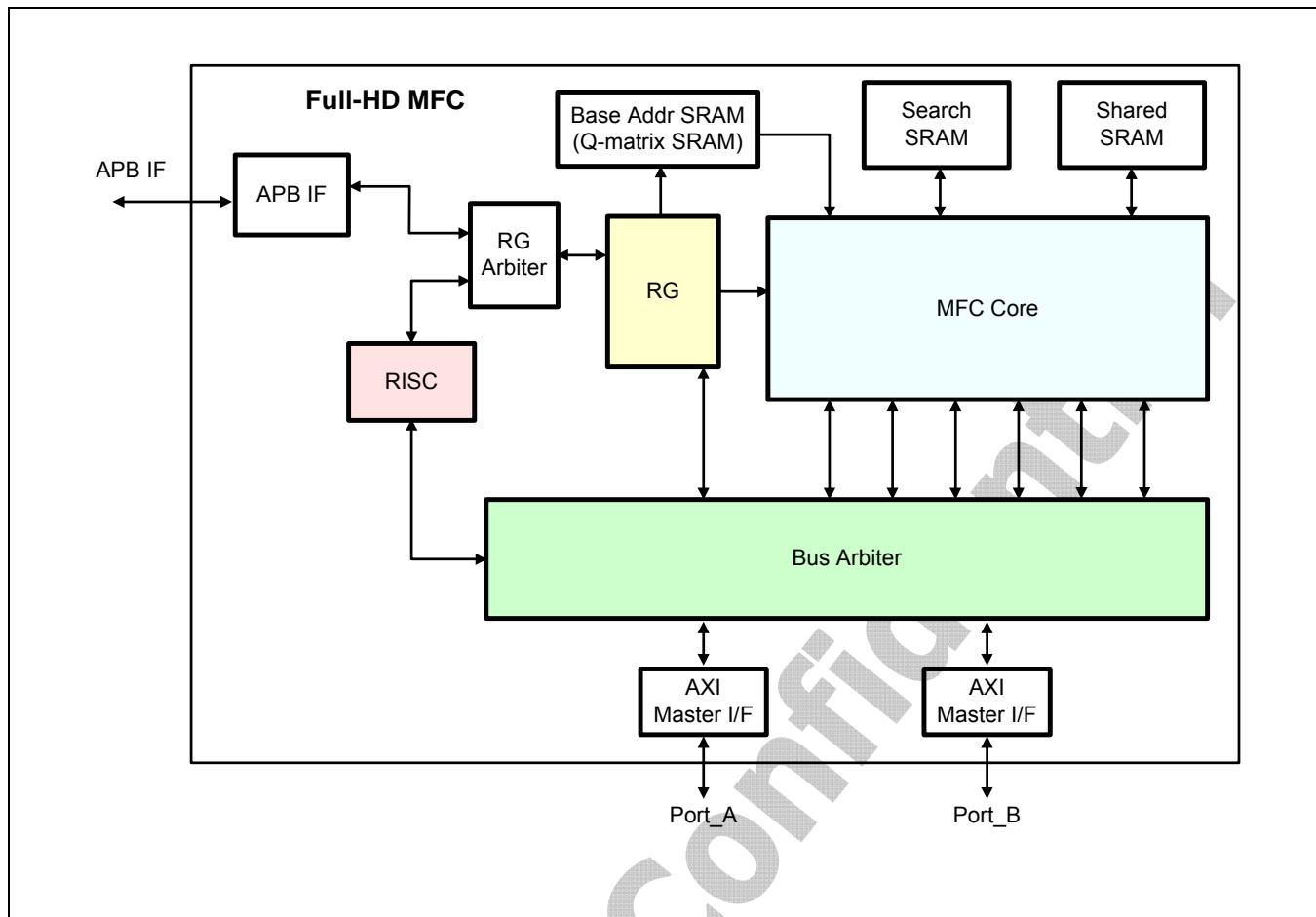


Figure 6-1 MFC Block Diagram

There are several internal masters in MFC core. Their interfaces are controlled by the bus arbiter. Internal masters have an index[6:0] register controlled by firmware. This register may be used to generate address, which will be an output of AXI interface.

After getting an index, the bus arbiter makes a decision which port to be used based on Index[6] (Index[6] = 0 for Port_A and 1 for Port_B). With Index[5:0], the bus arbiter get a base address from Q-matrix SRAM. The real address is calculated with the base address and DRAM_BASE_ADDR in AXI_MASTER. Therefore host must set a base address before starting codec.

6.2.2 FRAME MEMORY

A frame memory area is specified by base address, horizontal and vertical image size. A complete image consists of Y, Cb and Cr components. The Cb and Cr pixels are stored in a byte interleaved way. Therefore, an image needs 2 frame buffers, one for Y and another for Cb and Cr components, as shown in [Figure 6-2](#). The sum of image horizontal size and image horizontal offset should be multiple of 16: Horizontal offset makes the value of horizontal size to be multiple of 16. The vertical image size for Cb/Cr frame buffer should be half of the Y frame buffer.

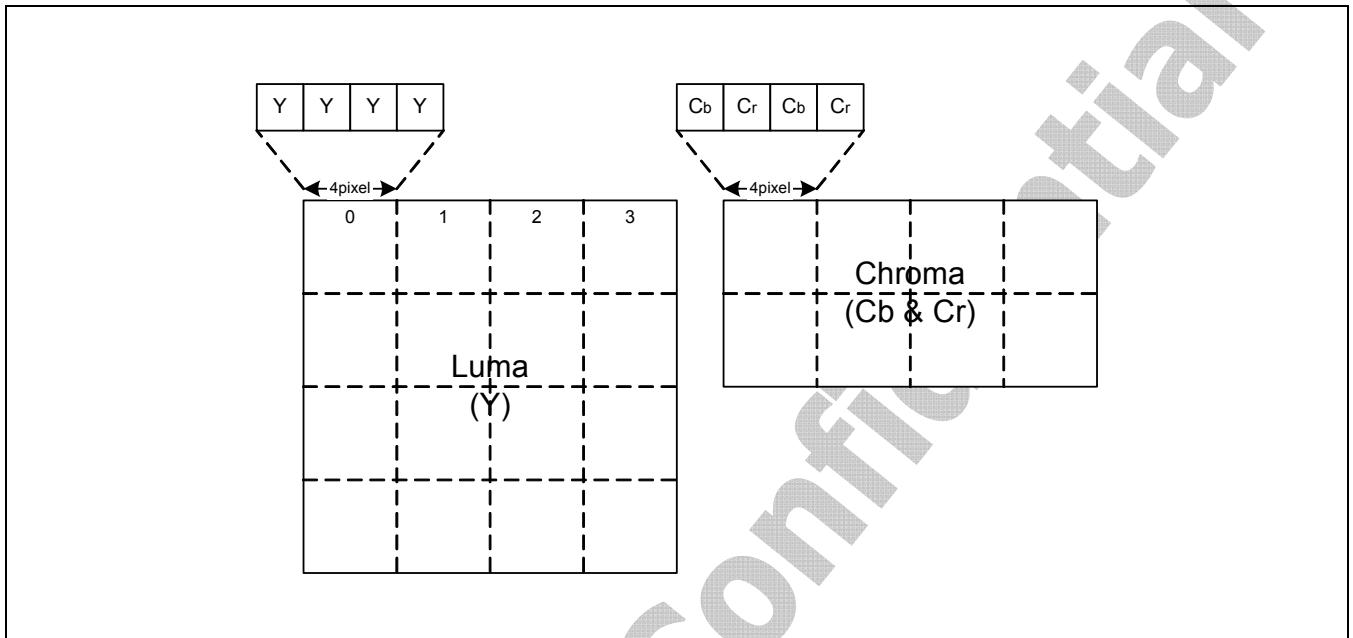


Figure 6-2 Luma and Chroma Pixel (8 bytes-aligned)

Reference picture is always made in the tile mode memory structure. Decoding reconstruction image is made in 64 pixels x 32 lines tiled mode. Encoding reconstruction image is made in 16 pixels x 16 lines tiled mode.

Current picture for encoding can be stored in two ways, linear memory structure or tile mode memory structure, depending on ENC_MAP_FOR_CUR register (0xF170_C51C). Host sets external memory parameters with the memory structure configuration. The physical memory address of each pixel data is determined by the memory structure, base address, and coordinates of pixel in the frame.

[Figure 6-3](#) and [Figure 6-4](#) shows the tile mode memory structure of a QCIF image for 16x16 and 64x32 configuration.

x_addr	0	4	8	12	16	20	24	28	32	36	40	44	48	52	56
y_addr	Logical mapping table (MB order)														
0	0	1	2	3	4	5	6	7	8	9	10				
16	11	12	13	14	15	16	17	18	19	20	21				
32	22	23	24	25	26	27	28	29	30	31	32				
48	33	34	35	36	37	38	39	40	41	42	43				
64	44	45	46	47	48	49	50	51	52	53	54				
80	55	56	57	58	59	60	61	62	63	64	65				
96	66	67	68	69	70	71	72	73	74	75	76				
112	77	78	79	80	81	82	83	84	85	86	87				
128	88	89	90	91	92	93	94	95	96	97	98				

** x_addr: word unit. Y_addr: line unit

Bank0	Bank1	Bank2	Bank3

```

pixel_x = horizontal_image_size (ex: 1280)
pixel_y = vertical_image_size (ex: 720)
pixel_x_minus = pixel_x - 1;
pixel_y_minus = pixel_y - 1;
roundup_x = ((pixel_x - 1)/16)*8 + 1;
roundup_y = ((pixel_y - 1)/16)*4 + 1;

if (((pixel_y - 32) <= y_addr) && (y_addr < pixel_y) &&
    (pixel_y_minus[5] == 0) && (y_addr[5] == 0)) {
    row_add = y_addr[13:6] * roundup_x + x_addr[20:6]
    bank_add = y_addr[4] ? {~x_addr[3], x_addr[2]} : {x_addr[3], x_addr[2]}
    col_add = {x_addr[5], x_addr[4:3], y_addr[3:0], x_addr[1:0]}
}
else {
    row_add = y_addr[13:6] * roundup_x + x_addr[20:5]
    bank_add = y_addr[4] ? {~x_addr[3], x_addr[2]} : {x_addr[3], x_addr[2]}
    col_add = {y_addr[5], x_addr[4:3], y_addr[3:0], x_addr[1:0]}
}

if (pixel_y_minus[5] == 0)
    pic_range = pixel_y_minus[14:6] * roundup_x + pixel_x_minus[14:8] + 1
else
    pic_range = roundup_x * roundup_y

```

Figure 6-3 QCIF Image in 16pixel x 16lines (1x1) Tiled Mode

x_addr	0	4	8	12	16	20	24	28	32	36	40	44	48
y_addr	Logical mapping table (MB order)												
0	0	1	2	3	4	5	6	7	8	9	10		
16	11	12	13	14	15	16	17	18	19	20	21		
32	22	23	24	25	26	27	28	29	30	31	32		
48	33	34	35	36	37	38	39	40	41	42	43		
64	44	45	46	47	48	49	50	51	52	53	54		
80	55	56	57	58	59	60	61	62	63	64	65		
96	66	67	68	69	70	71	72	73	74	75	76		
112	77	78	79	80	81	82	83	84	85	86	87		
128	88	89	90	91	92	93	94	95	96	97	98		
144													

Bank0	Bank1	Bank2	Bank3

** X_addr: word unit, Y_addr: line unit

```

pixel_x = horizontal_image_size (ex: 1280)
pixel_y = vertical_image_size (ex: 720)
pixel_x_minus = pixel_x - 1;
pixel_y_minus = pixel_y - 1;
roundup_x = ((pixel_x - 1)/16)/8 + 1;
roundup_y = ((pixel_y - 1)/16)/4 + 1;

if (((pixel_y - 32) <= y_addr) && (y_addr < pixel_y) &&
    (pixel_y_minus[5] == 0) && (y_addr[5] == 0)) {
    row_addr = y_addr[11:6] ^ roundup_x + x_addr[20:6]
    bank_addr = {x_addr[5], x_addr[4]}
    col_addr = {y_addr[4:0], x_addr[3:0]}
}
else {
    row_addr = y_addr[13:6] ^ roundup_x + x_addr[20:5]
    bank_addr = x_addr[5] ? {-y_addr[5], x_addr[4]} : {y_addr[5], x_addr[4]}
    col_addr = {y_addr[4:0], x_addr[3:0]}
}

if (pixel_y_minus[5] == 0)
    pic_range = pixel_y_minus[14:6] ^ roundup_x + pixel_x_minus[14:8] + 1
else
    pic_range = roundup_x ^ roundup_y

```

Figure 6-4 QCIF Image in 64pixel x 32lines (4x2) Tiled Mode

6.3 REGISTER DESCRIPTION

6.3.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
Control Registers				
MFC_SW_RESET	0xF170_0000	R/W	Soft reset for each module in MFC. Each bit has following interpretation. 0 = Reset 1 = Release	0x0000003fe
MFC_RISC_HOST_INT	0xF170_0008	R/W	MFC to host interrupt register. An interrupt is raised when MFC enables the INTERRUPT bit. Host CPU needs to check this register, properly process an ISR (Interrupt Service Routine), and clear the INTERRUPT bit.	0x00000000
MFC_HOST2RISC_COMMAND	0xF170_0030	R/W	Host to MFC command register. Host can send command to MFC to open instance or close instance.	0x00000000
MFC_HOST2RISC_ARG1	0xF170_0034	R/W	The first argument of the host command	0x00000000
MFC_HOST2RISC_ARG2	0xF170_0038	R/W	The second argument of the host command	0x00000000
MFC_HOST2RISC_ARG3	0xF170_003C	R/W	Context memory address	0x00000000
MFC_HOST2RISC_ARG4	0xF170_0040	R/W	Context memory size	0x00000000
MFC_RISC2HOST_COMMAND	0xF170_0044	R/W	MFC to host command register. MFC can respond to host using the MFC_RISC2HOST_COMMAND register	0x00000000
MFC_RISC2HOST_ARG1	0xF170_0048	R/W	MFC to host argument register. This register is used with the MFC_RISC2HOST_COMMAND register	0x00000000
MFC_RISC2HOST_ARG2	0xF170_004C	R/W	The second argument of the host command	0x00000000
MFC_RISC2HOST_ARG3	0xF170_0050	R/W	The Third argument of the host command	0x00000000
MFC_RISC2HOST_ARG4	0xF170_0054	R/W	The fourth argument of the host command	0x00000000
MFC_FIRMWARE_VERSION	0xF170_0058	R	Firmware version information register	0x00000000
DBG_INFO_OUTPUT1	0xF170_0064	R	Debug information output register 1	0x00000000
DBG_INFO_OUTPUT2	0xF170_0068	R	Debug information output register 2	0x00000000
MFC_FIRMWARE_STATUS	0xF170_0080	R	Firmware status register	0x00000000
MFC_MC_DRAMBASE_ADDR_A	0xF170_0508	R/W	DRAM base address which indicates the base address for the memory map of the port A	0xD3000000
MFC_MC_DRAMBASE_ADDR_B	0xF170_050C	R/W	DRAM base address which indicates the base address for the memory map of the port B	0x23000000

Register	Address	R/W	Description	Reset Value
MFC_MC_STATUS	0xF170_0510	R	Bus arbiter's status. This register can be used to check whether the bus is busy or not before resetting MFC.	0x0000000X
MFC_COMMON_BASE_ADDR_0~63 [16:0]	0xF170_0600~0xF170_06FC	R/W	Codec common memory region for start address.	X
MFC_COMMON_BASE_ADDR_64~127 [16:0]	0xF170_0700~0xF170_07FC	R/W	Codec common memory region for start address.	X
Codec Registers				
MFC_HSIZE_PX	0xF170_0818	R/W	Picture width at encoder	0x00000000
MFC_VSIZE_PX	0xF170_081C	R/W	Picture height at encoder In the interlaced-field coding mode, it specifies the coded height of a field. In the progressive or interlaced-frame coding mode, it is the coded height of a frame at encoder.	0x00000000
MFC_PROFILE	0xF170_0830	R/W	Profile and level control register at encoder	0x00000000
MFC_PICTURE_STRUCT	0xF170_083C	R/W	Field picture/frame picture flag register at encoder	0x00000000
MFC_LF_CONTROL	0xF170_0848	R/W	Loop filter control	0x00000000
MFC_LF_ALPHA_OFF	0xF170_084C	R/W	Loop filter alpha offset	0x00000000
MFC_LF_BETA_OFF	0xF170_0850	R/W	Loop filter beta offset	0x00000000
MFC_QP_OFFSET	0xF170_0C30	R/W	QP information offset from the DPB start address	0x00000000
MFC_QP_OUT_EN	0xF170_0C34	R/W	QP information enable at decoder	0x00000000
MFC_SI_RTN_CHID	0xF170_2000	R/W	Return channel instance ID register	0x00000000
MFC_COMMON_SI_RG_1 ~ 15	0xF170_2004~0xF170_203C	R/W	Return the status of MFC after processing	0x00000000
MFC_SI_CH0_INST_ID	0xF170_2040	R/W	CH0 instance ID and control register	0x00000000
MFC_SI_CH1_INST_ID	0xF170_2080	R/W	CH1 instance ID and control register	0x00000000
MFC_COMMON_CH0_RG_1 ~ 15	0xF170_2044~0xF170_207C	R/W	Host and MFC interface registers through CH0	0x00000000
MFC_COMMON_CH1_RG_1 ~ 15	0xF170_2084~0xF170_20BC	R/W	Host and MFC interface registers through CH1	0x00000000
MFC_COMMON_SI_RG_1	0xF170_2004	R	Vertical resolution register	0x00000000
MFC_COMMON_SI_RG_2	0xF170_2008	R	Horizontal resolution register	0x00000000
MFC_COMMON_SI_RG_3	0xF170_200C	R	Required buffer number register	0x00000000
MFC_COMMON_SI_RG_4	0xF170_2010	R	Luminance address register for display	0x00000000

Register	Address	R/W	Description	Reset Value
MFC_COMMON_SI_RG_5	0xF170_2014	R	Chrominance address register for display	0x00000000
MFC_COMMON_SI_RG_6	0xF170_2018	R	Decoded frame size for a frame	0x00000000
MFC_COMMON_SI_RG_7	0xF170_201C	R	Display status register	0x00000000
MFC_COMMON_SI_RG_8	0xF170_2020	R	Frame type register	0x00000000
MFC_COMMON_SI_RG_9	0xF170_2024	R	Luminance address register in decoding order	0x00000000
MFC_COMMON_SI_RG_10	0xF170_2028	R	Chrominance address setting register in decoding order	0x00000000
MFC_COMMON_SI_RG_11	0xF170_202C	R	Decoding status register	0x00000000
MFC_COMMON_CHx_R_G_1	0xF170_2044 or 0xF170_2084	R/W	Start address of the CPB (coded picture buffer) in the external stream buffer.	0x00000000
MFC_COMMON_CHx_R_G_2	0xF170_2048 or 0xF170_2088	R/W	Decoding unit size register	0x00000000
MFC_COMMON_CHx_R_G_3	0xF170_204C or 0xF170_208C	R/W	Channel descriptor buffer address	0x00000000
Reserved	0xF170_2050 or 0xF170_2090		Reserved	0x00000000
Reserved	0xF170_2054 or 0xF170_2094		Reserved	0x00000000
MFC_COMMON_CHx_R_G_6	0xF170_2058 or 0xF170_2098	R/W	CPB size register	0x00000000
MFC_COMMON_CHx_R_G_7	0xF170_205C or 0xF170_209C	R/W	Descriptor buffer size register	0x00000000
MFC_COMMON_CHx_R_G_8	0xF170_2060 or 0xF170_20A0	R/W	Release buffer register to specify the individual DPB availability	0x00000000
MFC_COMMON_CHx_R_G_9	0xF170_2064 or 0xF170_20A4	R/W	Shared memory address	0x00000000
MFC_COMMON_CHx_R_G_10	0xF170_2068 or 0xF170_20A8	R/W	DPB configuration that host prepared for decoding	0x00000000

Register	Address	R/W	Description	Reset Value
MFC_COMMON_CHx_R_G_11	0xF170_206C or 0xF170_20AC	R/W	Command sequence number from the host	0x00000000
MFC_COMMON_SI_RG_1	0xF170_2004	R	Encoded stream size register	0x00000000
MFC_COMMON_SI_RG_2	0xF170_2008	R	Encoded picture count register	0x00000000
MFC_COMMON_SI_RG_3	0xF170_200C	R	Stream buffer write pointer	0x00000000
MFC_COMMON_SI_RG_4	0xF170_2010	R/W	Slice type of the current frame to be encoded	0x00000000
MFC_COMMON_SI_RG_5	0xF170_2014	R/W	Encoded luma address	0x00000000
MFC_COMMON_SI_RG_6	0xF170_2018	R/W	Encoded chroma address	0x00000000
MFC_COMMON_CHx_R_G_1	0xF170_2044 or 0xF170_2084	R/W	Stream buffer start address at encoder.	0x00000000
MFC_COMMON_CHx_R_G_3	0xF170_204C or 0xF170_208C	R/W	Stream buffer size register	0x00000000
MFC_COMMON_CHx_R_G_4	0xF170_2050 or 0xF170_2090	R/W	Current luma address	0x00000000
MFC_COMMON_CHx_R_G_5	0xF170_2054 or 0xF170_2094	R/W	Current chroma address	0x00000000
MFC_COMMON_CHx_R_G_6	0xF170_2058 or 0xF170_2098	R/W	Frame insertion control register	0x00000000
MFC_COMMON_CHx_R_G_9	0xF170_2064 or 0xF170_20A4	R/W	Shared memory address	0x00000000
MFC_COMMON_CHx_R_G_10	0xF170_2068 or 0xF170_20A8	R/W	Flushing input buffer	0x00000000
MFC_COMMON_CHx_R_G_11	0xF170_206C or 0xF170_20AC	R/W	Command sequence number from the host	0x00000000
Encoding Registers				
ENC_PIC_TYPE_CTRL	0xF170_C504	R/W	Picture type control register	0x00000000
ENC_B_RECON_WRITE_ON	0xF170_C508	R/W	B-frame reconstructed data write control register	0x00000000
ENC_MSLICE_CTRL	0xF170_C50C	R/W	Multi-slice control register	0x00000000

Register	Address	R/W	Description	Reset Value
ENC_MSLICE_MB	0xF170_C510	R/W	Slice size register when multi slice is enabled. Fixed number of macroblocks is used to determine the size of one slice.	0x00000000
ENC_MSLICE_Bit	0xF170_C514	R/W	Slice size register when multi slice is enabled. Bit count is used to determine the size of one slice.	0x00000000
ENC_CIR_CTRL	0xF170_C518	R/W	Intra refresh macroblock setting register	0x00000000
ENC_MAP_FOR_CUR	0xF170_C51C	R/W	Memory structure setting register of the current frame.	0x00000000
ENC_PADDING_CTRL	0xF170_C520	R/W	Padding control register	0x00000000
ENC_COMMON_INTRA_BIAS	0xF170_C588	R/W	Intra mode bias register for the macroblock mode decision	0x00000000
ENC_COMMON_BI_DIRECT_BIAS	0xF170_C58C	R/W	Bi-directional mode bias register for the macroblock mode decision	0x00000000
RC_CONFIG	0xF170_C5A0	R/W	Configuration of the rate control	0x00000000
RC_FRAME_RATE	0xF170_D0D0	R/W	Frame rate for the frame level RC	0x00000000
RC_BIT_RATE	0xF170_C5A8	R/W	Target bit rate for the frame level RC	0x00000000
RC_QBOUND	0xF170_C5AC	R/W	Maximum and minimum value of the quantization parameter	0x00000000
RC_RPARA	0xF170_C5B0	R/W	Rate control reaction coefficient	0x00000000
RC_MB_CTRL	0xF170_C5B4	R/W	Control the macroblock adaptive scaling features	0x00000000
H264_ENC_ENTRP_MODE	0xF170_D004	R/W	Entropy coding mode.	0x00000000
H264_ENC_NUM_OF_REF	0xF170_D010	R/W	The maximum number of reference pictures.	0x00000000
H264_ENC_TRANS_8X8_FLAG	0xF170_D034	R/W	8x8 transform enable flag in PPS at high profile.	0x00000000
MPEG4_ENC_QUART_PXL	0xF170_E008	R/W	Quarter pel interpolation control register	0x00000000
Shared Memory Structure				
EXTENDED_DECODE_STATUS	0x0000	R	Extended decode status	Undef
SET_FRAME_TAG	0x0004	W	Setting frame tag of an output frame	Undef
GET_FRAME_TAG_TOP	0x0008	R	Getting the first frame tag of an output frame	Undef
GET_FRAME_TAG_BOTTOM	0x000C	R	Getting the second frame tag of an output frame	Undef
PIC_TIME_TOP	0x0010	R	Presentation time of an output frame or top field	Undef
PIC_TIME_BOTTOM	0x0014	R	Presentation time of the bottom field	Undef

Register	Address	R/W	Description	Reset Value
START_BYTE_NUM	0x0018	R/W	An offset of the start position in the stream when the start position is not aligned	Undef
CROP_INFO1	0x0020	R	Frame cropping information. Valid for H.264 only.	Undef
CROP_INFO2	0x0024	R	Frame cropping information. Valid for H.264 only.	Undef
EXT_ENC_CONTROL	0x0028	W	Encoder control	Undef
ENC_PARAM_CHANGE	0x002C	W	Encoding parameter change that signals the change of bitrate, frame rate, or the GOP size	Undef
VOP_TIMING	0x0030	W	VOP timing	Undef
HEC_PERIOD	0x0034	W	The number of consecutive video packet between header extension codes	Undef
METADATA_ENABLE	0x0038	W	Enable storing the metadata information to the shared memory	Undef
METADATA_STATUS	0x003C	R	Getting the presence of the metadata	Undef
METADATA_DISPLAY_INDEX	0x0040	R	DPB number when concealed macroblock or QP is enabled	Undef
EXT_METADATA_START_ADDR	0x0044	W	The start address of the metadata memory	Undef
PUT_EXTRADATA	0x0048	W	Signaling the existence of extra metadata.	Undef
EXTRADATA_ADDR	0x004C	W	The address of extra metadata.	Undef
ALLOCATED_LUMA_DPB_SIZE	0x0064	W	Size of luma DPB that host allocated for decoding	Undef
ALLOCATED_CHROMA_DPB_SIZE	0x0068	W	Size of chroma DPB that host allocated for decoding	Undef
ALLOCATED_MV_SIZE	0x006C	W	Size of motion vector buffers that host allocated for decoding	Undef
P_B_FRAME_QP	0x0070	W	P frame QP and B frame QP	Undef
ASPECT_RATIO_IDC	0x0074	W	VUI aspect ratio IDC for H.264 encoding	Undef
EXTENDED_SAR	0x0078	W	Extended sample aspect ratio for H.264 VUI encoding	Undef
DISP_PIC_PROFILE	0x007C	R	Profile info for displayed picture	Undef
FLUSH_CMD_TYPE	0x0080	R	Type of a flushed command	Undef
FLUSH_CMD_INBUF1	0x0084	R	Input buffer pointer of a flushed command	Undef
FLUSH_CMD_INBUF2	0x0088	R	Input buffer pointer of a flushed command	Undef
FLUSH_CMD_OUTBUF	0x008C	R	Output buffer pointer of a flushed command	Undef
NEW_RC_BIT_RATE	0x0090	W	Updated target bit rate	Undef
NEW_RC_FRAME_RATE	0x0094	W	Updated target frame rate	Undef
NEW_I_PERIOD	0x0098	W	Updated intra period	Undef

Register	Address	R/W	Description	Reset Value
H264_I_PERIOD	0x009C	W	Intra picture period to generate open GOP. Valid for H.264 encoder only	Undef
RC_CONTROL_CONFIG	0x00A0	W	Rate control configuration register for fixed target bit control	Undef
BATCH_INPUT_ADDR	0x00A4	W	Start address of the input structure for batch encoding	Undef
BATCH_OUTPUT_ADDR	0x00A8	W	Start address of the output structure for batch encoding	Undef
BATCH_OUTPUT_SIZE	0x00AC	W	Size of the output structure for batch encoding	Undef
MIN_LUMA_DPB_SIZE	0x00B0	R	Minimum size of luma DPB that host needs to allocate for decoding	Undef
H264_POC_TYPE	0x00B8	R	POC_TYPE of H.264 decoder.	Undef
MIN_CHROMA_DPB_SIZE	0x00BC	R	Minimum size of chroma DPB that host needs to allocate for decoding	Undef
DISP_PIC_FRAME_TYPE	0x00C0	R	Frame type of a displayed picture	Undef
FREE_LUMA_DPB	0x00C4	R	Available luma DPB address	Undef
ASPECT_RATIO_INFO	0x00C8	R	Aspect ratio information for MPEG4 decoding	Undef
EXTENDED_PAR	0x00CC	R	Extended pixel aspect ratio information for MPEG4 decoding	Undef
DBG_HISTORY_INPUT0	0x00D0	W	Debug history input register 0 for stage counter settings	Undef
DBG_HISTORY_INPUT1	0x00D4	W	Debug history input register 1 for stage counter settings	Undef
DBG_HISTORY_OUTPUT	0x00D8	R	Output size register for stage counter history	Undef
HIERARCHICAL_P_QP	0x00E0	W	QP for hierarchical P frames	Undef

6.3.2 CONTROL REGISTERS

6.3.2.1 MFC Core Control Register

6.3.2.1.1 MFC Software Reset Register (*MFC_SW_RESET*, R/W, Address = 0xF170_0000)

MFC_SW_RESET	Bit	Description	Initial State
Reserved	[31:10]	Reserved	0
RSTN_RG_MPEG2	[9]	Soft reset for RG_MPEG2	1
RSTN_RG_MPEG4	[8]	Soft reset for RG_MPEG4	1
RSTN_RG_VC1	[7]	Soft reset for RG_VC1	1
RSTN_RG_H264	[6]	Soft reset for RG_H264	1
RSTN_RG_COMMON	[5]	Soft reset for RG_COMMON and RG_DECCOM	1
RSTN_DMX	[4]	Soft reset for DMX 0	1
RSTN_VI	[3]	Soft reset for VI	1
RSTN_MFCCORE	[2]	Soft reset for MFC core	1
RSTN_MC	[1]	Soft reset for MC	1
RSTN_RISC	[0]	Soft reset for RISC core	0

6.3.2.1.2 RISC to Host Interrupt Register (*MFC_RISC_HOST_INT*, R/W, Address = 0xF170_0008)

MFC_RISC_HOST_INT	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0
INTERRUPT	[0]	0 = Interrupt clear 1 = Interrupt is raised by MFC	0

6.3.2.1.3 HOST2RISC Command Register (*MFC_HOST2RISC_COMMAND*, R/W, Address = 0xF170_0030)

MFC_HOST2RISC_COMMAND	Bit	Description	Initial State
HOST2RISC_COMMAND	[31:0]	0 = No operation 1 = OPEN_CH (open instance) 2 = CLOSE_CH (close instance) 3 = SYS_INIT (system initialization) 4 = FLUSH_COMMAND (flush commands in ch0, ch1) 5 = SLEEP 6 = WAKEUP 7 = CONTINUE_ENC (continue encoding) 8 = ABORT_ENC (abort encoding)	0

6.3.2.1.4 HOST2RISC Argument Registers (MFC_HOST2RISC_ARG1, R/W, Address = 0xF170_0034)

MFC_HOST2RISC_ARG1	Bit	Description	Initial State
HOST2RISC_ARG1	[31:0]	<p>As per HOST2RISC_COMMAND, HOST2RISC_ARG has different meanings as follows.</p> <p><OPEN_CH></p> <p>A codec type should be specified as follows.</p> <ul style="list-style-type: none"> 0 = H.264 Decoding 1 = VC1 Advanced Profile Decoding 2 = MPEG4 / XVid Decoding 3 = MPEG1/MPEG2 Decoding 4 = H.263 Decoding 5 = VC1 Simple/Main Profile Decoding 6 = Reserved 7 = Reserved 8 = Reserved 9 = Reserved 16 = H.264 Encoding 17 = MPEG4 Encoding 18 = H.263 Encoding 19 = VC9 Decoding 20 = VC1 Decoding without start code <p><CLOSE_CH></p> <p>An instance ID to close should be specified</p> <p><SYS_INIT></p> <p>Size of the memory for the firmware should be specified (currently 400KB)</p>	0

6.3.2.1.5 HOST2RISC Argument Registers (MFC_HOST2RISC_ARG2, R/W, Address = 0xF170_0038)

MFC_HOST2RISC_ARG2	Bit	Description	Initial State
Reserved	[31]	Reserved	0
HOST2RISC_ARG2	[30:0]	<p>When HOST2RISC_COMMAND is OPEN_CH, it enables/disables pixel cache</p> <p><Encoder></p> <ul style="list-style-type: none"> 0 = Enable pixel cache 3 = Disable pixel cache <p><Decoder></p> <ul style="list-style-type: none"> 0 = Enable pixel cache for P picture only 1 = Enable pixel cache for B picture only 2 = Enable pixel cache for both P and B pictures 3 = Disable pixel cache 	0

6.3.2.1.6 HOST2RISC Argument Registers (MFC_HOST2RISC_ARG3, R/W, Address = 0xF170_003C)

MFC_HOST2RISC_ARG3	Bit	Description	Initial State
CONTEXT_ADDR	[31:0]	Context memory address for an instance	0

6.3.2.1.7 HOST2RISC Argument Registers (MFC_HOST2RISC_ARG4, R/W, Address = 0xF170_0040)

MFC_HOST2RISC_ARG4	Bit	Description	Initial State
CONTEXT_SIZE	[31:0]	Context memory size for an instance. H.264 decoder requires 600KB and others require 10KB.	0

6.3.2.1.8 RISC2HOST Command Register (MFC_RISC2HOST_COMMAND, R/W, Address = 0xF170_0044)

MFC_RISC2HOST_COMMAND	Bit	Description	Initial State
RISC2HOST_COMMAND	[31:0]	0 = RISC2HOST_CMD_EMPTY 1 = RISC2HOST_CMD_OPEN_CH_RET 2 = RISC2HOST_CMD_CLOSE_CH_RET 3 = Reserved 4 = RISC2HOST_CMD_SEQ_DONE_RET 5 = RISC2HOST_CMD_FRAME_DONE_RET 6 = RISC2HOST_CMD_SLICE_DONE_RET 7 = RISC2HOST_CMD_ENC_COMPLETE_RET 8 = RISC2HOST_CMD_SYS_INIT_RET 9 = RISC2HOST_CMD_FIRMWARE_STATUS_RET 10 = RISC2HOST_CMD_SLEEP_RET 11 = RISC2HOST_CMD_WAKEUP_RET 12 = RISC2HOST_CMD_FLUSH_COMMAND_RET 13 = RISC2HOST_CMD_ABORT_RET 14 = RISC2HOST_CMD_BATCH_ENC_RET 15 = RISC2HOST_CMD_INIT_BUFFERS_RET 16 = RISC2HOST_CMD_EDFU_INT_RET 17~31 = Reserved 32 = RISC2HOST_CMD_ERROR_RET	0

6.3.2.1.9 RISC2HOST Argument Registers (MFC_RISC2HOST_ARG1, R/W, Address = 0xF170_0048)

MFC_RISC2HOST_ARG1	Bit	Description	Initial State
MFC_RISC2HOST_ARG1	[31:0]	<OPEN> An instance ID will be returned <SYS_INIT> Firmware memory size will be returned(currently 400KB) <SEQ_START, FRAME_START, LAST_SEQ, INIT_BUFFERS, FRAME_START_REALLOC> A channel ID will be returned <FLUSH_COMMAND> [31:16]: Instance ID of CH1 [15:0]: Instance ID of CH0	0

NOTE: When host receives FLUSH_COMMAND_RET, it should check the shared memory at 0x80, 0x8C to figure out the input and output pointers in each command channel. If [31:16] is not 0xFFFF, a command in CH1 has been flushed. If [15:0] is not 0xFFFF, a command in CH0 has been flushed.

6.3.2.1.10 RISC2HOST Argument Registers (MFC_RISC2HOST_ARG2, R/W, Address = 0xF170_004C)

MFC_RISC2HOST_ARG2	Bit	Description	Initial State
DISP_ERROR_STATUS	[31:16]	Error status for the displayed frame. Error codes are defined in 1.3.2.2	0
DEC_ERROR_STATUS	[15:0]	Error status for the decoded/encoded frame. Error codes are defined in 1.3.2.2	0

6.3.2.1.11 RISC2HOST Argument Registers (MFC_RISC2HOST_ARG3, R/W, Address = 0xF170_0050)

MFC_RISC2HOST_ARG3	Bit	Description	Initial State
MFC_RISC2HOST_ARG3	[31:0]	<CONTINUE_ENC> The size of the output stream	0

6.3.2.1.12 RISC2HOST Argument Registers (MFC_RISC2HOST_ARG4, R/W, Address = 0xF170_0054)

MFC_RISC2HOST_ARG4	Bit	Description	Initial State
MFC_RISC2HOST_ARG4	[31:0]	Reserved	0

6.3.2.1.13 FIRMWARE Version Register (MFC_FIRMWARE_VERSION, R, Address = 0xF170_0058)

MFC_FIRMWARE_VERSION	Bit	Description	Initial State
Reserved	[31:24]	Reserved	0
YEAR	[23:16]	Year : 00~99 (00 means 2000)	0
MONTH	[15:8]	Month : 1~12	0
DAY	[7:0]	Day : 1~31	0

6.3.2.1.14 Debug Information Output Register1 (DBG_INFO_OUTPUT1, R, Address = 0xF170_0064)

DBG_INFO_OUTPUT1	Bit	Description	Initial State
INTERMEDIATE_STAGE_COUNTER	[31:0]	Intermediate stage counter for different stages of MFC firmware. This counter values will have different interpretation for each codec.	0

6.3.2.1.15 Debug Information Output Register2 (DBG_INFO_OUTPUT2, R, Address = 0xF170_0068)

DBG_INFO_OUTPUT2	Bit	Description	Initial State
EXCEPTION_STATUS	[31:0]	The status of the exception handler 0x01 = Bus error handler 0x02 = Illegal instruction handler 0x04 = Tick handler 0x10 = Trap handler 0x20 = Align handler 0x40 = Range handler 0x80 = DTLB miss exception handler 0x100 = ITLB miss exception handler 0x200 = Data page fault exception handler 0x400 = Instruction page fault exception handler	0

6.3.2.1.16 Firmware Status Register (MFC_FIRMWARE_STATUS, R, Address = 0xF170_0080)

MFC_FIRMWARE_STATUS	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0
FIRMWARE_STATUS	[0]	0 = Not ready 1 = Ready	0

6.3.2.2 Error Codes

Error Code	Error Name	Description
<Command control errors>		
1	INVALID_CHANNEL_NUM BER	If the channel used is more than the allowed limit. Currently it should be within range 0-15.
2	INVALID_COMMAND_ID	If the command used is illegal. Please use the commands given in the CHx_INST_ID register specification.
3	CHANNEL_ALREADY_IN_ USE	If the channel is already open and host is again trying to open the channel before closing it.
4	CHANNEL_NOT_OPEN_B EFORE_CHANNEL_ CLOSE	If the CLOSE_CH is done before OPEN_CH (i.e., at OPEN_CH = 0).
5	OPEN_CH_ERROR_SEQ _START	If the channel is not open. (ERROR in SEQ_START)
6	SEQ_START_ALREADY_ CALLED	If SEQ_START is already done and again. SEQ_START is issued for the same channel.
7	OPEN_CH_ERROR_INIT_ BUFFERS	If the channel is not open in INIT_BUFFERS
8	SEQ_START_ERRORINI T_BUFFERS	If SEQ_START is not complete before INIT_BUFFERS.
9	INIT_BUFFER_ALREADY _CALLED	If INIT_BUFFERS is already done and again. INIT_BUFFERS is issued for the same channel.
10	OPEN_CH_ERROR_FRA ME_START	If the channel is not open. (ERROR in FRAME_START)
11	SEQ_START_ERROR_FR AME_START	If SEQ_START is not complete before FRAME_START.
12	INIT_BUFFERS_ERROR_ FRAME_START	If INIT_BUFFERS is not complete before FRAME_START
13	CODEC_LIMIT_EXCEEDE D	Number of codecs are more than 16 (Currently this is not applicable)
20	MEM_ALLOCATION_FAIL ED	Memory allocation failed in the firmware
25	INSUFFICIENT_CONTEX T_SIZE	Context buffer size is insufficient
<SEQ_START errors>		
27	UNSUPPORTED_FEATU RE_IN_PROFILE	Features like CABAC/Interlace are not supported in baseline profile
28	RESOLUTION_NOT_SUP PORTED	Resolution is not supported
<Decoder fatal errors on SEQ_START>		
52	HEADER_NOT_FOUND	Header not found
<Encoder fatal errors on SEQ_START>		

Error Code	Error Name	Description
61	RESERVED	Reserved
62	FRAME_RATE_NOT_SUPPORED	When Rate Control is enabled, Frame Rate cannot have a zero value
63	INVALID_QP_VALUE	Invalid Qp value set
64	INVALID_RCREACTION_COEFFICIENT	Invalid value for Rate Control reaction parameter. Value of zero is prohibited
65	INVALID_CPB_SIZE_AT_GIVEN_LEVEL	Invalid value of CPB/VBV size at given level. This violates Annex A in H.264
<INIT_BUFFERS errors>		
71	ALLOC_DPB_SIZE_NOT_SUFFICIENT	Allocated DPB SIZE is insufficient
72	RESERVED	Reserved
73	RESERVED	Reserved
74	NUM_DPB_OUT_OF_RANGE	NUM_DPB is out of range. It should be equal or greater than MIN_NUM_DPB and equal or smaller than 32.
77	NULL_METADATA_INPUT_POINTER	External metadata input structure address is null
78	NULL_DPB_POINTER	The allocated DPB address is null
79	NULL_OTH_EXT_BUF_A_DDR	Other external buffers for decoder are NULL.
80	NULL_MV_POINTER	MV address is null
<Common HW errors>		
81	DIVIDE_BY_ZERO	Divide by zero error
82	BIT_STREAM_BUF_EXHAUSTED	Bit stream buffer exhausted
83	DESCRIPTOR_BUFFER_EMPTY	Empty descriptor buffer (valid for H264 and VC1 decoders only)
84	DMA_TX_NOT_COMPLETE	DMA operation not complete
<Decoder HW errors>		
85	MB_HEADER_NOT_DONE	MB header decode not done
86	MB_COEFF_NOT_DONE	MB coeff (entropy decoding)
87	CODEC_SLICE_NOT_DONE	Codec slice done error
88	MFC_CORE_TIME_OUT	Time out happens during the HW processing
89	VC1_BITPLANE_DECODER_ERR	VC1 bit plane decode error
<Encoder HW errors>		
90	VSP_NOT_READY	VSP not ready

Error Code	Error Name	Description
91	BUFFER_FULL_STATE	Buffer full
<Decoder errors on FRAME_RUN>		
112	RESOLUTION_MISMATCH	GOV has a resolution which exceeds the values in the sequence header
113	NV_QUANT_ERR	Errors in the quantization parameters
114	SYNC_MARKER_ERR	Sync marker error
115	FEATURE_NOT_SUPPORTED	Unsupported feature in the profile
116	MEM_CORRUPTION	MFC core memory corruption
117	INVALID_REFERENCE_FRAME	Reference frame(s) not available
118	PICTURE_CODING_TYPE_ERR	PICTURE_CODING_TYPE error in MPEG2
119	MV_RANGE_ERR	Invalid Fcode (MV_Range)
120	PICTURE_STRUCTURE_ERR	Picture_structure (FRAME/TOP/BOTTOM field)
121	SLICE_ADDR_INVALID	Invalid slice address
122	NON_PAIED_FIELD_NOT_SUPPORTED	Non-paired field is not supported
123	NON_FRAME_DATA_RECEIVED	Frame data is not received. Only header (e.g., seq header, SPS/PPS, SEI) is received
124	INCOMPLETE_FRAME	Incomplete frame data is received (e.g., only part of slices are received)
125	NO_BUFFER_RELEASED_FROM_HOST	No Free buffer available. All the buffers are either locked by host or they are anchor frames and cannot be used.
128	NALU_HEADER_ERROR	Invalid NALU Header
129	SPS_PARSE_ERROR	Invalid syntax element in SPS
130	PPS_PARSE_ERROR	Invalid syntax element in PPS
131	SLICE_PARSE_ERROR	Invalid syntax element in slice header
< Common warnings>		
145	COMMAND_FLUSHED	FRAME_START command has been flushed from the command channels 0 and 1
< Decoder warnings>		
150	METADATA_NO_SPACE_NUM_CONCEAL_MB	Out of space for the number of concealed MB metadata output
151	METADATA_NO_SPACE_QP	Out of space for QP metadata output
152	METADATA_NO_SAPCE_CONCEAL_MB	Out of space for concealed MB output
153	METADATA_NO_SPACE_VC1_PARAM	Out of space for VC1 parameter output

Error Code	Error Name	Description
154	METADATA_NO_SPACE_SEI	Out of space for SEI information output
155	METADATA_NO_SPACE_VUI	Out of space for VUI information output
156	METADATA_NO_SPACE_EXTRA	Out of space for extra data output
157	METADATA_NO_SPACE_DATA_NONE	Out of space for DataNone
158	FRAME_RATE_UNKNOW_N	Frame rate unknown
159	ASPECT_RATIO_UNKNOW_N	Aspect ratio unknown
160	COLOR_PRIMARIES_UN_KNOWN	Invalid color primaries
161	TRASNFER_CHAR_UNK_WON	Invalid trasnsfer characterstics
162	MATRIX_COEFF_UNKNOWWN	Invalid matrix coefficients
163	NON_SEQ_SLICE_ADDR	New slice address is not sequencial with respect to the old one
164	BROKEN_LINK	Current GOV has B pictures whose anchor frame is in the previous GOV
165	FRAME_CONCEALED	Error Concealment done by MFC
166	PROFILE_UNKOWN	Profile unknown
167	LEVEL_UNKOWN	Level unknown
168	BIT_RATE_NOT_SUPPORTED	Bit rate not supported
169	COLOR_DIFF_FORMAT_NOT_SUPPORTED	Color format is not supported
170	NULL_EXTRA_METADATA_POINTER	The allocated memory for extra metadata is null
171	SYNC_POINT_NOT_RECEIVED_STARTED_DECODING	DPB is empty but received a non I/IDR frame. MFC has started decoding from Non IDR
172	NULL_FW_DEBUG_INFO_POINTER	FW debug info address issued is null
173	ALLOC_DEBUG_INFO_SIZE_INSUFFICIENT	Allocated size for debug info is insufficient
<Encoder warnings>		
180	METADATA_NO_SPACE_	Out of space for Macroblock information output.

Error Code	Error Name	Description
	MB_INFO	(Applicable only for H.264 encoder)
181	METADATA_NO_SPACE_SLICE_SIZE	Out of space for slice size output
182	RESOLUTION_WARNING	Resoultion setting is not supported for given H.263 encoder profile

Samsung Confidential

6.3.2.3 Memory Controller Registers

6.3.2.3.1 Channel A DRAM Base Address Register (MFC_MC_DRAMBASE_ADDR_A, R/W, Address = 0xF170_0508)

MFC_MC_DRAMBASE_ADDR_A	Bit	Description	Initial State
MC_DRAMBASE_ADDR_A	[31:17]	The DRAM base address must be aligned at 128KByte. MFC's access range through port A is from DRAMBASE_ADDR_A to DRAMBASE_ADDR_A + 256MByte	0x6980
Reserved	[16:0]	Reserved	0

6.3.2.3.2 Channel B DRAM Base Address Register (MFC_MC_DRAMBASE_ADDR_B, R/W, Address = 0xF170_050C)

MFC_MC_DRAMBASE_ADDR_B	Bit	Description	Initial State
MC_DRAMBASE_ADDR_B	[31:17]	The DRAM base address must be aligned at 128KByte. MFC's access range through port B is from DRAMBASE_ADDR_B to DRAMBASE_ADDR_B + 256MByte	0x1180
Reserved	[16:0]	Reserved	0

6.3.2.3.3 MC (Memory Controller) Status Register (MFC_MC_STATUS, R, Address = 0xF170_0510)

MFC_MC_STATUS	Bit	Description	Initial State
Reserved	[31:2]	Not used	0
MC_BUSY_B	[1]	Busy at port B 0 = Idle 1 = Busy	X
MC_BUSY_A	[0]	Busy at port A 0 = Idle 1 = Busy	X

NOTE: X stands for undetermined.

6.3.2.4 Common Address Control

Common base address is variously used for each codec. The interpretation of the common base address is varying. Detailed descriptions are shown in [6.3.2.5](#) and [6.3.2.6](#).

Base addresses of common 0~63 are defined for AXI_MEMORY_A, and common 64~127 are defined for AXI_MASTER_B. Base address is determined as follows.

Base address calculation: (MC_DRAMBASE_ADDR) + (MFC_COMMON_BASE_ADDR<<11)

6.3.2.4.1 For Port_A: Common Baseram Register 0 ~ 63

Common Baseram Register 0 ~ 63	Address	Description	R/W	Initial State
MFC_COMMON_BASE_ADDR_0 [16:0]	0xF170_0600	Codec common memory region for start address.	R/W	X
~				
MFC_COMMON_BASE_ADDR_63 [16:0]	0xF170_06FC	Codec common memory region for start address.	R/W	X

6.3.2.4.2 For Port_B: Common Baseram Register 64 ~ 127

Common Baseram Register 64 ~ 127	Address	Description	R/W	Initial State
MFC_COMMON_BASE_ADDR_64 [16:0]	0xF170_0700	Codec common memory region for start address.	R/W	X
~				
MFC_COMMON_BASE_ADDR_127 [16:0]	0xF170_07FC	Codec common memory region for start address.	R/W	X

6.3.2.5 Buffer Address of Decoder

The base address settings for different standards are described in this section.

- H264 Decoder

Memory Region	Register Name	Description
H264DEC_VERT_NB_MV	MFC_COMMON_BASE_ADDR_35	Vertical Neighbor Motion Vector Buffer
H264DEC_NB_IP	MFC_COMMON_BASE_ADDR_36	Neighbor pixels for Intra Prediction Buffer
H264DEC_LUMA_x	MFC_COMMON_BASE_ADDR_64~95	Luma DPB for master channel A
H264DEC_CHROMA_x	MFC_COMMON_BASE_ADDR_0~31	Chroma DPB for master channel B
H264DEC_MV_x	MFC_COMMON_BASE_ADDR_96~127	MV buffer for H.264

- MPEG4 Decoder

Memory Region	Register Name	Description
DEC_NB_DCAC	MFC_COMMON_BASE_ADDR_35	Neighbor information of stream parser
DEC_UPNB_MV	MFC_COMMON_BASE_ADDR_36	Neighbor information of stream parser
DEC_SUB_ANCHOR_MV	MFC_COMMON_BASE_ADDR_37	Neighbor information of stream parser
OVERLAP_TRANSFORM	MFC_COMMON_BASE_ADDR_38	Information for Motion Compensation
DEC_STX_PARSER	MFC_COMMON_BASE_ADDR_42	Syntax Parser Buffer
DEC_LUMA_x	MFC_COMMON_BASE_ADDR_64~95	Reconstructed luma plane
DEC_CHROMA_x	MFC_COMMON_BASE_ADDR_0~31	Reconstructed chroma plane

- MPEG2 Decoder

Memory Region	Register Name	Description
DEC_LUMA_x	MFC_COMMON_BASE_ADDR_64~95	Reconstructed luma plane
DEC_CHROMA_x	MFC_COMMON_BASE_ADDR_0~31	Reconstructed chroma plane

- H.263 Decoder

Memory Region	Register Name	Description
DEC_NB_DCAC	MFC_COMMON_BASE_ADDR_35	Neighbor information of stream parser
DEC_UPNB_MV	MFC_COMMON_BASE_ADDR_36	Neighbor information of stream parser
DEC_SUB_ANCHOR_MV	MFC_COMMON_BASE_ADDR_37	Neighbor information of stream parser
OVERLAP_TRANSFORM	MFC_COMMON_BASE_ADDR_38	Information for Motion Compensation
DEC_LUMA_x	MFC_COMMON_BASE_ADDR_64~95	Reconstructed luma plane
DEC_CHROMA_X	MFC_COMMON_BASE_ADDR_0~31	Reconstructed chroma plane

- VC1 Decoder

Memory Region	Register Name	Description
DEC_NB_DCAC	MFC_COMMON_BASE_ADDR_35	Neighbor information of stream parser
DEC_UPNB_MV	MFC_COMMON_BASE_ADDR_36	Neighbor information of stream parser
DEC_SUB_ANCHOR_MV	MFC_COMMON_BASE_ADDR_37	Neighbor information of stream parser
OVERLAP_TRANSFORM	MFC_COMMON_BASE_ADDR_38	Information for Motion Compensation
BITPLANE3	MFC_COMMON_BASE_ADDR_39	BitPlane
BITPLANE2	MFC_COMMON_BASE_ADDR_40	
BITPLANE1	MFC_COMMON_BASE_ADDR_41	
DEC_LUMA_x	MFC_COMMON_BASE_ADDR_64~95	Reconstructed luma plane
DEC_CHROMA_x	MFC_COMMON_BASE_ADDR_0~31	Reconstructed chroma plane

- Buffer Memory Size for Decoder

Memory Region	Size			
	H.264	MPEG4/H.263	VC1	MPEG2
CH_ES_ADDR	Up to 4MB			
CH_DESC_ADDR	Up to 128KB			
DEC_NB_DCAC	-	16KB	16KB	-
DEC_UPNB_MV	-	68KB	68KB	-
DEC_SUB_ANCHOR_MV	-	136KB	136KB	-
DEC_OVERLAP_TRANSFOR	-	32KB	32KB	-
DEC_BITPLANE3	-	-	2KB	-
DEC_BITPLANE2	-	-	2KB	-
DEC_BITPLANE1	-	-	2KB	-
DEC_STX_PARSER	-	68KB(not needed for H263)	-	-
DEC_LUMA_x	-	align(align(x_size, 128) * align(y_size, 32), 8192)	align(align(x_size, 128) * align(y_size/2, 32), 8192)	align(align(x_size, 128) * align(y_size/4, 32), 8192)
H264DEC_VERT_NB_MV	16KB	-	-	-
H264DEC_NB_IP	32KB	-	-	-
H264DEC_CHROMA_x	align(align(x_size, 128) * align(y_size/2, 32), 8192) + align(align(x_size, 128) * align(y_size/4, 32), 8192)	-	-	-
H264DEC_LUMA_x	align(align(x_size, 128) * align(y_size, 32), 8192)	-	-	-
H264DEC_MV_x	Quarter size of H.264 Luma DPB align(align(x_size, 128) * align(y_size/4, 32), 8192)	-	-	-

NOTE:

1. All linear information in this table should be aligned at 2KB, whereas tile mode information (luma/chroma DPB and H264DEC_MV) should be aligned at 8KB.
2. DPB size did not take into account the QP save area.
3. x ranges 0~31 for LUMA_x, CHROMA_x, MV_x.

6.3.2.6 Buffer Address of Encoder

- H.264 Encoder

Memory Region	Register Name in User's Manual	Description
ENC_DPB_Y0_ADDR	MFC_COMMON_BASE_ADDR_7	Reconstructed Y0 buffer
ENC_DPB_C0_ADDR	MFC_COMMON_BASE_ADDR_64	Reconstructed C0 buffer
ENC_DPB_Y1_ADDR	MFC_COMMON_BASE_ADDR_8	Reconstructed Y1 buffer
ENC_DPB_C1_ADDR	MFC_COMMON_BASE_ADDR_65	Reconstructed C1 buffer
ENC_DPB_Y2_ADDR	MFC_COMMON_BASE_ADDR_68	Reconstructed Y2 buffer
ENC_DPB_C2_ADDR	MFC_COMMON_BASE_ADDR_66	Reconstructed C2 buffer
ENC_DPB_Y3_ADDR	MFC_COMMON_BASE_ADDR_69	Reconstructed Y3 buffer
ENC_DPB_C3_ADDR	MFC_COMMON_BASE_ADDR_67	Reconstructed C3 buffer
UPPER_MV_ADDR	MFC_COMMON_BASE_ADDR_0	Upper row MV storage region
DIRECT_COLZERO_FLAG_ADDR	MFC_COMMON_BASE_ADDR_4	Direct colocated flag storage region
UPPER_INTRA_MD_ADDR	MFC_COMMON_BASE_ADDR_2	Upper row current pixel data storage region
UPPER_INTRA_PRED_ADDR	MFC_COMMON_BASE_ADDR_80	Upper row pre-filter reconstruction data storage region
NBOR_INFO_MPENC_ADDR	MFC_COMMON_BASE_ADDR_1	Neighbor MB information storage region

- H.263 Encoder

Memory Region	Register Name in User's Manual	Description
ENC_DPB_Y0_ADDR	MFC_COMMON_BASE_ADDR_7	Reconstructed Y0 buffer
ENC_DPB_C0_ADDR	MFC_COMMON_BASE_ADDR_64	Reconstructed C0 buffer
ENC_DPB_Y1_ADDR	MFC_COMMON_BASE_ADDR_8	Reconstructed Y1 buffer
ENC_DPB_C1_ADDR	MFC_COMMON_BASE_ADDR_65	Reconstructed C1 buffer
ENC_DPB_Y2_ADDR	MFC_COMMON_BASE_ADDR_68	Reconstructed Y2 buffer
ENC_DPB_C2_ADDR	MFC_COMMON_BASE_ADDR_66	Reconstructed C2 buffer
ENC_DPB_Y3_ADDR	MFC_COMMON_BASE_ADDR_69	Reconstructed Y3 buffer
ENC_DPB_C3_ADDR	MFC_COMMON_BASE_ADDR_67	Reconstructed C3 buffer
UPPER_MV_ADDR	MFC_COMMON_BASE_ADDR_0	Upper row MV storage region
ACDC_COEF_BASE_ADDR	MFC_COMMON_BASE_ADDR_1	Upper row inverse quantization coefficient storage region

- MPEG4 Encoder

Memory Region	Register Name in User's Manual	Description
ENC_DPB_Y0_ADDR	MFC_COMMON_BASE_ADDR_7	Reconstructed Y0 buffer
ENC_DPB_C0_ADDR	MFC_COMMON_BASE_ADDR_64	Reconstructed C0 buffer
ENC_DPB_Y1_ADDR	MFC_COMMON_BASE_ADDR_8	Reconstructed Y1 buffer
ENC_DPB_C1_ADDR	MFC_COMMON_BASE_ADDR_65	Reconstructed C1 buffer
ENC_DPB_Y2_ADDR	MFC_COMMON_BASE_ADDR_68	Reconstructed Y2 buffer
ENC_DPB_C2_ADDR	MFC_COMMON_BASE_ADDR_66	Reconstructed C2 buffer
ENC_DPB_Y3_ADDR	MFC_COMMON_BASE_ADDR_69	Reconstructed Y3 buffer
ENC_DPB_C3_ADDR	MFC_COMMON_BASE_ADDR_67	Reconstructed C3 buffer
UPPER_MV_ADDR	MFC_COMMON_BASE_ADDR_0	Upper row MV storage region
DIRECT_COLZERO_FLAG_ADDR	MFC_COMMON_BASE_ADDR_4	Skip flag storage region
ACDC_COEF_BASE_ADDR	MFC_COMMON_BASE_ADDR_1	Upper row inverse quantization coefficient storage region

- Buffer Memory Size for Encoder

Memory Region	H.264	MPEG4 / H.263
ENC_DPB_Y0_ADDR	align(align(x_size, 128) * align(y_size, 32), 8192)	
ENC_DPB_C0_ADDR	align(align(x_size, 128) * align(y_size/2, 32), 8192)	
ENC_DPB_Y1_ADDR	align(align(x_size, 128) * align(y_size, 32), 8192)	
ENC_DPB_C1_ADDR	align(align(x_size, 128) * align(y_size/2, 32), 8192)	
ENC_DPB_Y2_ADDR	align(align(x_size, 128) * align(y_size, 32), 8192)	
ENC_DPB_C2_ADDR	align(align(x_size, 128) * align(y_size/2, 32), 8192)	
ENC_DPB_Y3_ADDR	align(align(x_size, 128) * align(y_size, 32), 8192)	
ENC_DPB_C3_ADDR	align(align(x_size, 128) * align(y_size/2, 32), 8192)	
CH_SB_ADDR	Configurable. Limit1: Should be aligned at 2KB Limit2: Should be a multiple of 4KB	Configurable. Limit1: Should be aligned at 2KB Limit2: Should be a multiple of 4KB
UPPER_MV_ADDR	align(xMB_size * 2 * 8, 2048) byte	align(xMB_size * 2 * 8, 2048)byte
DIRECT_COLZERO_FLAG_ADDR	align(((xMB_size * yMB_size+7)/8) * 8, 2048) byte	align(((xMB_size * yMB_size+7)/8) * 8, 2048) byte (not required for H263 encoder)
UPPER_INTRA_MD_ADDR	align(xMB_size * 48, 2048) byte	-
UPPER_INTRA_PRED_ADDR	align(1024 * 2 * 8, 2048) byte	-
NBOR_INFO_MPENC_ADDR	CAVLC: align(xMB_size * 8 * 8, 2048) byte CABAC: align(xMB_size * 24 * 8, 2048) byte	-
ACDC_COEF_BASE_ADDR	-	align((x_size/2) * 8, 2048) byte

NOTE:

1. All linear information in this table should be aligned at 2KB, whereas tile mode information (luma/chroma DPB) should be aligned at 8KB.
2. The division operation in the table is an integer division.
3. When 1 reference P is used, the required DPBs are DPB_Y0, DPB_Y1, DPB_C0, and DPB_C1 only.
Current luma, chroma buffers are placed in the opposite port.

6.3.3 CODEC REGISTERS

6.3.3.1 Codec Common Registers

6.3.3.1.1 Picture Width in Pixel Register (**MFC_HSIZE_PX**, R/W, Address = 0xF170_0818)

MFC_HSIZE_PX	Bit	Description	Initial State
Reserved	[31:13]	Reserved	0
PICTURE_WIDTH	[12:0]	Coded width of a picture	0

6.3.3.1.2 Picture Height in Pixel Register (**MFC_VSIZE_PX**, R/W, Address = 0xF170_081C)

MFC_VSIZE_PX	Bit	Description	Initial State
Reserved	[31:13]	Reserved	0
PICTURE_HEIGHT	[12:0]	Coded height of a picture (field or frame)	0

6.3.3.1.3 Profile Register (**MFC_PROFILE**, R/W, Address = 0xF170_0830)

MFC_PROFILE	Bit	Description	Initial State
Reserved	[31:16]	Reserved	0
LEVEL	[15:8]	Level in MPEG4 and H.264. In H.264, 31 stands for level 3.1 and 9 stands for level 1b. In MPEG4, 3 stands for level 3, 7 stands for level 3b, and 9 stands for level 0b.	0
Reserved	[7:6]	Reserved	0
PROFILE	[5:0]	<MPEG4> [0]: MPEG4_PROFILE 0 = Simple profile 1 = Advanced simple profile <H.264> [1:0]: Profile 0 = Main profile 1 = High profile 2 = Baseline profile Unspecified bits must set to be 0.	0

6.3.3.1.4 Picture Structure Register (*MFC_PICTURE_STRUCT*, R/W, Address = 0xF170_083C)

MFC_PICTURE_STRUCT	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0
FIELD	[0]	<H.264, MPEG4> 0 = Frame picture only 1 = Field picture	0

6.3.3.1.5 Loop Filter Control Register (*MFC_LF_CONTROL*, R/W, Address = 0xF170_0848)

MFC_LF_CONTROL	Bit	Description	Initial State
Reserved	[31:2]	Reserved	0
LF_CONTROL	[1:0]	<H.264> [1:0]: Loop filter disable indicator which corresponds to disable_deblocking_filter_idc 0 = Enable 1 = Disable 2 = Disable at slice boundary <MPEG4> [1]: Reserved [0]: Deblocking filter enable (post filter) 0 = Disable 1 = Enable	0

NOTE:

1. This register can be used by both encoders and decoders
2. This register is not effective for MPEG4 encoder
3. This register returns disable_deblocking_filter_idc from the bitstream for H.264 decoding

6.3.3.1.6 H.264 Loop Filter Alpha Offset Register (*MFC_LF_ALPHA_OFF*, R/W, Address = 0xF170_084C)

MFC_LF_ALPHA_OFF	Bit	Description	Initial State
Reserved	[31:5]	Reserved	0
LF_ALPHA_OFF	[4:0]	Loop filter alpha offset for deblocking filter	0

6.3.3.1.7 H.264 Loop Filter Beta Offset Register (MFC_LF_BETA_OFF, R/W, Address = 0xF170_0850)

MFC_LF_BETA_OFF	Bit	Description	Initial State
Reserved	[31:5]	Reserved	0
LF_BETA_OFF	[4:0]	Loop filter beta offset for deblocking filter	0

6.3.3.1.8 QP Information Offset Register (MFC_QP_OFFSET, R/W, Address = 0xF170_0C30)

MFC_QP_OFFSET	Bit	Description	Initial State
MFC_QP_OFFSET	[31:0]	When MFC_QP_OUT_EN is set, QP information is stored at the offset from the luma DPB address. The unit of the offset is double word (64bits).	0

6.3.3.1.9 QP Information Enable Register (MFC_QP_OUT_EN, R/W, Address = 0xF170_0C34)

MFC_QP_OUT_EN	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0
MFC_QP_OUT_EN	[0]	If MFC_QP_OUT_EN is enabled, the quantization value of each macroblock will be stored in the luma DPB area. 0 = QP out disable 1 = QP out enable The address is calculated as follows.	0

```

x_pos = [0...(img_hsize_mb-1)];
y_pos = frame ? [0...(img_vsize_mb-1)] :
    top    ? [0, 2, 4 ... (img_vsize_mb*2-2)] :
        [1, 3, 5 ... (img_vsize_mb*2-1)] ;

< l_xszie >
if (img_hsize_mb < 64)           l_xszie = 64
else if ((img_hsize_mb & 0x3f)!=0) l_xszie = ((img_hsize_mb>>6)<<6) + 64
else                           l_xszie = img_hsize_mb

< l_xszie >
if (frame) begin
    if(img_vsize_mb < 32)          l_ysize = 32 else if((img_vsize_mb & 0x1f)!=0)
                                    l_ysize = ((img_vsize_mb>>5)<<5) + 32 + 32
                                    l_ysize = img_vsize_mb + 32
    else
end
else begin
    if(img_vsize_mb < 16)          l_ysize = 32
    else if((img_vsize_mb & 0x1f)!=0) l_ysize = ((img_vsize_mb>>4)<<5) + 32 + 32
    else                           l_ysize = (img_vsize_mb<<1) + 32
end

pixel_x_m1 = l_xszie -1 ;
pixel_y_m1 = l_ysize -1 ;
roundup_x = ((pixel_x_m1)/16/8 + 1) ;
roundup_y = ((pixel_x_m1)/16/4 + 1) ;

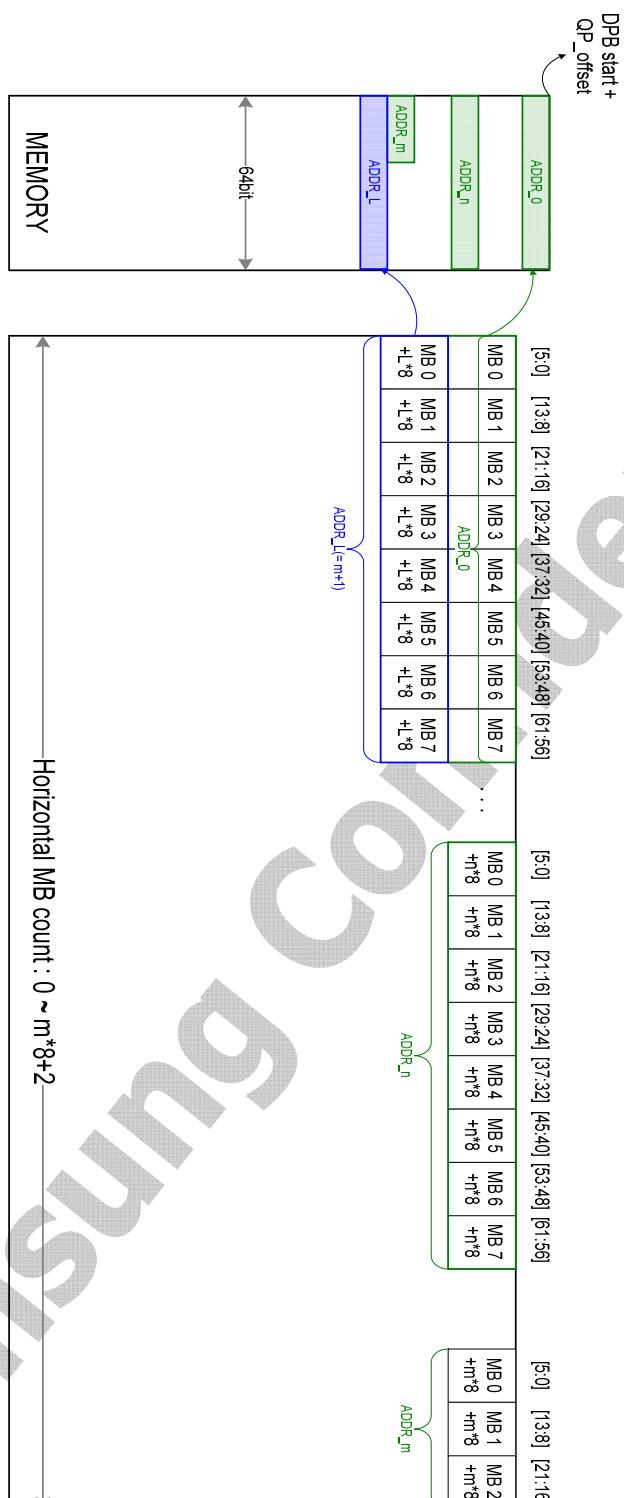
x_addr = x_pos/4;
linear_addr0 = (((y_pos  & 0x1f) <<4) |(x_addr & 0xf ) ) << 2 ;
linear_addr1 = (((y_pos >> 6) & 0xff) * roundup_x + ((x_addr >> 5) & 0x7f)) ;

if(  ((x_addr >> 5) & 0x1) ==  ((y_pos >> 5) & 0x1))
    bank_addr = ((x_addr >> 4) & 0x1);
else
    bank_addr = 0x2 | ((x_addr >> 4) & 0x1);

physical_addr = DRAM_BASE + DPB_OFFSET + QP_OFFSET + (linear_addr1 <<13) | (bank_addr << 11) |
linear_addr0 ;
qp_save_range = (pixel_y_minus[5]==0) ? pixel_y_minus[14:6] * roundup_x + pixel_x_minus[14:8] + 1 :
                                         roundup_x * roundup_y;

```

NOTE: QP values are set to zero for l_PCM macroblocks in H.264 and skipped macroblocks in VC1.
Host should allocate a physical memory size as follows:
- Memory size = ALIGN(img_hsize_mb, 64) * (ALIGN(img_vsize_mb, 32)+32)
Note that qp_save_range above specifies a virtual address area.



6.3.3.2 Channel and Stream Interface Registers

There are two sets of channels to communicate between host and MFC. Each channel has two types of registers. One is for response from MFC through MFC_SI_RTN_CHID and 15 MFC_COMMON_SI_RG registers. The other is for command from host through the MFC_SI_CH_INST_ID register and 15 MFC_COMMON_CHx_RG registers.

6.3.3.2.1 Return CH Instance ID Register (MFC_SI_RTN_CHID, R/W, Address = 0xF170_2000)

MFC_SI_RTN_CHID	Bit	Description	Initial State
RTN_CHID	[31:0]	Return channel instance ID which is used to identify which channel's operation is done	0

6.3.3.2.2 Common SI Register 1 ~ 15

- MFC_COMMON_SI_RG_1, R/W, Address = 0xF170_2004
- MFC_COMMON_SI_RG_2, R/W, Address = 0xF170_2008
- MFC_COMMON_SI_RG_3, R/W, Address = 0xF170_200C
- MFC_COMMON_SI_RG_4, R/W, Address = 0xF170_2010
- MFC_COMMON_SI_RG_5, R/W, Address = 0xF170_2014
- MFC_COMMON_SI_RG_6, R/W, Address = 0xF170_2018
- MFC_COMMON_SI_RG_7, R/W, Address = 0xF170_201C
- MFC_COMMON_SI_RG_8, R/W, Address = 0xF170_2020
- MFC_COMMON_SI_RG_9, R/W, Address = 0xF170_2024
- MFC_COMMON_SI_RG_10, R/W, Address = 0xF170_2028
- MFC_COMMON_SI_RG_11, R/W, Address = 0xF170_202C
- MFC_COMMON_SI_RG_12, R/W, Address = 0xF170_2030
- MFC_COMMON_SI_RG_13, R/W, Address = 0xF170_2034
- MFC_COMMON_SI_RG_14, R/W, Address = 0xF170_2038
- MFC_COMMON_SI_RG_15, R/W, Address = 0xF170_203C

MFC_COMMON_SI_RG_1 ~ 15	Bit	Description	Initial State
MFC_CH_COMMON_SI_RG_1 ~ 15	[31:0]	For specific meaning of each registers, refer to 6.3.3.3 and 6.3.3.4 .	0

NOTE: Note that the registers from 0xF170_2040 to 0xF170_207C have the same functionality as those from 0xF170_2080 to 0xF170_20BC. The registers from 0xF170_2040 to 0xF170_207C are used for channel 0 and those from 0xF170_2080 to 0xF170_20BC are for channel 1.

6.3.3.2.3 CH0 Instance ID Register (MFC_SI_CH0_INST_ID, R/W, Address = 0xF170_2040)

MFC_SI_CH0_INST_ID	Bit	Description	Initial State
Reserved	[31:19]	Reserved	0
CH_DEC_TYPE	[18:16]	[1:0]: CH0 control 1 = SEQ_START (sequence header processing) 2 = FRAME_START (frame decoding/encoding) 3 = LAST_SEQ (last frame decoding/encoding) 4 = INIT_BUFFERS (buffer initialization, Decoder only) 5 = FRAME_START_REALLOC (frame decoding for resolution change) 6 = FRAME_BATCH_START (frame batch encoding)	
CH_INST_ID	[15:0]	Instance ID for a codec	

6.3.3.2.4 CH1 Instance ID Register (MFC_SI_CH1_INST_ID, R/W, Address = 0xF170_2080)

MFC_SI_CH1_INST_ID	Bit	Description	Initial State
Reserved	[31:19]	Reserved	0
CH_DEC_TYPE	[18:16]	[1:0]: CH1 control 1 = SEQ_START (sequence header processing) 2 = FRAME_START (frame decoding/encoding) 3 = LAST_SEQ (last frame decoding/encoding) 4 = INIT_BUFFERS (buffer initialization. Decoder only) 5 = FRAME_START_REALLOC (frame decoding for resolution change) 6 = FRAME_BATCH_START (frame batch encoding)	
CH_INST_ID	[15:0]	Instance ID for Codec	

6.3.3.2.5 Common CH0 Register 1 ~ 15

- MFC_COMMON_CH0_RG_1, R/W, Address = 0xF170_2044
- MFC_COMMON_CH0_RG_2, R/W, Address = 0xF170_2048
- MFC_COMMON_CH0_RG_3, R/W, Address = 0xF170_204C
- MFC_COMMON_CH0_RG_4, R/W, Address = 0xF170_2050
- MFC_COMMON_CH0_RG_5, R/W, Address = 0xF170_2054
- MFC_COMMON_CH0_RG_6, R/W, Address = 0xF170_2058
- MFC_COMMON_CH0_RG_7, R/W, Address = 0xF170_205C
- MFC_COMMON_CH0_RG_8, R/W, Address = 0xF170_2060
- MFC_COMMON_CH0_RG_9, R/W, Address = 0xF170_2064
- MFC_COMMON_CH0_RG_10, R/W, Address = 0xF170_2068
- MFC_COMMON_CH0_RG_11, R/W, Address = 0xF170_206C
- MFC_COMMON_CH0_RG_12, R/W, Address = 0xF170_2070
- MFC_COMMON_CH0_RG_13, R/W, Address = 0xF170_2074
- MFC_COMMON_CH0_RG_14, R/W, Address = 0xF170_2078
- MFC_COMMON_CH0_RG_15, R/W, Address = 0xF170_207C

MFC_COMMON_CH0_RG_1 ~ 15	Bit	Description	Initial State
MFC_COMMON_CH0_RG_1 ~ 15	[31:0]	Host sets the parameters through these registers. For specific meaning of each register, refer to 6.3.3.3 and 6.3.3.4 .	0

6.3.3.2.6 Common CH1 Register 1 ~ 15

- MFC_COMMON_CH1_RG_1, R/W, Address = 0xF170_2084
- MFC_COMMON_CH1_RG_2, R/W, Address = 0xF170_2088
- MFC_COMMON_CH1_RG_3, R/W, Address = 0xF170_208C
- MFC_COMMON_CH1_RG_4, R/W, Address = 0xF170_2090
- MFC_COMMON_CH1_RG_5, R/W, Address = 0xF170_2094
- MFC_COMMON_CH1_RG_6, R/W, Address = 0xF170_2098
- MFC_COMMON_CH1_RG_7, R/W, Address = 0xF170_209C
- MFC_COMMON_CH1_RG_8, R/W, Address = 0xF170_20A0
- MFC_COMMON_CH1_RG_9, R/W, Address = 0xF170_20A4
- MFC_COMMON_CH1_RG_10, R/W, Address = 0xF170_20A8
- MFC_COMMON_CH1_RG_11, R/W, Address = 0xF170_20AC
- MFC_COMMON_CH1_RG_12, R/W, Address = 0xF170_20B0
- MFC_COMMON_CH1_RG_13, R/W, Address = 0xF170_20B4
- MFC_COMMON_CH1_RG_14, R/W, Address = 0xF170_20B8
- MFC_COMMON_CH1_RG_15, R/W, Address = 0xF170_20BC

MFC_COMMON_CH1_RG_1~15	Bit	Description	Initial State
MFC_COMMON_CH1_RG_1 ~ 15	[31:0]	While MFC is handling the request on CH0, host can communicate with MFC over CH1. MFC_COMMON_CH1_RG have the same meaning as MFC_COMMON_CH0_RG.	0

6.3.3.3 Decoder Channel and Stream Interface Registers

6.3.3.3.1 Vertical Resolution Register (MFC_COMMON_SI_RG_1, R, Address = 0xF170_2004)

MFC_COMMON_SI_RG_1	Bit	Description	Initial State
VER_RESOL	[31:0]	Vertical resolution of the current sequence or picture to be displayed. It should be read after decoding sequence header or after decoding a frame (in the case of resolution change).	0

6.3.3.3.2 Horizontal Resolution Register (MFC_COMMON_SI_RG_2, R, Address = 0xF170_2008)

MFC_COMMON_SI_RG_2	Bit	Description	Initial State
HOR_RESOL	[31:0]	Horizontal resolution of the current sequence or picture to be displayed. It should be read after decoding sequence header or after decoding a frame (in the case of resolution change).	0

6.3.3.3.3 Required Buffer Number Register (MFC_COMMON_SI_RG_3, R, Address = 0xF170_200C)

MFC_COMMON_SI_RG_3	Bit	Description	Initial State
MIN_NUM_DPB	[31:0]	Required decoded picture buffer number. After decoding sequence header, MFC sets the minimum number of required DPB buffers.	0

NOTE:

1. H.264: MaxDpbFrames+2 (MaxDpbFrames is defined in the standard)
2. H.263: 3
3. Other codecs: 4
4. The number above is determined by the number of reference buffers, one decoding buffer, and one display buffer

6.3.3.3.4 Display Order Luminance Address Register (MFC_COMMON_SI_RG_4, R, Address = 0xF170_2010)

MFC_COMMON_SI_RG_4	Bit	Description	Initial State
DISPLAY_Y_ADR	[31:0]	Display luminance address in display order	0

6.3.3.3.5 Display Order Chrominance Address Register (MFC_COMMON_SI_RG_5, R, Address = 0xF170_2014)

MFC_COMMON_SI_RG_5	Bit	Description	Initial State
DISPLAY_C_ADR	[31:0]	Display chrominance address in display order	0

6.3.3.3.6 Decoded Frame Size Register (MFC_COMMON_SI_RG_6, R, Address = 0xF170_2018)

MFC_COMMON_SI_RG_6	Bit	Description	Initial State
MFC_DEC_FRM_SIZE	[31:0]	Consumed number of bytes to decode a frame	0

6.3.3.3.7 Display Status Register (MFC_COMMON_SI_RG_7, R, Address = 0xF170_201C)

MFC_COMMON_SI_RG_7	Bit	Description	Initial State
Reserved	[31:6]	Reserved	0
DISPLAY_STATUS	[5:0]	<p>The status of the decoded picture to be displayed.</p> <p>[5:4] : Resolution change 0 = No change 1 = Resolution increased 2 = Resolution decreased</p> <p>[3] : Progressive/interlace 0 = Progressive frame. 1 = Interlace frame</p> <p>[2:0] : Display status 0 = Decoding only (no display) 1 = Decoding and display. 2 = Display only. 3 = DPB is empty and decoding is finished</p>	0

6.3.3.3.8 Frame Type Register (MFC_COMMON_SI_RG_8, R, Address = 0xF170_2020)

MFC_COMMON_SI_RG_8	Bit	Description	Initial State
Reserved	[31:3]	Reserved	0
FRAME_TYPE	[2:0]	<p>This register returns a frame type of a decoded frame</p> <p>0 = Not coded frame (skipped frame) 1 = I frame 2 = P frame 3 = B frame 4 = Others</p>	0

6.3.3.3.9 Decoding Order Luminance Address Register (MFC_COMMON_SI_RG_9, R, Address = 0xF170_2024)

MFC_COMMON_SI_RG_9	Bit	Description	Initial State
DECODE_Y_ADR	[31:0]	Luminance address in decoding order	0

6.3.3.3.10 Decoding Order Chrominance Address Register (MFC_COMMON_SI_RG_10, R, Address = 0xF170_2028)

MFC_COMMON_SI_RG_10	Bit	Description	Initial State
DECODE_C_ADR	[31:0]	Chrominance address in decoding order	0

6.3.3.3.11 Decoding Status Register (MFC_COMMON_SI_RG_11, R, Address = 0xF170_202C)

MFC_COMMON_SI_RG_11	Bit	Description	Initial State
Reserved	[31:4]	Reserved	0
DECODE_STATUS	[3:0]	The status of the decoded picture. [3] : Progressive/interlace 0 = Progressive frame. 1 = Interlace frame [2:0] : Decoding status 0 = Decoding only (no display) 1 = Decoding and display. 2 = Display only. 3 = DPB is empty and decoding is finished 4 = No decoding and no display (equivalent to DISPLAY_STATUS=2)	0

6.3.3.3.12 CH External Stream Buffer Address Register (MFC_COMMON_CHx_RG_1, R/W, Address = 0xF170_2044 or 0xF170_2084)

MFC_COMMON_CHx_RG_1	Bit	Description	Initial State
CH_ES_ADDR	[31:0]	Start address of the CPB of the elementary stream to be decoded	0

NOTE: The address should be in Port_A (i.e., the address should be between MC_DRAMBASE_ADDR_A and MC_DRAMBASE_ADDR_A+256MB).

6.3.3.3.13 CH Decoding Unit Size Register (MFC_COMMON_CHx_RG_2, R/W, Address = 0xF170_2048 or 0xF170_2088)

MFC_COMMON_CHx_RG_2	Bit	Description	Initial State
CH_ES_DEC_UNIT_SIZE	[31:0]	Decoding unit size in the CPB	0

6.3.3.3.14 CH Descriptor Buffer Address (MFC_COMMON_CHx_RG_3, R/W, Address = 0xF170_204C or 0xF170_208C)

MFC_COMMON_CHx_RG_3	Bit	Description	Initial State
CH_DESC_ADDR	[1:0]	Channel descriptor buffer address	0

NOTE: The address should be in Port_A (i.e., the address should be between MC_DRAMBASE_ADDR_A and MC_DRAMBASE_ADDR_A+256MB).

6.3.3.3.15 Reserved (MFC_COMMON_CHx_RG_4, W, Address = 0xF170_2050 or 0xF170_2090)

MFC_COMMON_CHx_RG_4	Bit	Description	Initial State
Reserved	[31:0]	Reserved	0

6.3.3.3.16 Reserved (MFC_COMMON_CHx_RG_5, W, Address = 0xF170_2054 or 0xF170_2094)

MFC_COMMON_CHx_RG_5	Bit	Description	Initial State
Reserved	[31:0]	Reserved	0

6.3.3.3.17 CPB Size Register (MFC_COMMON_CHx_RG_6, R/W, Address = 0xF170_2058 or 0xF170_2098)

MFC_COMMON_CHx_RG_6	Bit	Description	Initial State
CPB_SIZE	[31:0]	CPB size register should be set before SEQ_START and FRAME_START. The maximum CPB size is 4MB.	0

NOTE: CPB_SIZE = align(CH_ES_DEC_UNIT_SIZE+64, pow(2KB)) for H.264 and VC1 decoders when DMX is enabled, where pow(2KB)=1KB, 2KB, 4KB, 8KB, ..., 4MB.

6.3.3.3.18 Descriptor Buffer Size Register (MFC_COMMON_CHx_RG_7, R/W, Address = 0xF170_205C or 0xF170_209C)

MFC_COMMON_CHx_RG_7	Bit	Description	Initial State
DESC_SIZE	[31:0]	Descriptor buffer size register should be set before SEQ_START and FRAME_START. The maximum descriptor buffer size is 128KB.	0

6.3.3.3.19 Release Buffer Register (MFC_COMMON_CHx_RG_8, R/W, Address = 0xF170_2060 or 0xF170_20A0)

MFC_COMMON_CHx_RG_8	Bit	Description	Initial State
RELEASE_BUFFER	[31:0]	Release buffer register specifies the availability of each DPB. The nth bit specifies the availability of the nth DPB. 1 means free and 0 means busy.	0

6.3.3.3.20 Auxiliary Host Command Register (MFC_COMMON_CHx_RG_9, R/W, Address = 0xF170_2064 or 0xF170_20A4)

MFC_COMMON_CHx_RG_9	Bit	Description	Initial State
HOST_WR_ADR	[31:0]	The address points to a space of shared memory consisting of multiple commands which host can read/write. The detailed structure of the shared memory is described in chapter 6.4	0

6.3.3.3.21 DPB Configuration Control Register (MFC_COMMON_CHx_RG_10, R/W, Address = 0xF170_2068 or 0xF170_20A8)

MFC_COMMON_CHx_RG_10	Bit	Description	Initial State
SLICE_IF_ENABLE	[31]	Enable slice interface for decoding 0 = Disable 1 = Enable	0
CONFIG_DELAY_ENABLE	[30]	Enable configurable display delay for H.264 decoding 0 = Disable 1 = Enable	0
DISPLAY_DELAY	[29:16]	Number of frames for display delay. MFC is forced to return frames for display even if DPB is not filled. It is valid for H.264 decoder only.	0
DMX_DISABLE	[15]	Host may generate the descriptor information on behalf of MFC demux. This register is valid for H.264 and VC1 decoders only. 0 = Enable demux so that MFC generates the descriptor information. 1 = Disable demux so that host generates the descriptor information.	
DPB_FLUSH	[14]	Flushing DPB to discard all the output buffers in DPB 0 = Normal operation 1 = Flushing DPB	
NUM_DPB	[13:0]	Number of DPB that host prepared for decoding	0

NOTE:

- When demux is disabled, host has to fill out the descriptor buffer for each slice/NALU. For VC1, after constructing all the descriptors, one dummy descriptor needs to be created with start code suffix byte 0x82 (ID [7:0] = 0x82) followed by a zero word to indicate the end of descriptors.
- For both H264 and VC1, after all descriptor entries (including VC1 dummy descriptor entry), zero word is written to descriptor table to indicate end of descriptors. word is written to descriptor table to indicate end of descriptors.

ID[7:0]	Offset[2:0]	Start_addr[20:0]
	9'd0	Unit_size[22:0]
		32'd0
		32'd0

ID: start code suffix byte (first byte after start code: NALU header byte for H.264 and 0x82 for VC1)

Offset: (nal start address) & 0x07

Start_addr: ((nal start address) >> 3) << 1

Unit_size: size of NALU

6.3.3.3.22 Command Sequence Number Register (MFC_COMMON_CHx_RG_11, R/W, Address = 0xF170_206C or 0xF170_20AC)

MFC_COMMON_CHx_RG_11	Bit	Description	Initial State
CMD_SEQ_NUM	[31:0]	Command sequence number from the host. The sequence number is used for in order processing of commands.	0

6.3.3.4 Encoder Channel and Stream Interface Registers

6.3.3.4.1 Stream Size Register (MFC_COMMON_SI_RG_1, R, Address = 0xF170_2004)

MFC_COMMON_SI_RG_1	Bit	Description	Initial State
ENC_STREAM_SIZE	[31:0]	Encoded stream size in byte count	0

6.3.3.4.2 Encoded Picture Count Register (MFC_COMMON_SI_RG_2, R, Address = 0xF170_2008)

MFC_COMMON_SI_RG_2	Bit	Description	Initial State
ENC_PICTURE_CNT	[31:0]	Encoded picture count. In the interlaced streams, It increments field by field.	0

6.3.3.4.3 Write Pointer Register (MFC_COMMON_SI_RG_3, R, Address = 0xF170_200C)

MFC_COMMON_SI_RG_3	Bit	Description	Initial State
WRITE_POINTER	[31:0]	Stream buffer write pointer. EDFU updates external memory address at the end of encoding a frame.	0

6.3.3.4.4 Slice type Register (MFC_COMMON_SI_RG_4, R/W, Address = 0xF170_2010)

MFC_COMMON_SI_RG_4	Bit	Description	Initial State
ENC_SLICE_TYPE	[31:0]	0 = Not coded frame 1 = I frame 2 = P frame 3 = B frame 4 = Skipped frame 5 = Others	0

6.3.3.4.5 Encoded Y Address Register (MFC_COMMON_SI_RG_5, R/W, Address = 0xF170_2014)

MFC_COMMON_SI_RG_5	Bit	Description	Initial State
ENCODED_Y_ADDR	[31:0]	The address of the encoded luminance picture	0

6.3.3.4.6 Encoded C Address Register (MFC_COMMON_SI_RG_6, R/W, Address = 0xF170_2018)

MFC_COMMON_SI_RG_6	Bit	Description	Initial State
ENCODED_C_ADDR	[31:0]	The address of the encoded chrominance picture	0

6.3.3.4.7 CH External Stream Buffer Start Address Register (MFC_COMMON_CHx_RG_1, R/W, Address = 0xF170_2044 or 0xF170_2084)

MFC_COMMON_CHx_RG_1	Bit	Description	Initial State
CH_SB_ADDR	[31:0]	Start address of the stream buffer at encoder.	0

NOTE: The address should be in Port_A (i.e., the address should be between MC_DRAMBASE_ADDR_A and MC_DRAMBASE_ADDR_A+256MB).

6.3.3.4.8 Stream Buffer Size Register (MFC_COMMON_CHx_RG_3, R/W, Address = 0xF170_204C or 0xF170_208C)

MFC_COMMON_CHx_RG_3	Bit	Description	Initial State
BUFFER_SIZE	[31:0]	Buffer size of the encoded stream	0

6.3.3.4.9 Current Y Address Register (MFC_COMMON_CHx_RG_4, R/W, Address = 0xF170_2050 or 0xF170_2090)

MFC_COMMON_CHx_RG_4	Bit	Description	Initial State
CURRENT_Y_ADDR	[31:0]	The address of the current luminance picture to encode	0

NOTE: The address should be in Port_B (i.e., the address should be between MC_DRAMBASE_ADDR_B and MC_DRAMBASE_ADDR_B+256MB).

The buffer address should be aligned as follows.

- Tile mode : align(align(x_size, 128) * y_size, 8192)
- Linear mode : align(align(x_size, 16) * y_size, 2048)

6.3.3.4.10 Current C Address Register (MFC_COMMON_CHx_RG_5, R/W, Address = 0xF170_2054 or F170_2094)

MFC_COMMON_CHx_RG_5	Bit	Description	Initial State
CURRENT_C_ADDR	[31:0]	The address of the current chrominance picture to encode	0

NOTE: The address should be in Port_B (i.e., the address should be between MC_DRAMBASE_ADDR_B and MC_DRAMBASE_ADDR_B+256MB).

The buffer address should be aligned as follows.

- Tile mode : align(align(x_size, 128) * y_size/2, 8192)
- Linear mode : align(align(x_size, 16) * y_size/2, 2048)

6.3.3.4.11 Frame Insertion Control Register (MFC_COMMON_CHx_RG_6, R/W, Address = 0xF170_2058 or 0xF170_2098)

MFC_COMMON_CHx_RG_6	Bit	Description	Initial State
RESERVED	[31:0]	Reserved	0
NOT_CODED	[1]	Current frame must be encoded into a not coded frame	0
I_FRAME	[0]	Current frame must be encoded into an I frame. It will be effective at the next anchor frame.	0

6.3.3.4.12 Auxiliary Host Command Register (MFC_COMMON_CHx_RG_9, R/W, Address = 0xF170_2064 or 0xF170_20A4)

MFC_COMMON_CHx_RG_9	Bit	Description	Initial State
HOST_WR_ADDR	[31:0]	The address points to a space of shared memory consisting of multiple commands which host can read/write. The detailed structure of the shared memory is described in chapter 6.4	0

6.3.3.4.13 Encoder Input Buffer Flush Register (MFC_COMMON_CHx_RG_10, R/W, Address = 0xF170_2068 or 0xF170_20A8)

MFC_COMMON_CHx_RG_10	Bit	Description	Initial State
RESERVED	[31:15]	Reserved	0
INPUT_BUFFER_FLUSH	[14]	Flushing input buffer to discard all frame in the input buffer. 0 = Normal operation 1 = Flushing input buffer	0
RESERVED	[13:0]	Reserved	0

6.3.3.4.14 Command Sequence Number Register (MFC_COMMON_CHx_RG_11, R/W, Address = 0xF170_206C or 0xF170_20AC)

MFC_COMMON_CHx_RG_9	Bit	Description	Initial State
CMD_SEQ_NUM	[31:0]	Command sequence number from the host. The sequence number is used for in order processing of commands.	0

6.3.4 ENCODING REGISTERS

6.3.4.1 Common Encoder Register

6.3.4.1.1 Picture Type Control Register (ENC_PIC_TYPE_CTRL, R/W, Address = 0xF170_C504)

ENC_PIC_TYPE_CTRL	Bit	Description	Initial State
Reserved	[31:19]	Reserved	0
ENC_PIC_TYPE_ENABLE	[18]	0 = Disable ENC_PIC_TYPE_CTRL 1 = Enable ENC_PIC_TYPE_CTRL[17:0] for picture type setting	0
B_FRM_CTRL	[17:16]	0 = The number of B frames is zero 1 = The number of B frames is one 2 = The number of B frames is two 3 = Reserved	0
I_FRM_CTRL	[15:0]	0 = All P frames 1 = All I frames 2 = I – P – I – P 3 = I – P – P – I N = (N-1) P frames between two I frames	0

6.3.4.1.2 B-Picture Recon Picture Writing Control Register (ENC_B_RECON_WRITE_ON, R/W, Address = 0xF170_C508)

ENC_B_RECON_WRITE_ON	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0
B_RECON_ON	[0]	This register is used for debugging. By default, it is set to zero. If it is set, it is required to allocate the required memory. 0 = Disable recon data write at B-frame 1 = Enable recon data write at B-frame	0

NOTE: When B_RECON_ON is enabled, host has to allocate B_FRAME_RECON_LUMA_ADDR (0x062C) and B_FRAME_RECON_CHROMA_ADDR (0x0630). The size should be as follows:
 - sizeof(B_FRAME_RECON_LUMA_ADDR) = align(align(x_size, 128) * align(y_size, 32), 8192)
 - sizeof(B_FRAME_RECON_CHROMA_ADDR) = align(align(x_size, 128) * align(y_size/2, 32), 8192)

6.3.4.1.3 Multi-slice Control Register (ENC_MSLICE_CTRL, R/W, Address = 0xF170_C50C)

ENC_MSLICE_CTRL	Bit	Description	Initial State
Reserved	[31:3]	Reserved	0
MSLICE_MODE	[2:1]	0 = Multi slicing is done by MB count 1 = Multi slicing is done by byte count	0
MSLICE_ENA	[0]	0 = One slice per frame 1 = Enable multi slice or resync marker	0

6.3.4.1.4 Macroblock Number of Multi-slice Register (ENC_MSLICE_MB, R/W, Address = 0xF170_C510)

ENC_MSLICE_MB	Bit	Description	Initial State
Reserved	[31:16]	Reserved	0
MSLICE_MB	[15:0]	The number of macroblocks in one slice. Valid if MSLICE_MODE=0 and MSLICE_ENA=1.	0

6.3.4.1.5 Size of Multi-slice Register (ENC_MSLICE_BYTE, R/W, Address = 0xF170_C514)

ENC_MSLICE_BIT	Bit	Description	Initial State
MSLICE_BIT	[31:0]	The number of bit count in one slice. Valid if MSLICE_MODE=1 and MSLICE_ENA=1.	0

NOTE: The minimum size of MSLICE_BIT is 1900bits

6.3.4.1.6 Cyclic Intra Refresh Register (ENC_CIR_CTRL, R/W, Address = 0xF170_C518)

ENC_CIR_CTRL	Bit	Description	Initial State
Reserved	[31:16]	Reserved	0
CIR_NUM	[15:0]	Number of intra refresh macroblocks	0

6.3.4.1.7 Memory Structure Setting Register of Current Frame (ENC_MAP_FOR_CUR, R/W, Address = 0xF170_C51C)

ENC_MAP_FOR_CUR	Bit	Description	Initial State
Reserved	[31:2]	Reserved	0
ENC_MAP_FOR_CUR	[1:0]	Memory structure of the current frame 0 = Linear mode 3 = 64x32 tiled mode	0

6.3.4.1.8 Padding Value Control Register (ENC_PADDING_CTRL, R/W, Address = 0xF170_C520)

ENC_PADDING_CTRL	Bit	Description	Initial State
PAD_CTRL_ON	[31]	0 = Use boundary pixel for current image padding in case that its image size is not a multiple of 16 1 = Use ENC_PADDING_CTRL[23:0] for current image padding	0
Reserved	[30:24]	Reserved	0
CR_PAD_VAL	[23:16]	Value for original CR image's padding when PAD_CTRL_ON is 1.	0
CB_PAD_VAL	[15:8]	Value for original CB image's padding when PAD_CTRL_ON is 1.	0
LUMA_PAD_VAL	[7:0]	Value for original Y image's padding when PAD_CTRL_ON is 1.	0

6.3.4.1.9 Encoder Intra Mode Bias Register (ENC_COMMON_INTRA_BIAS, R/W, Address = 0xF170_C588)

ENC_INT_MASK	Bit	Description	Initial State
Reserved	[31:16]	Reserved	0
ENC_COMMON_INTRA_BIAS	[15:0]	The register is used in favor of the intra mode in the weighted macroblock mode decision. Mode decision will be done by comparing 1) inter_cost + ENC_COMMON_INTRA_BIAS and 2) intra_cost. If inter_cost is zero, the mode is always determined to inter macroblock. To disable this option, set ENC_COMMON_INTRA_BIAS to zero.	0

6.3.4.1.10 Encoder Bi-directional Mode Bias Register (*ENC_COMMON_BI_DIRECT_BIAS*, R/W, Address = 0xF170_C58C)

ENC_INT_MASK	Bit	Description	Initial State
Reserved	[31:16]	Reserved	0
ENC_COMMON_BI_DIRECT_BIAS	[15:0]	<p>The register is used against the bi-directional mode in the weighted macroblock mode decision.</p> <p>Mode decision will be done by comparing 1) uni_direction_cost and 2) bi_direction_cost + ENC_COMMON_BI_DIRECT_BIAS.</p> <p>If the macroblock type is INTRA, this register is of no effect.</p> <p>To disable this option, set ENC_COMMON_BI_DIRECT_BIAS to zero.</p> <p>This register is effective in MPEG4 encoding only.</p>	0

6.3.4.2 Rate Control Register

6.3.4.2.1 Rate Control Configuration Register (*RC_CONFIG*, R/W, Address = 0xF170_C5A0)

RC_CONFIG	Bit	Description	Initial State
Reserved	[31:10]	Reserved	0
FR_RC_EN	[9]	Frame level rate control enable 0 = Disable frame level rate control 1 = Enable frame level rate control	0
MB_RC_EN	[8]	Macroblock level rate control enable. Valid only for H.264 0 = Disable MB level rate control 1 = Enable MB level rate control	0
Reserved	[7:6]	Reserved	0
FRAME_QP	[5:0]	<p>Frame QP (quantization parameter) FRAME_QP is used for the first macroblock QP in a frame. This value should be set in the range of MIN_QP to MAX_QP in RC_QBOUND. The QP of the next macroblocks can be changed as per the value of FR_RC_EN and MB_RC_EN. The interpretation of FRAME_QP is varying as per RC_CONFIG[9:8] as follows.</p> <p><RC_CONFIG[9:8]></p> <p>2'b00: Constant QP is applied to all macroblocks. FRAME_QP is used for I frame. P_FRAME_QP and B_FRAME_QP are used for P and B frames.</p> <p>2'b01: The QP of the next macroblocks can vary with macroblock adaptive scaling. But it does not take into account the size of generated bits.</p> <p>2'b10: The QP of the next macroblocks can be changed by the difference between the numbers of target bit and generated bit during the encoding a picture. But macroblock adaptive scaling is not applied.</p> <p>2'b11: The QP of the next macroblocks can be changed by the difference between the numbers of target bit and generated bit during the encoding a picture. It also can vary with macroblock adaptive scaling.</p>	

6.3.4.2.2 RC Frame Rate Register (RC_FRAME_RATE, R/W, Address = 0xF170D0D0)

RC_FRAME_RATE	Bit	Description	Initial State
FRAME_RATE	[31:0]	Frames per second in 1000x scale (e.g., 7500 stands for 7.5 frames/sec). '0' is forbidden. Valid only when frame level RC is enabled.	0

6.3.4.2.3 RC Bit Rate Register (RC_BIT_RATE, R/W, Address = 0xF170_C5A8)

RC_BIT_RATE	Bit	Description	Initial State
BIT_RATE	[31:0]	Bits per second. '0' is forbidden. Valid only when Frame level RC is enabled.	0

6.3.4.2.4 RC Quantization Parameter Boundary Register (RC_QBOUND, R/W, Address = 0xF170_C5AC)

RC_QBOUND	Bit	Description	Initial State
Reserved	[31:14]	Reserved	0
MAX_QP	[13:8]	Maximum quantization parameter. The range is given as follows. - H.264: 0~51 - MPEG4, H.263: 1~31	0
Reserved	[7:6]	Reserved	0
MIN_QP	[5:0]	Minimum quantization parameter. The range is given as follows. - H.264: 0~51 - MPEG4, H.263: 1~31	0

NOTE: MAX_QP must be greater than or equal to MIN_QP.

6.3.4.2.5 Reaction Coefficient Register (RC_RPARA, R/W, Address = 0xF170_C5B0)

RC_RPARA	Bit	Description	Initial State
Reserved	[31:16]	Reserved	0
REACT_PARA	[15:0]	Rate control reaction coefficient. '0' is forbidden.	0

NOTE:

1. Valid only when the frame level RC is enabled.
2. For tight CBR, this field must be small (ex. 2 ~ 10).
For VBR, this field must be large (ex. 100 ~ 1000).
3. It is not recommended to use the greater number than FRAME_RATE * (10^9 / BIT_RATE).

6.3.4.2.6 Macroblock Level Rate Control Register (RC_MB_CTRL, R/W, Address = 0xF170_C5B4)

RC_MB_CTRL	Bit	Description	Initial State
Reserved	[31: 4]	Reserved	0
DARK_DISABLE	[3]	Disable dark region adaptive feature. 0 = Enable dark region adaptive feature 1 = Disable dark region adaptive feature QP of dark MB may not be smaller than frame QP although it is smooth, static or it has small activity.	0
SMOOTH_DISABLE	[2]	Disable smooth region adaptive feature. 0 = Enable smooth region adaptive feature. 1 = Disable smooth region adaptive feature. QP of smooth MB may be smaller than frame QP.	0
STATIC_DISABLE	[1]	Disable static region adaptive feature. 0 = Enable static region adaptive feature. 1 = Disable static region adaptive feature. QP of static MB may be smaller than frame QP.	0
ACT_DISABLE	[0]	Disable MB activity adaptive feature. 0 = Enable MB activity adaptive feature. 1 = Disable MB activity adaptive feature. QP of MB that has small activity may be smaller than frame QP and QP of MB that has large activity may be larger than frame QP	0

NOTE: Valid only when H.264 and macroblock level RC is enabled.

6.3.4.3 H.264 Encoder Register

6.3.4.3.1 H.264 Entropy Register (H264_ENC_ENTRP_MODE, R/W, Address = 0xF170_D004)

H264_ENC_ENTRP_MODE	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0
H264_ENC_ENTRP_MODE	[0]	0 = CAVLC 1 = CABAC	0

6.3.4.3.2 H.264 Number of Reference Register (H264_ENC_NUM_OF_REF, R/W, Address = 0xF170_D010)

H264_ENC_NUM_OF_REF	Bit	Description	Initial State
Reserved	[31:5]	Reserved	0
H264_ENC_P_NUM_OF_REF	[6:5]	The number reference pictures of P-picture. 1 = 1 reference frame 2 = 2 reference frame	0
H264_ENC_NUM_OF_REF	[4:0]	The maximum number of reference pictures. 1 = 1 reference frame 2 = 2 reference frame	0

6.3.4.3.3 H.264 8X8 Transform Enable Flag Register (H264_ENC_TRANS_8X8_FLAG, R/W, Address = 0xF170_D034)

H264_ENC_TRANS_8X8_FLAG	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0
ENC_TRANS_8X8_FLAG	[0]	0 = Disable 1 = Enable	0

6.3.4.4 MPEG4 Encoder Register

6.3.4.4.1 MPEG4 Quarter Pixel Interpolation Register (MPEG4_ENC_QUART_PXL, R/W, Address = 0xF170_E008)

MPEG4_ENC_QUART_PXL	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0
MPEG4_QUART_PXL	[0]	0 = Quarter pixel search disable 1 = Quarter pixel search enable	0

6.4 SHARED MEMORY INTERFACE

As per the growth of the diverse requirements for the codec features and additional enhancements of the MFC firmware, MFC provides the shared memory interface to exchange information with the host. Host is able to get returned parameters or set parameter values through the shared memory. Since the parameters are exchanged through the external memory, there is no limitation of the number of fields. Readers may consider it as a register group but the physical memory is allocated in the external memory.

This section describes the shared memory interface, such as shared memory allocation and the shared memory structure consisting of multiple fields.

6.4.1 HOST INTERFACE

Host has to allocate the shared memory and informs MFC of the buffer pointer through the HOST_WR_ADR register. It is recommended to allocate the shared memory before the sequence header parsing through SEQ_START. Since the number of fields in the shared memory is fixed in 0, host is required to allocate a buffer in the external memory to accommodate them.

Once the shared memory has been allocated, host and MFC are able to exchange information through the shared memory. Host can read and write each field in the shared memory at the byte offset defined in 0.

Following is a pseudo code to initialize the shared memory.

```
// shared mem allocation. Should be done before SEQ_START  
shared_mem_ptr = (int *) malloc(SHARED_MEM_SIZE);  
host_write_word(HOST_WR_ADR, shared_mem_ptr);
```

Following is a pseudo code to read and write the frame tags.

```
// write frame_tag  
host_write_word(shared_mem_ptr+ADR_SET_FRAE_TAG, frame_tag);  
  
// read frame_tag  
get_frame_tag = host_read_word(shared_mem_ptr+ADR_GET_FRAME_TAG_TOP);
```

Note that, at the beginning of a stream encoding/decoding, host must initialize all the fields in the shared memory corresponding to that stream to avoid any undefined behaviors.

6.4.2 SHARED MEMORY STRUCTURE

6.4.2.1 Decoding Control

6.4.2.1.1 Extended Decode Status (*EXTENDED_DECODE_STATUS*, R, Offset = 0x0000)

EXTENDED_DECODE_STATUS	Bit	Description	Initial State
Reserved	[31:1]	Reserved	Undef

6.4.2.1.2 Set Frame Tag (*SET_FRAME_TAG*, W, Offset = 0x0004)

SET_FRAME_TAG	Bit	Description	Initial State
SET_FRAME_TAG	[31:0]	Host sets to set a unique frame ID (e.g., application specific timestamp) for this output buffer.	Undef

6.4.2.1.3 Get Frame Tag Top (*GET_FRAME_TAG_TOP*, R, Offset = 0x0008)

GET_FRAME_TAG_TOP	Bit	Description	Initial State
GET_FRAME_TAG_TOP	[31:0]	MFC sets a frame tag for the output frame corresponding to the one set by host using SET_FRAME_TAG for this frame. Host reads a unique ID on display. This tag returns an application specific ID for the progressive frame. For an interlaced picture, this tag returns an application specific ID for the top field. Note that the value of (-1) indicates that there is no displayable picture.	Undef

6.4.2.1.4 Get Frame Tag Bottom (*GET_FRAME_TAG_BOTTOM*, R, Offset = 0x000C)

GET_FRAME_TAG_BOTTOM	Bit	Description	Initial State
GET_FRAME_TAG_BOTTOM	[31:0]	MFC sets a frame tag for the output frame corresponding to the one set by host using SET_FRAME_TAG for this frame. Host reads the ID for the bottom field. This tag is valid for the interlaced picture only. Note that the value of (-1) indicates that there is no displayable picture or a progressive picture.	Undef

6.4.2.1.5 Picture Time for Top Field (PIC_TIME_TOP, R, Offset = 0x0010)

PIC_TIME_TOP	Bit	Description	Initial State
PIC_TIME_TOP	[31:0]	<p>Presentation time of an output frame. It is determined by the header information in the bitstream. For an interlaced picture, it returns a presentation time for the top field.</p> <p>The time unit is in ms. However if the standard does not provide presentation time (e.g. H.263 and H.264), it returns a sequence number.</p>	Undef

NOTE: Specific interpretation for each standard is as follows.

1. MPEG2, H.263: Temporal reference
2. MPEG4: VOP time
3. H.264: POC
4. VC1 simple/main profile: zero
5. VC1 advanced profile: Temporal reference frame counter. If it is not present, zero is written.

6.4.2.1.6 Picture Time for Bottom Field (PIC_TIME_BOTTOM, R, Offset = 0x0014)

PIC_TIME_BOTTOM	Bit	Description	Initial State
PIC_TIME_BOTTOM	[31:0]	Presentation time of the bottom field. It is valid for the interlaced picture only. The specific interpretation for each standard is the same as PIC_TIME_TOP.	Undef

6.4.2.1.7 Start Byte Number (START_BYTE_NUM, R/W, Offset = 0x0018)

START_BYTE_NUM	Bit	Description	Initial State
START_BYTE_NUM	[31:0]	An offset of the stream when it is not aligned	Undef

6.4.2.1.8 Cropping Information One (CROP_INFO1, R, Offset = 0x0020)

CROP_INFO1	Bit	Description	Initial State
CROP_RIGHT_OFFSET	[31:16]	Cropping right offset	Undef
CROP_LEFT_OFFSET	[15:0]	Cropping left offset	Undef

6.4.2.1.9 Cropping Information Two (CROP_INFO2, R, Offset = 0x0024)

CROP_INFO2	Bit	Description	Initial State
CROP_BOTTOM_OFFSET	[31:16]	Cropping bottom offset	Undef
CROP_TOP_OFFSET	[15:0]	Cropping top offset	Undef

6.4.2.1.10 Profile info for displayed picture (DISP_PIC_PROFILE, R, Offset = 0x007C)

DISP_PIC_PROFILE	Bit	Description	Initial State
Reserved	[31:16]	Reserved	Undef
DISP_PIC_LEVEL	[15:8]	<p>Level in MPEG4 and H.264. In H.264, 31 stands for level 3.1 and 9 stands for level 1b. In MPEG4, 3 stands for level 3 and 7 stands for level 3b. In VC1, simple and main profile define.</p> <ul style="list-style-type: none"> 0 = Low 2 = Medium 4 = High <p>Advanced profile defines :</p> <ul style="list-style-type: none"> 0 = Level 0 1 = Level 1 2 = Level 2 3 = Level 3 4 = Level 4 <p>In MPEG2, levels are defined as follows:</p> <ul style="list-style-type: none"> 4 = High 6 = High 1440 8 = Main 10 = Low 	Undef
Reserved	[7:5]	Reserved	Undef
DISP_PIC_PROFILE	[4:0]	<p>MPEG4 :</p> <ul style="list-style-type: none"> 0 = SP 1 = ASP <p>H.264 :</p> <ul style="list-style-type: none"> 0 = Baseline 1 = Main 2 = High <p>H.263 :</p> <ul style="list-style-type: none"> 0 = Always (bitstream does not carry profile info) <p>VC1 :</p> <ul style="list-style-type: none"> 0 = Simple 1 = Main 2 = Advanced <p>MPEG2:</p> <ul style="list-style-type: none"> 4 = Main 5 = Simple 	Undef

6.4.2.1.11 H.264 Decoder POC_TYPE (H264_POC_TYPE, R, Offset = 0x00B8)

H264_POC_TYPE	Bit	Description	Initial State
H264_POC_TYPE	[31:0]	POC_TYPE of H.264 decoder. If POC_TYPE is 0 or 1, initial display delay is determined by MaxDpbSize in the standard. If POC_TYPE is 2, there is no initial display delay	Undef

NOTE: The actual display delay can be different because of the following reasons.

1. If there are multiple SPSes, MFC returns the POC_TYPE of the first SPS
If the first frame uses a different SPS, it will result in different initial display delay.
2. If the second IDR frame appears within MaxDpbSize, MFC has to start the DPB buffer flushing which will cause a reduced initial display delay.

6.4.2.1.12 Displayed Picture Frame Type (DISP_PIC_FRAME_TYPE, R, Offset = 0x00C0)

DISP_PIC_FRAME_TYPE	Bit	Description	Initial State
DISP_PIC_FRAME_TYPE	[31:0]	Frame type of a displayed picture 0 = Not coded frame 1 = I frame 2 = P frame 3 = B frame	Undef

6.4.2.1.13 Available Luma DPB Address (FREE_LUMA_DPB, R, Offset = 0x00C4)

FREE_LUMA_DPB	Bit	Description	Initial State
FREE_LUMA_DPB	[31:0]	Free DPB address to which host can copy a DPB. The free DPB can be used to handle not coded frames in VC1 and MPEG4 decoders.	Undef

6.4.2.1.14 Aspect Ratio Information (ASPECT_RATIO_INFO, R, Offset = 0x00C8)

ASPECT_RATIO_INFO	Bit	Description	Initial State
Reserved	[31:4]	Reserved	Undef
ASPECT_RATIO_INFO	[3:0]	1 = Square (1:1) 2 = 625 type for 4:3 picture (12:11) 3 = 525 type for 4:3 picture (10:11) 4 = 625 type stretched for 16:9 picture (16:11) 5 = 525 type stretched for 16:9 picture (40:33) 15 = Extended PAR	Undef

6.4.2.1.15 Extended Pixel Aspect Ratio (EXTENDED_PAR, R, Offset = 0x00CC)

EXTENDED_PAR	Bit	Description	Initial State
PAR_WIDTH	[31:16]	The value indicates the horizontal size of the sample aspect ratio. It is valid only if ASPECT_RATIO_INFO=15.	Undef
PAR_HEIGHT	[15:0]	The value indicates the vertical size of the sample aspect ratio. It is valid only if ASPECT_RATIO_INFO=15.	Undef

6.4.2.2 Encoding Control

6.4.2.2.1 Extended Encoder Control (EXT_ENC_CONTROL, W, Offset = 0x0028)

EXT_ENC_CONTROL	Bit	Description	Initial State
VBV_BUFFER_SIZE	[31:16]	VBV buffer size defined by host. The unit is in kilo bytes (i.e., actual buffer size = VBV_BUFFER_SIZE * 1024 bytes). Valid only when FRAME_SKIP_ENABLE=2.	Undef
ASPECT_RATIO_VUI_ENABLE	[15]	0 = Aspect ratio VUI is disabled in H.264 encoding 1 = Aspect ratio VUI is enabled in H.264 encoding	Undef
Reserved	[14:5]	Reserved	Undef
HIERARCHICAL_P_ENABLE	[4]	0 = Hierarchical P frame disable 1 = Hierarchical P frame enable	
SEQ_HEADER_CONTROL	[3]	0 = Sequence header is not generated on the first FRAME_START 1 = Sequence header is generated on both SEQ_START and the first FRAME_START	
FRAME_SKIP_ENABLE	[2:1]	0 = Frame skip is disabled. The chance of the rate overshoot will be increased when the generate bitrate is high. 1 = Frame skip is enabled using maximum buffer size defined by level. 2 = Frame skip is enabled using VBV_BUFFER_SIZE defined by HOST	Undef
HEC_ENABLE	[0]	0 = Header extension code (HEC) is disabled in the MPEG4 encoding 1 = HEC is enabled	Undef

6.4.2.2.2 Encoding Parameter Change (ENC_PARAM_CHANGE, W, Offset = 0x002C)

ENC_PARAM_CHANGE	Bit	Description	Initial State
-	[31:3]	Reserved	Undef
RC_BIT_RATE_CHANGE	[2]	0 = Normal operation 1 = Target bitrate is changed	Undef
RC_FRAME_RATE_CHANGE	[1]	0 = Normal operation 1 = Target frame rate is changed	Undef
I_PERIOD_CHANGE	[0]	0 = Normal operation 1 = GOP size is changed	Undef

6.4.2.2.3 VOP Timing (VOP_TIMING, W, Offset = 0x0030)

VOP_TIMING	Bit	Description	Initial State
VOP_TIMING_ENABLE	[31]	Enable computing vop_time_increment and modulo_time_base in MPEG4	Undef
VOP_TIME_RESOLUTION	[30:16]	Used to compute vop_time_increment and modulo_time_base in MPEG4	Undef
FRAME_DELTA	[15:0]	Used to compute vop_time_increment and modulo_time_base in MPEG4	Undef

6.4.2.2.4 Header Extension Code Period (HEC_PERIOD, W, Offset = 0x0034)

HEC_PERIOD	Bit	Description	Initial State
HEC_PERIOD	[31:0]	Insert a header extension code every HEC_PERIOD number of packets in the MPEG4 encoding	Undef

6.4.2.2.5 P Frame QP and B Frame QP (P_B_FRAME_QP, W, Offset = 0x0070)

P_B_FRAME_QP	Bit	Description	Initial State
Reserved	[31:12]	Reserved	Undef
B_FRAME_QP	[11:6]	The value is used for the B frame QP	Undef
P_FRAME_QP	[5:0]	The value is used for the P frame QP	Undef

NOTE: The frame QPs are valid only if FR_RC_EN=0 and MB_RC_EN=0

6.4.2.2.6 Aspect Ratio IDC (ASPECT_RATIO_IDC, W, Offset = 0x0074)

ASPECT_RATIO_IDC	Bit	Description	Initial State
Reserved	[31:8]	Reserved	Undef
ASPECT_RATIO_IDC	[7:0]	VUI aspect ratio IDC for H.264 encoding. The value is defined in VUI Table E-1 in the standard. It is valid only if ASPECT_RATIO_VUI_ENABLE=1.	Undef

6.4.2.2.7 Extended SAR (EXTENDED_SAR, W, Offset = 0x0078)

EXTENDED_SAR	Bit	Description	Initial State
SAR_WIDTH	[31:16]	The value indicates the horizontal size of the sample aspect ratio. It is valid only if ASPECT_RATIO_VUI_ENABLE=1.	Undef
SAR_HEIGHT	[15:0]	The value indicates the vertical size of the sample aspect ratio. It is valid only if ASPECT_RATIO_VUI_ENABLE=1.	Undef

6.4.2.2.8 New RC Bit Rate (NEW_RC_BIT_RATE, W, Offset = 0x0090)

NEW_RC_BIT_RATE	Bit	Description	Initial State
NEW_RC_BIT_RATE	[31:0]	Updated target bit rate at encoder which has the same format as RC_BIT_RATE (0xC5A8). It is valid only if RC_BIT_RATE_CHANGE=1.	Undef

6.4.2.2.9 New RC Frame Rate (NEW_RC_FRAME_RATE, W, Offset = 0x0094)

NEW_RC_FRAME_RATE	Bit	Description	Initial State
NEW_RC_FRAME_RATE	[31:0]	Updated target frame rate at encoder which has the same format as RC_FRAME_RATE (frames per second in 1000x scale). It is valid only if RC_FRAME_RATE_CHANGE=1.	Undef

6.4.2.2.10 New Intra Period (NEW_I_PERIOD, W, Offset = 0x0098)

NEW_I_PERIOD	Bit	Description	Initial State
NEW_I_PERIOD	[31:0]	Updated intra period at encoder which has the same format as I_FRM_CTRL (0xC504). It is valid only if I_PERIOD_CHANGE=1. 0 = All P frames 1 = All I frames 2 = I - P - I - P 3 = I - P - P - I N = (N-1) P frames between two I frames	Undef

6.4.2.2.11 H264 Intra Period (H264_I_PERIOD, W, Offset = 0x009C)

H264_I_PERIOD	Bit	Description	Initial State
Reserved	[31:17]	Reserved	Undef
H264_I_PERIOD_ENABLE	[16]	Enable I picture (not IDR) for H.264 encoder. This option results in open GOP.	Undef
H264_I_PERIOD_CONFIG	[15:0]	It is valid only if H264_I_PERIOD_ENABLE=1. 0 = All P frames 1 = All I frames 2 = I - P - I - P 3 = I - P - P - I N = (N-1) P frames between two I frames	Undef

NOTE: I and IDR are used differently in H.264. IDR is used for closed GOP, and I is used for open GOP. The interpretation of the MFC registers is as follows.

[H.264 encoding]

- IDR: I_FRAME, I_FRM_CTRL, NEW_I_PERIOD

- I: H264_I_PERIOD

[MPEG4 encoding]

- I: I_FRAME, I_FRM_CTRL, NEW_I_PERIOD
- There is no concept of IDR in MPEG4

6.4.2.2.12 Rate Control Configuration (RC_CONTROL_CONFIG, W, Offset = 0x00A0)

RC_CONTROL_CONFIG	Bit	Description	Initial State
Reserved	[31:1]	Reserved	Undef
RC_FIXED_TARGET_BIT	[0]	0: Disable 1: Enable	Undef

6.4.2.2.13 Input Structure Address for Batch Encoding (BATCH_INPUT_ADDR, W, Offset = 0x00A4)

BATCH_INPUT_ADDR	Bit	Description	Initial State
BATCH_INPUT_ADDR	[31:0]	Start address of the input structure for batch encoding	Undef

6.4.2.2.14 Output Structure Address for Batch Encoding (BATCH_OUTPUT_ADDR, W, Offset = 0x00A8)

BATCH_OUTPUT_ADDR	Bit	Description	Initial State
BATCH_OUTPUT_ADDR	[31:0]	Start address of the output structure for batch encoding	Undef

6.4.2.2.15 Output Structure Size for Batch Encoding (BATCH_OUTPUT_SIZE, W, Offset = 0x00AC)

BATCH_OUTPUT_SIZE	Bit	Description	Initial State
BATCH_OUTPUT_SIZE	[31:0]	Size of the output structure for batch encoding	Undef

6.4.2.2.16 Hierarchical P frame QP (HIERARCHICAL_P_QP, W, Offset = 0x00E0)

HIERARCHICAL_P_QP	Bit	Description	Initial State
Reserved	[31:18]	Reserved	Undef
T3_FRAME_QP	[17:12]	QP for T3 frames	Undef
T2_FRAME_QP	[11:6]	QP for T2 frames	Undef
T0_FRAME_QP	[5:0]	QP for T0 frames	Undef

6.4.2.3 Common Control

6.4.2.3.1 Allocated Luma DPB Size Register (**ALLOCATED_LUMA_DPB_SIZE**, W, Offset = 0x0064)

ALLOCATED_LUMA_DPB_SIZE	Bit	Description	Initial State
ALLOCATED_LUMA_DPB_SIZE	[31:0]	Size of luma DPB that host allocated for decoding. The allocated DPB size has to take into account the tile mode format.	Undef

6.4.2.3.2 Allocated Chroma DPB Size Register (**ALLOCATED_CHROMA_DPB_SIZE**, W, Offset = 0x0068)

ALLOCATED_CHROMA_DPB_SIZE	Bit	Description	Initial State
ALLOCATED_CHROMA_DPB_SIZE	[31:0]	Size of chroma DPB that host allocated for decoding. The allocated DPB size has to take into account the tile mode format.	Undef

6.4.2.3.3 Allocated Motion Vector Size Register (**ALLOCATED_MV_SIZE**, W, Offset = 0x006C)

ALLOCATED_MV_SIZE	Bit	Description	Initial State
ALLOCATED_MV_SIZE	[31:0]	Size of motion vector buffers that host allocated for decoding. The allocated DPB size has to take into account the tile mode format. This is valid for H.264 only.	Undef

6.4.2.3.4 Flush Command Type Register (**FLUSH_CMD_TYPE**, R, Offset = 0x0080)

FLUSH_CMD_TYPE	Bit	Description	Initial State
FLUSH_CMD_TYPE	[31:0]	0 = Encoder 1 = Decoder	Undef

6.4.2.3.5 Flush Command Input Buffer 1 Register (FLUSH_CMD_INBUF1, R, Offset = 0x0084)

FLUSH_CMD_INBUF1	Bit	Description	Initial State
FLUSH_CMD_INBUF1	[31:0]	11bit right-shifted address of an input buffer pointer Encoder: Current Y address Decoder: CPB buffer address	Undef

6.4.2.3.6 Flush Command Input Buffer 2 Register (FLUSH_CMD_INBUF2, R, Offset = 0x0088)

FLUSH_CMD_INBUF2	Bit	Description	Initial State
FLUSH_CMD_INBUF2	[31:0]	11bit right-shifted address of an input buffer pointer Encoder: Current C address Decoder: Descriptor buffer address	Undef

6.4.2.3.7 Flush Command Output Buffer Register (FLUSH_CMD_OUTBUF, R, Offset = 0x008C)

FLUSH_CMD_OUTBUF	Bit	Description	Initial State
FLUSH_CMD_OUTBUF	[31:0]	11bit right-shifted address of an output buffer pointer Encoder: Stream buffer start address Decoder: N/A	Undef

6.4.2.3.8 Minimum Luma DPB Size Register (MIN_LUMA_DPB_SIZE, R, Offset = 0x00B0)

MIN_LUMA_DPB_SIZE	Bit	Description	Initial State
MIN_LUMA_DPB_SIZE	[31:0]	The minimum size (in bytes) of luma DPB that host should allocate for decoding. The size takes into account the tile mode format.	Undef

6.4.2.3.9 Minimum Chroma DPB Size Register (MIN_CHROMA_DPB_SIZE, R, Offset = 0x00BC)

MIN_CHROMA_DPB_SIZE	Bit	Description	Initial State
MIN_CHROMA_DPB_SIZE	[31:0]	The minimum size (in bytes) of chroma DPB that host should allocate for decoding. The size takes into account the tile mode format.	Undef

6.4.2.3.10 Debug History Input Register (DBG_HISTORY_INPUT0, W, Offset = 0x00D0)

DBG_HISTORY_INPUT0	Bit	Description	Initial State
ALLOCATED_MEM_SIZE	[31:16]	Size of allocated memory for the stage counter history dump.	Undef
Reserved	[15:1]	Reserved	Undef
ENABLE_DEBUG_HISTORY	[0]	0: Disable debug history 1: Enable debug history	Undef

6.4.2.3.11 Debug History Input Register (DBG_HISTORY_INPUT1, W, Offset = 0x00D4)

DBG_HISTORY_INPUT1	Bit	Description	Initial State
ALLOCATED_MEM_ADDR	[31:0]	Address of allocated memory for the stage counter history dump.	Undef

6.4.2.3.12 Debug History Output Register (DBG_HISTORY_OUTPUT, R, Offset = 0x00D8)

DBG_HISTORY_OUTPUT	Bit	Description	Initial State
DBG_HISTORY_SIZE	[31:0]	Number of bytes which MFC dumps as the stage counter history.	Undef

6.4.2.4 Metadata Control

6.4.2.4.1 Metadata Enable (METADATA_ENABLE, W, Offset = 0x0038)

METADATA_ENABLE	Bit	Description	Initial State
Reserved	[31:8]	Reserved	Undef
NUM_CONCEALED_MB_ENABLE	[7]	0: Disable number of concealed macroblock info 1: Enable number of concealed macroblock info	
EXTRADATA_ENABLE	[6]	0 = Disable extra metadata 1 = Enable extra metadata	Undef
ENC_SLICE_SIZE_ENABLE	[5]	0 = Disable slice size info 1 = Enable slice size info	Undef
VUI_ENABLE	[4]	0 = Disable VUI info 1 = Enable VUI info	Undef
SEI_NAL_ENABLE	[3]	0 = Disable SEI NAL info 1 = Enable SEI NAL info	Undef
VC1_PARAM_ENABLE	[2]	0 = Disable VC1 parameter store 1 = Enable VC1 parameter store	Undef
CONCEALED_MB_ENABLE	[1]	0 = Disable concealed macroblock info 1 = Enable concealed macroblock info	Undef
QP_ENABLE	[0]	0 = Disable QP info 1 = Enable QP info	Undef

NOTE: When QP_ENABLE=1, MFC sets MFC_QP_OUT_EN and MFC_QP_OFFSET internally
METADATA_ENABLE and EXT_METADATA_START_ADDR should be set on SEQ_START

6.4.2.4.2 Metadata Status (METADATA_STATUS, R, Offset = 0x003C)

METADATA_STATUS	Bit	Description	Initial State
Reserved	[31:1]	Reserved	Undef
METADATA_STATUS	[0]	0 = Metadata does not exist 1 = Metadata exists	Undef

6.4.2.4.3 Metadata Display Index (METADATA_DISPLAY_INDEX, R, Offset = 0x0040)

METADATA_DISPLAY_INDEX	Bit	Description	Initial State
METADATA_DISPLAY_INDEX	[31:0]	When concealed macroblock or QP is enabled in the metadata info, it has to return the information of the displayed frame, not the decoded frame. It returns the index of metadata in the shared memory.	Undef

6.4.2.4.4 Metadata Start Address (EXT_METADATA_START_ADDR, W, Offset = 0x0044)

EXT_METADATA_START_ADDR	Bit	Description	Initial State
EXT_METADATA_START_ADDR	[31:0]	The start address of the metadata memory to configure the QP, concealed macroblock number, VC1 parameters, SEI, VUI, and slice size information	Undef

NOTE: The size of EXT_METADATA_START_ADDR buffer is 54words (216bytes).

METADATA_ENABLE and EXT_METADATA_START_ADDR should be set on SEQ_START

6.4.2.4.5 Put Extradata (PUT_EXTRADATA, W, Offset = 0x0048)

PUT_EXTRADATA	Bit	Description	Initial State
Reserved	[31:1]	Reserved	Undef
PUT_EXTRADATA	[0]	Host informs MFC of the existence of extra metadata for each frame decoding. Valid only if EXTRADATA_ENABLE is set. 0 = No extra metadata 1 = Extra metadata exists	Undef

6.4.2.4.6 Extradata Address Register (EXTRADATA_ADDR, W, Offset = 0x004C)

EXTRADATA_ADDR	Bit	Description	Initial State
EXTRADATA_ADDR	[31:0]	Host informs MFC of the address of extra metadata. MFC copies the extra metadata to the shared memory output. MFC will copy only if EXTRADATA_ENABLE and PUT_EXTRADATA are set	Undef

6.5 METADATA INTERFACE

Shared memory is used to exchange information between the MFC core and an external host. Host should allocate metadata input and output buffers in the shared memory, and inform MFC of the pointer through the EXT_METADATA_START_ADDR register.

The structure of the metadata buffer output is OpenMax compliant which is shown as follows:

```
typedef struct OMX_OTHER_EXTRADATATYPE {  
    OMX_U32          nSize;  
    OMX_VERSIONTYPE  nVersion;  
    OMX_U32          nPortIndex;  
    OMX_EXTRADATATYPE eType;  
    OMX_U32          nDataSize;  
    OMX_U8          data[1];  
} OMX_OTHER_EXTRADATATYPE;
```

Note that the start and end addresses should be informed through metadata buffer<n> addr and metadata buffer size.

6.5.1 SHARED MEMORY INTERFACE FOR DECODERS

MFC decoders use the shared memory to report QP, concealed macroblock information, VC1 parameters, SEI, VUI information. [Table 6-1](#) describes the payload in the shared memory, and [Figure 6-5](#) shows the shared memory input structure in more detail. The shared memory input structure should be provided to MFC on the INIT_BUFFERS command.

Table 6-1 Payload in the Shared Memory

Element	Payload
Metadata[0]	No more metadata
Metadata[1]	QP information of each decoded macroblocks
Metadata[2]	An error map of concealed macroblock information
Metadata[3]	VC1 parameters
Metadata[4]	SEI NAL information
Metadata[5]	VUI information
Metadata[6]	Number of concealed macroblocks

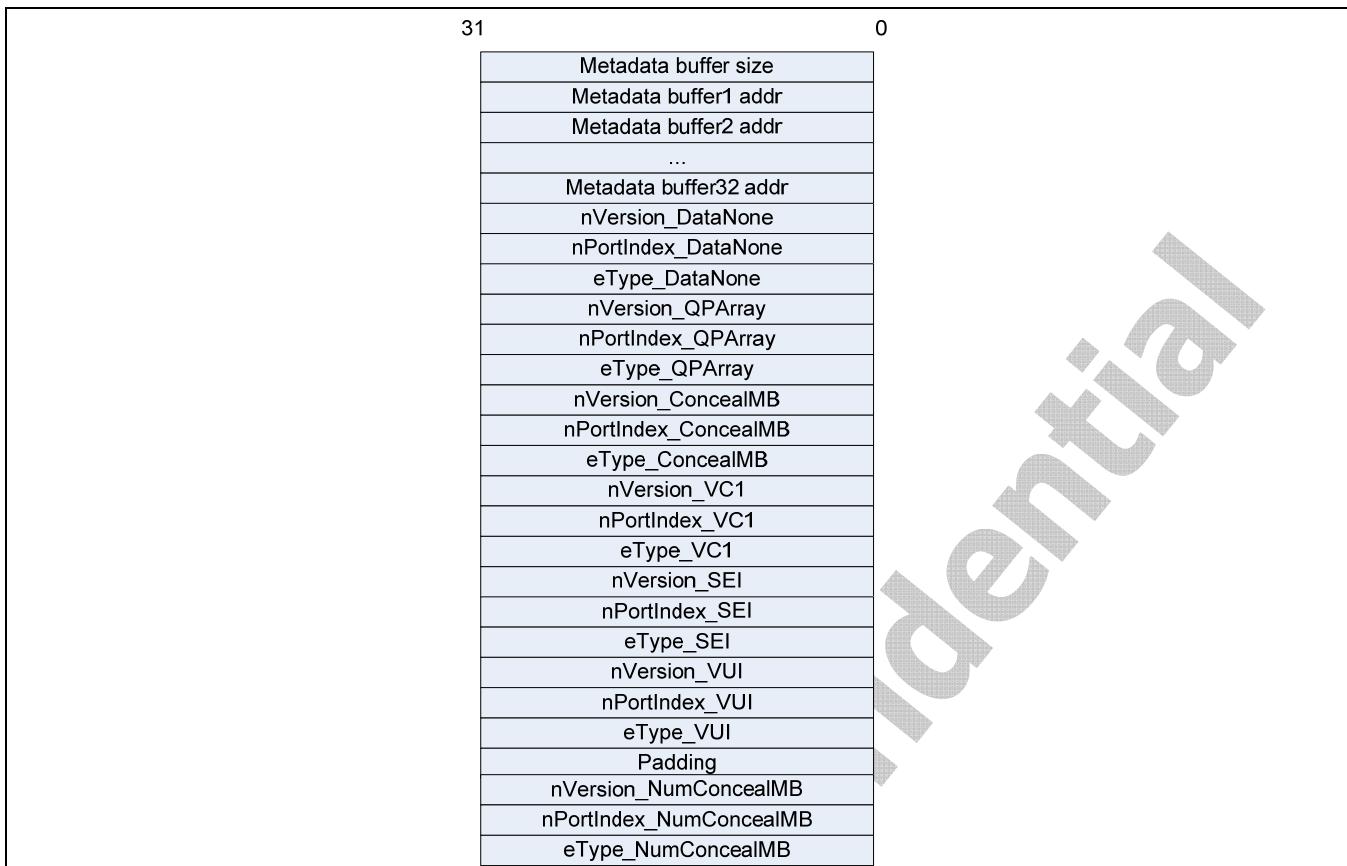


Figure 6-5 Shared Memory Input for Decoders

There is one-to-one mapping between each metadata output buffer and corresponding DPB. Hence it is host's responsibility to allocate NUM_DPB number of metadata buffers with the same size. The remaining metadata buffer addresses are not used in storing metadata. The size is communicated through the first fields in the shared memory input.

However since the VUI information is embedded in the sequence header (SPS in H.264), there is no one-to-one relationship between VUI and metadata buffers. MFC will return the appropriate VUI for a specific frame when there is an SPS change.

[Figure 6-6](#) shows the metadata output in the shared memory for each DPB. In the metadata output structure, the 20bytes header as well as the payload for each field will be present only if the corresponding bits in the METADATA_ENABLE register. However, if the metadata buffer is full, the payload will be discarded.

When the input bitstream consists of one frame worth of data and one or more extra data, it will be copied to the ExtraData metadata. In this case, the address of the extra data is informed through the EXTRADATA_ADDR register.

[Figure 6-7](#) shows more detailed data structure for the VC1 parameters. The numbers in the parenthesis specify the effective number of bits for each field.

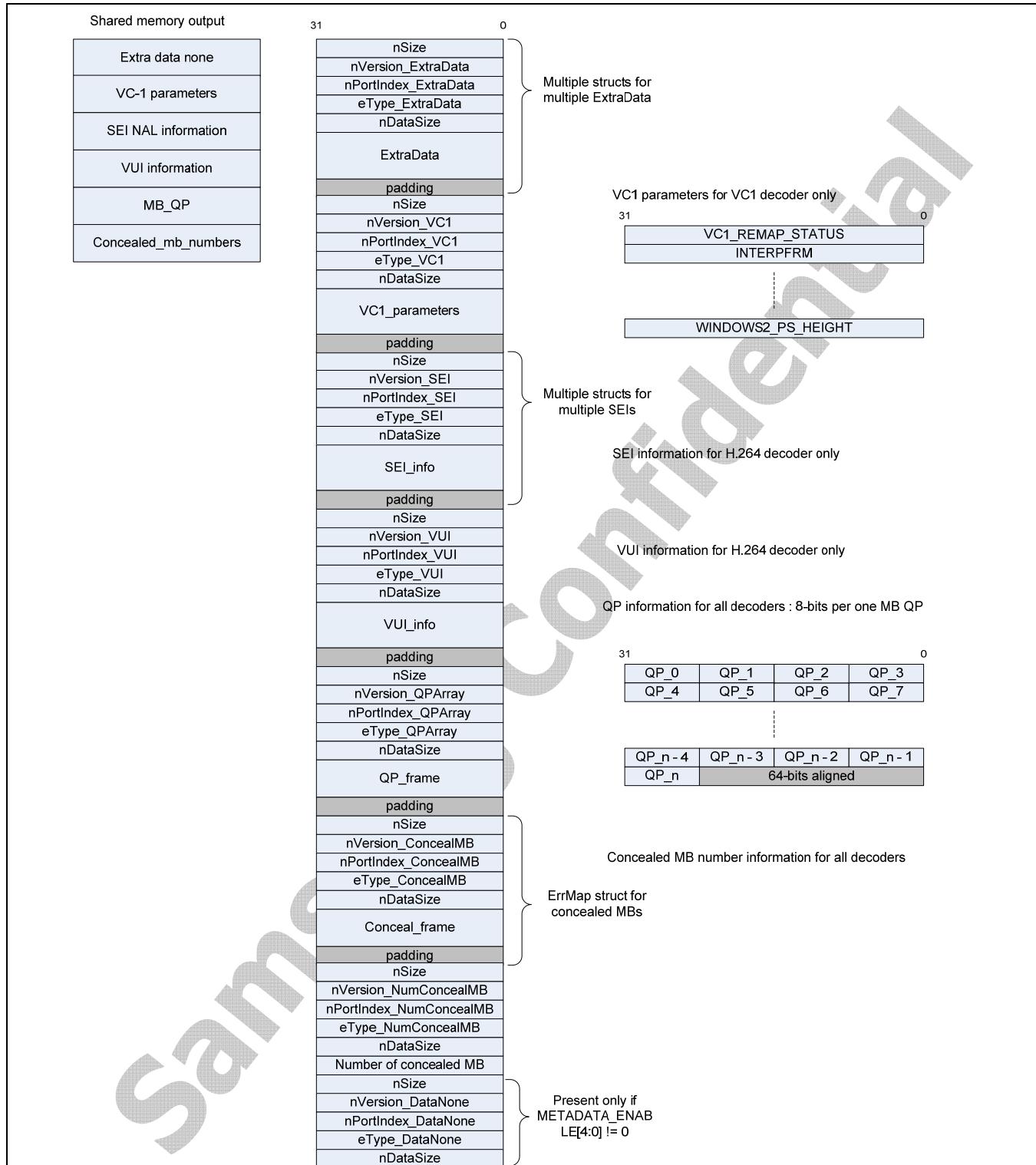


Figure 6-6 Shared Memory Output for Decoders

31	0
VC1_REMAP_STATUS (8)	
INTERPFRM (1)	
RESPIC (2)	
RPTFRM (2)	
TFF (1)	
RFF (1)	
PSF (1)	
UVSAMP (1)	
DISP_HORIZ_SIZE (14)	
DISP_VERT_SIZE (14)	
MAX_CODED_WIDTH (12)	
MAX_CODED_HEIGHT (12)	
CODED_WIDTH (12)	
CODED_HEIGHT (12)	
HORIZ_SIZE (12)	
VERT_SIZE (12)	
ASPECT_RATIO (4)	
ASPECT_WIDTH (8)	
ASPECT_HEIGHT (8)	
PANSCAN_FLAG (1)	
PS_PRESENT (1)	
NUM_PANSCAN_WINDOWS (2)	
WINDOW0_PS_HOFFSET (18)	
WINDOW0_PS_VOFFSET (18)	
WINDOW0_PS_WIDTH (14)	
WINDOW0_PS_HEIGHT (14)	
WINDOW1_PS_HOFFSET (18)	
WINDOW1_PS_VOFFSET (18)	
WINDOW1_PS_WIDTH (14)	
WINDOW1_PS_HEIGHT (14)	
WINDOW2_PS_HOFFSET (18)	
WINDOW2_PS_VOFFSET (18)	
WINDOW2_PS_WIDTH (14)	
WINDOW2_PS_HEIGHT (14)	
WINDOW3_PS_HOFFSET (18)	
WINDOW3_PS_VOFFSET (18)	
WINDOW3_PS_WIDTH (14)	
WINDOW3_PS_HEIGHT (14)	

Figure 6-7 VC1 Parameters

6.5.2 SHARED MEMORY INTERFACE FOR ENCODERS

MFC encoder uses the shared memory to report the slice information when a frame consists of multiple slices.

[Figure 6-8](#) shows the metadata input structure for encoders. The input structure is provided to MFC for each FRAME_START command so that host updates the metadata buffer address accordingly.

[Figure 6-9](#) shows the metadata output for encoders. Currently there is only one metadata field for encoders in which the slice size metadata provides the offset and length information for multiple slices.

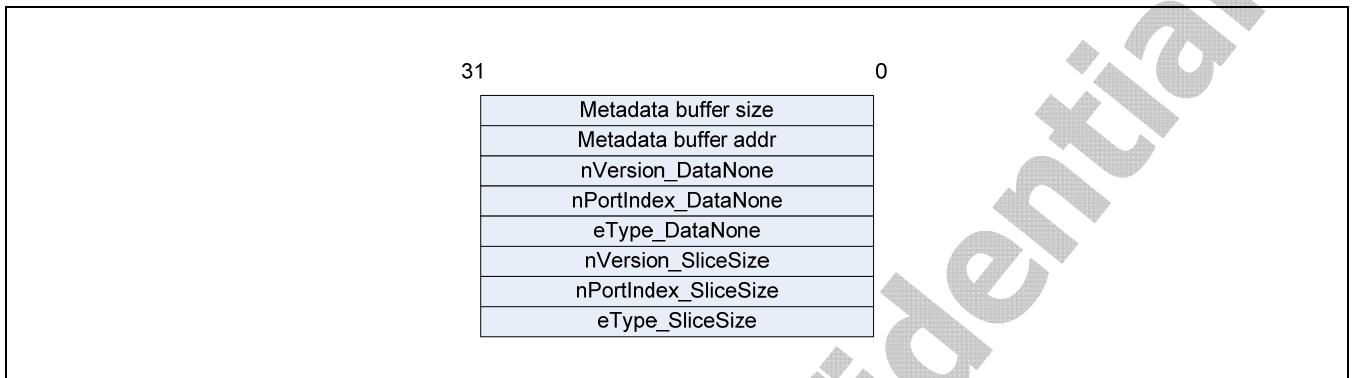


Figure 6-8 Shared Memory Input for Encoders

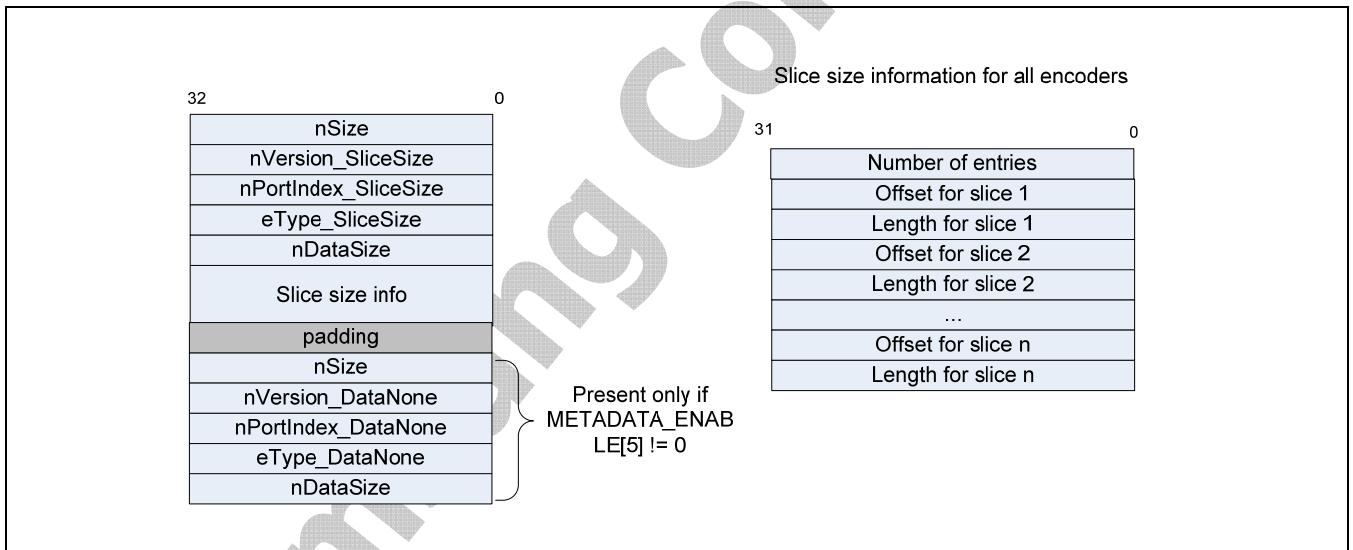


Figure 6-9 Shared Memory Output for Encoders

6.6 APPENDIX

6.6.1 SUMMARY OF BUFFER REQUIREMENTS

There are different types of buffers in MFC. Memory buffers for the firmware/host interface have different alignment and size requirements than the buffers for the firmware/hardware engine interface.

Type	Address	Size
Linear memory	Alignment: 2KB aligned Format: 11bit right shifted	See 1.3.2.4
	Example: CH_ES_ADDR, CH_DESC_ADDR, DEC_NB_DCAC, etc	
Tiled memory	Alignment: 8KB aligned Format: 11 bit right shifted	Multiple of 8KB
	Example: DEC_LUMA_x, DEC_CHROMA_x, DEC_MV_x	
Shared memory	4byte aligned	Multiple of 4bytes
	Example: HOST_WR_ADDR, EXT_METDATA_START_ADDR, etc	

NOTE:

1. MC_DRAMBASE_ADDR has to be added to compute the physical address
2. Linear memory and shared memory are set as an offset from MC_DRAMBASE_ADDR_A. Whereas tiled memory is set as an offset from MC_DRAMBASE_ADDR_A or MC_DRAMBASE_ADDR_B according to the base address index

NOTE:

6.6.2 BATCH ENCODING INTERFACE

MFC supports frame batch encoding so that host submits multiple frames in a single command and receives a response. MFC processes the multiple frames and raises an interrupt when all of the frames are encoded.

The batch encoding interface uses the shared memory in which host specifies the following information: the number of frames and a list of information for each frame. Using the information, MFC encodes multiple frames in the encoding order and generates an interrupt along with an output structure which describes details of encoded frames. Required shared memory fields are: BATCH_INPUT_ADDR, BATCH_OUTPUT_ADDR, and BATCH_OUTPUT_SIZE. The input and output structures are defined as follows:

```
typedef struct
{
    unsigned int data_id;      // FRAME_BATCH_CMD, FRAME_BATCH_RESP
    unsigned int data_size;
    unsigned int number_of_frames;

    struct frame_info_t
    {
        unsigned int cur_luma_addr;
        unsigned int cur_chroma_addr;
        unsigned int set_frame_tag;
        unsigned int vop_timing;
    } frame_info[number_of_frames];

    unsigned int number_of_stream_buffer;
    unsigned int stream_buffer_size;
    unsigned int stream_buffer_address[number_of_stream_buffer];
} input_struct;
```

Figure 6-10 Shared Memory Output for Encoders

```
typedef struct
{
    unsigned int data_id;      // FRAME_BATCH_CMD, FRAME_BATCH_RESP
    unsigned int data_size;    // data size that firmware generated

    unsigned int number_of_frames;
    struct stream_info_t
    {
```

```

        unsigned int stream_info_id;
        unsigned int stream_info_size;
        unsigned int cur_luma_addr;
        unsigned int cur_chroma_addr;
        unsigned int pic_count;
        unsigned int slice_type;
        unsigned int get_frame_tag;
        unsigned int size_of_encoded_streams;
        unsigned int number_of_stream_buffers;
        unsigned int stream_address[number_of_stream_buffers];
    } stream_info[number_of_frames];

    unsigned int number_of_frames_accepted;
    unsigned int number_of_stream_buffers_unused; // unused output buffers
    unsigned int stream_address_unused[number_of_stream_buffers_unused];
}

} output_struct;

```

Figure 6-11 Shared Memory Output for Encoders

When host invokes FRAME_BATCH_START, MFC first reads an input structure from the address pointed by BATCH_ININPUT_ADDR. The fields in the input_struct are as follows:

- data_id: FRAME_BATCH_CMD or FRAME_BATCH_RESP
- data_size: The size of input_struct which also includes the size of data_id
- number_of_frames: The number of frames that host wants to encode
- cur_luma_addr: The address of input luma picture that corresponds to CURRENT_Y_ADDR
- cur_chroma_addr: The address of input chroma picture that corresponds to CURRENT_C_ADDR
- set_frame_tag: A frame tag of an input frame. The same definition as SET_FRAME_TAG
- vop_timing: VOP timing of an input frame
- number_of_stream_buffer: The number of output buffers of encoded frames
- stream_buffer_size: The size of an output buffer of encoded frame
- stream_buffer_address: The address of output buffers of encoded frames
-

When MFC completes frame encoding, it generates an output_struct structure which is stored at BATCH_OUTPUT_ADDR with the size of BATCH_OUTPUT_SIZE. The structure provides the details of the encoded bitstreams:

- data_id: FRAME_BATCH_CMD or FRAME_BATCH_RESP
- data_size: The size of output_struct which also includes the size of data_id
- number_of_frames: The number of frames that host received for encoding

- number_of_frames_accepted: The number of frames that host accepted to complete encoding
- cur_luma_addr: The address of input luma picture that has been encoded
- cur_chroma_addr: The address of input chroma picture that has been encoded
- pic_count: A picture count in display order. pic_count provides information when there is reordering, since MFC encodes input frames in decoding order, not in display order.
- slice_type: The same definition as ENC_SLICE_TYPE
- get_frame_tag: A frame tag of an output frame. The same definition as GET_FRAME_TAG
- size_of_encoded_streams: Size of an encoded bit stream
- number_of_stream_buffers: The number of encoded stream buffers. If the size of generated stream is greater than stream_buffer_size in input_struct, more than one stream buffer will be required
- stream_address: The address of encoded stream buffers
- number_of_stream_buffers_unused: The number of unused stream buffers after an encoding batch
- stream_address_unused: The address of unused stream buffers after an encoding batch

Note that MFC stops encoding when either stream buffer is full or output_struct is full. In either case, number_of_frames_accepted will be smaller than number_of_frames. Host may resubmit the last frames from the previous input_struct as many as unaccepted in output_struct. The resubmitted frames will be encoded continuously.

When number_of_stream_buffer is greater than number_of_frames, it is possible that some of the stream buffers are not used. MFC returns them over number_of_stream_buffers_unused and stream_address_unused. Host may reuse them in the next batch.

7

TVOUT & VIDEO DAC

7.1 OVERVIEW OF TVOUT AND VIDEO DAC

The TVOUT module supports ITU-R BT.470 and EIA-770 compliant analog TV signals with 1 channel 10-bit DAC. The signal format is CVBS. It also supports EIA-608 compliant closed caption and extended data service, IEC61880 / ITU-R BT.1119 compliant wide screen signaling, and EIA-J CPR1204-1 compliant analog copy generation management system.

7.2 KEY FEATURES OF TVOUT AND VIDEO DAC

The TVOUT module includes following features:

7.2.1 I/O AND CONTROL

ITU-R BT.601 (YCbCr 4:4:4) input format

10-Bit, 4X over sampled CVBS output data to 1-channel 54 MHz DAC

7.2.2 VIDEO STANDARD COMPLIANCES FOR CVBS:

(M) NTSC, NTSC-J

(B/D/G/H/I) PAL, (M) PAL, (N) PAL, (Nc) PAL

PAL-60, NTSC4.43

7.2.3 ANCILLARY DATA INSERTION

EIA-608 compliant Closed Caption(CC) and Extended Data Service(XDS)

IEC61880 / ITU-R BT.1119 compliant Wide Screen Signaling(WSS)

EIA-J CPR1204-1 compliant analog copy generation management system(CGMS-A)

7.2.4 POST PROCESSING

Color Compensation for Invalid RGB Data

Programmable 23-Tap Luma/ Chroma Filters for Luma/ Chroma anti-aliasing for CVBS

Programmable 95-Tap oversampling filter capable of frequency response compensation

7.3 DATA FLOW

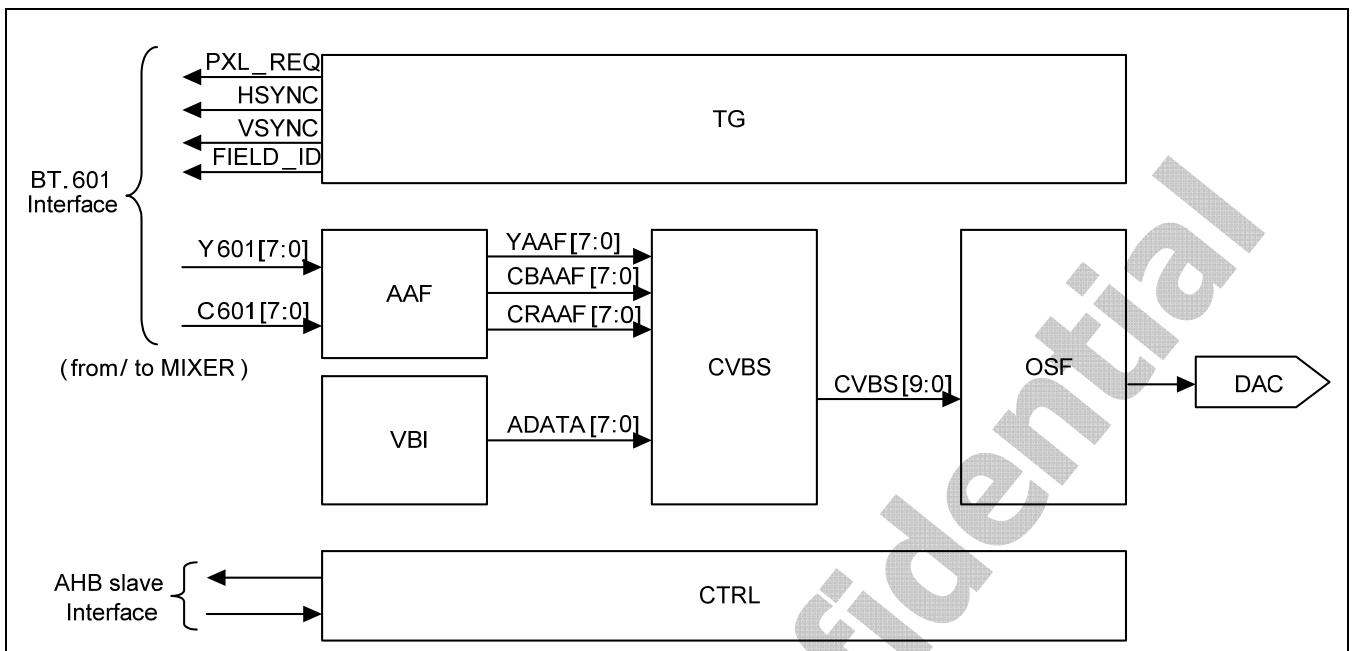


Figure 7-1 Data Flow of TVOUT Module

The TVOUT module is composed of the following data sub-modules:

- TG: Timing Generation
- CVBS: Waveform Generation, QAM Modulation, and YCbCr Video Processing
- AAF: Affine Transformation, luma/ chroma Anti-Aliasing Filter
- OSF: 4x Oversampling Filter
- VBI: Ancillary data insertion during vertical blanking interval
- CTRL: Register Control

NOTE: Image Mixer is directly connected with TVOUT and HDMI. Thus, to complete the connection, you must configure REG_DST_SEL in MIXER_CFG register.

7.4 TIMING GENERATION (TG MODULE)

The Timing Generation (TG) sub-module generates all the timing information signals required for ITU-R BT.470 compliant TV signals. An internal pixel counter generates horizontal and vertical timing signals and an internal Discrete Time Oscillator (DTO) makes the phase signals of sub-carrier.

There are two kinds of horizontal and vertical timings in TG module, namely:

7.4.1 525/60 Hz

Video standard: NTSC (M), NTSC-J, PAL (M), NTSC 4.43, and PAL 60

Horizontal frequency (FH): 15.734 kHz, 858 samples per line @ 13.5 MHz sample rate

Vertical frequency (FV): 59.94 Hz, 525 lines per frame

7.4.2 625/50 Hz

Video standard: PAL (BGHID), PAL (N), and PAL (Nc)

Horizontal frequency (FH): 15.625 kHz, 864 samples per line @ 13.5 MHz sample rate

Vertical frequency (FV): 50.00 Hz, 625 lines per frame

There are four kinds of discrete timing oscillation for sub-carrier generation in TG module, namely:

7.4.3 3.579545 MHz

Video standard: NTSC (M) and NTSC-J

Sub-carrier frequency (FSC): $910/4 * FH$

7.4.4 4.43361875 MHz

Video standard: PAL (BGHID), PAL (N), NTSC 4.43, and PAL 60

Sub-carrier frequency (FSC): $(1135/4 + 1/625) * FH$

7.4.5 3.57561149 MHz

Video standard: PAL (M)

Sub-carrier frequency (FSC): $909/4 * FH$

7.4.6 3.58205625 MHz

Video standard: PAL (Nc)

Sub-carrier frequency (FSC): $(917/4 + 1/625) *$

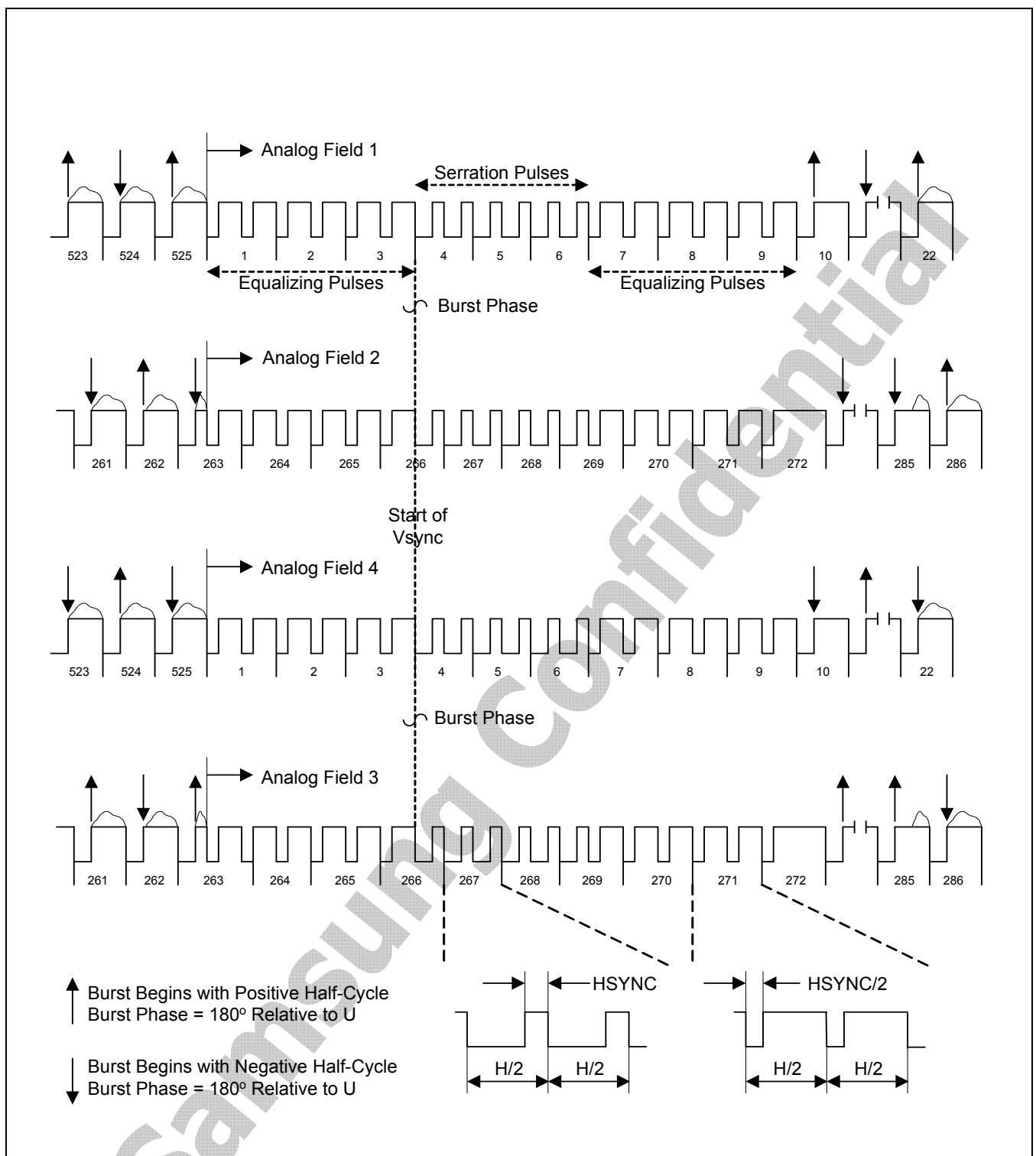


Figure 7-2 Four Field NTSC (M) Sequence and Burst Blanking

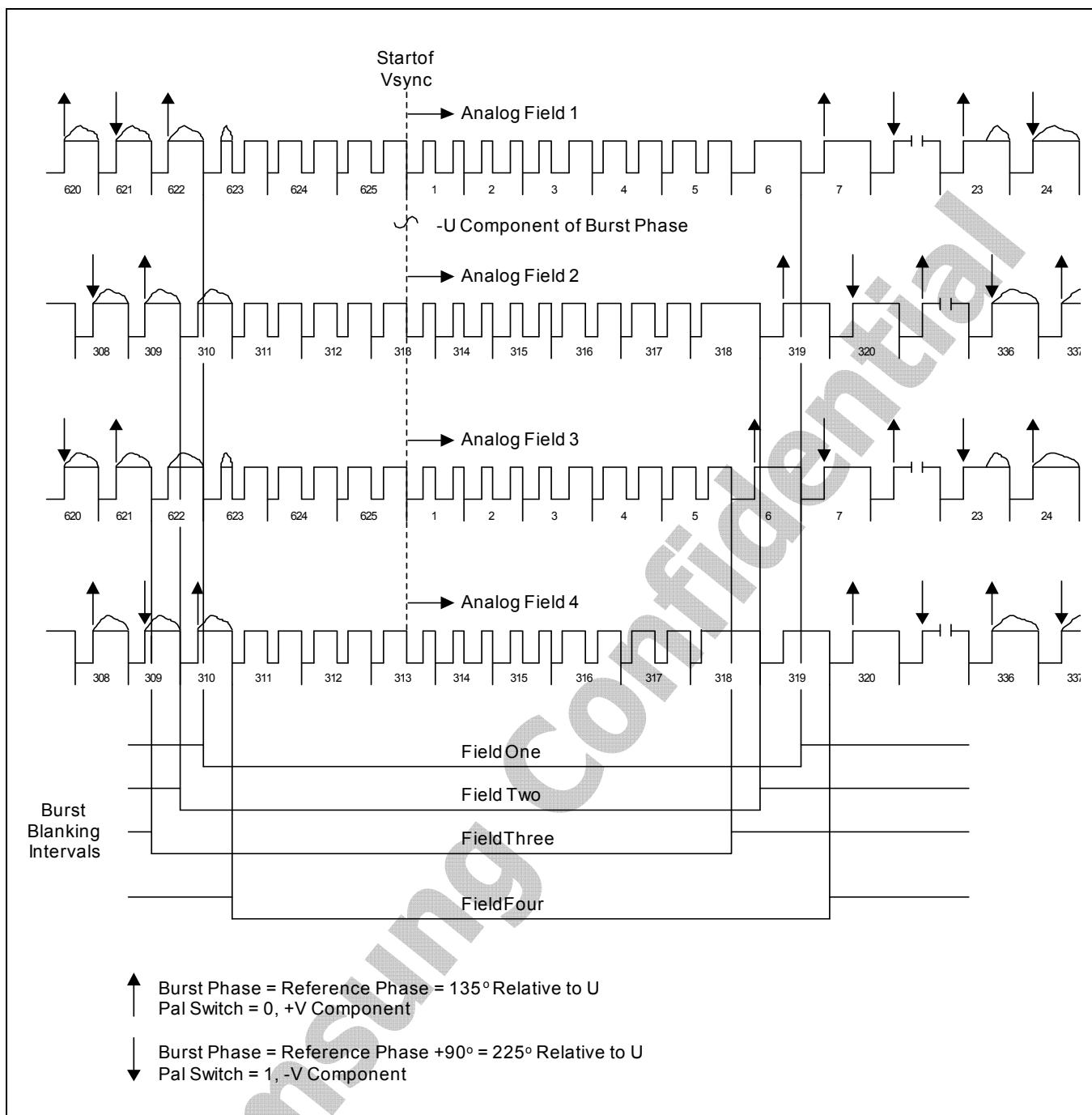


Figure 7-3 Field PAL (BGHIDNc) Sequence and Burst Blanking

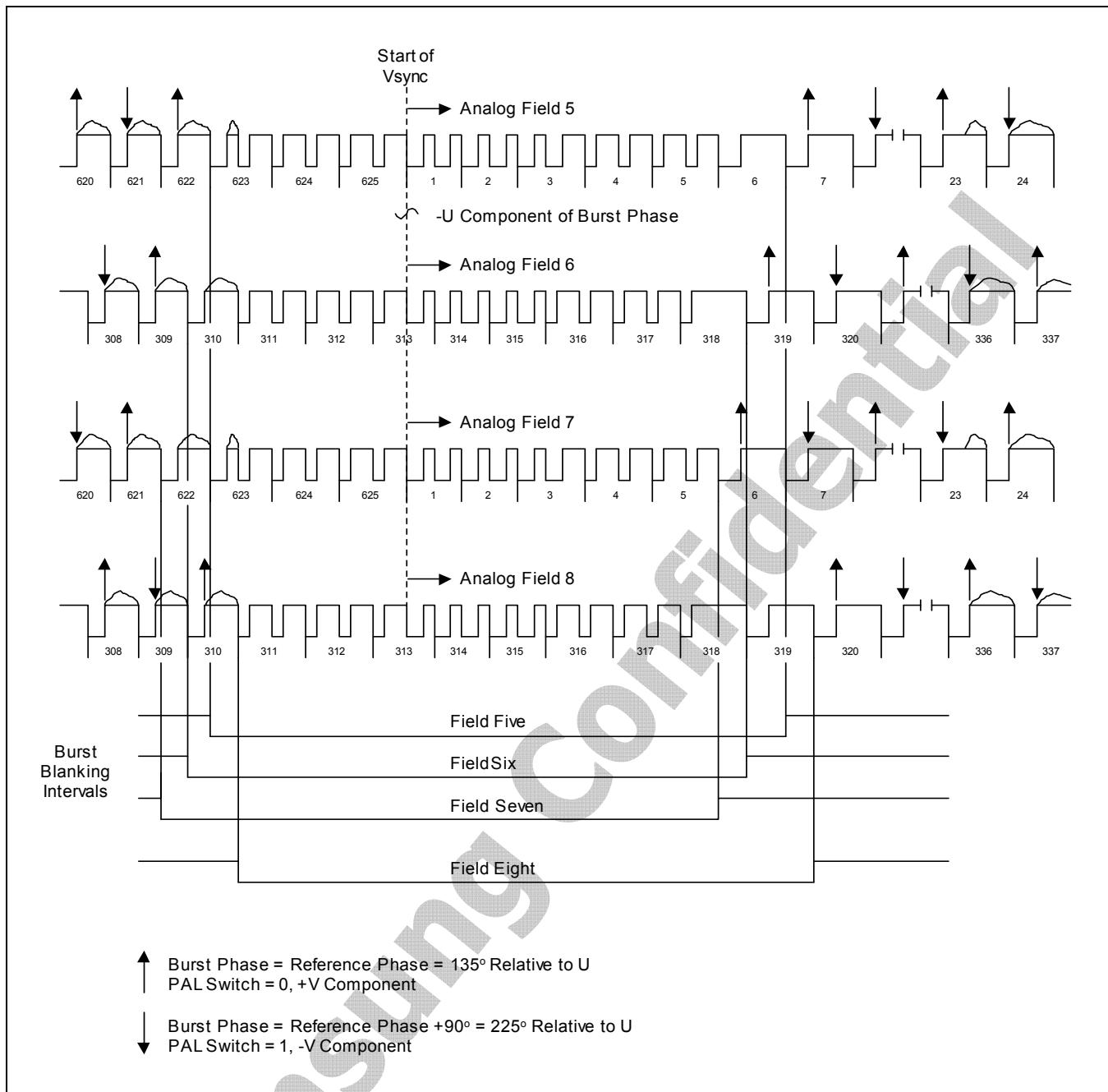


Figure 7-4 Eight Field PAL (BGHIDNc) Sequence and Burst Blanking

The internal pixel rate is 13.5 MHz which is 1/4 times of 54 MHz video clock which is used for DAC. With 13.5 MHz pixel rate, the horizontal blanking timing and active video timing are defined as follows:

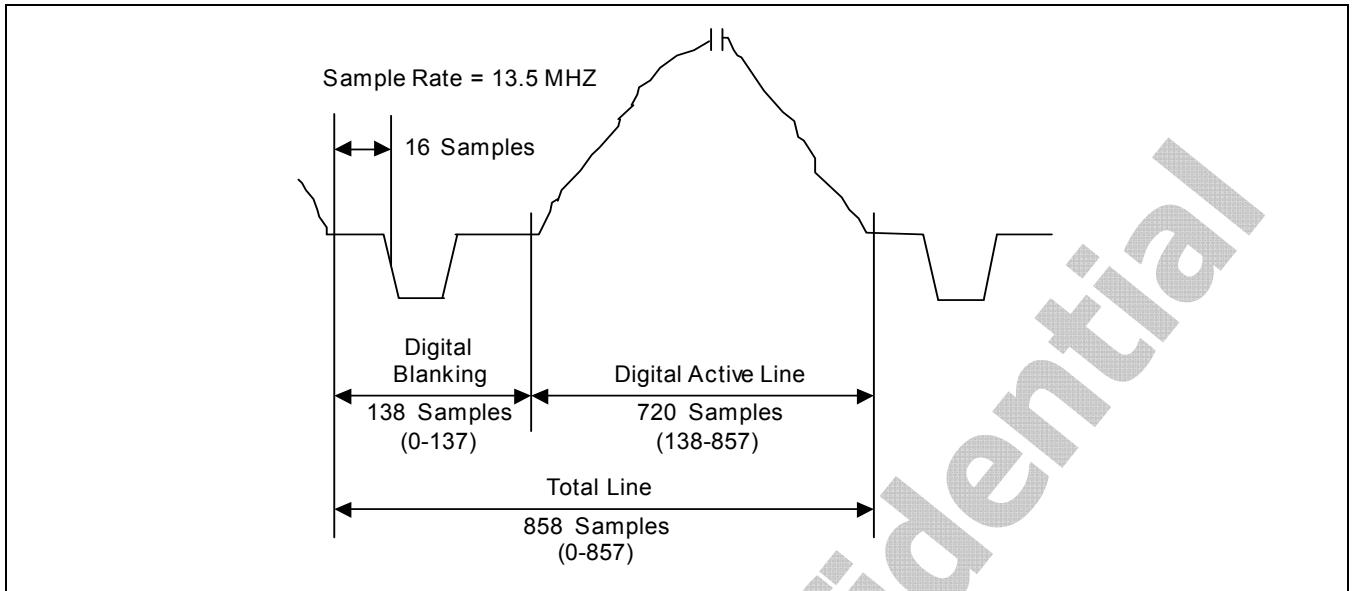


Figure 7-5 Horizontal Blanking and Active Video Timing @ 525/60 Hz

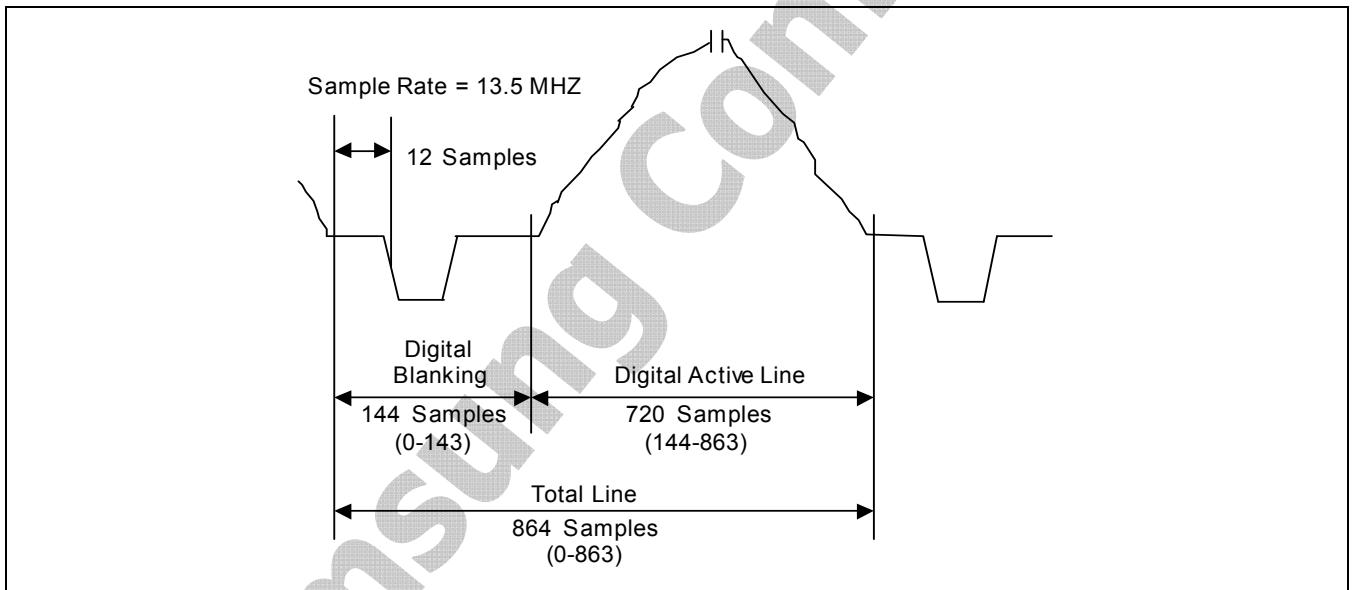


Figure 7-6 Horizontal Blanking and Active Video Timing @ 625/50 Hz

VIDEO [3:0] bits in SDO_CONFIG register controls all the timing signals in TG module. Finally note that all the internal counters and discrete time oscillators operate with 54 MHz video clock. The hsync, vsync, field_id, and data request signals are generated and used internally at 54 MHz rate. If they are delivered to 'DispPipe' module their timings are re-synchronized with 135 MHz system clock.

7.5 ANTI ALIASING FILTER (AAF MODULE)

The color TV signals are composed of luminance video, chrominance video and audio data which are modulated by different sub-carriers. Hence, some spectral shaping to avoid the aliasing between them is required. ITU-R BT.407 recommends that the luminance data (Y) should be band-limited within 4.2, 5.0, 5.5, or 6.0 MHz with respect to video standards to avoid the crosstalk between audio sub-carrier and the luminance data. It also recommends that the chrominance data (Cb and Cr) should be band-limited within 1.3 MHz to avoid the crosstalk between chrominance sub-carrier and the luminance data. If audio multiplexing is not required, band-limiting of luminance data (Y) may be skipped. Similarly, band-limiting of chrominance data (C) may be skipped at S-video encoding rather than CVBS encoding.

The AAF sub-module in TVOUT provides 23-tap linear FIR filters for spectral shaping of luminance data (Y) and chrominance data (Cb and Cr). The filter responses are fully programmable since filter coefficients themselves are controllable. Since linear FIR filters have symmetric coefficients, 11 coefficients completely defines the filter responses. The registers SDO_Y0, SDO_Y1... and SDO_Y11 are used to control the luminance filter response. The register SDO_CB0, SDO_CB1... SDO_CB11, and SDO_CR0, SDO_CR1... SDO_CR11 are used to control the chrominance Cb and Cr filter responses, respectively.

The [Table 7-1](#) show typical settings of filter coefficients. Note that there is no filtering on luminance (Y) data. Since many CE devices usually do not output multiplexed audio, filtering on luminance (Y) data is not required. Filter is not normalized and the dc gain of the filter may vary with the video scale. Refer to 3.1.5 SDO Video Scale Configuration Register and 3.1.10 ~ 3.1.45 SDO Anti Aliasing Filter Coefficients.

Table 7-1 Filter Coefficients of Anti-aliasing Filters for Luminance Y

Register	7.5 IRE Setup/ 7:3 Sync	7.5 IRE Setup/ 10:4 Sync	0 IRE Setup/ 7:3 Sync	0 IRE Setup/ 10:4 Sync
SDO_Y0	0	0	0	0
SDO_Y1	0	0	0	0
SDO_Y2	0	0	0	0
SDO_Y3	0	0	0	0
SDO_Y4	0	0	0	0
SDO_Y5	0	0	0	0
SDO_Y6	0	0	0	0
SDO_Y7	0	0	0	0
SDO_Y8	0	0	0	0
SDO_Y9	0	0	0	0
SDO_Y10	0	0	0	0
SDO_Y11	252	25D	281	28F

Table 7-2 Filter Coefficients of Anti-aliasing Filters for Chrominance Cb

Register	7.5 IRE Setup/ 7:3 Sync	7.5 IRE Setup/ 10:4 Sync	0 IRE Setup/ 7:3 Sync	0 IRE Setup/ 10:4 Sync
SDO_CB0	0	0	0	0
SDO_CB1	0	0	0	0
SDO_CB2	0	0	0	0
SDO_CB3	0	0	0	0
SDO_CB4	0	0	0	0
SDO_CB5	1	1	1	1
SDO_CB6	6	7	7	7
SDO_CB7	13	14	15	15
SDO_CB8	28	28	2A	2B
SDO_CB9	3F	3F	44	45
SDO_CB10	51	52	57	59
SDO_CB11	56	5A	5F	61

Table 7-3 Filter Coefficients of Anti-aliasing Filters for Chrominance Cr

Register	7.5 IRE Setup/ 7:3 Sync	7.5 IRE Setup/ 10:4 Sync	0 IRE Setup/ 7:3 Sync	0 IRE Setup/ 10:4 Sync
SDO_CR0	0	0	0	0
SDO_CR1	0	0	0	0
SDO_CR2	0	0	0	0
SDO_CR3	0	0	0	0
SDO_CR4	0	0	0	0
SDO_CR5	2	1	2	2
SDO_CR6	5	9	A	A
SDO_CR7	18	1C	1D	1E
SDO_CR8	37	39	3C	3D
SDO_CR9	5A	5A	5F	61
SDO_CR10	76	74	7B	7A
SDO_CR11	7E	7E	86	8F

The following figures show the magnitude and phase responses of CB and CR anti aliasing filters with the above settings. Note that these filters are applied only to CVBS.

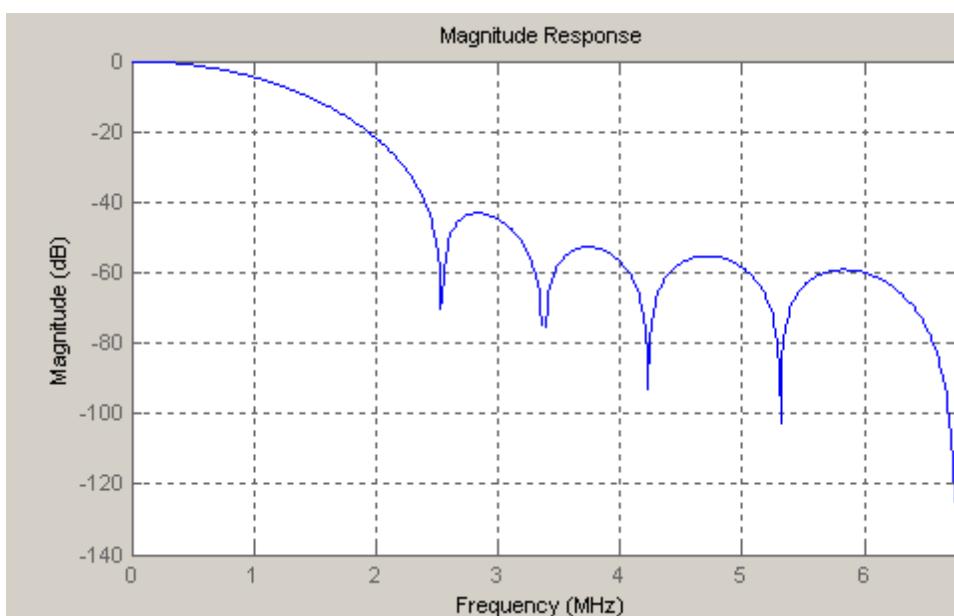


Figure 7-7 Magnitude Response of CB and CR Anti Aliasing Filter @ 13.5 MHz Sampling Rate

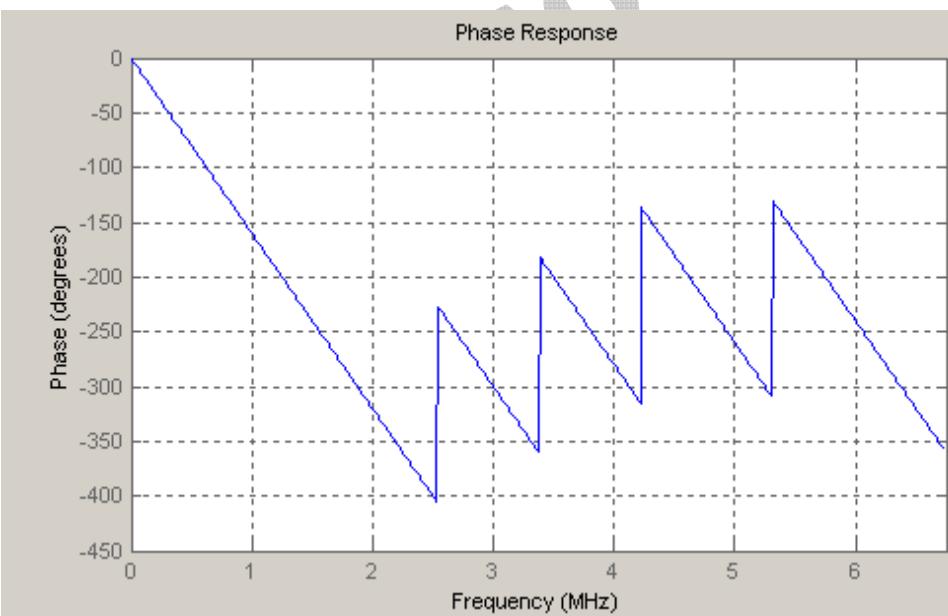


Figure 7-8 Phase Response of CB and CR Anti Aliasing Filter @ 13.5 MHz Sampling Rate

7.6 ANCILLARY DATA INSERTION (VBI MODULE)

The ITU-R BT.470 compliant TV signals include blanking lines, which do not possess video data. Some ancillary data such as closed caption, content information, display aspect ratio control, and copy control information are delivered within these blanking periods. The TVOUT module supports EIA-608 compliant closed caption (CC) and extended data service (XDS) and IEC61880 / ITU-R BT.1119 compliant wide screen signaling (WSS). The VBI sub-module draws the waveform of the ancillary data delivering signals.

The physical waveform of EIA-608 closed caption and extended data service signals is as follows:

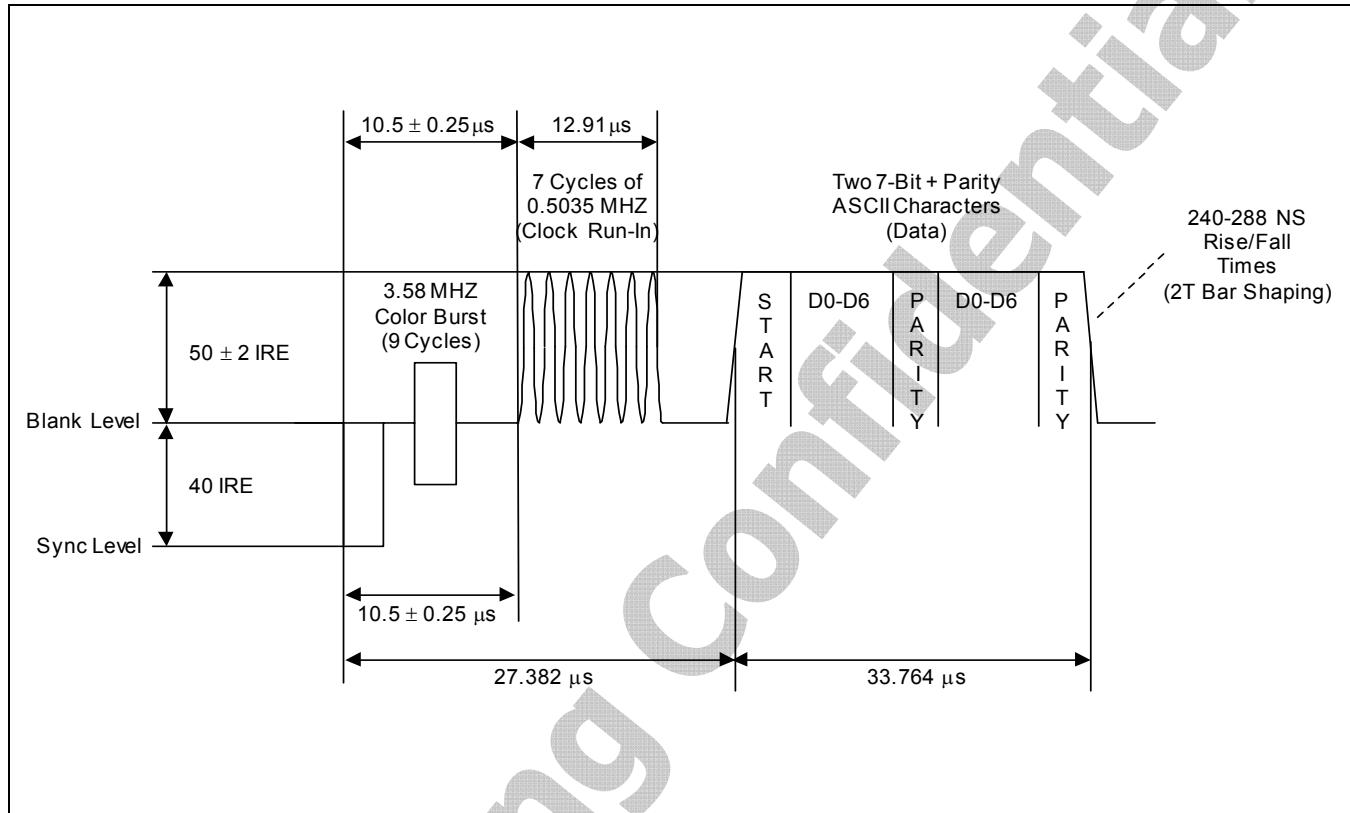


Figure 7-9 EIA-608 Closed Caption and Extended Data Service

This waveform is inserted at the 21'st line of 525/60 standard video. The register SDO_ARMCC is used for carrying the 2 byte data into the waveform including the parity bits. For the CC data and XDS data packet format and its usage, refer to the Recommendation EIA-608, "Recommended Practice for Line 21 Data Service".

The physical waveform of IEC 61880 compliant wide screen signalling (WSS) signals is as follows:

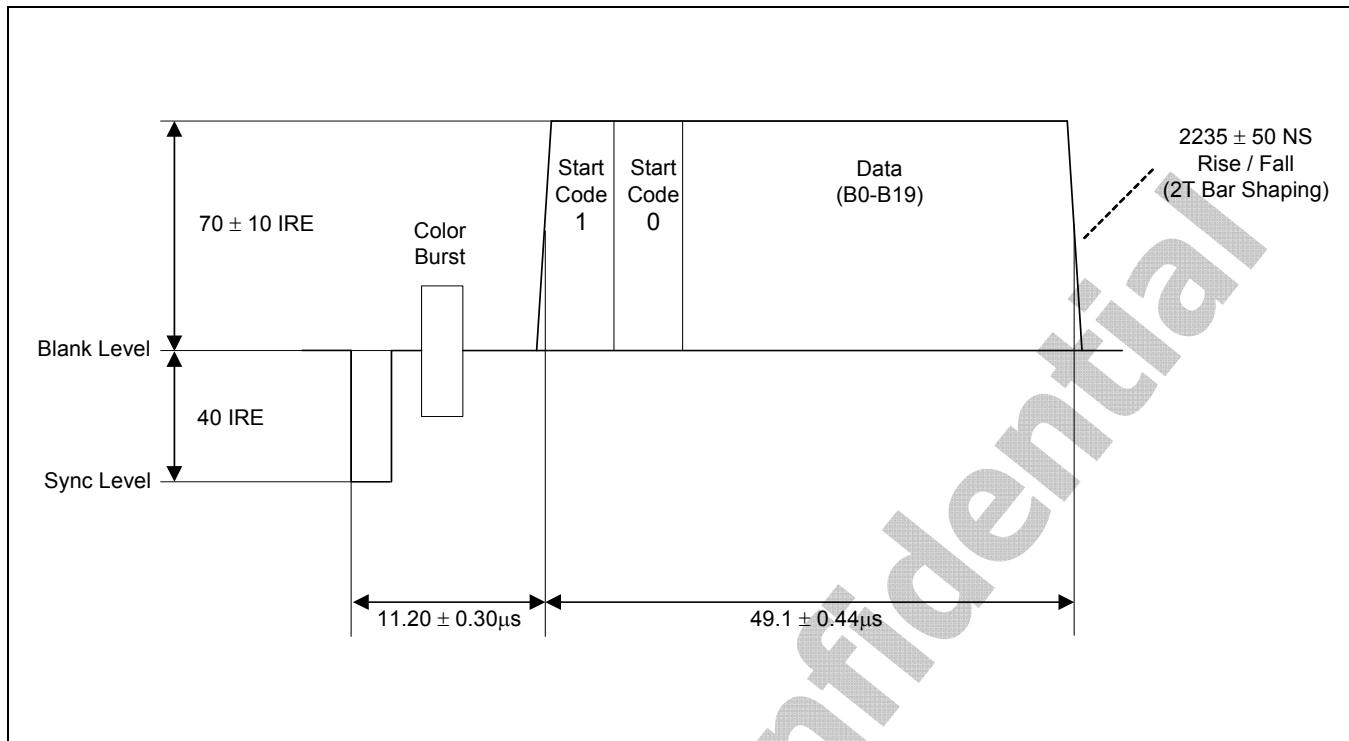


Figure 7-10 IEC 61880 Wide Screen Signaling

This waveform is inserted at the 20-th line of 525/60 standard video. The register SDO_ARMWSS525 is used to carry the 20 bit data into the waveform. Bits {b1, b0} define display aspect ratio control, bit {b7, b6, b5, b4, b3, b2} define copy control information, and bits {b13, b12, b11, b10, b9, b8} is used to specify the reserved signals. Bits {b19, b18, b17, b16, b15, b14} are used for CRC error check.

The physical waveform of ITU-R BT.1119 compliant Wide Screen Signalling (WSS) signals is as follows:

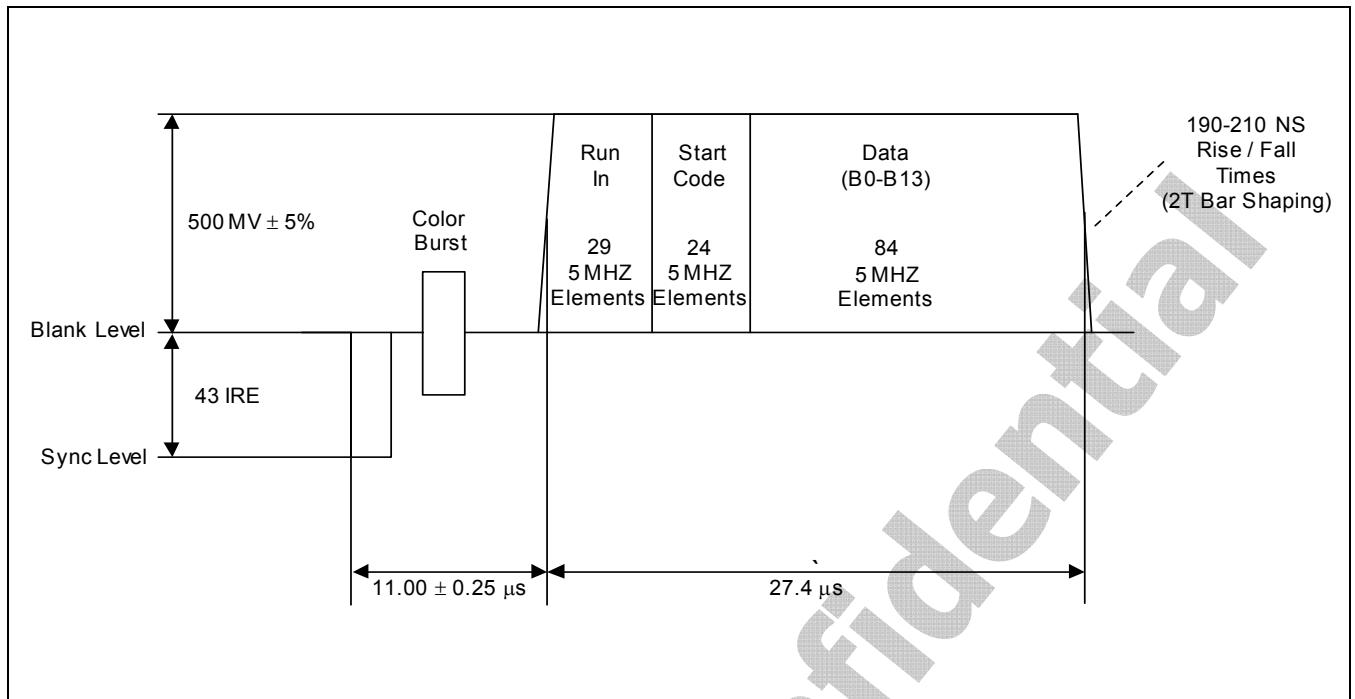


Figure 7-11 ITU-R BT.1119 Wide Screen Signaling

This waveform is inserted at the 23rd line of 625/50 standard video. The register SDO_ARMWSS625 is used for carrying the 14-bit data into the waveform. Bits {b3, b2, b1, b0} specify display aspect ratio control, bits {b7, b6, b5, b4} are used for enhanced TV service, bits {b10, b9, b8} is used for European teletext subtitle control, and bits {b13, b12, b11} are used for copy control.

7.7 WAVEFORM GENERATION AND CHROMA MODULATION (CVBS MODULE)

The CVBS sub-module combines the timing information and video data to generate the waveforms of ITU-R BT 470 TV signals. Two different data paths do this procedure, namely, one is for luminance data (Y) and the other is for chrominance data (C). For the luminance data path, horizontal/ vertical synchronization pulses are formed and merged to a properly scaled and offset luminance video data (Y). For the chrominance data paths, base band chrominance data (C_b and C_r) are modulated with a sub-carrier FSC along with the video standard, that is:

$$C(n) = U(n) * \sin(2\pi FSC * n) + V(n) * \cos(2\pi FSC * n),$$

where $U(n)$ and $V(n)$ denote properly scaled and offset versions of $C_b(n)$ and $C_r(n)$, respectively. Then a pilot sinusoidal waveform, called a burst, is formed and added prior to the start of modulated chrominance data C at each line. At the end of CVBS sub-module data paths, the luminance data (Y) and the chrominance data (C) is merged into one channel and form composite data (CVBS).

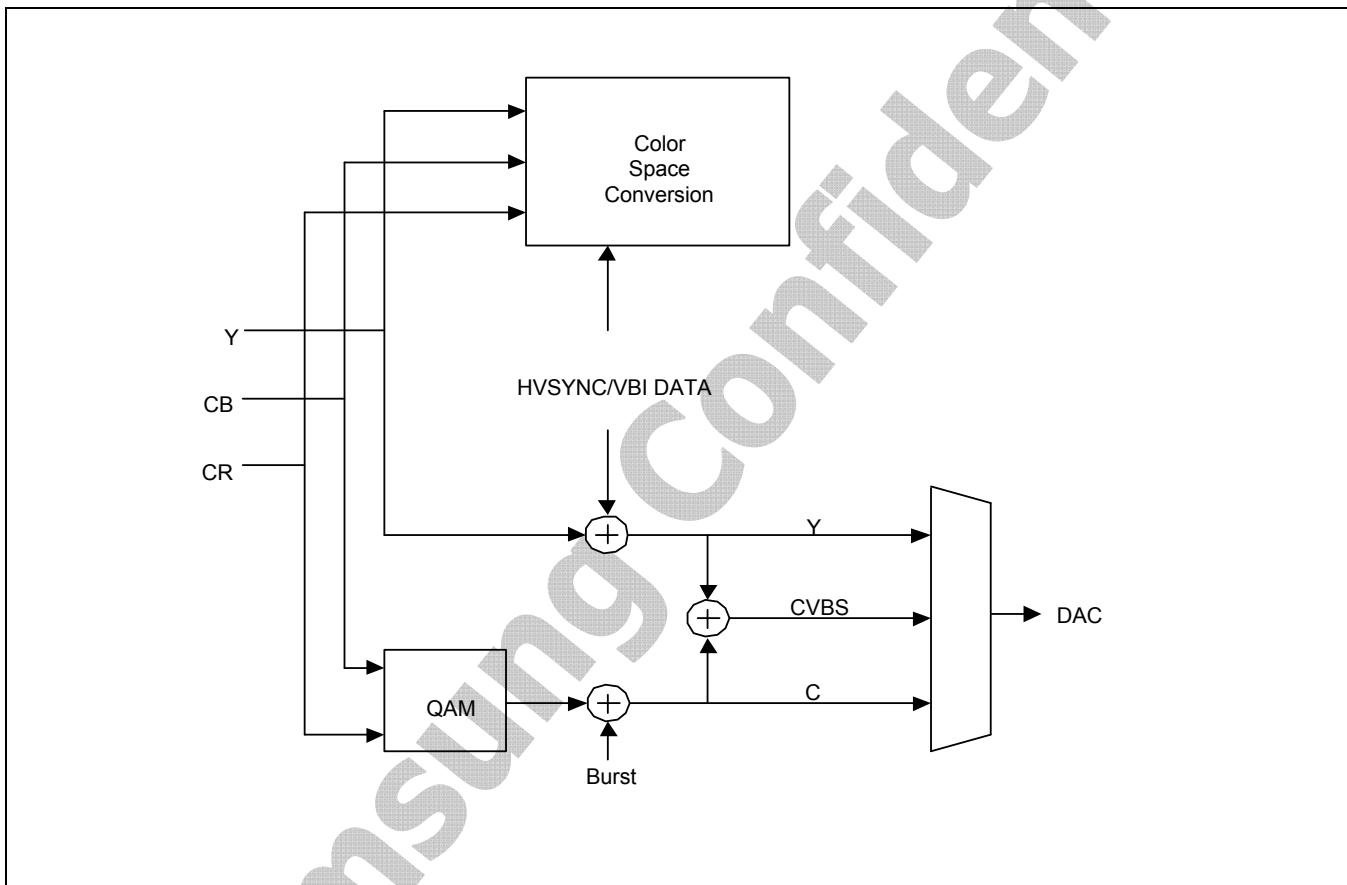


Figure 7-12 Data Flow of CVBS Sub-Module

The [Figure 7-13](#) and [Figure 7-14](#) depict the typical CVBS waveforms generated. Figure show that the setup level and the ratios of video scale to sync depth are different according to video standards. This is same in YPbPr/RGB outputs. The setup level and video-to-sync ratio are controlled by CSETUP, CSYNC, VSETUP and VSYNC bits in SDO_SCALE registers. Note that the configuration of setup level and video-to-sync ratio in our implementation are set regardless of video standards and output format.

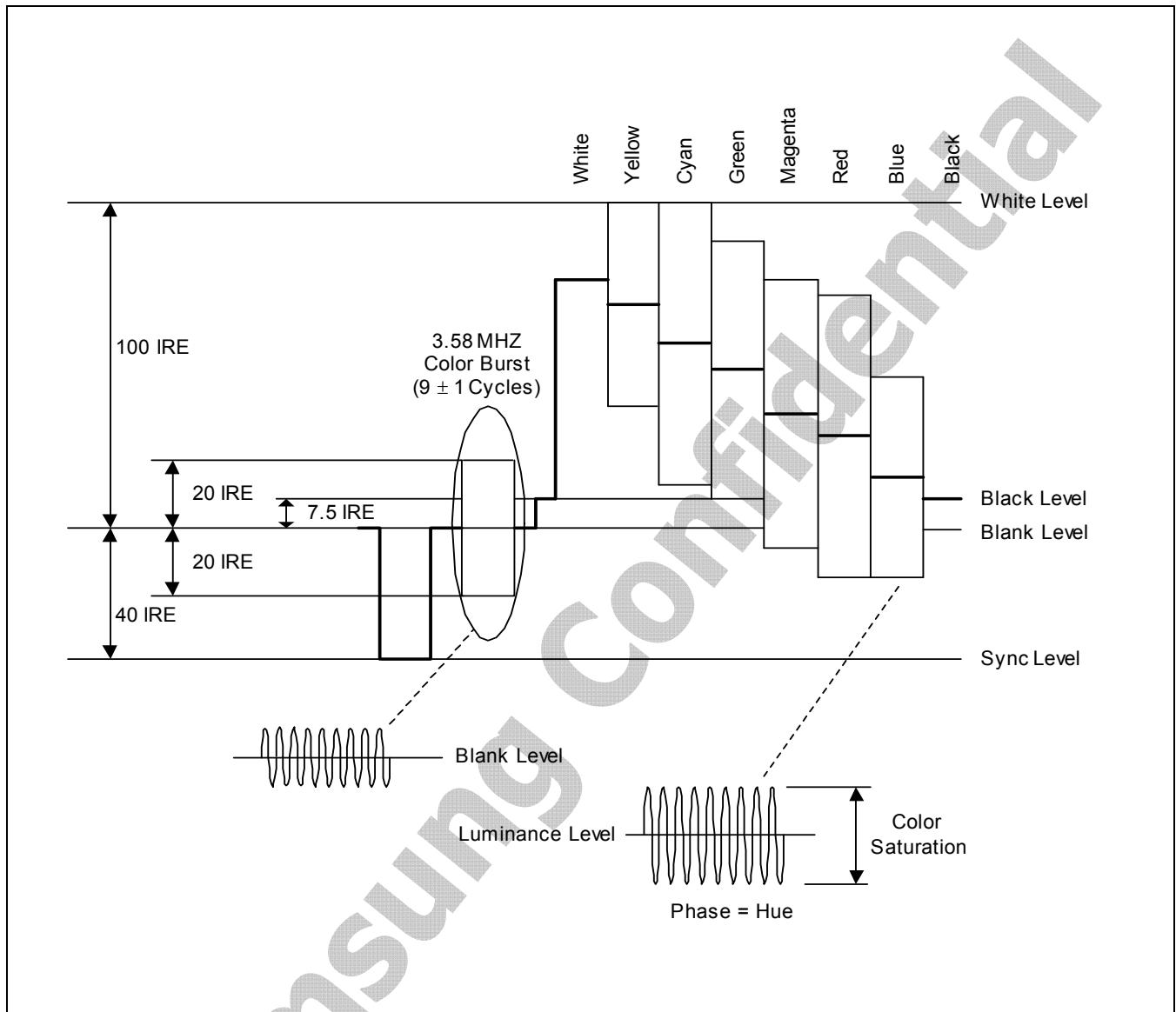


Figure 7-13 NTSC (M) Composite Video Signal with 75% Color Bars

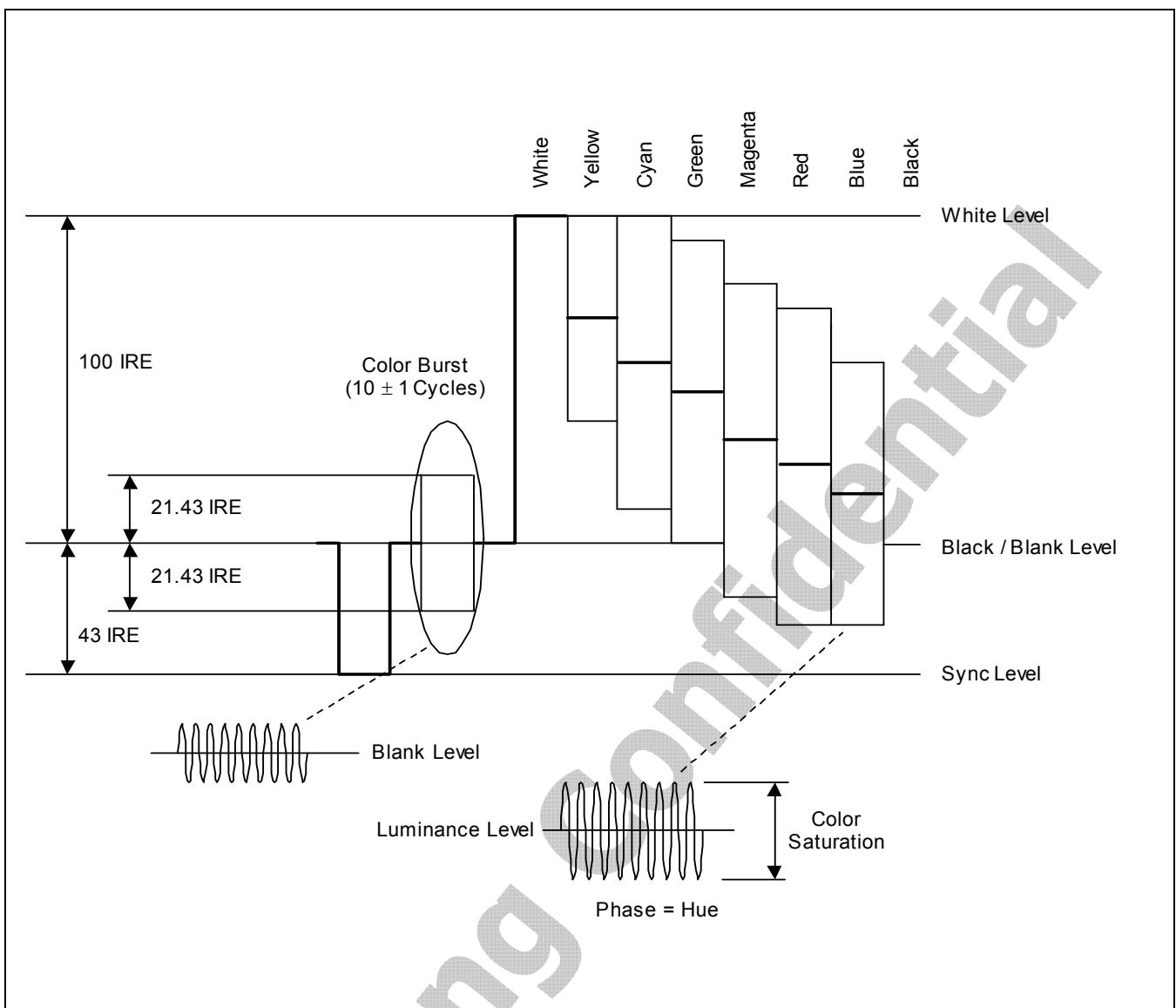


Figure 7-14 PAL (BGHIDNc) Composite Video Signal with 75% Color Bars

7.8 ILLEGAL COLOR COMPENSATION (CVBS MODULE)

The CVBS sub-module also supports color compensation for illegal RGB data. Video data are usually processed in the YCbCr coordinates. At the result of filtering and scaling, the values of YCbCr can exceed their nominal range and values become invalid when they are transformed into RGB coordinates. This causes unwanted artifacts at display. The Figure illustrates the relation between YCbCr coordinates and RGB coordinates:

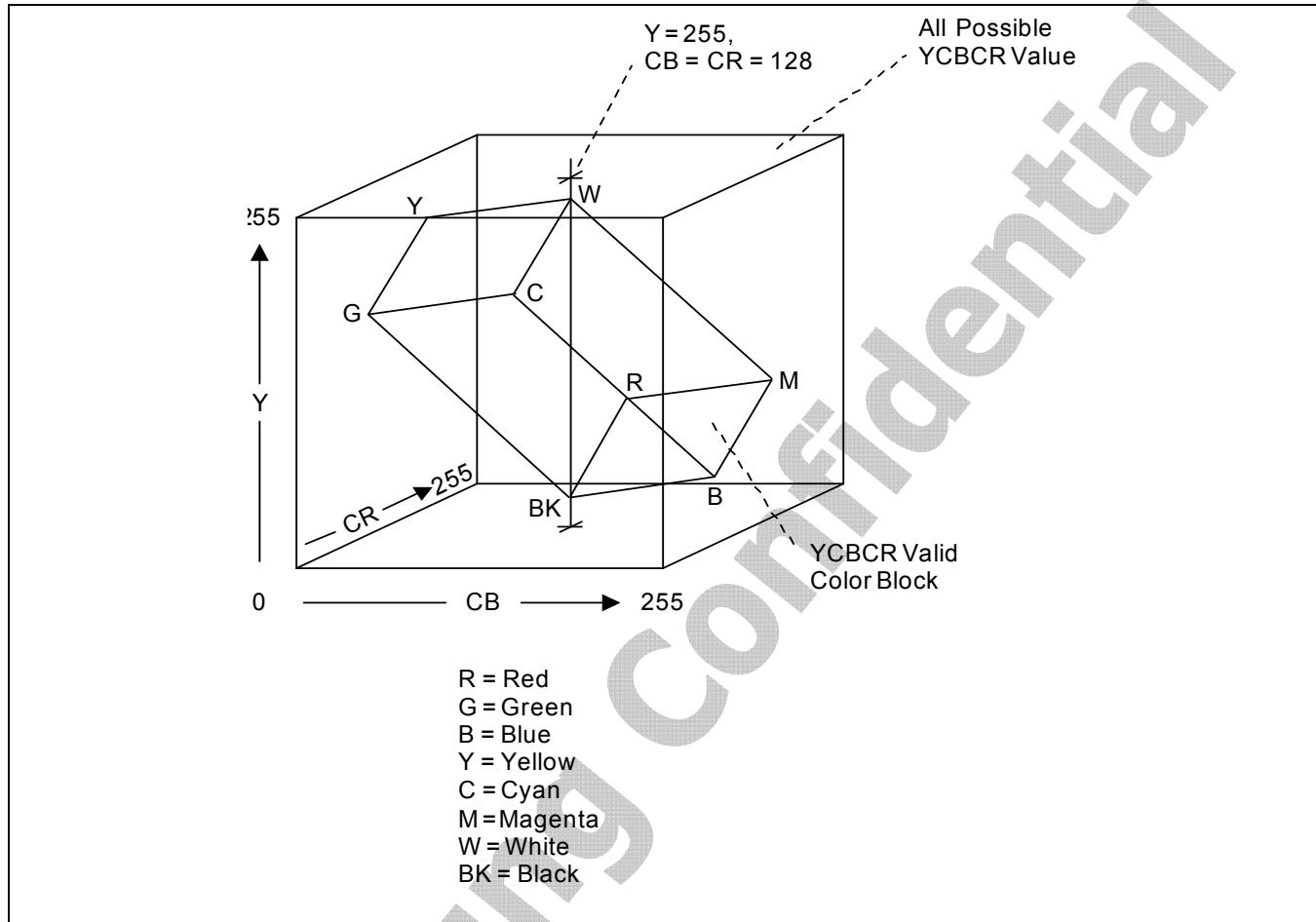


Figure 7-15 Color Cube Comparison

If YCbCr data converted to RGB data outside the RGB cube, the CVBS sub-module compensates the value so that the result falls within the RGB cube. A constant luma and constant hue approach is used for this compensation. The luminance Y is not altered while the chrominance Cb and Cr are limited to the maximum valid values having the same hue as the invalid color prior to limiting. The SDO_RGB_CC register controls the size of RGB cube that determine color compensation range. Meanwhile, if illegal YCbCr data are transformed to CVBS, there might be an overflow which exceeds DAC conversion range. The CVBS sub-module also compensates this error. The SDO_CVBS_CC_Y1, SDO_CVBS_CC_Y2, and SDO_CVBS_CC_C registers are used for the color compensation for CVBS output.

7.9 OVERSAMPLING & DAC COMPENSATION FILTER (OSF MODULE)

The TVOUT module provides 2x or 4X oversampling filter prior to DAC in order to ease the design of analog anti-image filter circuitry. For interlaced scanning case whose sample rate is 13.5MHz, four times oversampling provides the final output sampling rate as 54Msps (samples per second). For progressive scanning with double sample rate, 27MHz, OSF module performs two times oversampling with the result of also 54Msps. The DAC has high frequency attenuation which comes from the $\sin(x)/x$ characteristic of sample-and-hold nature and can have more attenuation for some specific reasons. As OSF has enough number of taps with programmable coefficients, compensation of such attenuation is done by making the filter response to boost at high frequency.

OSF operates in different ways according to the oversampling ratio; the number of taps and the meanings of the coefficient registers are different for each case.

4x oversampling case: 13.5Msps interlaced mode. It operates as 4-polyphase 95-tap FIR filters. All the coefficient registers (SDO_OSFC00_0~SDO_OSFC23_0) are used.

2x oversampling case: 27Msps progressive mode. It operates as 2-polyphase 47-tap FIR filter. Half of the filter coefficient registers (SDO_OSFC12_0~SDO_OSFC23_0) are used. The remaining values should be set to zero.

The target filter should be a centre symmetric, that is,

Let $h(i)$ a filter coefficient whose index is i .

N is odd number. (For 4x, $N = 95$ and for 2x, $N=47$)

$h(i) = h(N-i)$ for $i=0\sim[N/2]-1$, where $[]$ means rounding to the nearest integers towards zero.

Table 7-4 Over-sampling Filter Coefficients Configuration

SDO Register Name(_0,1,2)	registers (_0, 1, 2)	4x Case meaning (N=95)	2x case meaning (N=47)	number of bits (Including Sign Bit)
SDO_OSFC00	osf_coef00	$(h[0]+h[2])/2 = (h[94]+h[92])/2$	0	8
	osf_coef01	$h[1] = h[93]$	0	8
SDO_OSFC01	osf_coef02	$(h[0]-h[2])/2 = (h[94]-h[92])/2$	0	8
	osf_coef03	$h[3] = h[91]$	0	8
SDO_OSFC02	osf_coef04	$(h[4]+h[6])/2 = (h[90]+h[88])/2$	0	8
	osf_coef05	$h[5] = h[89]$	0	8
SDO_OSFC03	osf_coef06	$(h[4]-h[6])/2 = (h[90]-h[88])/2$	0	8
	osf_coef07	$h[7] = h[87]$	0	8
SDO_OSFC04	osf_coef08	$(h[8]+h[10])/2 = (h[86]+h[84])/2$	0	8
	osf_coef09	$h[9] = h[85]$	0	8
SDO_OSFC05	osf_coef10	$(h[8]-h[10])/2 = (h[86]-h[84])/2$	0	8
	osf_coef11	$h[11] = h[83]$	0	8
SDO_OSFC06	osf_coef12	$(h[12]+h[14])/2 = (h[82]+h[80])/2$	0	9
	osf_coef13	$h[13] = h[81]$	0	9
SDO_OSFC07	osf_coef14	$(h[12]-h[14])/2 = (h[82]-h[80])/2$	0	9
	osf_coef15	$h[15] = h[79]$	0	9
SDO_OSFC08	osf_coef16	$(h[16]+h[18])/2 = (h[78]+h[76])/2$	0	9
	osf_coef17	$h[17] = h[77]$	0	9
SDO_OSFC09	osf_coef18	$(h[16]-h[18])/2 = (h[78]-h[76])/2$	0	9
	osf_coef19	$h[19] = h[75]$	0	9
SDO_OSFC10	osf_coef20	$(h[20]+h[22])/2 = (h[74]+h[72])/2$	0	9
	osf_coef21	$h[21] = h[73]$	0	9
SDO_OSFC11	osf_coef22	$(h[20]-h[22])/2 = (h[74]-h[72])/2$	0	9
	osf_coef23	$h[23] = h[71]$	0	9
SDO_OSFC12	osf_coef24	$(h[24]+h[26])/2 = (h[70]+h[68])/2$	$h[0] = h[46]$	10
	osf_coef25	$h[25] = h[69]$	$h[1] = h[45]$	10
SDO_OSFC13	osf_coef26	$(h[24]-h[26])/2 = (h[70]-h[68])/2$	$h[2] = h[44]$	10
	osf_coef27	$h[27] = h[67]$	$h[3] = h[43]$	10
SDO_OSFC14	osf_coef28	$(h[28]+h[30])/2 = (h[66]+h[64])/2$	$h[4] = h[42]$	10
	osf_coef29	$h[29] = h[65]$	$h[5] = h[41]$	10
SDO_OSFC15	osf_coef30	$(h[28]-h[30])/2 = (h[66]-h[64])/2$	$h[6] = h[40]$	10

SDO Register Name(_0,1,2)	registers (_0, 1, 2)	4x Case meaning (N=95)	2x case meaning (N=47)	number of bits (Including Sign Bit)
	osf_coef31	$h[31] = h[63]$	$h[7] = h[39]$	10
SDO_OSFC16	osf_coef32	$(h[32]+h[34])/2 =$ $(h[62]+h[60])/2$	$h[8] = h[38]$	10
	osf_coef33	$h[33] = h[61]$	$h[9] = h[37]$	10
SDO_OSFC17	osf_coef34	$(h[32]-h[34])/2 = (h[62]-h[60])/2$	$h[10] = h[36]$	10
	osf_coef35	$h[35] = h[59]$	$h[11] = h[35]$	10
SDO_OSFC18	osf_coef36	$(h[36]+h[38])/2 =$ $(h[58]+h[56])/2$	$h[12] = h[34]$	10
	osf_coef37	$h[37] = h[57]$	$h[13] = h[33]$	10
SDO_OSFC19	osf_coef38	$(h[36]-h[38])/2 = (h[58]-h[56])/2$	$h[14] = h[32]$	10
	osf_coef39	$h[39] = h[55]$	$h[15] = h[31]$	10
SDO_OSFC20	osf_coef40	$(h[40]+h[42])/2 =$ $(h[54]+h[52])/2$	$h[16] = h[30]$	11
	osf_coef41	$h[41] = h[53]$	$h[17] = h[29]$	11
SDO_OSFC21	osf_coef42	$(h[40]-h[42])/2 = (h[54]-h[52])/2$	$h[18] = h[28]$	11
	osf_coef43	$h[43] = h[51]$	$h[19] = h[27]$	11
SDO_OSFC22	osf_coef44	$(h[44]+h[46])/2 =$ $(h[50]+h[48])/2$	$h[20] = h[26]$	12
	osf_coef45	$h[45] = h[49]$	$h[21] = h[25]$	12
SDO_OSFC23	osf_coef46	$(h[44]-h[46])/2 = (h[50]-$ $h[48])/2$	$h[22] = h[24]$	12
	osf_coef47	$h[47]$	$h[23]$	12

7.10 REGISTER CONTROL (CTRL MODULE)

The TVOUT module supports AHB+ slave bus interface for register control. All the registers are synchronized with system bus clock.

7.11 I/O DESCRIPTION

Signal	I/O	Description	Pad	Type
XdacCOMP	Output	Compensation Pin. This is a compensation pin for the internal reference amplifier. A 0.1uF ceramic capacitor must be connected between COMP and 3.3V PWR.	XdacCOMP	Dedicated
XdacIREF	Input	Full Scale Adjust control. The full-scale current drive on each of the output channels is determined by the value of a resistor RSET connected between this terminal and GND.	XdacIREF	Dedicated
XdacVREF	Input	Voltage reference for DAC. An Internal voltage reference of nominally 1.22V is provided. Can be driven with an external reference source.	XdacVREF	Dedicated
XdacOUT	Output	DAC current output.	XdacOUT	Dedicated

7.12 REGISTER DESCRIPTION

7.12.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
SDO_CLKCON	0xF900_0000	R/W	Clock Control Register	0x0000_0000
SDO_CONFIG	0xF900_0008	R/W	Video Standard Configuration Register	0x0024_2430
SDO_SCALE	0xF900_000C	R/W	Video Scale Configuration Register	0x0000_0006
SDO_VBI	0xF900_0014	R/W	VBI Configuration Register	0x0007_77FF
SDO_SCALE_CH0	0xF900_001C	R/W	Scale Control Register for DAC Channel0	0x0000_0800
SDO_YCDELAY	0xF900_0034	R/W	Video Delay Control Register	0x0000_FA00
SDO_SCHLOCK	0xF900_0038	R/W	SCH Phase Control Register	0x0000_0000
SDO_DAC	0xF900_003C	R/W	DAC Configuration Register	0x0000_0000
SDO_FINFO	0xF900_0040	R	Status Register	0x0000_0002
SDO_Y0	0xF900_0044	R/W	Y- AAF 1'st and 23'th Coefficient (AAF : Anti-Aliasing Filter)	0x0000_0000
SDO_Y1	0xF900_0048	R/W	Y- AAF 2'nd and 22'th Coefficient	0x0000_0000
SDO_Y2	0xF900_004C	R/W	Y- AAF 3'rd and 21'th Coefficient	0x0000_0000
SDO_Y3	0xF900_0050	R/W	Y- AAF 4'th and 20'th Coefficient	0x0000_0000
SDO_Y4	0xF900_0054	R/W	Y- AAF 5'th and 19'th Coefficient	0x0000_0000
SDO_Y5	0xF900_0058	R/W	Y- AAF 6'th and 18'th Coefficient	0x0000_0000
SDO_Y6	0xF900_005C	R/W	Y- AAF 7'th and 17'th Coefficient	0x0000_0000
SDO_Y7	0xF900_0060	R/W	Y- AAF 8'th and 16'th Coefficient	0x0000_0000
SDO_Y8	0xF900_0064	R/W	Y- AAF 9'th and 15'th Coefficient	0x0000_0000
SDO_Y9	0xF900_0068	R/W	Y- AAF 10'th and 14'th Coefficient	0x0000_0000
SDO_Y10	0xF900_006C	R/W	Y- AAF 11'th and 13'th Coefficient	0x0000_0000
SDO_Y11	0xF900_0070	R/W	Y- AAF 12'th Coefficient	0x0000_025D
SDO_CB0	0xF900_0080	R/W	CB- AAF 1'st and 23'th Coefficient	0x0000_0000
SDO_CB1	0xF900_0084	R/W	CB- AAF 2'nd and 22'th Coefficient	0x0000_0000
SDO_CB2	0xF900_0088	R/W	CB- AAF 3'rd and 21'th Coefficient	0x0000_0000
SDO_CB3	0xF900_008C	R/W	CB-AAF 4'th and 20'th Coefficient	0x0000_0000
SDO_CB4	0xF900_0090	R/W	CB- AAF 5'th and 19'th Coefficient	0x0000_0000
SDO_CB5	0xF900_0094	R/W	CB- AAF 6'th and 18'th Coefficient	0x0000_0001
SDO_CB6	0xF900_0098	R/W	CB- AAF 7'th and 17'th Coefficient	0x0000_0007
SDO_CB7	0xF900_009C	R/W	CB- AAF 8'th and 16'th Coefficient	0x0000_0014
SDO_CB8	0xF900_00A0	R/W	CB- AAF 9'th and 15'th Coefficient	0x0000_0028
SDO_CB9	0xF900_00A4	R/W	CB- AAF 10'th and 14'th Coefficient	0x0000_003F

Register	Address	R/W	Description	Reset Value
SDO_CB10	0xF900_00A8	R/W	CB- AAF 11'th and 13'th Coefficient	0x0000_0052
SDO_CB11	0xF900_00AC	R/W	CB- AAF 12'th Coefficient	0x0000_005A
SDO_CR0	0xF900_00C0	R/W	CR- AAF 1'st and 23'th Coefficient	0x0000_0000
SDO_CR1	0xF900_00C4	R/W	CR- AAF 2'nd and 22'th Coefficient	0x0000_0000
SDO_CR2	0xF900_00C8	R/W	CR- AAF 3'rd and 21'th Coefficient	0x0000_0000
SDO_CR3	0xF900_00CC	R/W	CR-AAF 4'th and 20'th Coefficient	0x0000_0000
SDO_CR4	0xF900_00D0	R/W	CR- AAF 5'th and 19'th Coefficient	0x0000_0000
SDO_CR5	0xF900_00D4	R/W	CR- AAF 6'th and 18'th Coefficient	0x0000_0001
SDO_CR6	0xF900_00D8	R/W	CR- AAF 7'th and 17'th Coefficient	0x0000_0009
SDO_CR7	0xF900_00DC	R/W	CR- AAF 8'th and 16'th Coefficient	0x0000_001C
SDO_CR8	0xF900_00E0	R/W	CR- AAF 9'th and 15'th Coefficient	0x0000_0039
SDO_CR9	0xF900_00E4	R/W	CR- AAF 10'th and 14'th Coefficient	0x0000_005A
SDO_CR10	0xF900_00E8	R/W	CR- AAF 11'th and 13'th Coefficient	0x0000_0074
SDO_CR11	0xF900_00EC	R/W	CR- AAF 12'th Coefficient	0x0000_007E
SDO_CCCON	0xF900_0180	R/W	Color Compensation On/ Off Control	0x0000_0000
SDO_YSCALE	0xF900_0184	R/W	Brightness Control for Y	0x0080_0000
SDO_CBSCALE	0xF900_0188	R/W	Hue/ Saturation Control for CB	0x0080_0000
SDO_CRSCALE	0xF900_018C	R/W	Hue/ Saturation Control for CR	0x0000_0080
SDO_CB_CR_OFFSET	0xF900_0190	R/W	Hue/ Sat Offset Control for CB/CR	0x0000_0000
SDO_CVBS_CC_Y1	0xF900_0198	R/W	Color Compensation of CVBS Output	0x0200_0000
SDO_CVBS_CC_Y2	0xF900_019C	R/W	Color Compensation of CVBS Output	0x03FF_0200
SDO_CVBS_CC_C	0xF900_01A0	R/W	Color Compensation of CVBS Output	0x0000_01FF
SDO_OSFC00_0	0xF900_0200	R/W	OverSampling Filter (OSF) Coefficient 1 & 0. of channel #0	0x00FD_00FE
SDO_OSFC01_0	0xF900_0204	R/W	OSF Coefficient 3 & 2 of Channel #0	0x0000_0000
SDO_OSFC02_0	0xF900_0208	R/W	OSF Coefficient 5 & 4 of Channel #0	0x0005_0004
SDO_OSFC03_0	0xF900_020C	R/W	OSF Coefficient 7 & 6 of Channel #0	0x0000_00FF
SDO_OSFC04_0	0xF900_0210	R/W	OSF Coefficient 9 & 8 of Channel #0	0x00F7_00FA
SDO_OSFC05_0	0xF900_0214	R/W	OSF Coefficient 11 & 10 of Channel #0	0x0000_0001
SDO_OSFC06_0	0xF900_0218	R/W	OSF Coefficient 13 & 12 of Channel #0	0x000E_000A
SDO_OSFC07_0	0xF900_021C	R/W	OSF Coefficient 15 & 14 of Channel #0	0x0000_01FF
SDO_OSFC08_0	0xF900_0220	R/W	OSF Coefficient 17 & 16 of Channel #0	0x01EC_01F2
SDO_OSFC09_0	0xF900_0224	R/W	OSF Coefficient 19 & 18 of Channel #0	0x0000_0001
SDO_OSFC10_0	0xF900_0228	R/W	OSF Coefficient 21 & 20 of Channel #0	0x001D_0014
SDO_OSFC11_0	0xF900_022C	R/W	OSF Coefficient 23 & 22 of Channel #0	0x0000_01FE
SDO_OSFC12_0	0xF900_0230	R/W	OSF Coefficient 25 & 24 of Channel #0	0x03D8_03E4
SDO_OSFC13_0	0xF900_0234	R/W	OSF Coefficient 27 & 26 of Channel #0	0x0000_0002

Register	Address	R/W	Description	Reset Value
SDO_OSFC14_0	0xF900_0238	R/W	OSF Coefficient 29 & 28 of Channel #0	0x0038_0028
SDO_OSFC15_0	0xF900_023C	R/W	OSF Coefficient 31 & 30 of Channel #0	0x0000_03FD
SDO_OSFC16_0	0xF900_0240	R/W	OSF Coefficient 33 & 32 of Channel #0	0x03B0_03C7
SDO_OSFC17_0	0xF900_0244	R/W	OSF Coefficient 35 & 34 of Channel #0	0x0000_0005
SDO_OSFC18_0	0xF900_0248	R/W	OSF Coefficient 37 & 36 of Channel #0	0x0079_0056
SDO_OSFC19_0	0xF900_024C	R/W	OSF Coefficient 39 & 38 of Channel #0	0x0000_03F6
SDO_OSFC20_0	0xF900_0250	R/W	OSF Coefficient 41 & 40 of Channel #0	0x072C_0766
SDO_OSFC21_0	0xF900_0254	R/W	OSF Coefficient 43 & 42 of Channel #0	0x0000_001B
SDO_OSFC22_0	0xF900_0258	R/W	OSF Coefficient 45 & 44 of Channel #0	0x028B_0265
SDO_OSFC23_0	0xF900_025C	R/W	OSF Coefficient 47 & 46 of Channel #0	0x0400_0ECC
SDO_XTALK0	0xF900_0260	R/W	Crosstalk Cancel Coefficient for Ch.0	0x0000_0000
SDO_BB_CTRL	0xF900_026C	R/W	Blackburst Test Control	0x0001_1A00
SDO_IRQ	0xF900_0280	R/W	Interrupt Request Register	0x0000_0000
SDO_IRQMASK	0xF900_0284	R/W	Interrupt Request Enable Register	0x0000_0000
Reserved	0xF900_02C0	R/W	Reserved	0x00FD_00FE
Reserved	0xF900_02C4	R/W	Reserved	0x0000_0000
Reserved	0xF900_02C8	R/W	Reserved	0x0005_0004
Reserved	0xF900_02CC	R/W	Reserved	0x0000_00FF
Reserved	0xF900_02D0	R/W	Reserved	0x00F7_00FA
Reserved	0xF900_02D4	R/W	Reserved	0x0000_0001
Reserved	0xF900_02D8	R/W	Reserved	0x000E_000A
Reserved	0xF900_02DC	R/W	Reserved	0x0000_01FF
Reserved	0xF900_02E0	R/W	Reserved	0x01EC_01F2
Reserved	0xF900_02E4	R/W	Reserved	0x0000_0001
Reserved	0xF900_02E8	R/W	Reserved	0x001D_0014
Reserved	0xF900_02EC	R/W	Reserved	0x0000_01FE
Reserved	0xF900_02F0	R/W	Reserved	0x03D8_03E4
Reserved	0xF900_02F4	R/W	Reserved	0x0000_0002
Reserved	0xF900_02F8	R/W	Reserved	0x0038_0028
Reserved	0xF900_02FC	R/W	Reserved	0x0000_03FD
Reserved	0xF900_0300	R/W	Reserved	0x03B0_03C7
Reserved	0xF900_0304	R/W	Reserved	0x0000_0005
Reserved	0xF900_0308	R/W	Reserved	0x0079_0056
Reserved	0xF900_030C	R/W	Reserved	0x0000_03F6
Reserved	0xF900_0310	R/W	Reserved	0x072C_0766
Reserved	0xF900_0314	R/W	Reserved	0x0000_001B
Reserved	0xF900_0318	R/W	Reserved	0x028B_0265

Register	Address	R/W	Description	Reset Value
Reserved	0xF900_031C	R/W	Reserved	0x0400_0ECC
Reserved	0xF900_0320	R/W	Reserved	0x00FD_00FE
Reserved	0xF900_0324	R/W	Reserved	0x0000_0000
Reserved	0xF900_0328	R/W	Reserved	0x0005_0004
Reserved	0xF900_032C	R/W	Reserved	0x0000_00FF
Reserved	0xF900_0330	R/W	Reserved	0x00F7_00FA
Reserved	0xF900_0334	R/W	Reserved	0x0000_0001
Reserved	0xF900_0338	R/W	Reserved	0x000E_000A
Reserved	0xF900_033C	R/W	Reserved	0x0000_01FF
Reserved	0xF900_0340	R/W	Reserved	0x01EC_01F2
Reserved	0xF900_0344	R/W	Reserved	0x0000_0001
Reserved	0xF900_0348	R/W	Reserved	0x001D_0014
Reserved	0xF900_034C	R/W	Reserved	0x0000_01FE
Reserved	0xF900_0350	R/W	Reserved	0x03D8_03E4
Reserved	0xF900_0354	R/W	Reserved	0x0000_0002
Reserved	0xF900_0358	R/W	Reserved	0x0038_0028
Reserved	0xF900_035C	R/W	Reserved	0x0000_03FD
Reserved	0xF900_0360	R/W	Reserved	0x03B0_03C7
Reserved	0xF900_0364	R/W	Reserved	0x0000_0005
Reserved	0xF900_0368	R/W	Reserved	0x0079_0056
Reserved	0xF900_036C	R/W	Reserved	0x0000_03F6
Reserved	0xF900_0370	R/W	Reserved	0x072C_0766
Reserved	0xF900_0374	R/W	Reserved	0x0000_001B
Reserved	0xF900_0378	R/W	Reserved	0x028B_0265
Reserved	0xF900_037C	R/W	Reserved	0x0400_0ECC
SDO_ARMCC	0xF900_03C0	R/W	Closed Caption Data Register	0x0000_0000
SDO_ARMWSS525	0xF900_03C4	R/W	WSS 525 Data Register	0x0000_0000
SDO_ARMWSS625	0xF900_03C8	R/W	WSS 625 Data Register	0x0000_0000
SDO_ARMCGMS525	0xF900_03CC	R/W	CGMS-A 525 Data Register	0x0000_0000
SDO_ARMCGMS625	0xF900_03D4	R/W	CGMS-A 625 Data Register	0x0000_0000
SDO_VERSION	0xF900_03D8	R	TVOUT Version Number Read Register	0x0000_000C
Shadow Register Description				
SDO_CC	0xF900_0380	R/W	Closed Caption Data Shadow register	0x0000_0000
SDO_WSS525	0xF900_0384	R/W	WSS 525 Data Shadow Register	0x0000_0000
SDO_WSS625	0xF900_0388	R/W	WSS 625 Data Shadow Register	0x0000_0000
SDO_CGMS525	0xF900_038C	R/W	CGMS-A 525 Data Shadow Register	0x0000_0000

Register	Address	R/W	Description	Reset Value
SDO_CGMS625	0xF900_0394	R/W	CGMS-A 625 Data Shadow Register	0x0000_0000

NOTE: SDO_ARMCC, SDO_ARMWSS525, SDO_ARMWSS625, SDO_ARMCGMS525, and SDO_ARMCGMS625 are paired with above shadow registers, respectively, as follows:

- SDO_ARMCC : SDO_CC
- SDO_ARMWSS525 : SDO_WSS525
- SDO_ARMWSS625 : SDO_WSS625
- SDO_ARMCGMS525: SDO_CGMS525
- SDO_ARMCGMS625: SDO_CGMS625

If MCU set values in the source registers with the prefix of SDO_ARMXXX, they are not immediately effective. But the values are copied into the corresponding shadow registers which are named with SDO_XXX. They are effective at the next vertical sync. Then the values become effective during VBI interval at the next field.

Avoid setting direct value to these shadow registers.

7.12.1.1 SDO Clock Control Register (SDO_CLKCON, R/W, Address = 0xF900_0000)

SDO_CLKCON	Bit	Description	Reset Value
Reserved	[31:5]	Reserved, read as zero, do not modify	0
SDO software reset	[4]	This bit controls software reset of TVOUT. Software reset is active high signal. 0 = No reset 1 = Enables software reset	0
Reserved	[3:2]	Reserved, read as zero, do not modify	
SDO clock down ready (read only)	[1]	Indicates whether host controller can stop the clock for the TVOUT. 0 = Clock-down not ready 1 = Clock-down ready Normally this bit is 0. After SDO_CLKCON [0] bit is 0, if the internal line counter and pixel counter are 0 (just before starting line 1), this bit will be 1.	0
SDO clock on	[0]	This bit determines run/ stop mode of TVOUT. 0 = TVOUT clock off. TVOUT requests for clock down to host controller. If SDO is ready for clock down, SDO_CLKCON [1] bit will be 1. The host controller should stop the clock for the TVOUT after that. 1 = TVOUT clock on. TVOUT starts running. NOTE: Vertical Sync of TVENC's Timing Generator updates the SFRs of Video Processor and Image Mixer. Thus, SFRs are configured before this bit is enabled. The sequence to enable TVSS is as follows: "VP -> MIXER -> TVENC". Also, because of the same reason, the disabling sequence is following as : "VP -> MIXER -> TVNEC".	0

Image Mixer transmits video data to TVENC. To connect Image Mixer and TVENC you must configure REG_DST_SEL at mixer_CFG register (0xF920_0004). To synchronous between Image Mixer and TVENC you must configure Image Mixer I/F clock (VCLKHS) and VCLKS (TVENC clock) (fixed by 54MHz). Thus, you must set MIXER_SEL register in CLK_SRC1(0xE010_0204). For more information, refer to CMU chapter.

7.12.1.2 SDO Video Standard Configuration Register (SDO_CONFIG, R/W, Address = 0xF900_0008)

SDO_CONFIG	Bit	Description	Initial State
Reserved	[31:22]	Reserved, read as zero, do not modify	0
Reserved	[21:20]	Reserved, read as zero, do not modify	2
Reserved	[19:18]	Reserved, read as zero, do not modify	1
Reserved	[17:16]	Reserved, read as zero, do not modify	0
Reserved	[15:14]	Reserved, read as zero, do not modify	0
Reserved	[13:12]	Reserved, read as zero, do not modify	2
Reserved	[11:10]	Reserved, read as zero, do not modify	1
Selection of Video Mux for DAC	[9:8]	0 = CVBS signal 1 = Y signal 2 = C signal	0
Reserved	[7]	Reserved, read as zero, do not modify	0
Reserved	[6]	Reserved, read as zero, do not modify	0
Reserved	[5]	Reserved, read as zero, do not modify	1
Reserved	[4]	Reserved, read as zero, do not modify	1
Selection of Video Standard	[3:0]	0 = NTSC (M) 1 = PAL (M) 2 = PAL (BGHID) 3 = PAL (N) 4 = PAL (Nc) 8 = NTSC 4.43 9 = PAL 60	0

7.12.1.3 SDO Video Scale Configuration Register (SDO_SCALE, R/W, Address = 0xF900_000C)

SDO_SCALE	Bit	Description	Initial State
Reserved	[31:4]	Reserved, read as zero, do not modify	0
Reserved	[3]	Reserved, read as zero, do not modify	0
Reserved	[2]	Reserved, read as zero, do not modify	1
Setup Level Selection for Composite	[1]	0 = 0 IRE 1 = 7.5 IRE This setting is valid if bit[6] of SDO_CONFIG register is set to composite.	1
Video-to-Sync Ratio Selection for Composite	[0]	0 = 10:4 1 = 7:3 This setting is valid if bit[6] of SDO_CONFIG register is set to composite.	0

NOTE: The Table 7-5 and Table 7-6 according to the value of bit [3:2]

7.12.1.4 SDO VBI Configuration Register (SDO_VBI, R/W, Address = 0xF900_0014)

SDO_VBI	Bit	Description	Initial State
Reserved	[31:15]	Reserved, read as zero, do not modify	E
Wide Screen Signaling Configuration for CVBS Channel	[14]	If the Selection of Video Standard in SDO_CONFIG Register is set to NTSC (M), PAL (M), PAL 60, or NTSC 4.43, 0 = No Ancillary Data Insertion 1 = 525 line WSS Insertion at 20H and 283H Otherwise, 0 = No Ancillary Data Insertion 1 = 625 line WSS Insertion at 23H This setting is valid if the bit [6] of SDO_CONFIG register is set to composite	1
Closed Caption Configuration for CVBS Channel	[13:12]	If the Selection of Video Standard in SDO_CONFIG Register is set to NTSC (M), PAL (M), PAL 60, or NTSC 4.43, 0 = No Ancillary Data Insertion 1 = US Closed Caption Insertion at 21H 2 = US Closed Caption Insertion at 21H and 284H 3 = Reserved for Other Use Otherwise, No Ancillary Data Insertion. (NOTE) This setting is valid if the bit [6] of SDO_CONFIG register is set to composite. Note: European closed caption is not supported.	3
Reserved	[11]	Reserved, read as zero, do not modify	0
Reserved	[10]	Reserved, read as zero, do not modify	1
Reserved	[9:8]	Reserved, read as zero, do not modify	3
Reserved	[7]	Reserved, read as zero, do not modify	1
Reserved	[6]	Reserved, read as zero, do not modify	1
Reserved	[5:4]	Reserved, read as zero, do not modify	3
Reserved	[3]	Reserved, read as zero, do not modify	1
Reserved	[2]	Reserved, read as zero, do not modify	1
Reserved	[1:0]	Reserved, read as zero, do not modify	3

7.12.1.5 SDO Channel #0 Scale Control Register (SDO_SCALE_CH0, R/W, Address = 0xF900_001C)

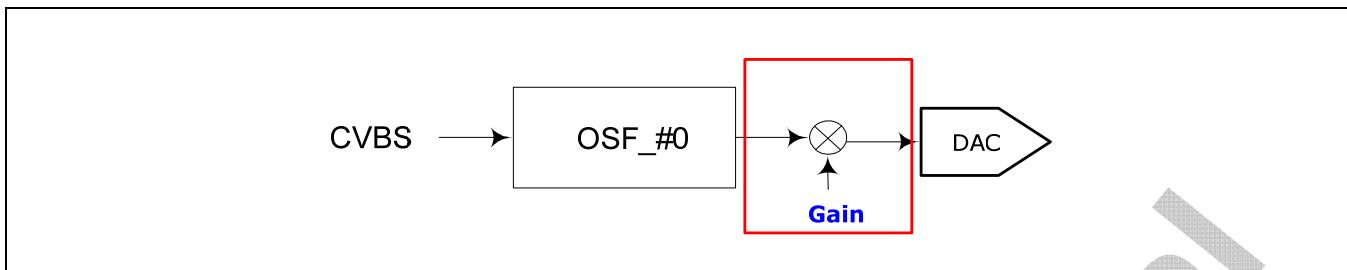


Figure 7-16 Individual Gain & Offset Control for DAC Channel Balancing

SDO_SCALE_CH0	Bit	Description	Initial State
Reserved	[31:26]	Reserved, read as zero, do not modify	0
Offset of Channel 0 Signal Scale Conversion	[25:16]	Function $F(x) = (X + \text{Offset}) * \text{Gain}$ 0x1FF = +511 ... 0x001 = +1 0x000 = 0 0x3FF = -1 ... 0x200 = -512 This setting is valid if the bit [6] of SDO_CONFIG register is set to component.	000
Reserved	[15:12]	Reserved, read as zero, do not modify	0
Gain of Channel 0 Signal Scale Conversion	[11:0]	Function $F(x) = (X + \text{Offset}) * \text{Gain}$ 0x000 = x0.0 ... 0x400 = x0.5 ... 0x800 = x1.0 ... 0xC00 = x1.5 ... 0xFFFF = x1.999512, $(2048^2 - 1)/2048$ This setting is valid if the bit [6] of SDO_CONFIG register is set to component.	800

7.12.1.6 SDO Video Delay Control Register (SDO_YCDELAY, R/W, Address = 0xF900_0034)

SDO_YCDELAY	Bit	Description	Initial State
Reserved	[31:20]	Reserved, read as zero, do not modify	0
Delay of Y Signal with Respect to C Signal	[19:16]	0x0 = 0.000 usec 0x1 = 0.074 usec ... 0xF = 1.111 usec	0
Offset of Video Active Start Position	[15:8]	0x3F = +4.667 usec ... 0x01 = +0.074 usec 0x00 = 0.000 usec 0xFF = -0.074 usec ... 0x40 = -4.741 usec	FA
Offset of Video Active End Position	[7:0]	0x3F = +4.667 usec ... 0x01 = +0.074 usec 0x00 = 0.000 usec 0xFF = -0.074 usec ... 0x40 = -4.741 usec	00

7.12.1.7 SDO SCH Phase Control Register (SDO_SCHLOCK, R/W, Address = 0xF900_0038)

SDO_SCHLOCK	Bit	Description	Initial State
Reserved	[31:1]	Reserved, read as zero, do not modify	0
Color Sub-Carrier Phase Adjustment	[0]	0 = Never adjusted 1 = Every field is adjusted such that color sub-carrier frequency and horizontal frequency are locked to each other.	0

7.12.1.8 SDO DAC Configuration Register (SDO_DAC, R/W, Address = 0xF900_003C)

SDO_DAC	Bit	Description	Initial State
Reserved	[31:1]	Reserved, read as zero, do not modify	0
Power Down for DAC	[0]	0 = DAC power down 1 = DAC power on	0

7.12.1.9 SDO Status Register (SDO_FINFO, R, Address = 0xF900_0040)

SDO_FINFO	Bit	Description	Initial State
Reserved	[31:26]	Reserved, read as zero, do not modify	0
Field Counter Modulo 1001	[25:16]	This counter is used for 59.94/60.0 Hz field rate conversion.	0
Reserved	[15:2]	Reserved, read as zero, do not modify	0
Field ID	[1]	0 = Top field 1 = Bottom field	1
Field ID with Progressive/Interlaced Indication	[0]	If the SDO_CONFIG register is set to interlaced, 0 = Top field 1 = Bottom field. If the SDO_CONFIG register is set to progressive, this bit would be fixed to zero	0

- 7.12.1.10 SDO Anti-Aliasing Filter Coefficients (SDO_Y0, R/W, Address = 0xF900_0044)
- 7.12.1.11 SDO Anti-Aliasing Filter Coefficients (SDO_Y1, R/W, Address = 0xF900_0048)
- 7.12.1.12 SDO Anti-Aliasing Filter Coefficients (SDO_Y2, R/W, Address = 0xF900_004C)
- 7.12.1.13 SDO Anti-Aliasing Filter Coefficients (SDO_Y3, R/W, Address = 0xF900_0050)
- 7.12.1.14 SDO Anti-Aliasing Filter Coefficients (SDO_Y4, R/W, Address = 0xF900_0054)
- 7.12.1.15 SDO Anti-Aliasing Filter Coefficients (SDO_Y5, R/W, Address = 0xF900_0058)
- 7.12.1.16 SDO Anti-Aliasing Filter Coefficients (SDO_Y6, R/W, Address = 0xF900_005C)
- 7.12.1.17 SDO Anti-Aliasing Filter Coefficients (SDO_Y7, R/W, Address = 0xF900_0060)
- 7.12.1.18 SDO Anti-Aliasing Filter Coefficients (SDO_Y8, R/W, Address = 0xF900_0064)
- 7.12.1.19 SDO Anti-Aliasing Filter Coefficients (SDO_Y9, R/W, Address = 0xF900_0068)
- 7.12.1.20 SDO Anti-Aliasing Filter Coefficients (SDO_Y10, R/W, Address = 0xF900_006C)
- 7.12.1.21 SDO Anti-Aliasing Filter Coefficients (SDO_Y11, R/W, Address = 0xF900_0070)
- 7.12.1.22 SDO Anti-Aliasing Filter Coefficients (SDO_CB0, R/W, Address = 0xF900_0080)
- 7.12.1.23 SDO Anti-Aliasing Filter Coefficients (SDO_CB1, R/W, Address = 0xF900_0084)
- 7.12.1.24 SDO Anti-Aliasing Filter Coefficients (SDO_CB2, R/W, Address = 0xF900_0088)
- 7.12.1.25 SDO Anti-Aliasing Filter Coefficients (SDO_CB3, R/W, Address = 0xF900_008C)
- 7.12.1.26 SDO Anti-Aliasing Filter Coefficients (SDO_CB4, R/W, Address = 0xF900_0090)
- 7.12.1.27 SDO Anti-Aliasing Filter Coefficients (SDO_CB5, R/W, Address = 0xF900_0094)
- 7.12.1.28 SDO Anti-Aliasing Filter Coefficients (SDO_CB6, R/W, Address = 0xF900_0098)
- 7.12.1.29 SDO Anti-Aliasing Filter Coefficients (SDO_CB7, R/W, Address = 0xF900_009C)
- 7.12.1.30 SDO Anti-Aliasing Filter Coefficients (SDO_CB8, R/W, Address = 0xF900_00A0)
- 7.12.1.31 SDO Anti-Aliasing Filter Coefficients (SDO_CB9, R/W, Address = 0xF900_00A4)
- 7.12.1.32 SDO Anti-Aliasing Filter Coefficients (SDO_CB10, R/W, Address = 0xF900_00A8)
- 7.12.1.33 SDO Anti-Aliasing Filter Coefficients (SDO_CB11, R/W, Address = 0xF900_00AC)
- 7.12.1.34 SDO Anti-Aliasing Filter Coefficients (SDO_CR0, R/W, Address = 0xF900_00C0)
- 7.12.1.35 SDO Anti-Aliasing Filter Coefficients (SDO_CR1, R/W, Address = 0xF900_00C4)
- 7.12.1.36 SDO Anti-Aliasing Filter Coefficients (SDO_CR2, R/W, Address = 0xF900_00C8)
- 7.12.1.37 SDO Anti-Aliasing Filter Coefficients (SDO_CR3, R/W, Address = 0xF900_00CC)
- 7.12.1.38 SDO Anti-Aliasing Filter Coefficients (SDO_CR4, R/W, Address = 0xF900_00D0)
- 7.12.1.39 SDO Anti-Aliasing Filter Coefficients (SDO_CR5, R/W, Address = 0xF900_00D4)
- 7.12.1.40 SDO Anti-Aliasing Filter Coefficients (SDO_CR6, R/W, Address = 0xF900_00D8)
- 7.12.1.41 SDO Anti-Aliasing Filter Coefficients (SDO_CR7, R/W, Address = 0xF900_00DC)
- 7.12.1.42 SDO Anti-Aliasing Filter Coefficients (SDO_CR8, R/W, Address = 0xF900_00E0)
- 7.12.1.43 SDO Anti-Aliasing Filter Coefficients (SDO_CR9, R/W, Address = 0xF900_00E4)
- 7.12.1.44 SDO Anti-Aliasing Filter Coefficients (SDO_CR10, R/W, Address = 0xF900_00E8)
- 7.12.1.45 SDO Anti-Aliasing Filter Coefficients (SDO_CR11, R/W, Address = 0xF900_00EC)

SDO_Yn / SDO_CBn / SDO_CRn	Bit	Description	Initial State
Reserved	[31:11]	Reserved, read as zero, do not modify	0
11 Bit Signed Filter Coefficients	[10:0]	<p>Setting values of anti-aliasing filter coefficients is constrained in such a way that total sum of filter coefficients should be equal to a predefined constant. Otherwise, DC component of filter output would be re-scaled from the original one. The value of the constant depends on the setting of SDO_SCALE register :for Y channel, 0x251, at 7.5 IRE setup and 7:3 ratio 0x25D, at 7.5 IRE setup and 10:4 ratio 0x281, at 0 IRE setup and 7:3 ratio 0x28F, at 0 IRE setup and 7:3 ratio,for CB channel, 0x1F3, at 7.5 IRE setup and 7:3 ratio 0x200, at 7.5 IRE setup and 10:4 ratio 0x21E, at 0 IRE setup and 7:3 ratio 0x228, at 0 IRE setup and 7:3 ratio, and for CR channel, 0x2C0, at 7.5 IRE setup and 7:3 ratio 0x2D1, at 7.5 IRE setup and 10:4 ratio 0x2C0, at 0 IRE setup and 7:3 ratio 0x30D, at 0 IRE setup and 7:3 ratio.</p> <p>This setting is valid if the bit [6] of SDO_CONFIG register is set to composite. The setting of Y Filter is applied only to CVBS output.</p>	The reset values are set for the case of ITU-R BT.470 compliant NTSC signal which has 7.5 IRE setup and 10.4 video-to-sync ratio. Refer to "1.15 RESISTERS DESCRIPTION".

7.12.1.46 SDO Color Compensation On/Off Control (SDO_CCCON, R/W, Address = 0xF900_0180)

SDO_CCCON	Bit	Description	Initial State
Reserved	[31:5]	Reserved, read as zero, do not modify	0
On/Off Control of Brightness/ Hue/Saturation Adjustment	[4]	0 = On 1 = Bypass This setting enables/ disables the brightness/ hue/ saturation controls which are controlled by SDO_YSCALE, SDO_CBSCALE, and SDO_CRSCALE.	0
Reserved	[3]	Reserved, read as zero, do not modify	0
Reserved	[2]	Reserved, read as zero, do not modify	0
Reserved	[1]	Reserved, read as zero, do not modify	0
On/Off Control of CVBS Color Compensation	[0]	0 = On 1 = Bypass The CVBS color compensation imposes a saturation operation on the CVBS data. Values which exceed DAC conversion range 0~1023, in 10 bit resolution, would be saturated.	0

7.12.1.47 SDO Brightness Control for Y (SDO_YSCALE, R/W, Address = 0xF900_0184)

SDO_YSCALE	Bit	Description	Initial State
Reserved	[31:24]	Reserved, read as zero, do not modify	0
Gain of Brightness Control with	[23:16]	F(Y) = Gain * Y + Offset 0x00 = 0.0 ... 0x80 = 1.0 ... 0xFF = 1.992188, (128*2 – 1)/128 This setting is valid for all the CVBS/Y/C/YpbPr/RGB outputs.	80
Reserved	[15:9]	Reserved, read as zero, do not modify	0
Offset of Brightness Control with	[8:0]	F(Y) = Gain * Y + Offset 0x0FF = +255 ... 0x001 = +1 0x000 = 0 0x1FF = -1 ... 0x100 = -256 This setting is valid for all the CVBS/Y/C/YpbPr/RGB outputs.	00

7.12.1.48 SDO Hue/Saturation Control for CB (SDO_CBSCALE, R/W, Address = 0xF900_0188)

SDO_CBSCALE	Bit	Description	Initial State
Reserved	[31:25]	Reserved, read as zero, do not modify	0
Gain0_CB	[24:16]	Gain0 of Hue/Saturation Control of CB with $F(\text{CB}, \text{CR}) = \text{CB} * \text{Gain0} + \text{CR} * \text{Gain1} + \text{Offset}$ 0xFF = 1.992188, $(128^2 - 1)/128$... 0x080 = 1.0 ... 0x000 = 0.0 ... 0x180 = -1.0 ... 0x100 = -2.0 This setting is valid for all the CVBS/Y/C/YpbPr/RGB outputs.	80
Reserved	[15:9]	Reserved, read as zero, do not modify	0
Gain1_CB	[8:0]	Gain1 of Hue/Saturation Control of CB with $F(\text{CB}, \text{CR}) = \text{CB} * \text{Gain0} + \text{CR} * \text{Gain1} + \text{Offset}$ 0xFF = 1.992188, $(128^2 - 1)/128$... 0x080 = 1.0 ... 0x000 = 0.0 ... 0x180 = -1.0 ... 0x100 = -2.0 This setting is valid for all the CVBS/Y/C/YpbPr/RGB outputs.	00

7.12.1.49 SDO Hue/Saturation Control for CR (SDO_CRSCALE, R/W, Address = 0xF900_018C)

SDO_CRSCALE	Bit	Description	Initial State
Reserved	[31:25]	Reserved, read as zero, do not modify	0
Gain0_CR	[24:16]	Gain0 of Hue/Saturation Control of CR with $F(CB,CR) = CB * Gain0 + CR * Gain1 + Offset$ 0x0FF = 1.992188, $(128^2 - 1)/128$... 0x080 = 1.0 ... 0x000 = 0.0 ... 0x180 = -1.0 ... 0x100 = -2.0 This setting is valid for all the CVBS/Y/C/YpbPr/RGB outputs.	00
Reserved	[15:9]	Reserved, read as zero, do not modify	0
Gain1_CR	[8:0]	Gain1 of Hue/Saturation Control of CR with $F(CB,CR) = CB * Gain0 + CR * Gain1 + Offset$ 0x0FF = 1.992188, $(128^2 - 1)/128$... 0x080 = 1.0 ... 0x000 = 0.0 ... 0x180 = -1.0 ... 0x100 = -2.0 This setting is valid for all the CVBS/Y/C/YpbPr/RGB outputs.	80

7.12.1.50 SDO Hue/Saturation Control for CB/CR (SDO_CB_CR_OFFSET, R/W, Address = 0xF900_0190)

SDO_CB_CR_OFFSET	Bit	Description	Initial State
Reserved	[31:26]	Reserved, read as zero, do not modify	0
Offset_CR	[25:16]	Offset of Hue/Saturation Control of CR with 0x1FF = +511 ... 0x001 = +1 0x000 = 0 0x3FF = -1 ... 0x200 = -512 This setting is valid for all the CVBS/Y/C/YpbPr/RGB outputs.	00
Reserved	[15:10]	Reserved, read as zero, do not modify	0
Offset_CB	[9:0]	Offset of Hue/Saturation Control of CB with 0x1FF = +511 ... 0x001 = +1 0x000 = 0 0x3FF = -1 ... 0x200 = -512 This setting is valid for all the CVBS/Y/C/YpbPr/RGB outputs.	00

7.12.1.51 Color Compensation Control Register for CVBS Output (SDO_CVBS_CC_Y1, R/W, Address = 0xF900_0198)

SDO_CVBS_CC_Y1	Bit	Description	Initial State
Reserved	[31:26]	Reserved, read as zero, do not modify	0
Y_Lower_Mid_CVBS_Corn	[25:16]	Lower Mid Y Boundary of Legal CVBS Corn 0x3FF = 1023 ... 0x000 = 0	200
Reserved	[15:10]	Reserved, read as zero, do not modify	0
Y_Bottom_CVBS_Corn	[9:0]	Bottom Y Boundary of Legal CVBS Corn 0x3FF = 1023 ... 0x000 = 0	000

7.12.1.52 Color Compensation Control Register for CVBS Output (SDO_CVBS_CC_Y2, R/W, Address = 0xF900_019C)

SDO_CVBS_CC_Y2	Bit	Description	Initial State
Reserved	[31:26]	Reserved, read as zero, do not modify	0
Y_Top_CVBS_Corn	[25:16]	Top Y Boundary of Legal CVBS Corn 0x3FF = 1023 ... 0x000 = 0	3FF
-	[15:10]	Reserved, read as zero, do not modify	0
Y_Upper_Mid_CVBS_Corn	[9:0]	Upper mid Y Boundary of Legal CVBS Corn 0x3FF = 1023 ... 0x000 = 0	200

7.12.1.53 Color Compensation Control Register for CVBS Output (SDO_CVBS_CC_C, R/W, Address = 0xF900_01A0)

SDO_CVBS_CC_C	Bit	Description	Initial State
Reserved	[31:9]	Reserved, read as zero, do not modify	0
Radius_CVBS_Corn	[8:0]	Radius of Legal CVBS Corn 0x1FF = 511 ... 0x000 = 0	1FF

It should be set such that Y_Top_CVBS_Corn >= Y_Upper_Mid_CVBS_Corn >= Y_Lower_Mid_CVBS_Corn >= Y_Bottom_CVBS_Corn. It is highly recommended for users not to alter their reset values. This setting is valid if SDO_CCCON [0] is set to 'On'.

7.12.1.54 SDO 525 Line Component Front/Back Porch Position Control Register (SDO_CSC_525_PORCH, R/W, Address = 0xF900_01B0)

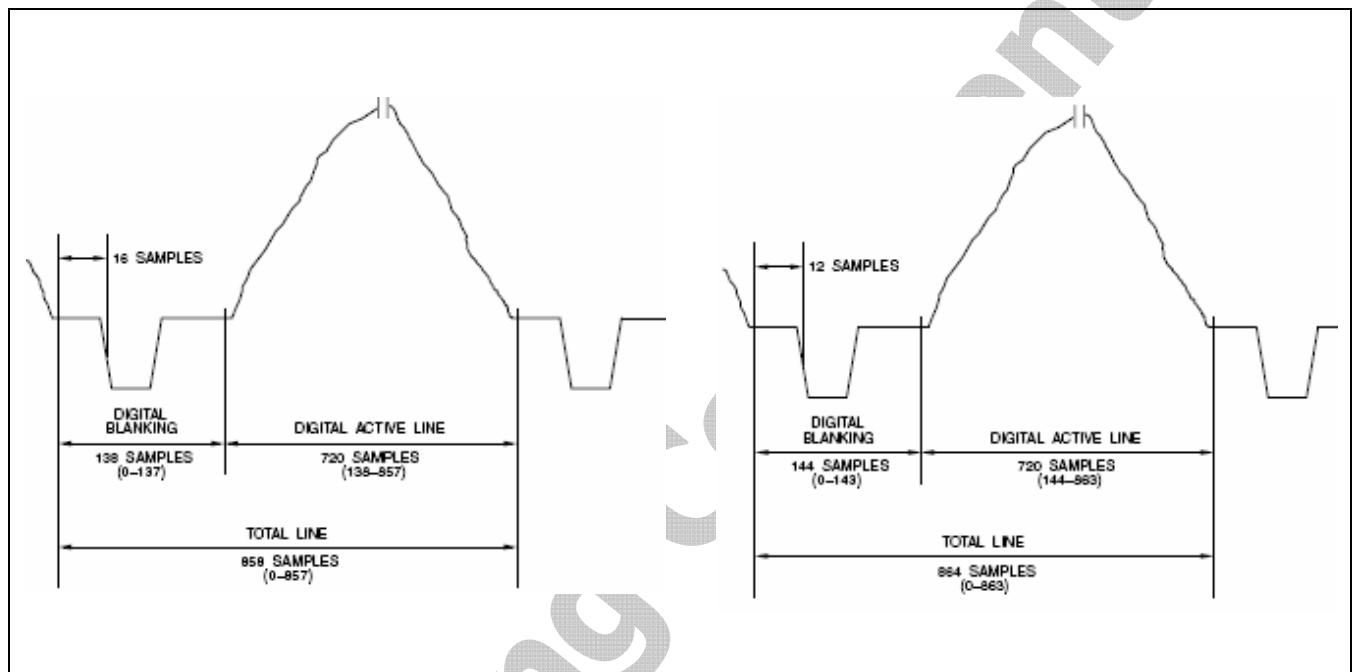
SDO_CSC_525_PORCH	Bit	Description	Initial State
Reserved	[31:26]	Reserved, read as zero, do not modify	0
525 line back-porch position	[25:16]	Back-porch start position	8A
Reserved	[15:10]	Reserved, read as zero, do not modify	0
525 line front-porch position	[9:0]	Front-porch start position	359

- Resolution in progressive: 1/27MHz, - resolution in interlaced: 1/13.5MHz
- Compare line count value with porch position

7.12.1.55 SDO 625 Line Component Front/Back Porch Position Control Register(SDO_CSC_625_PORCH, R/W, Address = 0xF900_01B4)

SDO_CSC_625_PORCH	Bit	Description	Initial State
Reserved	[31:26]	Reserved, read as zero, do not modify	0
625 line back-porch position	[25:16]	Back-porch start position	96
Reserved	[15:10]	Reserved, read as zero, do not modify	0
625 line front-porch position	[9:0]	Front-porch start position	35C

- Resolution in progressive : 1/27MHz, - resolution in interlaced : 1/13.5MHz
 - Compare line count value with porch position



* One Line Count Value of 525 Line

* One Line Count Value of 625 Line

7.12.1.56 SDO Oversampling #0 Filter Coefficient (SDO_OSFC00_0, R/W, Address = 0xF900_0200)

SDO_OSFCN_0 (N is 00~23)	Bit	Description	Initial State
osf_coef(2xN+1)	[23:16]~ [27:16]	(2xN+1)'th oversampling filter coefficient. Signed 8~12bit integer. Actual value is osf_coef/(210)	Refer to below table
osf_coef(2xN)	[7:0] ~[11:0]	(2xN)'th oversampling filter coefficient. Signed 8~12bit integer. Actual value is osf_coef/(210)	

- 7.12.1.57 SDO Oversampling #0 Filter Coefficient (SDO_OSFC01_0, R/W, Address = 0xF900_0204)
- 7.12.1.58 SDO Oversampling #0 Filter Coefficient (SDO_OSFC02_0, R/W, Address = 0xF900_0208)
- 7.12.1.59 SDO Oversampling #0 Filter Coefficient (SDO_OSFC03_0, R/W, Address = 0xF900_020C)
- 7.12.1.60 SDO Oversampling #0 Filter Coefficient (SDO_OSFC04_0, R/W, Address = 0xF900_0210)
- 7.12.1.61 SDO Oversampling #0 Filter Coefficient (SDO_OSFC05_0, R/W, Address = 0xF900_0214)
- 7.12.1.62 SDO Oversampling #0 Filter Coefficient (SDO_OSFC06_0, R/W, Address = 0xF900_0218)
- 7.12.1.63 SDO Oversampling #0 Filter Coefficient (SDO_OSFC07_0, R/W, Address = 0xF900_021C)
- 7.12.1.64 SDO Oversampling #0 Filter Coefficient (SDO_OSFC08_0, R/W, Address = 0xF900_0220)
- 7.12.1.65 SDO Oversampling #0 Filter Coefficient (SDO_OSFC09_0, R/W, Address = 0xF900_0224)
- 7.12.1.66 SDO Oversampling #0 Filter Coefficient (SDO_OSFC10_0, R/W, Address = 0xF900_0228)
- 7.12.1.67 SDO Oversampling #0 Filter Coefficient (SDO_OSFC11_0, R/W, Address = 0xF900_022C)
- 7.12.1.68 SDO Oversampling #0 Filter Coefficient (SDO_OSFC12_0, R/W, Address = 0xF900_0230)
- 7.12.1.69 SDO Oversampling #0 Filter Coefficient (SDO_OSFC13_0, R/W, Address = 0xF900_0234)
- 7.12.1.70 SDO Oversampling #0 Filter Coefficient (SDO_OSFC14_0, R/W, Address = 0xF900_0238)
- 7.12.1.71 SDO Oversampling #0 Filter Coefficient (SDO_OSFC15_0, R/W, Address = 0xF900_023C)
- 7.12.1.72 SDO Oversampling #0 Filter Coefficient (SDO_OSFC16_0, R/W, Address = 0xF900_0240)
- 7.12.1.73 SDO Oversampling #0 Filter Coefficient (SDO_OSFC17_0, R/W, Address = 0xF900_0244)
- 7.12.1.74 SDO Oversampling #0 Filter Coefficient (SDO_OSFC18_0, R/W, Address = 0xF900_0248)
- 7.12.1.75 SDO Oversampling #0 Filter Coefficient (SDO_OSFC19_0, R/W, Address = 0xF900_024C)
- 7.12.1.76 SDO Oversampling #0 Filter Coefficient (SDO_OSFC20_0, R/W, Address = 0xF900_0250)
- 7.12.1.77 SDO Oversampling #0 Filter Coefficient (SDO_OSFC21_0, R/W, Address = 0xF900_0254)
- 7.12.1.78 SDO Oversampling #0 Filter Coefficient (SDO_OSFC22_0, R/W, Address = 0xF900_0258)
- 7.12.1.79 SDO Oversampling #0 Filter Coefficient (SDO_OSFC23_0, R/W, Address = 0xF900_025C)

7.12.1.80 SDO Oversampling #1 Filter Coefficient (SDO_OSFC00_1, R/W, Address = 0xF900_02C0)

SDO_OSFCN_1 (N is 00~23)	Bit	Description	Initial State
osf_coef(2xN+1)	[23:16]~[27:16]	(2xN+1)'th oversampling filter coefficient. Signed 8~12bit integer. Actual value is osf_coef/(210)	Refer to below table
osf_coef(2xN)	[7:0] ~[11:0]	(2xN)'th oversampling filter coefficient. Signed 8~12bit integer. Actual value is osf_coef/(210)	

- 7.12.1.81 SDO Oversampling #1 Filter Coefficient (SDO_OSFC01_1, R/W, Address = 0xF900_02C4)
- 7.12.1.82 SDO Oversampling #1 Filter Coefficient (SDO_OSFC02_1, R/W, Address = 0xF900_02C8)
- 7.12.1.83 SDO Oversampling #1 Filter Coefficient (SDO_OSFC03_1, R/W, Address = 0xF900_02CC)
- 7.12.1.84 SDO Oversampling #1 Filter Coefficient (SDO_OSFC04_1, R/W, Address = 0xF900_02D0)
- 7.12.1.85 SDO Oversampling #1 Filter Coefficient (SDO_OSFC05_1, R/W, Address = 0xF900_02D4)
- 7.12.1.86 SDO Oversampling #1 Filter Coefficient (SDO_OSFC06_1, R/W, Address = 0xF900_02D8)
- 7.12.1.87 SDO Oversampling #1 Filter Coefficient (SDO_OSFC07_1, R/W, Address = 0xF900_02DC)
- 7.12.1.88 SDO Oversampling #1 Filter Coefficient (SDO_OSFC08_1, R/W, Address = 0xF900_02E0)
- 7.12.1.89 SDO Oversampling #1 Filter Coefficient (SDO_OSFC09_1, R/W, Address = 0xF900_02E4)
- 7.12.1.90 SDO Oversampling #1 Filter Coefficient (SDO_OSFC10_1, R/W, Address = 0xF900_02E8)
- 7.12.1.91 SDO Oversampling #1 Filter Coefficient (SDO_OSFC11_1, R/W, Address = 0xF900_02EC)
- 7.12.1.92 SDO Oversampling #1 Filter Coefficient (SDO_OSFC12_1, R/W, Address = 0xF900_02F0)
- 7.12.1.93 SDO Oversampling #1 Filter Coefficient (SDO_OSFC13_1, R/W, Address = 0xF900_02F4)
- 7.12.1.94 SDO Oversampling #1 Filter Coefficient (SDO_OSFC14_1, R/W, Address = 0xF900_02F8)
- 7.12.1.95 SDO Oversampling #1 Filter Coefficient (SDO_OSFC15_1, R/W, Address = 0xF900_02FC)
- 7.12.1.96 SDO Oversampling #1 Filter Coefficient (SDO_OSFC16_1, R/W, Address = 0xF900_0300)
- 7.12.1.97 SDO Oversampling #1 Filter Coefficient (SDO_OSFC17_1, R/W, Address = 0xF900_0304)
- 7.12.1.98 SDO Oversampling #1 Filter Coefficient (SDO_OSFC18_1, R/W, Address = 0xF900_0308)
- 7.12.1.99 SDO Oversampling #1 Filter Coefficient (SDO_OSFC19_1, R/W, Address = 0xF900_030C)
- 7.12.1.100 SDO Oversampling #1 Filter Coefficient (SDO_OSFC20_1, R/W, Address = 0xF900_0310)
- 7.12.1.101 SDO Oversampling #1 Filter Coefficient (SDO_OSFC21_1, R/W, Address = 0xF900_0314)
- 7.12.1.102 SDO Oversampling #1 Filter Coefficient (SDO_OSFC22_1, R/W, Address = 0xF900_0318)
- 7.12.1.103 SDO Oversampling #1 Filter Coefficient (SDO_OSFC23_1, R/W, Address = 0xF900_031C)

7.12.1.104 SDO Oversampling #2 filter coefficient (SDO_OSFC00_2, R/W, Address = 0xF900_0320)

SDO_OSFCN_2 (N is 00~23)	Bit	Description	Reset Value
osf_coef(2xN+1)	[23:16]~[27:16]	(2xN+1)'th oversampling filter coefficient. Signed 8~12bit integer. Actual value is osf_coef/(210)	Refer to below table
osf_coef(2xN)	[7:0] ~[11:0]	(2xN)'th oversampling filter coefficient. Signed 8~12bit integer. Actual value is osf_coef/(210)	

- 7.12.1.105 SDO Oversampling #2 Filter Coefficient (SDO_OSFC01_2, R/W, Address = 0xF900_0324)
- 7.12.1.106 SDO Oversampling #2 Filter Coefficient (SDO_OSFC02_2, R/W, Address = 0xF900_0328)
- 7.12.1.107 SDO Oversampling #2 Filter Coefficient (SDO_OSFC03_2, R/W, Address = 0xF900_032C)
- 7.12.1.108 SDO Oversampling #2 Filter Coefficient (SDO_OSFC04_2, R/W, Address = 0xF900_0330)
- 7.12.1.109 SDO Oversampling #2 Filter Coefficient (SDO_OSFC05_2, R/W, Address = 0xF900_0334)
- 7.12.1.110 SDO Oversampling #2 Filter Coefficient (SDO_OSFC06_2, R/W, Address = 0xF900_0338)
- 7.12.1.111 SDO Oversampling #2 Filter Coefficient (SDO_OSFC07_2, R/W, Address = 0xF900_033C)
- 7.12.1.112 SDO Oversampling #2 Filter Coefficient (SDO_OSFC08_2, R/W, Address = 0xF900_0340)
- 7.12.1.113 SDO Oversampling #2 Filter Coefficient (SDO_OSFC09_2, R/W, Address = 0xF900_0344)
- 7.12.1.114 SDO Oversampling #2 Filter Coefficient (SDO_OSFC10_2, R/W, Address = 0xF900_0348)
- 7.12.1.115 SDO Oversampling #2 Filter Coefficient (SDO_OSFC11_2, R/W, Address = 0xF900_034C)
- 7.12.1.116 SDO Oversampling #2 Filter Coefficient (SDO_OSFC12_2, R/W, Address = 0xF900_0350)
- 7.12.1.117 SDO Oversampling #2 Filter Coefficient (SDO_OSFC13_2, R/W, Address = 0xF900_0354)
- 7.12.1.118 SDO Oversampling #2 Filter Coefficient (SDO_OSFC14_2, R/W, Address = 0xF900_0358)
- 7.12.1.119 SDO Oversampling #2 Filter Coefficient (SDO_OSFC15_2, R/W, Address = 0xF900_035C)
- 7.12.1.120 SDO Oversampling #2 Filter Coefficient (SDO_OSFC16_2, R/W, Address = 0xF900_0360)
- 7.12.1.121 SDO Oversampling #2 Filter Coefficient (SDO_OSFC17_2, R/W, Address = 0xF900_0364)
- 7.12.1.122 SDO Oversampling #2 Filter Coefficient (SDO_OSFC18_2, R/W, Address = 0xF900_0368)
- 7.12.1.123 SDO Oversampling #2 Filter Coefficient (SDO_OSFC19_2, R/W, Address = 0xF900_036C)
- 7.12.1.124 SDO Oversampling #2 Filter Coefficient (SDO_OSFC20_2, R/W, Address = 0xF900_0370)
- 7.12.1.125 SDO Oversampling #2 Filter Coefficient (SDO_OSFC21_2, R/W, Address = 0xF900_0374)
- 7.12.1.126 SDO Oversampling #2 Filter Coefficient (SDO_OSFC22_2, R/W, Address = 0xF900_0378)
- 7.12.1.127 SDO Oversampling #2 Filter Coefficient (SDO_OSFC23_2, R/W, Address = 0xF900_037C)

4x Oversampling Case Coefficient (Flat response up to 6MHz) – Default reset value

Coefficient	Value	Coefficient	Value	Coefficient	Value
osf_coef00	-2	osf_coef16	-14	osf_coef32	-57
osf_coef01	-3	osf_coef17	-20	osf_coef33	-80
osf_coef02	0	osf_coef18	1	osf_coef34	5
osf_coef03	0	osf_coef19	0	osf_coef35	0
osf_coef04	4	osf_coef20	20	osf_coef36	86
osf_coef05	5	osf_coef21	29	osf_coef37	121
osf_coef06	-1	osf_coef22	-2	osf_coef38	-10
osf_coef07	0	osf_coef23	0	osf_coef39	0
osf_coef08	-6	osf_coef24	-28	osf_coef40	-154
osf_coef09	-9	osf_coef25	-40	osf_coef41	-212
osf_coef10	1	osf_coef26	2	osf_coef42	27
osf_coef11	0	osf_coef27	0	osf_coef43	0
osf_coef12	10	osf_coef28	40	osf_coef44	613
osf_coef13	14	osf_coef29	56	osf_coef45	651
osf_coef14	-1	osf_coef30	-3	osf_coef46	-308
osf_coef15	0	osf_coef31	0	osf_coef47	1024

2x Oversampling Case Coefficient (Flat response up to 12MHz)

Coefficient	Value	Coefficient	Value	Coefficient	Value
osf_coef00	0	osf_coef16	0	osf_coef32	-19
osf_coef01	0	osf_coef17	0	osf_coef33	0
osf_coef02	0	osf_coef18	0	osf_coef34	28
osf_coef03	0	osf_coef19	0	osf_coef35	0
osf_coef04	0	osf_coef20	0	osf_coef36	-39
osf_coef05	0	osf_coef21	0	osf_coef37	0
osf_coef06	0	osf_coef22	0	osf_coef38	55
osf_coef07	0	osf_coef23	0	osf_coef39	0
osf_coef08	0	osf_coef24	-3	osf_coef40	-79
osf_coef09	0	osf_coef25	0	osf_coef41	0
osf_coef10	0	osf_coef26	5	osf_coef42	120
osf_coef11	0	osf_coef27	0	osf_coef43	0
osf_coef12	0	osf_coef28	-8	osf_coef44	-211
osf_coef13	0	osf_coef29	0	osf_coef45	0
osf_coef14	0	osf_coef30	13	osf_coef46	650
osf_coef15	0	osf_coef31	0	osf_coef47	1024

7.12.1.128 SDO Channel Crosstalk Cancellation Coefficient for Ch. 0 (SDO_XTALK0, R/W, Address = 0xF900_0260)

SDO_XTALK0	Bit	Description	Initial State
Reserved	[31:24]	Reserved, read as zero, do not modify	0
xtalk_coef02	[23:16]	Signed 8 bit integer. Actual value is xtalk_coef02/(210) 0x7F : 0.124 0x7E : 0.123 ... 0x01 : 0.000977 0x00 : 0.000000 0xFF :-0.000977 0xFE :-0.001953 ... 0x81 :-0.124 0x80 : -0.125	00
Reserved	[15:8]	Reserved, read as zero, do not modify	0
xtalk_coef01	[7:0]	Same as xtalk_coef01	00

7.12.1.129 SDO Black Burst Control Register (SDO_BB_CTRL, R/W, Address = 0xF900_026C)

SDO_BB_CTRL	Bit	Description	Initial State
Reserved	[31:18]	Reserved, read as zero, do not modify	0
ref_bb_level	[17:8]	Black level setting value. It specifies the level during horizontal active video for black burst signal. The recommended values are NTSC : 0x11A (include 7.5 IRE setup) PAL : 0xFB (without setup)	0x11A
Reserved	[7:6]	Reserved, read as zero, do not modify	0
sel_bb_chan	[5:4]	Black burst (BB) test channel selection DAC0 DAC1 DAC2 00 = CVBS BB BB 01 = BB CVBS BB 10 = BB BB CVBS 11 = Reserved	00
Reserved	[3:1]	Reserved, read as zero, do not modify	0
BB mode	[0]	Black burst test mode enable. If set, entire bit fields in SDO_CONFIG register except 'Selection of Video Mux for DAC 0, 1, and 2' bit are discarded and SDO enters black burst test mode. In order to enable black burst test, 'Selection of Video Mux for DAC 0, 1, and 2' bit has to be set to zero.	0

7.12.1.130 SDO Interrupt Request Register (SDO_IRQ, R/W, Address = 0xF900_0280)

SDO_IRQ	Bit	Description	Initial State
Reserved	[31:1]	Reserved, read as zero, do not modify	0
Vsync Interrupt Request	[0]	0 = No interrupt 1 = Interrupt request pending (This interrupt is requested if TVOUT module generates the falling edge of vertical synchronization pulses at each field. Write 1 to reset this bit. Writing '0' has no effect.)	0

7.12.1.131 SDO Interrupt Request Masking Register (SDO_IRQMASK, R/W, Address = 0xF900_0284)

SDO_IRQMASK	Bit	Description	Initial State
Reserved	[31:1]	Reserved, read as zero, do not modify	0
Vsync Interrupt Request Masking	[0]	0 = Enables Interrupt request 1 = Disables Interrupt request (The status pending bit of SDO Interrupt Request Register is asserted even if the request is disabled. Only the request to MCU will be disabled.)	0

7.12.1.132 SDO Closed Caption Data Registers (SDO_ARMCC, R/W, Address = 0xF900_03C0)

SDO_ARMCC	Bit	Description	Initial State
Reserved	[31:16]	Reserved, read as zero, do not modify	0
Display Control Character of Closed Caption Data	[15:8]	Bit alignment of the Display Control Character register is in their incoming order. The first incoming bit becomes LSB, i.e. Display Control Character [7:0] = {p, b6, b5, b4, b3, b2, b1, b0}, where bn represents data bit with their incoming order n, and p denotes their odd parity bit.	0
Non Display Control Character of Closed Caption Data	[7:0]	Bit alignment of the Non Display Control Character register is in their incoming order. The first incoming bit becomes LSB, i.e. Non Display Control Character [7:0] = {p, b6, b5, b4, b3, b2, b1, b0}, where bn represents data bit with their incoming order n, and p denotes their odd parity bit.	0

NOTE: This register is used for European Caption as well as US Closed Caption.

7.12.1.133 SDO WSS 525 Data Registers (SDO_ARMWSS525, R/W, Address = 0xF900_03C4)

SDO_ARMWSS525	Bit	Description	Initial State
Reserved	[31:20]	Reserved, read as zero, do not modify	0
CRC of WSS 525 Data	[19:14]	Bit alignment of the CRC register is according to their incoming order. The first incoming bit becomes LSB, i.e. CRC [19:14] = {b19, b18, b17, b16, b15, b14}, where bn represents data bit with their incoming order n. The CRC used is $X^6 + X + 1$, all preset to 1.	0
Word 2 of WSS 525 Data	[13:6]	Bit alignment of the Word 2 register is according to their incoming order. The first incoming bit becomes LSB, i.e. Word 2 [13:6] = {b13, b12, b11, b10, b9, b8, b7, b6}, where bn represents data bit with their incoming order n. The Word 2 data are used for copy control: b7, b6 : 00 copying permitted 01 one copy permitted 10 reserved 11 no permission to copy b9 b8 : (reserved) b10 : 0 not analog pre-recorded medium 1 analog pre-recorded medium b13, b12, b11 : (reserved)	0
Word 1 of WSS 525 Data	[5:2]	Bit alignment of the Word 1 register is according to their incoming order. The first incoming bit becomes LSB, i.e. Word 1 [5:2] = {b5, b4, b3, b2}, where bn represents data bit with their incoming order n. The Word 1 data are used to indicate the existence of Word 2 data: b5, b4, b3, b2 : 0000 = copy control information 1111 = default	0
Word 0 of WSS 525 Data	[1:0]	Bit alignment of the Word 0 register is according to their incoming order. The first incoming bit becomes LSB, i.e. Word 0 [1:0] = {b1, b0}, where bn represents data bit with their incoming order n. The Word 0 data are used for display aspect ratio control: b1, b0 : 00 4:3 aspect ratio normal 01 16:9 aspect ratio anamorphic 10 4:3 aspect ratio letterbox 11 reserved	0

7.12.1.134 SDO WSS 625 Data Registers (SDO_ARMWSS625, R/W, Address = 0xF900_03C8)

SDO_ARMWSS625	Bit	Description	Initial State
Reserved	[31:14]	Reserved, read as zero, do not modify	0
Group D of WSS 625 Data	[13:11]	<p>Bit alignment of the Group D register is according to their incoming order. The first incoming bit becomes LSB, i.e. Group D [13:11] = {b13, b12, b11}, where bn represents data bit with their incoming order n.</p> <p>The Group D data are used for surround sound and copy control:</p> <ul style="list-style-type: none"> b11 : surround sound no yes b12 : copyright no copyright asserted or unknown copyright asserted b13 : copy protection copying not restricted copying restricted 	0
Group C of WSS 625 Data	[10:8]	<p>Bit alignment of the Group C register is according to their incoming order. The first incoming bit becomes LSB, i.e. Group C [10:8] = {b10, b9, b8}, where bn represents data bit with their incoming order n.</p> <p>The Group C data are used for subtitles:</p> <ul style="list-style-type: none"> b8 : teletext subtitles <ul style="list-style-type: none"> 0 no 1 yes b10 , b9 : open subtitles <ul style="list-style-type: none"> 00 no 01 inside active picture 10 outside active picture 11 reserved 	0

SDO_ARMWSS625	Bit	Description	Initial State																																																
Group B of WSS 625 Data	[7:4]	<p>Bit alignment of the Group B register is according to their incoming order. The first incoming bit becomes LSB, i.e. Group B [7:4] = {b7, b6, b5, b4}, where bn represents data bit with their incoming order n.</p> <p>The Group B data are used for enhanced video services:</p> <ul style="list-style-type: none"> b4 : mode <ul style="list-style-type: none"> 0 camera mode 1 film mode b5 : color encoding <ul style="list-style-type: none"> normal PAL Motion Adaptive ColorPlus b6 : helper signals <ul style="list-style-type: none"> not present present b7 : fixed to 0 	0																																																
Group A of WSS 625 Data	[3:0]	<p>Bit alignment of the Group A register is according to their incoming order. The first incoming bit becomes LSB, i.e. Group A [3:0] = {b3, b2, b1, b0}, where bn represents data bit with their incoming order n.</p> <p>The Group A data are used for display aspect ratio control:</p> <table border="0"> <tr> <td>b3, b2 b1, b0 :</td> <td>1000</td> <td>4:3</td> <td>full format</td> <td>-</td> <td>576 lines</td> </tr> <tr> <td></td> <td>0001</td> <td>14:9</td> <td>letterbox</td> <td>center</td> <td>504 lines</td> </tr> <tr> <td></td> <td>0010</td> <td>14:9</td> <td>letterbox</td> <td>top</td> <td>504 lines</td> </tr> <tr> <td></td> <td>1011</td> <td>16:9</td> <td>letterbox</td> <td>center</td> <td>430 lines</td> </tr> <tr> <td></td> <td>0100</td> <td>16:9</td> <td>letterbox</td> <td>top</td> <td>430 lines</td> </tr> <tr> <td></td> <td>1101</td> <td>>16:9</td> <td>letterbox</td> <td>center</td> <td>-</td> </tr> <tr> <td></td> <td>1110</td> <td>14:9</td> <td>full format</td> <td>center</td> <td>576 lines</td> </tr> <tr> <td></td> <td>0111</td> <td>16:9</td> <td>anamorphic</td> <td>-</td> <td>576 lines</td> </tr> </table>	b3, b2 b1, b0 :	1000	4:3	full format	-	576 lines		0001	14:9	letterbox	center	504 lines		0010	14:9	letterbox	top	504 lines		1011	16:9	letterbox	center	430 lines		0100	16:9	letterbox	top	430 lines		1101	>16:9	letterbox	center	-		1110	14:9	full format	center	576 lines		0111	16:9	anamorphic	-	576 lines	0
b3, b2 b1, b0 :	1000	4:3	full format	-	576 lines																																														
	0001	14:9	letterbox	center	504 lines																																														
	0010	14:9	letterbox	top	504 lines																																														
	1011	16:9	letterbox	center	430 lines																																														
	0100	16:9	letterbox	top	430 lines																																														
	1101	>16:9	letterbox	center	-																																														
	1110	14:9	full format	center	576 lines																																														
	0111	16:9	anamorphic	-	576 lines																																														

7.12.1.135 SDO CGMS-A 525 Data Registers (SDO_ARMCGMS525, R/W, Address = 0xF900_03CC)

SDO_ARMCGMS525	Bit	Description	Initial State
Reserved	[31:20]	Reserved, read as zero, do not modify	0
CRC of CGMS-A 525 Data	[19:14]	Bit alignment of the CRC register is according to their incoming order. The first incoming bit becomes LSB, i.e. CRC [19:14] = {b19, b18, b17, b16, b15, b14}, where bn represents data bit with their incoming order n. The CRC used is X^6+ X + 1, all preset to 1.	0
Word 2 of CGMS-A 525 Data	[13:6]	Bit alignment of the Word 2 register is according to their incoming order. The first incoming bit becomes LSB, i.e. Word 2 [13:6] = {b13, b12, b11, b10, b9, b8, b7, b6}, where bn represents data bit with their incoming order n. The Word 2 data are used for copy control: b7, b6 : 00 copying permitted 01 one copy permitted 10 reserved 11 no copying permitted b9 b8 : (reserved) b10 : 0 not analog pre-recorded medium 1 analog pre-recorded medium b13, b12, b11 : (reserved)	0
Word 1 of CGMS-A 525 Data	[5:2]	Bit alignment of the Word 1 register is according to their incoming order. The first incoming bit becomes LSB, i.e. Word 1 [5:2] = {b5, b4, b3, b2}, where bn represents data bit with their incoming order n. The Word 1 data are used to indicate the existence of Word 2 data: b5, b4, b3, b2 : 0000 copy control information 1111 default	0

SDO_ARMCGMS525	Bit	Description	Initial State																
Word 0 of CGMS-A 525 Data	[1:0]	<p>Bit alignment of the Word 0 register is according to their incoming order. The first incoming bit becomes LSB, i.e. Word 0 [1:0] = {b1, b0}, where bn represents data bit with their incoming order n.</p> <p>The Word 0 data are used for display aspect ratio control:</p> <table> <tr> <td>b1, b0 :</td> <td>00</td> <td>4:3 aspect ratio</td> <td>normal</td> </tr> <tr> <td></td> <td>01</td> <td>16:9 aspect ratio</td> <td>anamorphic</td> </tr> <tr> <td></td> <td>10</td> <td>4:3 aspect ratio</td> <td>letterbox</td> </tr> <tr> <td></td> <td>11</td> <td>reserved</td> <td></td> </tr> </table>	b1, b0 :	00	4:3 aspect ratio	normal		01	16:9 aspect ratio	anamorphic		10	4:3 aspect ratio	letterbox		11	reserved		0
b1, b0 :	00	4:3 aspect ratio	normal																
	01	16:9 aspect ratio	anamorphic																
	10	4:3 aspect ratio	letterbox																
	11	reserved																	

7.12.1.136 SDO CGMS-A 625 Data Registers (SDO_ARMCGMS625, R/W, Address = 0xF900_03D4)

SDO_ARMCGMS625	Bit	Description	Initial State
Reserved	[31:14]	Reserved, read as zero, do not modify	0
Group D of CGMS-A 625 Data	[13:11]	<p>Bit alignment of the Group D register is according to their incoming order. The first incoming bit becomes LSB, i.e. Group D [13:11] = {b13, b12, b11}, where bn represents data bit with their incoming order n.</p> <p>The Group D data are used for surround sound and copy control:</p> <ul style="list-style-type: none"> b11 : surround sound no yes b12 : copyright no copyright asserted or unknown copyright asserted b13 : copy protection copying not restricted copying restricted 	0
Group C of CGMS-A 625 Data	[10:8]	<p>Bit alignment of the Group C register is according to their incoming order. The first incoming bit becomes LSB, i.e. Group C [10:8] = {b10, b9, b8}, where bn represents data bit with their incoming order n.</p> <p>The Group C data are used for subtitles:</p> <ul style="list-style-type: none"> b8 : teletext subtitles 0 no 1 yes b10 , b9 : open subtitles 00 no 01 inside active picture 10 outside active picture 11 reserved 	0

SDO_ARMCGMS625	Bit	Description	Initial State																																																
Group B of CGMS-A 625 Data	[7:4]	<p>Bit alignment of the Group B register is according to their incoming order. The first incoming bit becomes LSB, i.e. Group B [7:4] = {b7, b6, b5, b4}, where bn represents data bit with their incoming order n.</p> <p>The Group B data are used for enhanced video services:</p> <ul style="list-style-type: none"> b4 : mode <ul style="list-style-type: none"> 0 camera mode 1 film mode b5 : color encoding <ul style="list-style-type: none"> normal PAL Motion Adaptive ColorPlus b6 : helper signals <ul style="list-style-type: none"> not present present B7 : fixed to 0 	0																																																
Group A of CGMS-A 625 Data	[3:0]	<p>Bit alignment of the Group A register is according to their incoming order. The first incoming bit becomes LSB, i.e. Group A [3:0] = {b3, b2, b1, b0}, where bn represents data bit with their incoming order n.</p> <p>The Group A data are used for display aspect ratio control:</p> <table border="0"> <tr> <td>b3, b2 b1, b0 :</td> <td>1000</td> <td>4:3</td> <td>full format</td> <td>-</td> <td>576 lines</td> </tr> <tr> <td></td> <td>0001</td> <td>14:9</td> <td>letterbox</td> <td>center</td> <td>504 lines</td> </tr> <tr> <td></td> <td>0010</td> <td>14:9</td> <td>letterbox</td> <td>top</td> <td>504 lines</td> </tr> <tr> <td></td> <td>1011</td> <td>16:9</td> <td>letterbox</td> <td>center</td> <td>430 lines</td> </tr> <tr> <td></td> <td>0100</td> <td>16:9</td> <td>letterbox</td> <td>top</td> <td>430 lines</td> </tr> <tr> <td></td> <td>1101</td> <td>>16:9</td> <td>letterbox</td> <td>center</td> <td>-</td> </tr> <tr> <td></td> <td>1110</td> <td>14:9</td> <td>full format</td> <td>center</td> <td>576 lines</td> </tr> <tr> <td></td> <td>0111</td> <td>16:9</td> <td>anamorphic</td> <td>-</td> <td>576 lines</td> </tr> </table>	b3, b2 b1, b0 :	1000	4:3	full format	-	576 lines		0001	14:9	letterbox	center	504 lines		0010	14:9	letterbox	top	504 lines		1011	16:9	letterbox	center	430 lines		0100	16:9	letterbox	top	430 lines		1101	>16:9	letterbox	center	-		1110	14:9	full format	center	576 lines		0111	16:9	anamorphic	-	576 lines	0
b3, b2 b1, b0 :	1000	4:3	full format	-	576 lines																																														
	0001	14:9	letterbox	center	504 lines																																														
	0010	14:9	letterbox	top	504 lines																																														
	1011	16:9	letterbox	center	430 lines																																														
	0100	16:9	letterbox	top	430 lines																																														
	1101	>16:9	letterbox	center	-																																														
	1110	14:9	full format	center	576 lines																																														
	0111	16:9	anamorphic	-	576 lines																																														

7.12.1.137 SDO Version Register (SDO_VERSION, R, Address = 0xF900_03D8)

SDO_VERSION	Bit	Description	Initial State
TVOUT version number	[31:0]	Read only register of TVOUT version number	C

7.12.2 SHADOW REGISTERS

7.12.2.1 SDO Closed Caption Data Shadow Register (SDO_CC, R/W, Address = 0xF900_0380)

SDO_CC	Bit	Description	Initial State
Reserved	[31:16]	Reserved, read as zero, do not modify	0
Display Control Character of Closed Caption Data	[15:8]	If MCU set values of SDO_ARMCC, the values are copied into this shadow register at the next vertical sync interrupt.	0
Non Display Control Character of Closed Caption Data	[7:0]	Do not set values to this shadow register.	0

7.12.2.2 SDO WSS 525 Data Shadow Registers (SDO_WSS525, R/W, Address = 0xF900_0384)

SDO_WSS525	Bit	Description	Initial State
Reserved	[31:20]	Reserved, read as zero, do not modify	0
CRC of WSS 525 Data	[19:14]	If MCU set values of SDO_ARMWSS525, the values are copied into this shadow register at the next vertical sync interrupt.	0
Word 2 of WSS 525 Data	[13:6]		0
Word 1 of WSS 525 Data	[5:2]		0
Word 0 of WSS 525 Data	[1:0]	Do not set values to this shadow register.	0

7.12.2.3 SDO WSS 625 Data Shadow Registers (SDO_WSS625, R/W, Address = 0xF900_0388)

SDO_ARMWSS625	Bit	Description	Initial State
Reserved	[31:14]	Reserved, read as zero, do not modify	0
Group D of WSS 625 Data	[13:11]	If MCU set values of SDO_ARMWSS625, the values are copied into this shadow register at the next vertical sync interrupt.	0
Group C of WSS 625 Data	[10:8]		0
Group B of WSS 625 Data	[7:4]		0
Group A of WSS 625 Data	[3:0]	Do not set values to this shadow register.	0

7.12.2.4 SDO CGMS-A 525 Data Shadow Registers (SDO_CGMS525, R/W, Address = 0xF900_038C)

SDO_CGMS525	Bit	Description	Initial State
Reserved	[31:20]	Reserved, read as zero, do not modify	0
CRC of CGMS-A 525 Data	[19:14]	If MCU set values of SDO_ARMCGMS525, the values are copied into this shadow register at the next vertical sync interrupt.	0
Word 2 of CGMS-A 525 Data	[13:6]		0
Word 1 of CGMS-A 525 Data	[5:2]		0
Word 0 of CGMS-A 525 Data	[1:0]	Do not set values to this shadow register.	0

7.12.2.5 SDO CGMS-A 625 Data Registers (SDO_CGMS625, R/W, Address = 0xF900_0394)

SDO_CGMS625	Bit	Description	Initial State
Reserved	[31:14]	Reserved, read as zero, do not modify	0
Group D of CGMS-A 625 Data	[13:11]	If MCU set values of SDO_ARMCGMS625, the values are copied into this shadow register at the next vertical sync interrupt.	0
Group C of CGMS-A 625 Data	[10:8]		0
Group B of CGMS-A 625 Data	[7:4]		0
Group A of CGMS-A 625 Data	[3:0]	Do not set values to this shadow register.	0

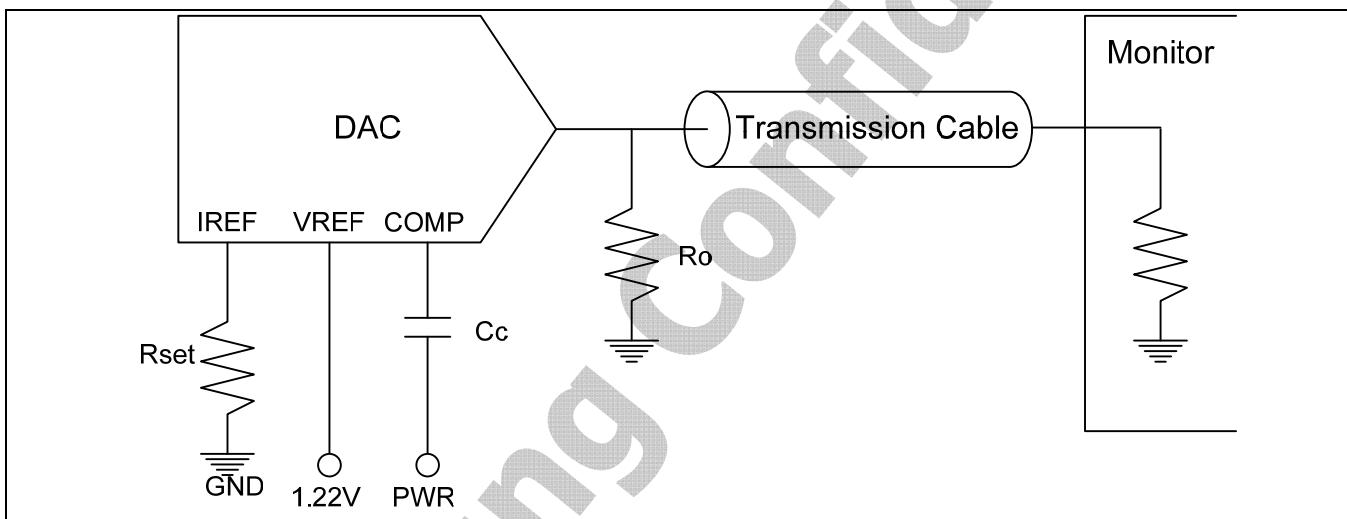
7.13 VIDEO DAC

7.13.1 GENERAL DESCRIPTION

The DAC is a 10-bit 1channel CMOS Digital-to-Analog converter for general applications. Its maximum conversion rate is 54MHz. It operates at analog power, 2.7V to 3.3V and provides full scale output currents of 26.7mA at one channel with 75 ohm load for 1.3V. The DAC has a power down mode to reduce power consumption during inactive periods.

7.13.2 FEATURES

- Maximum 54MHz Update Rate.
- 10-bit Current Output DAC
- 1.3Vpp Triple Output Compliance Range
- Internal Voltage Reference
- Fine Full Scale control: 91.1% ~ 114.8%
- Power Down Mode



7.13.3 CORE PORT DESCRIPTION

Table 7-5 Port Description of Video DAC

Name	Width	I/O	Description
VREF	-	AI	Voltage reference for DAC. An Internal voltage reference of nominally 1.22V is provided. Is driven with an external reference source.
IREF	-	AI	Full Scale Adjust control. The full-scale current drive on each of the output channels is determined by the value of a resistor RSET connected between this terminal and AVSS30A1.
IOUT1, IOUT2, IOUT3	-	AO	DAC current output. Full scale output is achieved if all input bits are set to binary 1.

COMP	-	AO	Compensation Pin. This is a compensation pin for the internal reference amplifier. A 0.1uF ceramic capacitor(Tolerance ± 10%) must be connected between COMP and AVDD30A1.
------	---	----	--

I/O Type Abbreviation.

- AI: Analog Input, DI: Digital Input, AO: Analog Output, AP: Analog Power, AG: Analog Ground,
- DP: Digital Power, DG: Digital Ground, LP: Logic Power, LG: Logic Ground

7.13.4 FULL SCALE VOLTAGE MODIFICATION

Table 7-6 Recommended RSET and RO According to Full Scale Voltage

Full Scale Voltage	RSET	RO
1.3V	1.2k(Tolerance ±1%) [ohm]	75(Tolerance ±5%) [ohm]
1.0V	1.54k(Tolerance ±1%) [ohm]	75(Tolerance ±5%) [ohm]

7.14 APPENDIX

7.14.1 VERTICAL BAR PHEOMENON

The goal of oversampling filter in TVOUT module is to obtain 54MHz sample-rate data for DAC from 13.5MHz/27MHz source sample-rate. The quality of interpolation result using FIR filter depends on the number of FIR filter taps. Unfortunately, ideal interpolation filter needs infinite number of taps, which cannot be implemented. Thus, practical consideration into the trade-off between interpolation quality and computational complexity is needed.

In TV application, it is the sub-carrier waveform that is the most sensitive to the quality of interpolation filter. Source rate of 13.5MHz for CVBS output is too coarse to represent the shape of sub-carrier and the quality of interpolation highly affects the shape of resulting 54 MHz sub-carriers. The error of sub-carrier induced by non-ideal interpolation filter is basically implies distortions in chrominance components. However, in the case of CVBS signal, the chrominance component and the luminance component are together to be mixed and transmitted via one channel. Then, parts of the interpolation error appears in luminance parts if the CVBS signal is separated into luminance component and chrominance components by the comb filter of TV decoder. According to an analysis on this interpolation error, the error pattern is repeated with a frequency of 102.3 kHz and the error spectrum is highly concentrated around 102.3 kHz. Thus, most of interpolation error is classified into luminance part by TV decoder and this error is shown in the shape of brightness distortion in TV monitor.

The oversampling filter in S5PV210 adopts 95-tap interpolation for CVBS signal in order to minimize the vertical bar phenomenon. Compared to 47-tap interpolation of legacy TVOUT version, the peak of vertical bar reduces by about 0.1 IRE assuming the use of 6th order Butterworth analog interpolation filter.

8

VIDEO PROCESSOR

8.1 OVERVIEW OF VIDEO PROCESSOR

Video processor (VP) is responsible for video scaling, de-interlacing, and video post processing of TV-out data path. VP reads reconstructed YCbCr 4:2:0 video sequences from DRAM, processes the sequence, and sends it to MIXER on-the-fly as shown in [Figure 8-1](#).

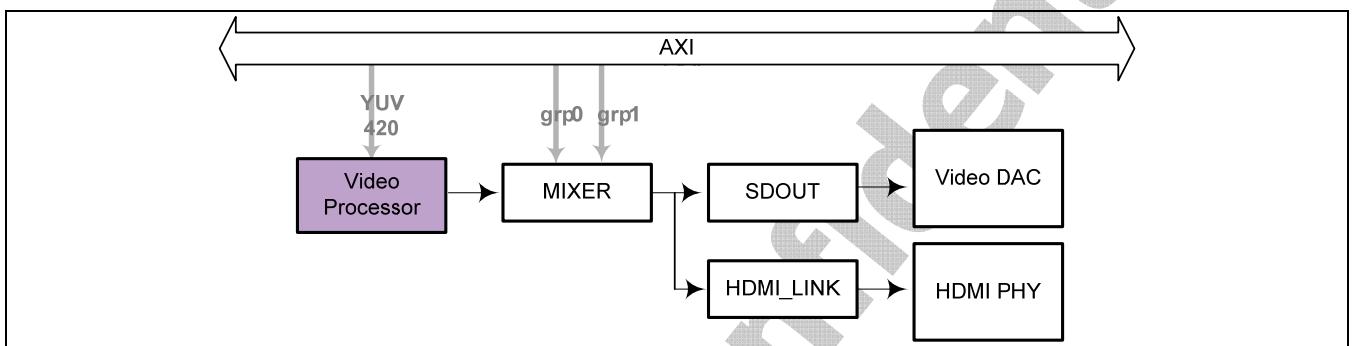


Figure 8-1 Video Data Path

8.1.1 KEY FEATURES OF VIDEO PROCESSOR

Input YCbCr sequence of VP is up to 1920x1080@30Hz. Basic features of VP are as follows:

- Supports BOB / TILE (YUV420 NV12 type, Note: refer to MFC user's manual for TILE)
- Input source size up to 1920x1080 (min : 32x4)
- Produces YCbCr 4:4:4 outputs to help MIXER to blend video and graphics
- Supports 1/4X to 16X vertical scaling with 4-tap/16-phase poly-phase filter
- Supports 1/4X to 16X horizontal scaling with 8-tap/16-phase poly-phase filter
- Supports Pan & Scan, Letterbox, and NTSC/PAL conversion using scaling
- Supports Flexible scaled video positioning within display area
- Supports 1/16 pixel resolution Pan&Scan mode
- Supports Flexible post video processing
 - Color saturation, Brightness/ Contrast enhancement, Edge enhancement for SDTV
 - Color space conversion between BT.601 and BT.709

8.2 BLOCK DIAGRAM OF VIDEO PROCESSOR

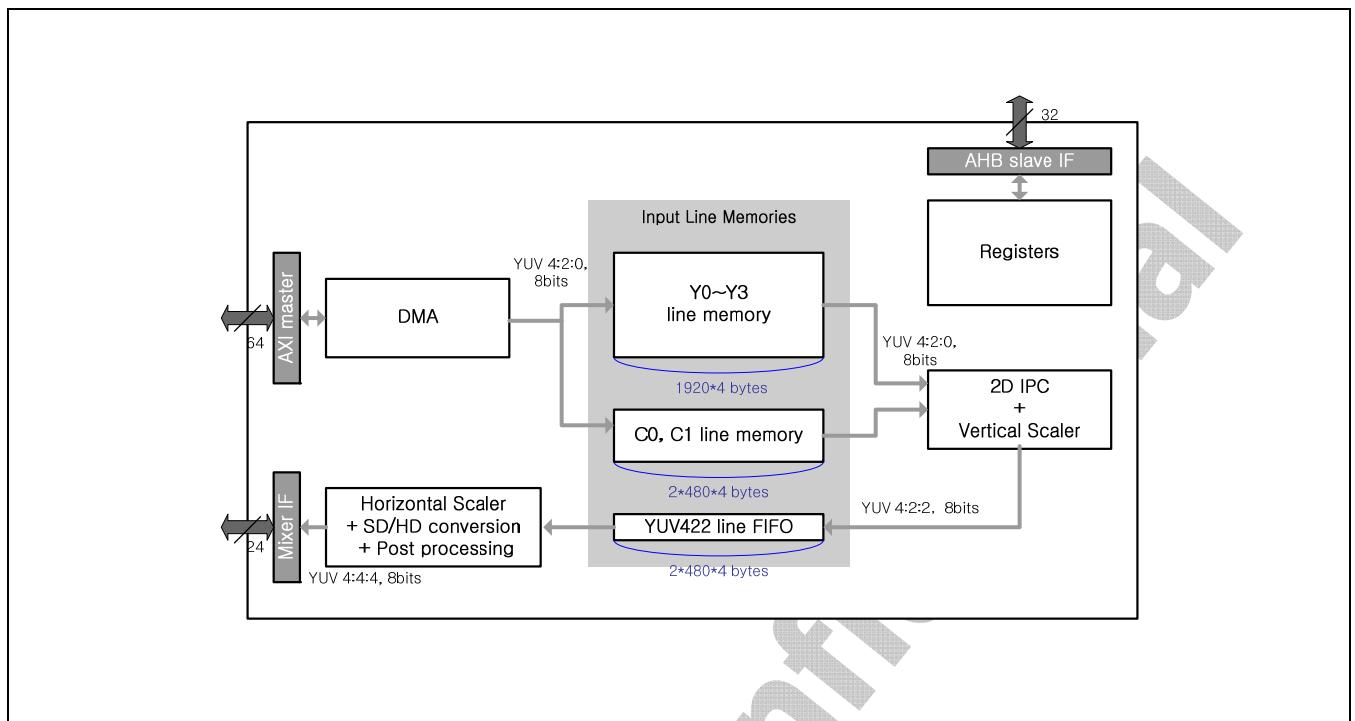


Figure 8-2 Block Diagram of Video Processor

Components in video processor:

- DMA: DMA reads the image from memory
- Input Line Memory: It stores data to process the image.
- Registers: The configuration of Video Processor
- 2D-IPC and Vertical Scaler: It performs IPC and vertical scaling.
- Horizontal Scaler: Horizontal scaling and post processing

8.3 FUNCTION DESCRIPTION OF VIDEO PROCESSOR

8.3.1 BOB IN VIDEO PROCESSOR

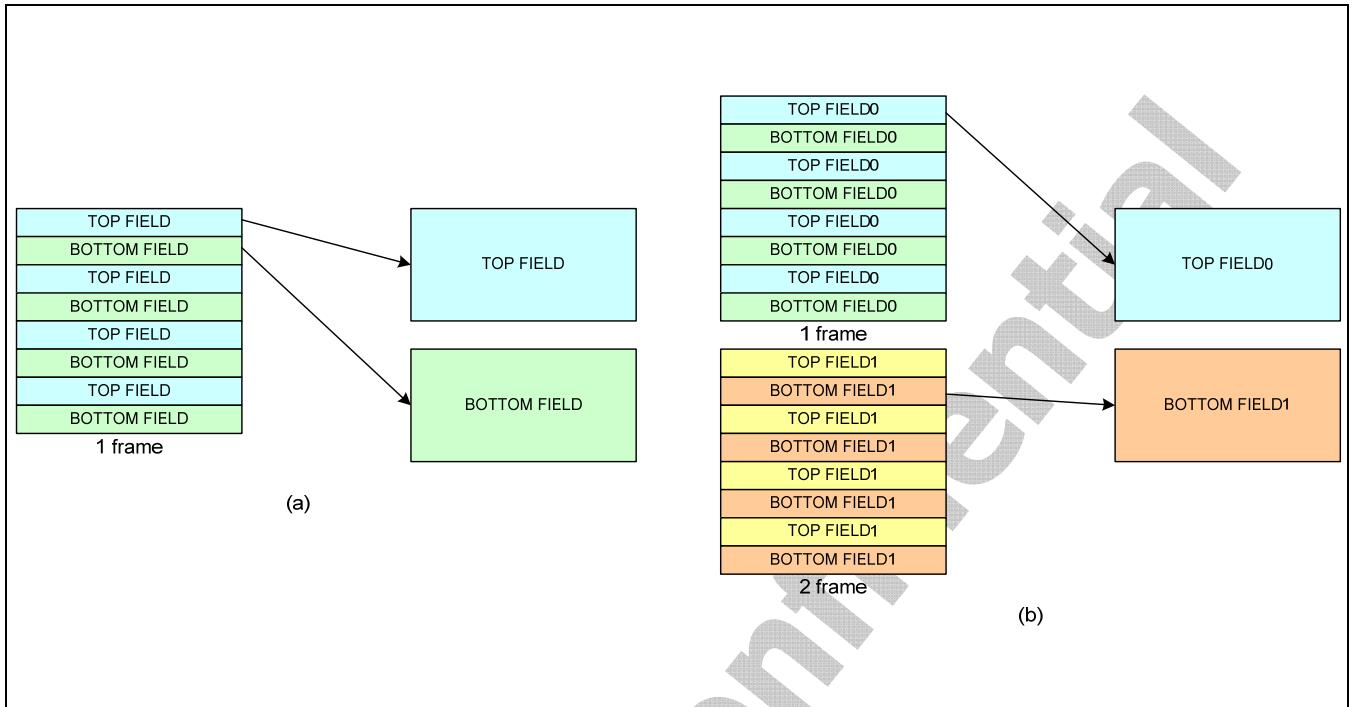


Figure 8-3 Data Type for BOB

In some applications, it is necessary to display an interlaced video signal on a non-interlaced display. Thus, some form of “de-interlacing” or “progressive can conversion” is required. Video mode is one of fundamental de-interlacing algorithm. Video mode de-interlacing can be further broken down into inter-field and intra-field processing. Particular, Intra-field processing in video mode is the simplest method to generate additional scan lines using only information in the original field. The computer industry has coined this technique as “BOB”.

BOB in VP consists of Intra-field or inter-field. Inter-field comes from a frame as shown in [Figure 8-3\(a\)](#). Also, Intra-field comes from two frames as shown in [Figure 8-3 \(b\)](#).

8.3.2 INTERLACE TO PROGRESSIVE CONVERSION

Interlace to Progressive Conversion (IPC) plays role to convert interlaced to progressive. It is distinguished with vertical x2 scale.

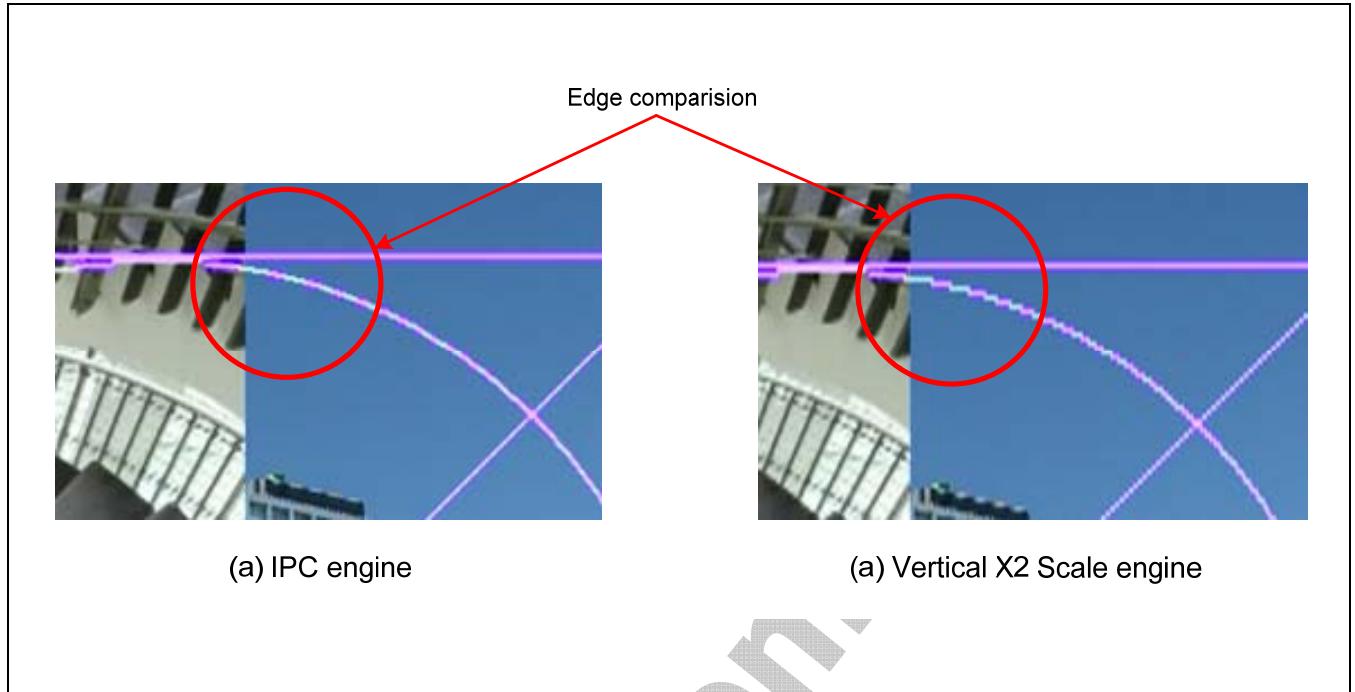


Figure 8-4 Difference Between IPC and X2 Scale-up

IPC engine executes “Edge Detection Function” which is based on the edge diagnosis method. This enables IPC to estimate edge line and display more natural image.

8.4 REGISTER DESCRIPTION

8.4.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
VP_ENABLE	0xF910_0000	R/W	Specifies the Power-Down Ready & Enable	0x0000_0002
VP_SRESET	0xF910_0004	R/W	Specifies the Software Reset	0x0000_0000
VP_SHADOW_UPDATE	0xF910_0008	R/W	Specifies the Shadow Register Update Enable	0x0000_0000
VP_FIELD_ID	0xF910_000C	R/W	Specifies the Field ID of the “Source” Image	0x0000_0000
VP_MODE	0xF910_0010	R/W	Specifies the VP Operation Mode	0x0000_0000
VP_IMG_SIZE_Y	0xF910_0014	R/W	Specifies the Luminance Date Size	0x0000_0000
VP_IMG_SIZE_C	0xF910_0018	R/W	Specifies the Chrominance Date Size	0x0000_0000
VP_TOP_Y_PTR	0xF910_0028	R/W	Specifies the Base Address for Y of Top Field (Frame)	0x0000_0000
VP_BOT_Y_PTR	0xF910_002C	R/W	Specifies the Base Address for Y of Bottom Field	0x0000_0000
VP_TOP_C_PTR	0xF910_0030	R/W	Specifies the Base Address for C of Top Field(frame)	0x0000_0000
VP_BOT_C_PTR	0xF910_0034	R/W	Specifies the Base Address for C of Bottom Field	0x0000_0000
VP_ENDIAN_MODE	0xF910_03CC	R/W	Specifies the Big/Little Endian Mode Selection	0x0000_0000
VP_SRC_H_POSITION	0xF910_0044	R/W	Specifies the Horizontal Offset in the Source Image	0x0000_0000
VP_SRC_V_POSITION	0xF910_0048	R/W	Specifies the Vertical Offset in the Source Image	0x0000_0000
VP_SRC_WIDTH	0xF910_004C	R/W	Specifies the Width of the Source Image	0x0000_0000
VP_SRC_HEIGHT	0xF910_0050	R/W	Specifies the Height of the Source Image	0x0000_0000
VP_DST_H_POSITION	0xF910_0054	R/W	Specifies the Horizontal Offset in the Display	0x0000_0000
VP_DST_V_POSITION	0xF910_0058	R/W	Specifies the Vertical Offset in the Display	0x0000_0000
VP_DST_WIDTH	0xF910_005C	R/W	Specifies the Width of the Display	0x0000_0000
VP_DST_HEIGHT	0xF910_0060	R/W	Specifies the Height of the Display	0x0000_0000
VP_H_RATIO	0xF910_0064	R/W	Specifies the Horizontal Zoom Ratio of SRC:DST	0x0000_0000
VP_V_RATIO	0xF910_0068	R/W	Specifies the Vertical Zoom Ratio of SRC:DST	0x0000_0000

Register	Address	R/W	Description	Reset Value
VP_POLY8_Y0_LL	0xF910_006C	R/W	Specifies the 8-Tap Poly-phase Filter Coefficients for Luminance Horizontal Scaling	0x0000_0000
VP_POLY8_Y0_LH	0xF910_0070	R/W	Specifies the 8-Tap Poly-phase Filter Coefficients for Luminance Horizontal Scaling	0x0000_0000
VP_POLY8_Y0_HL	0xF910_0074	R/W	Specifies the 8-Tap Poly-phase Filter Coefficients for Luminance Horizontal Scaling	0x0000_0000
VP_POLY8_Y0_HH	0xF910_0078	R/W	Specifies the 8-Tap Poly-phase Filter Coefficients for Luminance Horizontal Scaling	0x0000_0000
VP_POLY8_Y1_LL	0xF910_007C	R/W	Specifies the 8-Tap Poly-phase Filter Coefficients for Luminance Horizontal Scaling	0x0000_0000
VP_POLY8_Y1_LH	0xF910_0080	R/W	Specifies the 8-Tap Poly-phase Filter Coefficients for Luminance Horizontal Scaling	0x0000_0000
VP_POLY8_Y1_HL	0xF910_0084	R/W	Specifies the 8-Tap Poly-phase Filter Coefficients for Luminance Horizontal Scaling	0x0000_0000
VP_POLY8_Y1_HH	0xF910_0088	R/W	Specifies the 8-Tap Poly-phase Filter Coefficients for Luminance Horizontal Scaling	0x0000_0000
VP_POLY8_Y2_LL	0xF910_008C	R/W	Specifies the 8-Tap Poly-phase Filter Coefficients for Luminance Horizontal Scaling	0x0000_0000
VP_POLY8_Y2_LH	0xF910_0090	R/W	Specifies the 8-Tap Poly-phase Filter Coefficients for Luminance Horizontal Scaling	0x0000_0000
VP_POLY8_Y2_HL	0xF910_0094	R/W	Specifies the 8-Tap Poly-phase Filter Coefficients for Luminance Horizontal Scaling	0x0000_0000
VP_POLY8_Y2_HH	0xF910_0098	R/W	Specifies the 8-Tap Poly-phase Filter Coefficients for Luminance Horizontal Scaling	0x0000_0000
VP_POLY8_Y3_LL	0xF910_009C	R/W	Specifies the 8-Tap Poly-phase Filter Coefficients for Luminance Horizontal Scaling	0x0000_0000
VP_POLY8_Y3_LH	0xF910_00A0	R/W	Specifies the 8-Tap Poly-phase Filter Coefficients for Luminance Horizontal Scaling	0x0000_0000
VP_POLY8_Y3_HL	0xF910_00A4	R/W	Specifies the 8-Tap Poly-phase Filter Coefficients for Luminance Horizontal Scaling	0x0000_0000

Register	Address	R/W	Description	Reset Value
VP_POLY8_Y3_HH	0xF910_00A8	R/W	Specifies the 8-Tap Poly-phase Filter Coefficients for Luminance Horizontal Scaling	0x0000_0000
VP_POLY4_Y0_LL	0xF910_00EC	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Luminance Vertical Scaling	0x0000_0000
VP_POLY4_Y0_LH	0xF910_00F0	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Luminance Vertical Scaling	0x0000_0000
VP_POLY4_Y0_HL	0xF910_00F4	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Luminance Vertical Scaling	0x0000_0000
VP_POLY4_Y0_HH	0xF910_00F8	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Luminance Vertical Scaling	0x0000_0000
VP_POLY4_Y1_LL	0xF910_00FC	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Luminance Vertical Scaling	0x0000_0000
VP_POLY4_Y1_LH	0xF910_0100	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Luminance Vertical Scaling	0x0000_0000
VP_POLY4_Y1_HL	0xF910_0104	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Luminance Vertical Scaling	0x0000_0000
VP_POLY4_Y1_HH	0xF910_0108	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Luminance Vertical Scaling	0x0000_0000
VP_POLY4_Y2_LL	0xF910_010C	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Luminance Vertical Scaling	0x0000_0000
VP_POLY4_Y2_LH	0xF910_0110	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Luminance Vertical Scaling	0x0000_0000
VP_POLY4_Y2_HL	0xF910_0114	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Luminance Vertical Scaling	0x0000_0000
VP_POLY4_Y2_HH	0xF910_0118	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Luminance Vertical Scaling	0x0000_0000
VP_POLY4_Y3_LL	0xF910_011C	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Luminance Vertical Scaling	0x0000_0000
VP_POLY4_Y3_LH	0xF910_0120	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Luminance Vertical Scaling	0x0000_0000

Register	Address	R/W	Description	Reset Value
VP_POLY4_Y3_HL	0xF910_0124	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Luminance Vertical Scaling	0x0000_0000
VP_POLY4_Y3_HH	0xF910_0128	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Luminance Vertical Scaling	0x0000_0000
VP_POLY4_C0_LL	0xF910_012C	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Chrominance Horizontal Scaling	0x0000_0000
VP_POLY4_C0_LH	0xF910_0130	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Chrominance Horizontal Scaling	0x0000_0000
VP_POLY4_C0_HL	0xF910_0134	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Chrominance Horizontal Scaling	0x0000_0000
VP_POLY4_C0_HH	0xF910_0138	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Chrominance Horizontal Scaling	0x0000_0000
VP_POLY4_C1_LL	0xF910_013C	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Chrominance Horizontal Scaling	0x0000_0000
VP_POLY4_C1_LH	0xF910_0140	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Chrominance Horizontal Scaling	0x0000_0000
VP_POLY4_C1_HL	0xF910_0144	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Chrominance Horizontal Scaling	0x0000_0000
VP_POLY4_C1_HH	0xF910_0148	R/W	Specifies the 4-Tap Poly-phase Filter Coefficients for Chrominance Horizontal Scaling	0x0000_0000
PP_CSC_Y2Y_COEF	0xF910_01D4	R/W	Specifies the Y to Y CSC Coefficient Setting	0x0000_0000
PP_CSC_CB2Y_COEF	0xF910_01D8	R/W	Specifies the CB to Y CSC Coefficient Setting	0x0000_0000
PP_CSC_CR2Y_COEF	0xF910_01DC	R/W	Specifies the CR to Y CSC Coefficient Setting	0x0000_0000
PP_CSC_Y2CB_COEF	0xF910_01E0	R/W	Specifies the Y to Y CSC Coefficient Setting	0x0000_0000
PP_CSC_CB2CB_COEF	0xF910_01E4	R/W	Specifies the CB to Y CSC Coefficient Setting	0x0000_0000
PP_CSC_CR2CB_COEF	0xF910_01F0	R/W	Specifies the CR to Y CSC Coefficient Setting	0x0000_0000
PP_CSC_Y2CR_COEF	0xF910_01EC	R/W	Specifies the Y to Y CSC Coefficient Setting	0x0000_0000

Register	Address	R/W	Description	Reset Value
PP_CSC_CB2CR_COEF	0xF910_01E8	R/W	Specifies the CB to Y CSC Coefficient Setting	0x0000_0000
PP_CSC_CR2CR_COEF	0xF910_01F4	R/W	Specifies the CR to Y CSC Coefficient Setting	0x0000_0000
PP_BYPASS	0xF910_0200	R/W	Specifies the Disable the Post Image Processor	0x0000_0001
PP_SATURATION	0xF910_020C	R/W	Specifies the Color Saturation Factor	0x0000_0080
PP_SHARPNESS	0xF910_0210	R/W	Specifies the Control for the Edge Enhancement	0x0000_0500
PP_LINE_EQ0	0xF910_0218	R/W	Specifies the Line Equation for Contrast Duration "0"	0x0000_0000
PP_LINE_EQ1	0xF910_021C	R/W	Specifies the Line Equation for Contrast Duration "1"	0x0000_0000
PP_LINE_EQ2	0xF910_0220	R/W	Specifies the Line Equation for Contrast Duration "2"	0x0000_0000
PP_LINE_EQ3	0xF910_0224	R/W	Specifies the Line Equation for Contrast Duration "3"	0x0000_0000
PP_LINE_EQ4	0xF910_0228	R/W	Specifies the Line Equation for Contrast Duration "4"	0x0000_0000
PP_LINE_EQ5	0xF910_022C	R/W	Specifies the Line Equation for Contrast Duration "5"	0x0000_0000
PP_LINE_EQ6	0xF910_0230	R/W	Specifies the Line Equation for Contrast Duration "6"	0x0000_0000
PP_LINE_EQ7	0xF910_0234	R/W	Specifies the Line Equation for Contrast Duration "7"	0x0000_0000
PP_BRIGHT_OFFSET	0xF910_0238	R/W	Specifies the Brightness Offset Control for Y	0x0000_0000
PP_CSC_EN	0xF910_023C	R/W	Specifies the Color Space Conversion Control	0x0000_0002
VP_VERSION_INFO	0xF910_03FC	R	Specifies the VP Version Information	0x0000_0011
Shadow Register Description				
VP_FIELD_ID_S	0xF910_016C	R	Specifies the Field ID of the "Source" Image	0x0000_0000
VP_MODE_S	0xF910_0170	R	Specifies the VP Operation Mode	0x0000_0000
VP_IMG_SIZE_Y_S	0xF910_0174	R	Specifies the Luminance Date Tiled Size	0x0000_0000
VP_IMG_SIZE_C_S	0xF910_0178	R	Specifies the Chrominance Date Tiled Size	0x0000_0000
VP_TOP_Y_PTR_S	0xF910_0190	R	Specifies the Base Address for Y of Top Field	0x0000_0000
VP_BOT_Y_PTR_S	0xF910_0194	R	Specifies the Base Address for Y of Bottom Field	0x0000_0000

Register	Address	R/W	Description	Reset Value
VP_TOP_C_PTR_S	0xF910_0198	R	Specifies the Base Address for C of Top Frame	0x0000_0000
VP_BOT_C_PTR_S	0xF910_019C	R	Specifies the Base Address for C of Bottom field	0x0000_0000
VP_ENDIAN_MODE_S	0xF910_03EC	R	Specifies the Big/ Little Endian Mode Selection	0x0000_0000
VP_SRC_H_POSITION_S	0xF910_01AC	R	Specifies the Horizontal Offset in the Source Image	0x0000_0000
VP_SRC_V_POSITION_S	0xF910_01B0	R	Specifies the Vertical Offset in the Source Image	0x0000_0000
VP_SRC_WIDTH_S	0xF910_01B4	R	Specifies the Width of the Source Image	0x0000_0000
VP_SRC_HEIGHT_S	0xF910_01B8	R	Specifies the Height of the Source Image	0x0000_0000
VP_DST_H_POSITION_S	0xF910_01BC	R	Specifies the Horizontal Offset in the Display	0x0000_0000
VP_DST_V_POSITION_S	0xF910_01C0	R	Specifies the Vertical Offset in the Display	0x0000_0000
VP_DST_WIDTH_S	0xF910_01C4	R	Specifies the Width of the Display	0x0000_0000
VP_DST_HEIGHT_S	0xF910_01C8	R	Specifies the Height of the Display	0x0000_0000
VP_H_RATIO_S	0xF910_01CC	R	Specifies the Horizontal Zoom Ratio of SRC:DST	0x0000_0000
VP_V_RATIO_S	0xF910_01D0	R	Specifies the Vertical Zoom Ratio of SRC:DST	0x0000_0000
PP_BYPASS_S	0xF910_0258	R	Specifies the Disable the Post Image Processor	0x0000_0000
PP_SATURATION_S	0xF910_025C	R	Specifies the Color Saturation Factor	0x0000_0000
PP_SHARPNESS_S	0xF910_0260	R	Specifies the Control for the Edge Enhancement	0x0000_0000
PP_LINE_EQ0_S	0xF910_0268	R	Specifies the Line Equation for Contrast Duration “0”	0x0000_0000
PP_LINE_EQ1_S	0xF910_026C	R	Specifies the Line Equation for Contrast Duration “1”	0x0000_0000
PP_LINE_EQ2_S	0xF910_0270	R	Specifies the Line Equation for Contrast Duration “2”	0x0000_0000
PP_LINE_EQ3_S	0xF910_0274	R	Specifies the Line Equation for Contrast Duration “3”	0x0000_0000
PP_LINE_EQ4_S	0xF910_0278	R	Specifies the Line Equation for Contrast Duration “4”	0x0000_0000
PP_LINE_EQ5_S	0xF910_027C	R	Specifies the Line Equation for Contrast Duration “5”	0x0000_0000
PP_LINE_EQ6_S	0xF910_0280	R	Specifies the Line Equation for Contrast Duration “6”	0x0000_0000

Register	Address	R/W	Description	Reset Value
PP_LINE_EQ7_S	0xF910_0284	R	Specifies the Line Equation for Contrast Duration “7”	0x0000_0000
PP_BRIGHT_OFFSET_S	0xF910_0288	R	Specifies the Brightness Offset Control for Y	0x0000_0000
PP_CSC_EN_S	0xF910_028C	R	Specifies the Color Space Conversion Control	0x0000_0000
PP_CSC_Y2Y_COEF_S	0xF910_0290	R	Specifies the Y to Y CSC Coefficient Setting	0x0000_0000
PP_CSC_CB2Y_COEF_S	0xF910_0294	R	Specifies the CB to Y CSC Coefficient Setting	0x0000_0000
PP_CSC_CR2Y_COEF_S	0xF910_0298	R	Specifies the CR to Y CSC Coefficient Setting	0x0000_0000
PP_CSC_Y2CB_COEF_S	0xF910_029C	R	Specifies the Y to Y CSC Coefficient Setting	0x0000_0000
PP_CSC_CB2CB_COEF_S	0xF910_02A0	R	Specifies the CB to Y CSC Coefficient Setting	0x0000_0000
PP_CSC_CR2CB_COEF_S	0xF910_02AC	R	Specifies the CR to Y CSC Coefficient Setting	0x0000_0000
PP_CSC_Y2CR_COEF_S	0xF910_02A8	R	Specifies the Y to Y CSC Coefficient Setting	0x0000_0000
PP_CSC_CB2CR_COEF_S	0xF910_02A4	R	Specifies the CB to Y CSC Coefficient Setting	0x0000_0000
PP_CSC_CR2CR_COEF_S	0xF910_02B0	R	Specifies the CR to Y CSC Coefficient Setting	0x0000_0000

Video processor has special registers called “Shadow Register”. Software set the appropriate values to VP registers and this information are copied to the corresponding shadow registers if V-SYNC is invoked. Video processor is actually working according to these shadow registers.

8.4.1.1 Video Processor Enable/Disable Control Register (VP_ENABLE, Address = 0xF910_0000)

VP_ENABLE	Bit	Description	R/W	Initial State
Reserved	[31:3]	Reserved, read as zero, do not modify	R/W	0
VP_ON_S	[2]	This bit is read-only. Shadow bit of the bit [0]	R	0
VP_OPERATION_STATUS	[1]	This bit is read-only. 0 = VP is operating. 1 = VP is idle mode.	R	1
VP_ON	[0]	This bit is read-write. 0 = Disables 1 = Enables Note: The SFRs of Video Processor and Image Mixer is updated by Vertical Sync of TVENC's Timing Generator. Thus, SFRs are configured before this bit is enabled. The sequence to enable TVSS is as follows: "VP -> MIXER TVENC(HDMI)". Also, because SFRs are updated by Vertical Sync, the disabling sequence is following as : "VP -> MIXER -> TVNEC(HDMI)".	R/W	0

8.4.1.2 Video Processor Software Reset (VP_SRESET, R/W, Address = 0xF910_0004)

VP_SRESET	Bit	Description	Initial State
Reserved	[31:1]	Reserved, read as zero, do not modify	0
VP_SRESET	[0]	0 = Software reset is set and the last soft reset is complete. 1 = VP is processing software reset sequence.	0

8.4.1.3 Video Processor Shadow Register Update Enable Control Register (VP_SHADOW_UPDATE, W, Address = 0xF910_0008)

VP_SHADOW_UPDATE	Bit	Description	Initial State
Reserved	[31:1]	Reserved, read as zero, do not modify	0
VP_SHADOW_UPDATE	[0]	0 = Shadow registers are not updated at the rising edge of vertical sync. 1 = Shadow registers are updated and this register is cleared by H/W at the rising edge of vertical sync. (Shadow registers are listed in SHADOW REGISTER MAP table)	0

8.4.1.4 Video Processor Input Field ID Control Register (VP_FIELD_ID, R/W, Address = 0xF910_000C)

VP_FIELD_ID	Bit	Description	Initial State
Reserved	[31:1]	Reserved, read as zero, do not modify	0
VP_FIELD_ID	[0]	When VP_MODE[2] is set to 'high', this bit shows current FIELD information. Else, when VP_MODE[2] is set to 'low', this controls the pointer of top and bottom field. 0 = Top field 1 = Bottom field	0

8.4.1.5 Video Processor Operation Mode Control Register (VP_MODE, R/W, Address = 0xF910_0010)

VP_MODE	Bit	Description	Initial State
RTQoSTH	[31:24]	RTQoS threshold level configure. The Video Processor has the 192-depth internal DMA FIFO. Thus, you can adjust FIFO threshold level. 0 = Not available 1 ~ 191 = Threshold level 192 ~ 255 = Reserved	0
Reserved	[23:6]	Reserved, read as zero, do not modify	0
LINE_SKIP	[5]	This bit can control DMA operation. If it is set to '1', DMA skips a line per two lines while it reads line data. 0 = OFF 1 = ON	0
MEM_MODE	[4]	0 = Linear Mode 1 = Tile Mode (refer to MFC user's manual)	0
CROMA_EXPANSION	[3]	If it is set to '0', only refer to the chrominance of TOP filed. But set to '1', it uses the chrominance both TOP and BOTTOM. 0 = Using only C_TOP_PTR 1 = Using both C_TOP_PTR and C_BOT_PTR	0
FIELD_ID_AUTO_TOGGLING	[2]	0 = FIELD_ID is defined by user 1 = FIELD_ID is automatically toggled by V_SYNC DMA base address is changed by this bit. Note: VP_FIELD_ID_S register is toggled if this bit is 1, not VP_FIELD_ID	0
2D_IPC	[1]	Interlace to progressive conversion. VP displays progressive scan as using one filed image 0 = Disables 2D-IPC 1 = Enables 2D-IPC	0
Reserved	[0]	Reserved. It must be '0'	0

The guide of configuration

	LINE_SKIP	2D_IPC	FIELD_ID_AUTO_TOGGLE	FIELD_ID	Output
1. Interlace to Interlace	1 (On)	0 (Disable)	1 (Auto)	don't care	<p>VSYNC</p> <p>Field ID</p>
	0 (Off)	0 (Disable)	1 (Auto)	don't care	
2. Interlaced to Progressive					<p>VSYNC</p> <p>Field ID</p>
	1 (On)	1 (Enable)	0 (By user)	0: Top 1: Bottom	
3. Progressive to Interlace					<p>VSYNC</p> <p>Field ID</p>
	0 (Off)	1 (Enable)	0 (By user)	0: Top 1: Bottom	
4. Progressive to Progressive					<p>VSYNC</p> <p>Field ID</p>
	1 (On)	0 (Disable)	1 (Auto)	don't care	

Figure 8-5 Examples of Usage Cases

8.4.1.6 Video Processor Luminance Image Size Control Register (VP_IMG_SIZE_Y, R/W, Address = 0xF910_0014)

VP_IMG_SIZE_Y	Bit	Description	Initial State
Reserved	[31:30]	Reserved, read as zero, do not modify	0
VP_IMG_HSIZE_Y	[29:16]	Horizontal size of image (8~8192). (Without minus 1). LSB [2:0] must be 3'b000 for 64-bit interface. Zero value and values greater than 8192 are not allowed.	0
Reserved	[15:14]	Reserved, read as zero, do not modify	0
VP_IMG_VSIZE_Y	[13:0]	Vertical size of image (1 ~ 8192)	0

8.4.1.7 Video Processor Chrominance Image Size Control Register (VP_IMG_SIZE_C, R/W, Address = 0xF910_0018)

VP_IMG_SIZE_C	Bit	Description	Initial State
Reserved	[31:30]	Reserved, read as zero, do not modify	0
VP_IMG_HSIZE_C	[29:16]	Horizontal size of image (8~8192). (Without minus 1). LSB [2:0] must be 3'b000 for 64-bit interface. Zero value and values greater than 8192 are not allowed.	0
Reserved	[15:14]	Reserved, read as zero, do not modify	0
VP_IMG_VSIZE_C	[13:0]	Vertical size of image (1 ~ 8192)	0

8.4.1.8 Video Processor Top Luminance Picture Pointer Control Register (VP_TOP_Y_PTR, R/W, Address = 0xF910_0028)

VP_TOP_Y_PTR	Bit	Description	Initial State
VP_TOP_Y_PTR	[31:0]	Base address for luminance of top field. It should be integer multiples of 8. (LSB[2:0] must be 3'b000)	0

8.4.1.9 Video Processor Bottom Luminance Picture Pointer Control Register (VP_BOT_Y_PTR, R/W, Address = 0xF910_002C)

VP_TOP_C_PTR	Bit	Description	Initial State
VP_BOT_Y_PTR	[31:0]	Base address for luminance of bottom field. It should be integer multiples of 8. (LSB[2:0] must be 3'b000) If TILE mode is enable, $VP_BOT_Y_PTR = VP_TOP_Y_PTR + 0x40$	0

8.4.1.10 Video Processor Top Chrominance Picture Pointer Control Register (VP_TOP_C_PTR, R/W, Address = 0xF910_0030)

VP_CR_PTR	Bit	Description	Initial State
VP_TOP_C_PTR	[31:0]	Base address for chrominance of top field. It should be integer multiples of 8. (LSB[2:0] must be 3'b000)	0

8.4.1.11 Video Processor Bottom Chrominance Picture Pointer Control Register (VP_BOT_C_PTR, R/W, Address = 0xF910_0034)

VP_BOT_Y_PTR	Bit	Description	Initial State
VP_BOT_C_PTR	[31:0]	Base address for chrominance of bottom field. It should be integer multiples of 8. (LSB[2:0] must be 3'b000) If TILE mode is enable, $VP_BOT_C_PTR = VP_TOP_C_PTR + 0x40$	0

8.4.1.12 Video Processor Picture Endian Mode Control Register (VP_ENDIAN_MODE, R/W, Address = 0xF910_03CC)

VP_ENDIAN_MODE	Bit	Description	Initial State
Reserved	[31:1]	Reserved, read as zero, do not modify	0
VP_ENDIAN_MODE	[0]	0 = Big Endian 1 = Little Endian Refer to Figure 8-6	0

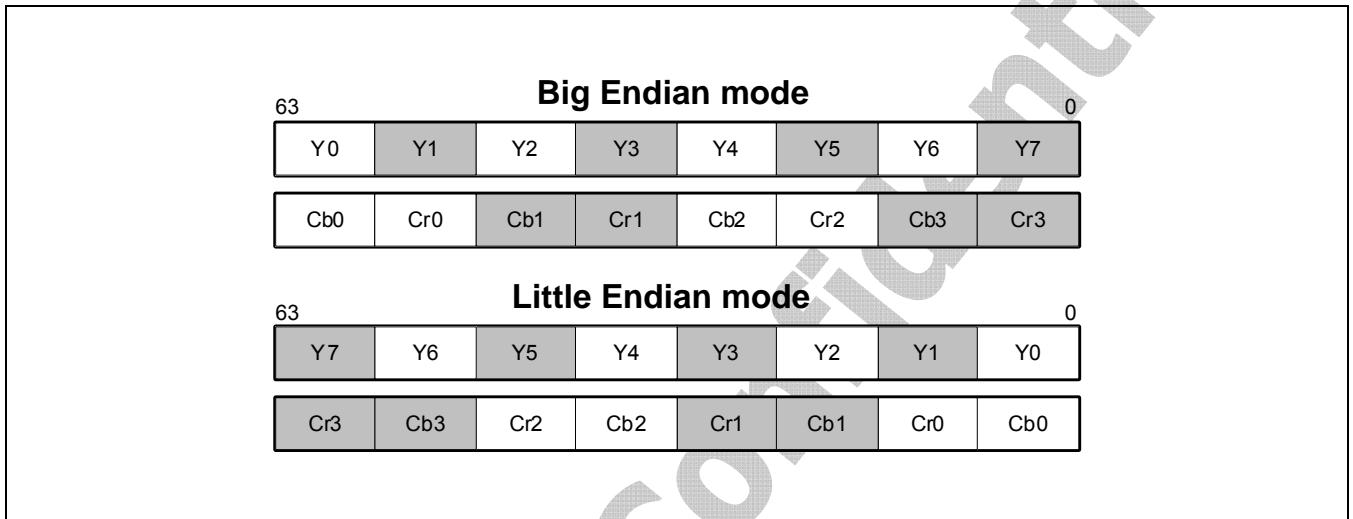


Figure 8-6 Endian Mode

8.4.1.13 Video Processor Horizontal Offset of Source Image Control Register (VP_SRC_H_POSITION, R/W, Address = 0xF910_0044)

VP_SRC_H_POSITION	Bit	Description	Initial State
Reserved	[31:15]	Reserved, read as zero, do not modify	0
VP_SRC_H_POSITION	[14:0]	<p>Horizontal offset in the source image - (11.4) format * For source image cropping, VP_SRC_H_POSITION + VP_SRC_WIDTH should be less than or equal to VP_IMG_HSIZE_Y Note: (11.4) format means that - '11' is an integer. - '4' is a fraction. Example) In case of H Position = 4 4(0x4(h)) = 0100(b)) is integer. Due to 4-bit fraction, 0100(b) is had to do 4 time left shift operation. As a result, register value is 4 * 2^4 = 64 = 0x40.</p>	0

8.4.1.14 Video Processor Vertical Offset of Source Image Control Register (VP_SRC_V_POSITION, R/W, Address = 0xF910_0048)

VP_SRC_V_POSITION	Bit	Description	Initial State
Reserved	[31:10]	Reserved, read as zero, do not modify	0
VP_SRC_V_POSITION	[10:0]	<p>Vertical offset in the source image. This value should be in the range between 0 and VP_SRC_HEIGHT. If LINE_SKIP is 1, VP_SRC_V_POSITION should be a half of that if LINE_SKIP is 0.</p>	0

8.4.1.15 Video Processor Width of Source Image Control Register (VP_SRC_WIDTH, R/W, Address = 0xF910_004C)

VP_SRC_WIDTH	Bit	Description	Initial State
Reserved	[31:11]	Reserved, read as zero, do not modify	0
VP_SRC_WIDTH	[10:0]	Width of the source image (MIN: 32)	0

8.4.1.16 Video Processor Height of Source Image Control Register (VP_SRC_HEIGHT, R/W, Address = 0xF910_0050)

VP_SRC_HEIGHT	Bit	Description	Initial State
Reserved	[31:10]	Reserved, read as zero, do not modify	0
VP_SRC_HEIGHT	[10:0]	Height of the source image If LINE_SKIP is 1, VP_SRC_HEIGHT should be a half of that if LINE_SKIP is 0. (MIN: 4)	0

8.4.1.17 Video Processor Horizontal Offset of Destination Image Control Register (VP_DST_H_POSITION, R/W, Address = 0xF910_0054)

VP_DST_H_POSITION	Bit	Description	Initial State
Reserved	[31:11]	Reserved, read as zero, do not modify	0
VP_DST_H_POSITION	[10:0]	Horizontal offset in the display	0

8.4.1.18 Video Processor Vertical Offset of Destination Image Control Register (VP_DST_V_POSITION, R/W, Address = 0xF910_0058)

VP_DST_V_POSITION	Bit	Description	Initial State
Reserved	[31:10]	Reserved, read as zero, do not modify	0
VP_DST_V_POSITION	[10:0]	Vertical offset in the display	0

8.4.1.19 Video Processor Width of Destination Image Control Register (VP_DST_WIDTH, R/W, Address = 0xF910_005C)

VP_DST_WIDTH	Bit	Description	Initial State
Reserved	[31:11]	Reserved, read as zero, do not modify	0
VP_DST_WIDTH	[10:0]	Width of the display	0

8.4.1.20 Video Processor Height of Destination Image Control Register (VP_DST_HEIGHT, R/W, Address = 0xF910_0060)

VP_DST_HEIGHT	Bit	Description	Initial State
Reserved	[31:10]	Reserved, read as zero, do not modify	0
VP_DST_HEIGHT	[10:0]	Height of the display	0

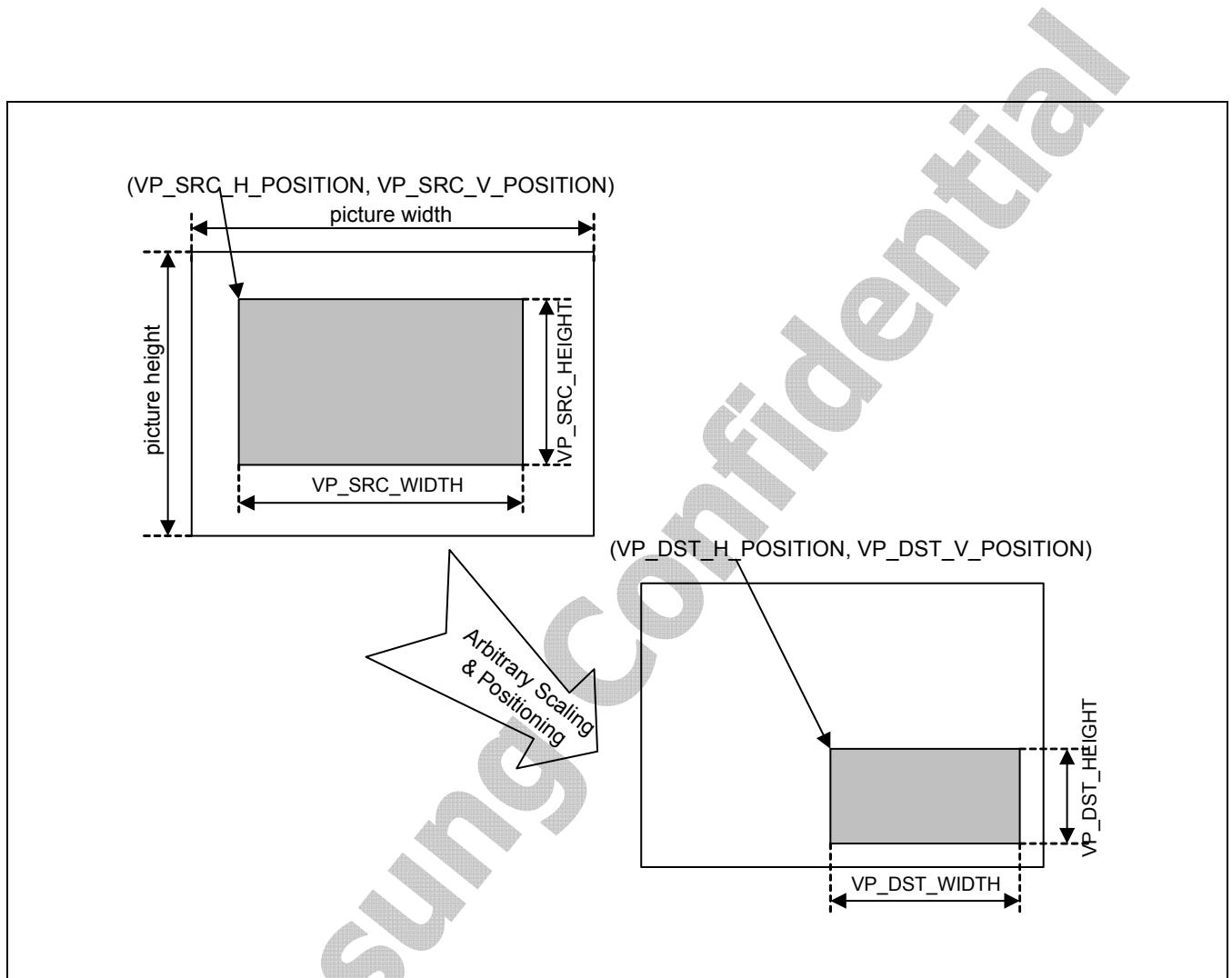


Figure 8-7 Video Scaling & Positioning on TV Display

8.4.1.21 Video Processor Horizontal Zoom Ratio (VP_H_RATIO, R/W, Address = 0xF910_0064)

VP_H_RATIO	Bit	Description	Initial State
Reserved	[31:19]	Reserved, read as zero, do not modify	0
VP_H_RATIO	[18:0]	<p>Horizontal zoom ratio of SRC:DST - 3.16 format Note: (3.16) format means that - '3' is an integer. - '16' is a fraction. Example) SRC : DST = 1 : 2 Because of 16-bit fraction, it is had to do 16 time left shift operation. As a result, register value is $1/2 * 2^{16} = 0x8000$</p>	0

8.4.1.22 Video Processor Vertical Zoom Ratio (VP_V_RATIO, R/W, Address = 0xF910_0068)

VP_V_RATIO	Bit	Description	Initial State
Reserved	[31:19]	Reserved, read as zero, do not modify	0
VP_V_RATIO	[18:0]	<p>Vertical zoom ratio of SRC:DST - 3.16 format This register should be as follows. (1) BOB mode, IPC disable $VP_V_RATIO = SRC / DST$ (2) BOB mode, IPC enable $VP_V_RATIO = 2 * SRC / DST$ (This is because destination line number is doubled by de-interlacing process itself) Note: (3.16) format means that - '3' is an integer. - '16' is a fraction. Example) SRC: DST = 1 : 2 Due to 16-bit fraction, it had to do 16 time left shift operation. As a result, register value is $1/2 * 2^{16} = 0x8000$</p>	0

8.4.1.23 Video Processor Luminance 8-tap Poly-phase Filter Coefficients (VP_POLY8_Y0_LL, R/W, Address = 0xF910_006C)

VP_POLY8_Y0_LL	Bit	Description	Initial State
Reserved	[31:27]	Reserved, read as zero, do not modify	0
vp_poly8_y0_ph0	[26:24]	Poly-phase Filter Coefficients	0
Reserved	[23:19]	Reserved, read as zero, do not modify	0
vp_poly8_y0_ph1	[18:16]	Poly-phase Filter Coefficients	0
Reserved	[15:11]	Reserved, read as zero, do not modify	0
vp_poly8_y0_ph2	[10:8]	Poly-phase Filter Coefficients	0
Reserved	[7:3]	Reserved, read as zero, do not modify	0
vp_poly8_y0_ph3	[2:0]	Poly-phase Filter Coefficients	0

8.4.1.24 Video Processor Luminance 8-tap Poly-phase Filter Coefficients (VP_POLY8_Y0_LH, R/W, Address = 0xF910_0070)

VP_POLY8_Y0_LH	Bit	Description	Initial State
Reserved	[31:27]	Reserved, read as zero, do not modify	0
vp_poly8_y0_ph4	[26:24]	Poly-phase Filter Coefficients	0
Reserved	[23:19]	Reserved, read as zero, do not modify	0
vp_poly8_y0_ph5	[18:16]	Poly-phase Filter Coefficients	0
Reserved	[15:11]	Reserved, read as zero, do not modify	0
vp_poly8_y0_ph6	[10:8]	Poly-phase Filter Coefficients	0
Reserved	[7:3]	Reserved, read as zero, do not modify	0
vp_poly8_y0_ph7	[2:0]	Poly-phase Filter Coefficients	0

8.4.1.25 Video Processor Luminance 8-tap Poly-phase Filter Coefficients (VP_POLY8_Y0_HL, R/W, Address = 0xF910_0074)

VP_POLY8_Y0_HL	Bit	Description	Initial State
Reserved	[31:27]	Reserved, read as zero, do not modify	0
vp_poly8_y0_ph8	[26:24]	Poly-phase Filter Coefficients	0
Reserved	[23:19]	Reserved, read as zero, do not modify	0
vp_poly8_y0_ph9	[18:16]	Poly-phase Filter Coefficients	0
Reserved	[15:11]	Reserved, read as zero, do not modify	0
vp_poly8_y0_ph10	[10:8]	Poly-phase Filter Coefficients	0
Reserved	[7:3]	Reserved, read as zero, do not modify	0
vp_poly8_y0_ph11	[2:0]	Poly-phase Filter Coefficients	0

8.4.1.26 Video Processor Luminance 8-tap Poly-phase Filter Coefficients (VP_POLY8_Y0_HH, R/W, Address = 0xF910_0078)

VP_POLY8_Y0_HH	Bit	Description	Initial State
Reserved	[31:27]	Reserved, read as zero, do not modify	0
vp_poly8_y0_ph12	[26:24]	Poly-phase Filter Coefficients	0
Reserved	[23:19]	Reserved, read as zero, do not modify	0
vp_poly8_y0_ph13	[18:16]	Poly-phase Filter Coefficients	0
Reserved	[15:11]	Reserved, read as zero, do not modify	0
vp_poly8_y0_ph14	[10:8]	Poly-phase Filter Coefficients	0
Reserved	[7:3]	Reserved, read as zero, do not modify	0
vp_poly8_y0_ph15	[2:0]	Poly-phase Filter Coefficients	0

8.4.1.27 Video Processor Luminance 8-tap Poly-phase Filter Coefficients (VP_POLY8_Y1_LL, R/W, Address = 0xF910_007C)

VP_POLY8_Y1_LL	Bit	Description	Initial State
Reserved	[31:29]	Reserved, read as zero, do not modify	0
vp_poly8_y1_ph0	[28:24]	Poly-phase Filter Coefficients	0
Reserved	[23:21]	Reserved, read as zero, do not modify	0
vp_poly8_y1_ph1	[20:16]	Poly-phase Filter Coefficients	0
Reserved	[15:13]	Reserved, read as zero, do not modify	0
vp_poly8_y1_ph2	[12:8]	Poly-phase Filter Coefficients	0
Reserved	[7:5]	Reserved, read as zero, do not modify	0
vp_poly8_y1_ph3	[4:0]	Poly-phase Filter Coefficients	0

8.4.1.28 Video Processor Luminance 8-tap Poly-phase Filter Coefficients (VP_POLY8_Y1_LH, R/W, Address = 0xF910_0080)

VP_POLY8_Y1_LH	Bit	Description	Initial State
Reserved	[31:29]	Reserved, read as zero, do not modify	0
vp_poly8_y1_ph4	[28:24]	Poly-phase Filter Coefficients	0
Reserved	[23:21]	Reserved, read as zero, do not modify	0
vp_poly8_y1_ph5	[20:16]	Poly-phase Filter Coefficients	0
Reserved	[15:13]	Reserved, read as zero, do not modify	0
vp_poly8_y1_ph6	[12:8]	Poly-phase Filter Coefficients	0
Reserved	[7:5]	Reserved, read as zero, do not modify	0
vp_poly8_y1_ph7	[4:0]	Poly-phase Filter Coefficients	0

8.4.1.29 Video Processor Luminance 8-tap Poly-phase Filter Coefficients (VP_POLY8_Y1_HL, R/W, Address = 0xF910_0084)

VP_POLY8_Y1_HL	Bit	Description	Initial State
Reserved	[31:29]	Reserved, read as zero, do not modify	0
vp_poly8_y1_ph8	[28:24]	Poly-phase Filter Coefficients	0
Reserved	[23:21]	Reserved, read as zero, do not modify	0
vp_poly8_y1_ph9	[20:16]	Poly-phase Filter Coefficients	0
Reserved	[15:13]	Reserved, read as zero, do not modify	0
vp_poly8_y1_ph10	[12:8]	Poly-phase Filter Coefficients	0
Reserved	[7:5]	Reserved, read as zero, do not modify	0
vp_poly8_y1_ph11	[4:0]	Poly-phase Filter Coefficients	0

8.4.1.30 Video Processor Luminance 8-tap Poly-phase Filter Coefficients (VP_POLY8_Y1_HH, R/W, Address = 0xF910_0088)

VP_POLY8_Y1_HH	Bit	Description	Initial State
Reserved	[31:29]	Reserved, read as zero, do not modify	0
vp_poly8_y1_ph12	[28:24]	Poly-phase Filter Coefficients	0
Reserved	[23:21]	Reserved, read as zero, do not modify	0
vp_poly8_y1_ph13	[20:16]	Poly-phase Filter Coefficients	0
Reserved	[15:13]	Reserved, read as zero, do not modify	0
vp_poly8_y1_ph14	[12:8]	Poly-phase Filter Coefficients	0
Reserved	[7:5]	Reserved, read as zero, do not modify	0
vp_poly8_y1_ph15	[4:0]	Poly-phase Filter Coefficients	0

8.4.1.31 Video Processor Luminance 8-tap Poly-phase Filter Coefficients (VP_POLY8_Y2_LL, R/W, Address = 0xF910_008C)

VP_POLY8_Y2_LL	Bit	Description	Initial State
Reserved	[31]	Reserved, read as zero, do not modify	0
vp_poly8_y2_ph0	[30:24]	Poly-phase Filter Coefficients	0
Reserved	[23]	Reserved, read as zero, do not modify	0
vp_poly8_y2_ph1	[22:16]	Poly-phase Filter Coefficients	0
Reserved	[15]	Reserved, read as zero, do not modify	0
vp_poly8_y2_ph2	[14:8]	Poly-phase Filter Coefficients	0
Reserved	[7]	Reserved, read as zero, do not modify	0
vp_poly8_y2_ph3	[6:0]	Poly-phase Filter Coefficients	0

8.4.1.32 Video Processor Luminance 8-tap Poly-phase Filter Coefficients (VP_POLY8_Y2_LH, R/W, Address = 0xF910_0090)

VP_POLY8_Y2_LH	Bit	Description	Initial State
Reserved	[31]	Reserved, read as zero, do not modify	0
vp_poly8_y2_ph4	[30:24]	Poly-phase Filter Coefficients	0
Reserved	[23]	Reserved, read as zero, do not modify	0
vp_poly8_y2_ph5	[22:16]	Poly-phase Filter Coefficients	0
Reserved	[15]	Reserved, read as zero, do not modify	0
vp_poly8_y2_ph6	[14:8]	Poly-phase Filter Coefficients	0
Reserved	[7]	Reserved, read as zero, do not modify	0
vp_poly8_y2_ph7	[6:0]	Poly-phase Filter Coefficients	0

8.4.1.33 Video Processor Luminance 8-tap Poly-phase Filter Coefficients (VP_POLY8_Y2_HL, R/W, Address = 0xF910_0094)

VP_POLY8_Y2_HL	Bit	Description	Initial State
Reserved	[31]	Reserved, read as zero, do not modify	0
vp_poly8_y2_ph8	[30:24]	Poly-phase Filter Coefficients	0
Reserved	[23]	Reserved, read as zero, do not modify	0
vp_poly8_y2_ph9	[22:16]	Poly-phase Filter Coefficients	0
Reserved	[15]	Reserved, read as zero, do not modify	0
vp_poly8_y2_ph10	[14:8]	Poly-phase Filter Coefficients	0
Reserved	[7]	Reserved, read as zero, do not modify	0
vp_poly8_y2_ph11	[6:0]	Poly-phase Filter Coefficients	0

8.4.1.34 Video Processor Luminance 8-tap Poly-phase Filter Coefficients (VP_POLY8_Y2_HH, R/W, Address = 0xF910_0098)

VP_POLY8_Y2_HH	Bit	Description	Initial State
Reserved	[31]	Reserved, read as zero, do not modify	0
vp_poly8_y2_ph12	[30:24]	Poly-phase Filter Coefficients	0
Reserved	[23]	Reserved, read as zero, do not modify	0
vp_poly8_y2_ph13	[22:16]	Poly-phase Filter Coefficients	0
Reserved	[15]	Reserved, read as zero, do not modify	0
vp_poly8_y2_ph14	[14:8]	Poly-phase Filter Coefficients	0
Reserved	[7]	Reserved, read as zero, do not modify	0
vp_poly8_y2_ph15	[6:0]	Poly-phase Filter Coefficients	0

8.4.1.35 Video Processor Luminance 8-tap Poly-phase Filter Coefficients (VP_POLY8_Y3_LL, R/W, Address = 0xF910_009C)

VP_POLY8_Y3_LL	Bit	Description	Initial State
Reserved	[31]	Reserved, read as zero, do not modify	0
vp_poly8_y3_ph0	[30:24]	Poly-phase Filter Coefficients	0
Reserved	[23]	Reserved, read as zero, do not modify	0
vp_poly8_y3_ph1	[22:16]	Poly-phase Filter Coefficients	0
Reserved	[15]	Reserved, read as zero, do not modify	0
vp_poly8_y3_ph2	[14:8]	Poly-phase Filter Coefficients	0
Reserved	[7]	Reserved, read as zero, do not modify	0
vp_poly8_y3_ph3	[6:0]	Poly-phase Filter Coefficients	0

8.4.1.36 Video Processor Luminance 8-tap Poly-phase Filter Coefficients (VP_POLY8_Y3_LH R/W, Address = 0xF910_00A0)

VP_POLY8_Y3_LH	Bit	Description	Initial State
Reserved	[31]	Reserved, read as zero, do not modify	0
vp_poly8_y3_ph4	[30:24]	Poly-phase Filter Coefficients	0
Reserved	[23]	Reserved, read as zero, do not modify	0
vp_poly8_y3_ph5	[22:16]	Poly-phase Filter Coefficients	0
Reserved	[15]	Reserved, read as zero, do not modify	0
vp_poly8_y3_ph6	[14:8]	Poly-phase Filter Coefficients	0
Reserved	[7]	Reserved, read as zero, do not modify	0
vp_poly8_y3_ph7	[6:0]	Poly-phase Filter Coefficients	0

8.4.1.37 Video Processor Luminance 8-tap Poly-phase Filter Coefficients (VP_POLY8_Y3_HL, R/W, Address = 0xF910_00A4)

VP_POLY8_Y3_HL	Bit	Description	Initial State
Reserved	[31]	Reserved, read as zero, do not modify	0
vp_poly8_y3_ph8	[30:24]	Poly-phase Filter Coefficients	0
Reserved	[23]	Reserved, read as zero, do not modify	0
vp_poly8_y3_ph9	[22:16]	Poly-phase Filter Coefficients	0
Reserved	[15]	Reserved, read as zero, do not modify	0
vp_poly8_y3_ph10	[14:8]	Poly-phase Filter Coefficients	0
Reserved	[7]	Reserved, read as zero, do not modify	0
vp_poly8_y3_ph11	[6:0]	Poly-phase Filter Coefficients	0

8.4.1.38 Video Processor Luminance 8-tap Poly-phase Filter Coefficients (VP_POLY8_Y3_HH, R/W, Address = 0xF910_00A8)

VP_POLY8_Y3_HH	Bit	Description	Initial State
Reserved	[31]	Reserved, read as zero, do not modify	0
vp_poly8_y3_ph12	[30:24]	Poly-phase Filter Coefficients	0
Reserved	[23]	Reserved, read as zero, do not modify	0
vp_poly8_y3_ph13	[22:16]	Poly-phase Filter Coefficients	0
Reserved	[15]	Reserved, read as zero, do not modify	0
vp_poly8_y3_ph14	[14:8]	Poly-phase Filter Coefficients	0
Reserved	[7]	Reserved, read as zero, do not modify	0
vp_poly8_y3_ph15	[6:0]	Poly-phase Filter Coefficients	0

8.4.1.39 Video Processor Luminance 4-tap Poly-phase Filter Coefficients (VP_POLY4_Y0_LL, R/W, Address = 0xF910_00EC)

VP_POLY4_Y0_LL	Bit	Description	Initial State
Reserved	[31:30]	Reserved, read as zero, do not modify	0
vp_poly4_y0_ph0	[29:24]	Poly-phase Filter Coefficients	0
Reserved	[23:22]	Reserved, read as zero, do not modify	0
vp_poly4_y0_ph1	[21:16]	Poly-phase Filter Coefficients	0
Reserved	[15:14]	Reserved, read as zero, do not modify	0
vp_poly4_y0_ph2	[13:8]	Poly-phase Filter Coefficients	0
Reserved	[7:6]	Reserved, read as zero, do not modify	0
vp_poly4_y0_ph3	[5:0]	Poly-phase Filter Coefficients	0

8.4.1.40 Video Processor Luminance 4-tap Poly-phase Filter Coefficients (VP_POLY4_Y0_LH, R/W, Address = 0xF910_00F0)

VP_POLY4_Y0_LH	Bit	Description	Initial State
Reserved	[31:30]	Reserved, read as zero, do not modify	0
vp_poly4_y0_ph4	[29:24]	Poly-phase Filter Coefficients	0
Reserved	[23:22]	Reserved, read as zero, do not modify	0
vp_poly4_y0_ph5	[21:16]	Poly-phase Filter Coefficients	0
Reserved	[15:14]	Reserved, read as zero, do not modify	0
vp_poly4_y0_ph6	[13:8]	Poly-phase Filter Coefficients	0
Reserved	[7:6]	Reserved, read as zero, do not modify	0
vp_poly4_y0_ph7	[5:0]	Poly-phase Filter Coefficients	0

8.4.1.41 Video Processor Luminance 4-tap Poly-phase Filter Coefficients (VP_POLY4_Y0_HL, R/W, Address = 0xF910_00F4)

VP_POLY4_Y0_HL	Bit	Description	Initial State
Reserved	[31:30]	Reserved, read as zero, do not modify	0
vp_poly4_y0_ph8	[29:24]	Poly-phase Filter Coefficients	0
Reserved	[23:22]	Reserved, read as zero, do not modify	0
vp_poly4_y0_ph9	[21:16]	Poly-phase Filter Coefficients	0
Reserved	[15:14]	Reserved, read as zero, do not modify	0
vp_poly4_y0_ph10	[13:8]	Poly-phase Filter Coefficients	0
Reserved	[7:6]	Reserved, read as zero, do not modify	0
vp_poly4_y0_ph11	[5:0]	Poly-phase Filter Coefficients	0

8.4.1.42 Video Processor Luminance 4-tap Poly-phase Filter Coefficients (VP_POLY4_Y0_HH, R/W, Address = 0xF910_00F8)

VP_POLY4_Y0_HH	Bit	Description	Initial State
Reserved	[31:30]	Reserved, read as zero, do not modify	0
vp_poly4_y0_ph12	[29:24]	Poly-phase Filter Coefficients	0
Reserved	[23:22]	Reserved, read as zero, do not modify	0
vp_poly4_y0_ph13	[21:16]	Poly-phase Filter Coefficients	0
Reserved	[15:14]	Reserved, read as zero, do not modify	0
vp_poly4_y0_ph14	[13:8]	Poly-phase Filter Coefficients	0
Reserved	[7:6]	Reserved, read as zero, do not modify	0
vp_poly4_y0_ph15	[5:0]	Poly-phase Filter Coefficients	0

8.4.1.43 Video Processor Luminance 4-tap Poly-phase Filter Coefficients (VP_POLY4_Y1_LL, R/W, Address = 0xF910_00FC)

VP_POLY4_Y1_LL	Bit	Description	Initial State
Reserved	[31]	Reserved, read as zero, do not modify	0
vp_poly4_y1_ph0	[30:24]	Poly-phase Filter Coefficients	0
Reserved	[23]	Reserved, read as zero, do not modify	0
vp_poly4_y1_ph1	[22:16]	Poly-phase Filter Coefficients	0
Reserved	[15]	Reserved, read as zero, do not modify	0
vp_poly4_y1_ph2	[14:8]	Poly-phase Filter Coefficients	0
Reserved	[7]	Reserved, read as zero, do not modify	0
vp_poly4_y1_ph3	[6:0]	Poly-phase Filter Coefficients	0

8.4.1.44 Video Processor Luminance 4-tap Poly-phase Filter Coefficients (VP_POLY4_Y1_LH, R/W, Address = 0xF910_0100)

VP_POLY4_Y1_LH	Bit	Description	Initial State
Reserved	[31]	Reserved, read as zero, do not modify	0
vp_poly4_y1_ph4	[30:24]	Poly-phase Filter Coefficients	0
Reserved	[23]	Reserved, read as zero, do not modify	0
vp_poly4_y1_ph5	[22:16]	Poly-phase Filter Coefficients	0
Reserved	[15]	Reserved, read as zero, do not modify	0
vp_poly4_y1_ph6	[14:8]	Poly-phase Filter Coefficients	0
Reserved	[7]	Reserved, read as zero, do not modify	0
vp_poly4_y1_ph7	[6:0]	Poly-phase Filter Coefficients	0

8.4.1.45 Video Processor Luminance 4-tap Poly-phase Filter Coefficients (VP_POLY4_Y1_HL, R/W, Address = 0xF910_0104)

VP_POLY4_Y1_HL	Bit	Description	Initial State
Reserved	[31]	Reserved, read as zero, do not modify	0
vp_poly4_y1_ph8	[30:24]	Poly-phase Filter Coefficients	0
Reserved	[23]	Reserved, read as zero, do not modify	0
vp_poly4_y1_ph9	[22:16]	Poly-phase Filter Coefficients	0
Reserved	[15]	Reserved, read as zero, do not modify	0
vp_poly4_y1_ph10	[14:8]	Poly-phase Filter Coefficients	0
Reserved	[7]	Reserved, read as zero, do not modify	0
vp_poly4_y1_ph11	[6:0]	Poly-phase Filter Coefficients	0

8.4.1.46 Video Processor Luminance 4-tap Poly-phase Filter Coefficients (VP_POLY4_Y1_HH, R/W, Address = 0xF910_0108)

VP_POLY4_Y1_HH	Bit	Description	Initial State
Reserved	[31]	Reserved, read as zero, do not modify	0
vp_poly4_y1_ph12	[30:24]	Poly-phase Filter Coefficients	0
Reserved	[23]	Reserved, read as zero, do not modify	0
vp_poly4_y1_ph13	[22:16]	Poly-phase Filter Coefficients	0
Reserved	[15]	Reserved, read as zero, do not modify	0
vp_poly4_y1_ph14	[14:8]	Poly-phase Filter Coefficients	0
Reserved	[7]	Reserved, read as zero, do not modify	0
vp_poly4_y1_ph15	[6:0]	Poly-phase Filter Coefficients	0

8.4.1.47 Video Processor Luminance 4-tap Poly-phase Filter Coefficients (VP_POLY4_Y2_LL, R/W, Address = 0xF910_010C)

VP_POLY4_Y2_LL	Bit	Description	Initial State
Reserved	[31]	Reserved, read as zero, do not modify	0
vp_poly4_y2_ph0	[30:24]	Poly-phase Filter Coefficients	0
Reserved	[23]	Reserved, read as zero, do not modify	0
vp_poly4_y2_ph1	[22:16]	Poly-phase Filter Coefficients	0
Reserved	[15]	Reserved, read as zero, do not modify	0
vp_poly4_y2_ph2	[14:8]	Poly-phase Filter Coefficients	0
Reserved	[7]	Reserved, read as zero, do not modify	0
vp_poly4_y2_ph3	[6:0]	Poly-phase Filter Coefficients	0

8.4.1.48 Video Processor Luminance 4-tap Poly-phase Filter Coefficients (VP_POLY4_Y2_LH, R/W, Address = 0xF910_0110)

VP_POLY4_Y2_LH	Bit	Description	Initial State
Reserved	[31]	Reserved, read as zero, do not modify	0
vp_poly4_y2_ph4	[30:24]	Poly-phase Filter Coefficients	0
Reserved	[23]	Reserved, read as zero, do not modify	0
vp_poly4_y2_ph5	[22:16]	Poly-phase Filter Coefficients	0
Reserved	[15]	Reserved, read as zero, do not modify	0
vp_poly4_y2_ph6	[14:8]	Poly-phase Filter Coefficients	0
Reserved	[7]	Reserved, read as zero, do not modify	0
vp_poly4_y2_ph7	[6:0]	Poly-phase Filter Coefficients	0

8.4.1.49 Video Processor Luminance 4-tap Poly-phase Filter Coefficients (VP_POLY4_Y2_HL, R/W, Address = 0xF910_0114)

VP_POLY4_Y2_HL	Bit	Description	Initial State
Reserved	[31]	Reserved, read as zero, do not modify	0
vp_poly4_y2_ph8	[30:24]	Poly-phase Filter Coefficients	0
Reserved	[23]	Reserved, read as zero, do not modify	0
vp_poly4_y2_ph9	[22:16]	Poly-phase Filter Coefficients	0
Reserved	[15]	Reserved, read as zero, do not modify	0
vp_poly4_y2_ph10	[14:8]	Poly-phase Filter Coefficients	0
Reserved	[7]	Reserved, read as zero, do not modify	0
vp_poly4_y2_ph11	[6:0]	Poly-phase Filter Coefficients	0

8.4.1.50 Video Processor Luminance 4-tap Poly-phase Filter Coefficients (VP_POLY4_Y2_HH, R/W, Address = 0xF910_0118)

VP_POLY4_Y2_HH	Bit	Description	Initial State
Reserved	[31]	Reserved, read as zero, do not modify	0
vp_poly4_y2_ph12	[30:24]	Poly-phase Filter Coefficients	0
Reserved	[23]	Reserved, read as zero, do not modify	0
vp_poly4_y2_ph13	[22:16]	Poly-phase Filter Coefficients	0
Reserved	[15]	Reserved, read as zero, do not modify	0
vp_poly4_y2_ph14	[14:8]	Poly-phase Filter Coefficients	0
Reserved	[7]	Reserved, read as zero, do not modify	0
vp_poly4_y2_ph15	[6:0]	Poly-phase Filter Coefficients	0

8.4.1.51 Video Processor Luminance 4-tap Poly-phase Filter Coefficients (VP_POLY4_Y3_LL, R/W, Address = 0xF910_011C)

VP_POLY4_Y3_LL	Bit	Description	Initial State
Reserved	[31:30]	Reserved, read as zero, do not modify	0
vp_poly4_y3_ph0	[29:24]	Poly-phase Filter Coefficients	0
Reserved	[23:22]	Reserved, read as zero, do not modify	0
vp_poly4_y3_ph1	[21:16]	Poly-phase Filter Coefficients	0
Reserved	[15:14]	Reserved, read as zero, do not modify	0
vp_poly4_y3_ph2	[13:8]	Poly-phase Filter Coefficients	0
Reserved	[7:6]	Reserved, read as zero, do not modify	0
vp_poly4_y3_ph3	[5:0]	Poly-phase Filter Coefficients	0

8.4.1.52 Video Processor Luminance 4-tap Poly-phase Filter Coefficients (VP_POLY4_Y3_LH, R/W, Address = 0xF910_0120)

VP_POLY4_Y3_LH	Bit	Description	Initial State
Reserved	[31:30]	Reserved, read as zero, do not modify	0
vp_poly4_y3_ph4	[29:24]	Poly-phase Filter Coefficients	0
Reserved	[23:22]	Reserved, read as zero, do not modify	0
vp_poly4_y3_ph5	[21:16]	Poly-phase Filter Coefficients	0
Reserved	[15:14]	Reserved, read as zero, do not modify	0
vp_poly4_y3_ph6	[13:8]	Poly-phase Filter Coefficients	0
Reserved	[7:6]	Reserved, read as zero, do not modify	0
vp_poly4_y3_ph7	[5:0]	Poly-phase Filter Coefficients	0

8.4.1.53 Video Processor Luminance 4-tap Poly-phase Filter Coefficients (VP_POLY4_Y3_HL, R/W, Address = 0xF910_0124)

VP_POLY4_Y3_HL	Bit	Description	Initial State
Reserved	[31:30]	Reserved, read as zero, do not modify	0
vp_poly4_y3_ph8	[29:24]	Poly-phase Filter Coefficients	0
Reserved	[23:22]	Reserved, read as zero, do not modify	0
vp_poly4_y3_ph9	[21:16]	Poly-phase Filter Coefficients	0
Reserved	[15:14]	Reserved, read as zero, do not modify	0
vp_poly4_y3_ph10	[13:8]	Poly-phase Filter Coefficients	0
Reserved	[7:6]	Reserved, read as zero, do not modify	0
vp_poly4_y3_ph11	[5:0]	Poly-phase Filter Coefficients	0

8.4.1.54 Video Processor Luminance 4-tap Poly-phase Filter Coefficients (VP_POLY4_Y3_HH, R/W, Address = 0xF910_0128)

VP_POLY4_Y3_LL	Bit	Description	Initial State
Reserved	[31:30]	Reserved, read as zero, do not modify	0
vp_poly4_y3_ph12	[29:24]	Poly-phase Filter Coefficients	0
Reserved	[23:22]	Reserved, read as zero, do not modify	0
vp_poly4_y3_ph13	[21:16]	Poly-phase Filter Coefficients	0
Reserved	[15:14]	Reserved, read as zero, do not modify	0
vp_poly4_y3_ph14	[13:8]	Poly-phase Filter Coefficients	0
Reserved	[7:6]	Reserved, read as zero, do not modify	0
vp_poly4_y3_ph15	[5:0]	Poly-phase Filter Coefficients	0

8.4.1.55 Video Processor Chrominance 4-tap Poly-phase Filter Coefficients (VP_POLY4_C0_LL, R/W, Address = 0xF910_012C)

Unlike VP_POLY4_Y registers, there are only a half of the coefficient registers for horizontal Chroma Scaler. The coefficients are assumed to be symmetric so that only half of them are kept. Some parts of them are unsigned integer and the other parts are signed integer. You must be careful while setting them.

VP_POLY4_C0_LL	Bit	Description	Initial State
Reserved	[31]	Reserved, read as zero, do not modify	0
vp_poly4_c0_ph0	[30:24]	Signed 7-bit integer (-64~63)	0
Reserved	[23]	Reserved, read as zero, do not modify	0
vp_poly4_c0_ph1	[22:16]	Signed 7-bit integer (-64~63)	0
Reserved	[15]	Reserved, read as zero, do not modify	0
vp_poly4_c0_ph2	[14:8]	Signed 7-bit integer (-64~63)	0
Reserved	[7]	Reserved, read as zero, do not modify	0
vp_poly4_c0_ph3	[6:0]	Signed 7-bit integer (-64~63)	0

8.4.1.56 Video Processor Chrominance 4-tap Poly-phase Filter Coefficients (VP_POLY4_C0_LH, R/W, Address = 0xF910_0130)

VP_POLY4_C0_LH	Bit	Description	Initial State
Reserved	[31:30]	Reserved, read as zero, do not modify	0
vp_poly4_c0_ph4	[29:24]	Signed 6-bit integer (-32~31)	0
Reserved	[23:22]	Reserved, read as zero, do not modify	0
vp_poly4_c0_ph5	[21:16]	Signed 6-bit integer (-32~31)	0
Reserved	[15:14]	Reserved, read as zero, do not modify	0
vp_poly4_c0_ph6	[13:8]	Signed 6-bit integer (-32~31)	0
Reserved	[7:6]	Reserved, read as zero, do not modify	0
vp_poly4_c0_ph7	[5:0]	Signed 6-bit integer (-32~31)	0

8.4.1.57 Video Processor Chrominance 4-tap Poly-phase Filter Coefficients (VP_POLY4_C0_HL, R/W, Address = 0xF910_0134)

VP_POLY4_C0_HL	Bit	Description	Initial State
Reserved	[31:30]	Reserved, read as zero, do not modify	0
vp_poly4_c0_ph8	[29:24]	Signed 6-bit integer (-32~31)	0
Reserved	[23:22]	Reserved, read as zero, do not modify	0
vp_poly4_c0_ph9	[21:16]	Signed 6-bit integer (-32~31)	0
Reserved	[15:14]	Reserved, read as zero, do not modify	0
vp_poly4_c0_ph10	[13:8]	Signed 6-bit integer (-32~31)	0
Reserved	[7:6]	Reserved, read as zero, do not modify	0
vp_poly4_c0_ph11	[5:0]	Signed 6-bit integer (-32~31)	0

8.4.1.58 Video Processor Chrominance 4-tap Poly-phase Filter Coefficients (VP_POLY4_C0_HH, R/W, Address = 0xF910_0138)

VP_POLY4_C0_HH	Bit	Description	Initial State
Reserved	[31:29]	Reserved, read as zero, do not modify	0
vp_poly4_c0_ph12	[28:24]	Signed 5-bit integer (-16~15)	0
Reserved	[23:21]	Reserved, read as zero, do not modify	0
vp_poly4_c0_ph13	[20:16]	Signed 5-bit integer (-16~15)	0
Reserved	[15:13]	Reserved, read as zero, do not modify	0
vp_poly4_c0_ph14	[12:8]	Signed 5-bit integer (-16~15)	0
Reserved	[7:5]	Reserved, read as zero, do not modify	0
vp_poly4_c0_ph15	[4:0]	Signed 5-bit integer (-16~15)	0

8.4.1.59 Video Processor Chrominance 4-tap Poly-phase Filter Coefficients (VP_POLY4_C1_LL, R/W, Address = 0xF910_013C)

VP_POLY4_C1_LL	Bit	Description	Initial State
vp_poly4_c1_ph0	[31:24]	Unsigned 8-bit integer (0~255)	0
vp_poly4_c1_ph1	[23:16]	Unsigned 8-bit integer (0~255)	0
vp_poly4_c1_ph2	[15:8]	Unsigned 8-bit integer (0~255)	0
vp_poly4_c1_ph3	[7:0]	Unsigned 8-bit integer (0~255)	0

8.4.1.60 Video Processor Chrominance 4-tap Poly-phase Filter Coefficients (VP_POLY4_C1_LH, R/W, Address = 0xF910_0140)

VP_POLY4_C1_LH	Bit	Description	Initial State
vp_poly4_c1_ph4	[31:24]	Unsigned 8-bit integer (0~255)	0
vp_poly4_c1_ph5	[23:16]	Unsigned 8-bit integer (0~255)	0
vp_poly4_c1_ph6	[15:8]	Unsigned 8-bit integer (0~255)	0
vp_poly4_c1_ph7	[7:0]	Unsigned 8-bit integer (0~255)	0

8.4.1.61 Video Processor Chrominance 4-tap Poly-phase Filter Coefficients (VP_POLY4_C1_HL, R/W, Address = 0xF910_0144)

VP_POLY4_C1_HL	Bit	Description	Initial State
vp_poly4_c1_ph8	[31:24]	Unsigned 8-bit integer (0~255)	0
vp_poly4_c1_ph9	[23:16]	Signed 8-bit integer (-128~127)	0
vp_poly4_c1_ph10	[15:8]	Signed 8-bit integer (-128~127)	0
vp_poly4_c1_ph11	[7:0]	Signed 8-bit integer (-128~127)	0

8.4.1.62 Video Processor Chrominance 4-tap Poly-phase Filter Coefficients (VP_POLY4_C1_HH, R/W, Address = 0xF910_0148)

VP_POLY4_C1_HH	Bit	Description	Initial State
Reserved	[31]	Reserved, read as zero, do not modify	0
vp_poly4_c1_ph12	[30:24]	Signed 7-bit integer (-64~63)	0
Reserved	[23]	Reserved, read as zero, do not modify	0
vp_poly4_c1_ph13	[22:16]	Signed 7-bit integer (-64~63)	0
Reserved	[15]	Reserved, read as zero, do not modify	0
vp_poly4_c1_ph14	[14:8]	Signed 7-bit integer (-64~63)	0
Reserved	[7]	Reserved, read as zero, do not modify	0
vp_poly4_c1_ph15	[6:0]	Signed 7-bit integer (-64~63)	0

8.4.1.63 Video Processor Post-processing Color Space Conversion Coefficient Register (PP_CSC_Y2Y_COEF, R/W, Address = 0xF910_01D4)

- BT.601 to BT.709 Color Space Conversion Matrix
 - $Y709 = 1.0 * Y601 - 0.118188 * Cb601 - 0.212685 * Cr601$
 - $Cb709 = 0.0 * Y601 + 1.018640 * Cb601 - 0.114618 * Cr601$
 - $Cr709 = 0.0 * Y601 + 0.075049 * Cb601 + 1.025327 * Cr601$
- BT.709 to BT.601 Color Space Conversion Matrix
 - $Y601 = 1.0 * Y709 + 0.101579 * Cb709 + 0.196076 * Cr709$
 - $Cb601 = 0.0 * Y709 + 0.989854 * Cb709 - 0.110653 * Cr709$
 - $Cr601 = 0.0 * Y709 - 0.072453 * Cb709 + 0.983398 * Cr709$
- Above all equations are written without interface offsets of +16 for Luminance and +128 for Chrominance.
- CSC module calculates above all equations without +128 offset for Chrominance, and generates final CSC results with +128 offset.
- In case of two Luminance equations, +16 offset is selectable by control register (PP_CSC_EN [1]). If Y offset (+16) exists in matrix input data, the coefficient of above equations should be redefined.

PP_CSC_Y2Y_COEF	Bit	Description	Initial State
Reserved	[31:12]	Reserved, read as zero, do not modify	0
PP_CSC_Y2Y_COEF	[11:0]	BT.601 to BT.709 (or BT.709 to BT.601) CSC coefficient for Y to Y [11]: Sign bit [10]: Integer bit [9:0]: Fraction bit 0x7FF: 1.999 ... 0x400: 1.0 ... 0x0: 0 0xFFFF: - 0.0001 ... 0xC00: - 1.0 ... 0x800: - 2.0	0

**8.4.1.64 Video Processor Post-processing Color Space Conversion Coefficient Register
(PP_CSC_CB2Y_COEF, R/W, Address = 0xF910_01D8)**

PP_CSC_CB2Y_COEF	Bit	Description	Initial State
Reserved	[31:12]	Reserved, read as zero, do not modify	0
PP_CSC_CB2Y_COEF	[11:0]	BT.601 to BT.709 (or BT.709 to BT.601) CSC coefficient for CB to Y	0

**8.4.1.65 Video Processor Post-processing Color Space Conversion Coefficient Register
(PP_CSC_CR2Y_COEF, R/W, Address = 0xF910_01DC)**

PP_CSC_CR2Y_COEF	Bit	Description	Initial State
Reserved	[31:12]	Reserved, read as zero, do not modify	0
PP_CSC_CR2Y_COEF	[11:0]	BT.601 to BT.709 (or BT.709 to BT.601) CSC coefficient for CR to Y	0

**8.4.1.66 Video Processor Post-processing Color Space Conversion Coefficient Register
(PP_CSC_Y2CB_COEF, R/W, Address = 0xF910_01E0)**

PP_CSC_Y2CB_COEF	Bit	Description	Initial State
Reserved	[31:12]	Reserved, read as zero, do not modify	0
PP_CSC_Y2CB_COEF	[11:0]	BT.601 to BT.709 (or BT.709 to BT.601) CSC coefficient for Y to CB	0

**8.4.1.67 Video Processor Post-processing Color Space Conversion Coefficient Register
(PP_CSC_CB2CB_COEF, R/W, Address = 0xF910_01E4)**

PP_CSC_CB2CB_COEF	Bit	Description	Initial State
Reserved	[31:12]	Reserved, read as zero, do not modify	0
PP_CSC_CB2CB_COEF	[11:0]	BT.601 to BT.709 (or BT.709 to BT.601) CSC coefficient for CB to CB	0

**8.4.1.68 Video Processor Post-processing Color Space Conversion Coefficient Register
(PP_CSC_CR2CB_COEF, R/W, Address = 0xF910_01F0)**

PP_CSC_CR2CB_COEF	Bit	Description	Initial State
Reserved	[31:12]	Reserved, read as zero, do not modify	0
PP_CSC_CR2CB_COEF	[11:0]	BT.601 to BT.709 (or BT.709 to BT.601) CSC coefficient for CR to CB	0

**8.4.1.69 Video Processor Post-processing Color Space Conversion Coefficient Register
(PP_CSC_Y2CR_COEF, R/W, Address = 0xF910_01EC)**

PP_CSC_Y2CR_COEF	Bit	Description	Initial State
Reserved	[31:12]	Reserved, read as zero, do not modify	0
PP_CSC_Y2CR_COEF	[11:0]	BT.601 to BT.709 (or BT.709 to BT.601) CSC coefficient for Y to CR	0

**8.4.1.70 Video Processor Post-processing Color Space Conversion Coefficient Register
(PP_CSC_CB2CR_COEF, R/W, Address = 0xF910_01E8)**

PP_CSC_CB2CR_COEF	Bit	Description	Initial State
Reserved	[31:12]	Reserved, read as zero, do not modify	0
PP_CSC_CB2CR_COEF	[11:0]	BT.601 to BT.709 (or BT.709 to BT.601) CSC coefficient for CB to CR	0

**8.4.1.71 Video Processor Post-processing Color Space Conversion Coefficient Register
(PP_CSC_CR2CR_COEF, R/W, Address = 0xF910_01F4)**

PP_CSC_CR2CR_COEF	Bit	Description	Initial State
Reserved	[31:12]	Reserved, read as zero, do not modify	0
PP_CSC_CR2CR_COEF	[11:0]	BT.601 to BT.709 (or BT.709 to BT.601) CSC coefficient for CR to CR	0

8.4.1.72 Video Processor Post-processing Image Bypass Mode Control Register (PP_BYPASS, R/W, Address = 0xF910_0200)

PP_BYPASS	Bit	Description	Initial State
Reserved	[31:1]	Reserved, read as zero, do not modify	0
PP_BYPASS	[0]	Disables the post image processor This is only for SDTV. We don't recommend you to use functions for HDTV. (Post image processor executes color saturation control, sharpness enhancement, contrast and brightness control.) 0 = Enables 1 = Disables (default)	1

8.4.1.73 Video Processor Color Saturation Control Register (PP_SATURATION, R/W, Address = 0xF910_020C)

PP_SATURATION	Bit	Description	Initial State
Reserved	[31:8]	Reserved, read as zero, do not modify	0
PP_SATURATION	[7:0]	Color saturation factor (Unsigned 1.7 format) 0x00 = 0.0 ... 0x80 = 1.0 ... 0xFF = 1.992188, (128*2 – 1)/128	80

8.4.1.74 Video Processor Picture Sharpness Control Register (PP_SHARPNESS, R/W, Address = 0xF910_0210)

PP_SHARPNESS	Bit	Description	Initial State
Reserved	[31:2]	Reserved, read as zero, do not modify	0
PP_TH_HNOISE	[15:8]	Threshold value setting to decide minimum vertical edge value	0x5
Reserved	[7:2]	Reserved, read as zero, do not modify	0
PP_SHARPNESS	[1:0]	Control for the edge enhancement 0 = No effect 1 = Minimum edge enhancement 2 = Moderate edge enhancement 3 = Maximum edge enhancement	0

8.4.1.75 Video Processor Brightness & Contrast Control Register (PP_LINE_EQ0 ~ PP_LINE_EQ7)

- PP_LINE_EQ0, R/W, Address = 0xF910_0218
- PP_LINE_EQ1, R/W, Address = 0xF910_021C
- PP_LINE_EQ2, R/W, Address = 0xF910_0220
- PP_LINE_EQ3, R/W, Address = 0xF910_0224
- PP_LINE_EQ4, R/W, Address = 0xF910_0228
- PP_LINE_EQ5, R/W, Address = 0xF910_022C
- PP_LINE_EQ6, R/W, Address = 0xF910_0230
- PP_LINE_EQ7, R/W, Address = 0xF910_0234

PP_LINE_EQx	Bit	Description	Initial State
Reserved	[31:24]	Reserved, read as zero, do not modify	0
LINE_INTC	[23:8]	<p>Intercept, signed 9.7 format Note: (9.7) format means that - '1' is a signed bit. - '8' is a integer. - '7' is a fraction. Example) INTC = 3 Due to 7-bit fraction, it is had to do 7 time left shift operation. As a result, register value is $3 * 2^7 = 0x18000$</p>	0
LINE_SLOPE	[7:0]	<p>Slope, unsigned 1.7 format. (Due to 1-bit integer, LINE_SLOPE has range from 0 to 1.9921875.) Note: (1.7) format means that - '1' is a integer - '7' is a fraction. Example) LINE_SLOPE = 0.5 = $1 * 2^{-1}$ Because of 7-bit fraction, it is had to do 7 time left shift operation. As a result, register value is $1/2 * 2^7 = 0x40$</p>	0

NOTE: 8 equations are related with Figure 8-8 Input luminance value between 0 ~ 255 is divide by 8 steps. Each of them is matched with each of 8 equations. Thus, we can make the new curve of the contrast and luminance as using 8 equation's combination. Each equation is matched like the following :

- PP_LINE_EQ0 = LINE_SLOPE0 * Y + LINE_INTC0 ($0 \leq Y \leq 31$)
- PP_LINE_EQ1 = LINE_SLOPE1 * Y + LINE_INTC1 ($32 \leq Y \leq 63$)
- PP_LINE_EQ2 = LINE_SLOPE2 * Y + LINE_INTC2 ($64 \leq Y \leq 95$)
- PP_LINE_EQ3 = LINE_SLOPE3 * Y + LINE_INTC3 ($96 \leq Y \leq 127$)
- PP_LINE_EQ4 = LINE_SLOPE4 * Y + LINE_INTC4 ($128 \leq Y \leq 159$)
- PP_LINE_EQ5 = LINE_SLOPE5 * Y + LINE_INTC5 ($160 \leq Y \leq 191$)
- PP_LINE_EQ6 = LINE_SLOPE6 * Y + LINE_INTC6 ($192 \leq Y \leq 223$)
- PP_LINE_EQ7 = LINE_SLOPE7 * Y + LINE_INTC7 ($224 \leq Y \leq 255$)

8.4.1.76 Video Processor Brightness offset Control Register for Y (PP_BRIGHT_OFFSET, R/W, Address = 0xF910_0238)

PP_BRIGHT_OFFSET	Bit	Description	Initial State
Reserved	[31:9]	Reserved, read as zero, do not modify	0
PP_BRIGHT_OFFSET	[8:0]	Offset for Y brightness control (Signed 1.8 format) Bright enhanced Y = Org Y + BRIGHT_OFFSET 0xFF : +255 ... 0x1 : +1 0x0 : 0 0x1FF : -1 ... 0x100 : -256	0

NOTE: Figure 8-8 shows examples of how VP controls brightness and contrast of image sequence using PP_LINE_EQ0 ~ PP_LINE_EQ7 registers and PP_BRIGHT_OFFSET register. Input to output luminance mapping curve is approximated by 8 sub-lines described by PP_LINE_EQ0 ~ PP_LINE_EQ7. Consequently, brightness and contrast is controlled in very flexible way.

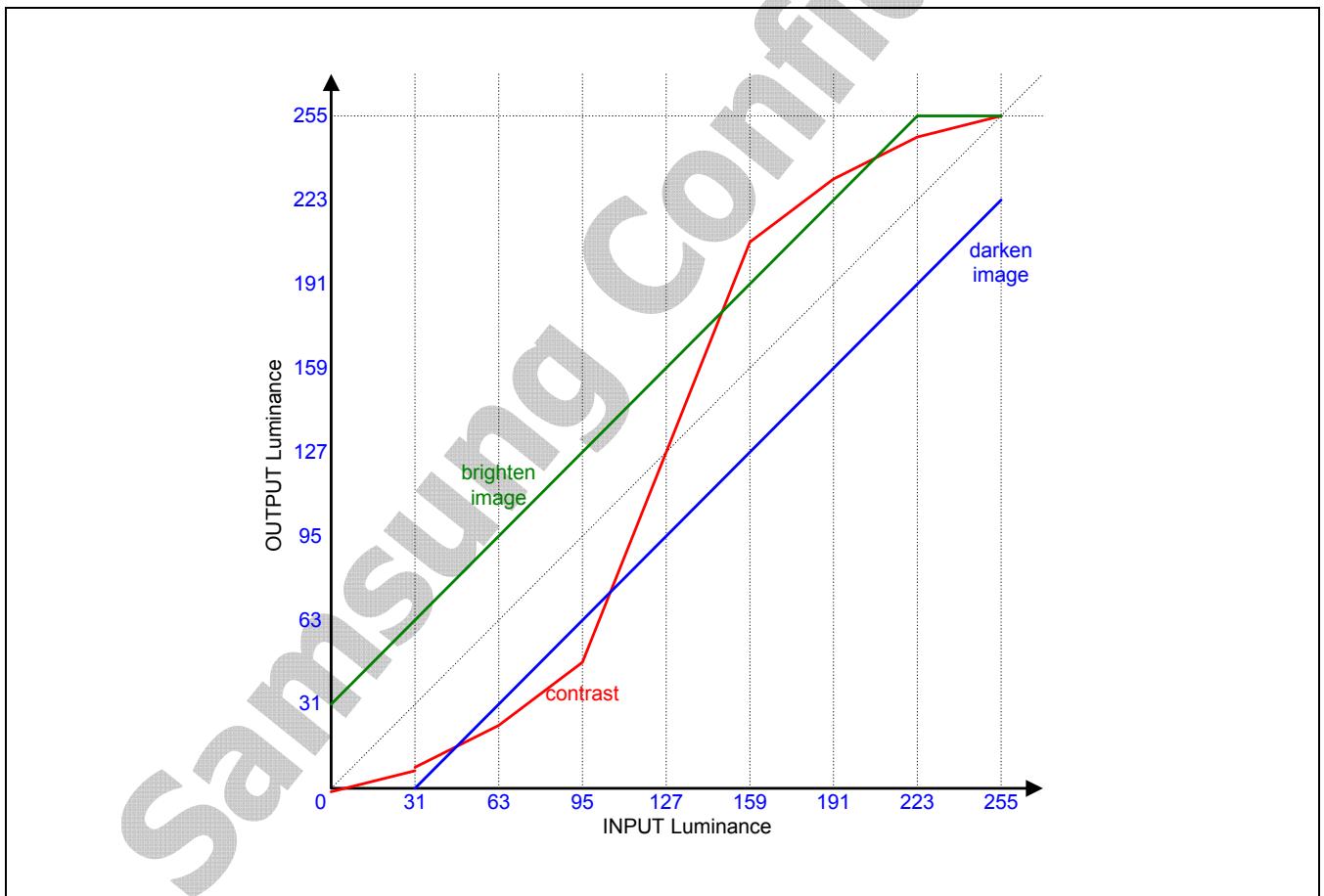


Figure 8-8 Image Brightness & Contrast Control

8.4.1.77 Video Processor Color Space Conversion Control Register (PP_CSC_EN, R/W, Address = 0xF910_023C)

PP_CSC_EN	Bit	Description	Initial State
Reserved	[31:2]	Reserved, read as zero, do not modify	0
SUB_Y_OFFSET_EN	[1]	Y offset control for color space conversion If (SUB_Y_OFFSET_EN == 1) $Y' = (Y-16)*Y2Y_coef + (Cb-128)*Cb2Y_coef + (Cr-128)*Cr2Y_coef$ $Cb' = (Y-16)*Y2Cb_coef + (Cb-128)*Cb2Cb_coef + (Cr-128)*Cr2Cb_coef$ $Cr' = (Y-16)*Y2Cr_coef + (Cb-128)*Cb2Cr_coef + (Cr-128)*Cr2Cr_coef$ Else $Y' = Y*Y2Y_coef + (Cb-128)*Cb2Y_coef + (Cr-128)*Cr2Y_coef$ $Cb' = Y*Y2Cb_coef + (Cb-128)*Cb2Cb_coef + (Cr-128)*Cr2Cb_coef$ $Cr' = Y*Y2Cr_coef + (Cb-128)*Cb2Cr_coef + (Cr-128)*Cr2Cr_coef$	1
CSC_EN	[0]	Color space conversion enable control 0 = Disable 1 = Enable	0

8.4.1.78 Video Processor Version Information Register (VP_VERSION_INFO, R, Address = 0xF910_03FC)

VP_VERSION_INFO	Bit	Description	Initial State
VERSION_INFO	[31:0]	VP version information	0x0000_0010

8.4.2 THE IDEA OF POLY-PHASE FILTERING IN VIDEO PROCESSOR

[Figure 8-9](#) shows basic concept of poly-phase filtering in video processor in case of 4-tap vertical luminance filter. Pixels highlighted in grey color are from decoded pictures and used to interpolate the dotted pixels, which are transferred to MIXER. The vertical positions of pixels to be interpolated are calculated with VP_SRC_V_POSITION and VP_V_RATIO. Once the vertical position is calculated, the nearest pixel phase (with 1/16 resolution) and which pixels are used for interpolation are decided.

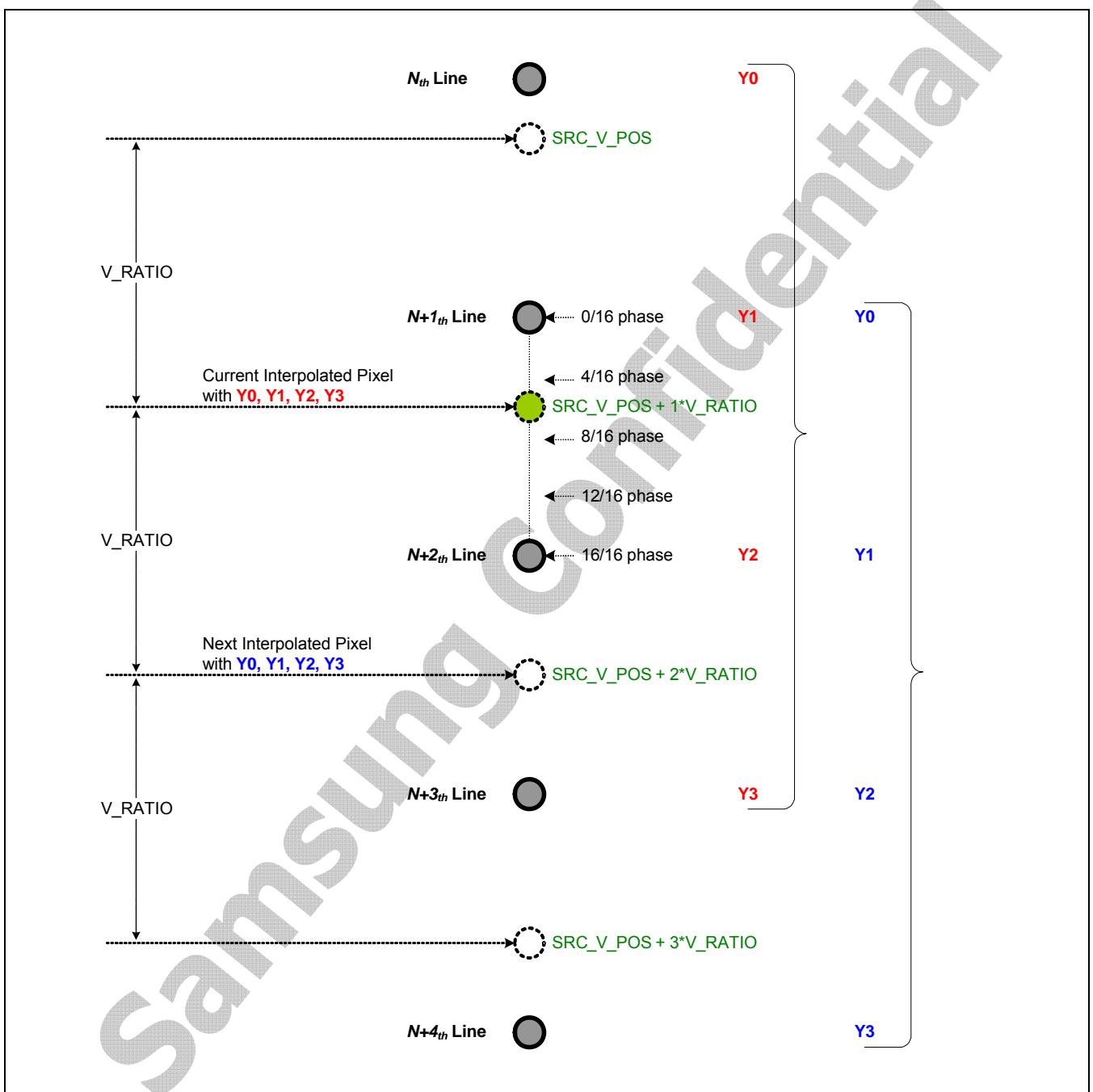


Figure 8-9 4-Tap Vertical Poly-phase Filter

If the calculated vertical position is 10.45, for example, pixels of 9th, 10th, 11th, and 12th lines are used for poly-phase filtering and the pixel phase is 7/16, which means the filter coefficients are vp_poly4_y0_ph7, vp_poly4_y1_ph7, vp_poly4_y2_ph7, and vp_poly4_y3_ph7.

8-tap luminance horizontal poly-phase filter and 4-tap chrominance horizontal poly-phase filter use the exact same scheme.

At the boundaries of pictures (top, bottom, left, and right), some pixels in filter window are not available. In this case, value of the nearest pixel is repeated as shown in [Figure 8-10](#).

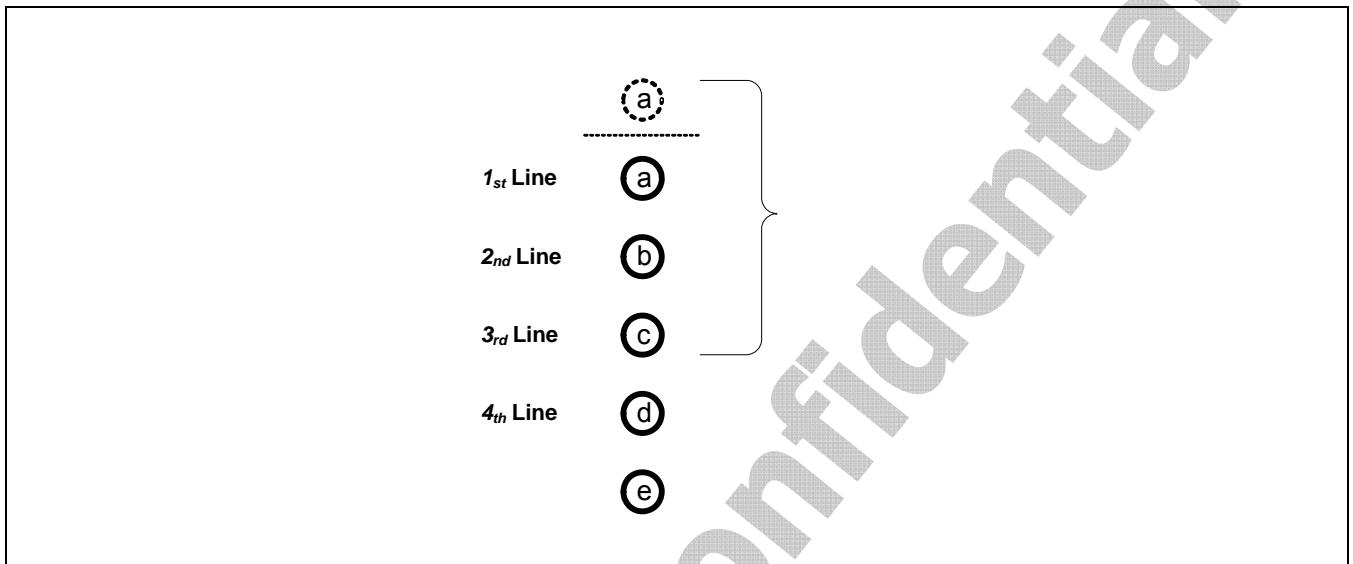


Figure 8-10 Pixel Repetition at Picture Boundary

9 MIXER

9.1 OVERVIEW OF MIXER

Mixer overlaps or blends the input data such as graphic, video, background and sends the result data to the TVOUT module. The TVOUT module generates all the video control signals.

Graphic data is transferred to the mixer from External DRAM memory via AXI interface. However, the video data is transferred to the mixer via direct connection with pre-defined protocol. The data generated from mixer is directly transferred to TVENC/HDMI module for the real time transfer.

9.1.1 KEY FEATURES OF MIXER

- Supports AXI Master & AHB Slave Interface
- AXI Master interface for graphic layer data fetch
- AHB Slave interface for control register setup
- Supports Little/ Big Endian for graphic layers data
- [Input]
 - Multiple Layers
 - Background layer
 - Graphic0 layer
 - Graphic1 layer
 - Video layer
 - Input Control features
 - Blending between each layer
 - Selectable graphic layer frame buffer
 - Enable/disable each layer
 - Source cropping for graphic layer
- [Output]
 - Overlapped and blended input layers
 - YCbCr 4:4:4 / RGB 8:8:8
 - Supports interlaced/ progressive scan
 - Supports 480i/p, 576i/p, 720p and 1080i/1080p display sizing (1080p is 30Hz.)
 - [Graphic0,1 Layer]
 - Source: External DRAM frame buffer memory

- Color Format : differently configurable between each graphic layer
 - 16bpp Direct RGB[565]
 - 16bpp Direct ARGB[1555]
 - 16bpp Direct ARGB[4444]
 - 32bpp Direct ARGB[8888]
- Maximum graphic layer size
 - 480i/p: 720x480 pixels
 - 576i/p: 720x576 pixels
 - 720p: 1280x720 pixels
 - 1080i/p: 1920x1080 pixel
- Blending
 - Maximum 256 level pixel and layer blending
 - Separately configurable layer blending factor between each layer
- Scale
 - Vertical line duplication : x2
 - Horizontal win-scale : x2
- [Video Layer]
- Source: Video processor module
- Color Format: 24bpp Direct YCbCr [888]
- Maximum Resolution
 - 480i/p: 720x480 pixels
 - 576i/p: 720x576 pixels
 - 720p: 1280x720 pixels
 - 1080i/p: 1920x1080 pixels
- The xvYcc limiter is supported.
- [Background layer]
- Source: Configuration register
- Lowest layer
- Layer ordering
- Background → (Video ↔ Graphic0 ↔ Graphic1)
- Video layer and 2 graphic layers are fully ordered and blended
- Blending
- Pixel blending and Layer blending
- Alpha blending
- Pre-multiplied blending

9.1.2 BLOCK DIAGRAM OF MIXER

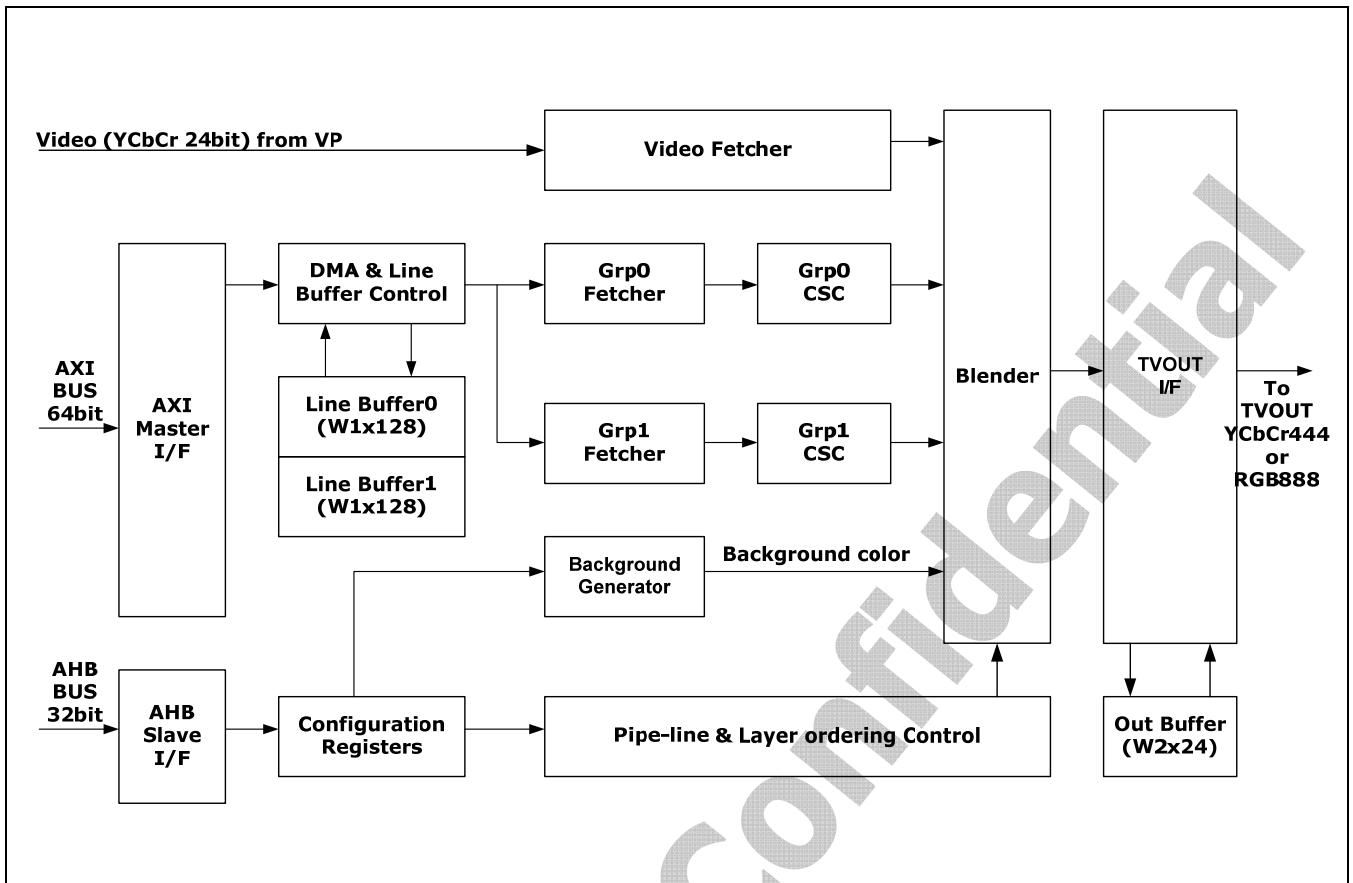


Figure 9-1 Block Diagram of Mixer

AXI Master I/F and DMA & Line Buffer Control: This block fetches data from the memory and stores data in Line Buffer0/1 block.

AHB Slave I/F and Configuration Registers: This block is SFRs to control the mixer.

Video Fetcher: This block fetches data from VP module.

Grp0/1 fetcher and Grp0/1 CSC: Grp0/1 fetchers pop up data from Line Buffer0/1 and delivers image data to Blender block through the color space converters (Grp0/1 CSC).

Background Generator: This block generates background patterns according to configurations.

Blender: The role of this block is to mix 4 image-layers such as Video, Graphic0/1 and Back-ground.

TVOUT I/F: This block temporally stores data from the Blender until either TVENC or HDMI requests data.

9.1.3 VIDEO CLOCK RELATION

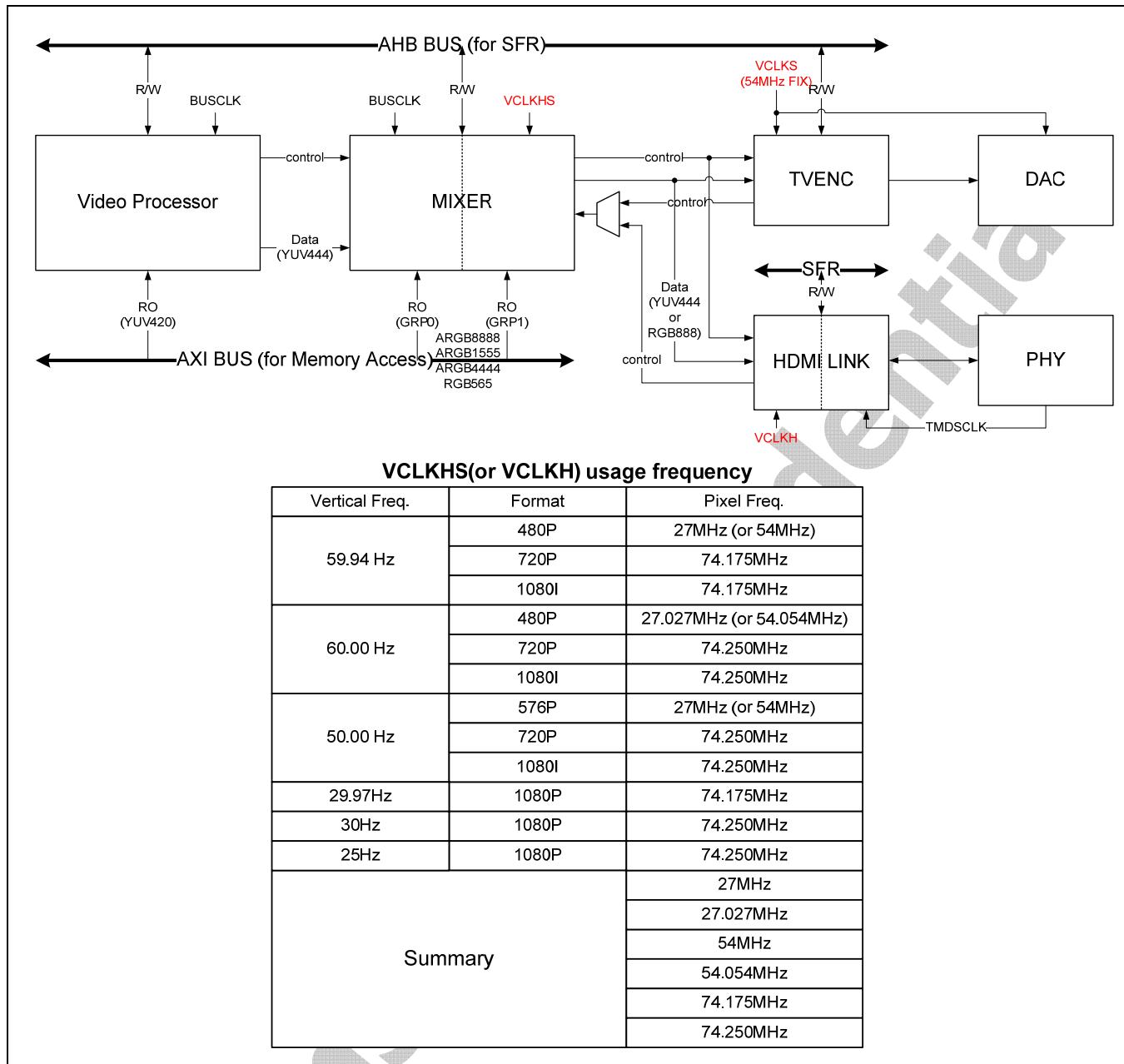


Figure 9-2 TV Sub-System Block Diagram and Usage Frequency

There are two paths from mixer to TVENC and HDMI. It is selected exclusively at the Clock Controller (refer to REG_DST_SEL at mixer_CFG register (0xF920_0004)). When TV-out is selected, Mixer I/F clock (VCLKHS) and VCLKS (TVENC clock) is fixed by 54MHz. Thus, you must set MIXER_SEL register in CLK_SRC1(0xE010_0204) for more information, refer to CMU chapter. Else, in HDMI-out selection, REG_DST_SEL register is configured properly. Then, you set the same clock configuration between MIXER I/F clock (VCLKHS) and VCLKH(HDMI pixel clock). The clock which is generated by embedded PLL in HDMI PHY must be selected using usage frequency table (refer to VCLKHS(or VCLKH). You can configure clock source through CLK_SRC1 register.

9.2 REGISTER DESCRIPTION

9.2.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
Mixer Global Setting				
MIXER_STATUS	0xF920_0000	R/W	Specifies the Status of MIXER Operation	0x0000_0006
MIXER_CFG	0xF920_0004	R/W	Specifies the MIXER Mode Setting	0x0000_0000
Mixer Interrupt				
MIXER_INT_EN	0xF920_0008	R/W	Specifies the Interrupt Enable	0x0000_0000
MIXER_INT_STATUS	0xF920_000C	R/W	Specifies the Interrupt Status	0x0000_0000
Video & Blender Configuration				
MIXER_LAYER_CFG	0xF920_0010	R/W	Specifies the Video & Graphic Layer Priority and On/ Off	0x0000_0000
MIXER_VIDEO_CFG	0xF920_0014	R/W	Specifies the Video Layer Configuration	0x0000_0000
MIXER_VIDEO_LIMITER_PA RA_CFG	0xF920_0018	R/W	Specifies the parameter of video layer limiter configuration	0xEB10_F010
Graphic0 Layer Configuration				
MIXER_GRAPHIC0_CFG	0xF920_0020	R/W	Specifies the Graphic Layer0 Configuration	0x0000_0000
MIXER_GRAPHIC0_BASE	0xF920_0024	R/W	Specifies the Base Address for Graphic Layer0	0x0000_0000
MIXER_GRAPHIC0_SPAN	0xF920_0028	R/W	Specifies the Span for Graphic Layer0	0x0000_0000
MIXER_GRAPHIC0_SXY	0xF920_002C	R/W	Specifies the Source X/Y Positions for Graphic Layer0	0x0000_0000
MIXER_GRAPHIC0_WH	0xF920_0030	R/W	Specifies the Width/ Height for Graphic Layer0	0x0000_0000
MIXER_GRAPHIC0_DXY	0xF920_0034	R/W	Specifies the Destination X/Y Positions for Graphic Layer0	0x0000_0000
MIXER_GRAPHIC0_BLANK	0xF920_0038	R/W	Specifies the Blank Pixel Value for Graphic Layer0	0x0000_0000
Graphic1 Layer Configuration				
MIXER_GRAPHIC1_CFG	0xF920_0040	R/W	Specifies the Graphic Layer1 Configuration	0x0000_0000
MIXER_GRAPHIC1_BASE	0xF920_0044	R/W	Specifies the Base Address for Graphic Layer1	0x0000_0000
MIXER_GRAPHIC1_SPAN	0xF920_0048	R/W	Specifies the Span for Graphic Layer1	0x0000_0000
MIXER_GRAPHIC1_SXY	0xF920_004C	R/W	Specifies the Source X/Y Positions for Graphic Layer1	0x0000_0000

Register	Address	R/W	Description	Reset Value
MIXER_GRAPHIC1_WH	0xF920_0050	R/W	Specifies the Width/ Height for Graphic Layer1	0x0000_0000
MIXER_GRAPHIC1_DXY	0xF920_0054	R/W	Specifies the Destination X/Y Positions for Graphic Layer1	0x0000_0000
MIXER_GRAPHIC1_BLANK	0xF920_0058	R/W	Specifies the Blank Pixel Value for Graphic Layer1	0x0000_0000
Background Layer Configuration				
MIXER_BG_COLOR0	0xF920_0064	R/W	Specifies the Background Color of First Point	0x0000_0000
MIXER_BG_COLOR1	0xF920_0068	R/W	Specifies the Background Color of Second Point	0x0000_0000
MIXER_BG_COLOR2	0xF920_006C	R/W	Specifies the Background Color of Last Point	0x0000_0000
Color Space Conversion Coefficient				
MIXER_CM_COEFF_Y	0xF920_0080	R/W	Specifies the Scaled Color Space Conversion (RGB to Y) Coefficient for Graphic Layer	0x0844_0832
MIXER_CM_COEFF_CB	0xF920_0084	R/W	Specifies the Scaled Color Space Conversion (RGB to CB) Coefficient for Graphic Layer	0x3b5d_b0e1
MIXER_CM_COEFF_CR	0xF920_0088	R/W	Specifies the Scaled Color Space Conversion (RGB to Cr) Coefficient for Graphic Layer	0x0e1d_13dc
Mixer Global Setting Shadowing Register				
MIXER_STATUS_S	0xF920_2000	R	Specifies the Status of MIXER Operation (Shadow)	0x0000_0006
MIXER_CFG_S	0xF920_2004	R	Specifies the MIXER Mode Setting (Shadow)	0x0000_0000
Video & Blender Configuration Shadowing Register				
MIXER_LAYER_CFG_S	0xF920_2010	R	Specifies the Video & Graphic Layer Priority and On/ Off (Shadow)	0x0000_0000
MIXER_VIDEO_CFG_S	0xF920_2014	R	Specifies the Video Layer Configuration (Shadow)	0x0000_0000
MIXER_VIDEO_LIMITER_PA RA_CFG_S	0xF920_2018	R	Specifies the The parameter of video layer limiter configuration	0xEB10_F010
Graphic0 Layer Configuration Shadowing Register				
MIXER_GRAPHIC0_CFG_S	0xF920_2020	R	Specifies the Graphic Layer0 Configuration (Shadow)	0x0000_0000
MIXER_GRAPHIC0_BASE_S	0xF920_2024	R	Specifies the Graphic0 Base Address (Shadow)	0x0000_0000
MIXER_GRAPHIC0_SPAN_S	0xF920_2028	R	Specifies the Graphic0 Span (Shadow)	0x0000_0000

Register	Address	R/W	Description	Reset Value
MIXER_GRAPHIC0_SXY_S	0xF920_202C	R	Specifies the Graphic0 Source X/Y Coordinates (Shadow)	0x0000_0000
MIXER_GRAPHIC0_WH_S	0xF920_2030	R	Specifies the Graphic0 Width/ Height (Shadow)	0x0000_0000
MIXER_GRAPHIC0_DXY_S	0xF920_2034	R	Specifies the Graphic0 Destination X/Y Coordinates (Shadow)	0x0000_0000
MIXER_GRAPHIC0_BLANK_PIXEL_S	0xF920_2038	R	Specifies the Graphic0 Blank Pixel (Shadow)	0x0000_0000
Graphic1 Layer Configuration Shadowing Register				
MIXER_GRAPHIC1_CFG_S	0xF920_2040	R	Specifies the Graphic Layer1 Configuration (Shadow)	0x0000_0000
MIXER_GRAPHIC1_BASE_S	0xF920_2044	R	Specifies the Graphic1 Base Address (Shadow)	0x0000_0000
MIXER_GRAPHIC1_SPAN_S	0xF920_2048	R	Specifies the Graphic1 Span (Shadow)	0x0000_0000
MIXER_GRAPHIC1_SXY_S	0xF920_204C	R	Specifies the Graphic1 Source X/Y Coordinates (Shadow)	0x0000_0000
MIXER_GRAPHIC1_WH_S	0xF920_2050	R	Specifies the Graphic1 Width/ Height (Shadow)	0x0000_0000
MIXER_GRAPHIC1_DXY_S	0xF920_2054	R	Specifies the Graphic1 Destination X/Y Coordinates (Shadow)	0x0000_0000
MIXER_GRAPHIC1_BLANK_PIXEL_S	0xF920_2058	R	Specifies the Graphic1 Blank Pixel (Shadow)	0x0000_0000
Background Layer Configuration Shadowing Register				
MIXER_BG_COLOR0_S	0xF020_2064	R	Specifies the Background First Color (Shadow)	0x0000_0000
MIXER_BG_COLOR1_S	0xF920_2068	R	Specifies the Background Second Color (Shadow)	0x0000_0000
MIXER_BG_COLOR2_S	0xF920_206C	R	Specifies the Background Last Color (Shadow)	0x0000_0000
Version Register				
MIXER_VER	0xF920_0100	R	Specifies the Mixer Version	0x0000_0010

9.2.2 SHADOW REGISTERS (READ ONLY)

If SYNC_ENABLE signal is set to 1, the written values to internal registers are not directly applied to the mixer operation. They are temporarily stored in the internal register and waits for next v_sync signal. After the v_sync signal occurs, the stored internal register values are updated to the shadow registers.

In interlaced display mode, the shadow registers are updated only at top-field. In progressive display mode, the shadow registers are updated at every v_sync.

9.2.2.1 Mixer_Status Register (MIXER_STATUS, R/W, Address = 0xF920_0000)

MIXER_STATUS	Bit	Description	Initial State
Reserved	[31:8]	Reserved, read as zero, do not modify	0
16_BURST_MODE	[7]	16 burst mode(64Bit Bus) enabled in DMA 1 = 16Beat Burst Mode 0 = 8Beat Burst Mode	0
Reserved	[6:4]	Reserved.	0
BIG_ENDIAN	[3]	0 = Little Endian Source Format 1 = Big Endian Source Format	0
SYNC_ENABLE	[2]	0 = Values set by user will not be applied to the mixer operation although v_sync is detected. 1 = Values set by user can be applied to the mixer operation after v_sync detected.	1
MIXER_OPERATION_STATUS	[1]	This bit is read-only. 0 = MIXER is operating. 1 = MIXER is idle mode. Note: If you want to stop operation, make REG_RUN "0" and check this bit whether it is "1"	1
REG_RUN	[0]	The mixer operation control. This register is also updated after V_SYNC. 0 = Mixer stops. 1 = Mixer starts processing. Note: The SFRs of Video Processor and Image Mixer is updated by Vertical Sync of TVENC's Timing Generator. Thus, SFRs are configured before this bit is enabled. The sequence to enable TVSS is as follows: "VP -> MIXER -> TVENC(HDMI)". Also, because SFRs are updated by Vertical Sync, the disabling sequence is following as: "VP -> MIXER -> TVNEC(HDMI)".	0

9.2.2.2 MIXER_CFG REGISTER (MIXER_CFG, R/W, Address = 0xF920_0004)

MIXER_CFG	Bit	Description	Initial State
Reserved	[31:11]	Reserved, read as zero, do not modify	0
REG_RGB_FORMAT	[10:9]	The RGB's range selection 0 = RGB601 , 0 ~ 255 1 = RGB601 , 16 ~ 235 2 = RGB709 , 0 ~ 255 3 = RGB709 , 16 ~ 235	0
REG_OUT_TYPE	[8]	The mixer's output type selection 0 = YCbCr444 1 = RGB888	0
REG_DST_SEL	[7]	The display IP selection 0 = TV Out 1 = HDMI Out	0
REG_HD_MODE	[6]	720p or 1080i selection 0 = 720p 1 = 1080i/1080p Note: 1080i = REG_SCAN_MODE is '0' 1080p = REG_SCAN_MODE is '1'	0
REG_GRAPHIC1_EN	[5]	Graphic1 layer display control bit. 0 = Disable 1 = Enable	0
REG_GRAPHIC0_EN	[4]	Graphic0 layer display control bit. 0 = Disable 1 = Enable	0
REG_VIDEO_EN	[3]	Video layer display control bit. 0 = Disable 1 = Enable	0
REG_SCAN_MODE	[2]	Display scanning mode of TV. 0 = Interlaced mode 1 = Progressive mode	0
REG_NTSC_PAL	[1]	Display standard of TV. If you set this bit '0' and set REG_SCAN_MODE '1', output has to be call 480p standard. This is only valid when REG_HD_SD is "0". 0 = NTSC (720x480) 1 = PAL (720x576)	0
REG_HD_SD	[0]	HD or SD selection 0 = SD 1 = HD If REG_HD_SD is 1, REG_HD_MODE = 0 for 720p REG_HD_MODE = 1 for 1080i.	0

NOTE: All changes to this register are valid on a vertical sync signal of next frame.

	Wide	Narrow
CSCY2R (601)	$R = Y + 1.371(Cr-128)$ $G = Y - 0.698(Cr-128) - 0.336(Cb-128)$ $B = Y + 1.732(Cb-128)$	$R = 1.164(Y-16) + 1.596(Cr-128)$ $G = 1.164(Y-16) - 0.813(Cr-128) - 0.391(Cb-128)$ $B = 1.164(Y-16) + 2.018(Cb-128)$
CSCY2R (709)	$R = Y + 1.540(Cr-128)$ $G = Y - 0.459(Cr-128) - 0.183(Cb-128)$ $B = Y + 1.816(Cb-128)$	$R = 1.164(Y-16) + 1.793(Cr-128)$ $G = 1.164(Y-16) - 0.534(Cr-128) - 0.213(Cb-128)$ $B = 1.164(Y-16) + 2.115(Cb-128)$

NOTE: This table refers to Video Demystified(Keith Jack).

9.2.2.3 MIXER_INT_EN Register (MIXER_INT_EN, R/W, Address = 0xF920_0008)

MIXER_INTR	Bit	Description	Initial State
Reserved	[31:12]	Reserved, read as zero, do not modify	0
INT_EN_VSYNC	[11]	The vertical sync. interrupt enable. (Write only) 0 = Disable interrupt 1 = Enable interrupt	0
INT_EN_VP	[10]	The VP underflow interrupt enable. 0 = Disables interrupt 1 = Enables interrupt Setting this bit to '0' disables only the interrupt request to host controller. It does not mask the change of the MIXER_INT_STATUS[10] bit status	0
INT_EN_GRP1	[9]	The graphic layer1 line buffer underflow interrupt enable. 0 = Disables interrupt 1 = Enables interrupt Setting this bit to '0' disables only the interrupt request to host controller. It does not mask the change of the MIXER_INT_STATUS[9] bit status	0
INT_EN_GRP0	[8]	The graphic layer0 line buffer underflow interrupt enable. 0 = Disables interrupt 1 = Enables interrupt Setting this bit to '0' disables only the interrupt request to host controller. It does not mask the change of the MIXER_INT_STATUS[8] bit status	0
Reserved	[7:0]	Reserved.	0

9.2.2.4 MIXER_INT_STATUS Register (MIXER_INT_STATUS, R/W, Address = 0xF920_000C)

MIXER_INTR	Bit	Description	Initial State
Reserved	[31:11]	Reserved, read as zero, do not modify	0
INT_CLEAR_VSYNC	[11]	The vertical sync. inerrupt clear bit. (Write only) 1 = Interrupt is cleared. Write '1' to this bit clears the interrupt. Also, Write '1' to this bit before you set INT_EN_VSYNC.	0
INT_STATUS_VP	[10]	The VP underflow interrupts status. 0 = Interrupt is not fired 1 = Interrupt is fired Writing '1' to this bit clears the interrupt. This interrupt is automatically asserted by line buffer controller if underflow is generated in line buffer.	0
INT_STATUS_GRP1	[9]	The graphic layer1 line buffer underflow interrupt status. 0 = Interrupt is not fired 1 = Interrupt is fired Writing '1' to this bit clears the interrupt. This interrupt is automatically asserted by line buffer controller if underflow is generated in line buffer.	0
INT_STATUS_GRP0	[8]	The graphic layer0 line buffer underflow interrupt status. 0 = Interrupt is not fired 1 = Interrupt is fired Writing '1' to this bit clears the interrupt. This interrupt is automatically asserted by line buffer controller if underflow is generated in line buffer.	0
Reserved	[7:1]	Reserved, read as zero, do not modify	0
INT_STATUS_VSYNC	[0]	The vertical sync. status. (Read only) 0 = Interrupt is not fired. 1 = Interrupt is fired. Note : If INT_STATUS_VSYNC & !INT_STATUS_VP & !INT_STATUS_GRP1 & !INT_STATUS_GRP0 is high, The vertical sync. is fired.	

9.2.2.5 MIXER_LAYER_CFG Register (MIXER_LAYER_CFG, R/W, Address = 0xF920_0010)

The priority value for video and each graphic layer can be set. The priority field is used to determine the priority of a graphic layer. The graphic layer of higher value has the higher priority. This field is also used as on/off switch. If one field is set to zero, the corresponding graphic layer is not displayed. If some layers have the same value, the priority is like this: graphic layer 1 > graphic layer 0 > video. The priority is only determined by difference of priority's value. For example, case1 and case 2 have the same effect.

- Case1: GRP1 priority is 2 , GRP0 priority 3 , Video Priority 1
- Case2: GRP1 priority is 14 , GRP0 priority 15 , Video Priority 13

MIXER_LAYER_CFG	Bit	Description	Initial State
Reserved	[31:12]	Reserved, read as zero, do not modify	0
Graphic layer 1 priority	[11:8]	15 ~ 1 = the priority value 0 = Hides the graphic layer 1	0
Graphic layer 0 priority	[7:4]	15 ~ 1 = the priority value 0 = Hides the graphic layer 0	0
Video layer priority	[3:0]	15 ~ 1 = the priority value 0 = Hides the video layer 0	0

NOTE: All the changes of this register are valid on a vertical sync signal of next frame when SYNC_ENABLE flag is set to one. The "Hide" means that layer data is ready but is not displayed.

9.2.2.6 MIXER_VIDEO_CFG Register (MIXER_VIDEO_CFG, R/W, Address = 0xF920_0014)

MIXER_VIDEO_CFG	Bit	Description	Initial State
Reserved	[31:18]	Reserved, read as zero, do not modify	0
REG_LIMITER_EN	[17]	YUV limiter for xvYcc 0 = Disables 1 = Enables	0
REG_BLEND_EN	[16]	If set to 1, it enables the blending of the entire video layer onto the lower layer using the blending factor, REG_ALPHA_VID.	0
Reserved	[15:8]	Reserved, read as zero, do not modify	0
REG_ALPHA_VID	[7:0]	Video layer blending factor. This factor is used over all the pixels in the video layer to blend with lower layer. $\alpha * \text{video_layer_pixel_value} + (1 - \alpha) * \text{lower_layer_pixel_value}$ If REG_ALPHA_VID is 0, α is 0. If REG_ALPHA_VID is not 0, $\alpha = (\text{REG_ALPHA_VID} + 1)/256.$	0

NOTE: All changes to this register are valid on a vertical sync signal of next frame.

9.2.2.7 MIXER_VIDEO_LIMITER_PARA_CFG Register (MIXER_VIDEO_LIMITER_PARA_CFG, R/W, Address = 0xF920_0018)

MIXER_VIDEO_LIMITER_PARA_CFG	Bit	Description	Initial State
REG PARA Y UPPER	[31:24]	Upper bound for Y parameter of the limiter	0xEB
REG PARA Y LOWER	[23:16]	Lower bound for Y parameter of the limiter	0x10
REG PARA C UPPER	[15:8]	Upper bound for C parameter of the limiter	0xF0
REG PARA C LOWER	[7:0]	Lower bound for C parameter of the limiter	0x10

NOTE: All changes to this register are valid on a vertical sync signal of next frame.

9.2.2.8 MIXER_GRAPHIC0_CFG Register (MIXER_GRAPHIC0_CFG, R/W, Address = 0xF920_0020)

MIXER_GRAPHIC0_CFG	Bit	Description	Initial State
RTQoS_GRP0	[31:23]	Real time QoS configuration. The size of graphic layer0 FIFO is 480depth(The each of level is 128bit) 0 = RTQoS Disable 1 ~ 480 = QoS threshold level 481 ~ = Reserved	0
Reserved	[22]	Reserved	0
BLANK_CHANGE0	[21]	0 = Blank key (color key) is enable. 1 = Blank key (color key) is disable.	0
PRE_MUL_MODE0	[20]	Pre Multiplied_blending mode in graphic layer0. 1 = Pre-Multiplied mode 0 = Normal mode In this mode, graphic pixel data must be pre-multiplied with graphic pixel alpha. In pre-multiplied mode, REG_PIXEL0_BLEND_EN must be enabled. Graphic and lower layer blending factor. This factor is used all over the pixels in the graphic and lower layer to blend with. A blending factor(α) is as follows depending on the a layer blending factor and a pixel blending factor values: See Table 9-1 .	0
Reserved	[19:18]	Reserved, read as zero	
REG_WIN0_BLEND_EN	[17]	Blending by a blending factor set by REG_ALPHA_WIN0 register on all over the graphic layer0.	0
REG_PIXEL0_BLEND_EN	[16]	Blending by a blending factor of each pixel is enabled in graphic layer0.	0
Reserved	[15:12]	Reserved, read as zero	0

MIXER_GRAPHIC0_CFG	Bit	Description	Initial State
EG_COLOR_FORMAT0	[11:8]	<p>Graphic layer0 color format.</p> <p>0 = Reserved 1 = Reserved 2 = Reserved 3 = Reserved 4 = RGB 565 5 = ARGB 1555 6 = ARGB 4444 7 = ARGB 8888 8 = Reserved 9 = Reserved A = Reserved B = Reserved C = Reserved D = Reserved E = Reserved F = Reserved</p>	0
REG_ALPHA_WIN0	[7:0]	<p>Graphic layer0 and lower layer blending factor. This factor is used all over the pixels in the graphic and lower layer to blend with.</p> <p>A blending factor(α) is as follows depending on the a layer blending factor and a pixel blending factor values:</p> $\alpha * \text{graphic_layer_pixel_value} + (1 - \alpha) * \text{lower_layer_pixel_value}$ <p>See Table 9-1.</p> <p>If REG_ALPHA_WIN0 is 0, blending_factor_layer is 0. If REG_ALPHA_WIN0 is not 0, $\text{blending_factor_layer} = (\text{REG_ALPHA_WIN} + 1) / 256$.</p> <p>If A(blending factor of each pixel) is 0, $\text{blending_factor_each_pixel}$ is 0</p> <p>If A(blending factor of each pixel) is not 0, $\text{blending_factor_each_pixel} = (A + 1) / 256$.</p> <p>The pixel blending factors comes from the pixel in direct modes except the direct 16bpp 565 format.</p>	0

NOTE: All the changes of this register are valid on a vertical sync signal of next frame.

9.2.2.9 MIXER_GRAPHIC1_CFG Register (MIXER_GRAPHIC1_CFG, R/W, Address = 0xF920_0040)

MIXER_GRAPHIC1_CFG	Bit	Description	Initial State
RTQoS_GRP1	[31:23]	Real time QoS configuration. The size of graphic layer0 FIFO is 480depth(The each of level is 128bit) 0 = RTQoS Disable 1 ~ 480 = QoS threshold level 481 ~ = Reserved	0
Reserved	[22]	Reserved	0
BLANK_CHANGE1	[21]	0 = Blank key (color key) is enable. 1 = Blank key (color key) is disable.	0
PRE_MUL_MODE1	[20]	Pre Multiplied_blending mode in graphic layer1. 1 = Pre-Multiplied mode 0 = Normal mode In this mode, graphic pixel data must be pre-multiplied with graphic pixel alpha. In pre-multiplied mode, REG_PIXEL1_BLEND_EN must be enabled. Graphic and lower layer blending factor. This factor is used all over the pixels in the graphic and lower layer to blend with. A blending factor(α) is as follows depending on the a layer blending factor and a pixel blending factor values: See Table 9-1 .	0
Reserved	[19:18]	Reserved, read as zero	0
REG_WIN1_BLEND_EN	[17]	Blending by a blending factor set by REG_ALPHA_WIN1 register on all over the graphic layer1.	0
REG_PIXEL1_BLEND_EN	[16]	Blending by a blending factor of each pixel is enabled in graphic layer1.	0
Reserved	[15:12]	Reserved, read as zero	0

MIXER_GRAPHIC1_CFG	Bit	Description	Initial State
REG_COLOR_FORMAT1	[11:8]	<p>Graphic layer1 color format.</p> <p>0 = Reserved 1 = Reserved 2 = Reserved 3 = Reserved 4 = RGB 565 5 = ARGB 1555 6 = ARGB 4444 7 = ARGB 8888 8 = Reserved 9 = Reserved A = Reserved B = Reserved C = Reserved D = Reserved E = Reserved F = Reserved</p>	0
REG_ALPHA_WIN1	[7:0]	<p>Graphic layer1 and lower layer blending factor. This factor is used all over the pixels in the graphic and lower layer to blend with. A blending factor(α) is as follows depending on the a layer blending factor and a pixel blending factor values:</p> $\alpha * \text{graphic_layer_pixel_value} + (1 - \alpha) * \text{lower_layer_pixel_value}$ <p>See Table 9-1.</p> <p>If REG_ALPHA_WIN1 is 0, blending_factor_layer is 0.</p> <p>If REG_ALPHA_WIN1 is not 0, blending_factor_layer = (REG_ALPHA_WIN + 1) / 256.</p> <p>If A(blending factor of each pixel) is 0, blending_factor_each_pixel is 0</p> <p>If A(blending factor of each pixel) is not 0, blending_factor_each_pixel = (A + 1) / 256.</p> <p>The pixel blending factors comes from the pixel in direct modes except the direct 16bpp 565 format.</p>	0

NOTE: All the changes of this register are valid on a vertical sync signal of next frame.

Table 9-1 Graphic Blending-factor Alpha in Case of Normal Mode

MIXER_GRAPHICx_CFG. REG_WINx_BLEND_EN	MIXER_GRAPHICx_CFGx. REG_PIXELx_BLEND_EN	Alpha Value (Blending factor of each pixel)
0	0	1
0	1	blending_factor_each_pixel(A)
1	0	blending_factor_layer (MIXER_GRAPHICn_CFG[7:0])
1	1	blending_factor_layer * blending_factor_each_pixel

Table 9-2 Graphic Blending Method

Pixel Blend	Window Blend	Normal Mode	Pre Multiplied Mode
0	0	-	-
0	1	$\text{Alpha}_{\text{gw}} * \text{graphic_pixel} + (1 - \text{alpha}_{\text{gw}}) * \text{lower layer}$	$\text{Alpha}_{\text{gw}} * \text{graphic_pixel} + (1 - \text{alpha}_{\text{gw}}) * \text{lower layer}$
1	0	$\text{Alpha}_{\text{gp}} * \text{graphic_pixel} + (1 - \text{alpha}_{\text{gp}}) * \text{lower layer}$	$\text{graphic_pixel} + (1 - \text{alpha}_{\text{gp}}) * \text{lower layer}$
1	1	$(\text{Alpha}_{\text{gp}} * \text{alpha}_{\text{gw}}) * \text{graphic_pixel} + (1 - \text{alpha}_{\text{gp}} * \text{alpha}_{\text{gw}}) * \text{lower layer}$	$\text{alpha}_{\text{gw}} * \text{graphic_pixel} + (1 - \text{alpha}_{\text{gp}} * \text{alpha}_{\text{gw}}) * \text{lower layer}$

In pre-multiplied mode, the input graphic data is multiplied by the pixel blending factor (alpha_{gp}) and truncated to the size of source format bits. For example, although the result of the multiplication of 8-bit data by 8-bit pixel blending factor is 16 bits which is the first term of the blending equation in the [Table 9-2. Normal mode](#), the supplied data is truncated to 8 bits that results the loss of the lower significant 8bits during calculation. In direct 32bpp modes, this loss data cannot be distinguished visually (+/- 1difference). But in direct 16bpp modes, the loss data can make visually different result from the original. For example, in 4633 direct mode, the pre-multiplied Cb data is truncated to 3 bits that result in the difference value from +15 to -15. This difference range can result to visual difference. You cannot reduce the error that is resulted from the 3 bits input source. Thus, it is recommended to use the 32bpp mode to use the pre-multiplied mode.

9.2.2.10 MIXER_GRAPHIC0_BASE, R/W, Address = 0xF920_0024, MIXER_GRAPHIC1_BASE, R/W, Address = 0xF920_0044

MIXER_GRAPHICn_BASE	Bit	Description	Initial State
REG_GRAPHICn_BASE	[31:0]	Base address of frame buffer for graphic layer. This address should be word aligned, so least 2 significant bits [1:0] will be set 2'b00 automatically	0

9.2.2.11 MIXER_GRAPHIC0_SPAN, R/W, Address = 0xF920_0028, MIXER_GRAPHIC1_SPAN, R/W, Address = 0xF920_0048

MIXER_GRAPHICn_SPAN	Bit	Description	Initial State
Reserved	[31:15]	Reserved, read as zero, do not modify	0
REG_GRAPHICn_SPAN	[14:0]	Horizontal pixel interval between line and line in graphic layer's source image Note: SPAN is the number of the original image's horizontal pixel count. For example, 640x480's span is '640'. It is not related to BPP(bit per pixel).	0

9.2.2.12 MIXER_GRAPHIC0_WH, R/W, Address = 0xF920_0030, MIXER_GRAPHIC1_WH, R/W, Address = 0xF920_0050

MIXER_GRAPHICn_WH	Bit	Description	Initial State
Reserved	[31:29]	Reserved, read as zero, do not modify	0
REG_H_SCALEn	[28]	Horizontal scaling Configuration 0 = Not Available 1 = X2 Scaling-Up	0
Reserved	[27]	Reserved, read as zero, do not modify	0
REG_GRAPHICn_W	[26:16]	Width of graphic layer (pixel unit)	0
Reserved	[15:13]	Reserved, read as zero, do not modify	0
REG_V_SCALEn	[12]	Vertical Duplication Configuration 0 = Not Available 1 = X2 Duplication	0
Reserved	[11]	Reserved, read as zero, do not modify	0
REG_GRAPHICn_H	[10:0]	Height of graphic layer (pixel unit).	0

NOTE: All the changes of this register are valid on a vertical sync signal of next frame.

When specifying the X coordinates and the width of a graphic layer, it should be located inside the display region, for example, 720x480 region in NTSC display mode and 720x576 region in PAL display. The coordinates (x, y) and the size (width, height) should be based on the progressive mode although it is interlaced display mode.

Graphic width and height should be larger than 0 if the corresponding REG_GRAPHICx_EN field(MIXER_Cfg[5], MIXER_Cfg[4]) is set to 1.

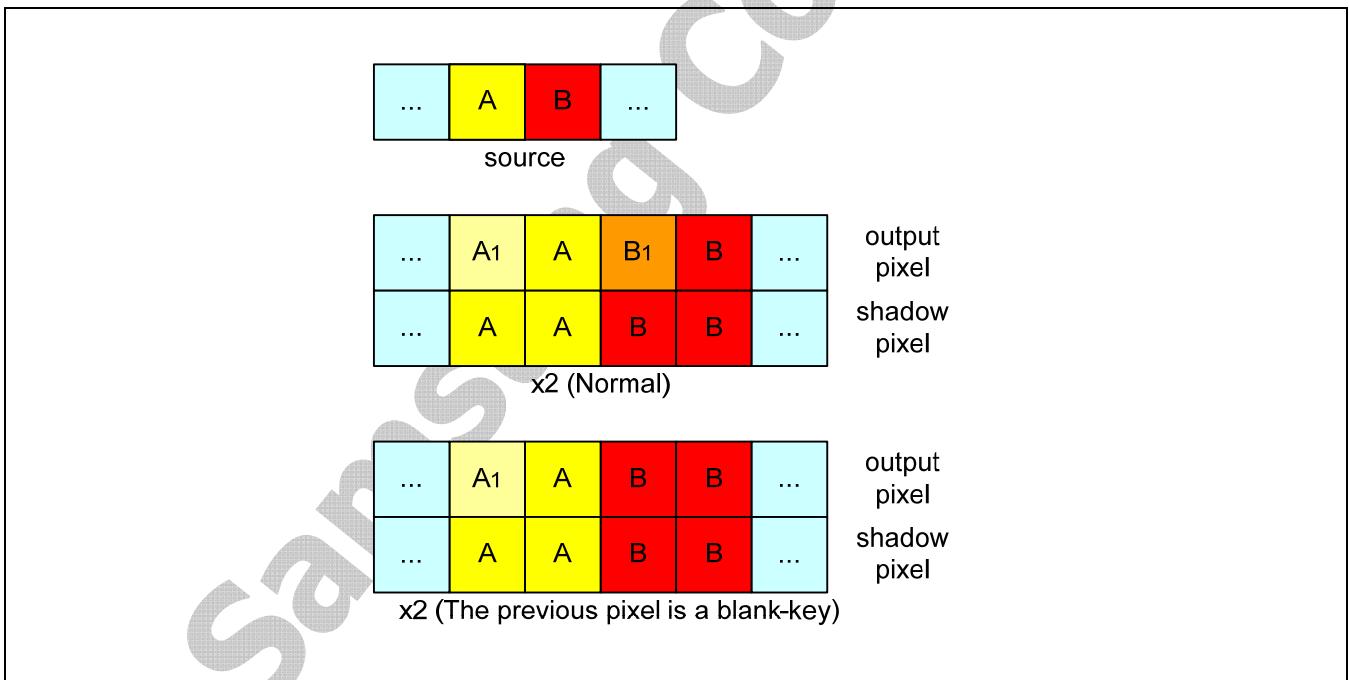


Figure 9-3 Mixer Horizontal Scale and Blank-key

9.2.2.13 MIXER_GRAPHICn_XY, R/W, Address = 0xF920_002C, MIXER_GRAPHIC1_SXY, R/W, Address = 0xF920_004C

MIXER_GRAPHICn_XY	Bit	Description	Initial State
Reserved	[31:27]	Reserved, read as zero, do not modify	0
REG_GRAPHICn_SX	[26:16]	X coordinate of upper left corner of graphic layer in source frame (pixel unit). Allowed range = 0 ~ 719 at SD mode = 0 ~ 1279 at HD mode(720p) = 0 ~ 1919 at HD mode(1080i/p)	0
Reserved	[15:11]	Reserved, read as zero, do not modify	0
REG_GRAPHICn_SY	[10:0]	Y coordinate of upper left corner of graphic layer in source frame (pixel unit). Allowed range = 0 ~ 479 at NTSC mode = 0 ~ 575 at PAL mode = 0 ~ 719 at HD mode(720p) = 0 ~ 1079 at HD mode(1080i/p)	0

9.2.2.14 MIXER_GRAPHIC0_DXY, R/W, Address = 0xF920_0034, MIXER_GRAPHIC1_DXY, R/W, Address = 0xF920_0054

MIXER_GRAPHICn_DXY	Bit	Description	Initial State
Reserved	[31:27]	Reserved, read as zero, do not modify	0
REG_GRAPHICn_DX	[26:16]	X coordinate of upper left corner of graphic layer in destination frame (pixel unit). Allowed range = 0 ~ 719 at SD mode = 0 ~ 1279 at HD mode(720p) = 0 ~ 1919 at HD mode(1080i/p)	0
Reserved	[15:11]	Reserved, read as zero, do not modify	0
REG_GRAPHICn_DY	[10:0]	Y coordinate of upper left corner of graphic layer in destination frame (pixel unit). Allowed range = 0 ~ 479 at NTSC mode = 0 ~ 575 at PAL mode = 0 ~ 719 at HD mode(720p) = 0 ~ 1079 at HD mode(1080i/p)	0

9.2.2.15 MIXER_GRAPHIC0_BLANK, R/W, Address = 0xF920_0038, MIXER_GRAPHIC1_BLANK, R/W, Address = 0xF920_0058

MIXER_GRAPHICn_BLANK	Bit	Description	Initial State
REG_GRAPHICn_BLANK	[31:0]	Blank pixel value for graphic layerN Note: When blanked pixel is ARGB, entire register value must be same with pixel value including alpha value.	0

**9.2.2.16 MIXER_BG_COLOR0, R/W, Address = 0xF920_0064,
MIXER_BG_COLOR1, R/W, Address = 0xF920_0068,
MIXER_BG_COLOR2, R/W, Address = 0xF920_006C**

MIXER_BG_COLOR0/1/2	Bit	Description	Initial State
-	[31:24]	Reserved, read as zero, do not modify	0
Y	[23:16]	Y component of background color	0
Cb	[15:8]	Cb component of background color	0
Cr	[7:0]	Cr component of background color	0

NOTE: You can choose proper YCbCr value for BT.601 or BT.709.

9.2.2.17 MIXER_COLOR_SPACE_CONVERSION_COEF_Y Register (MIXER_CM_COEFF_Y, R/W, Address = 0xF920 0080)

MIXER_CM_COEFF_Y	Bit	Description	Initial State
Reserved	[31]	Reserved, read as zero, do not modify	
WIDE_SEL	[30]	0 = Narrow 1 = Wide	0
REG_COEFF_00	[29:20]	Scaled color space conversion coefficient (C_{00}). [29]: Sign-bit [28:20]: Fractional bit (Default and Recommended value : 0.257 in decimal)	0x84
REG_COEFF_10	[19:10]	Scaled color space conversion coefficient (C_{10}). [19]: Sign-bit [18:10]: Fractional bit (Default and Recommended value : 0.504 in decimal)	0x102
REG_COEFF_20	[9:0]	Scaled color space conversion coefficient (C_{20}). [9]: Sign-bit [8:0]: Fractional bit (Default and Recommended value : 0.098 in decimal)	0x32

NOTE: RGB to YCbCr Conversion Equations

RGB data with 16-235 range

$$\begin{aligned} Y601 &= 0.299R + 0.587G + 0.114B \\ Cb &= -0.172R - 0.339G + 0.511B + 128 \\ Cr &= 0.511R - 0.428G - 0.083B + 128 \end{aligned}$$

$$\begin{aligned} Y709 &= 0.213R + 0.715G + 0.072B \\ Cb &= -0.117R - 0.394G + 0.511B + 128 \\ Cr &= 0.511R - 0.464G - 0.047B + 128 \end{aligned}$$

RGB data with 0-255 range

$$\begin{aligned} Y601 &= 0.257R + 0.504G + 0.098B + 16 \\ Cb &= -0.148R - 0.291G + 0.439B + 128 \\ Cr &= 0.439R - 0.368G - 0.071B + 128 \end{aligned}$$

$$\begin{aligned} Y709 &= 0.183R + 0.614G + 0.062B + 16 \\ Cb &= -0.101R - 0.338G + 0.439B + 128 \\ Cr &= 0.439 - 0.399G - 0.040B + 128 \end{aligned}$$

Fraction Number (Example, 1bit is Signed bit and 9bits are Fraction Bit)

$$0.098 = 0.5^* '0' + 0.25^* '0' + 0.125^* '0' + 0.0625^* '1' + 0.03125^* '1' + 0.015625^* '0' + 0.0078125^* '0' + 0.00390625^* '1' + 0.001953125^* '0' = 0(\text{signed bit}) 0\ 0\ 0\ 1\ 1\ 0\ 0\ 1\ 0 = 0x032$$

-0.148 -> First of all, let's think about 0.148

$$0.148 = 0.5^* '0' + 0.25^* '0' + 0.125^* '1' + 0.0625^* '0' + 0.03125^* '0' + 0.015625^* '1' + 0.0078125^* '0' + 0.00390625^* '1' + 0.001953125^* '1' = 0(\text{signed bit}) 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1 = 10'b0001001011$$

Now, to change the number from 0.148 to -0.148, we have to derive 2's compliment of 0.148

$$10'b\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1 \rightarrow 10'b\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0 \text{ (bitwise invert)} + 1 = 10'b\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 1 = 0x3B5$$

9.2.2.18 MIXER_COLOR_SPACE_CONVERSION_COEF_CB Register (MIXER_CM_COEFF_CB, R/W, Address = 0xF920_0084)

MIXER_CM_COEFF_CB	Bit	Description	Initial State
Reserved	[31:30]	Reserved, read as zero, do not modify	0
REG_COEFF_01	[29:20]	Scaled color space conversion coefficient (C_{01}). [29]: Sign-bit [28:20]: Fractional bit (Default value: -0.0742785 in decimal) (Recommended value: -0.148 in decimal, 0x3b5 in hexa-decimal)	0x3b4
REG_COEFF_11	[19:10]	Scaled color space conversion coefficient (C_{11}). [19]: Sign-bit [18:10]: Fractional bit (Default value: -0.1455078125 in decimal) (Recommended value: -0.291 in decimal, 0x36c in hexa-decimal)	0x36b
REG_COEFF_21	[9:0]	Scaled color space conversion coefficient (C_{21}). [9]: Sign-bit [8:0]: Fractional bit (Default and Recommended value: 0.439 in decimal)	0xe1

9.2.2.19 MIXER_COLOR_SPACE_CONVERSION_COEF_CR Register (MIXER_CM_COEFF_Cr, R/W, Address = 0xF920_0088)

MIXER_CM_COEFF_CR	Bit	Description	Initial State
Reserved	[31:30]	Reserved, read as zero, do not modify	0
REG_COEFF_02	[29:20]	Scaled color space conversion coefficient (C_{02}). [29]: Sign-bit [28:20]: Fractional bit (Default and Recommended value: 0.439 in decimal)	0xe1
REG_COEFF_12	[19:10]	Scaled color space conversion coefficient (C_{12}). [19]: Sign-bit [18:10]: Fractional bit (Default and Recommended value: -0.368 in decimal)	0x344
REG_COEFF_22	[9:0]	Scaled color space conversion coefficient (C_{22}). [9]: Sign-bit [8:0]: Fractional bits (Default and Recommended value: -0.071 in decimal)	0x3dc

9.3 LAYERS

Mixer blends all the image sources such as video layer, graphic layer, and background layer and transfers the blended pixel data to TVNEC/HDMI. Set layer's priority value to select the order of the blending operation. Video and 2 graphic layers can be fully ordered and blended. The background layer is always the lowest layer.

- Background → (Video ↔ Graphic0 ↔ Graphic1)

Video layer and graphic layer are enabled or disabled by the register setting. The blending factor differs layer by layer like following:

- Background layer: No blending factor, as it is the lowest layer.
- Video layer: Video layer has one blending factor that is applied to all the pixels in the video layer. MIXER_VIDEO_CFG [7:0](REG_ALPHA_VID) is the blending factor. The video blending is enabled or disabled.
- Graphic0 layer: Graphic0 layer supports pixel blending and window blending. Pixel blending factors are applied pixel-by-pixel although window blending factor is applied all the pixels in the graphic layer. These two blending factors are applied simultaneously to a pixel.
- Graphic1 layer: Graphic1 layer supports pixel blending and window blending. Pixel blending factors are applied pixel-by-pixel although window blending factor is applied on all the pixels in the graphic layer. These two blending factors are applied simultaneously to a pixel.

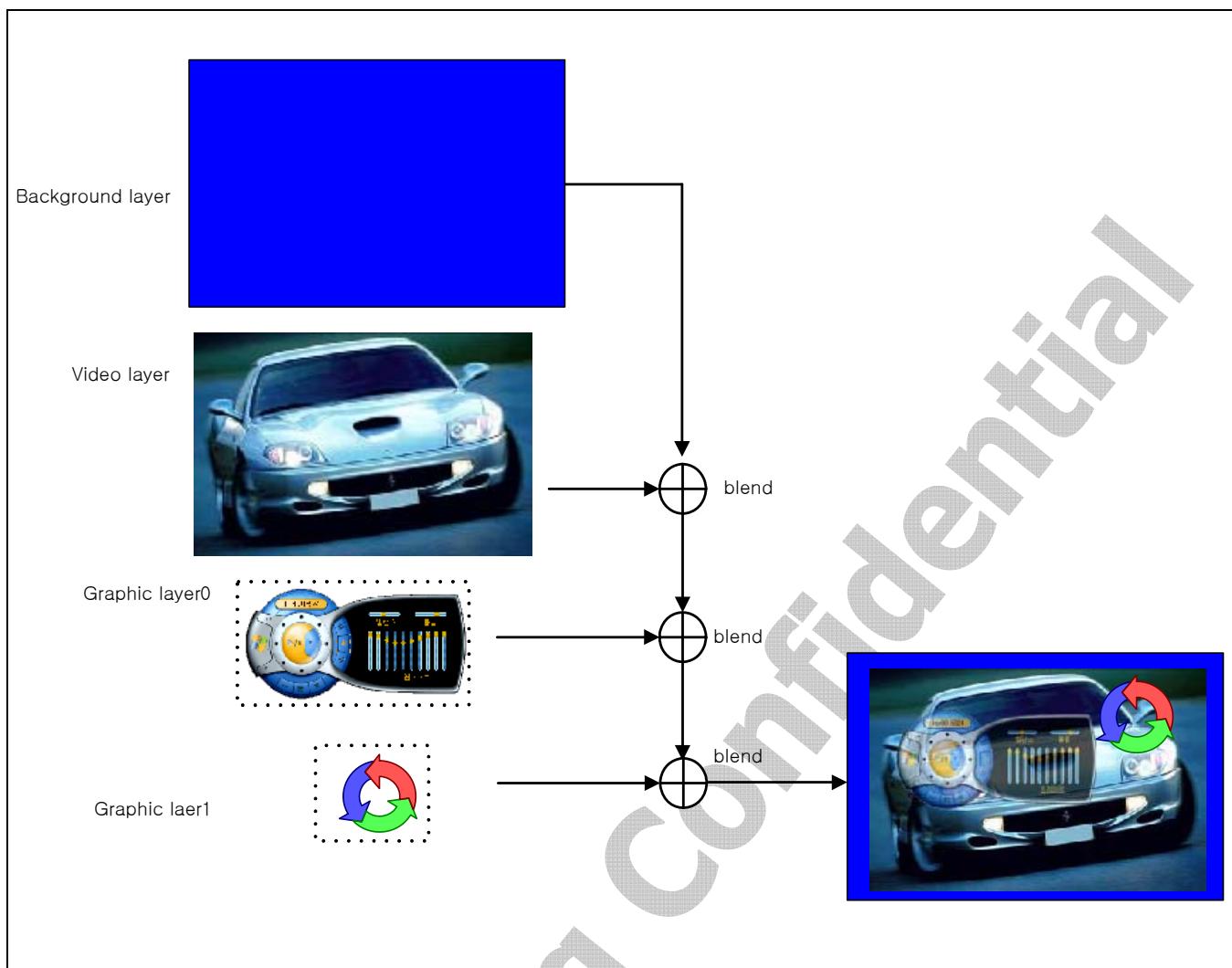


Figure 9-4 Mixer Blending

9.3.1 VIDEO LAYER

Video data is directly transferred from Video Processor to Mixer in YCbCr [888] 4:4:4 formats. As the Video Processor scales the source image in letterbox mode, the display region of the video data is smaller than the screen size. In this case, background layer is seen in the blank region.

9.3.2 GRAPHIC LAYER

ARM or Graphic Accelerator generates the graphic source data in the external memory and they are transferred to Mixer by AXI access. Mixer supports the following graphic formats.

- 16bpp RGB [565]
- 16bpp ARGB [1555]
- 16bpp ARGB [4444]
- 32bpp ARGB [8888]

In 16/32-bpp direct modes, the value of a pixel data directly indicates the RGB but the bit width for R, G, and B are different for each mode. For 16bpp direct ARGB [4444] mode, the RGB component is assigned to 16 bit length like the following.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
alpha factor				R				G				B			

Figure 9-5 16bpp ARGB Example

The internal data path is processed with YCbCr[888] format, therefore RGB format is converted to YCbCr by color matrix conversion.

If the bit-per-pixel (BPP) of color format is smaller than 8 bits, that value is used after expanding (Refer [Figure 9-6](#) example (ARGB [1555])).

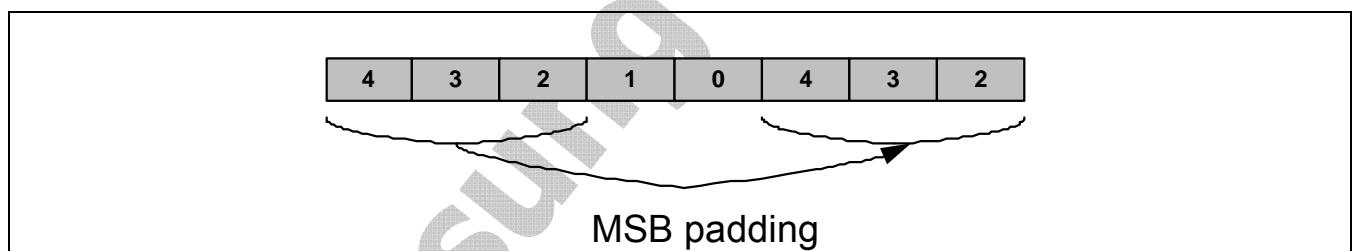


Figure 9-6 Example of Expanding

Mixer supports up to two graphic layers and one video layer. Each layer is enabled or disabled and user can configure the priority between layers. There is each blending factor between layers. The graphic layers has different color format.

When specifying the X/Y coordinates and the width/height of a graphic layer, the graphic layers should be located in the display region (720x480 in NTSC display mode, 720x576 in PAL display mode, 1280x720p/1920x1080i&p in HD display mode). The Mixer does not support the clipping operation for the pixels that are displayed out of screen.

9.3.3 BLANK PIXEL

Blank pixel data in graphic layer is a pixel data that is transparent to the lower layer. You can define a blank pixel data in the register (MIXER_GRAPHICn_BLANK) and if the graphic data is same as the blank pixel value, a lower image is seen instead of the blank pixel.

9.3.4 SOURCE DATA IN MEMORY

As the graphic data comes from the external memory through the bus, the memory format for the source data is dependent on the bus system endian. In little endian system, the lower 8 bits are stored in the lower address.

The source data format in the Mixer is aligned in little endian or big endian format. This different endian format is applicable to the graphic data.

Mixer supports many graphic formats. The register format of the supported source formats is shown in [Figure 9-7](#). The following picture shows the pixels in a display that is seen through human eyes.

LittleEndian															
ARGB8888															
A1	R1	G1	B1	A0	RO	GO	BO								
63	47			31		15	0								
ARGB 4444															
A3	R3	G3	B3	A2	R2	G2	B2	A1	R1	G1	B1	A0	RO	GO	BO
63	47				31			15						0	
RGB 565															
R3	G3	B3	R2	G2	B2	R1	G1	B1	RO	GO	BO				
63	47			31				15						0	
RGB 1555															
A	R3	G3	B3	A	R2	G2	B2	A	R1	G1	B1	A	RO	GO	BO
63	47				31			15						0	
BigEndian															
ARGB8888															
A0	RO	GO	BO	A1	R1	G1	B1	A1	R1	G1	B1				
63	47				31			15						0	
ARGB 4444															
AO	RO	GO	BO	A1	R1	G1	B1	A2	R2	G2	B2	A3	R3	G3	B3
63	47				31			15						0	
RGB 565															
RO	GO	BO	R1	G1	B1	R2	G2	B2	R3	G3	B3				
63	47			31				15						0	
RGB 1555															
A	RO	GO	BO	A	R1	G1	B1	A	R2	G2	B2	A	R3	G3	B3
63	47				31			15						0	

Figure 9-7 Graphic Data Format in Memory

Pixels with lower X coordinates are located from the left. These pixel data are represented by different digital data depending on the graphic format setting. For example, one graphic pixel is represented by 16-bit digital data in 16 BPP mode. If these pixels are processed in the Mixer, the word format to represent these pixels is different depending on the endian format.

In the big endian mode, the pixel with lower X coordinate is positioned to the MSB parts in the 64-bit register of Mixer. However, in the little endian mode, the pixel with lower X coordinate is positioned to the LSB parts in the 64-bit register.

9.3.5 BACKGROUND LAYER

If there is no video or graphic, background color is seen in the display region. Use YCbCr[888] format to set the background color in the register.

10

HIGH-DEFINITION MULTIMEDIA INTERFACE

10.1 OVERVIEW OF HIGH-DEFINITION MULTIMEDIA INTERFACE

The High-Definition Multimedia Interface (HDMI) 1.3 Tx Subsystem V1.0 comprises an HDMI Tx Core with I2S/SPDIF input interface, CEC block, and HDCP Key Block.

10.1.1 KEY FEATURES OF HDMI

The key features of HDMI include:

- Complies with HDMI 1.3, HDCP 1.1, and DVI 1.0

Supports the following video formats:

- 480p @59.94Hz/ 60Hz, 576p @50Hz
- 720p @50Hz/ 59.94Hz/ 60Hz
- 1080i @50Hz/ 59.94Hz/ 60Hz
- 1080p @25Hz/ 29.97Hz/ 30Hz
- Other various formats up to 74.25 MHz Pixel Clock

Supports Color Format: 4:4:4 RGB/ YCbCr

Supports Bit Per Color: 8-bit

Supports CEC function

Contains an Integrated HDCP Encryption Engine for Video/ Audio content protection

Does not include DDC

- There is a dedicated I2C for DDC in S5PV210 peripheral Bus

Supports audio RX transmission when I2S in audio sub-system is either master or slave mode.

-

10.1.2 BLOCK DIAGRAM OF HDMI

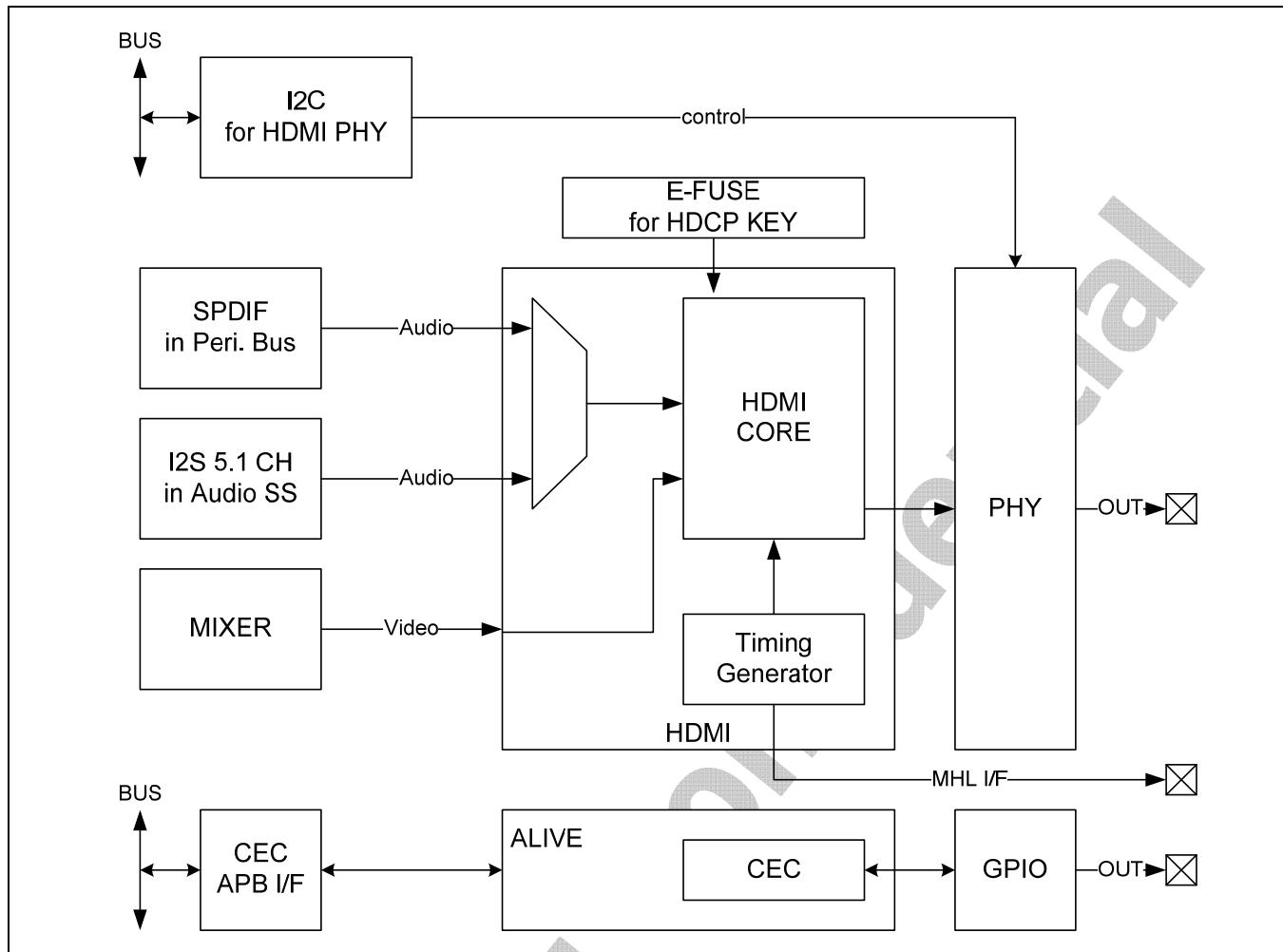


Figure 10-1 Block Diagram of HDMI

HDMI TX V1.3 consists of several blocks, each with a unique function. For instance, the Mixer specifies the source image of HDMI. It transmits image data, which can be either RGB888 or YUV444. Before it works, you must set the pixel clock properly. The ratio of pixel clock depends on the output resolution. You can configure both CMU and HDMI_PHY.

SPDIF in peripheral bus and I2S 5.1 channel in Audio sub-system feed audio data in HDMI TX V1.3. For more information, refer to SPDIF and I2S datasheets.

HDMI TX V1.3 in S5PV210 supports embedded HDCP key system. S5PV210 does not allow access to HDCP key.

A dedicated I2C is used to configure HDMI PHY. In addition, the HDMI PHY generates pixel and TMDS clock through I2C.

The CEC block is separate from HDMI TX, and is used by wake-up source in S5PV210. It belongs to the ALIVE block and communicates with external CEC through bi-directional GPIO.

10.1.3 BLOCK DIAGRAM OF HDMI SUB-SYSTEM IN S5PV210

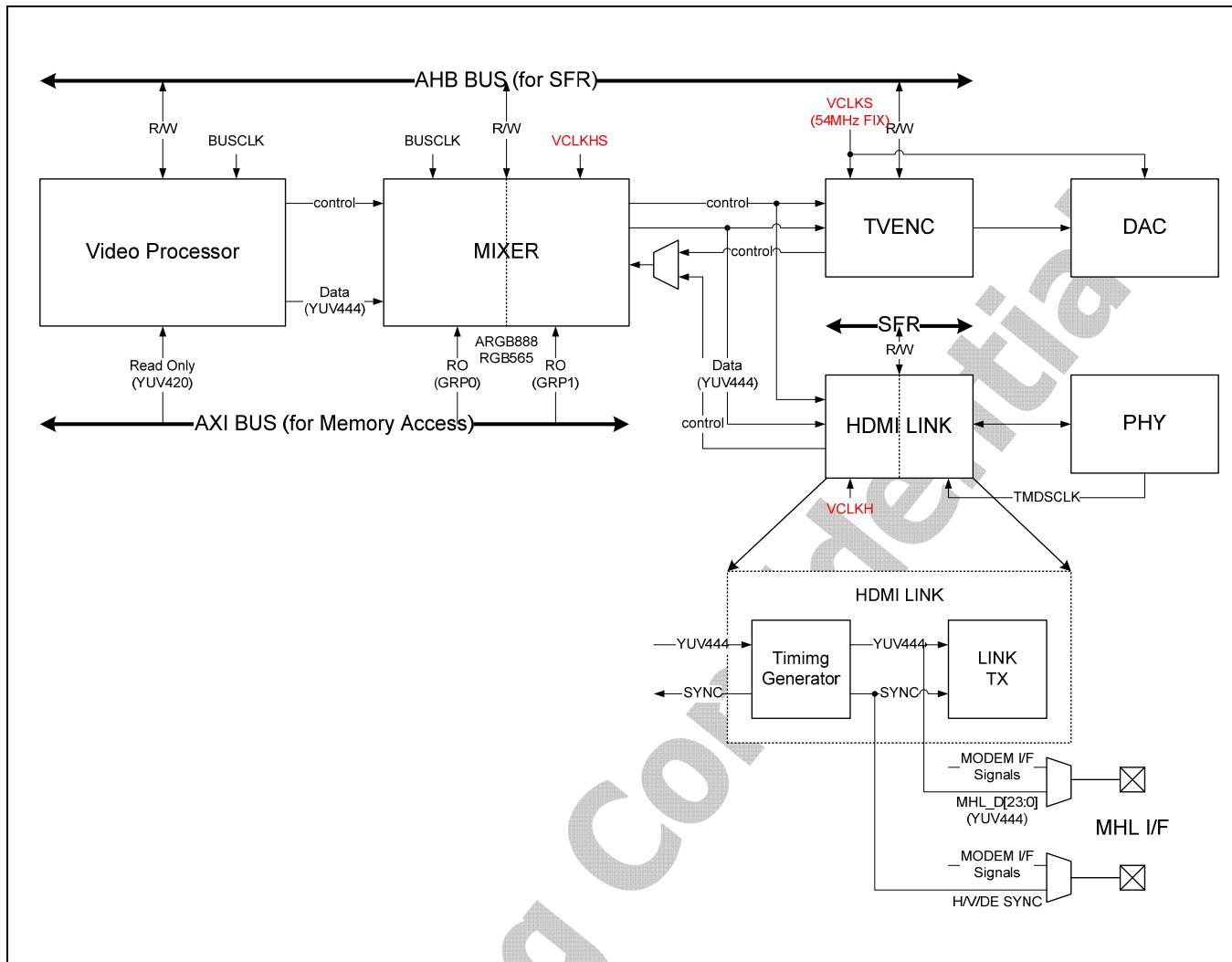


Figure 10-2 Block Diagram of HDMI SUB-System in S5PV210

10.1.4 BLOCK DIAGRAM OF HDCP KEY MANAGEMENT

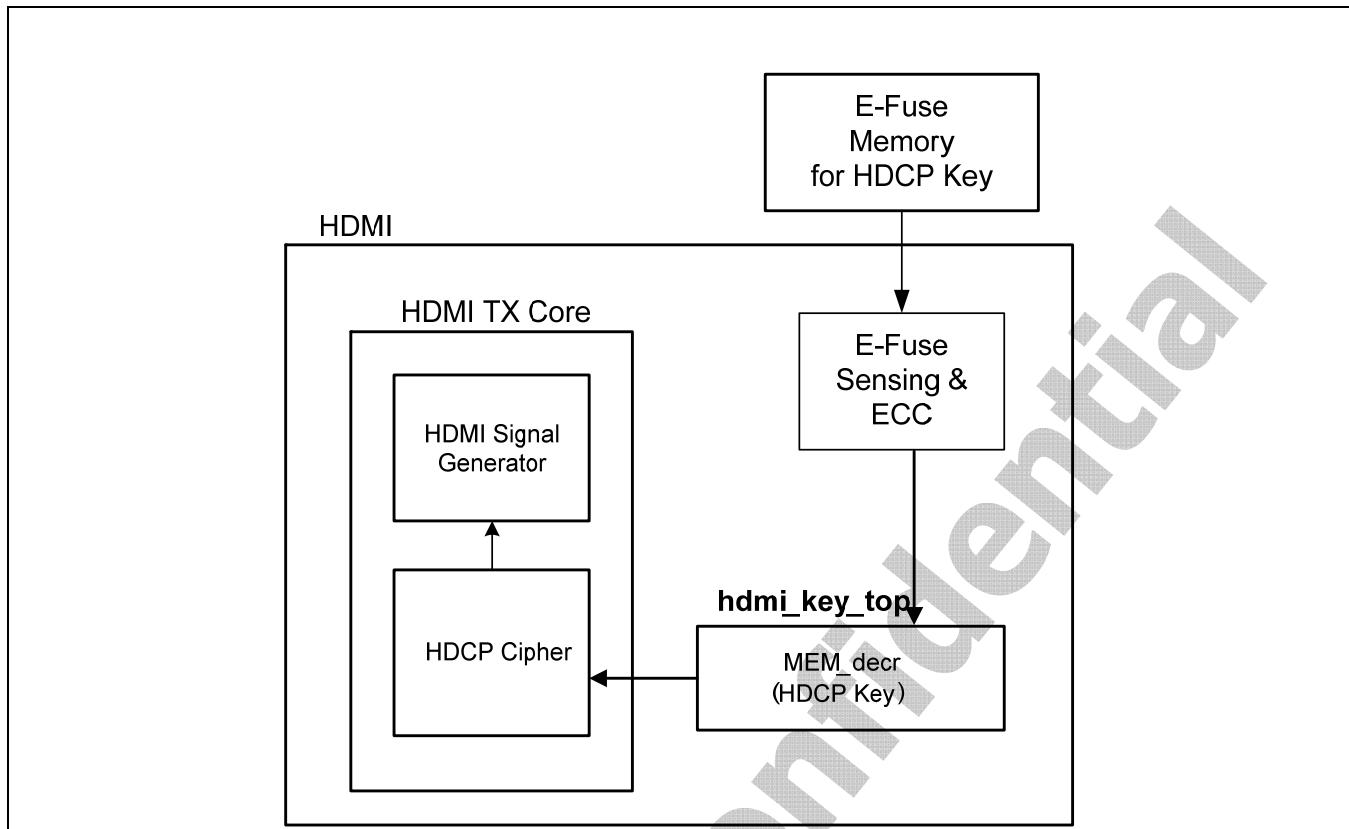


Figure 10-3 Block Diagram of HDCP Key Management

S5PV210 supports embedded HDCP key system. The HDCP key value is fused during fabrication, based on customers' request. S5PV210 strictly prohibits access to HDCP key value from any method. After S5PV210 boot up, the HDCP key is loaded using SFR from E-FUSE memory (HDCP_E_FUSE_CTRL, 0xFA16_000, [0] bit).

10.1.5 VIDEO INPUT TIMING GUIDE FOR HDMI TIMING GENERATOR

Table 10-1 HDMI LINK Timing Generator Configuration Guide

	720x480p	720x576p	1280x720p	1920x1080i	1920x1080p
TG_H_FSZ_L(0xFA15_0018)	0x5a	0x60	0x72	0x98	0x98
TG_H_FSZ_H(0xFA15_001C)	0x03	0x03	0x06	0x08	0x08
TG_HACT_ST_L(0xFA15_0020)	0x8a	0x90	0x72	0x18	0x18
TG_HACT_ST_H(0xFA15_0024)	0x00	0x00	0x01	0x01	0x01
TG_HACT_SZ_L(0xFA15_0028)	0xd0	0xd0	0x00	0x80	0x80
TG_HACT_SZ_H(0xFA15_002C)	0x02	0x02	0x05	0x07	0x07
TG_V_FSZ_L(0xFA15_0030)	0x0d	0x71	0xee	0x65	0x65
TG_V_FSZ_H(0xFA15_0034)	0x02	0x02	0x02	0x04	0x04
TG_VSYNC_L(0xFA15_0038)	0x01	0x01	0x01	0x01	0x01
TG_VSYNC_H(0xFA15_003C)	0x00	0x00	0x00	0x00	0x00
TG_VSYNC2_L(0xFA15_0040)	Reset value	Reset value	Reset value	0x33	Reset value
TG_VSYNC2_H(0xFA15_0044)	Reset value	Reset value	Reset value	0x02	Reset value
TG_VACT_ST_L(0xFA15_0048)	0x2d	0x31	0x1e	0x16	0x2d
TG_VACT_ST_H(0xFA15_004C)	0x00	0x00	0x00	0x00	0x00
TG_VACT_SZ_L(0xFA15_0050)	0xe0	0xe0	0xd0	0x1c	0x38
TG_VACT_SZ_H(0xFA15_0054)	0x01	0x01	0x02	0x02	0x04
TG_FIELD_CHG_L(0xFA15_0058)	Reset value	Reset value	Reset value	0x33	Reset value
TG_FIELD_CHG_H(0xFA15_005C)	Reset value	Reset value	Reset value	0x02	Reset value
TG_VACT_ST2_L(0xFA15_0060)	Reset value	Reset value	Reset value	0x49	Reset value
TG_VACT_ST2_H(0xFA15_0064)	Reset value	Reset value	Reset value	0x02	Reset value
TG_VSYNC_TOP_HDMI_L (0xFA15_0078)	0x01	0x01	0x01	0x01	0x01
TG_VSYNC_TOP_HDMI_H (0xFA15_007C)	0x00	0x00	0x00	0x00	0x00
TG_VSYNC_BOT_HDMI_L (0xFA15_0080)	Reset value	Reset value	Reset value	0x33	Reset value
TG_VSYNC_BOT_HDMI_H (0xFA15_0084)	Reset value	Reset value	Reset value	0x02	Reset value
TG_FIELD_TOP_HDMI_L (0xFA15_0088)	0x01	0x01	0x01	0x01	0x01
TG_FIELD_TOP_HDMI_H (0xFA15_008C)	0x00	0x00	0x00	0x00	0x00

	720x480p	720x576p	1280x720p	1920x1080i	1920x1080p
TG_FIELD_BOT_HDMI_L (0xFA15_0090)	Reset value	Reset value	Reset value	0x33	Reset value
TG_FIELD_BOT_HDMI_H (0xFA15_0094)	Reset value	Reset value	Reset value	0x02	Reset value

Samsung Confidential

10.1.6 HDMI PHY CONFIGURATION

HDMI PHY is configured using a dedicated I2C, which is only used in TX mode. The address of HDMI PHY is 0x70. The sequence of I2C data is shown in [Figure 10-4](#).

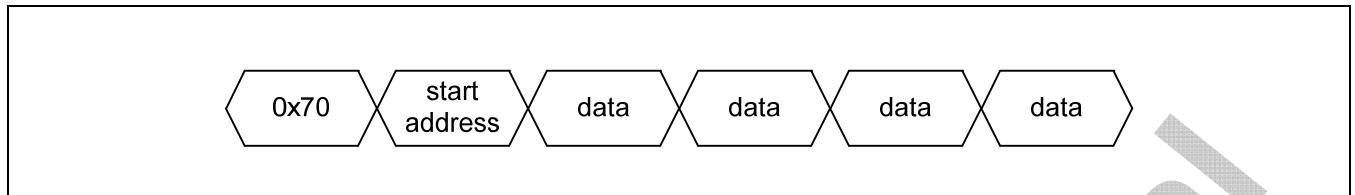


Figure 10-4 Sequence of I2C Data

We recommend following sequence for HDMI PHY configuration

- 1) Clock path change : CLK_SRC1 [0]_bit (0xE010_0204) is set to "0" (SCLK_PIXEL)
- 2) HDMI PHY configuration through I2C : refer to below table.
- 3) HDMI LINK core reset : CORE_RSTOUT [0]_bit (0xFA10_0020) is set to "0" for 100us.
- 4) PHY ready check
- 5) Clock path change : CLK_SRC1 [0]_bit (0xE010_0204) is set to "1" (SCLK_HDMIPHY)

Upper sequence is prior to configuration of VP, MIXER and HDMI LINK.

Due to the security policy, below table's configuration is only opened, as shown in [Table 10-2](#).

Table 10-2 HDMI PHY Configuration Table for 27MHz OSC_In

Addr	27MHz (Pixel Clock Ratio)	27.027MHz	74.176MHz	74.25MHz
	8b	8b	8b	8b
0x01	05h	05h	05h	05h
0x02	00h	00h	00h	00h
0x03	D8h	D8h	D8h	D8h
0x04	10h	10h	10h	10h
0x05	1Ch	9Ch	9Ch	1Ch
0x06	30h	02h	56h	30h
0x07	40h	32h	5Bh	40h
0x08	6Bh	6Bh	6Bh	6Bh
0x09	10h	10h	10h	10h
0x0A	02h	02h	01h	01h
0x0B	52h	52h	52h	52h
0x0C	4Fh	4Fh	Bfh	7Fh

Addr	27MHz (Pixel Clock Ratio)	27.027MHz	74.176MHz	74.25MHz
	8b	8b	8b	8b
0x0D	F1h	F1h	F1h	F3h
0x0E	54h	54h	54h	54h
0x0F	78h	78h	A5h	A5h
0x10	84h	84h	84h	84h
0x11	00h	00h	00h	00h
0x12	00h	00h	00h	20h
0x13	38h	38h	38h	38h
0x14	00h	00h	00h	00h
0x15	08h	08h	08h	08h
0x16	10h	10h	10h	10h
0x17	E0h	E0h	E0h	E0h
0x18	22h	22h	22h	22h
0x19	40h	40h	40h	40h
0x1A	FFh	FFh	B9h	B9h
0x1B	26h	26h	26h	26h
0x1C	00h	00h	01h	01h
0x1D	00h	00h	00h	00h
0x1E	00h	00h	00h	00h
0x1F	80h	80h	80h	80h

Address 0x1f specifies the PHY_START control. If PHY is configured, the address 0x1f must be 0x0.

NOTE: It can be various configurations which depend on PCB environment.

Table 10-3 HDMI PHY Configuration Table for 24MHz OSC_In

Addr	27MHz (Pixel Clock Ratio)	27.027MHz	74.176MHz	74.25MHz
	8b	8b	8b	8b
0x01	05h	05h	05h	05h
0x02	00h	00h	00h	00h
0x03	D8h	D4h	D8h	D8h
0x04	10h	10h	10h	10h
0x05	1Ch	9Ch	9Ch	9Ch
0x06	30h	09h	EFh	F8h
0x07	40h	64h	5Bh	40h
0x08	6Bh	6Bh	6Dh	6Ah
0x09	10h	10h	10h	10h
0x0A	02h	02h	01h	01h
0x0B	52h	52h	52h	52h
0x0C	DFh	DFh	EFh	FFh
0x0D	F2h	F2h	F3h	F1h
0x0E	54h	54h	54h	54h
0x0F	87h	87h	B9h	BAh
0x10	84h	84h	84h	84h
0x11	00h	00h	00h	00h
0x12	30h	30h	30h	10h
0x13	38h	38h	38h	38h
0x14	00h	00h	00h	00h
0x15	08h	08h	08h	08h
0x16	10h	10h	10h	10h
0x17	E0h	E0h	E0h	E0h
0x18	22h	22h	22h	22h
0x19	40h	40h	40h	40h
0x1A	E3h	E2h	A5h	A4h
0x1B	26h	26h	26h	26h
0x1C	00h	00h	01h	01h
0x1D	00h	00h	00h	00h
0x1E	00h	00h	00h	00h
0x1F	80h	80h	80h	80h

Address 0x1f specifies the PHY_START control. If PHY is configured, the address 0x1f must be 0x0.

NOTE: It can be various configurations which depend on PCB environment.

10.1.7 SELECTED I2C REGISTER CONTROL

Name	Code	Description
Reference Clock Selection (Reg 12 bit[4])	0	External Crystal
	1	Internal Oscillator Input(XXTI or XusbXTI)
	It selects reference clock between external crystal and internal oscillator input.	
Pixel Clock Selection (Reg 12 bit[7])	0	Internally generated pixel clock by internal video pll
	1	Externally supplied pixel clock by external VPLL. (refer to Figure 10-5)
	The internal video pll can be used for pixel clock generation. If the externally supplied pixel clock is used, the internal video pll can be configured as jitter-filter pll or by-passed. Because of clock-jitter, we recommend the internal video pll.	
Power Down	0	Operation mode
	1	Power-down mode
	REG01 bit[5] : Bias power down REG01 bit[7] : Sigma delta modulator clock generator power down REG05 bit[5] : PLL power down REG17 bit[0] : PCG power down REG17 bit[1] : TX power down	
TMDS Data Amplitude Control (Reg 18 bit[3:0])	4'b1111	Maximum amplitude
	4'b0000	Minimum amplitude
	TMDS data amplitude control	
TMDS Clock Amplitude Control (Reg 18 bit[7:4])	4'b1111	Maximum amplitude
	4'b0000	Minimum amplitude
	TMDS clock amplitude control	
TMDS Data Pre-emphasis Control (Reg 19 bit[2:0])	3'b000	No pre-emphasis
	3'b111	Max pre-emphasis
	TMDS data pre-emphasis	
Oscillator Pad Control (Reg 19 bit[7:6])	2'b01	Oscillator pad is on.
	2'b10	Oscillator pad is off.
	Oscillator pad control	
PHY Mode Set Done (Reg 0x1F bit[7])	0	Mode setting is in progress
	1	Mode setting is done
	For PHY mode setting without reset, this bit is used as mode setting status flag. If this bit is "0", mode setting through I2C is in progress. If all setting is done, this bit should be set "1" again.	

10.1.8 I/O DESCRIPTION OF HDMI PHY

Pin Name	Pin Function	Description
XhdmiTX0P	Output	TMDS output data pairs.
XhdmiTX0N	Output	
XhdmiTX1P	Output	
XhdmiTX1N	Output	
XhdmiTX2P	Output	
XhdmiTX2N	Output	
XhdmiTXCP	Output	TMDS output clock pair.
XhdmiTXCN	Output	
XhdmiREXT	Input	External Reference Resistor. External reference resistor input. A 4.6K, 1% resistor is connected to ground.
XhdmiXTI	Input	Reference Clock Input. Crystal oscillator input. It is used to generate internal clock signals. Its nominal frequency is 27MHz.
XhdmiXTO	Output	Reference Clock Output. Crystal oscillator output.

10.1.9 BLOCK DIAGRAM OF CLOCK STRATEGY FOR HDMI TX

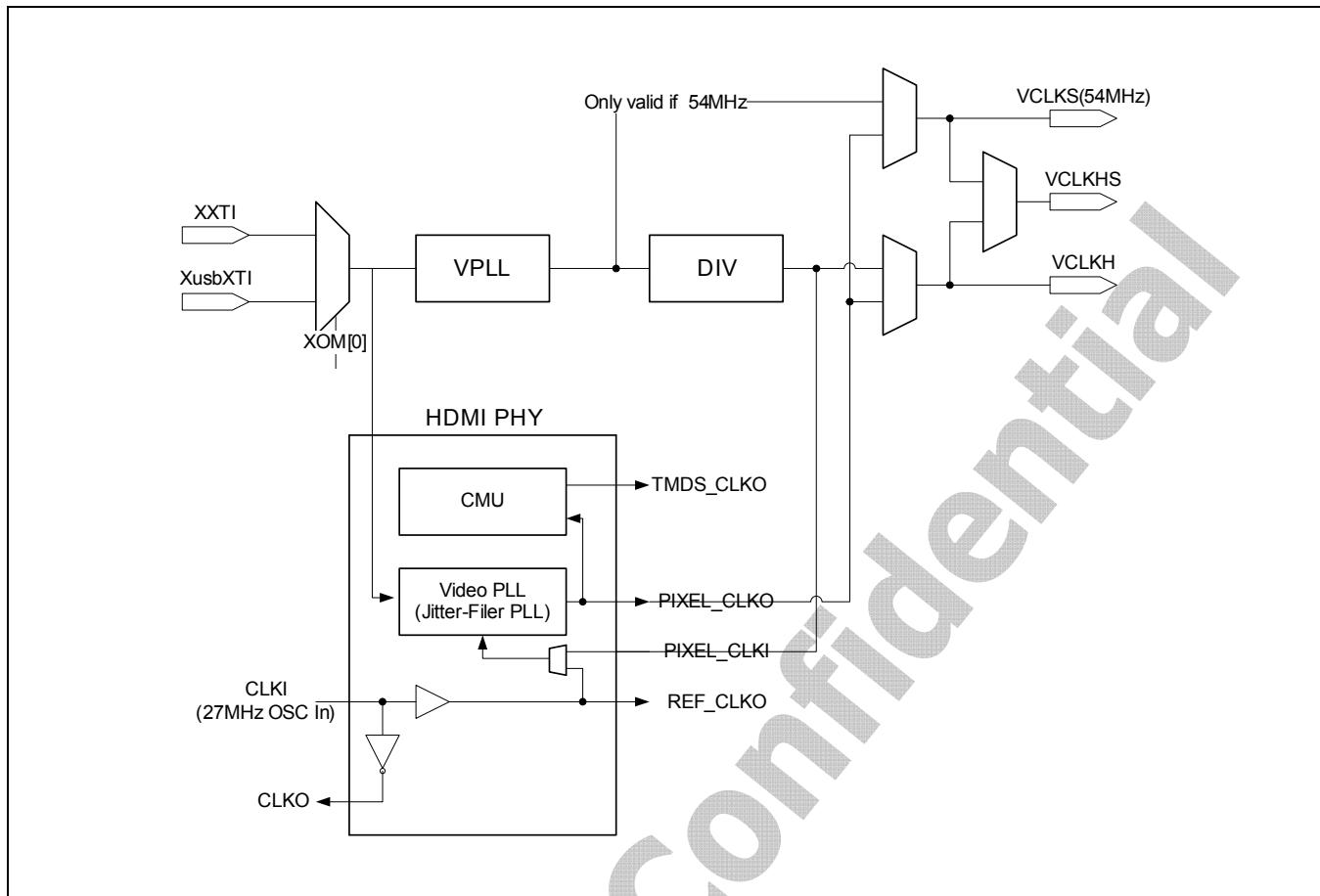


Figure 10-5 Block Diagram of HDMI TX Clock Scheme in S5PV210

The HDMI link part uses pixel and TMDS clock. Pixel and TMDS clocks are from HDMI PHY. You must configure it before use. VCLKHS (MIXER pixel clock) and VCLKH (HDMI pixel clock) are synchronous. Thus, the same clock is fed through VCLKHS and VCLKH. For pixel frequency, refer to [Figure 10-6](#).

VCLKHS(or VCLKH) usage frequency		
Vertical Freq.	Format	Pixel Freq.
59.94 Hz	480P	27MHz (or 54MHz)
	720P	74.175MHz
	1080I	74.175MHz
60.00 Hz	480P	27.027MHz (or 54.054MHz)
	720P	74.250MHz
	1080I	74.250MHz
50.00 Hz	576P	27MHz (or 54MHz)
	720P	74.250MHz
	1080I	74.250MHz
29.97Hz	1080P	74.175MHz
30Hz	1080P	74.250MHz
25Hz	1080P	74.250MHz
Summary		27MHz
		27.027MHz
		54MHz
		54.054MHz
		74.175MHz
		74.250MHz

Figure 10-6 Frequency Summary in Use

10.2 SPDIF (AUXILIARY INFORMATION)

10.2.1 FRAME FORMAT

A frame is composed of two sub-frames. The transmission rate of frames corresponds exactly to the source sampling frequency. In the 2-channel operation mode, the samples taken from both channels are transmitted by time multiplexing in consecutive sub-frames.

Usually, sub-frames related to channel 1 (left or "A" channel in stereophonic operation and primary channel in monophonic operation) use preamble M. However, the preamble is changed to preamble B once every 192 frames. This unit is composed of 192 frames. It defines the block structure used to organize the channel status information.

On the other hand, sub-frames of channel 2 (right or "B" in stereophonic operation and secondary channel in monophonic operation) always use preamble W. In single channel operation mode and broadcasting studio environment, the frame format is identical to 2-channel mode. The data is only carried in channel 1. In the sub-frames allocated to channel 2, time slot 28 (validity flag) is set to logical "1" (invalid).

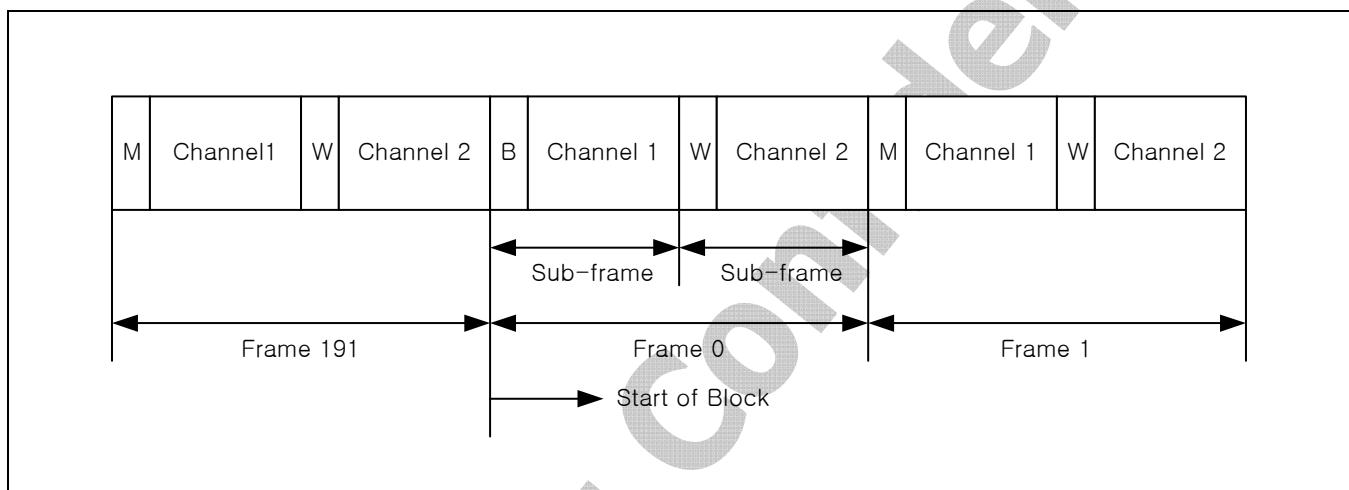


Figure 10-7 Frame Format

10.2.1.1 Sub-frame Format (IEC 60958)

Each sub-frame is divided into 32 time slots numbered from 0 to 31. Time slots from 0 to 3 carry one of the three permitted preambles. These slots affect the synchronization of sub-frames, frames, and blocks. Time slots from 4 to 27 carry the audio sample word in linear 2's complement representation.

The most significant bit (MSB) is carried by time slot 27. When a 24-bit coding range is used, the least significant bit (LSB) is in the time slot 4.

When a 20-bit coding range is sufficient, the LSB is in the time slot 8, and time slots from 4 to 7 may be used for other applications. Under these circumstances, the bits in the time slots 4 to 7 are designated auxiliary sample bits.

If the source provides fewer bits than what the interface allows (24 or 20), the unused LSBs are set to a logical "0". By this procedure, the equipment using different numbers of bits can be connected together.

- Time slot 28 carries the validity flag associated with audio sample word. This flag is set to logical "0" if the audio sample is reliable.
- Time slot 29 carries one bit of user data associated with audio channel that is transmitted in the same sub-frame. The default value of user bit is logical "0".
- Time slot 30 carries one bit of channel status words associated with audio channel that is transmitted in the same sub-frame.
- Time slot 31 carries a parity bit such that time slots from 4 to 31 (inclusive) will carry an even number of ones and an even number of zeros.

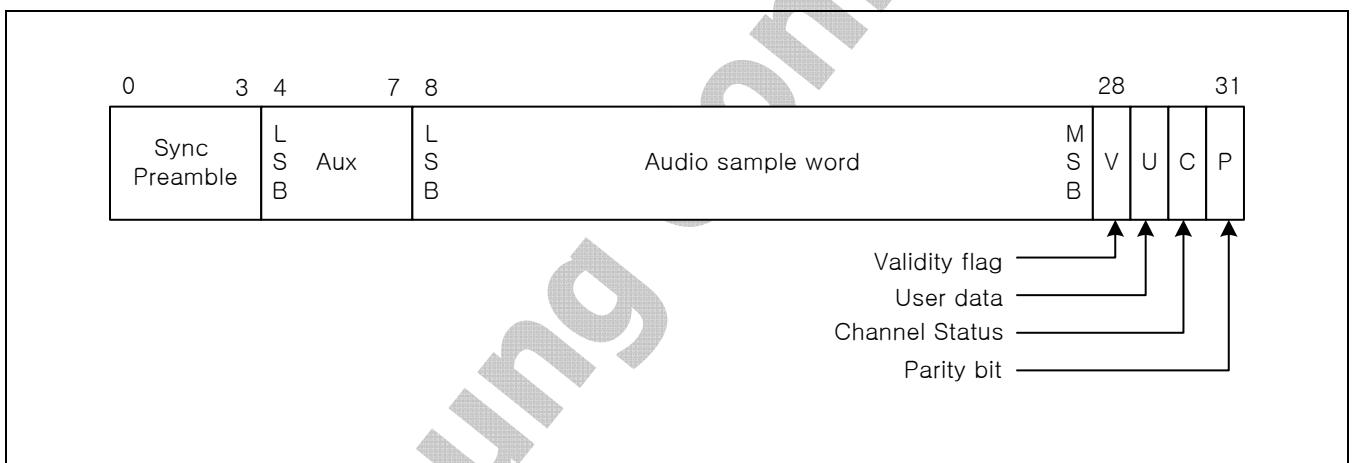


Figure 10-8 Sub-frame Format

10.2.1.2 Channel Status Block (IEC-60958-3)

Channel Status Block specifies the aggregation of Channel Status bit in each sub-frame, as shown in [Figure 10-9](#). As one frame consists of 192 frames, one channel status block can be obtained for one channel.

This block holds the information of the stream being transmitted such as application, stream type, sampling frequency, word length, and so on.

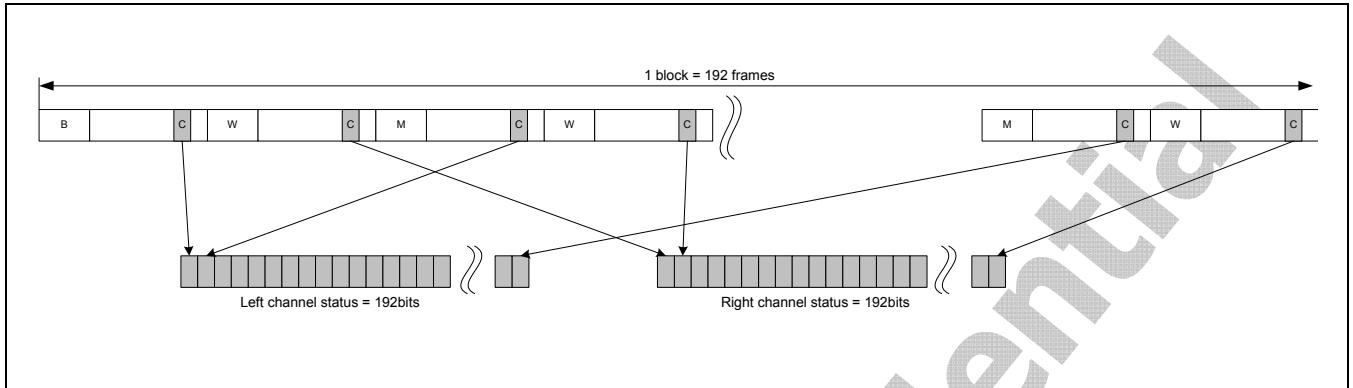


Figure 10-9 Channel Status Block Extract from SPDIF Stream

Byte								
0	a = "0"	b = "0"	c	d		Mode = "0 0"		
bit	0	1	2	3	4	5	6	7
Category code								
1	8	9	10	11	12	13	14	15
bit	16	17	18	19	20	21	22	23
Source number								
2	24	25	26	27	28	29	30	31
bit	Sampling frequency				Clock accuracy			
3	32	33	34	35	36	37	38	39
bit	40	41	42	43	44	45	46	47
5	48	49	50	51	52	53	54	55
bit	56	57	58	59	60	61	62	63
7	64	65	66	67	68	69	70	71
bit	72	73	74	75	76	77	78	79
10	80	81	82	83	84	85	86	87
bit	88	89	90	91	92	93	94	95
12	96	97	98	99	100	101	102	103
bit	104	105	106	107	108	109	110	111
14	112	113	114	115	116	117	118	119
bit	120	121	122	123	124	125	126	127
16	128	129	130	131	132	133	134	135
bit	136	137	138	139	140	141	142	143
18	144	145	146	147	148	149	150	151
bit	152	153	154	155	156	157	158	159
20	160	161	162	163	164	165	166	167
bit	168	169	170	171	172	173	174	175
22	176	177	178	179	180	181	182	183
bit	184	185	186	187	188	189	190	191
a: use of channel status block. b: linear PCM identification.				c: copyright information. d: additional format information.				

Figure 10-10 Channel Status Block

10.2.1.3 Channel Coding

Time slots from 4 to 31 are encoded in bi-phase mark to:

- Minimize the DC component on transmission line
- Facilitate clock recovery from the data stream
- Make the interface insensitive to polarity of connections

Each bit that needs to be transmitted is represented by a symbol comprising two consecutive binary states. The first state of a symbol is always different from the second state of previous symbol. The second state of the symbol is identical to the first if the bit to be transmitted is logical “0” and different from the first if the bit is logical “1”.

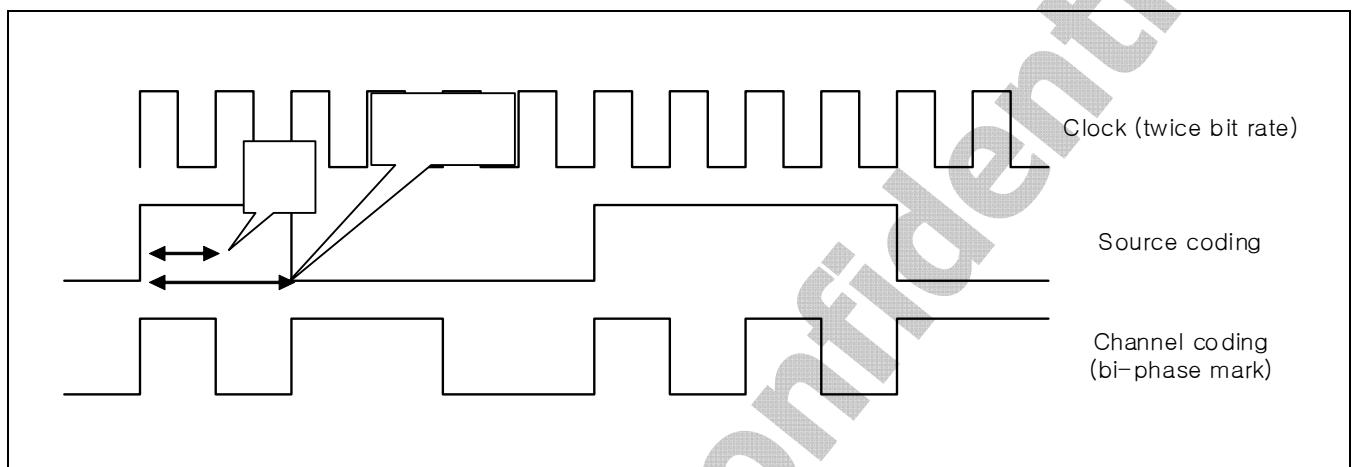


Figure 10-11 Channel coding

10.2.1.4 Preamble

Preambles are specific patterns providing synchronization and identification of the sub-frames and blocks. A set of three preambles is used. These preambles are transmitted in the time allocated to four time slots (time slots 0 to 3) and are represented by eight successive states. The first state of the preamble is always different from the second state of the previous symbol. Like bi-phase code, these preambles are DC independent and provide clock recovery. They differ in at least two states from any valid bi-phase sequence.

10.2.1.5 Non-Linear PCM Encoded Source (IEC 61937)

The non-linear PCM encoded audio bitstream is transferred using the basic 16-bit data area of the IEC 60958 subframes, that is, in time slots from 12 to 27. Each IEC 60958 frame transfers 32-bits of non-PCM data in consumer application mode.

When the SPDIF bitstream conveys linear PCM audio, the symbol frequency is 64 times the PCM sampling frequency (32 time slots per PCM sample times two channels).

When the interface transmits non-linear PCM encoded audio bitstream, the symbol frequency is 64 times the sampling rate of the encoded audio within that bitstream.

In case the interface containing audio with low sampling frequency conveys a non-linear PCM encoded audio bitstream, the symbol frequency is 128 times the sampling rate of the encoded audio within that bitstream.

Each data burst contains a burst-preamble consisting of four 16-bit words (Pa, Pb, Pc, and Pd); followed by the burst-payload that contains data of an encoded audio frame.

The burst-preamble consists of four mandatory fields. Pa and Pb represent a synchronization word; Pc gives information about the type of data and some information/ control for the receiver; and Pd gives the length of the burst-payload, limited to 216(=65,535) bits.

The four preamble words are contained in two sequential SPDIF frames. The frame beginning the data-burst contains preamble word Pa in sub-frame 1 and Pb in sub-frame 2. The next frame contains Pc in sub-frame 1 and Pd in sub-frame 2. When placed into a SPDIF sub-frame, the MSB of a 16-bit burst-preamble is placed into time slot 27 and the LSB is placed into time slot 12.

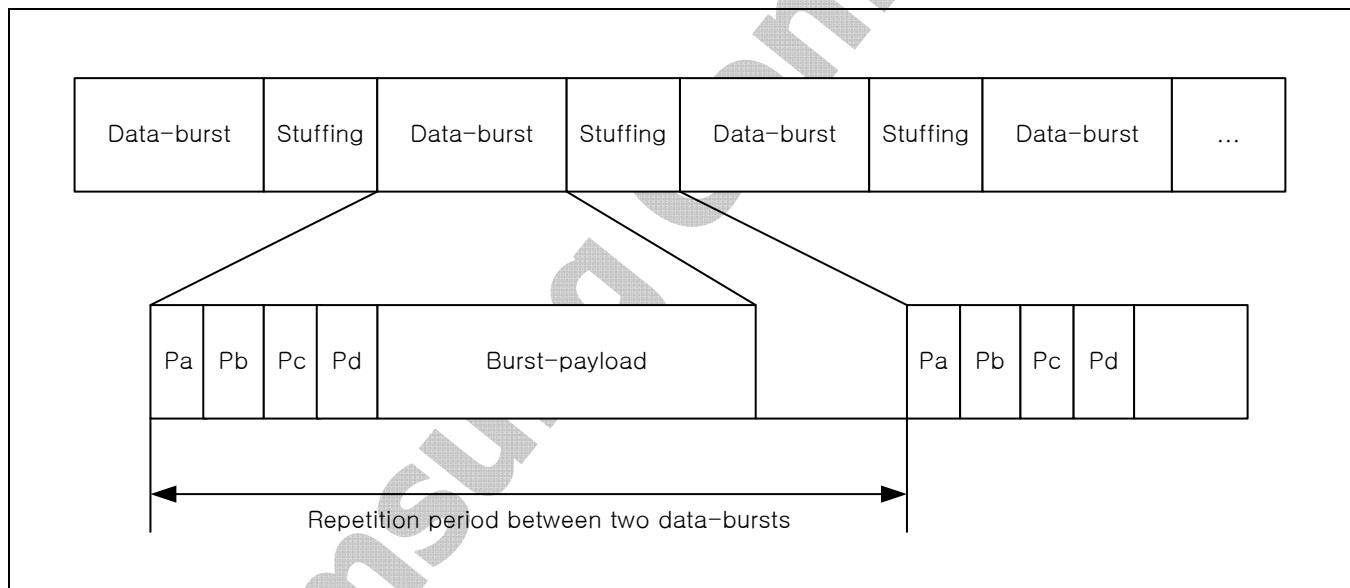


Figure 10-12 Non-linear PCM Format

10.3 REGISTERS DESCRIPTION

The register address map of HDMI 1.3 Tx Subsystem is divided into register address space of sub-modules, as shown in [Table 10-4](#).

Table 10-4 Register Address Map

Register Base	Address	Description
CTRL_BASE	0xFA10_0000	Specifies the controller register base address.
HDMI_CORE_BASE	0xFA11_0000	Specifies the HDMI register base address.
SPDIF_BASE	0xFA13_0000	Specifies the SPDIF receiver register base address.
I2S_BASE	0xFA14_0000	Specifies the I2S receiver register base address.
TG_BASE	0xFA15_0000	Specifies the HDMI timing generator register base address.
eFUSE_BASE	0xFA16_0000	Specifies the e-fuse related register base address.
CEC_BASE	0xE1B0_0000	Specifies the CEC register base address.

10.3.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
Control Registers				
INTC_CON	0xFA10_0000	R/W	Specifies the interrupt control register.	0x00
INTC_FLAG	0xFA10_0004	R/W	Specifies the interrupt flag register.	0x00
HDCP_KEY_LOAD	0xFA10_0008	R	Specifies the HDCP key status.	0x00
HPD_STATUS	0xFA10_000C	R	Specifies the value of HPD signal.	0x00
AUDIO_CLKSEL	0xFA10_0010	R/W	Selects the audio system clock.	0x00
PHY_RSTOUT	0xFA10_0014	R/W	Specifies the HDMI PHY reset out.	0x00
PHY_VPLL	0xFA10_0018	R	Specifies the HDMI PHY VPLL monitor.	0x00
PHY_CMU	0xFA10_001C	R	Specifies the HDMI PHY CMU monitor.	0x00
CORE_RSTOUT	0xFA10_0020	R/W	Specifies the HDMI TX core software reset.	0x01
HDMI Core Registers (Control Registers)				
HDMI_CON_0	0xFA11_0000	R/W	Specifies the HDMI system control register 0.	0x00
HDMI_CON_1	0xFA11_0004	R/W	Specifies the HDMI system control register 1.	0x00
HDMI_CON_2	0xFA11_0008	R/W	Specifies the HDMI system control register 2.	0x00
STATUS	0xFA11_0010	R/W	Specifies the HDMI system status register.	0x00
PHY_STATUS	0xFA11_0014	R	Specifies the PHY status register.	0x00
STATUS_EN	0xFA11_0020	R/W	Specifies the HDMI system status enable register.	0x00
HPD	0xFA11_0030	R/W	Specifies the HPD control register.	0x00
MODE_SEL	0xFA11_0040	R/W	Selects the HDMI/ DVI mode.	0x00
ENC_EN	0xFA11_0044	R/W	Specifies the HDCP encryption enable register.	0x00
HDMI Core Registers (Video Related Registers)				
BLUE_SCREEN_0	0xFA11_0050	R/W	Specifies the pixel values for blue screen.	0x00
BLUE_SCREEN_1	0xFA11_0054	R/W	Specifies the pixel values for blue screen.	0x00
BLUE_SCREEN_2	0xFA11_0058	R/W	Specifies the pixel values for blue screen.	0x00
HDMI_YMAX	0xFA11_0060	R/W	Specifies the maximum Y (or R, G, B) pixel value.	0xeb
HDMI_YMIN	0xFA11_0064	R/W	Specifies the minimum Y (or R, G, B) pixel value.	0x10
HDMI_CMAX	0xFA11_0068	R/W	Specifies the maximum Cb/ Cr pixel value.	0xf0
HDMI_CMIN	0xFA11_006C	R/W	Specifies the minimum Cb/ Cr pixel value.	0x10
H_BLANK_0	0xFA11_00A0	R/W	Specifies the horizontal blanking setting.	0x00
H_BLANK_1	0xFA11_00A4	R/W	Specifies the horizontal blanking setting.	0x00
V_BLANK_0	0xFA11_00B0	R/W	Specifies the vertical blanking setting.	0x00
V_BLANK_1	0xFA11_00B4	R/W	Specifies the vertical blanking setting.	0x00

Register	Address	R/W	Description	Reset Value
V_BLANK_2	0xFA11_00B8	R/W	Specifies the vertical blanking setting.	0x00
H_V_LINE_0	0xFA11_00C0	R/W	Specifies the horizontal and vertical line setting.	0x00
H_V_LINE_1	0xFA11_00C4	R/W	Specifies the horizontal and vertical line setting.	0x00
H_V_LINE_2	0xFA11_00C8	R/W	Specifies the horizontal and vertical line setting.	0x00
VSYNC_POL	0xFA11_00E4	R/W	Specifies the vertical sync polarity control register.	0x00
INT_PRO_MODE	0xFA11_00E8	R/W	Specifies the Interlace/ Progressive control register.	0x00
V_BLANK_F_0	0xFA11_0110	R/W	Specifies the vertical blanking setting for bottom field.	0x00
V_BLANK_F_1	0xFA11_0114	R/W	Specifies the vertical blanking setting for bottom field.	0x00
V_BLANK_F_2	0xFA11_0118	R/W	Specifies the vertical blanking setting for bottom field.	0x00
H_SYNC_GEN_0	0xFA11_0120	R/W	Specifies the horizontal sync generation setting.	0x00
H_SYNC_GEN_1	0xFA11_0124	R/W	Specifies the horizontal sync generation setting.	0x00
H_SYNC_GEN_2	0xFA11_0128	R/W	Specifies the horizontal sync generation setting.	0x00
V_SYNC_GEN1_0	0xFA11_0130	R/W	Specifies the vertical sync generation for top field or frame.	0x01
V_SYNC_GEN1_1	0xFA11_0134	R/W	Specifies the vertical sync generation for top field or frame.	0x10
V_SYNC_GEN1_2	0xFA11_0138	R/W	Specifies the vertical sync generation for top field or frame.	0x00
V_SYNC_GEN2_0	0xFA11_0140	R/W	Specifies the vertical sync generation for bottom field – vertical position.	0x01
V_SYNC_GEN2_1	0xFA11_0144	R/W	Specifies the vertical sync generation for bottom field – vertical position.	0x10
V_SYNC_GEN2_2	0xFA11_0148	R/W	Specifies the vertical sync generation for bottom field – vertical position.	0x00
V_SYNC_GEN3_0	0xFA11_0150	R/W	Specifies the vertical sync generation for bottom field – horizontal position.	0x01
V_SYNC_GEN3_1	0xFA11_0154	R/W	Specifies the vertical sync generation for bottom field – horizontal position.	0x10
V_SYNC_GEN3_2	0xFA11_0158	R/W	Specifies the vertical sync generation for bottom field – horizontal position.	0x00
HDMI Core Registers (Audio Related Registers)				

Register	Address	R/W	Description	Reset Value
ASP_CON	0xFA11_0160	R/W	Specifies the ASP packet control register.	0x00
ASP_SP_FLAT	0xFA11_0164	R/W	Specifies the ASP packet sp_flat bit control.	0x00
ASP_CHCFG0	0xFA11_0170	R/W	Specifies the ASP audio channel configuration.	0x08
ASP_CHCFG1	0xFA11_0174	R/W	Specifies the ASP audio channel configuration.	0x1a
ASP_CHCFG2	0xFA11_0178	R/W	Specifies the ASP audio channel configuration.	0x2c
ASP_CHCFG3	0xFA11_017C	R/W	Specifies the ASP audio channel configuration.	0x3e
ACR_CON	0xFA11_0180	R/W	Specifies the ACR packet control register.	0x00
ACR_MCTS0	0xFA11_0184	R/W	Specifies the measured CTS value.	0x01
ACR_MCTS1	0xFA11_0188	R/W	Specifies the measured CTS value.	0x00
ACR_MCTS2	0xFA11_018C	R/W	Specifies the measured CTS value.	0x00
ACR_CTS0	0xFA11_0190	R/W	Specifies the CTS value for fixed CTS transmission mode.	0xe8
ACR_CTS1	0xFA11_0194	R/W	Specifies the CTS value for fixed CTS transmission mode.	0x03
ACR_CTS2	0xFA11_0198	R/W	Specifies the CTS value for fixed CTS transmission mode.	0x00
ACR_N0	0xFA11_01A0	R/W	Specifies the N value for ACR packet.	0xe8
ACR_N1	0xFA11_01A4	R/W	Specifies the N value for ACR packet.	0x03
ACR_N2	0xFA11_01A8	R/W	Specifies the N value for ACR packet.	0x00
ACR LSB2	0xFA11_01B0	R/W	Specifies the alternate LSB for fixed CTS transmission mode.	0x00
ACR_TXCNT	0xFA11_01B4	R/W	Specifies the number of ACR packet transmission per frame.	0x1f
ACR_TXINTERVAL	0xFA11_01B8	R/W	Specifies the interval for ACR packet transmission.	0x63
ACR_CTS_OFFSET	0xFA11_01BC	R/W	Specifies the CTS offset for measured CTS mode.	0x00

HDMI Core Registers (Packet Related Registers)

GCP_CON	0xFA11_01C0	R/W	Specifies the ACR packet control register.	0x04
GCP_BYTE1	0xFA11_01D0	R/W	Specifies the GCP packet body.	0x00
GCP_BYTE2	0xFA11_01D4	R/W	Specifies the GCP packet body.	0x00
GCP_BYTE3	0xFA11_01D8	R/W	Specifies the GCP packet body.	0x00
ACP_CON	0xFA11_01E0	R/W	Specifies the ACP packet control register.	0x00
ACP_TYPE	0xFA11_01F0	R/W	Specifies the ACP packet header.	0x00

Register	Address	R/W	Description	Reset Value
ACP_DATA00~16	0xFA11_0200 ~ 0xFA11_0240	R/W	Specifies the ACP packet body.	0x00
ISRC_CON	0xFA11_0250	R/W	Specifies the ACR packet control register.	0x00
ISRC1_HEADER1	0xFA11_0264	R/W	Specifies the ISCR1 packet header.	0x00
ISRC1_DATA00~15	0xFA11_0270 ~ 0xFA11_02AC	R/W	Specifies the ISRC1 packet body.	0x00
ISRC2_DATA00~15	0xFA11_02B0 ~ 0xFA11_02EC	R/W	Specifies the ISRC2 packet body.	0x00
AVI_CON	0xFA11_0300	R/W	Specifies the AVI packet control register.	0x00
AVI_CHECK_SUM	0xFA11_0310	R/W	Specifies the AVI packet checksum.	0x00
AVI_BYTE01~13	0xFA11_0320 ~ 0xFA11_0350	R/W	Specifies the AVI packet body.	0x00
AUI_CON	0xFA11_0360	R/W	Specifies the AUI packet control register.	0x00
AUI_CHECK_SUM	0xFA11_0370	R/W	Specifies the AUI packet checksum.	0x00
AUI_BYTE1~5	0xFA11_0380 ~ 0xFA11_0390	R/W	Specifies the AUI packet body.	0x00
MPG_CON	0xFA11_03A0	R/W	Specifies the ACR packet control register.	0x00
MPG_CHECK_SUM	0xFA11_03B0	R/W	Specifies the MPG packet checksum.	0x00
MPG_BYTE1~5	0xFA11_03C0 ~ 0xFA11_03D0	R/W	Specifies the MPG packet body.	0x00
SPD_CON	0xFA11_0400	R/W	Specifies the SPD packet control register.	0x00
SPD_HEADER0	0xFA11_0410	R/W	Specifies the SPD packet header.	0x00
SPD_HEADER1	0xFA11_0414	R/W	Specifies the SPD packet header.	0x00
SPD_HEADER2	0xFA11_0418	R/W	Specifies the SPD packet header.	0x00
SPD_DATA00~27	0xFA11_0420 ~ 0xFA11_048C	R/W	Specifies the SPD packet body.	0x00
HDMI Core Registers (HDCP Related Register)				
HDCP_SHA1_00~19	0xFA11_0600 ~ 0xFA11_064C	R/W	Specifies the SHA-1 value from repeater.	0x00
HDCP_KSV_LIST_0 ~4	0xFA11_0650 ~ 0xFA11_0660	R/W	Specifies the KSV list from repeater.	0x00
HDCP_KSV_LIST_C ON	0xFA11_0664	R/W	Controls the KSV list.	0x01

Register	Address	R/W	Description	Reset Value
HDCP_SHA_RESULT	0xFA11_0670	R/W	Specifies the SHA-1 checking result register.	0x00
HDCP_CTRL1	0xFA11_0680	R/W	Specifies the HDCP control register1.	0x00
HDCP_CTRL2	0xFA11_0684	R/W	Specifies the HDCP control register2.	0x00
HDCP_CHECK_RESULT	0xFA11_0690	R/W	Checks the result of Ri and Pj values.	0x00
HDCP_BKSV_0~4	0xFA11_06A0 ~ 0xFA11_06B0	R/W	Specifies the KSV of Rx.	0x00
HDCP_AKSV_0~4	0xFA11_06C0 ~ 0xFA11_06D0	R/W	Specifies the KSV of Tx.	0x00
HDCP_An_0~7	0xFA11_06E0 ~ 0xFA11_06FC	R/W	Specifies the An value.	0x00
HDCP_BCAPS	0xFA11_0700	R/W	Specifies the BCAPS from Rx.	0x00
HDCP_BSTATUS_0	0xFA11_0710	R/W	Specifies the BSTATUS from Rx.	0x00
HDCP_BSTATUS_1	0xFA11_0714	R/W	Specifies the BSTATUS from Rx.	0x00
HDCP_Ri_0	0xFA11_0740	R/W	Specifies the Ri value of Tx.	0x00
HDCP_Ri_1	0xFA11_0744	R/W	Specifies the Ri value of Tx.	0x00
HDCP_I2C_INT	0xFA11_0780	R/W	Specifies the I2C interrupt flag.	0x00
HDCP_AN_INT	0xFA11_0790	R/W	Specifies the An value ready interrupt flag.	0x00
HDCP_WATCGDOG -INT	0xFA11_07A0	R/W	Specifies the Watchdog interrupt flag.	0x00
HDCP_Ri_INT	0xFA11_07B0	R/W	Specifies the Ri value update interrupt flag.	0x00
HDCP_Ri_Compare_0	0xFA11_07D0	R/W	Specifies the HDCP Ri interrupt frame number index register 0.	0x80
HDCP_Ri_Compare_1	0xFA11_07D4	R/W	Specifies the HDCP Ri interrupt frame number index register 1.	0x7f
HDCP_Frame_Count	0xFA11_07E0	R	Specifies the current value of frame count index in the hardware.	0x00
GAMUT_CON	0xFA11_0500	R/W	Specifies the GAMUT packet control register.	0x00
GAMUT_HEADER0	0xFA11_0504	R/W	Specifies the GAMUT packet header.	0x00
GAMUT_HEADER1	0xFA11_0508	R/W	Specifies the GAMUT packet header.	0x00
GAMUT_HEADER2	0xFA11_050C	R/W	Specifies the GAMUT packet header.	0x00
GAMUT_DATA00~27	0xFA11_0510 ~ 0xFA11_057C	R/W	Specifies the GAMUT packet body.	0x00
-	0xFA11_05C0	R/W	Reserved. Do not modify this.	0x00

Register	Address	R/W	Description	Reset Value
VIDEO_PATTERN_GEN	0xFA11_05C4	R/W	Specifies the video pattern generation register.	0x00
HPD_GEN	0xFA11_05C8	R/W	Specifies the HPD duration value register.	0x01
SPDIF Registers				
SPDIFIN_CLK_CTRL	0xFA13_0000	R/W	Specifies the SPDIFIN clock control register.	0x02
SPDIFIN_OP_CTRL	0xFA13_0004	R/W	Specifies the SPDIFIN operation control register 1.	0x00
SPDIFIN_IRQ_MASK	0xFA13_0008	R/W	Specifies the SPDIFIN interrupt request mask register.	0x00
SPDIFIN_IRQ_STATUS	0xFA13_000C	R/W	Specifies the SPDIFIN interrupt request status register.	0x00
SPDIFIN_CONFIG_1	0xFA13_0010	R/W	Specifies the SPDIFIN configuration register 1.	0x02
SPDIFIN_CONFIG_2	0xFA13_0014	R/W	Specifies the SPDIFIN configuration register 2.	0x00
-	0xFA13_0018	-	Reserved	-
-	0xFA13_001C	-	Reserved	-
SPDIFIN_USER_VALUE_1	0xFA13_0020	R/W	Specifies the SPDIFIN user value register 1.	0x00
SPDIFIN_USER_VALUE_2	0xFA13_0024	R/W	Specifies the SPDIFIN user value register 2.	0x00
SPDIFIN_USER_VALUE_3	0xFA13_0028	R/W	Specifies the SPDIFIN user value register 3.	0x00
SPDIFIN_USER_VALUE_4	0xFA13_002C	R/W	Specifies the SPDIFIN user value register 4.	0x00
SPDIFIN_CH_STAT_US_0_1	0xFA13_0030	R	Specifies the SPDIFIN channel status register 0-1.	0x00
SPDIFIN_CH_STAT_US_0_2	0xFA13_0034	R	Specifies the SPDIFIN channel status register 0-2.	0x00
SPDIFIN_CH_STAT_US_0_3	0xFA13_0038	R	Specifies the SPDIFIN channel status register 0-3.	0x00
SPDIFIN_CH_STAT_US_0_4	0xFA13_003C	R	Specifies the SPDIFIN channel status register 0-4.	0x00
SPDIFIN_CH_STAT_US_1	0xFA13_0040	R	Specifies the SPDIFIN channel status register 1.	0x00
-	0xFA13_0044	-	Reserved	-
SPDIFIN_FRAME_PERIOD_1	0xFA13_0048	R	Specifies the SPDIFIN frame period register 1.	0x00
SPDIFIN_FRAME_PERIOD_2	0xFA13_004C	R	Specifies the SPDIFIN frame period register 2.	0x00
SPDIFIN_Pc_INFO_1	0xFA13_0050	R	Specifies the SPDIFIN PC info register 1.	0x00

Register	Address	R/W	Description	Reset Value
SPDIFIN_Pc_INFO_2	0xFA13_0054	R	Specifies the SPDIFIN PC info register 2.	0x00
SPDIFIN_Pd_INFO_1	0xFA13_0058	R	Specifies the SPDIFIN PD info register 1.	0x00
SPDIFIN_Pd_INFO_2	0xFA13_005C	R	Specifies the SPDIFIN PD Info Register 2.	0x00
SPDIFIN_DATA_BU_F_0_1	0xFA13_0060	R	Specifies the SPDIFIN data buffer register 0_1.	0x00
SPDIFIN_DATA_BU_F_0_2	0xFA13_0064	R	Specifies the SPDIFIN data buffer register 0_2.	0x00
SPDIFIN_DATA_BU_F_0_3	0xFA13_0068	R	Specifies the SPDIFIN data buffer register 0_3.	0x00
SPDIFIN_USER_BU_F_0	0xFA13_006C	R	Specifies the SPDIFIN user buffer register 0.	0x00
SPDIFIN_DATA_BU_F_1_1	0xFA13_0070	R	Specifies the SPDIFIN data buffer register 1_1.	0x00
SPDIFIN_DATA_BU_F_1_2	0xFA13_0074	R	Specifies the SPDIFIN data buffer register 1_2.	0x00
SPDIFIN_DATA_BU_F_1_3	0xFA13_0078	R	Specifies the SPDIFIN data buffer register 1_3.	0x00
SPDIFIN_USER_BU_F_1	0xFA13_007C	R	Specifies the SPDIFIN user buffer register 1.	0x00
I2S Registers				
I2S_CLK_CON	0xFA14_0000	R/W	Specifies the I2S clock enable register.	0x00
I2S_CON_1	0xFA14_0004	R/W	Specifies the I2S control register 1.	0x00
I2S_CON_2	0xFA14_0008	R/W	Specifies the I2S control register 2.	0x16
I2S_PIN_SEL_0	0xFA14_000C	R/W	Specifies the I2S input pin selection register 0.	0x77
I2S_PIN_SEL_1	0xFA14_0010	R/W	Specifies the I2S input pin selection register 1.	0x77
I2S_PIN_SEL_2	0xFA14_0014	R/W	Specifies the I2S input pin selection register 2.	0x77
I2S_PIN_SEL_3	0xFA14_0018	R/W	Specifies the I2S input pin selection register 3.	0x07
I2S_DSD_CON	0xFA14_001C	R/W	Specifies the I2S DSD control register.	0x02
I2S_IN_MUX_CON	0xFA14_0020	R/W	Specifies the I2S In/ Mux control register.	0x60
I2S_CH_ST_CON	0xFA14_0024	R/W	Specifies the I2S channel status control register.	0x00
I2S_CH_ST_0	0xFA14_0028	R/W	Specifies the I2S channel status block 0.	0x00
I2S_CH_ST_1	0xFA14_002C	R/W	Specifies the I2S channel status block 1.	0x00
I2S_CH_ST_2	0xFA14_0030	R/W	Specifies the I2S channel status block 2.	0x00

Register	Address	R/W	Description	Reset Value
I2S_CH_ST_3	0xFA14_0034	R/W	Specifies the I2S channel status block 3.	0x00
I2S_CH_ST_4	0xFA14_0038	R/W	Specifies the I2S channel status block 4.	0x00
I2S_CH_ST_SH_0	0xFA14_003C	R	Specifies the I2S channel status block shadow register 0.	0x00
I2S_CH_ST_SH_1	0xFA14_0040	R	Specifies the I2S channel status block shadow register 1.	0x00
I2S_CH_ST_SH_2	0xFA14_0044	R	Specifies the I2S channel status block shadow register 2.	0x00
I2S_CH_ST_SH_3	0xFA14_0048	R	Specifies the I2S channel status block shadow register 3.	0x00
I2S_CH_ST_SH_4	0xFA14_004C	R	Specifies the I2S channel status block shadow register 4.	0x00
I2S_VD_DATA	0xFA14_0050	R/W	Specifies the I2S audio sample validity register.	0x00
I2S_MUX_CH	0xFA14_0054	R/W	Specifies the I2S channel enable register.	0x03
I2S_MUX_CUV	0xFA14_0058	R/W	Specifies the I2S CUV enable register.	0x03
I2S_IRQ_MASK	0xFA14_005C	R/W	Specifies the I2S interrupt request mask register.	0x00
I2S_IRQ_STATUS	0xFA14_0060	R/W	Specifies the I2S interrupt request status register.	0x00
I2S_CH0_L_0	0xFA14_0064	R	Specifies the I2S PCM output data register.	0x00
I2S_CH0_L_1	0xFA14_0068	R	Specifies the I2S PCM output data register.	0x00
I2S_CH0_L_2	0xFA14_006C	R	Specifies the I2S PCM output data register.	0x00
I2S_CH0_L_3	0xFA14_0070	R	Specifies the I2S PCM output data register.	0x00
I2S_CH0_R_0	0xFA14_0074	R	Specifies the I2S PCM output data register.	0x00
I2S_CH0_R_1	0xFA14_0078	R	Specifies the I2S PCM output data register.	0x00
I2S_CH0_R_2	0xFA14_007C	R	Specifies the I2S PCM output data register.	0x00
I2S_CH0_R_3	0xFA14_0080	R	Specifies the I2S PCM output data register.	0x00
I2S_CH1_L_0	0xFA14_0084	R	Specifies the I2S PCM output data register.	0x00
I2S_CH1_L_1	0xFA14_0088	R	Specifies the I2S PCM output data register.	0x00
I2S_CH1_L_2	0xFA14_008C	R	Specifies the I2S PCM output data register.	0x00
I2S_CH1_L_3	0xFA14_0090	R	Specifies the I2S PCM output data register.	0x00
I2S_CH1_R_0	0xFA14_0094	R	Specifies the I2S PCM output data register.	0x00
I2S_CH1_R_1	0xFA14_0098	R	Specifies the I2S PCM output data register.	0x00
I2S_CH1_R_2	0xFA14_009C	R	Specifies the I2S PCM output data register.	0x00
I2S_CH1_R_3	0xFA14_00A0	R	Specifies the I2S PCM output data register.	0x00
I2S_CH2_L_0	0xFA14_00A4	R	Specifies the I2S PCM output data register.	0x00
I2S_CH2_L_1	0xFA14_00A8	R	Specifies the I2S PCM output data register.	0x00
I2S_CH2_L_2	0xFA14_00AC	R	Specifies the I2S PCM output data register.	0x00

Register	Address	R/W	Description	Reset Value
I2S_CH2_L_3	0xFA14_00B0	R	Specifies the I2S PCM output data register.	0x00
I2S_CH2_R_0	0xFA14_00B4	R	Specifies the I2S PCM output data register.	0x00
I2S_CH2_R_1	0xFA14_00B8	R	Specifies the I2S PCM output data register.	0x00
I2S_CH2_R_2	0xFA14_00BC	R	Specifies the I2S PCM output data register.	0x00
I2S_Ch2_R_3	0xFA14_00C0	R	Specifies the I2S PCM output data register.	0x00
I2S_CH3_L_0	0xFA14_00C4	R	Specifies the I2S PCM output data register.	0x00
I2S_CH3_L_1	0xFA14_00C8	R	Specifies the I2S PCM output data register.	0x00
I2S_CH3_L_2	0xFA14_00CC	R	Specifies the I2S PCM output data register.	0x00
I2S_CH3_R_0	0xFA14_00D0	R	Specifies the I2S PCM output data register.	0x00
I2S_CH3_R_1	0xFA14_00D4	R	Specifies the I2S PCM output data register.	0x00
I2S_CH3_R_2	0xFA14_00D8	R	Specifies the I2S PCM output data register.	0x00
I2S_CUV_L_R	0xFA14_00DC	R	Specifies the I2S CUV output data register.	0x00

Timing Generator Registers (TG Configure/Status Registers)

TG_CMD	0xFA15_0000	R/W	Specifies the command register.	0x00
TG_H_FSZ_L	0xFA15_0018	R/W	Specifies the horizontal full size.	0x72
TG_H_FSZ_H	0xFA15_001C	R/W	Specifies the horizontal full size.	0x06
TG_HACT_ST_L	0xFA15_0020	R/W	Specifies the horizontal active start.	0x05
TG_HACT_ST_H	0xFA15_0024	R/W	Specifies the horizontal active start.	0x01
TG_HACT_SZ_L	0xFA15_0028	R/W	Specifies the horizontal active size.	0x00
TG_HACT_SZ_H	0xFA15_002C	R/W	Specifies the horizontal active size.	0x05
TG_V_FSZ_L	0xFA15_0030	R/W	Specifies the vertical full line size.	0xEE
TG_V_FSZ_H	0xFA15_0034	R/W	Specifies the vertical full line size.	0x02
TG_VSYNC_L	0xFA15_0038	R/W	Specifies the vertical sync position.	0x01
TG_VSYNC_H	0xFA15_003C	R/W	Specifies the vertical sync position.	0x00
TG_VSYNC2_L	0xFA15_0040	R/W	Specifies the vertical sync position for bottom field.	0x33
TG_VSYNC2_H	0xFA15_0044	R/W	Specifies the vertical sync position for bottom field.	0x02
TG_VACT_ST_L	0xFA15_0048	R/W	Specifies the vertical sync active start position.	0x1a
TG_VACT_ST_H	0xFA15_004C	R/W	Specifies the vertical sync active start position.	0x00
TG_VACT_SZ_L	0xFA15_0050	R/W	Specifies the vertical active size.	0xd0
TG_VACT_SZ_H	0xFA15_0054	R/W	Specifies the vertical active size.	0x02
TG_FIELD_CHG_L	0xFA15_0058	R/W	Specifies the HDMI field change position.	0x33
TG_FIELD_CHG_H	0xFA15_005C	R/W	Specifies the HDMI field change position.	0x02
TG_VACT_ST2_L	0xFA15_0060	R/W	Specifies the HDMI vertical active start position for bottom field.	0x48

Register	Address	R/W	Description	Reset Value
TG_VACT_ST2_H	0xFA15_0064	R/W	Specifies the HDMI vertical active start position for bottom field.	0x02
TG_VSYNC_TOP_H_DMI_L	0xFA15_0078	R/W	Specifies the HDMI VSYNC position for top field.	0x01
TG_VSYNC_TOP_H_DMI_H	0xFA15_007C	R/W	Specifies the HDMI VSYNC position for top field.	0x00
TG_VSYNC_BOT_H_DMI_L	0xFA15_0080	R/W	Specifies the HDMI VSYNC position for bottom field.	0x01
TG_VSYNC_BOT_H_DMI_H	0xFA15_0084	R/W	Specifies the HDMI VSYNC position for bottom field.	0x00
TG_FIELD_TOP_HDMI_L	0xFA15_0088	R/W	Specifies the HDMI top field start position.	0x01
TG_FIELD_TOP_HDMI_H	0xFA15_008C	R/W	Specifies the HDMI top field start position.	0x00
TG_FIELD_BOT_HDMI_L	0xFA15_0090	R/W	Specifies the HDMI bottom field start position.	0X33
TG_FIELD_BOT_HDMI_H	0xFA15_0094	R/W	Specifies the HDMI bottom field start position.	0x02
MHL_HSYNC_WIDTH	0xFA15_017C	R/W	Specifies the HSYNC width configuration.	0x0F
MHL_VSYNC_WIDTH	0xFA15_0180	R/W	Specifies the VSYNC width configuration.	0x01
MHL_CLK_INV	0xFA15_0184	R/W	Specifies the MHL clock-out inversion.	0x00

10.3.2 CONTROL REGISTER

10.3.2.1 Control Register (INTC_CON, R/W, Address = 0xFA10_0000)

INTC_CON	Bit	Description	Initial State
IntrPol	[7]	Specifies the interrupt polarity. 0 = Active high 1 = Active low	0
IntrEnGlobal	[6]	0 = Disables all interrupts 1 = Enables or disables interrupts by INTC_CON5:0]	0
IntrEnI2S	[5]	Enables I2S interrupt. 0 = Disables 1 = Enables	0
IntrEnCEC	[4]	Enables CEC interrupt. 0 = Disables 1 = Enables	0
IntrEnHPDplug	[3]	Enables HPD plugged interrupt. 0 = Disables 1 = Enables	0
IntrEnHPDunplug	[2]	Enables HPD unplugged interrupt. 0 = Disables 1 = Enables	0
IntrEnSPDIF	[1]	Enables SPDIF interrupt. 0 = Disables 1 = Enables	0
IntrEnHDCP	[0]	Enables HDCP interrupt. 0 = Disables 1 = Enables	0

10.3.2.2 Control Register (INTC_FLAG, R/W, Address = 0xFA10_0004)

INTC_FLAG	Bit	Description	Initial State
-	[7:6]	Reserved	2b00
IntrI2S	[5]	Specifies the I2S interrupt flag (read only). 0 = Interrupt does not occur 1 = Interrupt occurs	0
IntrCEC	[4]	Specifies the CEC interrupt flag (read only). 0 = Interrupt does not occur 1 = Interrupt occurs	0
IntrHPDplug	[3]	Specifies the HPD plugged interrupt flag. If it is written by 1, it is cleared. 0 = Interrupt does not occur 1 = HPD plugged interrupt occurs	0
IntrHPDunplug	[2]	Specifies the HPD unplugged interrupt flag. If it is written by 1, it is cleared. 0 = Interrupt does not occur 1 = HPD unplugged interrupt occurs	0
IntrSPDIF	[1]	Specifies the SPDIF interrupt flag (read only). 0 = Interrupt does not occur 1 = Interrupt occurs	0
IntrHDCP	[0]	Specifies the HDCP interrupt flag (read only). 0 = Interrupt does not occur 1 = Interrupt occurs	0

10.3.2.3 Control Register (HDCP_KEY_LOAD_DONE, R, Address = 0xFA10_0008)

HDCP_KEY_LOAD_DONE	Bit	Description	Initial State
-	[7:1]	Reserved	7b0000000
HDCP_KEY_LOAD_DONE	[0]	Loads the HDCP key from e-fuse. 0 = Not available 1 = Completes loading HDCP key from e-fuse	0

10.3.2.4 Control Register (HPD_STATUS, R, Address = 0xFA10_000C)

HPD_STATUS	Bit	Description	Initial State
-	[7:1]	Reserved	0x00
HPD_Value	[0]	Specifies the value of HPD signal. 0 = Unplugged 1 = Plugged	0

10.3.2.5 Control Register (AUDIO_CLKSEL, R/W, Address = 0xFA10_0010)

AUDIO_CLKSEL	Bit	Description	Initial State
-	[7:1]	Reserved	0x00
AUDIO_CLK	[0]	<p>Specifies the clock selection of Audio system (Must be higher than 512*fs).</p> <p>0 = PCLK 1 = SPDIF clock</p> <p>Note: For audio data capture, the frequency of audio clock is higher than 512*fs[Hz]. If the frequency of audio clock is less than 512fs[Hz], audio data may be missed. Thus, if the frequency of PCLK is below 512fs[Hz], select SPDIF clock after it makes the frequency of SPDIF clock be higher than 512fs[Hz].</p>	0

10.3.2.6 Control Register (HDMI_PHY_RSTOUT, R/W, Address = 0xFA10_0014)

PHY_RSTOUT	Bit	Description	Initial State
-	[7:1]	Reserved	0x00
RSTOUT	[0]	<p>Specifies the HDMI PHY Software Reset out (active high).</p> <p>0 = Normal 1 = Reset</p>	0

10.3.2.7 Control Register (HDMI_PHY_VPLL, R, Address = 0xFA10_0018)

PHY_VPLL	Bit	Description	Initial State
VPLL_LOCK	[7]	Specifies the HDMI PHY VPLL Locking.	0x0
-	[6:4]	Reserved	0x0
VPLL_CODE	[3:0]	Specifies the HDMI PHY VPLL Code.	0x0

10.3.2.8 Control Register (HDMI_PHY_CMU, R, Address = 0xFA10_001C)

PHY_CMU	Bit	Description	Initial State
CMU_LOCK	[7]	Specifies the HDMI PHY CMU Locking.	0x0
-	[6:4]	Reserved	0x0
CMU_CODE	[3:0]	Specifies the HDMI PHY CMU Code.	0x0

10.3.2.9 Control Register (HDMI_CORE_RSTOUT, R/W, Address = 0xFA10_0020)

CORE_RSTOUT	Bit	Description	Initial State
-	[7:1]	Reserved	0x00
RSTOUT	[0]	Specifies the HDMI TX core software reset out (active low). 1 = Normal 0 = Reset	0x1

10.3.3 HDMI CORE REGISTER

10.3.3.1 Control Registers (HDMI_CON_0, R/W, Address = 0xFA11_0000)

HDMI_CON_0	Bit	Description	Initial State
MHL_CLK_En	[7]	Controls the MHL interface clock. 0 = Disables 1 = Enables	0
MHL_En	[6]	Enables the MHL interface. 0 = Disables 1 = Enables	0
Blue_Scr_En	[5]	Enables blue screen mode. When set, the input video pixels are discarded and blue screen register values are transmitted for all video data period. 0 = Disables 1 = Enables	0
Encoding_Option	[4]	Specifies the 10-bit TMDS encoding bit order option. 0 = Reverses the bit order during 10-bit encoding (to be set to 1 when connecting to TMDS PHY 1.3) 1 = Retains the bit order as is	0
-	[3]	Reserved	0
Asp_E	[2]	Generates audio sample packet. This bit is only valid when SYSTEM_EN is set. 0 = Discards audio sample 1 = Generates audio sample packet after receiving the audio sample	0
-	[1]	Reserved	0
System_En	[0]	Enables HDMI system. 0 = No op 1 = Enables HDMI	0

10.3.3.2 Control Registers (HDMI_CON_1, R/W, Address = 0xFA11_0004)

HDMI_CON_1	Bit	Description	Initial State
-	[7]	Reserved	0
Pxl_Lmt_Ctrl	[6:5]	<p>Controls the pixel value limitation. 2b00 = By-pass (Does not limit the pixel value) 2b01 = RGB mode</p> <p>Every channel's video input pixels are limited based on YMAX and YMIN register values. 2b10 = YCbCr mode</p> <p>The value of I_VIDEO_G is limited based on YMAX and YMIN register values.</p> <p>The values of I_VIDEO_B and I_VIDEO_R are limited based on CMAX and CMIN register values. 2b11 = Reserved</p>	3b00
-	[4:2]	Reserved	3b000
-	[1:0]	Reserved	0

10.3.3.3 Control Registers (HDMI_CON_2, R/W, Address = 0xFA11_0008)

HDMI_CON_2	Bit	Description	Initial State
-	[7:6]	Reserved	3b00
Vid_Period_En	[5]	<p>Controls the video preamble. 0 = Video preamble is applied (HDMI mode) 1 = Video preamble is not applied (DVI mode)</p>	0
-	[4:2]	Reserved	3b000
Dvi_Band_En	[1]	<p>In DVI mode, the leading guard band is not used. 0 = Guard band is applied (HDMI mode) 1 = Guard band is not applied (DVI mode)</p>	0
-	[0]	Reserved	0

10.3.3.4 Control Registers (STATUS, S/W, Address = 0xFA11_0010)

STATUS	Bit	Description	Initial State
Authen_Ack	[7]	<p>When HDCP is authenticated, this read-only bit occurs. It keeps the authentication signal without interruption. It is not cleared at all.</p> <p>This bit specifies just one delayed signal of authen_ack from HDCP block. It is not an interrupt source.</p> <p>0 = Not authenticated 1 = Authenticated</p>	0
Aud_Fifo_Ovf	[6]	<p>If audio FIFO overflows, this bit is set. Once set, it should be cleared by the host.</p> <p>0 = Not full 1 = Full</p>	0
	[5]	Reserved	0
Update_Ri_Int	[4]	<p>Specifies the Ri interrupt status bit. If it is written by 1, it is cleared.</p> <p>0 = Interrupt does not occur 1 = Interrupt occurs</p>	0
-	[3]	Reserved	0
An_Write_Int	[2]	<p>Indicates that {An} random value is ready. If it is written by 1, it is cleared.</p> <p>0 = Interrupt does not occur 1 = Interrupt occurs</p>	0
Watchdog_Int	[1]	<p>Indicates that the 2nd part of HDCP authentication protocol is initiated, and CPU should set a watchdog timer to check 5 seconds interval.</p> <p>If it is written by 1, it is cleared.</p> <p>0 = Interrupt does not occur 1 = Interrupt occurs</p>	0
I2c_Init_Int	[0]	<p>Indicates that the 1st part of HDCP authentication protocol can start.</p> <p>If it is written by 1, it is cleared.</p> <p>0 = Interrupt does not occur 1 = Interrupt occurs</p>	0

10.3.3.5 Control Registers (PHY_STATUS, R, Address = 0xFA11_0014)

PHY_STATUS	Bit	Description	Initial State
-	[7:1]	Reserved	7b0000000
Phy_Ready	[0]	Indicates that PHY is ready to receive the HDMI signals from link. 0 = Not Ready 1 = Ready	0

10.3.3.6 Control Registers (STATUS_EN, R/W, Address = 0xFA11_0020)

STATUS_EN	Bit	Description	Initial State
-	[7]	Reserved	0
Aud_Fido_Ovf_Ee	[6]	Enables audio buffer overflow interrupt. If it is set to '1', interrupt assertion is written on status registers. 0 = Disables 1 = Enables	0
-	[5]	Reserved	0
Update_Ri_Int_En	[4]	Enables UPDATE_RI_INT interrupt. 0 = Disables 1 = Enables	0
-	[3]	Reserved	0
An_Write_Int_En	[2]	Enables AN_WRITE_INT interrupt. 0 = Disables 1 = Enables	0
Watchdog_Int_En	[1]	Enables WATCHDOG_INT interrupt. 0 = Disables 1 = Enables	0
I2c_Int_En	[0]	Enables I2C_INT interrupt. 0 = Disables 1 = Enables	0

10.3.3.7 Control Registers (HPD, R/W, Address = 0xFA11_0030)

HPD	Bit	Description	Initial State
-	[7:2]	Reserved	6b000000
Sw_Hpd	[1]	If HPD_SEL bit is set, this SW_HPD signal is used for HPD (HDMI/ DVI cable plugging). However, if this bit is set to low during HDMI transmission, status machines in HDCP core are reset. Note that other HDCP register values are not influenced. 0 = Low (unplugged) 1 = High (plugged)	0
Hpd_Sel	[0]	If this bit is cleared, the I_HPD signal from the I/O port is used for HPD. If set, the SW_HPD signal is used for HPD. 0 = HPD signal 1 = SW_HPD internal HPD signal	0

NOTE: If ENC_EN (0xFA11_0044) is disabled (not using HDCP), HPD must be controlled by S/W. If you don't use S/W control, it is possible that HDMI core works abnormally.

10.3.3.8 Control Registers (MODE_SEL, R/W, Address = 0xFA11_0040)

MODE_SEL	Bit	Description	Initial State
-	[7:2]	Reserved	6b000000
Hdmi_Mode	[1]	Selects a mode. 0 = Disables 1 = Enables	0
Dvi_Mode	[0]	Selects a mode. 0 = Disables 1 = Enables	0

* DVI mode gets a higher priority than HDMI.

10.3.3.9 Control Registers (ENC_EN, R/W, Address = 0xFA11_0044)

ENC_EN	Bit	Description	Initial State
-	[7:1]	Reserved	7b0000000
Hdcp_Enc_En	[0]	If this bit is set, the HDCP encryption is applied. Before setting this bit, the HDCP authentication process has to be completed. 0 = Encryption disables 1 = Encryption enables	0

10.3.3.10 Video Related Register (BLUESCREEN_0/1/2)

- BLUESCREEN_0, R/W, Address = 0xFA11_0050
- BLUE_SCREEN_1, R/W, Address = 0xFA11_0054
- BLUE_SCREEN_2, R/W, Address = 0xFA11_0058

BLUESCREEN_0/1/2	Bit	Description	Initial State
BLUESCREEN_0	[7:0]	Specifies the Channel 0 color setting (Cb or B).	0x0
BLUESCREEN_1	[7:0]	Specifies the Channel 1 color setting (Y or G).	0x0
BLUESCREEN_2	[7:0]	Specifies the Channel 2 color setting (Cr or R).	0x0

10.3.3.11 Video Related Register (HDMI_YMAX / HDMI_YMIN / HDMI_CMAX / HDMI_CMIN)

- HDMI_YMAX, R/W, Address = 0xFA11_0060
- HDMI_YMIN, R/W, Address = 0xFA11_0064
- HDMI_CMAX, R/W, Address = 0xFA11_0068
- HDMI_CMIN, R/W, Address = 0xFA11_006C

(HDMI_YMAX / HDMI_YMIN / HDMI_CMAX / HDMI_CMIN	Bit	Description	Initial State
HDMI_YMAX	[7:0]	These registers are used based upon the PX_LMT_CTRL bits in HDMI_CON_1 register.	0xEB
HDMI_YMIN	[7:0]		0x10
HDMI_CMAX	[7:0]	For RGB mode if (i_video_x > HDMI_YMAX x 16) output = HDMI_YMAX x 16 else if (i_video_x < HDMI_YMIN x 16) output = HDMI_YMIN x 16 else output = i_video_x	0xF0
HDMI_CMIN	[7:0]	For YCbCr mode, the Y input is dealt in a similar way as shown above. For Cb and Cr values, if (i_video_x > HDMI_CMAX x 16) output = HDMI_CMAX x 16 else if (i_video_x < HDMI_CMIN x 16) output = HDMI_CMIN x 16 else output = i_video_x	0x10
		Note: The value 16 in each line compensates the difference of bit width between the input pixel and register value.	

10.3.3.12 Video Related Register (H_BLANK_0/1)

- H_BLANK_0, R/W, Address = 0xFA11_00A0
- H_BLANK_1, R/W, Address = 0xFA11_00A4

H_BLANK_0/1	Bit	Description	Initial State
-	[15:10]	Reserved	6b000000
H_BLANK	[9:0]	Specifies the clock cycles of horizontal blanking size. For more details on H_BLANK, refer to "Reference CEA-861D".	0x000

60Hz	720x480p	1280x720p	1920x1080i	1920x1080p
H_BLANK	138(8Ah)	370(172h)	280(118h)	280(118h)
50Hz	720x576p	1280x720p	1920x1080i	1920x1080p
H_BLANK	144(90h)	700(2bch)	720(2d0h)	720(2d0h)

NOTE: 1080p is 25/29.97/30Hz.

10.3.3.13 Video Related Register (V_BLANK_0/1/2)

- V_BLANK_0, R/W, Address = 0xFA11_00B0
- V_BLANK_1, R/W, Address = 0xFA11_00B4
- V_BLANK_2, R/W, Address = 0xFA11_00B8

V_BLANK_0/1/2	Bit	Description	Initial State
-	[23:22]	Reserved	0x0
V1_BLANK	[21:11]	Specifies the vertical blanking line size (front part). For more details on V1_BLANK, refer to "Reference CEA-861D".	0x000
V2_BLANK	[10:0]	Specifies V1_BLANK+Active Lines (end part). This value is the same as V_LINE value for progressive mode. For interlace mode, use the reference value as mentioned in the table below. For more details on V2_BLANK, refer to "CEA-861D".	0x000

60Hz	720x480p	1280x720p	1920x1080i	1920x1080p
V2_BLANK	525(d)	750(d)	562(d)	1125(d)
V1_BLANK	45(d)	30(d)	22(d)	45(d)
V_BLANK	16a0d(h)	f2ee(h)	b232(h)	1_6c65(h)
50Hz	720x576p	1280x720p	1920x1080i	1920x1080p
V2_BLANK	625(d)	750(d)	562(d)	1125(d)
V1_BLANK	49(d)	30(d)	22(d)	45(d)
V_BLANK	18a71(h)	F2ee(h)	B232(h)	1_6c65(h)

10.3.3.14 Video Related Register (H_V_LINE_0/1/2)

- H_V_LINE_0, R/W, Address = 0xFA11_00C0
- H_V_LINE_1, R/W, Address = 0xFA11_00C4
- H_V_LINE_2, R/W, Address = 0xFA11_00C8

H_V_LINE_0/1/2	Bit	Description	Initial State
H_LINE	[23:12]	Specifies the horizontal line length. For more details on H_LINE, refer to “Reference CEA-861D”.	0x000
V_LINE	[11:0]	Specifies the vertical line length. For more details on V_LINE, refer to “Reference CEA-861D”.	0x000

60Hz	720x480p	1280x720p	1920x1080i	1920x1080p
V_LINE	525(d)	750(d)	1125(d)	1125(d)
H_LINE	858(d)	1650(d)	2200(d)	2200(d)
H_V_LINE	35a20d(h)	6722ee(h)	898465(h)	898465(h)
50Hz	720x576p	1280x720p	1920x1080i	1920x1080p
V_LINE	625(d)	750(d)	1125(d)	1125(d)
H_LINE	864(d)	1980(d)	2640(d)	2640(d)
H_V_LINE	360271(h)	7bc2ee(h)	a50465(h)	a50465(h)

10.3.3.15 Video Related Register (VSYNC_POL, R/W, Address = 0xFA11_00E4)

VSYNC_POL	Bit	Description	Initial State
-	[7:1]	Reserved	0x00
V_Sync_Pol_Sel	[0]	Specifies the start point detection polarity selection bit. The sync shapes for 720p or 1080i are different from 480p and 576p. They have inverted shapes. 0 = Active high 1 = Active low	0

50/ 60 Hz	720x480p	720x576p	1280x720p	1920x1080i	1920x1080p
VSYNC_POL	1	1	0	0	0

10.3.3.16 Video Related Register (INT_PRO_MODE, R/W, Address = 0xFA11_00E8)

INT_PRO_MODE	Bit	Description	Initial State
-	[7:1]	Reserved	7b0000000
INT_PRO_MODE	[0]	Selects the interlaced or progressive mode. For more details on INT_PRO_MODE, refer to "Reference CEA-861D". 0 = progressive 1 = interlaced	0

10.3.3.17 Video Related Register (V_BLANK_F_0/1/2)

- V_BLANK_F_0, R/W, Address = 0xFA11_0110
- V_BLANK_F_1, R/W, Address = 0xFA11_0114
- V_BLANK_F_2, R/W, Address = 0xFA11_0118

V_BLANK_F_0/1/2	Bit	Description	Initial State
-	[23:22]	Reserved	0x0
V_BOT_END	[21:11]	In the interlace mode, v_blank length of even and odd field is different. This register specifies the end position of bottom field's active region. For more details on V_BOT_END, refer to "Reference CEA-861D".	0x000
V_BOT_ST	[10:0]	Specifies the start position of bottom field's active region. This value is the same as V_LINE value for interlace mode. For progressive mode, this value is not used. For more details on V_BOT_ST, refer to "Reference CEA-861D".	0x000

* The above register only affects the interlace mode.

50/ 60 Hz	720x480p	720x576p	1280x720p	1920x1080i	1920x1080p
V_BOT_ST	Don't care	Don't care	Don't care	585(d) 1125(d) 232a49(h)	Don't care
V_BOT_END					
V_BLANK_F					

10.3.3.18 Video Related Register (H_SYNC_GEN_0/1/2)

- H_SYNC_GEN_0, R/W, Address = 0xFA11_0120
- H_SYNC_GEN_1, R/W, Address = 0xFA11_0124
- H_SYNC_GEN_2, R/W, Address = 0xFA11_0128

H_SYNC_GEN_0/1/2	Bit	Description	Initial State
-	[23:21]	Reserved	0x0
Hsync_Pol	[20]	Inverts the generated signal to meet the modes. In 720p and 1080i modes, you don't need to invert the signal. Other modes need to be inverted. For more details on Hsync_Pol, refer to "Reference CEA-861D". 0 = Active high 1 = Active low	0
Hsync_Edn	[19:10]	Sets the end point of H sync. For more details on Hsync_Edn, refer to "Reference CEA-861D".	0x000
Hsync_Start	[9:0]	Sets the start point of H sync. For more details on Hsync_Start, refer to "Reference CEA-861D".	0x000

60Hz	720x480p	1280x720p	1920x1080i	1920x1080p
HSYNC_START	14(d)	108(d)	86(d)	86(d)
HSYNC_END	76(d)	148(d)	130(d)	130(d)
HSYNC_POL	1	0	0	0
H_SYNC_GEN	11300e(h)	2506c(h)	20856(h)	20856(h)
50Hz	720x576p	1280x720p	1920x1080i	1920x1080p
HSYNC_START	10(d)	438(d)	526(d)	526(d)
HSYNC_END	74(d)	478(d)	570(d)	570(d)
HSYNC_POL	1	0	0	0
H_SYNC_GEN	11280a(h)	779b6(h)	8ea0e(h)	8ea0e(h)

10.3.3.19 Video Related Register (V_SYNC_GEN1_0/1/2)

Progressive mode only has one v_sync, whereas interlace mode has two. This register is used for generating first v_sync in both cases.

- V_SYNC_GEN1_0, R/W, Address = 0xFA11_0130
- V_SYNC_GEN1_1, R/W, Address = 0xFA11_0134
- V_SYNC_GEN1_2, R/W, Address = 0xFA11_0138

V_SYNC_GEN1_0/1/2	Bit	Description	Initial State
Vsync_T_St	[23:12]	Specifies the top field (or frame) V sync start line number. For more details on Vsync_T_St, refer to "Reference CEA-861D".	0x001
Vsync_T_End	[11:0]	Specifies the top field (or frame) V sync end line number. For more details on Vsync_T_End, refer to "Reference CEA-861D".	0x001

50/ 60 Hz	720x480p	720x576p	1280x720p	1920x1080i	1920x1080p
VSYNC_T_END	15(d)	10(d)	10(d)	7(d)	9(d)
VSYNC_T_ST	9(d)	5(d)	5(d)	2(d)	4(d)
V_SYNC_GEN1	900f(h)	500a(h)	500a(h)	2007(h)	4009(h)

10.3.3.20 Video Related Register (V_SYNC_GEN2_0/1/2)

Progressive mode only has one v_sync, whereas interlace mode has two. This register is used for generating second v_sync of interlace case.

- V_SYNC_GEN2_0, R/W, Address = 0xFA11_0140
- V_SYNC_GEN2_1, R/W, Address = 0xFA11_0144
- V_SYNC_GEN2_2, R/W, Address = 0xFA11_0148

V_SYNC_GEN2_0/1/2	Bit	Description	Initial State
Vsync_B_St	[23:12]	Specifies the bottom field V sync start line number. For more details on Vsync_B_St, refer to "Reference CEA-861D".	0x001
Vsync_B_End	[11:0]	Specifies the bottom field V sync end line number. For more details on Vsync_B_End, refer to "Reference CEA-861D".	0x001

50/ 60 Hz	1920x1080i	Other cases
VSYNC_B_END	569(d)	Don't care
VSYNC_B_ST	564(d)	
V_SYNC_GEN2	234239(h)	

10.3.3.21 Video Related Register (V_SYNC_GEN3_0/1/2)

Progressive mode has only one v_sync, whereas interlace mode has two. This register is used for generating second v_sync of interlace case.

- V_SYNC_GEN3_0, R/W, Address = 0xFA11_0150
- V_SYNC_GEN3_1, R/W, Address = 0xFA11_0154
- V_SYNC_GEN3_2, R/W, Address = 0xFA11_0158

V_SYNC_GEN3_0/1/2	Bit	Description	Initial State
Vsync_H_Pos_St	[23:12]	Specifies the bottom field V sync start transition point. For more details on Vsync_H_Pos_St, refer to “Reference CEA-861D”.	0x001
Vsync_H_Pos_End	[11:0]	Specifies the bottom field V sync end transition point. For more details on Vsync_H_Pos_End, refer to “Reference CEA-861D”.	0x001

60 Hz	1920x1080i	Other cases
VSYNC_H_POS_ST VSYNC_H_POS_END V_SYNC_GEN3	1188(d) 1188(d) 4A44A4(h)	Don't care
50 Hz	1920x1080i	Other cases
VSYNC_H_POS_ST VSYNC_H_POS_END V_SYNC_GEN3	1848(d) 1848(d) 738738(h)	Don't care

10.3.3.22 Audio Related Packet Register (ASP_CON, R/W, Address = 0xFA11_0160)

ASP_CON	Bit	Description	Initial State
DST_Double	[7]	Specifies the DST double.	0
Aud_Type	[6:5]	Specifies the packet type instead of audio type. 00 = Audio Sample Packet 01 = One-bit audio packet 10 = HBR packet 11 = DST packet	2b00
Aud_Mode	[4]	Selects the two channel or multi-channel mode. This bit is also used for layout bit in ASP header. 0 = Two channel mode 1 = Multi-channel mode Set this bit to transmit HBR packets.	0
SP_Pre	[3:0]	Controls sub-packet usage for multi-channel mode only. When using two channel mode, this register value is not used. [0]: AUDIO0 control (0: disable, 1: enable) [1]: AUDIO1 control (0: disable, 1: enable) [2]: AUDIO2 control (0: disable, 1: enable) [3]: AUDIO3 control (0: disable, 1: enable)	4b0000

10.3.3.23 Audio Related Packet Register (ASP_SP_FLAT, R/W, Address = 0xFA11_0164)

ASP_SP_FLAT	Bit	Description	Initial State
-	[7:4]	Reserved	4b0000
SP_Flat	[3:0]	Specifies the sp_flat or sample_invalid value for ASP header. For more information on SP_Flat, refer to the HDMI specification v1.3 (5.3.4 and 5.3.9).	0x0

10.3.3.24 Audio Related Packet Register (ASP_CHCFG0/1/2/3)

- ASP_CHCFG0, R/W, Address = 0xFA11_0170
- ASP_CHCFG1, R/W, Address = 0xFA11_0174
- ASP_CHCFG2, R/W, Address = 0xFA11_0178
- ASP_CHCFG3, R/W, Address = 0xFA11_017C

ASP_CHCFG0/1/2/3	Bit	Description	Initial State
-	[31:30]	Reserved	2b00
Spk3R_Sel	[29:27]	Selects the audio channel for subpacket 3 right channel data in multi-channel mode. 000 = i_pcm0L is used for sub packet 3 left channel 001 = i_pcm0R is used for sub packet 3 right channel 010 = i_pcm1L is used for sub packet 3 left channel 011 = i_pcm1R is used for sub packet 3 right channel 100 = i_pcm2L is used for sub packet 3 left channel 101 = i_pcm2R is used for sub packet 3 right channel 110 = i_pcm3L is used for sub packet 3 left channel 111 = i_pcm3R is used for sub packet 3 right channel	3b111
Spk3L_Sel	[26:24]	Selects the audio channel for subpacket 3 left channel data in multi-channel mode. The meaning is the same as SPK3R_SEL.	3b110
-	[23:22]	Reserved	2b00
Spk2R_Sel	[21:19]	Selects the audio channel for subpacket 2 right channel data in multi-channel mode. The meaning is the same as SPK2R_SEL.	3b101
Spk2L_Sel	[18:16]	Selects the audio channel selection for subpacket 2 left channel data in multi-channel mode. The meaning is the same as SPK2R_SEL.	3b100
-	[15:14]	Reserved	2b00
SPK1R_SEL	[13:11]	Selects the audio channel for subpacket 1 right channel data in multi-channel mode. The meaning is the same as SPK1R_SEL.	3b011
Spk1L_Sel	[10:8]	Selects the audio channel for subpacket 1 left channel data in multi-channel mode. The meaning is the same as SPK1R_SEL.	3b010
-	[7:6]	Reserved	2b00
Spk0R_Sel	[5:3]	Selects the audio channel for subpacket 0 right channel data in multi-channel mode. The meaning is the same as SPK0R_SEL.	3b001
Spk0L_Sel	[2:0]	Selects the audio channel selection for subpacket 0 left channel data in multi-channel mode. The meaning is the same as SPK0R_SEL.	3b000

10.3.3.25 Audio Related Packet Register (ACR_CON, R/W, Address = 0xFA11_0180)

ACR_CON	Bit	Description	Initial State
-	[7:5]	Reserved	3b000
Alt_Cts_Rate	[4:3]	<p>In some audio formats, the CTS value can be changed alternately.</p> <p>CTS value 1 = ACR_CTS[19:0] CTS value 2 = {ARC_CTS[19:8], ACR_LSB2}</p> <p>These two values can be transmitted alternately at the ratio of this register setting.</p> <p>00 = Always CTS value 1 01 = 1:1 (CTS value 1: CTS value2) 10 = 2:1 (CTS value 1: CTS value2) 11 = 3:1 (CTS value 1: CTS value2)</p> <p>Measured CTS mode, this value is not used.</p>	2b00
ACR_Tx_Mode	[2:0]	<p>000 = Does not transfer (Tx) the ACR packet. 001 = Tx once – Transmits ACR packet once; anytime available after this value is set. After transmitting, these bits are reset to all zero. 010 = Tx ACR_TXCNT times during every VBI period 011 = Tx by counting i_clk_vid for a given CTS value in the ACR_CTS0~2 registers. 100 = Measured CTS mode. Makes ACR packet with CTS value by counting TMDS clock for $F_s \times 128 / N$ duration. In this case, the 7 LSBs of N value (ACR_N register) should be all zero.</p>	3b000

10.3.3.26 Audio Related Packet Register (ACR_MCTS0/1/2)

- ACR_MCTS0, R, Address = 0xFA11_0184
- ACR_MCTS1, R, Address = 0xFA11_0188
- ACR_MCTS2, R, Address = 0xFA11_018C

ACR_MCTS0/1/2	Bit	Description	Initial State
-	[23:20]	Reserved	0x0
ACR_MCTS	[19:0]	Specifies the TMDS clock cycles for N[19:7] number of audio sample inputs. Only valid when measured CTS mode is set on ACR_CON register.	0x00001

10.3.3.27 Audio Related Packet Register (ACR_CTS0/1/2)

- ACR_CTS0, R/W, Address = 0xFA11_0190
- ACR_CTS1, R/W, Address = 0xFA11_0194
- ACR_CTS2, R/W, Address = 0xFA11_0198

ACR_CTS0/1/2	Bit	Description	Initial State
-	[23:20]	Reserved	0x0
ACR_CTS	[19:0]	Specifies the CTS value for transmission mode other than 'measured CTS' mode.	0x0003E8

10.3.3.28 Audio Related Packet Register (ACR_N0/1/2)

- ACR_N0, R/W, Address = 0xFA11_01A0
- ACR_N1, R/W, Address = 0xFA11_01A4
- ACR_N2, R/W, Address = 0xFA11_01A8

ACR_N0/1/2	Bit	Description	Initial State
-	[23:20]	Reserved	0
ACR_N	[19:0]	Specifies the N value in ACR packet.	0x003E8

10.3.3.29 Audio Related Packet Register (ACR_LSB2, R/W, Address = 0xFA11_01B0)

ACR_LSB2	Bit	Description	Initial State
ACR_LSB2	[7:0]	Specifies the alternate CTS least significant byte. For more information, see ALT_CTS_RATE in ACR_CON register.	0x00

10.3.3.30 Audio Related Packet Register (ACR_TXCNT, R/W, Address = 0xFA11_01B4)

ACR_TXCNT	Bit	Description	Initial State
-	[7:5]	Reserved	0
ACR_TXCNT	[4:0]	If ACR_TX_MODE is '10', the ACR packet will be transmitted 'ACR_TXCNT + 1' times per VBI period. ALT_CTS_RATE is also applied. This register is only valid if ACR_TX_MODE is '10'.	0x1F

10.3.3.31 Audio Related Packet Register (ACR_TXINTERVAL, R/W, Address = 0xFA11_01B8)

ACR_TXINTERVAL	Bit	Description	Initial State
ACR_TX_INTERVAL	[7:0]	If ACR_TX_MODE is '10', the ACR packet will be transmitted ACR_TXCNT times during VBI. This register specifies the number of cycles between each ACR packets and avoids continuous transmission in more than 18 packets within single DI band. It is only valid if ACR_TX_MODE is '10'.	0x63

10.3.3.32 Audio Related Packet Register (ACR_CTS_OFFSET, R/W, Address = 0xFA11_01BC)

ACR_CTS_OFFSET	Bit	Description	Initial State
ACR_CTS_OFFSET	[7:0]	If 'measured CTS mode' is used, the CTS value will be measured by counting the TMDS clock for a given duration. This value is added to measured CTS value. It is 8-bit signed integer, so subtraction is possible.	0x00

10.3.3.33 Audio Related Packet Register (GCP_CON, R/W, Address = 0xFA11_01C0)

GCP_CON	Bit	Description	Initial State
-	[7:3]	Reserved	5b00000
ENABLE_1st_VSYNC	[3]	<p>For interlace mode, enable this bit to transfer the GCP packet on the 1st VSYNC in a frame.</p> <p>0 = Does not transfer GCP packet 1 = Transfers the GCP packet</p> <p>On the other hand, for progressive mode, GCP packet is transferred regardless of this bit, that is, GCP packet in progressive mode is transferred every vsync if GCP_CON is 2b1x.</p>	1b0
ENABLE_2nd_VSYNC	[2]	<p>For interlace mode, enable this bit to transfer the GCP packet on the 2nd VSYNC in a frame.</p> <p>0 = Does not transfer GCP packet 1 = Transfers GCP packet</p>	1b1
GCP_CON	[1:0]	<p>00 = Does not transmit 01 = Transmits once 1x = Transmits every vsync</p> <p>Transmits GCP packet within 384 cycles after active vsync.</p>	2b00

10.3.3.34 Audio Related Packet Register (GCP_BYTE1, R/W, Address = 0xFA11_01D0)

GCP_BYTE1	Bit	Description	Initial State
GCP_BYTE1	[7:0]	Specifies the GCP packet's first data byte. It is either 0x10 (Clear AVMUTE) or 0x01 (Set AVMUTE). For more information, refer to Table 5-17 of HDMI specification.	0x00

10.3.3.35 Audio Related Packet Register (GCP_BYTE2, R/W, Address = 0xFA11_01D4)

GCP_BYTE2	Bit	Description	Initial State
PP	[7:4]	Specifies the Packing Phase (PP). This bit is read only.	0x0
CD	[3:0]	<p>Only supports 24bit mode.</p> <p>0100 : 24 bit</p> <p>Others : Reserved</p>	0x0

10.3.3.36 Audio Related Packet Register (GCP_BYTE3, R/W, Address = 0xFA11_01D8)

GCP_BYTE3	Bit	Description	Initial State
-	[7:1]	Reserved	7b0000000
GCP_BYTE3	[0]	Specifies the default state.	0

10.3.3.37 ACP Packet Register (ACP_CON, R/W, Address = 0xFA11_01E0)

ACP_CON	Bit	Description	Initial State
ACP_FR_RATE	[7:3]	Transmits the ACP packet once per every ACP_FR_RATE+1 frames (or fields).	5b00000
-	[2]	Reserved	0
ACP_TX_CON	[1:0]	00 = Does not transmit 01 = Transmits once 1x = Transmits every vsync with ACP_FR_RATE	2b00

10.3.3.38 ACP Packet Register (ACP_TYPE, R/W, Address = 0xFA11_01F0)

ACP_TYPE	Bit	Description	Initial State
ACP_TYPE	[7:0]	Specifies the HB1 of ACP packet header. For more information, see Table 5-18 in HDMI v1.3 specification.	0x00

10.3.3.39 ACP Packet Register (ACP_DATA00~16)

- ACP_DATA00, R/W, Address = 0xFA11_0200
- ACP_DATA01, R/W, Address = 0xFA11_0204
- ACP_DATA02, R/W, Address = 0xFA11_0208
- ACP_DATA03, R/W, Address = 0xFA11_020C
- ACP_DATA04, R/W, Address = 0xFA11_0210
- ACP_DATA05, R/W, Address = 0xFA11_0214
- ACP_DATA06, R/W, Address = 0xFA11_0218
- ACP_DATA07, R/W, Address = 0xFA11_021C
- ACP_DATA08, R/W, Address = 0xFA11_0220
- ACP_DATA09, R/W, Address = 0xFA11_0224
- ACP_DATA10, R/W, Address = 0xFA11_0228
- ACP_DATA11, R/W, Address = 0xFA11_022C
- ACP_DATA12, R/W, Address = 0xFA11_0230
- ACP_DATA13, R/W, Address = 0xFA11_0234
- ACP_DATA14, R/W, Address = 0xFA11_0238
- ACP_DATA15, R/W, Address = 0xFA11_023C
- ACP_DATA16, R/W, Address = 0xFA11_0240

ACP_DATA00~16	Bit	Description	Initial State
ACP_DATA00~16	[7:0]	Specifies the ACP packet body data registers (PB0~PB16 of ACP packet body). For more information, see Section 9.3 in HDMI v1.3 specification.	0x00

10.3.3.40 ISRC1/2 Packet Register (ISRC_CON, R/W, Address = 0xFA11_0250)

ISRC_CON	Bit	Description	Initial State
ISRC_FR_RATE	[7:3]	Transmits ISRC1 (with or without ISRC2) packet once every ISRC_FR_RATE+1 frames (or fields).	5b00000
ISRC2_EN	[2]	Transmits ISRC2 packet with ISRC1 packet.	0
ISRC_TX_CON	[1:0]	00 = Does not transmit 01 = Transmits once 1x = Transmits every vsync with ISRC_FR_RATE	2b00

10.3.3.41 ISRC1/2 Packet Register (ISRC1_HEADER1, R/W, Address = 0xFA11_0264)

ISRC1_HEADER1	Bit	Description	Initial State
ISRC_Cont	[7]	See Table 5-20 in HDMI v1.3 specification.	0
ISRC_Valid	[6]	See Table 5-20 in HDMI v1.3 specification.	0
-	[5:3]	Reserved	3b000
ISRC status	[2:0]	See Table 5-20 in HDMI v1.3 specification.	3b000

10.3.3.42 ISRC1/2 Packet Register (ISRC1_DATA 00~15)

- ISRC1_DATA 00, R/W, Address = 0xFA11_0270
- ISRC1_DATA 01, R/W, Address = 0xFA11_0274
- ISRC1_DATA 02, R/W, Address = 0xFA11_0278
- ISRC1_DATA 03, R/W, Address = 0xFA11_027C
- ISRC1_DATA 04, R/W, Address = 0xFA11_0280
- ISRC1_DATA 05, R/W, Address = 0xFA11_0284
- ISRC1_DATA 06, R/W, Address = 0xFA11_0288
- ISRC1_DATA 07, R/W, Address = 0xFA11_028C
- ISRC1_DATA 08, R/W, Address = 0xFA11_0290
- ISRC1_DATA 09, R/W, Address = 0xFA11_0294
- ISRC1_DATA 10, R/W, Address = 0xFA11_0298
- ISRC1_DATA 11, R/W, Address = 0xFA11_029C
- ISRC1_DATA 12, R/W, Address = 0xFA11_02A0
- ISRC1_DATA 13, R/W, Address = 0xFA11_02A4
- ISRC1_DATA 14, R/W, Address = 0xFA11_02A8
- ISRC1_DATA 15, R/W, Address = 0xFA11_02AC

ISRC1_DATA 00~15	Bit	Description	Initial State
ISRC1_DATA00~15	[7:0]	Specifies the ISRC2 packet body data (PB0~15 of ISRC2 packet body). For more information, see Table 5-21 in HDMI v1.3 specification.	0x00

10.3.3.43 ISRC1/2 Packet Register (ISRC2_DATA 00~15)

- ISRC2_DATA 00, R/W, Address = 0xFA11_02B0
- ISRC2_DATA 01, R/W, Address = 0xFA11_02B4
- ISRC2_DATA 02, R/W, Address = 0xFA11_02B8
- ISRC2_DATA 03, R/W, Address = 0xFA11_02B8
- ISRC2_DATA 04, R/W, Address = 0xFA11_02C0
- ISRC2_DATA 05, R/W, Address = 0xFA11_02C4
- ISRC2_DATA 06, R/W, Address = 0xFA11_02C8
- ISRC2_DATA 07, R/W, Address = 0xFA11_02CC
- ISRC2_DATA 08, R/W, Address = 0xFA11_02D0
- ISRC2_DATA 09, R/W, Address = 0xFA11_02D4
- ISRC2_DATA 10, R/W, Address = 0xFA11_02D8
- ISRC2_DATA 11, R/W, Address = 0xFA11_02DC
- ISRC2_DATA 12, R/W, Address = 0xFA11_02E0
- ISRC2_DATA 13, R/W, Address = 0xFA11_02E4
- ISRC2_DATA 14, R/W, Address = 0xFA11_02E8
- ISRC2_DATA 15, R/W, Address = 0xFA11_02EC

ISRC2_DATA 00~15	Bit	Description	Initial State
ISRC2_DATA00~15	[7:0]	Specifies the ISRC2 packet body data (PB0~15 of ISRC2 packet body). For more information, see Table 5-23 in HDMI v1.3 specification.	0x00

10.3.3.44 AVI InfoFrame Register (AVI_CON, R/W, Address = 0xFA11_0300)

AVI_CON	Bit	Description	Initial State
-	[7:2]	Reserved	6b000000
AVI_TX_CON	[1:0]	00 = Does not transmit 01 = Transmits once 1x = Transmits every vsync	2b00

10.3.3.45 AVI InfoFrame Register (AVI_CHECK_SUM, R/W, Address = 0xFA11_0310)

AVI_CHECK_SUM	Bit	Description	Initial State
AVI_CHECK_SUM	[7:0]	Specifies the AVI InfoFrame checksum byte (PB0 byte of AVI packet body).	0x00

10.3.3.46 AVI InfoFrame Register (AVI_DATA01~13)

- AVI_DATA01, R/W, Address = 0xFA11_0320
- AVI_DATA02, R/W, Address = 0xFA11_0324
- AVI_DATA03, R/W, Address = 0xFA11_0328
- AVI_DATA04, R/W, Address = 0xFA11_032C
- AVI_DATA05, R/W, Address = 0xFA11_0330
- AVI_DATA06, R/W, Address = 0xFA11_0334
- AVI_DATA07, R/W, Address = 0xFA11_0338
- AVI_DATA08, R/W, Address = 0xFA11_033C
- AVI_DATA09, R/W, Address = 0xFA11_0340
- AVI_DATA10, R/W, Address = 0xFA11_0344
- AVI_DATA11, R/W, Address = 0xFA11_0348
- AVI_DATA12, R/W, Address = 0xFA11_034C
- AVI_DATA13, R/W, Address = 0xFA11_0350

AVI_DATA01~AVI_DATA13	Bit	Description	Initial State
AVI_DATA01~AVI_DATA13	[7:0]	Specifies the AVI Infoframe packet data registers (PB1~PB13 bytes of AVI packet body).	0x00

10.3.3.47 Audio InfoFrame Register (AUI_CON, R/W, Address = 0xFA11_0360)

AUI_CON	Bit	Description	Initial State
-	[7:2]	Reserved	6b000000
AUI_TX_CON	[1:0]	00 = Does not transmit 01 = Transmits once 1x = Transmits every vsync	2b00

10.3.3.48 Audio InfoFrame Register (AUI_CHECK_SUM, R/W, Address = 0xFA11_0370)

AUI_CHECK_SUM	Bit	Description	Initial State
AUI_CHECK_SUM	[7:0]	Specifies the AUI checksum data (PB0 byte of AUI packet body).	0x00

10.3.3.49 Audio InfoFrame Register (AUI_DATA1~5)

- AUI_DATA1, R/W, Address = 0xFA11_0380
- AUI_DATA2, R/W, Address = 0xFA11_0384
- AUI_DATA3, R/W, Address = 0xFA11_0388
- AUI_DATA4, R/W, Address = 0xFA11_038C
- AUI_DATA5, R/W, Address = 0xFA11_0390

AUI_DATA1~AUI_DATA5	Bit	Description	Initial State
AUI_BYTET1 ~ AUI_BYTET5	[7:0]	Specifies the AUI packet body (PB1~PB5 bytes of AUI packet body). Note: AUI_BYTET5 is matched as: Write: AUI_BYTET5[7:3] Read: AUI_BYTET5[4:0]	0x00

10.3.3.50 MPEG Source InfoFrame (MPG_CON, R/W, Address = 0xFA11_03A0)

MPG_CON	Bit	Description	Initial State
-	[7:2]	Reserved	6b000000
MPG_TX_CON	[1:0]	00 = Does not transmit 01 = Transmits once 1x = Transmits every vsync	2b00

10.3.3.51 MPEG Source InfoFrame (MPG_CHECK_SUM, R/W, Address = 0xFA11_03B0)

MPG_CHECK_SUM	Bit	Description	Initial State
MPG_CHECK_SUM	[7:0]	Specifies the MPG infoframe checksum register (PB0 byte of MPG packet body).	0x00

10.3.3.52 MPEG Source InfoFrame (MPG_DATA1~5)

- MPG_DATA1, R/W, Address = 0xFA11_03C0
- MPG_DATA2, R/W, Address = 0xFA11_03C4
- MPG_DATA3, R/W, Address = 0xFA11_03C8
- MPG_DATA4, R/W, Address = 0xFA11_03CC
- MPG_DATA5, R/W, Address = 0xFA11_03D0

MPG_DATA1~MPG_DATA5	Bit	Description	Initial State
MPG_DTAT1~MPG_DATA5	[7:0]	Specifies the MPG Infoframe packet data (PB1~PB5 bytes of MPG packet body).	all zeros

These registers can be used for Source Product Descriptor (SPD) packet transmission. Furthermore, they consist of full configurable header and packet body registers (3 bytes header register and 28 bytes packet body registers), so that they can be used for transmission of any type of packet.

10.3.3.53 Source Product Descriptor Infoframe (SPD_CON, R/W, Address = 0xFA11_0400)

SPD_CON	Bit	Description	Initial State
-	[7:2]	Reserved	6b000000
SPD_TX_CON	[1:0]	00 = Does not transmit 01 = Transmits once 1x = Transmits every vsync	2b00

10.3.3.54 Source Product Descriptor Infoframe (SPD_HEADER0/1/2)

- SPD_HEADER0, R/W, Address = 0xFA11_0410
- SPD_HEADER1, R/W, Address = 0xFA11_0414
- SPD_HEADER2, R/W, Address = 0xFA11_0418

SPD_HEADER0/1/2	Bit	Description	Initial State
SPD_HEADER0	[7:0]	Specifies the HB0 byte of SPD packet header.	0x00
SPD_HEADER1	[7:0]	Specifies the HB1 byte of SPD packet header.	0x00
PD_HEADER2	[7:0]	Specifies the HB2 byte of SPD packet header.	0x00

10.3.3.55 Source Product Descriptor Infoframe (SPD_DATA00~27)

- SPD_DATA00, R/W, Address = 0xFA11_0420
- SPD_DATA01, R/W, Address = 0xFA11_0424
- SPD_DATA02, R/W, Address = 0xFA11_0428
- SPD_DATA03, R/W, Address = 0xFA11_042C
- SPD_DATA04, R/W, Address = 0xFA11_0430
- SPD_DATA05, R/W, Address = 0xFA11_0434
- SPD_DATA06, R/W, Address = 0xFA11_0438
- SPD_DATA07, R/W, Address = 0xFA11_043C
- SPD_DATA08, R/W, Address = 0xFA11_0440
- SPD_DATA09, R/W, Address = 0xFA11_0444
- SPD_DATA10, R/W, Address = 0xFA11_0448
- SPD_DATA11, R/W, Address = 0xFA11_044C
- SPD_DATA12, R/W, Address = 0xFA11_0450
- SPD_DATA13, R/W, Address = 0xFA11_0454
- SPD_DATA14, R/W, Address = 0xFA11_0458
- SPD_DATA15, R/W, Address = 0xFA11_045C
- SPD_DATA16, R/W, Address = 0xFA11_0460
- SPD_DATA17, R/W, Address = 0xFA11_0464
- SPD_DATA18, R/W, Address = 0xFA11_0468
- SPD_DATA19, R/W, Address = 0xFA11_046C
- SPD_DATA20, R/W, Address = 0xFA11_0470
- SPD_DATA21, R/W, Address = 0xFA11_0474
- SPD_DATA22, R/W, Address = 0xFA11_0478
- SPD_DATA23, R/W, Address = 0xFA11_047C
- SPD_DATA24, R/W, Address = 0xFA11_0480
- SPD_DATA25, R/W, Address = 0xFA11_0484
- SPD_DATA26, R/W, Address = 0xFA11_0488
- SPD_DATA27, R/W, Address = 0xFA11_048C

SPD_DATA00~27	Bit	Description	Initial State
SPD_DATA00 ~ SPD_DATA27	[7:0]	Specifies the SPD packet data registers (PB0~PB27 bytes).	0x00

10.3.3.56 HDCP Register Description (HDCP_SHA1_00~19)

- HDCP_SHA1_00, R/W, Address = 0xFA11_0600
- HDCP_SHA1_01, R/W, Address = 0xFA11_0604
- HDCP_SHA1_02, R/W, Address = 0xFA11_0608
- HDCP_SHA1_03, R/W, Address = 0xFA11_060C
- HDCP_SHA1_04, R/W, Address = 0xFA11_0610
- HDCP_SHA1_05, R/W, Address = 0xFA11_0614
- HDCP_SHA1_06, R/W, Address = 0xFA11_0618
- HDCP_SHA1_07, R/W, Address = 0xFA11_061C
- HDCP_SHA1_08, R/W, Address = 0xFA11_0620
- HDCP_SHA1_09, R/W, Address = 0xFA11_0624
- HDCP_SHA1_10, R/W, Address = 0xFA11_0628
- HDCP_SHA1_11, R/W, Address = 0xFA11_062C
- HDCP_SHA1_12, R/W, Address = 0xFA11_0630
- HDCP_SHA1_13, R/W, Address = 0xFA11_0634
- HDCP_SHA1_14, R/W, Address = 0xFA11_0638
- HDCP_SHA1_15, R/W, Address = 0xFA11_063C
- HDCP_SHA1_16, R/W, Address = 0xFA11_0640
- HDCP_SHA1_17, R/W, Address = 0xFA11_0644
- HDCP_SHA1_18, R/W, Address = 0xFA11_0648
- HDCP_SHA1_19, R/W, Address = 0xFA11_064C

HDCP_SHA1_00~19	Bit	Description	Initial State
HDCP_SHA1	[159:0]	Specifies the SHA-1 value of 160-bit HDCP repeater (LSB first). These registers are readable but they are not modified by HDCP H/W.	All zeros

10.3.3.57 HDCP Register Description (HDCP_SHA_RESULT, R/W, Address = 0xFA11_0670)

HDCP_SHA_RESULT	Bit	Description	Initial State
-	[7:2]	Reserved	6b000000
Hdcp_Sha_Valid_Ready	[1]	Indicates that the HW performed the SHA comparison. Must be cleared by SW by writing 0. 0 = Not ready 1 = Ready	0
Hdcp_Sha_Valid	[0]	Indicates that the SHA-1 comparison succeeds. Must be cleared by SW by writing 0. 0 = Valid 1 = Not valid	0

10.3.3.58 HDCP Register Description (HDCP_KSV_LIST_0~4)

- HDCP_KSV_LIST_0, R/W, Address = 0xFA11_0650
- HDCP_KSV_LIST_1, R/W, Address = 0xFA11_0654
- HDCP_KSV_LIST_2, R/W, Address = 0xFA11_0658
- HDCP_KSV_LIST_3, R/W, Address = 0xFA11_065C
- HDCP_KSV_LIST_4, R/W, Address = 0xFA11_0660

HDCP_KSV_LIST_0~4	Bit	Description	Initial State
HDCP_KSV_LIST	[39:0]	Specifies little endian addressing and one KSV value from the HDCP repeater's KSV list. These registers are readable.	All zeros

10.3.3.59 HDCP Register Description (HDCP_KSV_LIST_CON, R/W, Address = 0xFA11_0664)

HDCP_KSV_LIST_CON	Bit	Description	Initial State
-	[7:4]	Reserved	4b0000
Hdcp_Ksv_Write_Done	[3]	After writing KSV data into HDCP_KSV_LIST_X registers and then writing “1” to this register, the HW processes the written KSV value and clears this bit to “0”. 0 = Does not write 1 = Writes KSV data into HDCP_KSV_LIST_X registers and then writes “1” to this register	0
Hdcp_Ksv_List_Empty	[2]	If the number of KSV list is zero, set this value to enable the SHA-1 module calculate without KSV list. 0 = Not empty 1 = Empty	0
Hdcp_Ksv_End	[1]	Indicates that current KSV value in HDCP_KSV_LIST_X registers is the last one. 0 = Not End 1 = End	0
Hdcp_Ksv_Read	[0]	After writing KSV data in HDCP_KSV_LIST_X registers, the HDCP SHA-1 module keeps the KSV value in internal buffer and sets this flag to ‘1’ for notifying it has been read. After setting the flag to ‘1’, the SW clears to ‘0’ at the same time when writing the HDCP_KSV_WRITE_DONE bit for next KSV list value. 0 = Not Read 1 = Read	1

10.3.3.60 HDCP Register Description (HDCP_CTRL1, R/W, Address = 0xFA11_0680)

HDCP_CTRL1	Bit	Description	Initial State
-	[7:4]	Reserved	4b0000
-	[3]	Reserved	0
Timeout	[2]	Sets if Rx is the repeater and its KSV list is not ready until five seconds. 0 = Not timeout 1 = Timeout (KSV Ready bit in the HDCP_BCAPS register is not high until five seconds) and re-starts the 1st authentication. Refer to Figure 2-6 in the HDCP 1.3 specification.	0
CP_Desired	[1]	Enables HDCP. 0 = Not Desired 1 = Desired	0
-	[0]	Reserved	0

10.3.3.61 HDCP Register Description (HDCP_CTRL2, R/W, Address = 0xFA11_0684)

HDCP_CTRL2	Bit	Description	Initial State
-	[7:1]	Reserved	7b0000000
Revocation_Set	[0]	Specifies the KSV list that is on the revocation list. Setting this bit fails the 2nd authentication. 0 = Does not set revocation 1 = Sets revocation	0

10.3.3.62 HDCP Register Description (HDCP_CHECK_RESULT, R/W, Address = 0xFA11_0690)

HDCP_CHECK_RESULT	Bit	Description	Initial State
-	[7:2]	Reserved	6b000000
Ri_Match_Result	[1:0]	Writes the result of comparison between Ri of Rx and Tx as the following values: (Ri : Tx, Ri' : Rx) Must be cleared by SW after setting 10 or 11 before next Ri interrupt occurs. 0x = Don't care 10 = Ri ≠ Ri' 11 = Ri = Ri'	2b00

10.3.3.63 HDCP Register Description (HDCP_BKSV0~4)

- HDCP_BKSV0, R/W, Address = 0xFA11_06A0
- HDCP_BKSV1, R/W, Address = 0xFA11_06A4
- HDCP_BKSV2, R/W, Address = 0xFA11_06A8
- HDCP_BKSV3, R/W, Address = 0xFA11_06AC
- HDCP_BKSV4, R/W, Address = 0xFA11_06B0

HDCP_BKSV0~4	Bit	Description	Initial State
HDCP_BKSV	[39:0]	Specifies the key selection vector (KSV) value from the receiver.	All zeros

10.3.3.64 HDCP Register Description (HDCP_AKSV0~4)

- HDCP_AKSV0, R, Address = 0xFA11_06C0
- HDCP_AKSV1, R, Address = 0xFA11_06C4
- HDCP_AKSV2, R, Address = 0xFA11_06C8
- HDCP_AKSV3, R, Address = 0xFA11_06CC
- HDCP_AKSV4, R, Address = 0xFA11_06D0

HDCP_AKSV0~4	Bit	Description	Initial State
HDCP_AKSV	[39:0]	Specifies the KSV value of transmitter.	All zeros

10.3.3.65 HDCP Register Description (HDCP_An_0~7)

- HDCP_An_0, R, Address = 0xFA11_06E0
- HDCP_An_1, R, Address = 0xFA11_06E4
- HDCP_An_2, R, Address = 0xFA11_06E8
- HDCP_An_3, R, Address = 0xFA11_06EC
- HDCP_An_4, R, Address = 0xFA11_06F0
- HDCP_An_5, R, Address = 0xFA11_06F4
- HDCP_An_6, R, Address = 0xFA11_06F8
- HDCP_An_7, R, Address = 0xFA11_06FC

HDCP_An_0~7	Bit	Description	Initial State
HDCP_An	[63:0]	Specifies the 64-bit random number generated by Tx (An).	All zeros

10.3.3.66 HDCP Register Description (HDCP_BCAPS, R/W, Address = 0xFA11_0700)

HDCP_BCAPS	Bit	Description	Initial State
-	[7]	Reserved	0
Repeater	[6]	Specifies the receiver that supports downstream connections. 0 = Not Repeater 1 = Repeater	0
Ready	[5]	Indicates KSV FIFO, SHA-1 is calculation ready. 0 = Not Ready 1 = Ready	0
Fast	[4]	Specifies the receiver devices that support 400KHz transfer. 0 = Not Fast 1 = Fast	0
Reserved	[3:2]	Must be 0.	0
1.1_Features	[1]	Supports EESS, advance cipher, and enhanced link verification. 0 = Does not set 1 = Sets	0
Fast_Reauthentication	[0]	Specifies all HDMI receivers that are capable of reauthentication. 0 = Does not set 1 = Sets	0

10.3.3.67 HDCP Register Description (HDCP_BSTATUS_0/1)

- HDCP_BSTATUS_0, R/W, Address = 0xFA11_0710
- HDCP_BSTATUS_1, R/W, Address = 0xFA11_0714

HDCP_BSTATUS_0/1	Bit	Description	Initial State
-	[15:13]	Reserved	3b000
Hdmi_Mode	[12]	Specifies the HDMI mode. If set, HDCP works in HDMI mode.	0
Max_Cascade_Exceeded	[11]	Specifies the topology error.	0
Depth	[10:8]	Specifies the cascade depth.	3b000
Max_Devs_Exceeded	[7]	Indicates the topology error. 0 = No Error 1 = Error	0
Device_Count	[6:0]	Specifies the total number of attached downstream devices.	0

10.3.3.68 HDCP Register Description (HDCP_Ri_0/1)

- HDCP_Ri_0, R, Address = 0xFA11_0740
- HDCP_Ri_1, R, Address = 0xFA11_0744

HDCP_Ri_0/1	Bit	Description	Initial State
HDCP_Ri	[15:0]	Specifies the HDCP Ri value of transmitter.	0x0000

10.3.3.69 HDCP Register Description (HDCP_I2C_INT, R/W, Address = 0xFA11_0780)

HDCP_I2C_INT	Bit	Description	Initial State
-	[7:1]	Reserved	7b00000000
HDCP_I2C_INT	[0]	Specifies the HDCP I2C interrupt status (active high). It indicates the start of I2C transaction if it is set. After active, it should be cleared by S/W by writing 0. 0 = Does not occur 1 = Occurs	0

10.3.3.70 HDCP Register Description (HDCP_AN_INT, R/W, Address = 0xFA11_0790)

HDCP_AN_INT	Bit	Description	Initial State
-	[7:1]	Reserved	7b00000000
HDCP_AN_INT	[0]	Specifies the HDCP An Interrupt status (active high). If An value is available, it is set. After active, it should be cleared by S/W by writing 0. 0 = Does not occur 1 = Occurs	0

10.3.3.71 HDCP Register Description (HDCP_WATCHDOG_INT, R/W, Address = 0xFA11_07A0)

HDCP_WATCHDOG_INT	Bit	Description	Initial State
-	[7:1]	Reserved	7b0000000
HDCP_WATCHDOG_INT	[0]	Specifies the HDCP watchdog interrupt status (active high). If the repeater bit value is set after 1st authentication success, this bit is set. After active, it should be cleared by S/W by writing 0. 0 = Does not occur 1 = Occurs	0

10.3.3.72 HDCP Register Description (HDCP_RI_INT, R/W, Address = 0xFA11_07B0)

HDCP_RI_INT	Bit	Description	Initial State
-	[7:1]	Reserved	7b0000000
HDCP_RI_INT	[0]	If Ri value is updated internally (at every 128 video frames), it is set to high. After set, it should be cleared by S/W by writing 0. 0 = Does not occur 1 = Occurs	0

10.3.3.73 Ri Check Register (HDCP_Ri_Compare_0, R/W, Address = 0xFA11_07D0)

HDCP_Ri_Compare_0	Bit	Description	Initial State
Enable	[7]	Enables the interrupt for this frame number index.	1
Frame Number index	[6:0]	If the frame count reaches “frame number index”, an Ri link integrity check interrupt occurs.	7b0000000

10.3.3.74 Ri Check Register (HDCP_Ri_Compare_1, R/W, Address = 0xFA11_07D4)

HDCP_Ri_Compare_1	Bit	Description	Initial State
Enable	[7]	Enables the interrupt for this frame number index.	0
Frame Number index	[6:0]	If the frame count reaches “frame number index”, an Ri Link integrity check interrupt occurs.	7b1111111

10.3.3.75 Ri Check Register (HDCP_Frame_Count, R, Address = 0xFA11_07E0)

HDCP_Frame_Count	Bit	Description	Initial State
-	[7]	Reserved	0
Frame Count	[6:0]	Specifies the current value of frame count index in hardware.	7b0000000

10.3.3.76 Gamut Metadata Packet Register (GAMUT_CON, R/W, Address = 0xFA11_0500)

GAMUT_CON	Bit	Description	Initial State
-	[7:2]	Reserved	6b00000
GAMUT_CON	[1:0]	00 = Does not transmit 01 = Transmits once 1x = Transmits every vsync	2b00

10.3.3.77 Gamut Metadata Packet Register (GAMUT_HEADER0, R/W, Address = 0xFA11_0504)

GAMUT_HEADER0	Bit	Description	Initial State
HB0	[7:0]	Specifies the HB0 value in Table 5-30. For more information, refer to HDMI 1.3 specification.	0x00

10.3.3.78 Gamut Metadata Packet Register (GAMUT_HEADER1, R/W, Address = 0xFA11_0508)

GAMUT_HEADER1	Bit	Description	Initial State
Next_Field	[7]	Indicates the effectiveness of GBD carried in this packet on the next video field.	0
GBD_profile	[6:4]	Specifies the transmission profile number (only profile 0 is supported).	3b000
Affected_Gamut_Seq_Num	[3:0]	Indicates which video fields are relevant for this metadata.	4b0000

10.3.3.79 Gamut Metadata Packet Register (GAMUT_HEADER2, R/W, Address = 0xFA11_050C)

GAMUT_HEADER2	Bit	Description	Initial State
No_Crnt_GBD	[7]	Indicates that there is no gamut metadata available for the currently transmitted video.	0
Reserved	[6]	Reserved	0
Packet_Seq	[5:4]	Indicates whether this packet is first, intermediate, last, or the only packet in a gamut metadata packet sequence.	2b00
Current_Gamut_Seq_Num	[3:0]	Indicates the gamut number of the currently transmitted video stream.	4b0000

10.3.3.80 Gamut Metadata Packet Register (GAMUT_METADATA0~27)

- GAMUT_METADATA0, R/W, Address = 0xFA11_0510
- GAMUT_METADATA0, R/W, Address = 0xFA11_0514
- GAMUT_METADATA2, R/W, Address = 0xFA11_0518
- GAMUT_METADATA3, R/W, Address = 0xFA11_051C
- GAMUT_METADATA4, R/W, Address = 0xFA11_0520
- GAMUT_METADATA5, R/W, Address = 0xFA11_0524
- GAMUT_METADATA6, R/W, Address = 0xFA11_0528
- GAMUT_METADATA7, R/W, Address = 0xFA11_052C
- GAMUT_METADATA8, R/W, Address = 0xFA11_0530
- GAMUT_METADATA9, R/W, Address = 0xFA11_0534
- GAMUT_METADATA10, R/W, Address = 0xFA11_0538
- GAMUT_METADATA11, R/W, Address = 0xFA11_053C
- GAMUT_METADATA12, R/W, Address = 0xFA11_0540
- GAMUT_METADATA13, R/W, Address = 0xFA11_0544
- GAMUT_METADATA14, R/W, Address = 0xFA11_0548
- GAMUT_METADATA15, R/W, Address = 0xFA11_054C
- GAMUT_METADATA16, R/W, Address = 0xFA11_0550
- GAMUT_METADATA17, R/W, Address = 0xFA11_0554
- GAMUT_METADATA18, R/W, Address = 0xFA11_0558
- GAMUT_METADATA19, R/W, Address = 0xFA11_055C
- GAMUT_METADATA20, R/W, Address = 0xFA11_0560
- GAMUT_METADATA21, R/W, Address = 0xFA11_0564
- GAMUT_METADATA22, R/W, Address = 0xFA11_0568
- GAMUT_METADATA23, R/W, Address = 0xFA11_056C
- GAMUT_METADATA24, R/W, Address = 0xFA11_0570
- GAMUT_METADATA25, R/W, Address = 0xFA11_0574
- GAMUT_METADATA26, R/W, Address = 0xFA11_0578
- GAMUT_METADATA27, R/W, Address = 0xFA11_057C

GAMUT_METADATA	Bit	Description	Initial State
GAMUT_METADATA0~ GAMUT_METADATA27	[7:0]	Specifies the gamut metadata packet body for P0 transmission profile.	0x00

10.3.3.81 Video Mode Register (VIDEO_PATTERN_GEN, R/W, Address = 0xFA11_05C4)

VIDEO_PATTERN_GEN	Bit	Description	Initial State
-	[7:2]	Reserved	6b000000
Ext_Video_En	[1]	0 = Ext off 1 = Ext on	0
Video Pattern Enable	[0]	0 = Disables 1 = Uses internally generated video pattern	0

10.3.3.82 Video Mode Register (HPD_GEN, R/W, Address = 0xFA11_05C8)

HPD_GEN	Bit	Description	Initial State
HPD_Duration	[7:0]	Specifies the number of cycles for determining stable HPD input. Internal count = TMDS clock * HPD_Duration * 16 (cycles) Default value = 0x1 (16 TMDS clock cycles)	0x01

10.3.4 SPDIF REGISTER

10.3.4.1 SPDIF Register (SPDIFIN_CLK_CTRL, R/W, Address = 0xFA13_0000)

SPDIFIN_CLK_CTRL	Bit	Description	Initial State
-	[7:2]	Reserved	6b000000
ready_clk_down	[1]	0 = Enables clock 1 = Readies for disabling clock (default)	1
power_on	[0]	0 = Disables clock (default) 1 = Activates clock If this bit is reset, SPDIFIN stops checking the input signal just before the next 'subframe' of SPDIF signal format and waits for the 'acknowledge' signal from HDMI for unresolved previous 'request' towards HDMI. It then asserts 'ready_clk_down' as high. To initialize internal states, assert software reset, that is, SPDIFIN_OP_CTRL. op_ctrl=00b right after activating clock again.	0

The spdif_clk is gated by an external clock gating module for low-power. Disabling the clock should not cause stalling of HDMI data transfer. Therefore, the system processor requests disabling of the clock by setting the power_on register to low and the module acknowledges this request by setting the ready_clk_down register to high after a current transaction on the I2C bus, and HDMI is finished. The module must not commence a new bus transaction until the system processor sets the power_on register to high again.

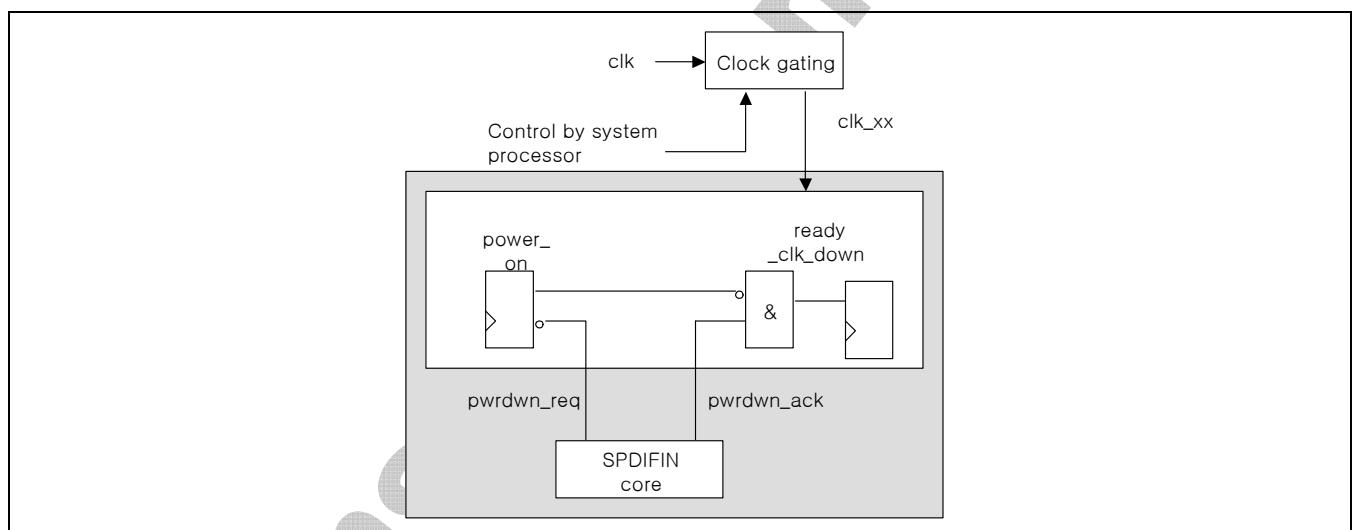


Figure 10-13 Structure of Power Down Circuit

The system processor switches off clk_xx when the ready_clk_down bit is one and the power_on bit is zero.

The system processor switches on the clock at any time. After having switched on clk_xx, the system processor sets the power_on bit to 1, which forces the ready_clk_down bit to zero.

Once the reset clock of SPDIFIN is switched off, the power_on bit is set to zero and the ready_clk_down bit is set to one.

10.3.4.2 SPDIF Registers (SPDIFIN_OP_CTRL, R/W, Address = 0xFA13_0004)

SPDIFIN_OP_CTRL	Bit	Description	Initial State
-	[7:2]	Reserved	6b000000
op_ctrl	[1:0]	<p>00b = Specifies the software reset 01b = Specifies the status checking mode (run) 11b = Specifies the status checking and HDMI operation modes (run with HDMI) Others = Undefined, do not use</p> <p>00b = During a software reset, all state machines are set to the idle or init state and all internal registers are set to their initial values. Interrupt status registers are cleared, all other registers that are writable by the system processor keep their values.</p> <p>01b = This command should be asserted after SPDIFIN_CLK_CTRL.power_on is set. SPDIFIN starts the clock recovery. When recovery is done, SPDIFIN detects preambles of SPDIF signal format and stream data header, abnormal time signal input, abnormal signal input, and also reports these status via interrupts in SPDIFIN_IRQ_STATUS.</p> <p>11b = Specifies the “01b” case operations, checks internal buffer overflow, and writes received data, which can be either audio sample word of PCM or payload of stream. Data will be transferred via HDMI.</p> <ul style="list-style-type: none"> - Assert ‘op_ctrl’=11b after SPDIFIN_IRQ_STATUS. ch_status_recovered_ir is asserted at least once for linear PCM data. - Assert ‘op_ctrl’=11b after SPDIFIN_IRQ_STATUS. stream_header_detected_ir is asserted at least once for non-linear PCM stream data. 	0

10.3.4.3 SPDIF Register (SPDIFIN_IRQ_MASK, R/W, Address = 0xFA13_0008)

SPDIFIN_IRQ_MASK	Bit	Description	Initial State
buf_overflow_ir_en	[7]	Specifies the mask bit for Interrupt 7.	0
-	[6]	Reserved	0
-	[5]	Reserved	0
stream_header_detected_ir_en	[4]	Specifies the mask bit for Interrupt 4.	0
stream_header_not_detected_ir_en	[3]	Specifies the mask bit for Interrupt 3.	0
wrong_preamble_ir_en	[2]	Specifies the mask bit for Interrupt 2.	0
ch_status_recovered_ir_en	[1]	Specifies the mask bit for Interrupt 1.	0
wrong_signal_ir_en	[0]	Specifies the mask bit for Interrupt 0.	0
For every bit:			
0 = Disables interrupt generation			
1 = Enables interrupt generation			

10.3.4.4 SPDIF Register (SPDIFIN_IRQ_STATUS, R/W, Address = 0xFA13_000C)

SPDIFIN_IRQ_STATUS	Bit	Description	Initial State
buf_overflow_ir	[7]	<p>0 = No interrupt 1 = Internal buffer overflow SPDIFIN internal buffer(s) (SPDIFIN_DATA_BUF_x) overflows if HDMI fails to transfer data from buffer(s) to memory on time.</p> <ul style="list-style-type: none"> - This interrupt is asserted only if SPDIFIN_OP_CTRL.op_ctrl is set to "011". - If this interrupt is not handled, SPDIFIN overwrites the next subframe data to internal data buffer (SPDIFIN_DATA_BUF_x) and continues data transfer via HDMI. 	0
-	[6]	Reserved	0
-	[5]	Reserved	0
stream_header_detected_ir	[4]	<p>0 = No interrupt 1 = Detects stream data header (Pa~Pd)</p> <ul style="list-style-type: none"> - This interrupt is asserted if SPDIFIN_OP_CTRL.op_ctrl is equal to 001b or 011b. - Cases for interrupt <ul style="list-style-type: none"> Case1: Initially after power_on Case2: Next stream header at right time if receiving stream data with SPDIFIN_CONFIG.data_type set as 'stream mode' Case3: Initially detects stream header if receiving stream data with SPDIFIN_CONFIG.data_type is set as 'PCM mode' 	0
stream_header_not_detected_ir	[3]	<p>0 = No interrupt 1 = Does not detect stream data header for 4096 repetition time</p> <ul style="list-style-type: none"> - This interrupt will be asserted if SPDIFIN_OP_CTRL.op_ctrl is equal to 001b or 011b. - Cases for interrupt <ul style="list-style-type: none"> Case1: Initially after power_on Case2: SPDIFIN receives the stream but is unable to find the next stream header for 4096 repetition time since previous stream header Case3: Is unable to find stream header for 4096 repetition time since previous reset of repetition time counter after previous interrupt of 'stream_header_not_detected_ir'. 	0

SPDIFIN_IRQ_STATUS	Bit	Description	Initial State
wrong_preamble_ir	[2]	<p>0 = No interrupt 1 = Detects preamble but indicates a problem with the detected time</p> <ul style="list-style-type: none"> - This interrupt is asserted when SPDIFIN_OP_CTRL.op_ctrl is equal to 001b or 011b. - Meaningless until ch_status_recovered_ir is asserted initially after SPDIFIN_OP_CTRL.op_ctrl=01b - Cases for interrupt <ul style="list-style-type: none"> Case1: Detects preamble in the middle of a subframe audio sample word time Case2: Does not detect the next preamble at exact time after a subframe duration Case3: Does not detect preamble B (or M or W) but detects other preamble at that time 	0
ch_status_recovered_ir	[1]	<p>0 = No interrupt 1 = Recovered channel status</p> <p>Detects two consecutive B-preambles; thus recovers 192-bit wide channel status.</p> <ul style="list-style-type: none"> - Only supports consumer mode, so only 36-bits are reconstructed. If you want to see the channel status bits through SPDIFIN_CH_STATUS_x, read two consecutive 'ch_status_recovered_ir' and the register each time. If these two channel status values are the same, you can rely on that value. 	0
wrong_signal_ir	[0]	<p>0 = No interrupt 1 = Clock recovery fails</p> <p>Cannot recover the clock from input due to tolerable range violation (unlock), no signal from outside, or non-biphase in non-preamble duration.</p> <ul style="list-style-type: none"> - Meaningless until ch_status_recovered_ir is asserted initially after SPDIFIN_OP_CTRL.op_ctrl=01b. 	0
For every bit, the following holds: Reading returns interrupt request status. Writing '0' has no effect. Writing '1' clears the interrupt request.			
1) Detection of stream header Waits for matching of Pa, Pb and 0xF872, 0x4E1F respectively. Waits for repetition time (from decoded PC value or user-defined PC in SPDIFIN_USER_VALUE.repetition_time_manual, based on SPDIFIN_CONFIG.PcPd_value_mode) Check for matching of Pa, Pb on right time.			

10.3.4.5 SPDIF Register (SPDIFIN_CONFIG_1, R/W, Address = 0xFA13_0010)

SPDIFIN_CONFIG_1	Bit	Description	Initial State
-	[7]	Reserved	0
noise_filter_samples	[6]	<p>0 = Filtering with three consecutive samples 1 = Filtering with two consecutive samples Noise filtering is done for over-sampled SPDIF input signal. This operation will be done as follows: If 'noise_filter_samples' is 0, three consecutive over-sampled signals are regarded as high or low only if those samples are all high or all low respectively. If one or two samples are low or high for three over-sample duration, those noise-filtered signals will keep the previous value. If 'noise_filter_samples' is 1, two consecutive over-sampled signals are regarded as a high or low only if those samples are all high or low respectively. This setting can be used for reduced over-sampling ratio; recommended over-sampling ratio is 10 (see also 'clk_divisor').</p>	0
-	[5]	Reserved (Must be '0')	0
PcPd_value_mode	[4]	<p>0 = Automatically sets 1 = Manually sets If '0' is used for automatic setting, Pc and Pd values are chosen by value of Pc and Pd from decoded stream header, as reported in SPDIFIN_Px_INFO. If you set this register, the receiver will use SPDIFIN_USER_VALUE[31:16] and SPDIFIN_USER_VALUE[15:4] values as Pc and Pd respectively, instead of decoded data from stream header, as reported in SPDIFIN_Px_INFO. If the (cf) burst payload length is automatically or manually set, it affects the data size to be written in memory via HDMI—by dumping the full sub-frame for last bit in burst payload length. For example, if burst payload length is 257-bit, that is, 16 sub-frames * 16-bit + 1-bit, then HDMI will write data in 17 consecutive sub-frames.</p>	0
word_length_value_mode	[3]	<p>0 = Automatically sets 1 = Manually sets If '0' is used for automatic setting, the word length value will be chosen by value of channel status from decoded SPDIF format, as reported in SPDIFIN_CH_STATUS_1.word_length. If user sets this register, the receiver will use SPDIFIN_USER_VALUE[3:0] value as word length instead of decoded data from channel status, as reported in SPDIFIN_CH_STATUS_1.word_length.</p>	0

SPDIFIN_CONFIG_1	Bit	Description	Initial State
U_V_C_P_report	[2]	<p>0 = Neglects 'user_bit', 'validity_bit', 'channel status', and 'parity_bit' of SPDIF format.</p> <p>1 = Reports 'user_bit', 'validity_bit', 'channel status', and 'parity_bit' of SPDIF format</p> <p>The report will be delivered via HDMI for each sub-frame. Valid only if SPDIFIN_CONFIG.data_align is set for 32-bit mode. For more information, see SPDIFIN_DATA_BUF_x.</p>	0
-	[1]	Reserved (Must be '1')	1
data_align	[0]	<p>0 = 16-bit mode 1 = 32-bit mode</p> <p>16-bit: Only takes 16-bits from MSB in a sub-frame of SPDIF format, and then concatenates two consecutive 16-bit data in one 32-bit register of SPDIFIN_DATA_BUF_x.</p> <p>32-bit: Only takes data from one subframe with zero padding to MSB part. For example, 0x00ffff for 24-bit data.</p> <p>With stream mode, set 'word_length_value_mode' as 1 and set SPDIFIN_USER_VALUE.word_length_manual as 3b000.</p> <p>- These two modes will be applied to both modes of SPDIFIN_CONFIG.data_type, that is, PCM or stream. For more information, see SPDIFIN_DATA_BUF_x.</p>	0

10.3.4.6 SPDIF Register (SPDIFIN_CONFIG_2, R/W, Address = 0xFA13_0014)

SPDIFIN_CONFIG_2	Bit	Description	Initial State
-	[7:4]	Reserved	0x0
clk_divisor	[3:0]	<p>SPDIFIN_internal_clock = system_clock / (clk_divisor + 1) (SPDIFIN_internal_clock ≤ 135 Mhz)</p> <p>SPDIFIN over-samples the SPDIF input signal with internal clock that is divided from system clock.</p> <p>Recommended over-sampling ratio is 8~10, thus the following calculation holds:</p> <p>Recommended SPDIFIN_internal_clock = Sampling Frequency of SPDIF Input Signal * 64-bits * 10 times over-sampling</p> <p>For example, 48 kHz * 64-bits * 10 times over-sampling = 31 Mhz.</p>	0x0

10.3.4.7 SPDIF Register (SPDIFIN_USER_VALUE_1, R/W, Address = 0xFA13_0020)

SPDIFIN_USER_VALUE_1	Bit	Description	Initial State																		
repetition_time_manual_low	[7:4]	<p>Specifies the repetition time[3:0].</p> <p>Repetition_time_manual register has 12-bits value.</p> <p>This register is low by 4-bits.</p> <p>It counts one block of stream data and is valid only if SPDIFIN_CONFIG.PcPd_value_mode is set for manual mode.</p> <p>Unit: frames (1 frame = 2 sub-frames) of SPDIF format.</p> <p>The value should be actual repetition time minus one. For example, if you want to manually set the repetition time as 1536, you should write 1535.</p>	0x0																		
word_length_manual	[3:0]	<p>Specifies the word length.</p> <p>Used as size for transferring data to memory via HDMI; valid only when SPDIFIN_CONFIG.word_length_value_mode is set for manual mode.</p> <p>For more information, see SPDIFIN_DATA_BUF_x.</p> <table> <tr> <td>[0] is 1</td> <td>[0] is 0</td> <td>[3:1]</td> </tr> <tr> <td>101:</td> <td>24-bits</td> <td>20-bits</td> </tr> <tr> <td>001:</td> <td>23-bits</td> <td>19-bits</td> </tr> <tr> <td>010:</td> <td>22-bits</td> <td>18-bits</td> </tr> <tr> <td>011:</td> <td>21-bits</td> <td>17-bits</td> </tr> <tr> <td>100:</td> <td>20-bits</td> <td>16-bits</td> </tr> </table>	[0] is 1	[0] is 0	[3:1]	101:	24-bits	20-bits	001:	23-bits	19-bits	010:	22-bits	18-bits	011:	21-bits	17-bits	100:	20-bits	16-bits	0x0
[0] is 1	[0] is 0	[3:1]																			
101:	24-bits	20-bits																			
001:	23-bits	19-bits																			
010:	22-bits	18-bits																			
011:	21-bits	17-bits																			
100:	20-bits	16-bits																			

10.3.4.8 SPDIF Register (SPDIFIN_USER_VALUE_2, R/W, Address = 0xFA13_0024)

SPDIFIN_USER_VALUE_2	Bit	Description	Initial State
repetition_time_manual_high	[7:0]	<p>Specifies the repetition time[11:4].</p> <p>Repetition_time_manual register has 12-bits value.</p> <p>This register is high by 8-bits.</p> <p>Counts one block of stream data; valid only if SPDIFIN_CONFIG.PcPd_value_mode is set for manual mode.</p> <p>Unit: frames (1 frame = 2 sub-frames) of SPDIF format</p> <p>The value should be actual repetition time minus one. For example, if you want to manually set the repetition time as 1536, you should write 1535.</p>	0x00

10.3.4.9 SPDIF Register (SPDIFIN_USER_VALUE_3, R/W, Address = 0xFA13_0028)

SPDIFIN_USER_VALUE_3	Bit	Description	Initial State
burst_payload_length_manual_low	[7:0]	Specifies the burst_payload_length_manual[7:0]. Burst_payload_length register has 16-bits value. This register is low by 8-bits Valid only if SPDIFIN_CONFIG.PcPd_value_mode is set for manual mode. Unit: bits	0

10.3.4.10 SPDIF Register (SPDIFIN_USER_VALUE_4, R/W, Address = 0xFA13_002C)

SPDIFIN_USER_VALUE_4	Bit	Description	Initial State
burst_payload_length_manual_high	[7:0]	Specifies the burst_payload_length_manual[15:8]. Burst_payload_length register has 16-bits value. This register is high by 8-bits. Valid only if SPDIFIN_CONFIG.PcPd_value_mode is set for manual mode. Unit: bits.	0

The channel status registers are updated every 192 frames (1 block) of SPDIF format only for consumer mode.

10.3.4.11 Channel Status Register (SPDIFIN_CH_STATUS_0_1, R, Address = 0xFA13_0030)

SPDIFIN_CH_STATUS_0_1	Bit	Description	Initial State
channel_status_mode	[7:6]	00 = Mode 0 others = Reserved	2b00
emphasis	[5:3]	000 = Emphasis not indicated 100 = Emphasis – CD type	3b000
copyright_assertion	[2]	0 = Copyright 1 = No copyright	0
audio_sample_word	[1]	0 = Linear PCM 1 = Non-linear PCM	0
channel_status_block	[0]	0 = Consumer format 1 = Professional format	0
This register is updated every 192 frames (1 block) of SPDIF format. SPDIFIN_CH_STATUS_0_1 [7:0] is the matched internal register SPDIFIN_CH_STATUS_0 [7:0].			

10.3.4.12 Channel Status Register (SPDIFIN_CH_STATUS_0_2, R, Address = 0xFA13_0034)

SPDIFIN_CH_STATUS_0_2	Bit	Description	Initial State
category_code	[7:0]	Equipment type: [8:15] CD player: 1000_0000 DAT player: 1100_000L DCC player: 1100_001L Mini disc: 1001_001L (L: information about generation status of the material)	0x00

10.3.4.13 Channel Status Register (SPDIFIN_CH_STATUS_0_3, R, Address = 0xFA13_0038)

SPDIFIN_CH_STATUS_0_3	Bit	Description	Initial State
channel_number	[7:4]	Specifies the channel number (Bit 20 is LSB).	0x0
source_number	[3:0]	Specifies the source number (Bit 16 is LSB).	0x0

10.3.4.14 Channel Status Register (SPDIFIN_CH_STATUS_0_4, R, Address = 0xFA13_003C)

SPDIFIN_CH_STATUS_0_4	Bit	Description	Initial State
-	[7:6]	Reserved	2b00
clock_accuracy	[5:4]	Specifies the clock accuracy. 00 = level II, ±1000ppm 01 = level I, ±50ppm 10 = level III, variable pitch shifted	2b00
sampling_frequency	[3:0]	Specifies the sampling frequency. 0100 = 22.05kHz 0000 = 44.1kHz 1000 = 88.2kHz 1100 = 176.4kHz 0110 = 24kHz 0010 = 48kHz 1010 = 96kHz 1110 = 192kHz 0011 = 32kHz	0x0

10.3.4.15 Channel Status Register (SPDIFIN_CH_STATUS_1, R, Address = 0xFA13_0040)

SPDIFIN_CH_STATUS_1	Bit	Description	Initial State
-	[7:4]	Reserved	0x0
word_length	[3:1]	Specifies the word length. (field_size = 1) (field_size = 0) 000 = not indicated not indicated 101 = 24-bits 20-bits 100 = 23-bits 19-bits 010 = 22-bits 18-bits 110 = 21-bits 17-bits 001 = 20-bits 16-bits	3b000
field_size	[0]	Specifies the field size. 0 = Maximum length 20-bits 1 = Maximum length 24-bits	0

10.3.4.16 SPDIFIN Info Register (SPDIFIN_CH_STATUS_1, R, Address = 0xFA13_0048)

SPDIFIN_CH_STATUS_1	Bit	Description	Initial State
frame_cnt_low	[7:0]	<p>Specifies the frame count value [7:0].</p> <p>Frame_cnt register has 16-bits value. This is low by 8-bits.</p> <p>The period of a frame (two sub-frames) and register is updated every two sub-frames.</p> <p>It will be measured by 'SPDIFIN_internal_clk' made with SPDIFIN_CONFIG.clk_divisor.</p> <p>Unit: SPDIF_internal_clk cycles</p> <p>Recommended value for locking incoming signals: Over 0x220 (8.5timesx64-bits)</p>	0x00

10.3.4.17 SPDIFIN Info Register (SPDIFIN_CH_STATUS_2, R, Address = 0xFA13_004C)

SPDIFIN_CH_STATUS_2	Bit	Description	Initial State
frame_cnt_high	[7:0]	<p>Specifies the frame count value [15:8].</p> <p>Frame_cnt register has 16-bits value. This is high by 8-bits.</p> <p>The period of a frame (two sub-frames) and register is updated every two sub-frames. It is measured by 'SPDIFIN_internal_clk' made with SPDIFIN_CONFIG.clk_divisor.</p> <p>Unit: SPDIF_internal_clk cycles</p> <p>Recommended value for locking incoming signals: Over 0x220 (8.5timesx64-bits)</p>	0x0

10.3.4.18 SPDIFIN Info Register (SPDIFIN_Pc_INFO_1, R, Address = 0xFA13_0050)

SPDIFIN_Pc_INFO_1	Bit	Description	Initial State
error_flag	[7]	0 = Valid burst payload 1 = Burst payload may contain errors	0
-	[6:5]	Reserved	2b00
compressed_data_type	[4:0]	0d = Null data 1d = Dolby AC-3 2d = Reserved 3d = Pause 4d = MPEG-1 layer 1 5d = MPEG-1 layer 2 or 3 or MPEG-2 w/o extension 6d = MPEG-2 w/ extension 7d = Reserved 8d = MPEG-2 layer 1 low sampling freq. 9d = MPEG-2 layer 2 or 3 low sampling freq. 10d = Reserved 11d, 12d, 13d = DTS 14d~31d = Reserved	5b00000

10.3.4.19 SPDIFIN Info Register (SPDIFIN_Pc_INFO_2, R, Address = 0xFA13_0054)

SPDIFIN_Pc_INFO_2	Bit	Description	Initial State
bit_stream_number	[7:5]	Specifies the bit stream number.	3b000
data_type_dependent_info	[4:0]	Specifies the data type dependent information.	5b00000

10.3.4.20 SPDIFIN Info Register (SPDIFIN_Pd_INFO_1, R, Address = 0xFA13_0058)

SPDIFIN_Pd_INFO_1	Bit	Description	Initial State
burst_payload_length_low	[7:0]	Specifies the length of burst payload [7:0] (Unit: bits).	0x00

10.3.4.21 SPDIFIN Info Register (SPDIFIN_Pd_INFO_2, R, Address = 0xFA13_005C)

SPDIFIN_Pd_INFO_2	Bit	Description	Initial State
burst_payload_length_high	[7:0]	Specifies the length of burst payload [15:8] (Unit: bits).	0x00

10.3.4.22 SPDIFIN Info Register (SPDIFIN_DATA_BUFO_1/2/3)

- SPDIFIN_DATA_BUFO_1, R, Address = 0xFA13_0060
- SPDIFIN_DATA_BUFO_2, R, Address = 0xFA13_0064
- SPDIFIN_DATA_BUFO_3, R, Address = 0xFA13_0068

SPDIFIN_DATA_BUFO_1/2/3	Bit	Description	Initial State
received_data_0_1	[7:0]	Specifies the PCM or stream data for 1st burst of HDMI.	0x00
received_data_0_2	[7:0]	SPDIFIN_DATA_BUFO_1 = SPDIFIN_DATA_BUFO_0[7:0]	
received_data_0_3	[7:0]	SPDIFIN_DATA_BUFO_2 = SPDIFIN_DATA_BUFO_0[15:8] SPDIFIN_DATA_BUFO_3 = SPDIFIN_DATA_BUFO_0[23:16] If SPDIFIN_CONFIG.data_align is '0' for 16-bit, received_data is equal to {data_(N)th, data_(N+1)th}. If SPDIFIN_CONFIG.data_align is '1' for 32-bit, received_data is equal to {U, V, C, P, zero-padding, and data[n:0]}. If SPDIFIN_CONFIG.U_V_P_report is '0', received_data is equal to {zero-padding, data[n:0]}, , where, 'n' is dependent on SPDIFIN_CH_STATUS_1 and word_length if SPDIFIN_CONFIG.data_type is 0 for PCM. 'n' is equal to 15 if SPDIFIN_CONFIG.data_type is 1 for stream. If SPDIFIN_CONFIG.HDMI_burst_size is 1 burst, HDMI accesses only this data register.	

10.3.4.23 SPDIFIN Info Register (SPDIFIN_USER_BUFO, R, Address = 0xFA13_006C)

SPDIFIN_USER_BUFO	Bit	Description	Initial State
received_data_user_0	[7:4]	Specifies the user bit of 1st burst of HDMI received_data[7:4] = SPDIFIN_DATA_BUFO_0[31:28].	0x0
-	[3:0]	Reserved	0x0

10.3.4.24 SPDIFIN Info Register (SPDIFIN_DATA_BUF1_1/2/3)

- SPDIFIN_DATA_BUF_1_1, R, Address = 0xFA13_0070
- SPDIFIN_DATA_BUF_1_2, R, Address = 0xFA13_0074
- SPDIFIN_DATA_BUF_1_3, R, Address = 0xFA13_0078

SPDIFIN_DATA_BUF1_1/2/3	Bit	Description	Initial State
received_data_data_1_1	[7:0]	Specifies the PCM or stream data for 2nd burst of HDMI.	0x00
received_data_data_1_2	[7:0]	SPDIFIN_DATA_BUF_0_1 = SPDIFIN_DATA_BUF_0[7:0]	
received_data_data_1_3	[7:0]	SPDIFIN_DATA_BUF_0_2 = SPDIFIN_DATA_BUF_0[15:8] SPDIFIN_DATA_BUF_0_3 = SPDIFIN_DATA_BUF_0[23:16]	
		If SPDIFIN_CONFIG.data_align is '0' for 16-bit, received_data is equal to {data_(N)th, data_(N+1)th}. If SPDIFIN_CONFIG.data_align is '1' for 32-bit, received_data is equal to {U, V, C, P, zero-padding, data[n:0]}. If SPDIFIN_CONFIG.U_V_P_report is '0', received_data is equal to {zero-padding, data[n:0]}. , where 'n' is dependent on SPDIFIN_CH_STATUS_1 and word_length if SPDIFIN_CONFIG.data_type is '0' for PCM. 'n' is equal to 15 if SPDIFIN_CONFIG.data_type is '1' for stream. If SPDIFIN_CONFIG.HDMI_burst_size is 1 burst, HDMI accesses only this data register.	

10.3.4.25 SPDIFIN Info Register (SPDIFIN_USER_BUF_1, R, Address = 0xFA13_007C)

SPDIFIN_USER_BUF_1	Bit	Description	Initial State
received_data_user_1	[7:4]	Specifies the user bit of 2nd burst of HDMI received_data[7:4] = SPDIFIN_DATA_BUF_1[31:28]	0x0
-	[3:0]	Reserved	0x0

10.3.5 I2S REGISTER

10.3.5.1 I2S Register (I2S_CLK_CON, R/W, Address = 0xFA14_0000)

I2S_CLK_CON	Bit	Description	Initial State
-	[7:1]	Reserved	0
i2s_en	[0]	Enables the I2S clock. 0 = I2S will be disabled (default) 1 = I2S will be activated Sets i2s_en after other registers are configured. If you want to reset the I2S, this register is 0 → 1.	0

10.3.5.2 I2S Register (I2S_CON_1, R/W, Address = 0xFA14_0004)

I2S_CON_1	Bit	Description	Initial State
-	[7:2]	Reserved	6b000000
r_sc_pol	[1]	Specifies the SDATA is synchronous to 0 = SCLK falling edge 1 = SCLK rising edge	0
r_ch_pol	[0]	Specifies the LRCLK polarity. 0 = Left channel for low polarity 1 = Left channel for high polarity	0

10.3.5.3 I2S Register (I2S_CON_2, R/W, Address = 0xFA14_0008)

I2S_CON_2	Bit	Description	Initial State
-	[7]	Reserved	0
mlsb	[6]	0 = MSB first mode 1 = LSB first mode	0
bit_ch	[5:4]	Specifies the bit clock per frame (Frame = left + right). 2b00 = 32fs 2b01 = 48fs 2b10 = Reserved	2b01
data_num	[3:2]	Specifies the serial data bit per channel. 2b01 = 16-bit 2b10 = 20-bit 2b11 = 24-bit	2b01
i2s_mode	[1:0]	2b00 = I2S basic format 2b10 = left justified format 2b11 = right justified format	2b10

10.3.5.4 I2S Register (I2S_PIN_SEL_0, R/W, Address = 0xFA14_000C)

I2S_PIN_SEL_0	Bit	Description	Initial State
-	[7]	Reserved	0
pin_sel_1	[6:4]	Selects the SCLK (I2S). 3b111 = i_i2s_in[1] 3b110 = i_i2s_in[6] 3b101 = i_i2s_in[5] 3b100 = i_i2s_in[4] 3b011 = i_i2s_in[3] 3b010 = i_i2s_in[2] 3b001 = i_i2s_in[1] 3b000 = i_i2s_in[0] Note: SCLK is selected with i_i2s_in[5](0x101).	3b111
-	[3]	Reserved	0
pin_sel_0	[2:0]	Selects the LRCK (I2S). 3b111 = i_i2s_in[0] 3b110 = i_i2s_in[6] 3b101 = i_i2s_in[5] 3b100 = i_i2s_in[4] 3b011 = i_i2s_in[3] 3b010 = i_i2s_in[2] 3b001 = i_i2s_in[1] 3b000 = i_i2s_in[0] Note: LRCK is selected with i_i2s_in[6](0x110).	3b111

10.3.5.5 I2S Register (I2S_PIN_SEL_1, R/W, Address = 0xFA14_0010)

I2S_PIN_SEL_1	Bit	Description	Initial State
-	[7]	Reserved	0
pin_sel_3	[6:4]	Selects the SDATA_1 (I2S). 3b111 = i_i2s_in[3] 3b110 = i_i2s_in[6] 3b101 = i_i2s_in[5] 3b100 = i_i2s_in[4] 3b011 = i_i2s_in[3] 3b010 = i_i2s_in[2] 3b001 = i_i2s_in[1] 3b000 = i_i2s_in[0] Note: SDATA_1 is selected with i_i2s_in[3](0x011).	3b111
-	[3]	0	0
pin_sel_2	[2:0]	Selects the SDATA_0 (I2S). 3b111 = i_i2s_in[2] 3b110 = i_i2s_in[6] 3b101 = i_i2s_in[5] 3b100 = i_i2s_in[4] 3b011 = i_i2s_in[3] 3b010 = i_i2s_in[2] 3b001 = i_i2s_in[1] 3b000 = i_i2s_in[0] Note: SDATA_0 is selected with i_i2s_in[4](0x100).	3b111

10.3.5.6 I2S Register (I2S_PIN_SEL_2, R/W, Address = 0xFA14_0014)

I2S_PIN_SEL_2	Bit	Description	Initial State
-	[7]	Reserved	0
pin_sel_5	[6:4]	Selects the SDATA_3 (I2S). 3b111 = i_i2s_in[5] 3b110 = i_i2s_in[6] 3b101 = i_i2s_in[5] 3b100 = i_i2s_in[4] 3b011 = i_i2s_in[3] 3b010 = i_i2s_in[2] 3b001 = i_i2s_in[1] 3b000 = i_i2s_in[0] Note: SDATA_3 is selected with i_i2s_in[1](0x001).	3b111
-	[3]	0	0
pin_sel_4	[2:0]	Selects the SDATA_2 (I2S). 3b111 = i_i2s_in[4] 3b110 = i_i2s_in[6] 3b101 = i_i2s_in[5] 3b100 = i_i2s_in[4] 3b011 = i_i2s_in[3] 3b010 = i_i2s_in[2] 3b001 = i_i2s_in[1] 3b000 = i_i2s_in[0] Note: SDATA_2 is selected with i_i2s_in[2](0x010).	3b111

10.3.5.7 I2S Register (I2S_PIN_SEL_3, R/W, Address = 0xFA14_0018)

I2S_PIN_SEL_3	Bit	Description	Initial State
-	[7:3]	Reserved	0
pin_sel_6	[2:0]	Selects the DSD_D5(DSD). 3b111 = i_i2s_in[6] 3b110 = i_i2s_in[6] 3b101 = i_i2s_in[5] 3b100 = i_i2s_in[4] 3b011 = i_i2s_in[3] 3b010 = i_i2s_in[2] 3b001 = i_i2s_in[1] 3b000 = i_i2s_in[0] Note: DSD_D5 is selected with i_i2s_in[0](0x000).	3b111

10.3.5.8 I2S Register (I2S_DSD_CON, R/W, Address = 0xFA14_001C)

I2S_DSD_CON	Bit	Description	Initial State
-	[7:2]	Reserved	0
r_dsd_pol	[1]	1 = DSD_DATA changes at DSD_CLK rising edge 0 = DSD_DATA changes at DSD_CLK falling edge	1
dsd_en	[0]	1 = Enables DSD module 0 = Disables DSD module	0

10.3.5.9 I2S Register (I2S_IN_MUX_CON, R/W, Address = 0xFA14_0020)

I2S_IN_MUX_CON	Bit	Description	Initial State
f_num	[7:5]	Specifies the number of stage of noise filter for I2S input pins. 000 = no filtering 001 = 2-stage filter 010 = 3-stage filter 011 = 4-stage filter 100 = 5-stage filter Others = Reserved	3b011
in_en	[4]	Enables i2s_in, which is a sub-module at the input stage. 0 = Disables i2s_in module 1 = Enables i2s_in module If disabled, all output data is '0'.	0
audio_sel	[3:2]	Selects the audio. 2b00 = Enables SPDIF audio data 2b01 = Enables I2S audio data 2b10 = Enables DSD audio data	0
CUV_sel	[1]	Selects the CUV. 0 = Enables SPDIF CUV data 1 = Enables I2S CUV data	0
mux_en	[0]	Enables i2s_mux, which is a sub-module for audio selection. 0 = Disables i2s_mux module 1 = Enables i2s_mux module If disabled, all output data is '0'.	0

10.3.5.10 Channel Status Register (I2S_CH_ST_CON, R/W, Address = 0xFA14_0024)

I2S_CH_ST_CON	Bit	Description	Initial State
-	[7:1]	Reserved	0
channel_status_reload	[0]	0 = Updates the shadow channel status registers 1 = Sets this bit to update the shadow channel status registers with the values updated in I2S_CH_ST_0 ~ I2S_CH_ST_4. This bit is cleared if the shadow channel status registers are updated.	0

Channel status information needs to be applied to the audio stream at the IEC 60958 block boundary. For this synchronization, there are two register sets for channel status block. You can set the channel status registers, I2S_CH_ST_0~I2S_CH_ST_4, while the I2S Rx module still refers to the shadow channel status registers, I2S_CH_ST_SH_0~I2S_CH_ST_CH4.

To reflect the user configuration in the channel status registers, set 'channel_status_reload' bit in I2S_CH_ST_CON, then I2S Rx module copies the channel status registers into the shadow channel status registers at the beginning of an IEC-60958 block.

10.3.5.11 Channel Status Register (I2S_CH_ST_0, I2S_CH_ST_SH_0)

- I2S_CH_ST_0, R/W, Address = 0xFA14_0028
- I2S_CH_ST_SH_0, R, Address = 0xFA14_003C

I2S_CH_ST_0, I2S_CH_ST_SH_0	Bit	Description	Initial State
channel_status_mode	[7:6]	2b00 = Mode 0 Others = Reserved	0
emphasis	[5:3]	If bit1 = 0, 3b000 = 2 audio channels without pre-emphasis* 3b001 = 2 audio channels with 50us/ 15us pre-emphasis If bit1 = 1, 3b000 = default state	0
copyright	[2]	0 = Copyright 1 = No copyright	0
audio_sample_word	[1]	0 = linear PCM 1 = Non-linear PCM	0
channel_status_block	[0]	0 = Consumer format 1 = Professional format	0

NOTE: The bits listed here in channel status registers look swapped from those in IEC-60958-3 specification, as the bit order is different (LSB is right-most bit).

10.3.5.12 Channel Status Register (I2S_CH_ST_1, I2S_CH_ST_SH_1)

- I2S_CH_ST_1, R/W, Address = 0xFA14_002C
- I2S_CH_ST_SH_1, R, Address = 0xFA14_0040

I2S_CH_ST_1, I2S_CH_ST_SH_1	Bit	Description	Initial State
category	[7:0]	Specifies the equipment type. CD player: 0000_0001 DAT player: L000_0011 DCC player: L100_0011 Mini disc: L100_1001 (L: information about generation status of the material)	0

10.3.5.13 Channel Status Register (I2S_CH_ST_2, I2S_CH_ST_SH_2)

- I2S_CH_ST_2, R/W, Address = 0xFA14_0030
- I2S_CH_ST_SH_2, R, Address = 0xFA14_0044

I2S_CH_ST_2, I2S_CH_ST_SH_2	Bit	Description	Initial State
channel_number	[7:4]	Specifies the channel number. Note: bit4 is LSB.	0
source_number	[3:0]	Specifies the source number. Note: bit0 is LSB.	0

10.3.5.14 Channel Status Register (I2S_CH_ST_3, I2S_CH_ST_SH_3)

- I2S_CH_ST_3, R/W, Address = 0xFA14_0034
- I2S_CH_ST_SH_3, R, Address = 0xFA14_0048

I2S_CH_ST_3, I2S_CH_ST_SH_3	Bit	Description	Initial State
-	[7:6]	Reserved	2b00
Clock_Accuracy	[5:4]	Specifies the clock accuracy, as specified in IEC-60958-3. 2b01 = Level I, ±50 ppm 2b00 = Level II, ±1000 ppm 2b10 = Level III, variable pitch shifted	2b00
Sampling_Frequency	[3:0]	Specifies the sampling frequency, as specified in IEC-60958-3. 4b0000 = 44.1 kHz 4b0010 = 48 kHz 4b0011 = 32 kHz 4b1010 = 96 kHz ...	0

10.3.5.15 Channel Status Register (I2S_CH_ST_4, I2S_CH_ST_SH_4)

- I2S_CH_ST_4, R/W, Address = 0xFA14_0038
- I2S_CH_ST_SH_4, R, Address = 0xFA14_004C

I2S_CH_ST_4, I2S_CH_ST_SH_4	Bit	Description	Initial State																					
Org_Sampling_Freq	[7:4]	<p>Specifies the original sampling frequency.</p> <p>4b1111 = 44.1KHz 4b0111 = 88.2KHz 4b1011 = 22.05KHz 4b0011 = 176.4KHz ... For other frequencies, refer to the original sampling frequency specified in IEC-60958-3.</p>	0x0																					
Word_Length	[3:1]	<p>Specifies the word length.</p> <table> <tr><td>Max. length</td><td>24-bits</td><td>20-bits</td></tr> <tr><td>3b000</td><td>= not defined</td><td>not defined</td></tr> <tr><td>3b001</td><td>= 20-bits</td><td>16-bits</td></tr> <tr><td>3b010</td><td>= 22-bits</td><td>18-bits</td></tr> <tr><td>3b100</td><td>= 23-bits</td><td>19-bits</td></tr> <tr><td>3b101</td><td>= 24-bits</td><td>20-bits</td></tr> <tr><td>3b110</td><td>= 21-bits</td><td>17-bits</td></tr> </table>	Max. length	24-bits	20-bits	3b000	= not defined	not defined	3b001	= 20-bits	16-bits	3b010	= 22-bits	18-bits	3b100	= 23-bits	19-bits	3b101	= 24-bits	20-bits	3b110	= 21-bits	17-bits	3b000
Max. length	24-bits	20-bits																						
3b000	= not defined	not defined																						
3b001	= 20-bits	16-bits																						
3b010	= 22-bits	18-bits																						
3b100	= 23-bits	19-bits																						
3b101	= 24-bits	20-bits																						
3b110	= 21-bits	17-bits																						
Max_Word_Length	[0]	<p>Specifies the maximum sample word length.</p> <p>1 = 24-bits 0 = 20-bits</p>	0																					

10.3.5.16 Channel Status Register (I2S_VD_DATA, R/W, Address = 0xFA14_0050)

I2S_VD_DATA	Bit	Description	Initial State
-	[7:1]	Reserved	7b0000000
validity_flag	[0]	<p>Specifies the validity bit.</p> <p>0 = Audio sample is reliable 1 = Audio sample is unreliable</p>	0

10.3.5.17 MUX Control Register (I2S_MUX_CH, R/W, Address = 0xFA14_0054)

I2S_MUX_CH	Bit	Description	Initial State
CH3_R_en	[7]	0 = Disables channel 3 right audio data output 1 = Enables channel 3 right audio data output	0
CH3_L_en	[6]	0 = Disables channel 3 left audio data output 1 = Enables channel 3 left audio data output	0
CH2_R_en	[5]	0 = Disables channel 2 right audio data output 1 = Enables channel 2 right audio data output	0
CH2_L_en	[4]	0 = Disables channel 2 left audio data output 1 = Enables channel 2 left audio data output	0
CH1_R_en	[3]	0 = Disables channel 1 right audio data output 1 = Enables channel 1 right audio data output	0
CH1_L_en	[2]	0 = Disables channel 1 left audio data output 1 = Enables channel 1 left audio data output	0
CH0_R_en	[1]	0 = Disables channel 0 right audio data output 1 = Enables channel 0 right audio data output	1
CH0_L_en	[0]	0 = Disables channel 0 left audio data output 1 = Enables channel 0 left audio data output	1

10.3.5.18 MUX Control Register (I2S_MUX_CUV, R/W, Address = 0xFA14_0058)

I2S_MUX_CUV	Bit	Description	Initial State
-	[7:2]	Reserved	6b000000
CUV_R_en	[1]	0 = Disables right channel CUV data 1 = Enables right channel CUV data	1
CUV_L_en	[0]	0 = Disables left channel CUV data 1 = Enables left channel CUV data	1

10.3.5.19 Interrupt Control Register (I2S_IRQ_MASK, R/W, Address = 0xFA14_005C)

I2S_IRQ_MASK	Bit	Description	Initial State
-	[7:2]	Reserved	6b000000
int_2_mask	[1]	Disables interrupt request by int_2 interrupt. 0 = Disables int_2 interrupt 1 = Enables int_2 interrupt	0
-	[0]	Reserved	0

10.3.5.20 Interrupt Control Register (I2S_IRQ_STATUS, R/W, Address = 0xFA14_0060)

I2S_IRQ_STATUS	Bit	Description	Initial State
-	[7:2]	Reserved	0
int_2	[1]	Specifies the interrupt status, that is, the wrong register setting. This interrupt is asserted if the I2S_CON_2.bit_ch is set to 32fs, while I2S_CON_2.data_num is set to either 20-bit or 24-bit. According to the wrong register setting, some audio data MSB bits may be removed. The audio data is not available.	0
-	[0]	Reserved	0

For bit[1], the following holds:
Reading returns the interrupt request status.

Warning: 0: Has no effect.
1: Clears the interrupt request.

10.3.5.21 Output Buffer Register (I2S_CHX_Y_Z, R, Address = 0xFA14_0064~0xFA14_00D8)

I2S_CHX_Y_Z	Bit	Description	Initial State
I2S_CHX_Y_Z	[7:0]	<p>Specifies the PCM output data from I2S Rx module.</p> <p>X = Channel = 0, 1, 2 Y = Left/Right = L, R Z = Byte number</p> <p>I2S_CHX_Y_0 = PCM_X[7:0] I2S_CHX_Y_1 = PCM_X[15:8] I2S_CHX_Y_2 = PCM_X[23:16] I2S_CHX_Y_3 = PCM_X[27:24]</p> <p>Channel 3 has 24-bit width.</p> <p>I2S_CH3_Y_0 = PCM3_Y[7:0] I2S_CH3_Y_1 = PCM3_Y[15:8] I2S_CH3_Y_2 = PCM3_Y[23:16]</p>	0x00

10.3.5.22 Output Buffer Register (I2S_CUV_L_R, R, Address = 0xFA14_00DC)

I2S_CUV_L_R	Bit	Description	Initial State
-	[7]	Reserved	0
CUV_R	[6:4]	Specifies the VUCP data of right channel. CUV_R[3:0] = {valid bit, user bit, channel state bit, parity bit}	3b000
-	[3]	Reserved	0
CUV_L	[2:0]	Specifies the VUCP data of left channel. CUV_L[3:0] = {valid bit, user bit, channel state bit, parity bit}	3b000

10.3.6 TIMING GENERATOR REGISTER (TG CONFIGURE/STATUS REGISTER)

10.3.6.1 TG Command Register (TG_CMD, R/W, Address = 0xFA15_0000)

TG_CMD	Bit	Description	Initial State
Reserved	[7:5]	Reserved	000
getsync_type	[4]	Specifies the timing correction enable bit. If this bit is set, the input VSYNC timing error relative to output VSYNC is corrected. 0 = Disables 1 = Enables	0
getsync_en	[3]	Enables BT656 input synchronization.	0
Reserved	[2]	Reserved	0
field_en	[1]	Enables field mode. For 1080i, this should be enabled.	0
tg_en	[0]	Specifies the TG global enable bit.	0

10.3.6.2 Horizontal Full Size (TG_H_FSZ_L, R/W, Address = 0xFA15_0018)

TG_H_FSZ_L	Bit	Description	Initial State
TG_H_FSZ_L	[7:0]	Specifies the horizontal full size (1~8191) (Lower part).	0x72

10.3.6.3 Horizontal Full Size (TG_H_FSZ_H, R/W, Address = 0xFA15_001C)

TG_H_FSZ_H	Bit	Description	Initial State
Reserved	[7:5]	Reserved	0x0
TG_H_FSZ_H	[4:0]	Specifies the horizontal full size (1~8191) (Upper part).	0x6

10.3.6.4 Horizontal Active Start Position (TG_HACT_ST_L, R/W, Address = 0xFA15_0020)

TG_HACT_ST_L	Bit	Description	Initial State
TG_HACT_ST_L	[7:0]	Specifies the horizontal active start position (1~4095) (Lower part).	0x05

10.3.6.5 Horizontal Active Start Position (TG_HACT_ST_H, R/W, Address = 0xFA15_0024)

TG_HACT_ST_H	Bit	Description	Initial State
Reserved	[7:4]	Reserved	0x0
TG_HACT_ST_H	[3:0]	Specifies the horizontal active start position (1~4095) (Upper part).	0x1

10.3.6.6 Horizontal Active Size (TG_HACT_SZ_L, R/W, Address = 0xFA15_0028)

TG_HACT_SZ_L	Bit	Description	Initial State
TG_HACT_SZ_L	[7:0]	Specifies the horizontal active size (0~4095) (Lower part).	0x00

10.3.6.7 Horizontal Active Size (TG_HACT_SZ_H, R/W, Address = 0xFA15_002C)

TG_HACT_SZ_H	Bit	Description	Initial State
Reserved	[7:4]	Reserved	0x0
TG_HACT_SZ_H	[3:0]	Specifies the horizontal active size (0~4095) (Upper part).	0x5

10.3.6.8 Vertical Full Size (TG_V_FSZ_L, R/W, Address = 0xFA15_0030)

TG_V_FSZ_L	Bit	Description	Initial State
TG_V_FSZ_L	[7:0]	Specifies the vertical full size (1~2047) (Lower part).	0xEE

10.3.6.9 Vertical Full Size (TG_V_FSZ_H, R/W, Address = 0xFA15_0034)

TG_V_FSZ_H	Bit	Description	Initial State
Reserved	[7:3]	Reserved	0x0
TG_V_FSZ_H	[2:0]	Specifies the vertical full size (1~2047) (Upper part).	0x2

10.3.6.10 VSYNC Position (TG_VSYNC_L, R/W, Address = 0xFA15_0038)

TG_VSYNC_L	Bit	Description	Initial State
TG_VSYNC_L	[7:0]	Specifies the vertical sync position. If field enable is set, this bit takes the top field vsync position (1~2047) (Lower part).	0x01

10.3.6.11 VSYNC Position (TG_VSYNC_H, R/W, Address = 0xFA15_003C)

TG_VSYNC_H	Bit	Description	Initial State
Reserved	[7:3]	Reserved	0x0
TG_VSYNC_H	[2:0]	Specifies the vertical sync position. If field enable is set, this bit takes the top field vsync position (1~2047) (Upper part).	0x0

10.3.6.12 Bottom Field VSYNC Position (TG_VSYNC2_L, R/W, Address = 0xFA15_0040)

TG_VSYNC2_L	Bit	Description	Initial State
TG_VSYNC2_L	[7:0]	Specifies the vertical sync position for bottom field (1~2047) (Lower part).	0x33

10.3.6.13 Bottom Field VSYNC Position (TG_VSYNC2_H, R/W, Address = 0xFA15_0044)

TG_VSYNC2_H	Bit	Description	Initial State
Reserved	[7:0]	Reserved	0x0
TG_VSYNC2_H	[2:0]	Specifies the vertical sync position for bottom field (1~2047) (Upper part).	0x2

10.3.6.14 Vertical Active Start Position (G_VACT_ST_L, R/W, Address = 0xFA15_0048)

TG_VACT_ST_L	Bit	Description	Initial State
TG_VACT_ST_L	[7:0]	Specifies the vertical active start position (1~2047) (Lower part).	0x1a

10.3.6.15 Vertical Active Start Position (TG_TACT_ST_H, R/W, Address = 0xFA15_004C)

TG_VACT_ST_H	Bit	Description	Initial State
Reserved	[7:3]	Reserved	0x0
TG_VACT_ST_H	[2:0]	Specifies the vertical active start position (1~2047) (Upper part).	0x0

10.3.6.16 Vertical Active Size (TG_VACT_SZ_L, R/W, Address = 0xFA15_0050)

TG_VACT_SZ_L	Bit	Description	Initial State
TG_VACT_SZ_L	[7:0]	Specifies the vertical active size (0~2047) (Lower part).	0xD0

10.3.6.17 Vertical Active Size (TG_TACT_SZ_H, R/W, Address = 0xFA15_0054)

TG_VACT_SZ_H	Bit	Description	Initial State
Reserved	[7:3]	Reserved	0x0
TG_VACT_SZ_H	[2:0]	Specifies the vertical active size (0~2047) (Upper part).	0x2

10.3.6.18 Field Change Position (TG_FIELD_CHG_L, R/W, Address = 0xFA15_0058)

TG_FIELD_CHG_L	Bit	Description	Initial State
TG_FIELD_CHG_L	[7:0]	Specifies the HDMI field position. (Lower part).	0x33

10.3.6.19 Field Change Position (TG_FIELD_CHG_H, R/W, Address = 0xFA15_005C)

TG_FIELD_CHG_H	Bit	Description	Initial State
Reserved	[7:3]	Reserved	0x0
TG_FIELD_CHG_H	[2:0]	Specifies the HDMI field position. (Upper part).	0x2

10.3.6.20 Bottom Field Vertical Active Start Position (TG_VACT_ST2_L, R/W, Address = 0xFA15_0060)

TG_VACT_ST2_L	Bit	Description	Initial State
TG_VACT_ST2_L	[7:0]	Specifies the HDMI vertical active start position for bottom field (Lower part).	0x48

10.3.6.21 Bottom Field VSYNC Position for HDMI (TG_VACT_ST2_H, R/W, Address = 0xFA15_0064)

TG_VACT_ST2_H	Bit	Description	Initial State
Reserved	[7:3]	Reserved	0x0
TG_VACT_ST2_L	[2:0]	Specifies the HDMI vertical active start position for bottom field (Upper part).	0x2

10.3.6.22 VSYNC Position for HDMI (TG_VSYNC_TOP_HDMI_L, R/W, Address = 0xFA15_0078)

TG_VSYNC_TOP_HDMI_L	Bit	Description	Initial State
TG_VSYNC_TOP_HDMI_L	[7:0]	Specifies the HDMI VSYNC position for top field (Lower part).	0x01

10.3.6.23 VSYNC Position for HDMI (TG_VSYNC_TOP_HDMI_H, R/W, Address = 0xFA15_007C)

TG_VSYNC_TOP_HDMI_H	Bit	Description	Initial State
Reserved	[7:3]	Reserved	0x0
TG_VSYNC_TOP_HDMI_H	[2:0]	Specifies the HDMI VSYNC position for top field (Upper part).	0x0

10.3.6.24 Bottom Field VSYNC Position for HDMI (TG_VSYNC_BOT_HDMI_L, R/W, Address = 0xFA15_0080)

TG_VSYNC_BOT_HDMI_L	Bit	Description	Initial State
TG_VSYNC_BOT_HDMI_L	[7:0]	Specifies the HDMI VSYNC position for bottom field (Lower part).	0x01

10.3.6.25 Bottom Field VSYNC Position for HDMI (TG_VSYNC_BOT_HDMI_H, R/W, Address = 0xFA15_0084)

TG_VSYNC_BOT_HDMI_H	Bit	Description	Initial State
Reserved	[7:3]	Reserved	0x0
TG_VSYNC_BOT_HDMI_H	[2:0]	Specifies the HDMI VSYNC position for bottom field (Upper part).	0x0

10.3.6.26 Top Field Change Start Position for HDMI (TG_FIELD_TOP_HDMI_L, R/W, Address = 0xFA15_0088)

TG_FIELD_TOP_HDMI_L	Bit	Description	Initial State
TG_FIELD_TOP_HDMI_L	[7:0]	Specifies the HDMI top field start position (Lower part).	0x01

10.3.6.27 Top Field Change Start Position for HDMI (TG_FIELD_TOP_HDMI_H, R/W, Address = 0xFA15_008C)

TG_FIELD_TOP_HDMI_H	Bit	Description	Initial State
Reserved	[7:3]	Reserved	0x0
TG_FIELD_TOP_HDMI_H	[2:0]	Specifies the HDMI top field start position (Upper part).	0x0

10.3.6.28 Bottom Field Change Start Position for HDMI (TG_FIELD_BOT_HDMI_L, R/W, Address = 0xFA15_0090)

TG_FIELD_BOT_HDMI_L	Bit	Description	Initial State
TG_FIELD_BOT_HDMI_L	[7:0]	Specifies the HDMI bottom field start position (Lower part).	0x33

10.3.6.29 Bottom Field VSYNC Start Position for HDMI (TG_FIELD_BOT_HDMI_H, R/W, Address = 0xFA15_0094)

TG_FIELD_BOT_HDMI_H	Bit	Description	Initial State
Reserved	[7:3]	Reserved	0x0
TG_FIELD_BOT_HDMI_H	[2:0]	Specifies the HDMI bottom field start position (Upper part).	0x2

10.3.6.30 HSYNC width Configuration for MHL Interface (MHL_HSYNC_WIDTH, R/W, Address = 0xFA15_017C)

MHL_HSYNC_WIDTH	Bit	Description	Initial State
MHL_HSYNC_WIDTH	[7:0]	Specifies the HSYNC width for MHL interface (Unit: pixel clock). The HSYNC width is MHL_HSYNC_WIDTH + 1.	0xF

10.3.6.31 VSYNC width Configuration for MHL Interface (MHL_VSYNC_WIDTH, R/W, Address = 0xFA15_0180)

MHL_VSYNC_WIDTH	Bit	Description	Initial State
MHL_VSYNC_WIDTH	[7:0]	Specifies the VSYNC width for MHL interface (Unit: line).	0x1

10.3.6.32 RGB Clock Inversion for MHL Interface (MHL_CLK_INV, R/W, Address = 0xFA15_0184)

MHL_CLK_INV	Bit	Description	Initial State
MHL_CLK_INV	[0]	Specifies the clock out inversion for MHL interface. 0 = Normal 1 = Inversion	0x0

10.3.6.33 HDCP E-FUSE Control Register (HDCP_E_FUSE_CTRL, W, Address = 0xFA16_0000)

HDCP_E_FUSE_CTRL	Bit	Description	Initial State
-	[7:1]	Reserved	4b0000
HDCP_KEY_READ	[0]	0 = Normal 1 = To read HDCP key from e-fuse.	0

10.3.6.34 HDCP E-FUSE Control Register (HDCP_E_FUSE_STATUS, R, Address = 0xFA16_0004)

HDCP_E_FUSE_STATUS	Bit	Description	Initial State
-	[7:3]	Reserved	4b0000
EFUSE_ECC_FAIL	[2]	0 = Normal 1 = ECC fail	
EFUSE_ECC_BUSY	[1]	0 = Not busy 1 = Busy	
EFUSE_ECC_DONE	[0]	0 = Normal 1 = ECC done	0

10.3.6.35 HDCP E-FUSE Control Register (EFUSE_ADDR_WIDTH, R/W, Address = 0xFA16_0008)

EFUSE_ADDR_WIDTH	Bit	Description	Initial State
EFUSE_ADDR_WIDTH	[7:0]	Specifies the address width (Unit: HDMI link PCLK, default: 83MHz, 12n).	0x14

10.3.6.36 HDCP E-FUSE Control Register (EFUSE_SIGDEV_ASSERT, R/W, Address = 0xFA16_000C)

EFUSE_SIGDEV_ASSERT	Bit	Description	Initial State
EFUSE_SIGDEV_ASSERT	[7:0]	Specifies the SIGDEV asserting position (Unit: HDMI Link PCLK, default: 83MHz, 12n).	0x0

10.3.6.37 HDCP E-FUSE Control Register (EFUSE_SIGDEV_DE-ASSERT, R/W, Address = 0xFA16_0010)

EFUSE_SIGDEV_DEASSERT	Bit	Description	Initial State
EFUSE_SIGDEV_DEASSERT	[7:0]	Specifies the SIGDEV de-asserting position (Unit: HDMI Link PCLK, default: 83MHz, 12n)	0x8

10.3.6.38 HDCP E-FUSE Control Register (EFUSE_PRCHG_ASSERT, R/W, Address = 0xFA16_0014)

EFUSE_PRCHG_ASSERT	Bit	Description	Initial State
EFUSE_PRCHG_ASSERT	[7:0]	Specifies the PRCHG asserting position (Unit: HDMI Link PCLK, default: 83MHz, 12n)	0x0

10.3.6.39 HDCP E-FUSE Control Register (EFUSE_PRCHG_DE-ASSERT, R/W, Address = 0xFA16_0018)

EFUSE_PRCHG_DE-ASSERT	Bit	Description	Initial State
EFUSE_PRCHG_DEASSERT	[7:0]	Specifies the PRCHG de-asserting position (Unit: HDMI link PCLK, default: 83MHz, 12n).	0xC

10.3.6.40 HDCP E-FUSE Control Register (EFUSE_FSET_ASSERT, R/W, Address = 0xFA16_001C)

EFUSE_FSET_ASSERT	Bit	Description	Initial State
EFUSE_FSET_ASSERT	[7:0]	Specifies the FSET asserting position (Unit: HDMI link PCLK, default: 83MHz, 12n)	0x4

10.3.6.41 HDCP E-FUSE Control Register (EFUSE_FSET_DE-ASSERT, R/W, Address = 0xFA16_0020)

EFUSE_FSET_DEASSERT	Bit	Description	Initial State
EFUSE_FSET_DEASSERT	[7:0]	Specifies the FSET de-asserting position (Unit: HDMI link PCLK, default: 83MHz, 12n)	0x10

10.3.6.42 HDCP E-FUSE Control Register (EFUSE_SENSING, R/W, Address = 0xFA16_0024)

EFUSE_SENSING	Bit	Description	Initial State
EFUSE_SENSING	[7:0]	Specifies the sensing width (Unit: HDMI link PCLK, default: 83MHz, 12n).	0x14

10.3.6.43 HDCP E-FUSE Control Register (EFUSE_SCK_ASSERT, R/W, Address = 0xFA16_0028)

EFUSE_SCK_ASSERT	Bit	Description	Initial State
EFUSE_SCK_ASSERT	[7:0]	Specifies the SCK asserting position (Unit: HDMI link PCLK, default: 83MHz, 12n)	0x4

10.3.6.44 HDCP E-FUSE Control Register (EFUSE_SCK_DEASSERT, R/W, Address = 0xFA16_002C)

EFUSE_SCK_DEASSERT	Bit	Description	Initial State
EFUSE_SCK_DEASSERT	[7:0]	Specifies the SCK de-asserting position (Unit: HDMI link PCLK, default: 83MHz, 12n)	0xC

10.3.6.45 HDCP E-FUSE Control Register (EFUSE_SDOUT_OFFSET, R/W, Address = 0xFA16_0030)

EFUSE_SDOUT_OFFSET	Bit	Description	Initial State
EFUSE_SDOUT_OFFSET	[7:0]	Specifies the SDOUT offset (Unit: HDMI link PCLK, default: 83MHz, 12n)	0x10

10.3.6.46 HDCP E-FUSE Control Register (EFUSE_READ_OFFSET, R/W, Address = 0xFA16_0034)

EFUSE_READ_OFFSET	Bit	Description	Initial State
EFUSE_READ_OFFSET	[7:0]	Specifies the READ Offset (Unit: HDMI link PCLK, default: 83MHz, 12n).	0x14

10.3.6.47 CEC Configure Register (CEC_TX_STATUS_0, R, Address = 0xE1B0_0000)

CEC_TX_STATUS_0	Bit	Description	Initial State
-	[7:4]	Reserved	4b0000
Tx_Error	[3]	<p>Specifies the CEC Tx_Error interrupt flag. This bit field also specifies the status of Tx_Error interrupt and is valid only if Tx_Done bit is set.</p> <p>0 = No error occurs 1 = An error occurs during CEC Tx transfer It will be cleared</p> <ul style="list-style-type: none"> - if set to 0 by Tx_Enable bit of CEC_TX_CTRL register - if set Clear_Intr_Tx_Done or Clear_Intr_Tx_Error bit in CEC_INTR_CLEAR register 	0
Tx_Done	[2]	<p>Specifies the CEC Tx_Done interrupt flag. This bit field also specifies the status of Tx_Done interrupt.</p> <p>0 = Running or idle 1 = Finishes CEC Tx transfer It will be cleared</p> <ul style="list-style-type: none"> - if Tx_Enable bit of CEC_TX_CTRL_0 is reset - if Clear_Intr_Tx_Done or Clear_Intr_Tx_Error bit in CEC_INTR_CLEAR register is set 	0
Tx_Transferring	[1]	<p>If TX-Running is set, this field is valid.</p> <p>0 = Tx waits for the CEC Bus 1 = CEC Tx transfers data via CEC Bus</p>	0
Tx_Running	[0]	<p>0 = Tx Idle 1 = Enables CEC Tx, and waits for the CEC bus or transfers the message.</p>	0

10.3.6.48 CEC Configure Register (CEC_TX_STATUS_1, R, Address = 0xE1B0_0004)

CEC_TX_STATUS_1	Bit	Description	Initial State
Tx_Bytes_Transferred	[7:0]	Specifies the number of blocks transferred (1 byte = 1 block in a CEC message). After sending the CEC message, this field will be updated. It will be cleared if Clear_Intr_Tx_Done or Clear_Intr_Tx_Error bit is set in CEC_Intr_Clear register.	0

10.3.6.49 CEC Configure Register (CEC_RX_STATUS_0, R, Address = 0xE1B0_0008)

CEC_RX_STATUS_0	Bit	Description	Initial State
-	[7:5]	Reserved	3b000
Rx_BCast	[4]	<p>Specifies the broadcast message flag. 0 = Received CEC message is the address to a single device 1 = Received CEC message is the broadcast message It will be cleared</p> <ul style="list-style-type: none"> - if Rx_Enable bit of CEC_RX_CTRL_0 is reset - if Clear_Intr_Rx_Done or Clear_Intr_Rx_Error bit in CEC_INTR_CLEAR register is set 	0
Rx_Error	[3]	<p>Specifies the CEC Rx_Error interrupt flag. This bit field also specifies the status of Rx_Error interrupt and is valid only if Rx_Done bit is set. 0 = No error occurs 1 = An error occurs while receiving a CEC message It will be cleared</p> <ul style="list-style-type: none"> - if Rx_Enable bit of CEC_RX_CTRL_0 is reset - if Clear_Intr_Rx_Done or Clear_Intr_Rx_Error bit in CEC_INTR_CLEAR register is set 	0
Rx_Done	[2]	<p>Specifies the CEC Rx done interrupt flag. This bit field also specifies the status of Rx_Done interrupt. 0 = Running or Idle 1 = Finishes CEC Rx transfer It will be cleared:</p> <ul style="list-style-type: none"> - if Rx_Enable bit of CEC_RX_CTRL_0 is reset - if Clear_Intr_Rx_Done or Clear_Intr_Rx_Error bit in CEC_INTR_CLEAR register is set 	0
Rx_Receiving	[1]	0 = Rx waits for a CEC message 1 = Rx receives data via CEC bus	0
Rx_Running	[0]	0 = Disables Rx 1 = Enables CEC Rx and waits for a message on the CEC bus	0

10.3.6.50 CEC Configure Register (CEC_RX_STATUS_1, R, Address = 0xE1B0_000C)

CEC_RX_STATUS_1	Bit	Description	Initial State
Rx_Bytes_Received	[7:0]	Specifies the number of blocks received (1 byte = 1 block in a CEC message). After receiving the CEC message, the field will be updated. It will be cleared if Clear_Intr_Rx_Done or Clear_Intr_Rx_Error bit in CEC_Intr_Clear register is set.	0

10.3.6.51 CEC Configure Register (CEC_INTR_MASK, R/W, Address = 0xE1B0_0010)

CEC_INTR_MASK	Bit	Description	Initial State
-	[7:6]	Reserved	2b00
Mask_Intr_Rx_Error	[5]	Specifies the Rx_Error interrupt mask bit. 0 = Enables 1 = Disables	0
Mask_Intr_Rx_Done	[4]	Specifies the Rx_Done interrupt mask bit. 0 = Enables 1 = Disables	0
-	[3:2]	Reserved	2b00
Mask_Intr_Tx_Error	[1]	Specifies the Tx_Error interrupt mask bit. 0 = Enables 1 = Disables	0
Mask_Intr_Tx_Done	[0]	Specifies the Tx_Done interrupt mask bit. 0 = Enables 1 = Disables	0

10.3.6.52 CEC Configure Register (CEC_INTR_CLEAR, R/W, Address = 0xE1B0_0014)

CEC_INTR_CLEAR	Bit	Description	Initial State
-	[7:6]	Reserved	2b00
Clear_Intr_Rx_Error	[5]	Specifies the Rx_Error interrupt clear bit. 0 = No effect 1 = Clears Rx_Error and Rx_Bytes_Received fields in CEC_RX_STATUS_0 and 1 registers. It will be cleared after one clock.	0
Clear_Intr_Rx_Done	[4]	Specifies the Rx_Done interrupt clear bit. 0 = No effect 1 = Clears Rx_Done and Rx_Bytes_Received fields in CEC_RX_STATUS_0 and 1 registers. Resets to 0 after one clock.	0
-	[3:2]	Reserved	2b00
Clear_Intr_Tx_Error	[1]	Specifies the Tx_Error interrupt clear bit. 0 = No effect 1 = Clears Tx_Error and Tx_Bytes_Received fields in CEC_TX_STATUS_0 and 1 registers. Resets to 0 after one clock.	0
Clear_Intr_Tx_Done	[0]	Specifies the Tx_Done interrupt clear bit. 0 = No effect 1 = Clears Tx_Done and Tx_Bytes_Received fields in CEC_TX_STATUS_0 and 1 registers. Resets to 0 after one clock.	0

10.3.6.53 CEC Configure Register (CEC_LOGIC_ADDR, R/W, Address = 0xE1B0_0020)

CEC_LOGIC_ADDR	Bit	Description	Initial State
-	[7:4]	Reserved	4b0000
Logic_Addr	[3:0]	Specifies the HDMI Tx logical address (0~15).	4b0000

10.3.6.54 CEC Configure Register (CEC_DIVISOR_0 ~ CEC_DIVISOR_3, R/W, Address = 0xE1B0_0030)

CEC_DIVISOR_0~CEC_DIVISOR_3	Bit	Description	Initial State
CEC_Divisor	[7:0]	Specifies the divisor used in counting 0.05ms period. This divisor should satisfy the following equation: $(CEC_DIVISOR+1) \times (\text{clock cycle time(ns)}) = 0.05\text{ms}$ Note: To apply CEC_Divisor, it should be '0' for Tx_Reset and Rx_Reset, while Tx_Start and Rx_Start are '0'.	0

10.3.6.55 Tx Related Register (CEC_TX_CTRL, R/W, Address = 0xE1B0_0040)

CEC_TX_CTRL	Bit	Description	Initial State
Reset	[7]	Specifies the CEC Tx reset bit. 0 = No effect 1 = Immediately resets CEC Tx related registers and state machines to its reset value. Resets to 0 after one clock.	0
Tx_Retrans_Num	[6:4]	Specifies the number of retransmissions tried (according to CEC specification on page CEC-13). Based on the specification, this value should be set to 5.	3b001
-	[3:2]	Reserved	2b00
Tx_BCast	[1]	Specifies the CEC Tx broadcast message bit. This bit also specifies the CEC message in CEC_TX_BUFFER_00~15, which is directly addressed (addressed to a single device) or broadcast. This bit determines whether a block transfer is acknowledged or not (according to ACK scheme in CEC specification (section CEC 6.1.2)) 0 = Directly addressed message 1 = Broadcast message	0
Tx_Start	[0]	Specifies the CEC Tx start bit. 0 = Tx idle 1 = Starts CEC message transfer (Resets to 0 after start)	0

10.3.6.56 Tx Related Register (CEC_TX_BYTE_NUM, R/W, Address = 0xE1B0_0044)

CEC_TX_BYTE_NUM	Bit	Description	Initial State
Tx_Byte_Num	[7:0]	Specifies the number of blocks in a message that has to be sent (1 byte = 1 block in a CEC message).	0

10.3.6.57 Tx Related Register (CEC_TX_STATUS_2, R, Address = 0xE1B0_0060)

CEC_TX_STATUS_2	Bit	Description	Initial State
Tx_Wait	[7]	Specifies the CEC Tx signal free time waiting flag bit. 0 = Tx is in other state 1 = CEC Tx waits for signal free time (stops sending messages after earlier attempts to send message).	0
Tx_Sending_Start_Bit	[6]	Specifies the CEC Tx start bit sending flag bit. 0 = Tx is in other state 1 = CEC Tx sends a start bit	0
Tx_Sending_Hdr_Blk	[5]	Specifies the CEC Tx header block sending flag bit. 0 = Tx is in other state 1 = CEC Tx sends the header block	0
Tx_Sending_Data_Blk	[4]	Specifies the CEC Tx data block sending flag bit. 0 = Tx is in other state 1 = CEC Tx sends data blocks	0
Tx_Latest_Initiator	[3]	Specifies the CEC Tx last initiator flag bit. 0 = This device is not the latest initiator on the CEC bus 1 = This CEC device is the latest initiator to send a CEC message; no other CEC device sends a message. It will be cleared if Rx detects a start bit on the CEC line or sets Tx_Enable bit of CEC_Tx_Ctrl_0 (that is becomes a new initiator)	0
-	[2:0]	Reserved	3b000

10.3.6.58 Tx Related Register (CEC_TX_STATUS_3, R, Address = 0xE1B0_0064)

CEC_TX_STATUS_3	Bit	Description	Initial State
Reserved	[7]	Reserved	0
Tx_Wait_SFT_Succ	[6]	Specifies the CEC Tx signal free time for successive message transfer waiting flag bit. 0 = Tx is in other state 1 = Tx waits for signal free time (SFT) with a precondition that Tx is the most recent initiator on the CEC bus and wants to send another frame immediately after its previous frame (SFT \geq 7x2.4ms).	0
Tx_Wait_SFT_New	[5]	Specifies the CEC Tx signal free time for a new initiator waiting flag bit. 0 = Tx is in other state 1 = Tx waits for SFT with a precondition that Tx is the new initiator and wants to send a frame (SFT \geq 5x2.4ms).	0
Tx_Wait_SFT_Retrans	[4]	Specifies the CEC Tx signal free time for a new initiator waiting flag bit. 0 = Tx is in other state 1 = Tx waits for SFT with a precondition (the precondition is that Tx should attempt to retransmit the message (SFT \geq 3 x2.4ms))	0
Tx_Retrans_Cnt	[3:1]	Specifies the current retransmission count. If '0', no retransmission occurs. It will be cleared if Clear_Intr_Tx_Done or Clear_Intr_Tx_Error bit in CEC_Intr_Clear register is set.	3b000
Tx_ACK_Failed	[0]	Specifies the CEC Tx acknowledge failed flag bit. 0 = Tx is in the other state 1 = Tx is not acknowledged. This bit is set if - ACK bit in a block is logical 1 in a directly addressed message - ACK bit in a block is logical 0 in a broadcast message	0

10.3.6.59 Tx Related Register (CEC_TX_BUFFER_0 ~ 15)

- CEC_TX_BUFFER_0, R/W, Address = 0xE1B0_0080
- CEC_TX_BUFFER_1, R/W, Address = 0xE1B0_0084
- CEC_TX_BUFFER_2, R/W, Address = 0xE1B0_0088
- CEC_TX_BUFFER_3, R/W, Address = 0xE1B0_008C
- CEC_TX_BUFFER_4, R/W, Address = 0xE1B0_0090
- CEC_TX_BUFFER_5, R/W, Address = 0xE1B0_0094
- CEC_TX_BUFFER_6, R/W, Address = 0xE1B0_0098
- CEC_TX_BUFFER_7, R/W, Address = 0xE1B0_009C
- CEC_TX_BUFFER_8, R/W, Address = 0xE1B0_00A0
- CEC_TX_BUFFER_9, R/W, Address = 0xE1B0_00A4
- CEC_TX_BUFFER_10, R/W, Address = 0xE1B0_00A8
- CEC_TX_BUFFER_11, R/W, Address = 0xE1B0_00AC
- CEC_TX_BUFFER_12, R/W, Address = 0xE1B0_00B0
- CEC_TX_BUFFER_13, R/W, Address = 0xE1B0_00B4
- CEC_TX_BUFFER_14, R/W, Address = 0xE1B0_00B8
- CEC_TX_BUFFER_15, R/W, Address = 0xE1B0_00BC

CEC_TX_BUFFER_0~ CEC_TX_BUFFER_15	Bit	Description	Initial State
Tx_Block_0 ~ Tx_Block_15	[7:0]	Specifies the byte #0 ~ #15 of CEC message. Each byte corresponds to a block in a message. Block_0 is the header block and block_1 ~15 are data blocks. Note: The initiator and destination logical address in a header block should be written by software.	0

10.3.6.60 Rx Related Register (CEC_RX_CTRL, R/W, Address = E1B0_00C0)

CEC_RX_CTRL_0	Bit	Description	Initial State
Reset	[7]	Specifies the CEC Rx reset bit. 0 = No effect 1 = Immediately resets CEC Rx related registers and state machines to its reset value. It will be cleared after one clock	0
Check_Sampling_Error	[6]	Specifies the CEC Rx sampling error check enable bit. 0 = Does not check sampling error 1 = Checks sampling error while receiving data bits CEC Rx samples the CEC bus three times (at 1.00, 1.05, and 1.10 ms) and checks whether the three samples are identical.	0
Check_Low_Time_Error	[5]	Specifies the CEC Rx low-time error check enable bit. 0 = Does not check low-time error 1 = Checks low-time error while receiving data bits In receiving each bit from the CEC bus, CEC Rx checks the duration of logical 0 from the starting of one bit transfer (falling edge on the CEC bus). Rx checks whether the duration is longer than the maximum time the CEC bus can be in logical 0 (max 1.7 ms).	0
Check_Start_Bit_Error	[4]	Specifies the CEC Rx start bit error check enable bit 0 = Does not check start bit error. 1 = Checks start bit error while receiving a start bit. After receiving a start bit from the CEC bus, CEC Rx checks the duration of logical 0 and 1 of start bit (as specified in CEC specification on page CEC-8). Rx checks whether the duration meets the specification.	0
-	[3:2]	Reserved	2b00
Rx_Host_Busy	[1]	Specifies the CEC Rx host busy bit. 0 = Rx receives incoming message and sends acknowledgement. 1 = A host processor is unavailable to receive and process CEC messages. Rx sends “not acknowledged” signal to a message initiator to indicate that a host processor is unavailable and to receive and process CEC messages.	0
Rx_Enable	[0]	Specifies the CEC Rx start bit. 0 = Disables Rx 1 = Enables CEC Rx module to receive a message This bit is cleared after receiving a message.	0

10.3.6.61 Rx Related Register (CEC_RX_STATUS_2, R, Address = 0xE1B0_00E0)

CEC_RX_STATUS_2	Bit	Description	Initial State
Rx_Waiting	[7]	Specifies the CEC Rx waiting flag bit. 0 = Rx is in other state 1 = CEC Rx waits for a message	0
Rx_Receiving_Start_Bit	[6]	Specifies the CEC Rx start bit receiving flag bit. 0 = Rx is in other state 1 = CEC Rx receives a start bit	0
Rx_Receiving_Hdr_Blk	[5]	Specifies the CEC Rx header block receiving flag bit. 0 = Rx is in other state 1 = CEC Rx receives a header block	0
Rx_Receiving_Data_Blk	[4]	Specifies the CEC Rx data block receiving flag bit. 0 = Rx is in other state 1 = CEC Rx receives data blocks	0
-	[3:0]	Reserved	4b0000

10.3.6.62 Rx Related Register (CEC_RX_STATUS_3, R, Address = 0xE1B0_00E4)

CEC_RX_STATUS_3	Bit	Description	Initial State
-	[7]	Reserved	0
Sampling_Error	[6]	<p>Specifies the CEC Rx sampling error flag bit.</p> <p>0 = No sampling error occurs 1 = A sampling error occurs while receiving a message</p> <p>CEC Rx samples the CEC bus three times (at 1.00, 1.05, and 1.10 ms) and sets this bit if Check_Sampling_Error bit in CEC_RX_CTRL_0 is set and if three samples are not identical.</p> <p>It will be cleared if Clear_Intr_Rx_Done or Clear_Intr_Rx_Error bit in CEC_INTR_CLEAR register is set to 0.</p>	0
Low_Time_Error	[5]	<p>Specifies the CEC Rx low-time error flag bit.</p> <p>0 = No low-time error occurs 1 = A low-time error occurs while receiving a message</p> <p>While receiving each bit from the CEC bus, CEC Rx checks the duration of logical 0 from the start of one-bit transfer (falling edge on the CEC bus). If the duration is longer than the maximum time, the CEC bus can be logical 0 (maximum 1.7 ms). CEC RX sets this bit.</p> <p>This bit field will be set to 0 if Clear_Intr_Rx_Done or Clear_Intr_Rx_Error bit in CEC_INTR_CLEAR register is set.</p>	0
Start_Bit_Error	[4]	<p>Specifies the CEC Rx start bit error flag bit.</p> <p>0 = No start bit error occurs 1 = A start bit error occurs while receiving a message</p> <p>While receiving a start bit from the CEC bus, CEC Rx checks the duration of logical 0 and 1 of a starting bit (as specified in CEC spec. page CEC-8). If the duration does not meet the spec., CEC RX sets this bit.</p> <p>This bit field will be set to 0 if Clear_Intr_Rx_Done or Clear_Intr_Rx_Error bit in CEC_INTR_CLEAR register is set.</p>	0
-	[3:1]	Reserved	3b000
CEC_Line_Error	[0]	<p>Specifies the CEC Rx line error flag bit.</p> <p>0 = No line error occurs 1 = A start bit error line occurs while receiving a message</p> <p>In CEC spec. page CEC-13, CEC line error occurs in a period when two consecutive falling edges is smaller than a minimum data bit period. Rx checks for this condition, and if it occurs, it sends the line error notification, that is, sends logical 0 for more than 1.4~1.6 times of the nominal data bit period (2.4ms).</p> <p>This bit will be cleared:</p> <ul style="list-style-type: none"> - if Rx_Enable bit of CEC_RX_CTRL_0 is set - if Clear_Intr_Rx_Done or Clear_Intr_Rx_Error bit in 	0

CEC_RX_STATUS_3	Bit	Description	Initial State
		CEC_INTR_CLEAR register is set	

10.3.6.63 Rx Related Register (CEC_RX_BUFFER_0 ~ CEC_RX_BUFFER_15)

- CEC_RX_BUFFER_0, R, Address = 0xE1B0_0100
- CEC_RX_BUFFER_1, R, Address = 0xE1B0_0104
- CEC_RX_BUFFER_2, R, Address = 0xE1B0_0108
- CEC_RX_BUFFER_3, R, Address = 0xE1B0_010C
- CEC_RX_BUFFER_4, R, Address = 0xE1B0_0110
- CEC_RX_BUFFER_5, R, Address = 0xE1B0_0114
- CEC_RX_BUFFER_6, R, Address = 0xE1B0_0118
- CEC_RX_BUFFER_7, R, Address = 0xE1B0_011C
- CEC_RX_BUFFER_8, R, Address = 0xE1B0_0120
- CEC_RX_BUFFER_9, R, Address = 0xE1B0_0124
- CEC_RX_BUFFER_10, R, Address = 0xE1B0_0128
- CEC_RX_BUFFER_11, R, Address = 0xE1B0_012C
- CEC_RX_BUFFER_12, R, Address = 0xE1B0_0130
- CEC_RX_BUFFER_13, R, Address = 0xE1B0_0134
- CEC_RX_BUFFER_14, R, Address = 0xE1B0_0138
- CEC_RX_BUFFER_15, R, Address = 0xE1B0_013C

CEC_RX_BUFFER_0~ CEC_RX_BUFFER_15	Bit	Description	Initial State
Rx_Block_0 ~ Rx_Block_15	[7:0]	Specifies byte #0 ~ #15 of CEC message. Each byte corresponds to a block in a message. Block_0 is the header block and Block_1 ~15 are data blocks.	0

10.3.6.64 Input Filtering Register (CEC_FILTER_CTRL, R/W, Address = 0xE1B0_0180)

CEC_FILTER_CTRL	Bit	Description	Initial State
Filter_Cur_Val	[7]	Specifies the CEC filter current value bit. Indicates current value fed to CEC Tx, Rx. If the filter is enabled, this bit specifies the latest value on the CEC bus that is stable for more than Filter_Th cycles.	1
-	[6:1]	Reserved	6b000000
Filter_Enable	[0]	Specifies the CEC filter enable bit. 0 = Disables filter. Directly passes CEC input to CEC Tx, Rx. 1 = Enables filter. Filter propagates signals stable for more Filter_Th cycles.	1

10.3.6.65 Input Filtering Register (CEC_FILTER_TH, R/W, Address = 0xE1B0_0184)

CEC_FILTER_TH	Bit	Description	Initial State
Filter_Th	[7:0]	Specifies the filter threshold value. If the filter is enabled, it filters out signals that are less stable than Filter_Th cycles.	8b00000011

11 IMAGE ROTATOR

11.1 OVERVIEW OF IMAGE ROTATOR

Image Rotator performs rotating/flipping image data. It is composed of Rotate FSM, Rotate Buffer, AMBA 3.0 AXI master and APB slave interface, and Register files. Overall features are summarized as follows.

11.2 KEY FEATURES OF IMAGE ROTATOR

The features of image rotator include:

- Image format: YCbCr 4:2:2(interleave), YCbCr 4:2:0(non-interleave, 2-plane and 3-plane), RGB565 and RGB888 (unpacked)
- Rotate degree: 0, 90, 180, and 270 with flip vertical and flip horizontal
- Windows offset function
- Image size: up to 64K by 64K (The maximum sizes are different from the types of image format)
- Image size restriction: memory size shouldn't exceed 16-bit address size. For example, RGB888 has a size limitation up to 14-bit image size, only [13:0] is valid and [15:14] are ignored.

11.3 BLOCK DIAGRAM OF IMAGE ROTATOR

The [Figure 11-1](#) shows the block diagram of Image Rotator.

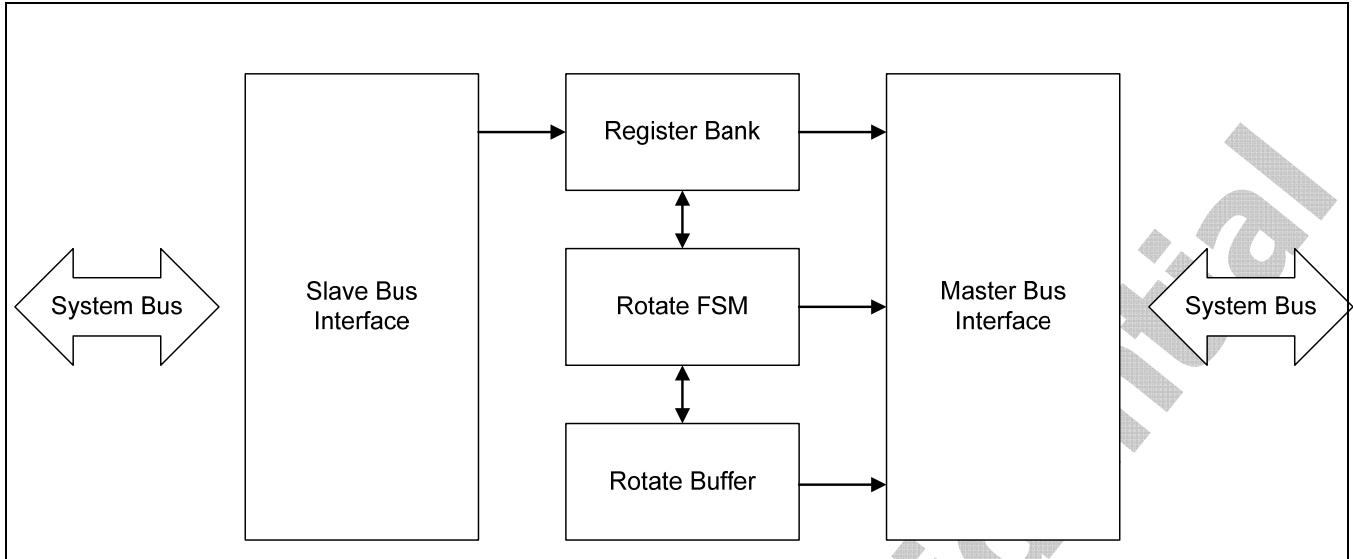


Figure 11-1 Image Rotator Block Diagram

11.4 SUPPORTED IMAGE ROTATION FUNCTIONS

The [Figure 11-2](#) shows the rotation functions supported by Image Rotator.



Figure 11-2 Ported Image Rotation Functions

11.5 IMAGE ROTATION WITH WINDOWS OFFSET

Image rotator supports image rotation with window offset function. It is useful function to move from a small portion of a large image, to move in a portion of a large image.

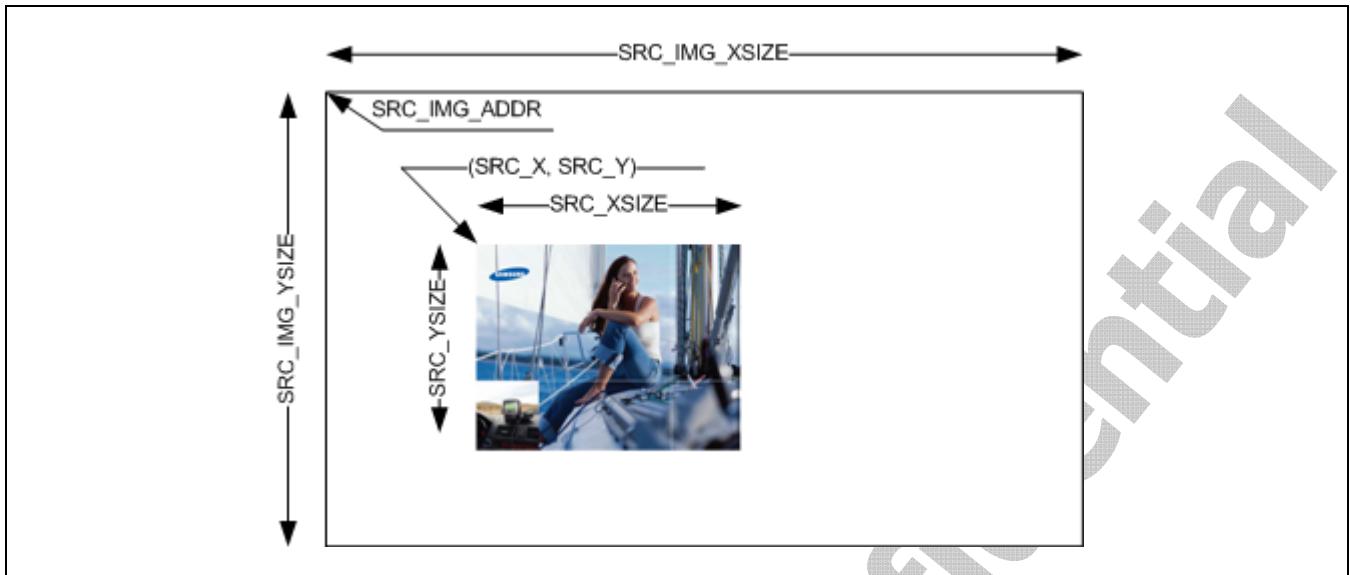


Figure 11-3 Source Image Example (with window offset function)

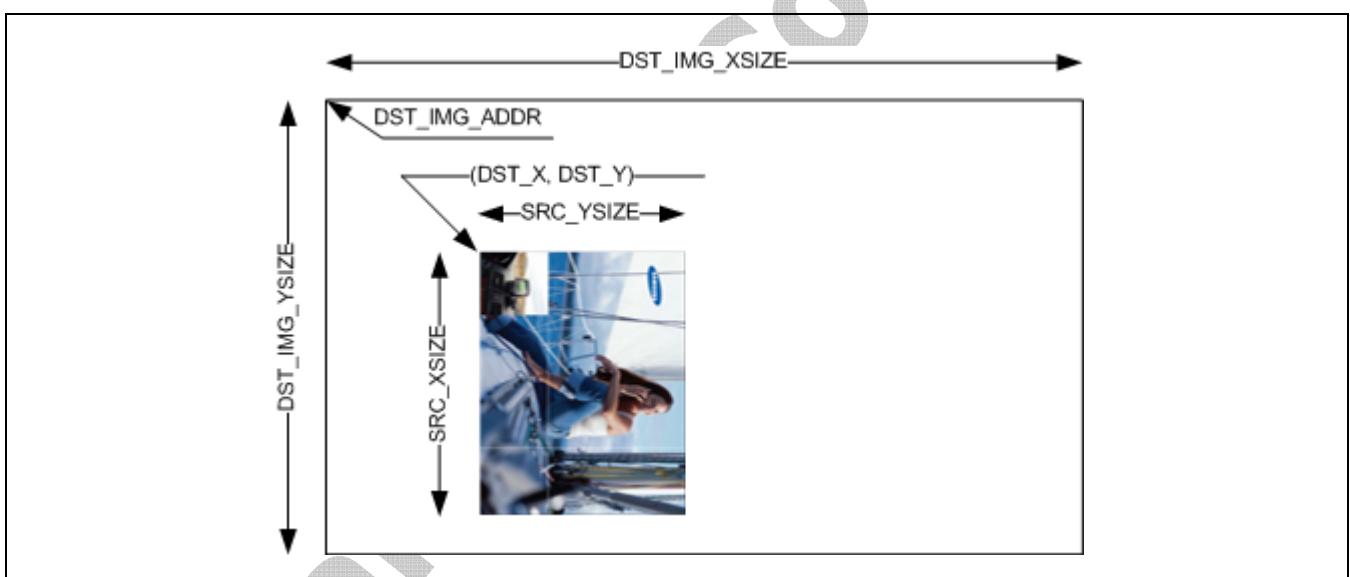


Figure 11-4 Destination Image Example (90 degree rotated with window offset function)

11.6 PROGRAMMING GUIDE

11.6.1 REGISTER SETTING

All registers with CONTROL register except CONTROL [0] should set for proper operation previously, and User should set CONTROL [0] bit for starting image rotator.

11.6.2 RESTRICTIONS ON THE IMAGE SIZE

Image rotator has some restrictions on the image size. User should not violate these restrictions.

- Image base address:
The bit [2:0] should be zero for the SRCADDRREG0/1/2 and DSTADDRREG0/1/2 registers.
- Image size: SRCIMGSIZE and DSTIMGSIZE should be set as follows,

Image Format	Minimum Size	Maximum Size
RGB888	8 x 8	16K x 16K
RGB565	16 x 16	32K x 32K
YCbCr422	16 x 16	32K x 32K
YCbCr420 2-Plane	32 x 32	64K x 64K (in case of Y components)
YCbCr420 3-Plane	64 x 32	64K x 64K (in case of Y components)

- Image coordinates to be rotated:
SRC_XY and DST_XY should set as follows,

Image Formats	Image Size Restrictions
RGB888	X and Y pixel size should be multiple of 2.
RGB565	X and Y pixel size should be multiple of 4.
YCbCr422	X and Y pixel size should be multiple of 4.
YCbCr420 2-Plane	X and Y pixel size should be multiple of 8.
YCbCr420 3-Plane	X and Y pixel size should be multiple of 16.

11.7 REGISTER DESCRIPTION

11.7.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
CONFIG	0xFA30_0000	R/W	Rotator Configuration	0x0000_0000
CONTROL	0xFA30_0010	R/W	Rotator Image0 Control	0x0000_0000
STATUS	0xFA30_0020	R	Rotator Status	0x0000_0000
SRCBASEADDR0	0xFA30_0030	R/W	Rotator Source Image Base Address	0x0000_0000
SRCBASEADDR1	0xFA30_0034	R/W	Rotator Source Image Base Address	0x0000_0000
SRCBASEADDR2	0xFA30_0038	R/W	Rotator Source Image Base Address	0x0000_0000
SRCTIMGSIZE	0xFA30_003C	R/W	Rotator Source Image X, Y Size	0x0000_0000
SRC_XY	0xFA30_0040	R/W	Rotator Source Image X, Y Coordinates	0x0000_0000
SRCCROTSIZE	0xFA30_0044	R/W	Rotator Source Image Rotation Size	0x0000_0000
DSTBASEADDR0	0xFA30_0050	R/W	Rotator Destination Image Base Address	0x0000_0000
DSTBASEADDR1	0xFA30_0054	R/W	Rotator Destination Image Base Address	0x0000_0000
DSTBASEADDR2	0xFA30_0058	R/W	Rotator Destination Image Base Address	0x0000_0000
DSTIMGSIZE	0xFA30_005C	R/W	Rotator Destination Image X, Y Size	0x0000_0000
DST_XY	0xFA30_0060	R/W	Rotator Destination Image X, Y Coordinates	0x0000_0000

11.7.1.1 Rotator Configuration Register (CONFIG, R/W, Address= 0xFA30_0000)

CONFIG	Bit	Description	Initial State
Reserved	[31:9]	Reserved	000_0000b
Enable Interrupt	[9]	Interrupt enable to indicate illegally configured setting 0 = Disables interrupt 1 = Enables interrupt Note: In this case, rotator doesn't work.	1b
Enable Interrupt	[8]	Interrupt enable to indicate that a image rotation is finished 0 = Disables interrupt 1 = Enables interrupt	0b
Reserved	[7:0]	Reserved	0x00

11.7.1.2 Rotator Control Register (CONTROL, R/W, Address= 0xFA30_0010)

CONTROL	Bit	Description	Initial State
Reserved	[31:11]	Reserved	0x0000
Pattern Writing	[16]	Write pattern to fill a destination image with a designated pattern 0 = Disable pattern writing 1 = Enable pattern writing Note: if this bit set, the information of a source image is ignored.	0b
Reserved	[15:11]	Reserved	00000b
Input Image Format	[10:8]	Input image format to be rotated 000 = YCbCr 4:2:0(3-plane) 001 = YCbCr 4:2:0(2-plane) 010 = Reserved 011 = YCbCr 4:2:2 (interleave) 100 = RGB565 110 = RGB888 (Unpacked) 111 = Reserved	000b
Flip Direction	[7:6]	Flip direction 0x = No flip 10 = Flip vertical 11 = Flip horizontal	00b
Rotation Degree	[5:4]	Rotation degree 00 = 0 degree 01 = 90 degree 10 = 180 degree 11 = 270 degree	00b
Reserved	[3:1]	Reserved	000b
Start Rotate	[0]	Rotate enable signal. Whenever this bit set, Rotator starts the operation. This bit is cleared if rotator starts to move an image. 1 = Start rotate operation	0b

11.7.1.3 Rotator Status Register (STATAUS, R, Address = 0xFA30_0020)

STATREG	Bit	Description	Initial State
Reserved	[31:10]	Reserved	0x00
Interrupt Pending	[9]	This bit is set if the SFR set illegally. Writing '1' makes this bit clear.	1b
Interrupt Pending	[8]	This bit is set if an image rotation is complete. Writing '1' makes this bit clear.	0b
Reserved	[3:1]	Reserved	0x00
Rotator status	[1:0]	These bits show the rotator operation status. 00 = IDLE status 01 = Reserved 10 = Rotating a image (BUSY) 11 = Rotating a image, and has one more job to rotate (BUSY)	00b

11.7.1.4 Rotator Source Image Base Address Register 0 (SRCBASEADDR0, R/W, Address = 0xFA30_0030)

SRCADDRREG0	Bit	Description	Initial State
SRC_IMG_ADDR	[31:0]	Base address of source image for RGB or Y component	0x0000_0000

11.7.1.5 Rotator Source Image Base Address Register 1 (SRCBASEADDR1, R/W, Address = 0xFA30_0034)

SRCADDRREG1	Bit	Description	Initial State
SRC_IMG_ADDR	[31:0]	Base Address of source image for Cb component.	0x0000_0000

11.7.1.6 Rotator Source Image Base Address Register 2 (SRCBASEADDR2, R/W, Address = 0xFA30_0038)

SRCADDRREG2	Bit	Description	Initial State
SRC_IMG_ADDR	[31:0]	Base Address of source image for Cr component.	0x0000_0000

11.7.1.7 Rotator Source Base Image Size Register (SRCIMGSIZE, R/W, Address = 0xFA30_003C)

SRCIMGSIZE	Bit	Description	Initial State
SRC_YSIZE	[31:16]	Vertical pixel size of a source image	0x0000
SRC_XSIZE	[15:0]	Horizontal pixel size of a source image	0x0000

11.7.1.8 Rotator Source Image Coordinates Register (SRC_XY, R/W, Address = 0xFA30_0040)

SRC_XY	Bit	Description	Initial State
SRC_Y	[31:16]	The pixel coordinates on Y-axis of a image to be rotated	0x0000
SRC_X	[15:0]	The pixel coordinates on X-axis of a image to be rotated	0x0000

11.7.1.9 Rotator Source Base Rotation Size Register (SRCROTSIZE, R/W, Address = 0xFA30_0044)

SRCROTSIZE	Bit	Description	Initial State
SRC_YSIZE	[31:16]	Vertical pixel size of a image to be rotated	0x0000
SRC_XSIZE	[15:0]	Horizontal pixel size of a image to be rotated	0x0000

11.7.1.10 Rotator Destination Image Base Address Register 0 (DSTBASEADDR0, R/W, Address = 0xFA30_0050)

DSTBASEADDR0	Bit	Description	Initial State
DST_IMG_ADDR	[31:0]	Address of destination image for RGB or Y component.	0x0000_0000

11.7.1.11 Rotator Destination Image Base Address Register 1 (DSTBASEADDR1, R/W, Address = 0xFA30_0054)

DSTBASEADDR1	Bit	Description	Initial State
DST_IMG_ADDR	[31:0]	Address of destination image for CB component.	0x0000_0000

11.7.1.12 Rotator Destination Image Base Address Register 2 (DSTBASEADDR2, R/W, Address = 0xFA30_0058)

DSTBASEADDR2	Bit	Description	Initial State
DST_IMG_ADDR	[31:0]	Address of destination image for Cr component.	0x0000_0000

11.7.1.13 Rotator Destination Base Image Size Register (DSTIMGSIZE, R/W, Address = 0xFA30_005C)

DSTIMGSIZE	Bit	Description	Initial State
DST_YSIZE	[31:16]	Vertical pixel size of a target image	0x0000
DST_XSIZE	[15:0]	Horizontal pixel size of a target image	0x0000

11.7.1.14 Rotator Destination Image Coordinates Register (DST_XY, R/W, Address = 0xFA30_0060)

DST_XY	Bit	Description	Initial State
DST_Y	[31:16]	The pixel coordinates on Y-axis of a image to be rotated	0x0000
DST_X	[15:0]	The pixel coordinates on X-axis of a image to be rotated	0x0000

12 JPEG

12.1 OVERVIEW OF JPEG CODEC

JPEG codec compresses the original raw image and decompress the JPEG encoded image. It performs all functions required for image compression/ decompression such as Discrete Cosine Transform (DCT), Quantization and Huffman coding. It comprises of control circuit, DCT/Quantization, Huffman codec, marker process block and AHB interface control as shown in [Figure 12-1](#). It is possible to set the operation modes and conditions such as the Huffman table number and restart interval value into internal control registers.

12.2 KEY FEATURES OF JPEG CODEC

- Compression/ decompression up to 8192x8192
- Minimum Image size for compression/decompression is 32x32
- Supports following format of compression (Refer to figure 9.13-1)
 - Input raw image: YCbCr4:2:2 or RGB565
 - Output JPEG file: Baseline JPEG of YCbCr4:2:2 or YCbCr4:2:0
 - Supports following format of decompression
 - Input JPEG file: Baseline JPEG of YCbCr4:4:4, YCbCr4:2:2, YCbCr4:2:0, gray
 - Output raw image: YCbCr4:2:2 or YCbCr4:2:0
- Supports general-purpose color-space converter
- Support Baseline JPEG. (Progressive mode does not operate.)

12.3 BLOCK DIAGRAM OF JPEG CODEC

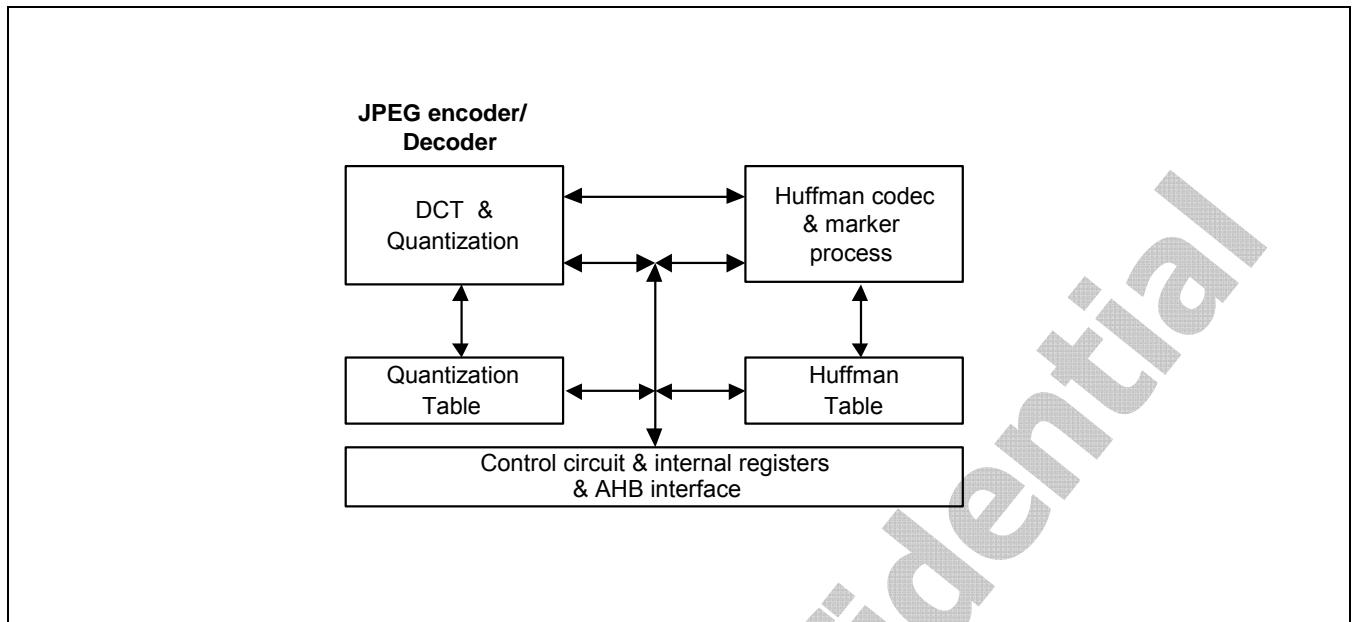


Figure 12-1 JPEG

12.4 BLOCK DIAGRAM IN/OUT DATA FORMAT

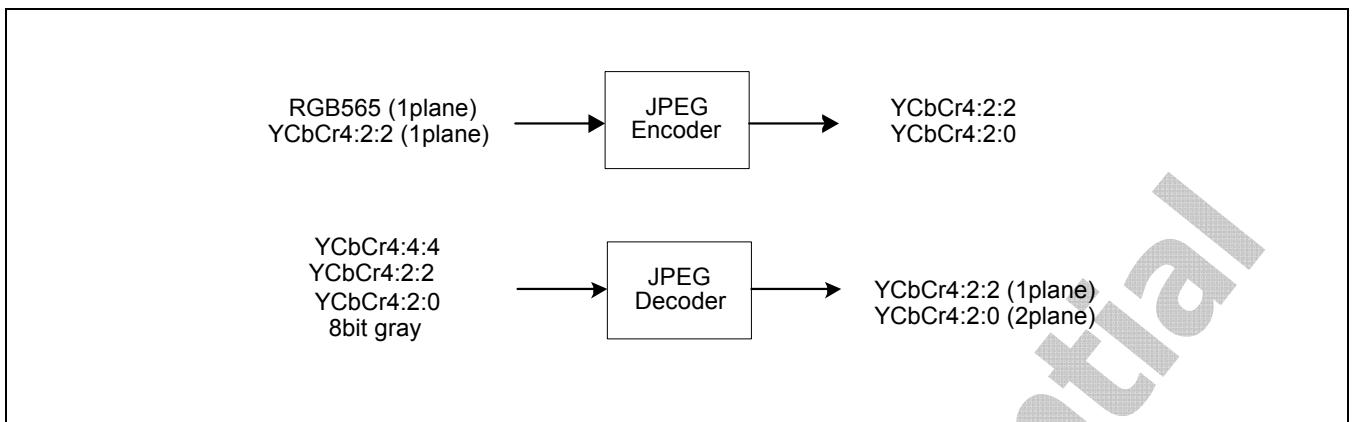


Figure 12-2 JPEG In/Output Data Format

12.4.1 CONTROL CIRCUIT AND AHB INTERFACE

This block sets and initializes the operation mode with internal registers. Use this register to set the operation modes and conditions such as Huffman table number and restart interval value.

12.4.2 DCT/ QUANTIZATION

During compression, it transforms 8x8 image data to DCT coefficients. Then the quantization process is performed over DCT coefficients by utilizing the quantization tables. During decompression, dequantization is done and then a DCT coefficient is transformed into image data.

12.4.3 HUFFMAN CODER AND MARKER PROCESS

During compression, Huffman encoding is performed based on the Huffman table and marker process generates the JPEG bit stream. During decompression, marker process parses a JPEG file and Huffman decoding is done.

12.4.4 QUANTIZATION TABLE

It is the place to store quantization tables.

12.4.5 HUFFMAN TABLE

It is the place to store Huffman tables.

12.4.6 PERFORMANCE

JPEG IP supports compression/ decompression of image file with size up to 8192x8192. Supported minimum size is 32x32.

12.5 DESCRIPTION OF SUPPORTED COLOR FORMAT

JPEG supports several color formats during compression/ decompression.

12.5.1 IN COMPRESSION MODE

Before compression starts, raw image data must be in main memory. The raw image data address is specified in IMGADR register. The raw images are stored in interleaved YCbCr4:2:2 or RGB565 color format as shown in [Figure 12-3](#). After compression is complete, result file is baseline JPEG in YCbCr4:2:0 or YCbCr4:2:2 format with interleaved scan. Therefore color space conversion (RGB→YCbCr) and decimation of chrominance component is necessary. JPEG IP has its own color space converter. COEF1, COEF2 and COEF3 register sets the coefficients of color space converter. Decimation of JPEG IP is downsampling process. For example, decimation from YCbCr4:2:2 to YCbCr4:2:0 needs 2:1 vertical downsampling for Cb and Cr component as shown in [Figure 12-4](#).

12.5.2 IN DECOMPRESSION MODE

In decompression mode, input file is baseline JPEG in YCbCr4:4:4, YCbCr4:2:2, YCbCr4:2:0, gray with interleaved scan and output raw image has interleaved YCbCr4:2:2 or YCbCr4:2:0 formats. Therefore for input file with YCbCr format, decimation and interpolation process is done during decompression. In this case, decimation is same as downsampling and interpolation is sample-and-hold (repetition of recent value) process. Each operation is described in [Figure 12-3](#). The result of gray format JPEG file input is YCbCr4:2:2 or YCbCr4:2:0 raw images. This raw images have Y component, which is result of decompression, and Cb and Cr component with “128” value.

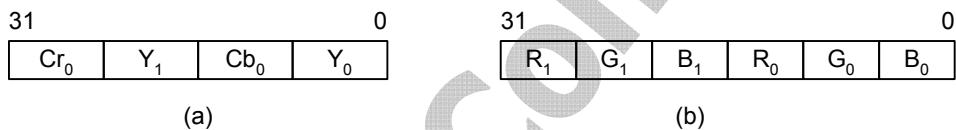


Figure 12-3 Raw Image Format in Memory (a) YCbCr4:2:2 (b) RGB5:6:5

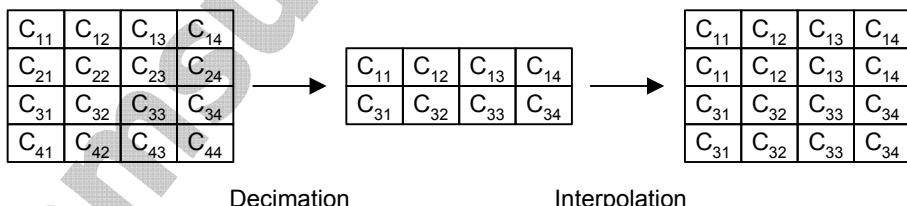


Figure 12-4 Decimation and 1:2 Interpolation in Vertical Direction

[Figure 12-5](#) illustrates the input format of YCbCr4:2:2. This JPEG codec supports type (a).

If input YCbCr4:2:2 format is type (b), decoder does not work.

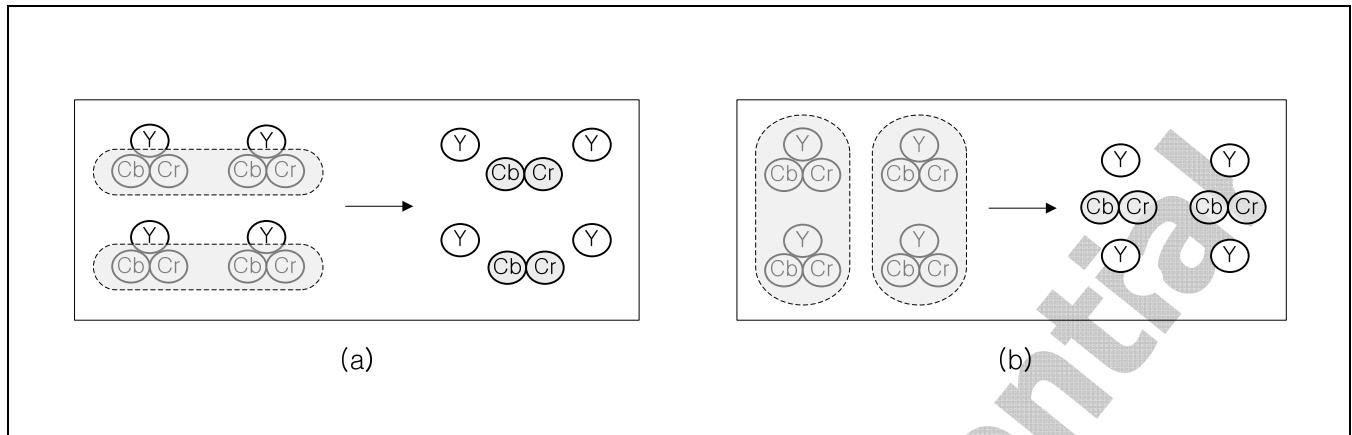


Figure 12-5 YCbCr4:2:2 Color Format

12.6 PROCESS

12.6.1 REGISTER ACCESS

The registers are modified:

After reset, until a new job starts, or

After the generation of the process completion interruption signal, until a new job starts.

Other conditions indicate that the core is in the normal operation, thus, register modification is not allowed.

12.6.2 TABLE ACCESS

Four Huffman tables (AC & DC, 2 tables for each) and four quantization tables must be configured before compression. To set any quantization table and Huffman table, first access the corresponding table entry register. Then write transfers should follow. To understand the write transfer, refer to [Figure 12-6](#). The access order for each table is shown below.

1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64

Figure 12-6 Access Order in Quantizer Table

12.6.3 STARTING PROCESS

Process start signal from the scheduler instructs to start compression or decompression process of one picture after setting various registers. After getting the start signal, the core processing starts and then JPGOPR register is read as 1. Operation cannot be guaranteed if the start signal is issued again during processing.

Table 12-1 Registers that Must be Configured Before Compression

Registers	Description	At comp	At Decomp
JPGMOD	Process Mode Register	Essential	Essential
QTBL	Quantization Table Number Register	Essential	--
JPGDRI_U	Reset Interval Registers (upper 8bit)	Essential	--
JPGDRI_L	Reset Interval Registers (lower 8bit)	Essential	
JPGY_U	Vertical Size Register (upper 8bit)	Essential	
JPGY_L	Vertical Size Register (lower 8bit)	Essential	
JPGX_U	Horizontal Size Register (upper 8bit)	Essential	
JPGX_L	Horizontal Size Register (lower 8bit)	Essential	
QTBL0	Quantizer Table0 Entries Register.	Essential	--
QTBL1	Quantizer Table1 Entry Register	Essential	--
QTBL2	Quantizer Table2 Entry Register	Essential	--
QTBL3	Quantizer Table3 Entry Register	Essential	--
OUTFORM	Output Color Format of Decompression	--	Essential
ENC_STREAM_INTSE	Compressed Stream Size Interrupt Setting Register	Essential	--
HDTBL0, HDCTBLG0	DC Huffman Table0 Entry Register	Essential	--
HACTBL0, HACTBLG0	AC Huffman Table0 Entry Register	Essential	--
HDCTBL1, HDCTBLG1	DC Huffman Table1 Entry Register	Essential	--
HACTBL1, HACTBLG1	AC Huffman Table1 Entry Register	Essential	--

Contents of registers in [Table 12-1](#) will be changed in following cases.

1. After user writes the registers again.
2. After reset operation is done.
3. After decompression of arbitrary JPEG file. In this case, the registers have the value from header of input JPEG file after header parsing process.

Except in the above cases, it is possible to process next picture by only performing the process start signal after process of a picture is completed.

12.6.4 PROCESS FOR IMAGE SIZE

Size of images in JPEG file has to be a specific value, which is the multiple of block size because JPEG file is composed of blocks. If image size in JPEG file header is not multiple of block size, actual image size in the file is the minimum among values which are the multiple of the block size and larger than the value in the header, but decoder shows cropped image with the size in file header. Minimum size for compression and decompression process is double of Block Size. Color format determines the block size as described in [Table 12-2](#).

Table 12-2 Relationship between Block Size and Color Format

Color format	MCU Block size (WxH)	Minimum size
YCbCr4:4:4	8x8	16x8 or 8x16
YCbCr4:2:2	16x8	32x8 or 16x16
YCbCr4:2:0	16x16	32x16 or 16x32
Gray (Y only)	8x8	16x8 or 8x16

1. Decompression

Input JPEG file has information about color format, width and height of the image in the frame header. User knows the information about the JPEG file after header parsing process. Actual raw image size after decompression is the minimum among the values which are the multiple of block size (known from color format) and larger than or equal to the image size in the header. For example, if JPEG file is YCbCr4:2:0 format and its size is 170x170, actual size of decompressed raw image is 176x176.

2. Proper process such as cropping is needed to display or store the result raw image in the width and height of the file header.

3. Compression

Width and height of input raw image must be the multiple of the block size corresponding to the output JPEG color format. If input raw image has arbitrary size, use padding process to modify the size to the multiple of block size. The modified size is the minimum values which are the multiple of the block size and larger than or equal to the original value. However, register setting value of width and height for compression must be the original value.

12.6.5 PROCESS FOR INPUT STREAM SIZE

For decompression of an illegal JPEG stream, JPEG core does not recognize the end of the stream if some important markers are damaged. Therefore, it is necessary to notify the input stream size to JPEG core.

12.6.6 INTERRUPT SIGNAL

Interrupt signal is generated under the following conditions, and the JPGINTST register identifies causes:

1. Compression or decompression process for one picture is complete,
2. Internal timer counting ends before completion of compression or decompression.
3. During compression, the byte size of output stream is larger than the predefined bound size in ENC_STREAM_BOUND.

In condition 1, the normal process is finished. To clear the interrupt request, read the JPGINTST register and JPGOPR register. If there is no encoding or decoding error, JPGINTST must be read as 0x40. If another value is read, the operation result may not be correct.

In condition 2, TIMER_INT_STAT is read as 1 and it is cleared by writing 1 in TIMER_INT_STAT register. In this case, JPEG requires reset or S/W reset before next operation.

In condition 3, ENC_STREAM_INT_STAT is read as 1, JPEG operation is stopped and there is no further memory access by JPEG. This interrupt is cleared by writing 1 in ENC_STREAM_INT_STAT register. In this case, JPEG needs reset or S/W reset before next operation.

12.6.7 INTERRUPT SETTING

If this JPGINTSE register is set, it invokes the interrupt when the input file for decompression is illegal.

To enable timer interrupt, set TIMER_INT_EN to 1 before start or restart.

To deal with an illegal input jpeg file for decompression, set RSTM_INT_EN , DATA_NUM_INT_EN or FINAL MCU_NUM_INT_EN.

To enable compressed stream size interrupt, set ENC_STREAM_INT_EN to 1 before starting compression.

12.6.8 S/W RESET

JPEG IP has a register for S/W reset. Steps to perform S/W reset:

1. Set 1 in the SW_RESET register.
2. Wait until SW_RESET register value changes to 0.
3. After the value changes to 0, set the proper register for next operation and start the operation.

If JPEG core is terminated abnormally or holds operation, S/W reset is needed to start new operation.

12.6.9 MARKER PROCESS

The following markers are generated during compression.

Table 12-3 Markers in JPEG Codec

Marker	Codes (hex)	Description
SOI	FFD8	Start of image
SOF0	FFC0	Baseline JPEG
SOS	FFDA	Start of scan
DQT	FFDB	Define quantization table
DHT	FFC4	Define Huffman table
DRI	FFDD	Define restart interval
RSTm	FFD0~FFD7	Restart with module 8 count "m"
EOI	FFD9	End of image

The markers [Table 12-3](#) are subject to process during decompression. The other markers except SOF1~SOFF and JPG are ignored.

12.6.10 BITSTREAM OF COMPRESSED FILE

The created JPEG bit stream is shown in [Figure 12-7](#). In the figure, ECS is an acronym of 'entropy-coded segment', which is a sequence of entropy-coded bytes.

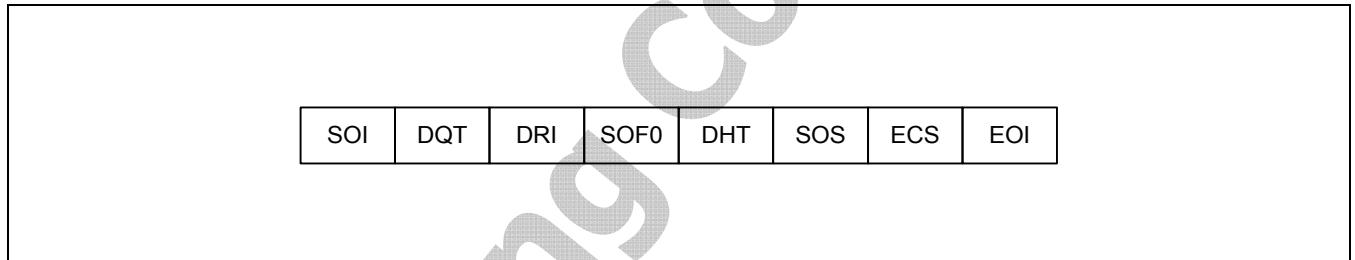


Figure 12-7 Bitstream of Compressed File

12.6.11 JPEG COMPRESSION FLOW

This is a pseudo instruction set that indicates the write or read on specific registers. It is assumed that the Huffman and quantization tables were written already.

```
// JPEG encoder initialization

Write SW_RESET      0x1
Write JPGCLKCON    0x1
Write JPGCMOD      0x20      // Mode selection
Write JPGMOD        0x1      // Encoding and YCbCr4:2:2
Write JPGDRI        0x0      // No DRI if 0. Set an appropriate value.
Write QTBL          0x0      // Choose the appropriate table index for Huffman & Quantization
tables.
Write JPGY_U        0x0      // Vertical resolution Upper byte
Write JPGY_L        0x0      // Vertical resolution Lower byte
Write JPGX_U        0x0      // Horizontal resolution Upper byte
Write JPGX_L        0x0      // Horizontal resolution Lower byte
Write IMGADR       0x1000_0000 // Address for an image to compress
Write JPGADR       0x1001_0000 // Address for the compressed JPEG file
Write COEF1         0x4D_971E // Color converter coefficients
Write COEF2         0x2c_5783 // Color converter coefficients
Write COEF3         0x83_6e13 // Color converter coefficients

// Encoding start.

Write JSTART        0x1

// After interrupt is detected, clear the pending register

Read JPGCNT_U      0x0      // Read the file size in bytes
Read JPGCNT_M      0x0      // Read the file size in bytes
Read JPGCNT_L      0x0      // Read the file size in bytes
Read JPGINTST     0x0      // It must be read 0x40.

Write JPGCOM        0x4      // Clear interrupt

Read JPGOPR        0x0      // It must be read zero.
```

12.6.12 JPEG DECOMPRESSION FLOW

This is a pseudo instruction set that indicates the write or read on specific registers.

```
// JPEG decoder initialization

Write SW_RESET           0x1
Write JPGCLKCON          0x1
Write JPGMOD              0x8      // Decoding mode
Write JPGINTSE             0x0     // Interrupt setting
Write OUTFORM              0x1     // Output raw image is YCbCr4:2:0
Write IMGADR                0x1000_0000 // Address for a decompressed raw image
Write JPGADR                0x1001_0000 // Address for a JPEG file to decompress

// Decoding operation start

Write JSTART                0x1
// After interrupt is detected, clear the pending register

Read JPGINTST               // It must be read 0x40.(it means normal end)
Write JPGCOM                  0x4      // Clear interrupt
Read JPGOPR                  // It must be read 0x0.
```

12.7 REGISTER DESCRIPTION

12.7.1 REGISTER MAP

JPEG has the control registers as shown in [Table 12-4](#) and table assignment in [Table 12-6](#).

Table 12-4 JPEG Codec Control Registers

Register	Address	R/W	Description	Reset Value
JPGMOD	0xFB60_0000	R/W	Specifies the Sub-sampling mode register	0x0000_0000
JPGOPR	0xFB60_0004	R	Specifies the operation status register	0x0000_0000
QTBL	0xFB60_0008	R/W	Specifies the Quantization Table Number Register	0x0000_0000
HTBL	0xFB60_000C	R/W	Specifies the Huffman table number register	0x0000_0000
JPGDRI_U	0xFB60_0010	R/W	Specifies the MCU, which inserts RST marker(upper 8-bit)	0x0000_0000
JPGDRI_L	0xFB60_0014	R/W	Specifies the MCU, which inserts RST marker (lower 8-bit)	0x0000_0000
JPGY_U	0xFB60_0018	R/W	Specifies the vertical resolution (upper 8-bit)	0x0000_0000
JPGY_L	0xFB60_001C	R/W	Specifies the vertical resolution (lower 8-bit)	0x0000_0000
JPGX_U	0xFB60_0020	R/W	Specifies the Horizontal resolution (upper 8-bit)	0x0000_0000
JPGX_L	0xFB60_0024	R/W	Specifies the Horizontal resolution (lower 8-bit)	0x0000_0000
JPGCNT_U	0xFB60_0028	R	Specifies the amount of the compressed data in bytes (upper 8-bit)	0x0000_0000
JPGCNT_M	0xFB60_002C	R	Specifies the amount of the compressed data in bytes (middle 8-bit)	0x0000_0000
JPGCNT_L	0xFB60_0030	R	Specifies the amount of the compressed data in bytes (lower 8-bit)	0x0000_0000
JPGINTSE	0xFB60_0034	R/W	Specifies the Interrupt Setting Register	0x0000_0000
JPGINTST	0xFB60_0038	R	Specifies the Interrupt Status Register	0x0000_0000
Reserved	0xFB60_003C 0xFB60_0048	-	-	-
JPGCOM	0xFB60_004C	W	Specifies the command register	0x0000_0000
IMGADR	0xFB60_0050	R/W	Specifies the source or destination image address	0x0000_0000
Reserved	0xFB60_0054	-	-	-
JPGADR	0xFB60_0058	R/W	Specifies the source or destination JPEG file address	0x0000_0000

Register	Address	R/W	Description	Reset Value
COEF1	0xFB60_005C	R/W	Specifies the coefficient values for RGB ↔ YCbCr Converter	0x0000_0000
COEF2	0xFB60_0060	R/W	Specifies the coefficient values for RGB ↔ YCbCr Converter	0x0000_0000
COEF3	0xFB60_0064	R/W	Specifies the coefficient values for RGB ↔ YCbCr converter	0x0000_0000
JPGCMOD	0xFB60_0068	R/W	Specifies the Mode Selection and Core Clock Setting	0x0000_0020
JPGCLKCON	0xFB60_006C	R/W	Specifies the Power On/ Off and clock down control	0x0000_0002
JSTART	0xFB60_0070	W	Specifies the start compression or decompression	0x0000_0000
Reserved	0xFB60_0074	W		0x0000_0000
SW_RESET	0xFB60_0078	R/W	Specifies the S/W reset	0x0000_0000
TIMER_SE	0xFB60_007C	R/W	Specifies the internal timer setting register	0xFFFF_FFFF
TIMER_ST	0xFB60_0080	R/W	Specifies the internal timer status register	0xFFFF_FFFF
COMSTAT	0xFB60_0084	R	Specifies the command status register	0x0000_0000
OUTFORM	0xFB60_0088	R/W	Specifies the output color format of decompression	0x0000_0000
VERSION	0xFB60_008C	R	Specifies the version register	0x0000_0003
Reserved	0xFB60_0090	-	-	-
ENC_STREAM_INTSE	0xFB60_0098	R/W	Specifies the compressed stream size interrupt setting register	0x00FF_FFE0
ENC_STREAM_INTST	0xFB60_009C	R/W	Specifies the compressed stream size interrupt status register	0x0000_0000
QTBL0	0xFB60_0400 0xFB60_04FC	R/W	Specifies the quantization table 0	0x0000_0000
QTBL1	0xFB60_0500 0xFB60_05FC	R/W	Specifies the quantization table 1	0x0000_0000
QTBL2	0xFB60_0600 0xFB60_06FC	R/W	Specifies the quantization table 2	0x0000_0000
QTBL3	0xFB60_0700 0xFB60_07FC	R/W	Specifies the quantization table 3	0x0000_0000
HDCTBL0	0xFB60_0800 0xFB60_083C	W	Specifies the Huffman DC Table 0 - the number of code per code length	0x0000_0000

Register	Address	R/W	Description	Reset Value
HDCTBLG0	0xFB60_0840 0xFB60_086C	W	Specifies the Huffman DC Table 0 - Group number of the order for occurrence	0x0000_0000
HACTBL0	0xFB60_0880 0xFB60_08BC	W	Specifies the Huffman AC Table 0 - the number of code per code length	0x0000_0000
HACTBLG0	0xFB60_08C0 0xFB60_0B44	W	Specifies the Huffman AC Table 0 - Group number of the order for occurrence	0x0000_0000
HDCTBL1	0xFB60_0C00 0xFB60_0C3C	W	Specifies the Huffman DC Table 1 - the number of code per code length	0x0000_0000
HDCTBLG1	0xFB60_0C40 0xFB60_0C6C	W	Specifies the Huffman DC Table 1 - Group number of the order for occurrence	0x0000_0000
HACTBL1	0xFB60_0C80 0xFB60_0CBC	W	Specifies the Huffman AC Table 1 - the number of code per code length	0x0000_0000
HACTBLG1	0xFB60_0CC0 0xFB60_0F44	W	Specifies the Huffman AC Table 1 - Group number of the order for occurrence	0x0000_0000

12.7.1.1 JPEG Mode Register (JPGMOD, R/W, Address = 0xFB60_0000)

JPGMOD	Bit	Description	Initial State
Reserved	[31:4]	Reserved but it should be 0x0	0
PROC_MODE	[3]	Process mode. 0 = Compression process. 1 = Decompression process.	0
SUBSAMPLING_MODE	[2:0]	Sub sampling mode 0x0 = chroma 4:4:4 format 0x1 = chroma 4:2:2 format. 0x2 = chroma 4:2:0 format 0x3 = Gray format (Single Component) Others are reserved. During decompression, these are read-only. During compression, only 0x1 or 0x2 are allowed.	0

12.7.1.2 JPEG Operation Status Register (JPGOPR, R, Address = 0xFB60_0004)

JPGOPR	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0
JPGOPR	[0]	0 = JPEG is not operating. 1 = JPEG is operating.	0

12.7.1.3 Quantization Table Number Register (QTBL, R/W, Address = 0xFB60_0008)

QTBL	Bit	Description	Initial State
Reserved	[31:8]	Reserved	0
QT_NUM4	[7:6]	Quantization table number for 4 th component.	0
QT_NUM3	[5:4]	Quantization table number for 3 rd component.	0
QT_NUM2	[3:2]	Quantization table number for 2 nd component	0
QT_NUM1	[1:0]	Quantization table number for 1 st component.	0

12.7.1.4 Huffman Table Number Register (HTBL, R/W, Address = 0xFB60_000C)

HTBL	Bit	Description	Initial State
Reserved	[31:8]	Reserved	0
HT_NUM4_AC	[7]	Huffman table number for 4 th color component AC	0
HT_NUM4_DC	[6]	Huffman table number for 4 th color component DC.	0
HT_NUM3_AC	[5]	Huffman table number for 3 rd color component AC.	0
HT_NUM3_DC	[4]	Huffman table number for 3 rd color component DC.	0
HT_NUM2_AC	[3]	Huffman table number for 2 nd color component AC.	0
HT_NUM2_DC	[2]	Huffman table number for 2 nd color component DC.	0
HT_NUM1_AC	[1]	Huffman table number for 1 st color component AC.	0
HT_NUM1_DC	[0]	Huffman table number for 1 st color component DC.	0

12.7.1.5 JPEG Restart Interval Upper byte Register (JPGDRI_U, R/W, Address = 0xFB60_0010)

JPGDRI_U	Bit	Description	Initial State
Reserved	[31:8]	Reserved	0
JPGDRI_U	[7:0]	<p>This is a restart interval that identifies the distance between two adjacent Restart Maker (RST) in terms of Minimum Coded Unit (MCU).</p> <p>It is valid in compression mode.</p> <p>If JPGDRI is set to 0, Define Restart Interval Marker (DRI) and RST is not inserted.</p>	0

12.7.1.6 JPEG Restart Interval Lower byte Register (JPGDRI_L, R/W, Address = 0xFB60_0014)

JPGDRI_L	Bit	Description	Initial State
Reserved	[31:8]	Reserved	0
JPGDRI_L	[7:0]	<p>It is a restart interval that identifies the distance between two adjacent Restart Maker (RST) in terms of Minimum Coded Unit (MCU).</p> <p>It is valid in compression mode.</p> <p>If JPGDRI is set to 0, Define Restart Interval Marker (DRI) and RST is not inserted.</p>	0

12.7.1.7 JPEG Vertical Resolution Upper byte Register (JPGY_U, R/W, Address = 0xFB60_0018)

JPGY_U	Bit	Description	Initial State
Reserved	[31:16]	Reserved	0
VER_RES	[15:0]	Upper byte of the Image size value in the vertical direction. You are not allowed to Set 0. This register is read-only during decompression.	0

12.7.1.8 JPEG Vertical Resolution Lower byte Register (JPGY_L, R/W, Address = 0xFB60_001C)

JPGY_L	Bit	Description	Initial State
Reserved	[31:16]	Reserved	0
VER_RES	[15:0]	Lower byte of the Image size value in the vertical direction. You are not allowed to Set 0. This register is read-only during decompression.	0

12.7.1.9 JPEG Horizontal Resolution Upper byte Register (JPGX_U, R/W, Address = 0xFB60_0020)

JPGX_U	Bit	Description	Initial State
Reserved	[31:16]	Reserved	0
HOR_RES	[15:0]	Upper byte of the Image size value in the horizontal direction. You are not allowed to Set 0. This register is read-only during decompression.	0

12.7.1.10 JPEG Horizontal Resolution Lower byte Register (JPGX_L, R/W, Address = 0xFB60_0024)

JPGX_L	Bit	Description	Initial State
Reserved	[31:16]	Reserved	0
HOR_RES	[15:0]	Lower byte of the Image size value in the horizontal direction. You are not allowed to Set 0. This register is read-only during decompression.	0

12.7.1.11 JPEG Byte Count Upper byte Register (JPGCNT_U, R, Address = 0xFB60_0028)

JPGCNT_U	Bit	Description	Initial State
Reserved	[31:24]	Reserved	0
BYTE_CNT	[23:0]	Upper byte of the count value the width of 24bits of the amount of compression data. Value of the register will be clear when processing starts. This register is valid in compression mode.	0x00_0000

12.7.1.12 JPEG Byte Count Middle byte Register (JPGCNT_M, R, Address = 0xFB60_002C)

JPGCNT_M	Bit	Description	Initial State
Reserved	[31:24]	Reserved	0
BYTE_CNT	[23:0]	Middle byte of the count value the width of 24bits of the amount of compression data. Value of the register will be clear when processing starts. This register is valid in compression mode.	0x00_0000

12.7.1.13 JPEG Byte Count Lower byte Register (JPGCNT_L, R, Address = 0xFB60_0030)

JPGCNT_L	Bit	Description	Initial State
Reserved	[31:24]	Reserved	0
BYTE_CNT	[23:0]	Lower byte of the count value the width of 24bits of the amount of compression data. Value of the register will be clear when processing starts. This register is valid in compression mode.	0x00_0000

12.7.1.14 JPEG Interrupt Setting Register (JPGINTSE, R/W, Address = 0xFB60_0034)

JPGINTSE	Bit	Description	Initial State
Reserved	[31:8]	Reserved	0
RSTm_INT_EN	[7]	The bit which decides whether interrupt is allowed or not, in case there is abnormality in restart interval period, data number in Huffman coding segments during decompression process. In case it is not set, error code will not be returned.	0
DATA_NUM_INT_EN	[6]	The bit which decides whether interrupt is allowed or not, in case there is abnormality in total data number in Huffman coding segments at decompression process. In case it is not set, error code will not be returned.	0
FINAL MCU_NUM_INT_EN	[5]	The bit which decides whether interrupt is allowed or not, in case there is abnormality in final MCU data number in Huffman coding segments at decompression process. In case it is not set, error code will not be returned.	0
Reserved	[4:0]	Reserved, but should be 0x0	0

12.7.1.15 JPEG Interrupt Status Register (JPGINTST, R, Address = 0xFB60_0038)

JPGINTST	Bit	Description	Initial State
Reserved	[31:7]	Reserved	0
RESULT_STAT	[6]	Result status. 0 = Processing was finished abnormally. 1 = Processing was done normally	0
STREAM_STAT	[5]	Bitstream error status. Valid during decompression only. 0 = There is no syntax error on the compressed file. 1 = There is syntax error on the compressed file.	0
Reserved	[4:0]	Reserved	0

12.7.1.16 JPEG command Register (JPGCOM, W, Address = 0xFB60_004C)

JPGCOM	Bit	Description	Initial State
Reserved	[31:3]	Reserved	0
INT_RELEASE	[2]	Interrupt signal release. When Interrupt occurs , set '1' If you set '1', Interrupt is canceled.	0
Reserved	[1:0]	Reserved but should be 0x0	0

12.7.1.17 Raw Image Data R/W Address Register (IMGADR, R/W, Address = 0xFB60_0050)

IMGADR	Bit	Description	Initial State
IMG_ADR	[31:0]	It is start address of raw image data. Value for this register has to be multiple of 32. In compression mode, raw image before compression is read from this address. In decompression mode, raw image after decompression is stored from address.	0

12.7.1.18 JPEG File R/W Address Register (JPGADR, R/W, Address = 0xFB60_0058)

JPGADR	Bit	Description	Initial State
JPG_ADR	[31:0]	It is start address of JPEG file data. Value for this register has to be multiple of 32. In compression mode, JPEG file after compression is stored from this address. In decompression mode, JPEG file before compression is read from this address.	0

12.7.1.19 Coefficient for RGB-to-YCbCr Converter Register (COEF1, R/W, Address = 0xFB60_005C)

COEF1	Bit	Description	Initial State
Reserved	[31:24]	Reserved	0
COEF11	[23:16]	Coefficient value of COEF11	0
COEF12	[15:8]	Coefficient value of COEF12	0
COEF13	[7:0]	Coefficient value of COEF13	0

12.7.1.20 Coefficient for RGB-to-YCbCr Converter Register (COEF2, R/W, Address = 0xFB60_0060)

COEF2	Bit	Description	Initial State
Reserved	[31:24]	Reserved	0
COEF21	[23:16]	Coefficient value of COEF21	0
COEF22	[15:8]	Coefficient value of COEF22	0
COEF23	[7:0]	Coefficient value of COEF23	0

12.7.1.21 Coefficient for RGB-to-YCbCr Converter Register (COEF3, R/W, Address = 0xFB60_0064)

COEF3	Bit	Description	Initial State
Reserved	[31:24]	Reserved	0
COEF31	[23:16]	Coefficient value of COEF31	0
COEF32	[15:8]	Coefficient value of COEF32	0
COEF33	[7:0]	Coefficient value of COEF33	0

The expression of 8-bit COEFxx is like following. For example, if COEFxx is set as 1100_0000b, the decimal value of COEFxx is 0.75. (= 0.5 + 0.25)

Table 12-5 Bitwise Expression of COEFxx

Bit	7	6	5	4	3	2	1	0
Value	0.5	0.25	0.125	0.0625	0.03125	0.015625	0.0078125	0.00390625

$$\begin{pmatrix} Y \\ Cb \\ Cr \end{pmatrix} = \begin{pmatrix} +COEF11 & +COEF12 & +COEF13 \\ -COEF21 & -COEF22 & +COEF23 \\ +COEF31 & -COEF32 & -COEF33 \end{pmatrix} \times \begin{pmatrix} R \\ G \\ B \end{pmatrix} + \begin{pmatrix} c1 \\ 128 \\ 128 \end{pmatrix}$$

12.7.1.22 JPEG Color Mode Register (JPGCMOD, R/W, Address = 0xFB60_0068)

JPGCMOD	Bit	Description	Initial State
Reserved	[31:8]	Reserved	0
MOD_SEL	[7:5]	Color space of input raw image 0x1 = YCbCr4:2:2 0x2 = RGB 565 Others are reserved.	1
Reserved	[4:2]	It must be set 0x0.	0
MODE_Y16	[1]	Y_16 selector for Y component 0 = c1 = 0 1 = c1 = 16 c1 is used in RGB-to-YCbCr converter, refer to Table 12-5 Bitwise Expression of COEFxx.	0
Reserved	[0]	Reserved, but should be 0x0	0

12.7.1.23 JPEG Clock Control Register (JPGLKCON, R/W, Address = 0xFB60_006C)

JPGLKCON	Bit	Description	Initial State
Reserved	[31:2]	Reserved	0
CLK_DOWN_READY	[1]	0 = Clock is enabled. 1 = JPEG is ready for disabling clock (Default). This value is changed to 1 if POWER_ON is set as 0 and JPEG is not working. This value is changed to 0 if POWER_ON is set as 1. If this value is 1, JSTART command is ineffective. This bit is read only.	1
POWER_ON	[0]	0 = Disables Clock (Default). 1 = Activates Clock.	0

12.7.1.24 JPEG Start Register (JSTART, W, Address = 0xFB60_0070)

JSTART	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0
JSTART	[0]	To start compression/ decompression, set this value to 1. After one clock, it is cleared with 0 internally. Before starting operation, you must set essential registers.	0

12.7.1.25 JPEG SW Reset Register (SW_RESET, R/W, Address = 0xFB60_0078)

SW_RESET	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0
SW_RESET	[0]	Writing 1 resets JPEG IP. Before reset finishes, its value is kept as 1. After reset is done, it is cleared with 0 internally. Therefore it is necessary to check this value is 0 before setting registers and starting operation.	0

12.7.1.26 JPEG Timer Setting Register (TIME_SE, R/W, Address = 0xFB60_007C)

TIMER_SE	Bit	Description	Initial State
TIMER_INT_EN	[31]	0 = Disables Interrupt by timer. 1= Enables Interrupt by timer.	0
TIMER_INIT	[30:0]	Target counting value is stored in this register. After start or restart, timer starts to down-count from this value to 0.	0x7FFF_FFFF

12.7.1.27 JPEG Timer Status Register (TIMER_ST, R/W, Address = 0xFB60_0080)

TIMER_ST	Bit	Description	Initial State
TIMER_INT_STAT	[31]	Timer interrupt status. If timer interrupt is enabled and timer counting value reaches 0, it is set 1. Writing 1 clears this value. Writing 0 has no effect.	0
TIMER_CNT	[30:0]	Timer counting value. If start or restart, it is initiated by TIMER_INIT value and starts to down-count. If JPEG operation finishes before end of counting, it holds the counter value at that time. This bit is read only.	0x7FFF_FFFF

12.7.1.28 JPEG Decompression Output Format Register (OUTFORM, R/W, Address = 0xFB60_0088)

OUTFORM	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0
DEC_OUT_FORMAT	[0]	Output color format of decompressed raw image during decompression. 0 = YCbCr 4:2:2 1 = YCbCr 4:2:0	0

12.7.1.29 JPEG Version Register (VERSION, R, Address = 0xFB60_008C)

VERSION	Bit	Description	Initial State
VERSION	[31:0]	Version Register	0x0003_0001

12.7.1.30 JPEG Compressed Stream Size Interrupt Setting Register (ENC_STREAM_INTSE, R/W, Address = 0xFB60_0098)

TIMER_SE	Bit	Description	Initial State
Reserved	[31:25]	Reserved	0
ENC_STREAM_INT_EN	[24]	0 = Disables Compressed stream size interrupt. 1 = Enables Compressed stream size interrupt.	0
ENC_STREAM_BOUND	[23:0]	The upper bound of the byte size of output compressed stream is stored in this register. This value should be multiple of 32.	0xFF_FFE0

12.7.1.31 JPEG Compressed Stream Size Interrupt Status Register (ENC_STREAM_INTST, R/W, Address = 0xFB60_009C)

TIMER_ST	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0
ENC_STREAM_INT_STAT	[0]	Compressed stream size interrupt status. If the byte size of output compressed stream is larger than the size predefined in ENC_STREAM_BOUND, it is set 1. Writing 1 clears this value. Writing 0 has no effect.	0

12.7.2 JPEG HUFFMAN AND QUANTIZATION REGISTER TABLES

Each data uses the least significant 8 bits of 32-bit register.

Table 12-6 JPEG Codec Table Assignment

Register	Address	R/W	Description	Reset Value
QTBL0	0xFB60_0400 0xFB60_0404 0xFB60_04FC	W	Quantization of table number 0 (64 data with the distance of 4 on address)	0x0000 0000 for 64 each data
QTBL1	0xFB60_0500 0xFB60_0504 0xFB60_05FC	W	Quantization of table number 1 (64 data with the distance of 4 on address)	0x0000 0000 for 64 each data
QTBL2	0xFB60_0600 0xFB60_0604 0xFB60_06FC	W	Quantization of table number 2 (64 data with the distance of 4 on address)	0x0000 0000 for 64 each data
QTBL3	0xFB60_0700 0xFB60_0704 0xFB60_07FC	W	Quantization of table number 3 (64 data with the distance of 4 on address)	0x0000 0000 for 64 each data
HDCTBL0	0xFB60_0800 0xFB60_0804 0xFB60_083C	W	JPEG DC Huffman Table 0 Register The number of code per code length (16 data with the distance of 4 on address)	-
HDCTBLG0	0xFB60_0840 0xFB60_0844 0xFB60_086C	W	JPEG DC Huffman Table 0 Register Group number of the order for occurrence (12 data with the distance of 4 on address)	-
HACTBL0	0xFB60_0880 0xFB60_0884 0xFB60_08BC	W	JPEG AC Huffman Table 0 Register The number of code per code length (16 data with the distance of 4 on address)	-
HACTBLG0	0xFB60_08C0 0xFB60_08C4 0xFB60_0B44	W	JPEG AC Huffman Table 0 Register Group number of the order for occurrence/ Group number (162 data with the distance of 4 on address)	-
HDCTBL1	0xFB60_0C00 0xFB60_0C04 0xFB60_0C3C	W	JPEG DC Huffman Table 1 Register The number of code per code length (16 data with the distance of 4 on address)	-

Register	Address	R/W	Description	Reset Value
HDCTBLG1	0xFB60_0C40 0xFB60_0C44 0xFB60_0C6C	W	JPEG DC Huffman Table 1 Register Group number of the order for occurrence (12 data with the distance of 4 on address)	-
HACTBL1	0xFB60_0C80 0xFB60_0C84 0xFB60_0CBC	W	JPEG AC Huffman Table 1 Register The number of code per code length (16 data with the distance of 4 on address)	-
HACTBLG1	0xFB60_0CC0 0xFB60_0CC4 0xFB60_0F44	W	JPEG AC Huffman Table 1 Register Group number of the order for occurrence/ Group number (162 data with the distance of 4 on address)	-

Each data uses the least significant 8 bits of 32-bit register.

13 G2D

13.1 INTRODUCTION

FIMG-2D is a 2D graphics accelerator that supports Bit Block Transfer (BitBLT).

Rendering a primitive takes two steps: 1) configure the rendering parameters, such as foreground color and the coordinate data, by setting the drawing-context registers; 2) start the rendering process by setting the relevant command registers accordingly.

13.2 FEATURES

- Primitives
 - BitBLT
 - o Stretched BitBLT support (Nearest sampling using Bresnham algorithm)
 - o Memory to Screen
 - o Memory to Memory
 - o Reverse Addressing (X Positive/Negative, Y Positive/Negative)
- Per-pixel Operation
 - Maximum 8000x8000 image size
 - Window Clipping
 - 90° /180° /270° Rotation
 - X-flip/Y-flip
 - Totally 4-operand Raster Operation (ROP4)
 - o Mask, Pattern, Source, Destination
 - Alpha Blending
 - o Alpha Blending with a user-specified constant alpha
 - o Per-pixel Alpha Blending
 - o Alpha Blending with both a constant alpha and per-pixel alpha
 - Color Key
- Data Format
 - 16/24/32-bpp, Packed 24bpp color format support
- Supports up to 222 MHz core clock
- Core clock must be faster than APB clock

13.3 COLOR FORMAT CONVERSION

FIMG-2D V3.0 supports eight color formats: XRGB_8888, ARGB_8888, RGB_565, XRGB_1555, ARGB_1555, XRGB_4444, ARGB_4444, and PACKED_RGB_888. The structure of each color format is illustrated in the figure below. FIMG-2D V3.0 supports four channel orders: ARGB, RGBA, ABGR, and BGRA.

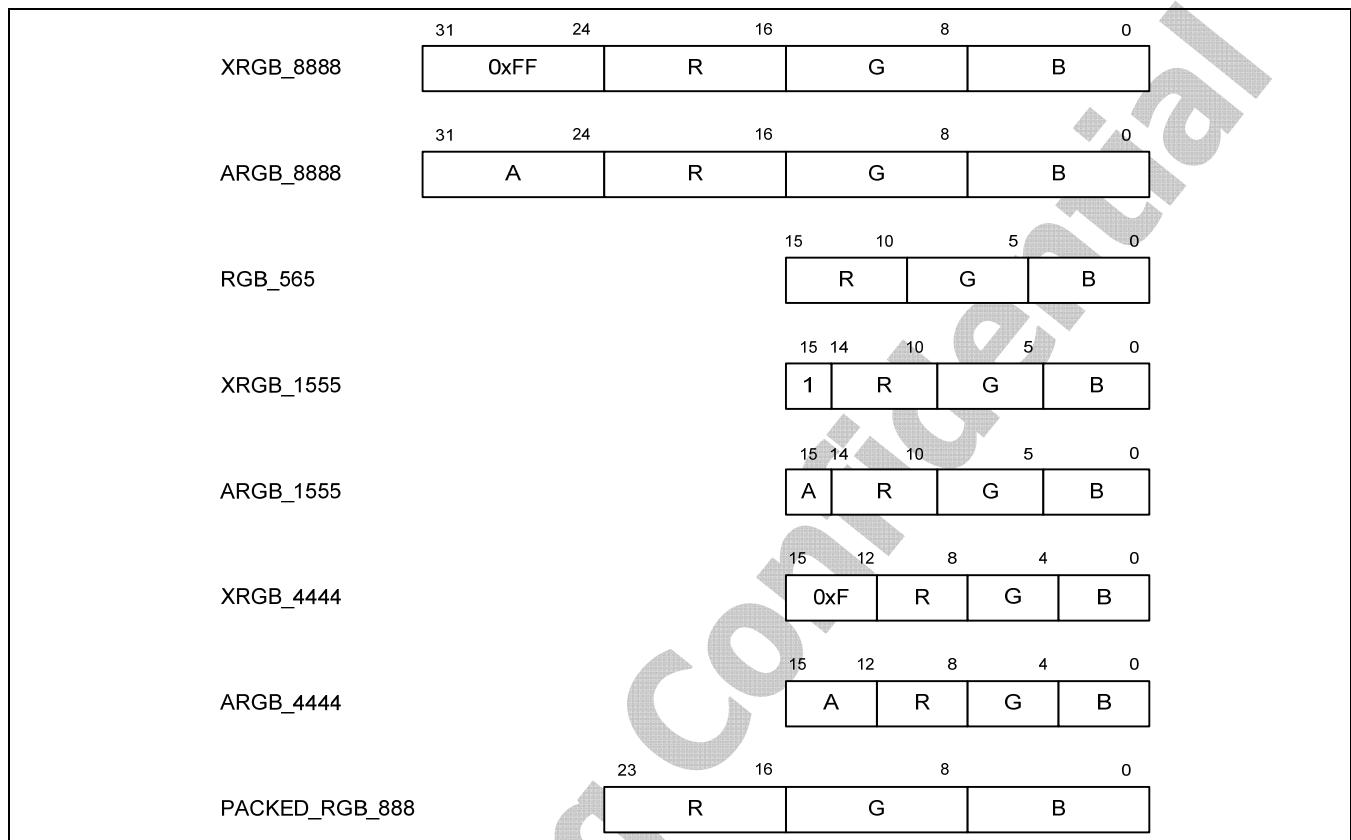


Figure 13-1 Color Format

The internal computations use ARGB_8888 format. All data (source, destination, foreground, background, blue-screen, pattern) are converted to ARGB_8888 format before computation, and the final result are converted to the color format specified by DST_COLOR_MODE_REG before writing to frame buffer.

When a 16-bit color data is converted to 32-bit, the expanded color data is made by the following rule: the data of each field is shifted $(8 - x)$ bits to left, where x is the bit-width of the field. The least significant x bits of the new field data are padded with the most significant x bits of the original field data. For example, if the R value in RGB_565 format is 5'b11010, it will be converted to 8'b11010110, with three LSBs padded with three MSBs (3'b110) from the original R value. Note that, the A field in RGBA_5551 and ARGB_1555 only has one bit, so it is converted to either 8'b00000000 or 8'b11111111 (A=1'b1).

When a 32-bit color data is converted to 16-bit, the data of each field is truncated to x bits, where x is the bit-width of the field in the new color format. For example, if the R value in ARGB_8888 format is 8'b11001110, it will be converted to 5'b11001 in the RGB_565 format, with the three LSBs discarded. Note that, if the A field of the 32-bit color data is not 0, the A field in ARGB_1555 will be 1'b1; otherwise, 1'b0.

13.4 RENDERING PIPELINE

The rendering pipeline of FIMG-2D V3.0 is illustrated in [Figure 13-2](#). The functionality and related registers of each stage are introduced in detail in the rest of this chapter.

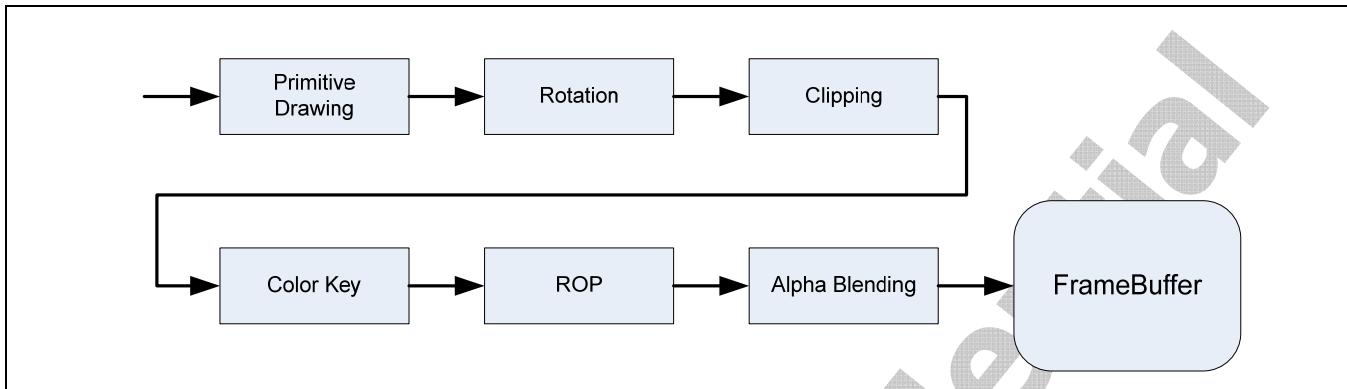


Figure 13-2 FIMG-2D Rendering Pipeline

13.4.1 PRIMITIVE DRAWING

Primitive Drawing determines the pixels to fill, and pass their coordinates to the next stage for further operations.

FIMG-2D V3.0 supports bit block transfer.

13.4.1.1 Bit Block Transfer

A Bit Block Transfer is a transformation of a rectangular block of pixels. Typical applications include copying the off-screen pixel data to frame buffer, selecting one of two raster operations by mask value, combining two bitmap patterns by the selected raster operation, changing the dimension of a rectangular image and so on.

Stretch Bit Block Transfer is implemented using Bresnham algorithm and nearest sampling.

13.4.1.1.1 On-Screen Rendering

On-screen bit block transfer copies a rectangular block of pixels on screen to another position on the same screen. Note that on-screen rendering has the following restriction:

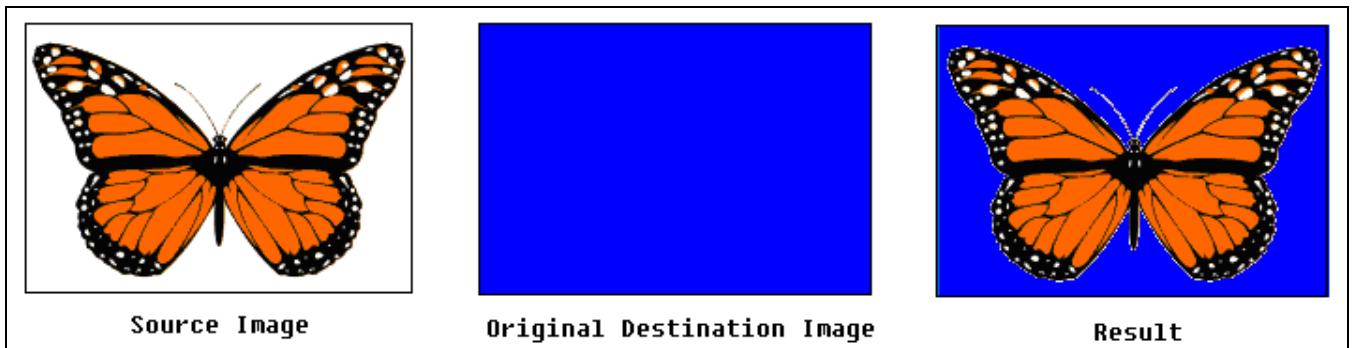
- SRC_BASE_ADDR_REG = DST_BASE_ADDR_REG
- SRC_SIZE_REG = DST_SIZE_REG
- SRC_COLOR_MODE_REG = DST_COLOR_MODE_REG

13.4.1.1.2 Off-Screen Rendering

Off-screen bit block transfer copies pixel data from off-screen memory to frame buffer. Color format conversion is performed automatically if SRC_COLOR_MODE_REG differs from DST_COLOR_MODE_REG.

13.4.1.1.3 Transparent Mode

FIMG-2D V3.0 can render image in Transparent Mode. In this mode, the pixels having the same color with blue screen color (BS_COLOR_REG) are discarded, resulting in a transparent effect. The function of Transparent Mode is illustrated in the images below, in which the BS_COLOR_REG is set to white.



FIMG-2D V3.0 also support Blue Screen Mode, in which the pixels having the same color with blue screen color (BS_COLOR_REG) are replaced by the background color (BG_COLOR_REG).

FIMG-2D V3.0 supports both memory-to-memory mode and memory-to-screen mode of BLT.

13.4.1.1.4 Related Registers

SRC_LEFT_TOP_REG	Coordinate of the leftmost topmost coordinate of the source image.
SRC_RIGHT_BOTTOM_REG	Coordinate of the rightmost bottommost coordinate of the source image.
SRC_SELECT_REG	Select one of the following cases - 2'b00: normal mode (source image in the memory), 2'b01: foreground color, 2'b10: background color.
DST_LEFT_TOP_REG	Coordinate of the leftmost topmost coordinate of the destination image.
DST_RIGHT_BOTTOM_REG	Coordinate of the rightmost bottommost coordinate of the destination image.
DST_SELECT_REG	Select one of the following cases - 2'b00: normal mode (destination image in the memory), 2'b01: foreground color, 2'b10: background color.
SRC_BASE_ADDR_REG	The base address of the source image (when normal mode is used in SRC_SELECT_REG).
DST_BASE_ADDR_REG	The base address of the destination image (usually the frame buffer base address).
SRC_SIZE_REG	The width and height of the source image.
DST_SIZE_REG	The width and height of the destination image.
SRC_COLOR_MODE_REG	The color format and channel order of the source image.
DST_COLOR_MODE_REG	The color format and channel order of the destination image.
FG_COLOR_REG	Foreground color value.
BG_COLOR_REG	Background color value.
BS_COLOR_REG	Blue screen color value.
BITBLT_COMMAND_REG	Enable/disable Transparent Mode or Blue Screen Mode.

13.4.2 ROTATION AND ADDRESSING DIRECTION (FLIP)

The pixels can be rotated by 90 degree clockwise or flipped around X-axis or Y-axis. The flip operation can be performed by direction of source read and destination read. The effects of rotation and flip options are summarized in the following table and illustrated in [Figure 13-3](#).

13.4.2.1.1 Related Registers

ROTATE_REG	Enable 90 degree rotation
SRC_MSK_DIRECT_REG	Addressing direction of source/mask memory to read
DST_PAT_DIRECT_REG	Addressing direction of destination/pattern memory to read and write

13.4.2.1.2 Rotation Effect

0°	Rotated X = Original X Rotated Y = Original Y
90°	Rotated X = Original Y Rotated Y = Original Width - 1 - Original X

13.4.2.1.3 Addressing Direction Effect

	Effect
Src X Direction = Dst X Direction	No flip over X axis
Src X Direction ≠ Dst X Direction	Horizontal flip
Src Y Direction = Dst Y Direction	No flip over Y axis
Src Y Direction ≠ Dst Y Direction	Vertical flip

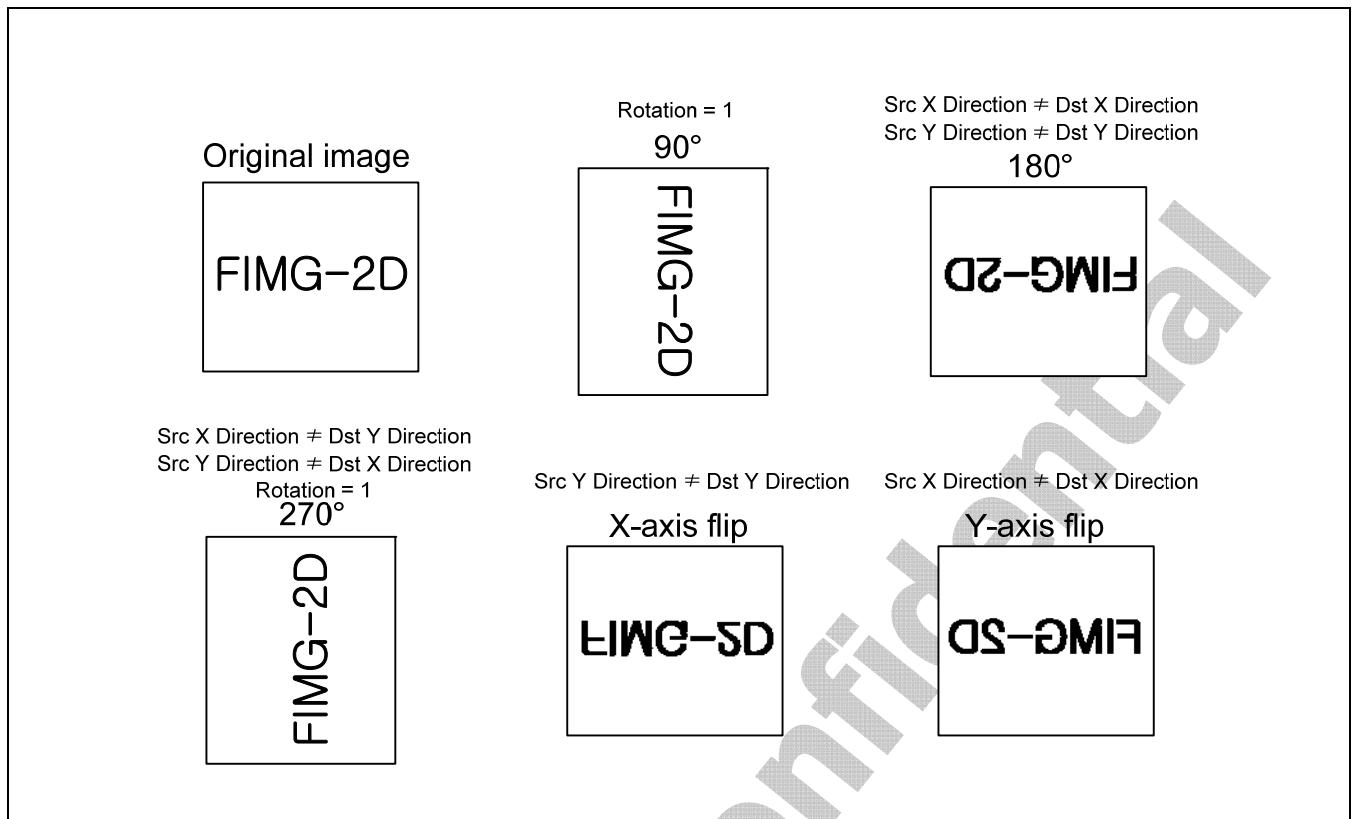


Figure 13-3 Rotation and Flip Example

13.4.3 CLIPPING

Clipping discards the pixels (after rotation) outside the clipping window. The discarded pixels will not go through the rest of rendering pipelines.

Note that the clipping windows must reside totally inside the screen. Setting the clipping window the same size with the screen will disable the clipping effect, and a clipping window bigger than the screen size is not allowed.

13.4.3.1.1 Related Registers

BITBLT_COMMAND_REG	Enable/disable Clipping Window (CWE _n Field)
CW_LT_REG	Coordinate of the leftmost topmost point of the clipping window
CW_RB_REG	Coordinate of the rightmost bottommost point of the clipping window

13.4.4 COLOR KEY

The Color Key conditionally discards a pixel based on the outcome of a comparison between the color value of this pixel of the source/destination image and the DR(min)/DR(max) values. If each field (R, G, B, A) of the color value falls in the range of [DR(min), DR(max)], this pixel is passed to the next stage; otherwise, discarded. User can disable the stencil test on a specific field by clearing the corresponding bits in COLORKEY_CNTL.

13.4.4.1.1 Related Registers

SRC_COLORKEY_CTRL_REG	Source Stencil Test configurations, such as enable/disable the test and so on.
SRC_COLORKEY_DR_MIN_REG	Set the source DR(min) value for each field
SRC_COLORKEY_DR_MAX_REG	Set the source DR(max) value for each field
DST_COLORKEY_CTRL_REG	Destination Stencil Test configurations, such as enable/disable the test and so on.
DST_COLORKEY_DR_MIN_REG	Set the DR(min) value for each field
DST_COLORKEY_DR_MAX_REG	Set the DR(max) value for each field

13.4.5 RASTER OPERATION

Raster operation performs Boolean operations on four operands: Mask, third operand, source, and destination according to two 8-bit-ROP3 values specified by the user. User can choose unmasked ROP3 value and masked ROP3 value with binary mask image. Mask should be the same size as source image.

The following table is the truth table of ROP3.

Third Operand	Source	Destination	ROP Value
0	0	0	Bit0
0	0	1	Bit1
0	1	0	Bit2
0	1	1	Bit3
1	0	0	Bit4
1	0	1	Bit5
1	1	0	Bit6
1	1	1	Bit7

The third operand can be pattern, foreground color, or background color; configurable by THIRD_OPERAND_REG.

The pattern supports all the format of source image or destination image. The following equation is used to calculate the pattern pixel coordinate (x, y):

- $X = (\text{PatternOffsetX} + x) \% \text{PatternWidth}$
- $Y = (\text{PatternOffsetY} + y) \% \text{PatternHeight}$,

where PatternOffsetY and PatternOffsetX are the offset value specified in register PAT_OFFSET_REG, and PatternWidth and PatternHeight are size of the pattern specified in register PAT_SIZE_REG

Here are some examples on how to use the ROP3 value to perform the operations:

- Final Data = Source. Only the Source data matter, so ROP Value = “0xCC”.
- Final Data = Destination. Only the Destination data matter, so ROP Value = “0xAA”.
- Final Data = Pattern. Only the Pattern data matter, so ROP Value = “0xF0”.
- Final Data = Source AND Destination. ROP Value = “0xCC” & “0xAA” = “0x88”
- Final Data = Source OR Pattern. ROP Value = “0xCC” | “0xF0” = “0xFC”.

Note that the Raster Operation only applies on Red, Green and, Blue fields of the color data; the Alpha field will not be affected.

13.4.5.1.1 Related Registers

PAT_BASE_ADDR_REG	Base address of the pattern image
PAT_SIZE_REG	Size of the pattern
PAT_COLOR_MODE_REG	Color channel order and color format of the pattern
PAT_OFFSET_REG	Coordinate offset of the pattern
MASK_BASE_ADDR_REG	Base address of the mask image
THIRD_OPERAND_REG	Third operand selection for unmasked ROP3 and masked ROP3
ROP4_REG	ROP4 Value

13.4.6 ALPHA BLENDING

Alpha Blending combines the source color and the destination color in the frame buffer to get the new destination color. Alpha Blending equation is decided by source alpha value and user-specified alpha value.

13.4.6.1.1 Related Registers

BITBLT_COMMAND_REG	Alpha blending configurations: alpha blending disable/enable, fading disable/enable.
ALPHA_REG	Alpha value and fading offset value.

13.5 REGISTER DESCRIPTION

13.5.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
General Registers				
SOFT_RESET_REG	0xFA00_0000	W	Software reset register	0x0000_0000
INTEN_REG	0xFA00_0004	R/W	Interrupt Enable register	0x0000_0000
INTC_PEND_REG	0xFA00_000C	R/W	Interrupt Control Pending register	0x0000_0000
FIFO_STAT_REG	0xFA00_0010	R	Command FIFO Status register	0x0000_0001
AXI_ID_MODE_REG	0xFA00_0014	R/W	AXI Read ID Mode register	0x0000_0000
CACHECTL_REG	0xFA00_0018	R/W	Cache & Buffer clear register	0x0000_0000
Command Registers				
BITBLT_START_REG	0xFA00_0100	W	BitBLT Start register	-
BITBLT_COMMAND_REG	0xFA00_0104	R/W	Command register for BitBLT	0x0000_0000
Parameter Setting Registers (Rotate & Direction)				
ROTATE_REG	0xFA00_0200	R/W	Rotation register	0x0000_0000
SRC_MSK_DIRECT_REG	0xFA00_0204	R/W	Source and Mask Direction register	0x0000_0000
DST_PAT_DIRECT_REG	0xFA00_0208	R/W	Destination and Pattern Direction register	0x0000_0000
Parameter Setting Registers (Source)				
SRC_SELECT_REG	0xFA00_0300	R/W	Source Image Selection register	0x0000_0000
SRC_BASE_ADDR_REG	0xFA00_0304	R/W	Source Image Base Address register	0x0000_0000
SRC_STRIDE_REG	0xFA00_0308	R/W	Source Stride register	0x0000_0000
SRC_COLOR_MODE_REG	0xFA00_030C	R/W	Source Image Color Mode register	0x0000_0000
SRC_LEFT_TOP_REG	0xFA00_0310	R/W	Source Left Top Coordinate register	0x0000_0000
SRC_RIGHT_BOTTOM_REG	0xFA00_0314	R/W	Source Right Bottom Coordinate register	0x0000_0000
Parameter Setting Registers (Destination)				
DST_SELECT_REG	0xFA00_0400	R/W	Destination Image Selection register	0x0000_0000
DST_BASE_ADDR_REG	0xFA00_0404	R/W	Destination Image Base Address register	0x0000_0000
DST_STRIDE_REG	0xFA00_0408	R/W	Destination Stride register	0x0000_0000
DST_COLOR_MODE_REG	0xFA00_040C	R/W	Destination Image Color Mode register	0x0000_0000

Register	Address	R/W	Description	Reset Value
DST_LEFT_TOP_REG	0xFA00_0410	R/W	Destination Left Top Coordinate register	0x0000_0000
DST_RIGHT_BOTTOM_REG	0xFA00_0414	R/W	Destination Right Bottom Coordinate register	0x0000_0000
Parameter Setting Registers (Pattern)				
PAT_BASE_ADDR_REG	0xFA00_0500	R/W	Pattern Image Base Address register	0x0000_0000
PAT_SIZE_REG	0xFA00_0504	R/W	Pattern Image Size register	0x0001_0001
PAT_COLOR_MODE_REG	0xFA00_0508	R/W	Pattern Image Color Mode register	0x0000_0000
PAT_OFFSET_REG	0xFA00_050C	R/W	Pattern Left Top Coordinate register	0x0000_0000
PAT_STRIDE_REG	0xFA00_0510	R/W	Pattern Stride register	0x0000_0000
Parameter Setting Registers (Mask)				
MASK_BASE_ADDR_REG	0xFA00_0520	R/W	Mask Base Address register	0x0000_0000
MASK_STRIDE_REG	0xFA00_0524	R/W	Mask Stride register	0x0000_0000
Parameter Setting Registers (Clipping Window)				
CW_LT_REG	0xFA00_0600	R/W	LeftTop coordinates of Clip Window	0x0000_0000
CW_RB_REG	0xFA00_0604	R/W	RightBottom coordinates of Clip Window	0x0000_0000
Parameter Setting Registers (ROP & Alpha Setting)				
THIRD_OPERAND_REG	0xFA00_0610	R/W	Third Operand Selection register	0x0000_0000
ROP4_REG	0xFA00_0614	R/W	Raster Operation register	0x0000_0000
ALPHA_REG	0xFA00_0618	R/W	Alpha value, Fading offset value	0x0000_0000
Parameter Setting Registers (Color)				
FG_COLOR_REG	0xFA00_0700	R/W	Foreground Color register	0x0000_0000
BG_COLOR_REG	0xFA00_0704	R/W	Background Color register	0x0000_0000
BS_COLOR_REG	0xFA00_0708	R/W	Blue Screen Color register	0x0000_0000
Parameter Setting Registers (Color Key)				
SRC_COLORKEY_CTRL_REG	0xFA00_0710	R/W	Source Colorkey control register	0x0000_0000
SRC_COLORKEY_DR_MIN_REG	0xFA00_0714	R/W	Source Colorkey Decision Reference Minimum register	0x0000_0000
SRC_COLORKEY_DR_MAX_REG	0xFA00_0718	R/W	Source Colorkey Decision Reference Maximum register	0xFFFF_FFFF
DST_COLORKEY_CTRL_REG	0xFA00_071C	R/W	Destination Colorkey control register	0x0000_0000
DST_COLORKEY_	0xFA00_0720	R/W	Destination Colorkey Decision Reference	0x0000_0000

Register	Address	R/W	Description	Reset Value
DR_MIN_REG			Minimum register	
DST_COLORKEY_DR_MAX_REG	0xFA00_0724	R/W	Destination Colorkey Decision Reference Maximum register	0xFFFF_FFFF

Samsung Confidential

13.5.2 GENERAL REGISTERS

13.5.2.1 Software Reset Register (SOFT_RESET_REG, W, Address = 0xFA00_0000)

SOFT_RESET_REG	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0x0
R	[0]	Software Reset Write to this bit results in a one-cycle reset signal to FIMG2D graphics engine. Every command register and parameter setting register will be assigned the “Reset Value”,	0x0

13.5.2.2 Interrupt Enable Register (INTEN_REG, R/W, Address = 0xFA00_0004)

INTEN_REG	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0x0
INT_TYPE	[1]	Must be set to 0 (cannot be 1)	0x0
CF	[0]	Command Finished interrupt enable. If this bit is set, when the graphics engine finishes the execution of command, an interrupt occurs, and the INTP_CMD_FIN flag in INTC_PEND_REG will be set.	0x0

13.5.2.3 Interrupt Pending Register (INTC_PEND_REG, R/W, Address = 0xFA00_000C)

INTC_PEND_REG	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0x0
INTP_CMD_FIN	[0]	Command Finished interrupt flag. Writing ‘1’ to this bit clears this flag It is recommended to clear this bit before Start_BitBLT because of previous Start_BitBLT’s residue	0x0

13.5.2.4 FIFO Status Register (FIFO_STAT_REG, R, Address = 0xFA00_0010)

FIFO_STAT_REG	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0x0
CMD_FIN	[0]	1 = The graphics engine finishes the execution of command. 0 = In the middle of rendering process.	0x1

13.5.2.5 AXI ID Mode Register (AXI_ID_MODE_REG, R/W, Address = 0xFA00_0014)

AXI_ID_MODE_REG	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0x0
AXI_RD_ID_MODE	[0]	This bit is for out of ordering of AXI Master Read. If this bit is set, the several read port of this engine have each AXI RID on only one AXI MASTER I/F 1 = Multiple ID (out of order) 0 = Single ID fixing on 4'b0 (In order)	0x0

13.5.2.6 Cache Control Register (CACHECTL_REG, R/W, Address = 0xFA00_0018)

CACHECTL_REG	Bit	Description	Initial State
Reserved	[31:3]	Reserved	0x0
PATCACHE_CLEAR	[2]	Pattern cache clear (Automatically set to 0b after a cycle) This bit is used to invalidate the contents of pattern cache. 0 = Default stages; pattern cache invalidation unchanged 1 = Pattern cache starts invalidation	0x0
SRCBUFFER_CLEAR	[1]	Source buffer clear (Automatically set to 0b after a cycle) This bit is used to invalidate the contents of source buffer. 0 = Default stages; source buffer invalidation unchanged 1 = Source buffer starts invalidation	0x0
MASKBUFFER_CLEAR	[0]	Mask buffer clear (Automatically set to 0b after a cycle) This bit is used to invalidate the contents of mask buffer. 0 = Default stages; Mask buffer invalidation unchanged 1 = Mask buffer starts invalidation	0x0

13.5.3 COMMAND REGISTERS

13.5.3.1 BitBLT Start Register (BITBLT_START_REG, W, Address = 0xFA00_0100)

BITBLT_START_REG	Bit	Description	Initial State
Reserved	[31:2]	Reserved	0x0
Start_BitBLT	[0]	Start BitBLT Operation. When this bit is set, it is automatically clear after one clock cycle.	0x0

13.5.3.2 BitBLT Command Register (BITBLT_COMMAND_REG, R/W, Address = 0xFA00_0104)

BITBLT_COMMAND_REG	Bit	Description	Initial State
Reserved	[31:24]	Reserved	0x0
SrcNonPreBlendMode	[23:22]	Source Non-premultiplied 2'b00: Disable 2'b01: Alpha Blend with Constant Alpha 2'b10: Alpha Blend with PerPixel Alpha 2'b11: Reserved	0x0
AlphaBlendMode	[21:20]	Alpha Blending Mode 2'b00: No Alpha Blending 2'b01: Alpha Blending 2'b10: Fading 2'b11: Reserved	0x0
Reserved	[19:18]	Reserved	0x0
ColorKeyMode	[17:16]	2'b00: Disable colorkey 2'b01: Enable source colorkey 2'b10: Enable destination colorkey 2'b11: Enable source colorkey and destination colorkey	0x0
Reserved	[15:14]	Reserved	0x0
Transparent Mode	[13:12]	2'b00: Opaque Mode 2'b01: Transparent Mode 2'b10: BlueScreen Mode 2'b11: Reserved	0x0
Reserved	[11:9]	Reserved	0x0
CWEn	[8]	Enable Clipping Window	0x0
Reserved	[7:5]	Reserved	0x0
StretchEn	[4]	Enable Stretch Mode	0x0
Reserved	[3:1]	Reserved	0x0
MaskEn	[0]	Enable Mask Operand (ROP4 Operation)	0x0

13.5.4 PARAMETER SETTING REGISTERS (ROTATION & DIRECTION)

13.5.4.1 Rotation Register (ROTATE_REG, R/W, Address = 0xFA00_0200)

ROTATE_REG	Bit	Description	Initial State
Reserved	[31:2]	Reserved	0x0
Rotate	[0]	0 = No rotation 1 = 90 degree rotation	0x0

13.5.4.2 Source and Mask Direction Register (SRC_MSK_DIRECT_REG, R/W, Address = 0xFA00_0204)

SRC_MSK_DIRECT_REG	Bit	Description	Initial State
Reserved	[31:6]	Reserved	0x0
MskYDirect	[5]	0 = Y positive 1 = Y negative	0x0
MskXDirect	[4]	0 = X positive 1 = X negative	0x0
Reserved	[3:2]	Reserved	0x0
SrcYDirect	[1]	0 = Y positive 1 = Y negative	0x0
SrcXDirect	[0]	0 = X positive 1 = X negative	0x0

NOTE: Mask direction is usually same as source direction.

13.5.4.3 Destination and Pattern Direction Register(DST_PAT_DIRECT_REG, R/W, Address = 0xFA00_0208)

DST_PAT_DIRECT_REG	Bit	Description	Initial State
Reserved	[31:6]	Reserved	0x0
PatYDirect	[5]	0 = Y positive 1 = Y negative	0x0
PatXDirect	[4]	0 = X positive 1 = X negative	0x0
Reserved	[3:2]	Reserved	0x0
DstYDirect	[1]	0 = Y positive 1 = Y negative	0x0
DstXDirect	[0]	0 = X positive 1 = X negative	0x0

13.5.5 PARAMETER SETTING REGISTERS (SOURCE)

13.5.5.1 Source Image Selection Register (SRC_SELECT_REG, W, Address = 0xFA00_0300)

SRC_SELECT_REG	Bit	Description	Initial State
Reserved	[31:2]	Reserved	0x0
SrcSelect	[1:0]	Select Source 2'b00: Normal Mode(Using source image in the external memory) 2'b01: Using foreground color as source image 2'b10: Using background color as source image 2'b11: Reserved	0x0

13.5.5.2 Source Image Base Address Register (SRC_BASE_ADDR_REG, W, Address = 0xFA00_0304)

SRC_BASE_ADDR_REG	Bit	Description	Initial State
SrcAddr	[31:0]	Base address of the source image	0x0

13.5.5.3 Source Stride Register (SRC_STRIDE_REG, R/W, Address = 0xFA00_0308)

SRC_STRIDE_REG	Bit	Description	Initial State
Reserved	[31:16]	Reserved	0x0
SrcStride	[15:0]	Source stride (2's complement value)	0x0

13.5.5.4 Source Image Color Mode Register (SRC_COLOR_MODE_REG, R/W, Address = 0xFA00_030C)

SRC_COLOR_MODE_REG	Bit	Description	Initial State
Reserved	[31:6]	Reserved	0x0
SrcChannelOrder	[5:4]	2'b00: {A,X}RGB 2'b01: RGB{A,X} 2'b10: {A,X}BGR 2'b11: BGR{A,X}	0x0
Reserved	[3]	Reserved	0x0
SrcColorFormat	[2:0]	3'b000: XRGB_8888 3'b001: ARGB_8888 3'b010: RGB_565 3'b011: XRGB_1555 3'b100: ARGB_1555 3'b101: XRGB_4444 3'b110: ARGB_4444 3'b111: PACKED_RGB_888	0x0

13.5.5.5 Source Left Top Coordinate Register (SRC_LEFT_TOP_REG, R/W, Address = 0xFA00_0310)

SRC_LEFT_TOP_REG	Bit	Description	Initial State
Reserved	[31:29]	Reserved	0x0
SrcTopY	[28:16]	Left Top Y Coordinate of Source Image Range: 0 ~ 8000 (Requirement: SrcTopY < SrcBottomY)	0x0
Reserved	[15:13]	Reserved	0x0
SrcLeftX	[12:0]	Left Top X Coordinate of Source Image Range: 0 ~ 8000 (Requirement: SrcLeftX < SrcRightX)	0x0

13.5.5.6 Source Right Bottom Coordinate Register (SRC_RIGHT_BOTTOM_REG, R/W, Address = 0xFA00_0314)

SRC_RIGHT_BOTTOM_REG	Bit	Description	Initial State
Reserved	[31:29]	Reserved	0x0
SrcBottomY	[28:16]	Right Bottom Y Coordinate of Source Image Range: 0 ~ 8000	0x0
Reserved	[15:13]	Reserved	0x0
SrcRightX	[12:0]	Right Bottom X Coordinate of Source Image Range: 0 ~ 8000	0x0

13.5.6 PARAMETER SETTING REGISTERS (DESTINATION)

13.5.6.1 Destination Image Selection Register (DST_SELECT_REG, W, Address = 0xFA00_0400)

DST_SELECT_REG	Bit	Description	Initial State
Reserved	[31:2]	Reserved	0x0
DstSelect	[1:0]	Select Destination 2'b00: Normal Mode(Using destination image in the external memory) 2'b01: Using foreground color as destination image 2'b10: Using background color as destination image 2'b11: Reserved	0x0

13.5.6.2 Destination Image Base Address Register (DST_BASE_ADDR_REG, W, Address = 0xFA00_0404)

DST_BASE_ADDR_REG	Bit	Description	Initial State
DstAddr	[31:0]	Base address of the destination image	0x0

13.5.6.3 Destination Stride Register (DST_STRIDE_REG, R/W, Address = 0xFA00_0408)

DST_STRIDE_REG	Bit	Description	Initial State
Reserved	[31:16]	Reserved	0x0
DstStride	[15:0]	Destination stride (2's complement value).	0x0

13.5.6.4 Destination Image Color Mode Register(DST_COLOR_MODE_REG, R/W, Address = 0xFA00_040C)

DST_COLOR_MODE_REG	Bit	Description	Initial State
Reserved	[31:6]	Reserved	0x0
DstChannelOrder	[5:4]	2'b00: {A,X}RGB 2'b01: RGB{A,X} 2'b10: {A,X}BGR 2'b11: BGR{A,X}	0x0
Reserved	[3]	Reserved	0x0
DstColorFormat	[2:0]	3'b000: XRGB_8888 3'b001: ARGB_8888 3'b010: RGB_565 3'b011: XRGB_1555 3'b100: ARGB_1555 3'b101: XRGB_4444 3'b110: ARGB_4444 3'b111: PACKED_RGB_888	0x0

13.5.6.5 Destination Left Top Coordinate Register (DST_LEFT_TOP_REG, R/W, Address = 0xFA00_0410)

DST_LEFT_TOP_REG	Bit	Description	Initial State
Reserved	[31:29]	Reserved	0x0
DstTopY	[28:16]	Left Top Y Coordinate of destination image Range: 0 ~ 8000 (Requirement: DstTopY < DstBottomY)	0x0
Reserved	[15:13]	Reserved	0x0
DstLeftX	[12:0]	Left Top X Coordinate of destination image Range: 0 ~ 8000 (Requirement: DstLeftX < DstRightX)	0x0

13.5.6.6 Destination Right Bottom Coordinate Register (DST_RIGHT_BOTTOM_REG, R/W, Address = 0xFA00_0414)

DST_RIGHT_BOTTOM_REG	Bit	Description	Initial State
Reserved	[31:29]	Reserved	0x0
DstBottomY	[28:16]	Right Bottom Y Coordinate of destination image Range: 0 ~ 8000	0x0
Reserved	[15:13]	Reserved	0x0
DstRightX	[12:0]	Right Bottom X Coordinate of destination image Range: 0 ~ 8000	0x0

13.5.7 PARAMETER SETTING REGISTERS (PATTERN)

13.5.7.1 Pattern Image Base Address Register (PAT_BASE_ADDR_REG, W, Address = 0xFA00_0500)

PAT_BASE_ADDR_REG	Bit	Description	Initial State
PatAddr	[31:0]	Base address of the pattern image	0x0

13.5.7.2 Pattern Image Size Register (PAT_SIZE_REG, R/W, Address = 0xFA00_0504)

PAT_SIZE_REG	Bit	Description	Initial State
Reserved	[31:29]	Reserved	0x0
PatHeight	[28:16]	Height of pattern image. Range: 1 ~ 8000	0x1
Reserved	[15:13]	Reserved	0x0
PatWidth	[12:0]	Width of pattern image. Range: 1 ~ 8000.	0x1

13.5.7.3 Pattern Image Color Mode Register (PAT_COLOR_MODE_REG, R/W, Address = 0xFA00_0508)

PAT_COLOR_MODE_REG	Bit	Description	Initial State
Reserved	[31:6]	Reserved	0x0
PatChannelOrder	[5:4]	2'b00: {A,X}RGB 2'b01: RGB{A,X} 2'b10: {A,X}BGR 2'b11: BGR{A,X}	0x0
Reserved	[3]	Reserved	0x0
PatColorFormat	[2:0]	3'b000: XRGB_8888 3'b001: ARGB_8888 3'b010: RGB_565 3'b011: XRGB_1555 3'b100: ARGB_1555 3'b101: XRGB_4444 3'b110: ARGB_4444 3'b111: PACKED_RGB_888	0x0

13.5.7.4 Pattern Offset Register (PAT_OFFSET_REG, R/W, Address = 0xFA00_050C)

PAT_OFFSET_REG	Bit	Description	Initial State
Reserved	[31:29]	Reserved	0x0
PatOffsetY	[28:16]	Y value of pattern offset. Range: 1 ~ 8000	0x0
Reserved	[15:13]	Reserved	0x0
PatOffsetX	[12:0]	X value of pattern offset. Range: 1 ~ 8000.	0x0

13.5.7.5 Pattern Stride Register (PAT_STRIDE_REG, R/W, Address = 0xFA00_0510)

PAT_STRIDE_REG	Bit	Description	Initial State
Reserved	[31:16]	Reserved	0x0
PatStride	[15:0]	Pattern stride (2's complement value)	0x0

13.5.8 PARAMETER SETTING REGISTERS (MASK)

13.5.8.1 Mask Base Address Register (MASK_BASE_ADDR_REG, W, Address = 0xFA00_0520)

MASK_BASE_ADDR_REG	Bit	Description	Initial State
MaskAddr	[31:0]	Base address of the mask image	0x0

13.5.8.2 Mask Stride Register (MASK_STRIDE_REG, R/W, Address = 0xFA00_0524)

MASK_STRIDE_REG	Bit	Description	Initial State
Reserved	[31:16]	Reserved	0x0
MaskStride	[15:0]	Mask stride (2's complement value).	0x0

NOTE: - MaskLeftX, MaskTopY, MaskRightX, and MaskBottomY are same as source image
- FIMG-2D V3.0 supports only 1bpp mask image format.

13.5.9 PARAMETER SETTING REGISTERS (CLIPPING WINDOW)

13.5.9.1 LeftTop Clipping Window Register (CW_LT_REG, R/W, Address = 0xFA00_0600)

CW_LT_REG	Bit	Description	Initial State
Reserved	[31:29]	Reserved	0x0
CWTopY	[28:16]	Top Y Clipping Window Requirement: DstTopY <= CWTopY < CWBOTTOMY	0x0
Reserved	[15:13]	Reserved	0x0
CWLeftX	[12:0]	Left X Coordinate of Clipping Window. Requirement: DstLeftX <= CWLeftX < CWRIGHTX	0x0

13.5.9.2 RightBottom Clipping Window Register (CW_RB_REG, R/W, Address = 0xFA00_0604)

CW_RB_REG	Bit	Description	Initial State
Reserved	[31:29]	Reserved	0x0
CWBOTTOMY	[28:16]	Bottom Y Clipping Window Requirement: CWBOTTOMY <= DstBottomY	0x0
Reserved	[15:13]	Reserved	0x0
CWRIGHTX	[12:0]	Right X Clipping Window Requirement: CWRIGHTX <= DstRightX	0x0

13.5.10 PARAMETER SETTING REGISTERS (ROP & ALPHA SETTING)

13.5.10.1 Third Operand Selection Register (THIRD_OPERAND_REG, R/W, Address = 0xFA00_0610)

THIRD_OPERAND_REG	Bit	Description	Initial State
Reserved	[31:6]	Reserved	0x0
MaskedSelect	[5:4]	2'00: Pattern 2'01: Foreground color 2'10: Background color Others: Reserved	0x0
Reserved	[3:2]	Reserved	-
UnmaskedSelect	[1:0]	2'00: Pattern 2'01: Foreground color 2'10: Background color Others: Reserved	0x0

13.5.10.2 Raster Operation Register (ROP4_REG, R/W, Address = 0xFA00_0614)

ROP4_REG	Bit	Description	Initial State
Reserved	[31:16]	Reserved	0x0
MaskedROP3	[15:8]	Raster Operation Value	0x0
UnmaskedROP3	[7:0]	Raster Operation Value	0x0

13.5.10.3 Alpha Register (ALPHA_REG, R/W, Address = 0xFA00_0618)

ALPHA_REG	Bit	Description	Initial State
Reserved	[31:16]	Reserved	0x0
FadingOffset	[15:8]	Fading Offset Value	0x0
AlphaValue	[7:0]	Alpha Value	0x0

13.5.11 PARAMETER SETTING REGISTERS (COLOR)

13.5.11.1 Foreground Color Register (FG_COLOR_REG, R/W, Address = 0xFA00_0700)

The color format of the foreground color is the same as the destination color format.

FG_COLOR_REG	Bit	Description	Initial State
ForegroundColor	[31:0]	Foreground Color Value. The alpha field of the foreground color will be discarded.	0x0

13.5.11.2 Background Color Register (BG_COLOR_REG, R/W, Address = 0xFA00_0704)

The color format of the background color is the same as the destination color format.

BG_COLOR_REG	Bit	Description	Initial State
BackgroundColor	[31:0]	Background Color Value. The alpha field of the background color will be discarded.	0x0

13.5.11.3 BlueScreen Color Register (BS_COLOR_REG, R/W, Address = 0xFA00_0708)

The color format of the bluescreen color is generally the same as the source color format.

But if the source color is selected as the foreground color or the background color, the color format of the bluescreen color is the destination color format because the color format of the foreground and the background are the destination color format.

BS_COLOR_REG	Bit	Description	Initial State
BlueScreenColor	[31:0]	BlueScreen Color Value. The alpha field of the blue screen color will be discarded.	0x0

13.5.12 PARAMETER SETTING REGISTERS (COLOR KEY)

13.5.12.1 Source Colorkey Control Register (SRC_COLORKEY_CTRL_REG, R/W, Address = 0xFA00_0710)

SRC_COLORKEY_CTRL_REG	Bit	Description	Initial State
Reserved	[31:17]	Reserved	0x0
SrcStencilInv	[16]	0 = Normal stencil test 1 = Inversed stencil test	0x0
Reserved	[15:13]	Reserved	
SrcStencilOnA	[12]	0 = Stencil Test Off for A value 1 = Stencil Test On for A value	0x0
Reserved	[11:9]	Reserved	
SrcStencilOnR	[8]	0 = Stencil Test Off for R value 1 = Stencil Test On for R value	0x0
Reserved	[7:5]	Reserved	
SrcStencilOnG	[4]	0 = Stencil Test Off for G value 1 = Stencil Test On for G value	0x0
Reserved	[3:1]	Reserved	
SrcStencilOnB	[0]	0 = Stencil Test Off for B value 1 = Stencil Test On for B value	0x0

13.5.12.2 Source Colorkey Decision Reference Minimum Register (SRC_COLORKEY_DR_MIN_REG, R/W, Address = 0xFA00_0714)

The color format of source colorkey decision reference register is generally the same as the source color format.

But if the source color is selected as the foreground color or the background color, the color format of source colorkey register is the destination color format because the color format of the foreground and the background are the destination color format.

SRC_COLORKEY_DR_MIN_REG	Bit	Description	Initial State
SrcDRMinA	[31:24]	Alpha DR MIN value	0x0
SrcDRMinR	[23:16]	Red DR MIN value	0x0
SrcDRMinG	[15:8]	Green DR MIN value	0x0
SrcDRMinB	[7:0]	Blue DR MIN value	0x0

13.5.12.3 Source Colorkey Decision Reference Maximum Register (SRC_COLORKEY_DR_MAX_REG, R/W, Address = 0xFA00_0718)

The color format of source colorkey decision reference register is generally the same as the source color format.

But if the source color is selected as the foreground color or the background color, the color format of source colorkey register is the destination color format because the color format of the foreground and the background are the destination color format.

SRC_COLORKEY_DR_MAX_REG	Bit	Description	Initial State
SrcDRMaxA	[31:24]	Alpha DR MAX value	0xFF
SrcDRMaxR	[23:16]	Red DR MAX value	0xFF
SrcDRMaxG	[15:8]	Green DR MAX value	0xFF
SrcDRMaxB	[7:0]	Blue DR MAX value	0xFF

13.5.12.4 Destination Colorkey Control Register (DST_COLORKEY_CTRL_REG, R/W, Address = 0xFA00_071C)

DST_COLORKEY_CTRL_REG	Bit	Description	Initial State
Reserved	[31:17]	Reserved	0x0
DstStencilInv	[16]	0 = Normal stencil test 1 = Inversed stencil test	0x0
Reserved	[15:13]	Reserved	
DstStencilOnA	[12]	0 = Stencil Test Off for A value 1 = Stencil Test On for A value	0x0
Reserved	[11:9]	Reserved	
DstStencilOnR	[8]	0 = Stencil Test Off for R value 1 = Stencil Test On for R value	0x0
Reserved	[7:5]	Reserved	
DstStencilOnG	[4]	0 = Stencil Test Off for G value 1 = Stencil Test On for G value	0x0
Reserved	[3:1]	Reserved	
DstStencilOnB	[0]	0 = Stencil Test Off for B value 1 = Stencil Test On for B value	0x0

**13.5.12.5 Destination Colorkey Decision Reference Minimum Register
(DST_COLORKEY_DR_MIN_REG, R/W, Address = 0xFA00_0720)**

The color format of destination colorkey decision reference register is the same as the destination color format.

DST_COLORKEY_DR_MIN_REG	Bit	Description	Initial State
DstDRMinA	[31:24]	Alpha DR MIN value	0x0
DstDRMinR	[23:16]	Red DR MIN value	0x0
DstDRMinG	[15:8]	Green DR MIN value	0x0
DstDRMinB	[7:0]	Blue DR MIN value	0x0

**13.5.12.6 Destination Colorkey Decision Reference Maximum Register
(DST_COLORKEY_MAX_REG, R/W, Address = 0xFA00_0724)**

The color format of destination colorkey decision reference register is the same as the destination color format.

DST_COLORKEY_MAX_REG	Bit	Description	Initial State
DstDRMaxA	[31:24]	Alpha DR MAX value	0xFF
DstDRMaxR	[23:16]	Red DR MAX value	0xFF
DstDRMaxG	[15:8]	Green DR MAX value	0xFF
DstDRMaxB	[7:0]	Blue DR MAX value	0xFF

Section 10

AUDIO / ETC

Table of Contents

1	Audio Subsystem	1-1
1.1	Overview of the Audio Subsystem.....	1-1
1.2	Key Features of Audio Subsystem.....	1-1
1.3	Input/ Output Description	1-2
1.4	Block Diagram of Audio Subsystem.....	1-3
1.5	Functional Description	1-5
1.5.1	Reconfigurable Processor	1-5
1.5.2	ASS CLK CON.....	1-5
1.5.3	Commbox	1-5
1.5.4	I2S_V51	1-6
1.5.5	SRAM	1-6
1.6	Register Description.....	1-7
1.6.1	Register Map	1-7
1.6.2	Audio Subsystem CLK CON.....	1-9
1.6.3	Commbox	1-11
2	IIS Multi Audio Interface	2-1
2.1	Overview of IIS Multi Audio Interface.....	2-1
2.2	Key Features of IIS Multi Audio Interface	2-1
2.3	Block Diagram of IIS Multi Audio Interface	2-2
2.4	Functional Descriptions.....	2-3
2.4.1	Master/Slave Mode.....	2-3
2.5	Audio Serial Data Format.....	2-8
2.5.1	IIS-bus Format	2-8
2.5.2	MSB (Left) Justified	2-8
2.5.3	LSB (Right) Justified	2-9
2.6	PCM BIT Length (BLC), RFS Divider and BFS Divider for Sampling Frequency (iISrcclk), SERIAL bitCLK(iiSSclk), and ROOt Clock(RCLK).....	2-10
2.6.1	PCM Word Length and BFS Divider.....	2-10
2.6.2	BFS Divider and RFS Divider	2-10
2.6.3	RFS Divider and ROOT CLock.....	2-11
2.7	Programming Guide	2-12
2.7.1	Initialization	2-12
2.7.2	Play Mode (TX mode) with DMA	2-12
2.7.3	Recording Mode (RX mode) with DMA	2-12
2.7.4	Example Code	2-13
2.8	IO Description	2-19
2.9	Register Description.....	2-20
2.9.1	Register Map	2-20
3	IIS-Bus Interface	3-1
3.1	Overview of IIS-Bus Interface	3-1
3.2	Key Features of IIS-Bus Interface.....	3-1
3.3	Block Diagram of IIS-Bus Interface.....	3-2
3.4	Functional Descriptions.....	3-3
3.4.1	Master/Slave Mode.....	3-3
3.4.2	DMA Transfer	3-4

3.4.3 Audio Serial DATA Format	3-4
3.4.4 PCM Word Length and BFS Divider.....	3-6
3.4.5 BFS Divider and RFS Divider	3-6
3.4.6 RFS Divider and ROOT Clock.....	3-7
3.5 Programming Guide	3-8
3.5.1 Initialization	3-8
3.5.2 Play Mode (TX mode) with DMA	3-8
3.5.3 Recording Mode (RX mode) with DMA	3-8
3.5.4 Example Code	3-9
3.6 I/O Description	3-14
3.7 Register Description.....	3-15
3.7.1 Register Map	3-15
4 AC97 Controller	4-1
4.1 Overview of AC97 Controller	4-1
4.2 Key Features of AC97 Controller	4-1
4.3 AC97 Controller Operation.....	4-2
4.3.1 Block Diagram of AC97 Controller.....	4-2
4.3.2 Internal Data Path.....	4-3
4.3.3 Operation Flow Chart	4-4
4.3.4 AC-link Digital Interface Protocol.....	4-5
4.3.5 AC-link Input Frame (SDATA_IN).....	4-7
4.3.6 AC97 Power-down.....	4-9
4.4 I/O Description	4-12
4.5 Register Description.....	4-13
4.5.1 Register Map	4-13
5 PCM Audio Interface	5-1
5.1 Overview of PCM Audio Interface	5-1
5.2 Key Features of PCM Audio Interface	5-1
5.3 PCM Audio Interface.....	5-2
5.4 PCM Timing	5-3
5.5 I/O Description	5-5
5.6 Register Description.....	5-6
5.6.1 Register Map	5-6
6 SPDIF Transmitter	6-1
6.1 Overview of SPDIF Transmitter	6-1
6.2 Key Features of SPDIF Transmitter.....	6-1
6.3 Block Diagram of SPDIF Transmitter.....	6-2
6.4 Functional Descriptions.....	6-3
6.4.1 Data Format of SPDIF	6-3
6.4.2 Channel Coding	6-5
6.4.3 Preamble	6-5
6.4.4 Non-Linear PCM Encoded Source (IEC 61937).....	6-6
6.4.5 SPDIF Operation	6-7
6.4.6 Shadowed Register	6-8
6.5 I/O Description	6-9
6.6 Register Description.....	6-10
6.6.1 Register Map	6-10

7 ADC & Touch Screen Interface	7-1
7.1 Overview of ADC & TOUCH Screen Interface.....	7-1
7.2 KEY Features of ADC & TOUCH Screen Interface	7-1
7.3 TOUCH Screen Interface Operation	7-2
7.3.1 Block Diagram ADC & TOUCH Screen Interface.....	7-2
7.4 Function Descriptions	7-3
7.4.1 A/D Conversion Time	7-3
7.4.2 Touch Screen Interface Mode	7-3
7.4.3 Standby Mode.....	7-5
7.4.4 Two Touch Screen Interfaces.....	7-5
7.5 ADC & Touch Screen Interface Input Clock Diagram.....	7-7
7.6 I/O Descriptions.....	7-8
7.7 Register Description.....	7-9
7.7.1 Register Map	7-9
8 Keypad Interface	8-1
8.1 Overview of Keypad Interface.....	8-1
8.2 Debouncing Filter.....	8-3
8.3 Filter Clock	8-4
8.4 Wakeup Source.....	8-4
8.5 Keypad Scanning Procedure	8-5
8.6 I/O Description	8-9
8.7 Register Description.....	8-11
8.7.1 Register Map	8-11

List of Figures

Figure Number	Title	Page Number
Figure 1-1	Block Diagram of Audio Subsystem	1-3
Figure 1-2	Audio Subsystem Block Diagram	1-4
Figure 1-3	Clock Controller in Audio Subsystem	1-5
Figure 2-1	IIS-Bus Block Diagram.....	2-2
Figure 2-2	Clock Controller in Audio Sub-System	2-3
Figure 2-3	IIS Clock Control Block Diagram	2-4
Figure 2-4	Master/Slave Modes of IIS.....	2-5
Figure 2-5	Concept of Mixer in IIS	2-7
Figure 2-6	IIS Audio Serial Data Formats	2-9
Figure 2-7	TX FIFO Structure for BLC = 00 or BLC = 01.....	2-14
Figure 2-8	TX FIFO Structure for BLC = 10 (24-bit/channel).....	2-15
Figure 2-9	RX FIFO Structure for BLC = 00 or BLC = 01	2-17
Figure 2-10	RX FIFO Structure for BLC = 10 (24-bit/channel)	2-18
Figure 3-1	IIS-Bus Block Diagram.....	3-2
Figure 3-2	IIS Clock Control Block Diagram	3-3
Figure 3-3	IIS Audio Serial Data Formats	3-5
Figure 3-4	TX FIFO Structure for BLC = 00 or BLC = 01.....	3-9
Figure 3-5	TX FIFO Structure for BLC = 10 (24-bit/channel).....	3-10
Figure 3-6	RX FIFO Structure for BLC = 00 or BLC = 01	3-12
Figure 3-7	RX FIFO Structure for BLC = 10 (24-bits/channel)	3-13
Figure 4-1	AC97 Block Diagram	4-2
Figure 4-2	Internal Data Path	4-3
Figure 4-3	AC97 Operation Flow Chart.....	4-4
Figure 4-4	Bi-directional AC-link Frame with Slot Assignments.....	4-5
Figure 4-5	AC-link Output Frame	4-6
Figure 4-6	AC-link Input Frame	4-8
Figure 4-7	AC97 Power-down Timing	4-9
Figure 4-8	AC97 Power down/Power up Flow	4-10
Figure 4-9	AC97 State Diagram.....	4-11
Figure 5-1	PCM timing, POS_MSB_WR/RD = 0	5-3
Figure 5-2	PCM timing, POS_MSB_WR/RD = 1	5-4
Figure 5-3	Input Clock Diagram for PCM	5-4
Figure 6-1	Block Diagram of SPDIFOUT	6-2
Figure 6-2	SPDIF Frame Format	6-3
Figure 6-3	SPDIF Sub-frame Format	6-4
Figure 6-4	Channel Coding	6-5
Figure 6-5	Format of Burst Payload	6-6
Figure 7-1	ADC and Touch Screen Interface Functional Block Diagram	7-2

Figure 7-2	ADC and Touch Screen Operation Signal.....	7-6
Figure 7-3	Input Clock Diagram for ADC & Touch Screen Interface	7-7
Figure 8-1	Key Matrix Interface External Connection Guide.....	8-2
Figure 8-2	Internal Debouncing Filter Operation.....	8-3
Figure 8-3	Keypad Scanning Procedure	8-5
Figure 8-4	Keypad Scanning Procedure II	8-6
Figure 8-5	Keypad Scanning Procedure III	8-6
Figure 8-6	Keypad Scanning Procedure when the two-key Pressed with Different Row	8-7
Figure 8-7	Keypad I/F Block Diagram	8-8

List of Tables

Table Number	Title	Page Number
Table 2-1	Typical Usage of Master/Slave Modes	2-5
Table 2-2	Allowed BFS Value as BLC	2-10
Table 2-3	Allowed RFS Value as BFS	2-10
Table 2-4	Root Clock Table (MHz).....	2-11
Table 3-1	Allowed BFS Value as BLC	3-6
Table 3-2	Allowed RFS Value as BFS	3-6
Table 3-3	Root Clock Table (MHz).....	3-7
Table 4-1	Input Slot 1 Bit Definitions.....	4-7
Table 6-1	Burst Preamble Words.....	6-6
Table 8-1	Keypad interface I/O Description	8-9

1

AUDIO SUBSYSTEM

1.1 OVERVIEW OF THE AUDIO SUBSYSTEM

The audio subsystem is a special subsystem that supports playback of sound with low power. In other words, this subsystem is used to increase the playback time. It consists of a reconfigurable processor (RP) DSP core, I2S (v5.1) including AHB master port, and 214KB SRAM. Combined with low power scheme, the S5PV210 is designed to play audio with extremely low power.

1.2 KEY FEATURES OF AUDIO SUBSYSTEM

The key features of audio subsystem include:

- Renders low power music play
- Small and adequate for audio application RP
- Supports various audio codecs
- Contains I2S (including AHB master port) to get data from subsystem internal SRAM
- Contains totally 214KB SRAM (160KB continuous) for audio subsystem's internal and external usage

* To know more about the available audio codecs, contact SEC.

1.3 INPUT/ OUTPUT DESCRIPTION

Signal	I/O	Description	Pad	Type
Xi2s0SCLK	I/O	Specifies bit clock input.	Xi2s0SCLK	Dedicated
Xi2s0LRCLK	I/O	Specifies LR channel clock input.	Xi2s0LRCLK	Dedicated
Xi2s0CDCLK	I/O	Specifies codec clock out.	Xi2s0CDCLK	Dedicated
Xi2s0SDI	I	Specifies I2S serial data input.	Xi2s0SDI	Dedicated
Xi2s0SDO0	O	Specifies I2S serial data out 0.	Xi2s0SDO0	Dedicated
Xi2s0SDO1	O	Specifies I2S serial data out 1.	Xi2s0SDO1	Dedicated
Xi2s0SDO2	O	Specifies I2S serial data out 2.	Xi2s0SDO2	Dedicated

1.4 BLOCK DIAGRAM OF AUDIO SUBSYSTEM

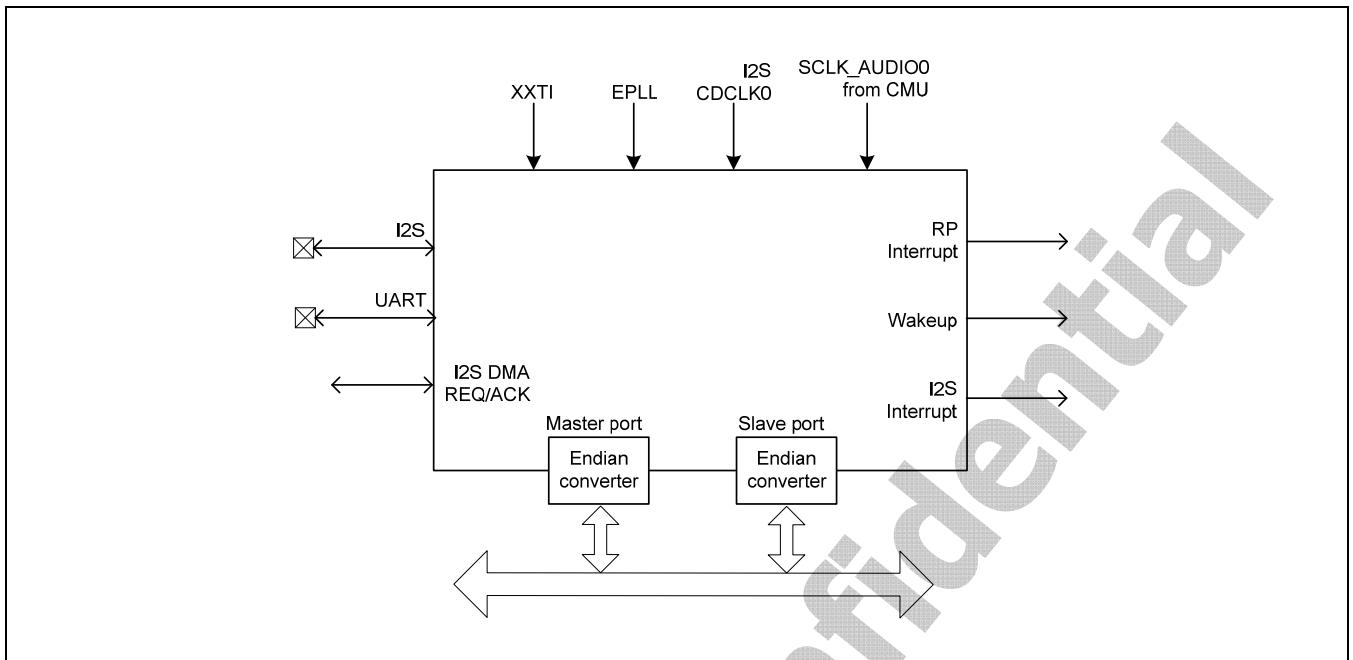


Figure 1-1 Block Diagram of Audio Subsystem

There are two modes for turning the power on and off, namely, normal mode and deep-idle mode.

Both audio subsystem and TOP (S5PV210) work in the Normal mode. However, to save power, you can turn off the TOP. Turning off the TOP means that the S5PV210 goes into the Deep-idle mode. In this mode, the audio subsystem can remain on. The audio subsystem can wake up the TOP (S5PV210) from deep-idle mode, using wake up source such as RP and I2S V51.

The audio subsystem receives four kinds of clock from CMU, namely, XXTI, EPLL, I2SCDCLK0, and SCLK_AUDIO0. The XXTI is a selected clock between main OSC(XXTI) and USB OSC(XusbXTI) which is selected by OM[0].

XXTI, EPLL, and I2SCDCLK0 can be supplied to the audio subsystem when the audio system is on and TOP (S5PV210) is off.

The internal RP uses big-endian scheme, while the external audio subsystem uses little-endian scheme. Therefore, endian converters are used to convert data ordering without using ARM or RP.

Audio subsystem comprises of I2S V5.1 and its own interrupt, and DMA REQ/ ACK ports.

The master port accesses DRAM and IRAM using bus-master modules in audio subsystem.

On the other hand, the slave port accesses all modules in audio subsystem using external audio subsystem modules. All bus-master modules in S5PV210 can access modules in audio subsystem (excluding RP core).

UART for RP debugging and UART pad are muxed with system UART2. For more information, refer to GPIO User's Manual.

NOTE: For more information about power modes such as Deep-idle, refer to Power Management Unit (PMU) User's Manual.

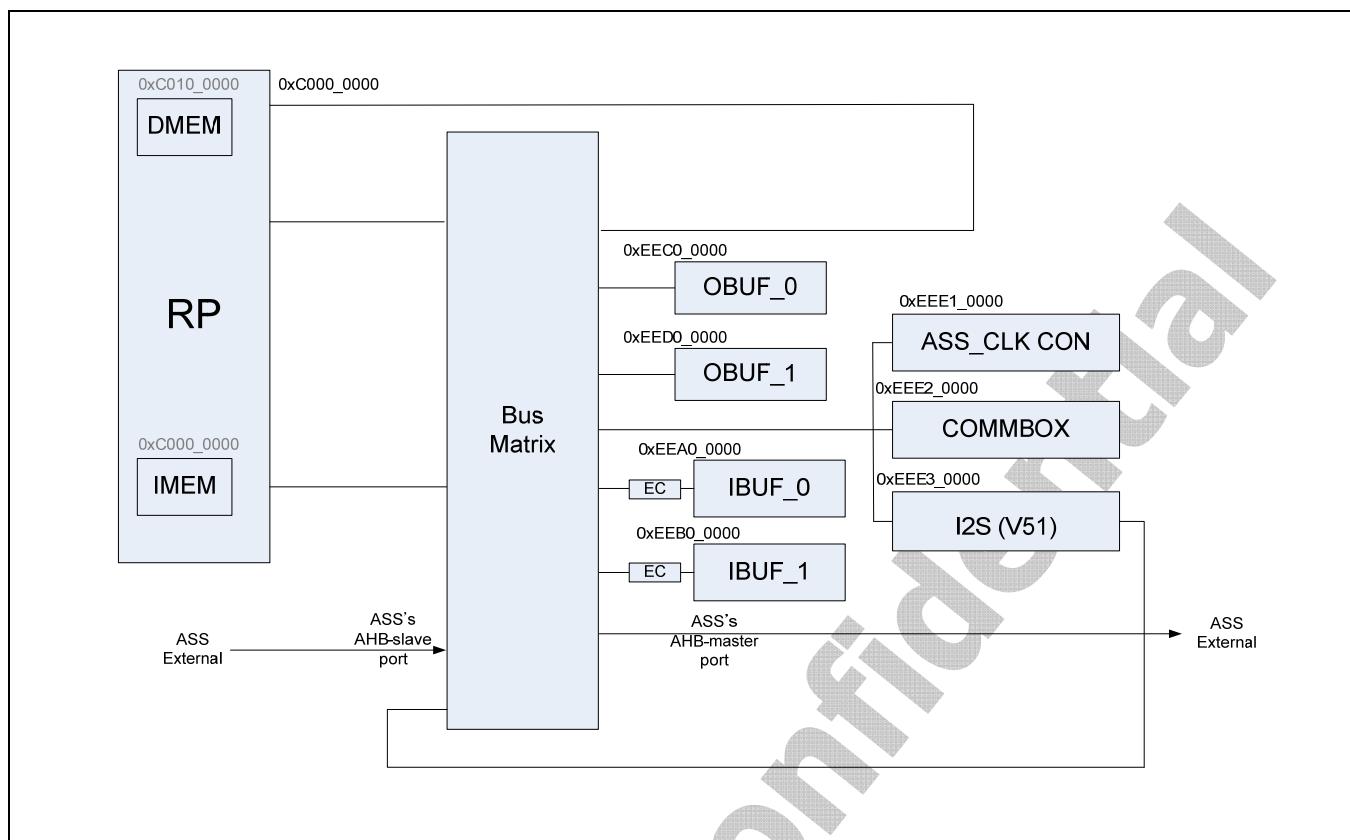


Figure 1-2 Audio Subsystem Block Diagram

The audio subsystem comprises of the following blocks:

- RP: Specifies the DSP core, which is an audio-dedicated DSP for S5PV210.
- IMEM and DMEM: IMEM is used for instruction cache of RP, while DMEM is used for data memory of RP. If RP is not used, the external modules in audio subsystem use IMEM and DMEM for SRAM (IMEM: 64KB and DMEM: 96KB).
- IBUF_0, IBUF_1: Specifies input buffers 0 and 1. 18KB is allocated for each input buffers in RP. These buffers are used by external audio subsystem as data reservoir. There are Endian Converters (ECs) besides IBUF_0 and IBUF_1.
- OBUF_0, OBUF_1: Specifies output buffers 0 and 1. 9KB is allocated for each output buffers in RP. These buffers can be used by external audio subsystem as data reservoir.
- ASS_CLKCON: Specifies the internal clock controller in audio subsystem.
- COMMBOX: Specifies the communication channel between ARM and RP.
- I2S V51: Specifies the main I2S module of S5PV210.

1.5 FUNCTIONAL DESCRIPTION

1.5.1 RECONFIGURABLE PROCESSOR

Reconfigurable Processor (RP) is a Samsung proprietary configurable DSP core. RP in S5PV210 is configured for low power audio applications.

1.5.2 ASS CLK CON

ASS CLK CON specifies the clock controller for audio subsystem. It also provides clock for modules in the audio subsystem.

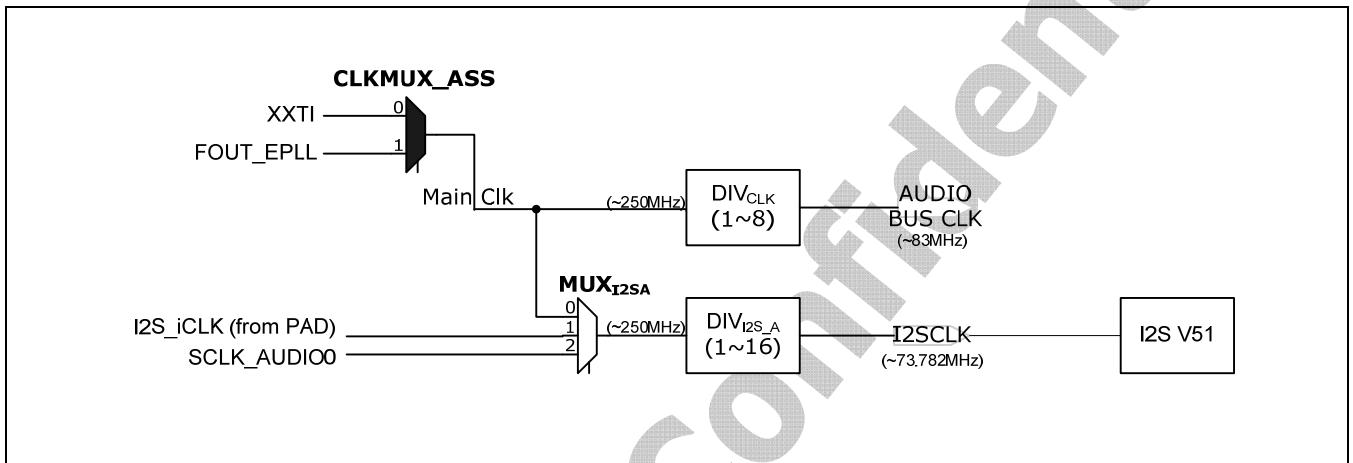


Figure 1-3 Clock Controller in Audio Subsystem

CLKMUX_ASS can be changed at any time. However, it should stop running before MUXI2S_A is changed. The value of divider can also be changed at any time.

1.5.3 COMMBOX

COMMBOX specifies the communication box. It denotes the SFR communication channel between ARM and RP.

1.5.4 I2S_V51

The major difference between I2S V3.2 and I2S V5.1 lies in the presence of small AHB DMA. The previous version of I2S did not allow the audio data to be played without external DMA. I2S V5.1 allows the audio data to play with its own DMA. I2S DMA issues only 32-bit single read transaction.

I2S V5.1 comprises of an interrupt request signal to wake up ARM if S5PV210 is in the Idle and Deep-idle modes. Interrupt request signal occurs if the pre-defined configuration of I2S DMA operations is complete. After CPU wakes up, CPU prepares, generates, and saves the next audio data to be played in SRAM at audio subsystem. Then CPU is powered off again to save power.

For more information, refer to the I2S_V51 User's Manual.

1.5.5 SRAM

The total memory size in audio subsystem is 214KB SRAM, out of which 64KB is reserved for IMEM, 96KB is reserved for DMEM, 36KB is reserved for IBUF0 and IBUF1, and 18KB is reserved for OBUF0 and OBUF1. These memories can be accessed by both internal and external modules in audio subsystem.

For more information on how to use this SRAM for low power music play, refer to Section 6, "Programming Guide". If low power audio functionality is not needed, 214KB SRAM can be used as data reservoir.

1.6 REGISTER DESCRIPTION

1.6.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
Audio Subsystem Internal Memory				
IMEM	0xC000_0000 ~0xC000_FFFC	R/W	For RP, instruction cache For external audio subsystem, 64KB SRAM	-
DMEM	ARM decode mode (MISC SFR[3]=0) 0xC001_0000 ~0xC002_7FFC RP decode mode (MISC SFR[3]=1) 0xC010_0000 ~0xC011_7FFC	R/W	For RP, data memory For external audio subsystem, 96KB SRAM	-
IBUF0	0xEEA0_0000 ~0xEEA0_47FC	R/W	For RP, input buffer 0 For external audio subsystem, 18KB SRAM	-
IBUF1	0xEEB0_0000 ~0xEEB0_47FC	R/W	For RP, input buffer 1 For external audio subsystem, 18KB SRAM	-
OBUF0	0xEEC0_0000 ~0xEEC0_23FC	R/W	For RP, output buffer 0 For external audio subsystem, 9KB SRAM	-
OBUF1	0xEED0_0000 ~0xEED0_23FC	R/W	For RP, output buffer 1 For external audio subsystem, 9KB SRAM	-
Audio Subsystem CLK CON				
ASS CLK SRC	0EEE1_0000	R/W	Specifies the clock source select register.	0x0
ASS CLK DIV	0EEE1_0004	R/W	Specifies the clock divider register.	0x0
ASS CLK GATE	0EEE1_0008	R/W	Specifies the clock gate register.	0x7f
Combbox				
ASS_INTR	0EEE2_0000	R/W	Specifies the interrupt from audio subsystem to ARM. Also, it can be used as wake up source.	0x0
SW_DEFINE00	0EEE2_0004	R/W	Specifies an SFR that can be freely used in the application.	0x0
SW_DEFINE01	0EEE2_0008	R/W	Specifies an SFR that can be freely used in the application.	0x0
SW_DEFINE02	0EEE2_000C	R/W	Specifies an SFR that can be freely used in the application.	0x0
SW_DEFINE03	0EEE2_0010	R/W	Specifies an SFR that can be freely used in the application.	0x0
INST_START_ADDR	0EEE2_0014	R/W	Specifies the instruction code start address for external booting.	0x0
SW_DEFINE04	0EEE2_0018	R/W	Specifies an SFR that can be freely used in the application.	0x0

Register	Address	R/W	Description	Reset Value
RESET	0xEEE2_0100	R/W	Specifies the software reset of audio sub-system.	0x1
RP_PENDING	0xEEE2_0104	R/W	Specifies the pending control of RP.	0x1
FRM_SIZE	0xEEE2_0108	R/W	Specifies the frame size (word) per output buffer (OBUF0, 1). Upper 20-bits can be used freely.	0x0
SW_DEFINE05	0xEEE2_010C	R/W	Specifies an SFR that can be freely used in the application.	0x0
SW_DEFINE06	0xEEE2_0110	R/W	Specifies an SFR that can be freely used in the application.	0x0
SW_DEFINE07	0xEEE2_0114	R/W	Specifies an SFR that can be freely used in the application.	0x0
SW_DEFINE08	0xEEE2_0118	R/W	Specifies an SFR that can be freely used in the application.	0x0
SW_DEFINE09	0xEEE2_011C	R/W	Specifies an SFR that can be freely used in the application.	0x0
SW_DEFINE10	0xEEE2_0120	R/W	Specifies an SFR that can be freely used in the application.	0x0
SW_DEFINE11	0xEEE2_0124	R/W	Specifies an SFR that can be freely used in the application.	0x0
SW_DEFINE12	0xEEE2_0128	R/W	Specifies an SFR that can be freely used in the application.	0x0
RP_BOOT	0xEEE2_012C	R/W	Controls the RP booting type. Upper 31 bits can be used freely.	0x0
SW_DEFINE13	0xEEE2_0130	R/W	Specifies an SFR that can be freely used in the application.	0x0
SW_DEFINE14	0xEEE2_0134	R/W	Specifies an SFR that can be freely used in the application.	0x0
SW_DEFINE15	0xEEE2_0138	R/W	Specifies an SFR that can be freely used in the application.	0x0
PAD_PDN_CTRL	0xEEE2_0204	R/W	Controls GPIO PDN for power down.	0x0
MISC	0xEEE2_0208	R/W	Specifies the endian converter. Also, selects the audio decoder.	0x0

1.6.2 AUDIO SUBSYSTEM CLK CON

To set the registers of audio subsystem CLK CON, refer to [Figure 1-3](#).

1.6.2.1 Audio Subsystem Clock Source Register (Audio Subsystem CLK SRC, R/W, Address = 0xEEE1_0000)

AUDIO SUBSYSTEM CLK SRC	Bit	Description	Initial State
Reserved	[31:4]	Reserved	0
MUX_I2S_A	[3:2]	10 = SCLK_AUDIO0 01 = IISCDCLK0 (from PAD) 00 = Main CLK	0
Reserved	[1]	This bit must be set as 0	0
CLKMUX_ASS	[0]	1 = FOUT_EPLL 0 = XXTI	0

1.6.2.2 Audio Subsystem Clock Divider Register (Audio Subsystem CLK DIV, R/W, Address = 0xEEE1_0004)

AUDIO SUBSYSTEM CLK DIV	Bit	Description	Initial State
Reserved	[31:8]	Reserved	0
I2S_A_RATIO	[7:4]	Specifies the I2S_A clock divider ratio. $I2SCLK = MOUT_{I2S_A} / (I2S_A_RATIO+1)$	0
AUDIO_BUS_CLK_RATIO	[3:0]	Specifies the AUD_BUS clock divider ratio. $AUDIO_BUS_CLK = MOUT_{AUD_BUS} / (AUD_BUS_RATIO+1)$	0

1.6.2.3 Audio Subsystem Clock Gate Register (Audio Subsystem CLK GATE R/W, Address = 0xEEE1_0008)

AUDIO SUBSYSTEM CLK GATE	Bit	Description	Initial State
Reserved	[31:7]	Reserved	0
CLK_I2S	[6]	Specifies the gating clock of I2SCLK to I2S (0: mask, 1: pass).	1
AUDIO_BUS_CLK_I2S	[5]	Specifies the gating AUDIO BUS CLK to I2S (0: mask, 1: pass).	1
AUDIO_BUS_CLK_UART	[4]	Specifies the gating AUDIO BUS CLK to UART (0: mask, 1: pass).	1
AUDIO_BUS_CLK_HWA	[3]	Specifies the gating AUDIO BUS CLK to HWA (0: mask, 1: pass).	1
AUDIO_BUS_CLK_DMA	[2]	Specifies the gating AUDIO BUS CLK to DMA (0: mask, 1: pass).	1
AUDIO_BUS_CLK_BUF	[1]	Specifies the gating AUDIO BUS CLK to BUF (IBUF0, 1/ OBUF0, 1) (0: mask, 1: pass).	1
AUDIO_BUS_CLK_RP	[0]	Specifies the gating AUDIO BUS CLK to RP (including IMEM and DMEM) (0: mask, 1: pass).	1

1.6.3 COMMBOX

Each SW_DEFINE is a 32-bit register, and can be freely used for application.

1.6.3.1 Audio Subsystem Interrupt Register (ASS_INTR, R/W, Address = 0xEEE2_0000)

ASS_INTR	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0
Interrupt	[0]	1 = Interrupt and wake-up 0 = No action	0

1.6.3.2 Instruction Start Address Register (INST_START_ADDR, R/W, Address = 0xEEE2_0014)

INST_START_ADDR	Bit	Description	Initial State
Address	[31:0]	Specifies the instruction code start address for external booting.	0

1.6.3.3 Reset Register (RESET, R/W, Address = 0xEEE2_0100)

RESET	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0
Reset	[0]	1 = No action 0 = Resets and returns to 1 after two clock cycles	1

1.6.3.4 RP Pending Register (RP_PENDING, R/W, Address = 0xEEE2_0104)

RP_PENDING	Bit	Description	Initial State
Reserved	[31:1]	Reserved	0
Pending	[0]	1 = RP pending 0 = RP running	1

1.6.3.5 FRAME Size Register (FRM_SIZE, R/W, Address = 0xEEE2_0108)

FRM_SIZE	Bit	Description	Initial State
SW_DEFINE	[31:12]	Specifies an SFR that can be freely used in the application.	0
Frame size	[11:0]	Specifies the frame size (word) per output buffer.	0

1.6.3.6 RP Boot Register (RP_BOOT, R/W, Address = 0xEEE2_012C)

RP_BOOT	Bit	Description	Initial State
SW_DEFINE	[31:1]	Specifies an SFR that can be freely used in the application.	0
RP boot	[0]	1 = Internal (Instruction is located in IMEM) 0 = External (Instruction is located in DRAM and start address is defined at INST_START_ADDR SFR)	0

At ARM decode mode, RP boot bit must be 1.

1.6.3.7 PAD Power Down Control Register (PAD_PDN_CTRL, R/W, Address = 0xEEE2_0204)

GPIO PDN controls power down. The value of each PAD set by this SFR is maintained at Sleep power mode.

PAD_PDN_CTRL	Bit	Description	Initial State
Reserved	[31:9]	Reserved	0
SDO_PDN[2]	[8]	Configure output value of I2S0 SDO[2] PAD 0 = Output 0 1 = Output 1	0
SDO_PDN[1]	[7]	Configure output value of I2S0 SDO[1] PAD 0 = Output 0 1 = Output 1	0
SDO_PDN[0]	[6]	Configure output value of I2S0 SDO[0] PAD 0 = Output 0 1 = Output 1	0
SCLKO_PDN	[5]	Configure output value of I2S0 SCLK PAD 0 = Output 0 1 = Output 1	0
CDCLKO_PDN	[4]	Configure output value of I2S0 CDCLK PAD 0 = Output 0 1 = Output 1	0
LRCLKO_PDN	[3]	Configure output value of I2S0 LRCLK PAD 0 = Output 0 1 = Output 1	0
SCLKO_EN_PDN	[2]	Configure direction of I2S0 SLCK PAD 0 = Input 1 = Output	0
CDCLKO_EN_PDN	[1]	Configure direction of I2S0 CDCLK PAD 0 = Input 1 = Output	0
LRCLKO_EN_PDN	[0]	Configure direction of I2S0 LRCLK PAD 0 = Input 1 = Output	0

1.6.3.8 MISC Register (MISC, R/W, Address = 0xEEE2_0208)

MISC	Bit	Description	Initial State
Reserved	[31:4]	Reserved	0
Audio decoder select	[3]	Specifies address range of DMEM 1 = 0xC010_0000 ~ 0xC011_7FFC, when RP is used for decoder 0 = 0xC001_0000 ~ 0xC002_7FFC, when ARM is used for decoder	0
System timer debug	[2]	Specifies usage of Xi2s0SDO1 and Xi2s0SDO2 pad 1 = Xi2s0SDO1 is used as monitor of system timer's tick, and Xi2s0SDO1 is used as monitor of system timer's interrupt 0 = Xi2s0SDO1 is used as I2SD1, and Xi2s0SDO2 is used as I2SD2	0
Endian converter	[1]	Specifies the endian converter for IBUF1 write path. 1 = Big endian 0 = Little endian	0
Endian converter	[0]	Specifies the endian converter for IBUF0 write path 1 = Big endian 0 = Little endian	0

There are endian converters between external and internal audio subsystems (refer to [Figure 1-1](#). To set them, AUDIO_ENDIAN SFR at Clock Management Unit (CMU) is used.

* ENDIAN converters setting guide

Register[Bit]	Path	Guide value
AUDIO_ENDIAN[3]	RP read	1
AUDIO_ENDIAN[2]	RP write	0
AUDIO_ENDIAN[1]	ARM read	0
AUDIO_ENDIAN[0]	ARM write	0
MISC[1]	IBUF1 write	1
MISC[0]	IBUF0 write	1

Since having ENDIAN converters, Read/write accesses must always be in 32-bit units (byte or half word accesses are not allowed).

For more details on I2S V51, refer to the I2S V51 User's Manual.

2 IIS MULTI AUDIO INTERFACE

2.1 OVERVIEW OF IIS MULTI AUDIO INTERFACE

Inter-IC Sound (IIS) is one of the popular digital audio interface. The IIS bus handles audio data and the other signals, namely, sub-coding and control, are transferred separately. It is possible to transmit data between two IIS bus. To minimize the number of pins required and to keep wiring simple, basically, a 3-line serial bus is used. This consists of a line for two time-multiplexed data channels, a word select line and a clock line.

IIS interface transmits or receives sound data from external stereo audio codec. To transmit and receive data, 32bitsx64 FIFOs (First-In-First-Out) for each channel are included and DMA transfer mode to transmit and receive samples can be supported. IIS-specific clock can be supplied from internal system clock controller through IIS clock divider or direct clock source.

IIS V5.1 can handle up to 2 sound sources. For example, OS (Operating Sound)-controlled sound can be delivered to primary sound path and OS-independent sound can be delivered to secondary sound path. IIS V5.1 can mix primary sound source and secondary sound source.

2.2 KEY FEATURES OF IIS MULTI AUDIO INTERFACE

- Mixes up to two sound sources: Primary and Secondary sound source.
- Primary sound source can drive up to 5.1ch IIS-bus for audio interface with external DMA-based operation
- Secondary sound source can support stereo sound channels with internal DMA
- Serial, 8/16/24-bit per channel data transfers
- Supports IIS, MSB-justified and LSB-justified data format
- IIS v5.1 interrupt can wake-up system from IDLE and DEEP_IDLE mode.
- Master /slave mode support
- Auxiliary clock out support for codec chip

2.3 BLOCK DIAGRAM OF IIS MULTI AUDIO INTERFACE

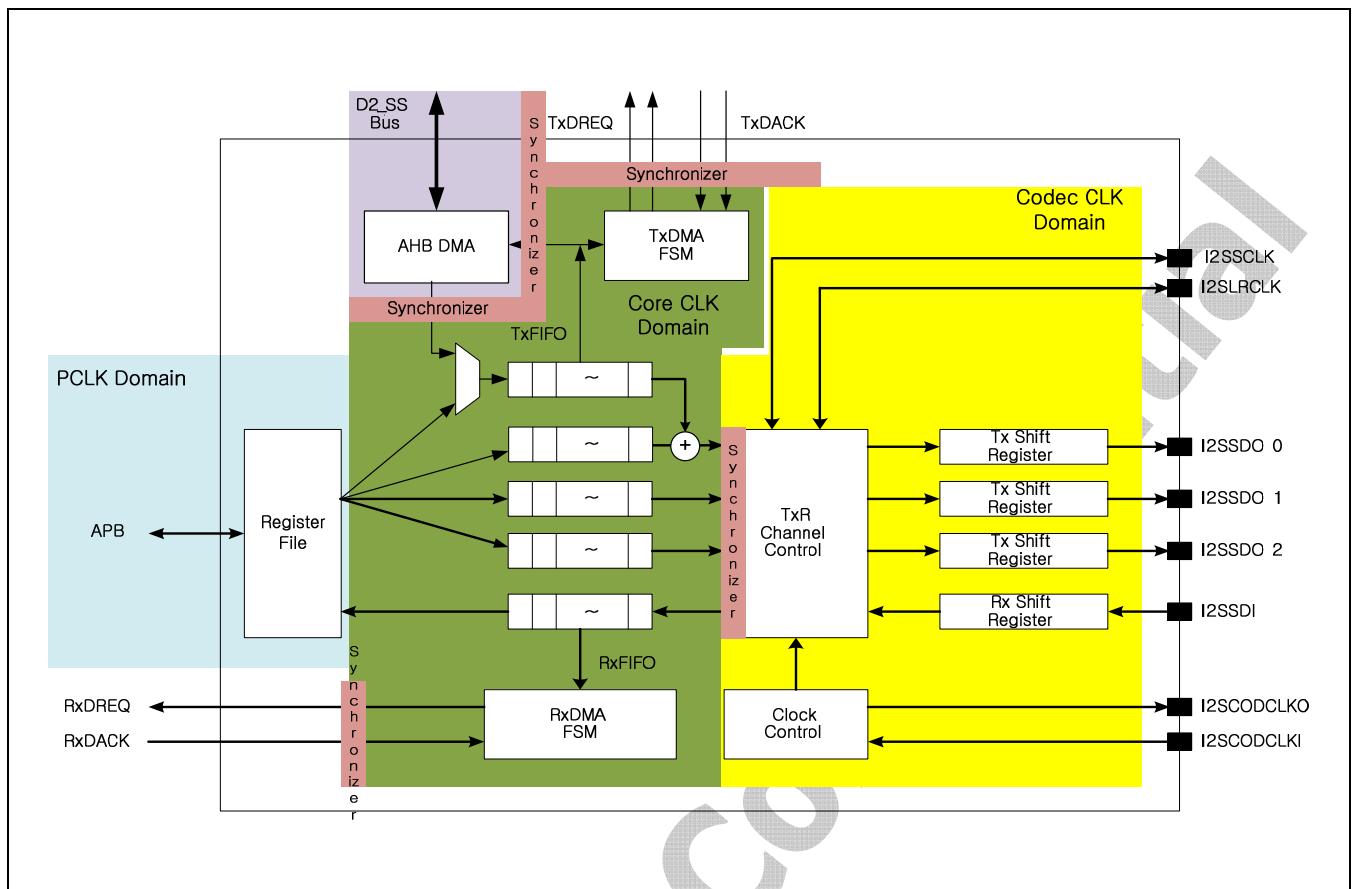


Figure 2-1 IIS-Bus Block Diagram

2.4 FUNCTIONAL DESCRIPTIONS

IIS interface consists of register bank, FIFOs, shift registers, clock control, DMA finite state machine, and channel control block as shown in [Figure 2-1](#). Note that each FIFO has 32-bit width and 64-depth structure, which contains left/right channel data. Thus, FIFO access and data transfer are handled with left/right pair unit. Figure 2-1 shows the functional block diagram of IIS interface.

2.4.1 MASTER/SLAVE MODE

Master/Slave mode shows direction of I2SLRCLK and I2SSCLK. If IIS bus interface transmits I2SLRCLK and I2SSCLK to IIS codec, IIS bus is master mode. If IIS bus interface receives I2SLRCLK and I2SSCLK from IIS codec, IIS bus is slave mode. To select master or slave mode, set MSS bit of IISMOD register.

TX/RX mode indicates the direction of data flow. If IIS bus interface transmits data to IIS codec, this indicates TX mode. Conversely, IIS bus interface receives data from IIS codec, this indicates RX mode.

[Figure 2-2](#) shows AUDIO BUS CLK. For more information, refer to 10.01.S5PV210_Low Power Audio Subsystem.

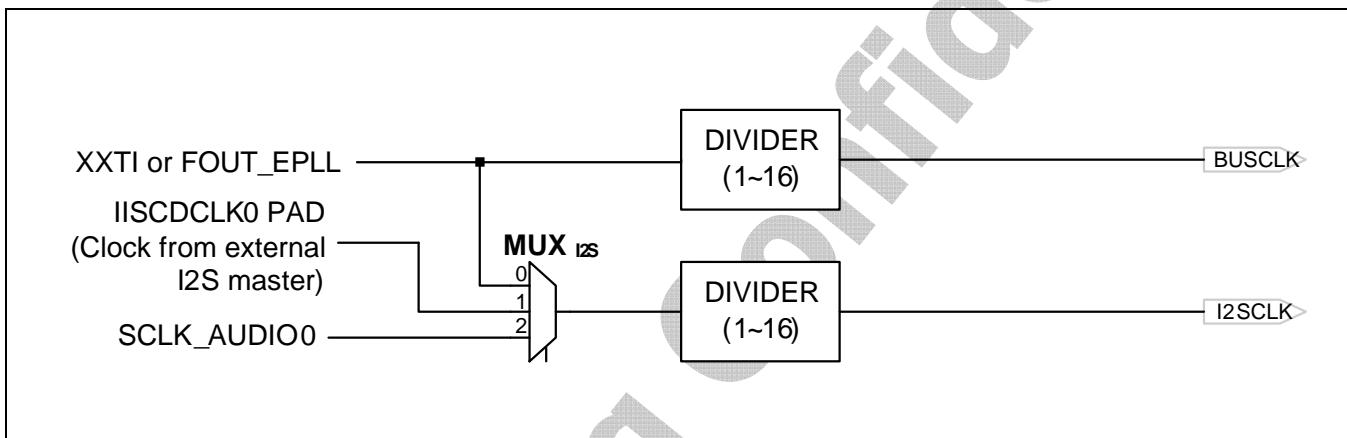


Figure 2-2 Clock Controller in Audio Sub-System

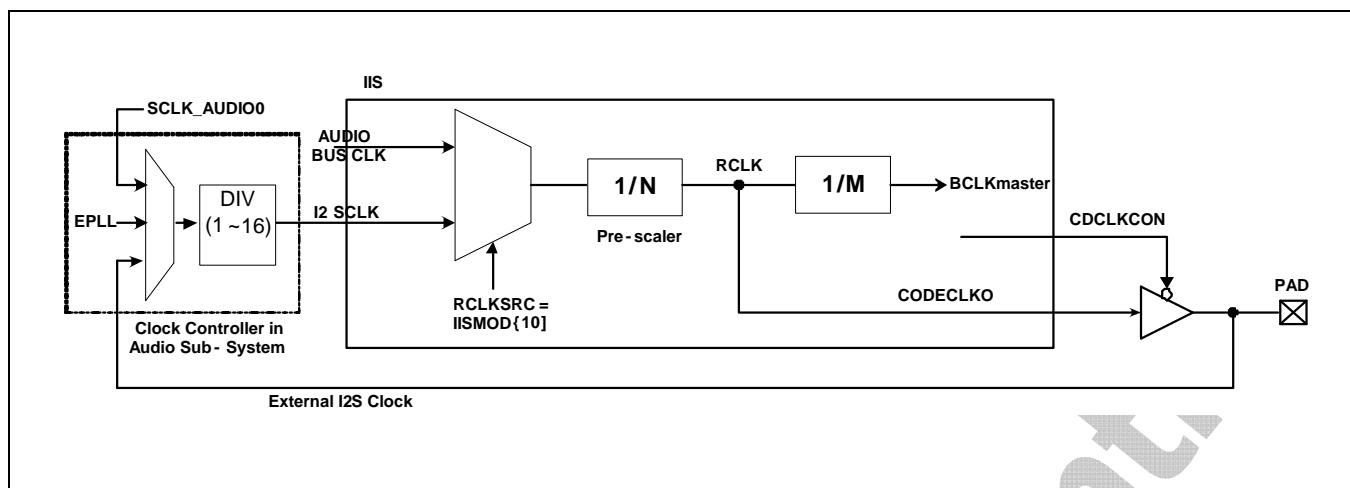


Figure 2-3 IIS Clock Control Block Diagram

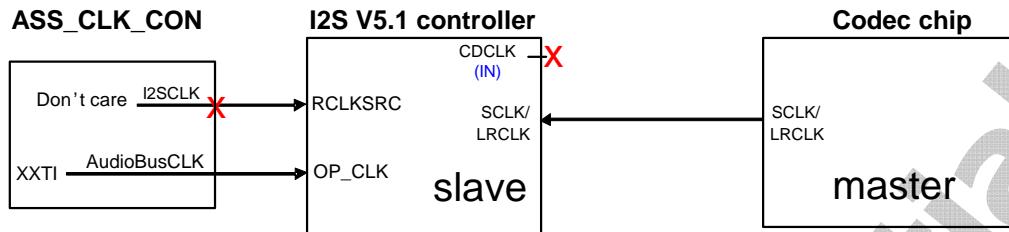
[Figure 2-3](#) shows the route of the root clock with setting in IIS clock control block and system controller. RCLK indicates root clock and RCLKSRC chooses a clock source of RCLK between AUDIO BUS CLK and I2SCLK. The IIS pre-scaler (clock divider) is employed to generate a root clock with divided frequency from source clock.

In master mode, the root clock is divided to generate I2SSCLK and I2SLRCLK. In slave mode, this clock is not used to generate I2SSCLK and I2SLRCLK.

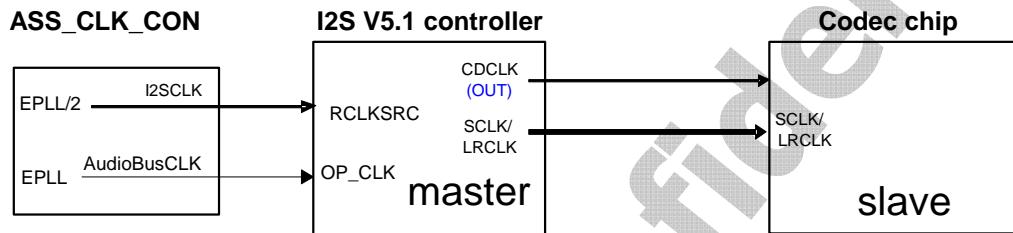
CDCLKCON controls direction of CDCLK GPIO pad. The direction is set by CDCLKCON SFR bit (IISMOD[12]). When CDCLKCON SRF bit is 0, auxiliary clock out is supported for Codec chip at both cases of Master/Slave mode. In this case, RCLK can be supplied to external IIS CODEC chip. When CDCLKCON SRF bit is 1, External I2S clock is supplied from external device. This is useful when internal clock sources are not adequate for generating exact I2SSCLK and I2SLRCLK.

[Figure 2-4](#) and [Table 2-1](#) shows typical usage example of Master EPLL out, Master External clock and Slave.

1) Slave mode



2) Master mode (EPLL out)



3) Master mode (External Clock in)

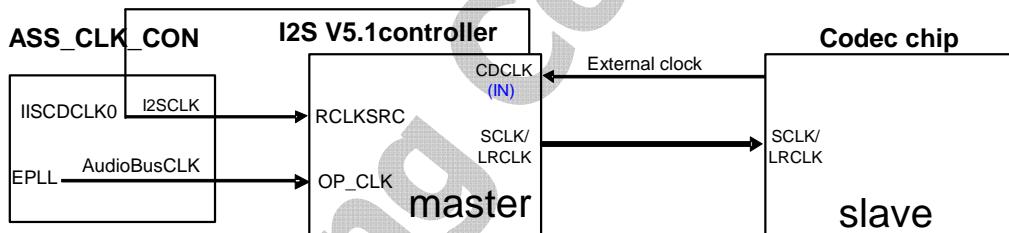


Figure 2-4 Master/Slave Modes of IIS

Table 2-1 Typical Usage of Master/Slave Modes

Mode	AudioSS CLK_CON		IIS v5.1 IISMOD			
	AudioBusClk	I2SCLK	MSS	RCLKSRC	OP_CLK	CDCLKCON
Slave mode	XXTI or EPLL	Gating	1 (Slave)	1 (I2SCLK)	3 (AudioBusClk)	1 (In)
Master mode (EPLL out)	EPLL	EPLL/2	0 (Master)	1 (I2SCLK)	3 (AudioBusClk)	0 (Out)
Master mode (External clock in)	EPLL	IISCDCLK0	0 (Master)	1 (I2SCLK)	3 (AudioBusClk)	1 (In)

2.4.1.1 External DMA Transfer

To transfer up to 5.1 channel primary sound from s/w mixer to IIS, use external DMA or SFR interface. To play primary sound for 5.1 channels or record 2 channel sound, IIS has TXFIFO0, TXFIFO1, TXFIFO2, TXFIFO_S and RXFIFO registers. IIS will mix primary sound in TXFIFO0 and secondary sound in TXFIFO_S and output mixed sound stream to external codec logic.

In the external DMA transfer mode, use external DMA controller to access the transmitter or receiver FIFO. The transmitter or receiver FIFO state activates DMA service request internally. The FTXEMPT, FRXEMPT, FTXFULL, and FRXFULL bits of I2SCON register represent the transmitter or receiver FIFO data state. Especially, FTXEMPT and FRXFULL bit are the ready flag for DMA service request; the transmit DMA service request is activated when TXFIFO is not empty and the receiver DMA service request is activated when RXFIFO is not full.

The external DMA transfer uses only handshaking method for single data. Note that during external DMA acknowledge activation; the data read or write operation should be performed.

* Reference: DMA request point

- TX mode: (FIFO is not full) & (TXDMAACTIVE is active)
- RX mode: (FIFO is not empty) & (RXDMAACTIVE is active)

2.4.1.2 Internal DMA Transfer

To transfer up to 2 channel secondary sound to IIS, use internal DMA or SFR interface. To play secondary sound for 2 channels, internal DMA in IIS gets sound data from address range between 0xC000_0000 and 0xC01F_FFFF (when ARM decodes encoded music file.) or between OBUF0 and OBUF1 (when RP decodes encoded music file.) to TXFIFO_S. IIS will mix primary sound in TXFIFO0 and secondary sound in TXFIFO_S and output mixed sound stream to external codec logic.

Like external DMA transfer mode, in the internal DMA transfer mode, the internal DMA is activated when TXFIFO_S is not full. After activation, internal DMA runs according to SFR configurations and signals an interrupt after completion.

- It only supports single transfer in both Internal & External DMA transfer mode.
- Refer OBUF0 and OBUF1 at 10.01.S5PV210_Low Power Audio Subsystem chapter.

2.4.1.3 Sound Mixing

IIS can mix primary sound in TXFIFO0 and secondary sound in TXFIFO_S when two sound sources have the same sampling rate and PCM format.

- If overflow occurs, then mixer saturates output value.
- Mixer can handle Different Bit Length. (Controlled by BLC bit at IISMOD SFR)

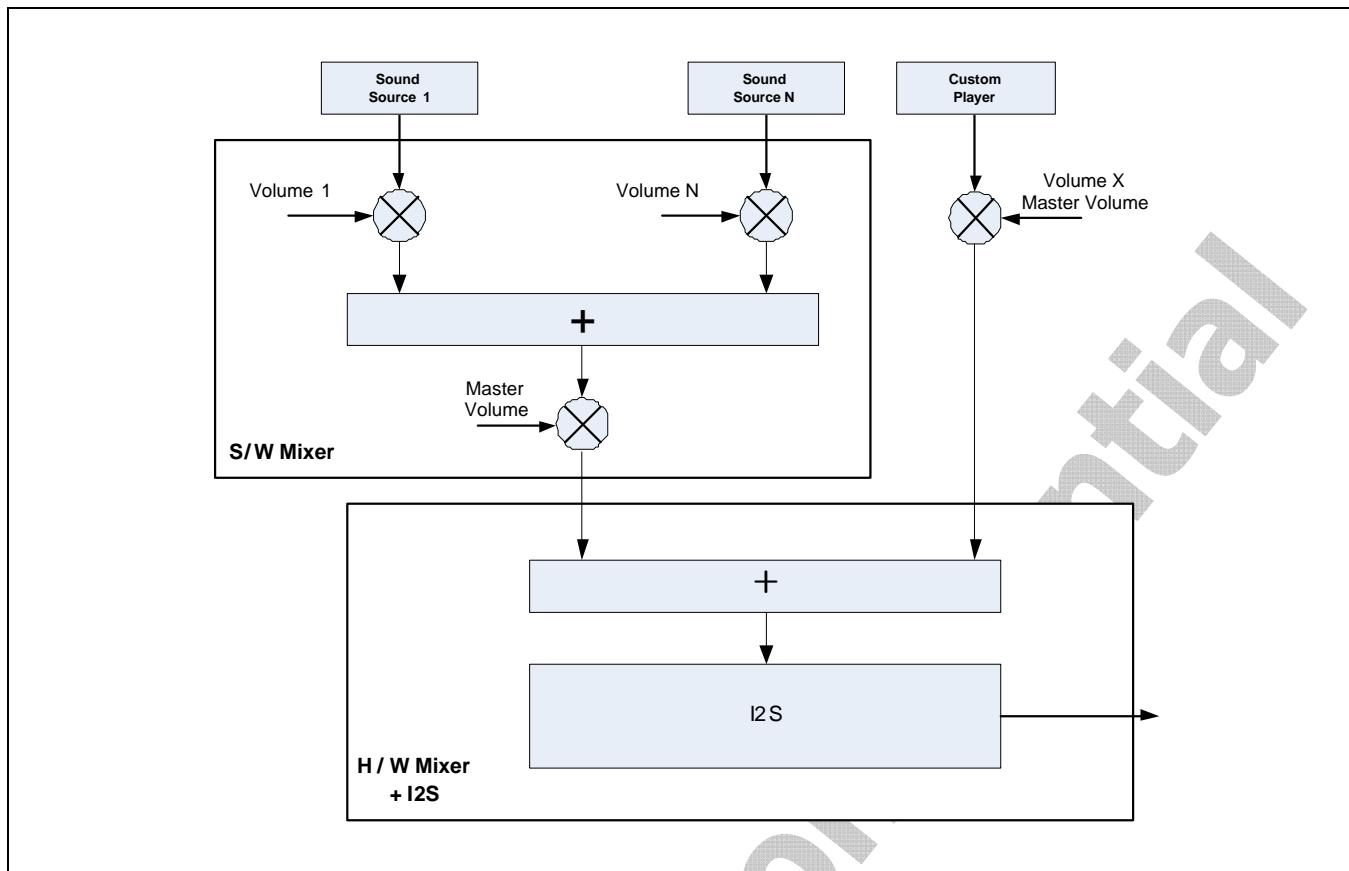


Figure 2-5 Concept of Mixer in IIS

This function has two limitations:

1. Normalization should be pre-processed in S/W configurations or settings.
2. Synchronization between two sound sources is not guaranteed.

2.5 AUDIO SERIAL DATA FORMAT

2.5.1 IIS-BUS FORMAT

The IIS bus has four lines including serial data input I2SSDI, serial data output I2SSDO, left/right channel select clock I2SLRCLK, and serial bit clock I2SSCLK; master generates I2SLRCLK and I2SSCLK.

Serial data is transmitted in 2's complement with the MSB first with a fixed position, whereas the position of the LSB depends on the word length. The transmitter sends the MSB of the next word at one clock period after the I2SLRCLK is changed. Serial data sent by the transmitter can be synchronized either with the trailing or with the leading edge of the clock signal.

The LR channel select line indicates the direction of left or right channel being transmitted. I2SLRCLK may be changed either on a trailing or leading edge of the serial clock, but it does not need to be symmetrical. In the slave, this signal is latched on the leading edge of the clock signal. The I2SLRCLK line changes one clock period before the MSB is transmitted. This allows the slave transmitter to derive synchronous timing of the serial data that will be set up for transmission. Furthermore, it enables the receiver to store the previous word and clear the input for the next word.

2.5.2 MSB (LEFT) JUSTIFIED

MSB-Justified (Left-Justified) format is similar to IIS bus format, except that in MSB-justified format, the transmitter always sends the MSB of the next word at the same time whenever the I2SLRCLK is changed.

2.5.3 LSB (RIGHT) JUSTIFIED

LSB-Justified (Right-Justified) format is opposite to the MSB-justified format. In other word, the transferring serial data is aligned with ending point of I2SLRCLK transition.

Figure 2-6 shows the audio serial format of IIS, MSB-justified, and LSB-justified. Note that in this figure, the word length is 16-bit and I2SLRCLK makes transition every 24 cycle of I2SSCLK (BFS is 48 fs, where fs is sampling frequency; I2SLRCLK frequency).

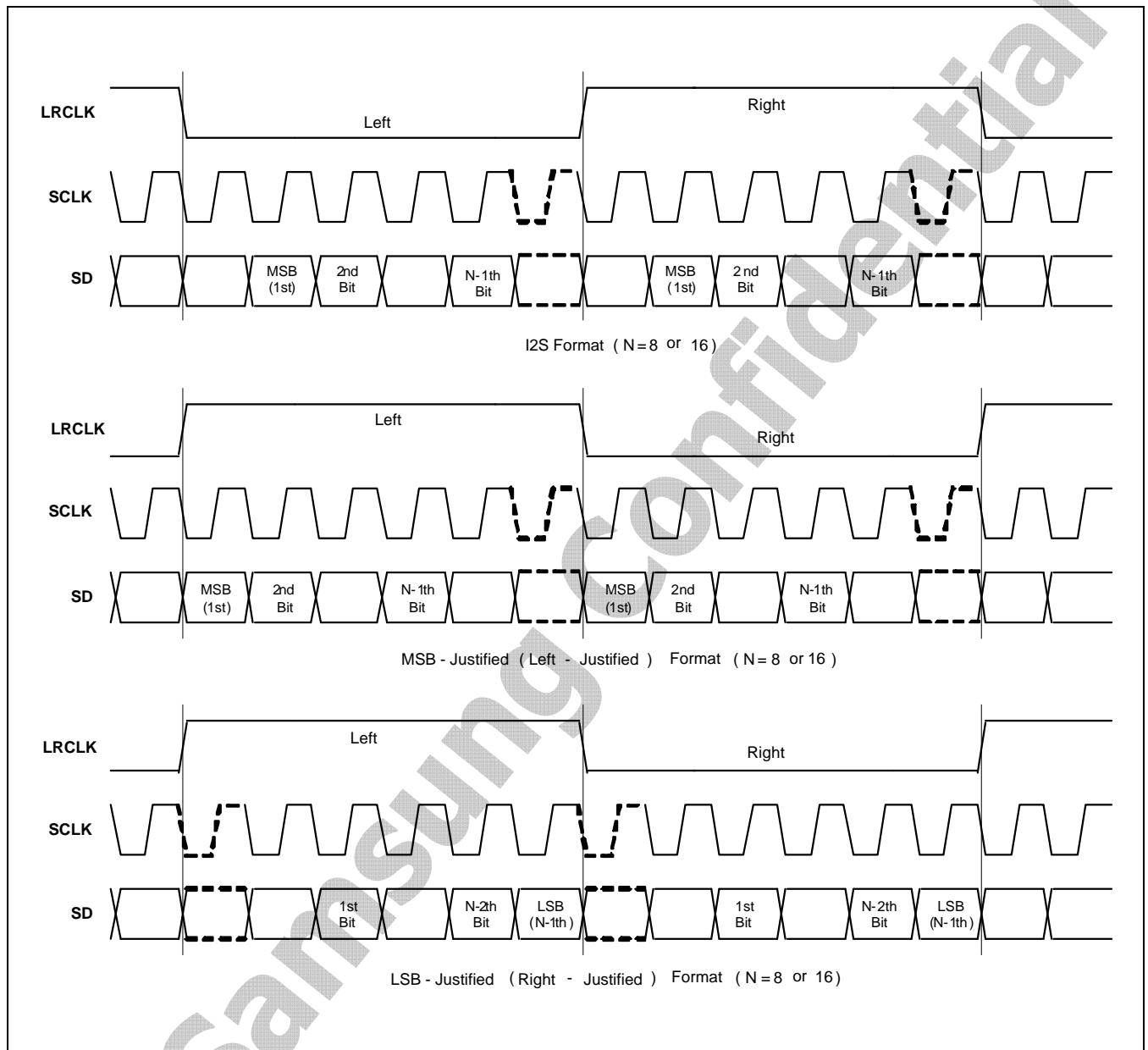


Figure 2-6 IIS Audio Serial Data Formats

2.6 PCM BIT LENGTH (BLC), RFS DIVIDER AND BFS DIVIDER FOR SAMPLING FREQUENCY (IISLRCLK), SERIAL BITCLK(IISSCLK), AND ROOT CLOCK(RCLK)

When IIS interface Controller operates as master, IIS interface Controller generates IISLRCLK and IISSCLK that is Root Clock is divided using RFS and BFS value. To decide Sampling Frequency – IISLRCLK -, BLC, BFS, and RFS are selected first. Optionally, IIS interface Controller clocks out Root clock as IIISCDCLK for codec master clock (if source of root clock is not IISEXTCDCLK).

In slave mode, you must set the value of BLC, BFS and RFS similar to master (ex: Codec). Because IIS interface controller needs these value for correct operation.

2.6.1 PCM WORD LENGTH AND BFS DIVIDER

PCM Word Length (BLC) value is selected first, because the value affects BFS value. [Table 2-2](#) shows BFS available value as BLC.

Table 2-2 Allowed BFS Value as BLC

PCM Bit length(BLC)	8-bit	16-bit	24-bit
Available BFS value	16fs, 24fs, 32fs, 48fs	32fs, 48fs	48fs

2.6.2 BFS DIVIDER AND RFS DIVIDER

RFS value is selected when BFS is selected [Table 2-3](#) shows RFS available value as BFS.

Table 2-3 Allowed RFS Value as BFS

BFS Divider	16fs, 32fs	24fs, 48fs
Available RFS value	256fs, 384fs, 512fs, 768fs.	384fs, 768fs.

2.6.3 RFS DIVIDER AND ROOT CLOCK

Table 2-4 shows relationship between ROOT CLOCK, IISLRCLK and RFS. RCLK is clock divided by IIS pre-scaler (IISPSR) that is selected by IMS

Table 2-4 Root Clock Table (MHz)

IISLRCK RFS \	8.000 kHz	11.025 kHz	16.000 kHz	22.050 kHz	32.000 kHz	44.100 kHz	48.000 kHz	64.000 kHz	88.200 kHz	96.000 kHz
256fs	2.0480	2.8224	4.0960	5.6448	8.1920	11.2896	12.2880	16.3840	22.5792	24.5760
384fs	3.0720	4.2336	6.1440	8.4672	12.2880	16.9344	18.4320	24.5760	33.8688	36.8640
512fs	4.0960	5.6448	8.1920	11.2896	16.3840	22.5792	24.5760	32.7680	45.1584	49.1520
768fs	6.1440	8.4672	12.2880	16.9344	24.5760	33.8688	36.8640	49.1520	67.7376	73.7280

Root Clock Frequency = fs * (256, 384, 512 or 768)

2.7 PROGRAMMING GUIDE

The IIS bus interface can be accessed either by the processor using programmed I/O instructions or by the DMA controller.

2.7.1 INITIALIZATION

1. Before you use IIS bus interface, you must configure GPIOs to IIS mode. Check signal's direction. I2SLRCLK, I2SSCLK and I2SCDCLK is inout-type. The I2SSDI and I2SSDO is input and output respectively.
2. Select clock source. S5PV210 has three clock sources, namely, Audio bus clock, EPLL and external codec. For more information, refer [Figure 2-2](#) and [Figure 2-3](#).

2.7.2 PLAY MODE (TX MODE) WITH DMA

1. TXFIFO is flushed before operation. If you do not distinguish Master/Slave mode from TX/RX mode, you must study Master/Slave mode and TX/RX mode. Refer Master/Slave chapter.
2. Configure I2SMOD register and I2SPSR (IIS pre-scaler register).
3. To operate system in stability, the internal TXFIFO should be almost full before transmission. To satisfy this, start TXDMA before asserting I2SACTIVE.
4. Basically, IIS bus does not support the interrupt. Therefore, you can only check state by polling through accessing SFR.
5. If TXFIFO is full, you can assert I2SACTIVE.

2.7.3 RECORDING MODE (RX MODE) WITH DMA

1. RXFIFO is flushed before operation. Also, if you don't distinguish between Master/Slave mode and TX/RX mode, you must study Master/Slave mode and TX/RX mode. Refer Master/Slave chapter.
2. Configure I2SMOD register and I2SPSR (IIS pre-scaler register).
3. To operate system in stability, the internal RXFIFO should have at least one data before DMA operation. To satisfy this, assert I2SACTIVE before starting RXDMA.
4. Check RXFIFO state by polling through accessing SFR.
5. If RXFIFO is not empty, start RXDMACTIVE.

2.7.4 EXAMPLE CODE

2.7.4.1 TX Channel

The IIS TX channel provides a single stereo compliant output. The transmit channel can operate in master or Slave mode. Data is transferred between the processor and the IIS controller via an APB access or a DMA access.

The processor must write words in multiples of two (i.e. for left and right audio sample). The words are serially shifted out timed with respect to the audio bitclk, SCLK and word select clock, LRCLK.

TX Channel has 64X32 bits wide FIFO where the processor or DMA can write upto 16 left/right data samples after enabling the channel for transmission.

An Example sequence is as follows:

Ensure the Audio bus clock and CDCLK are coming correctly to the IIS controller and FLUSH the TX FIFO using the TFLUSH bit in the I2SFIC Register (IIS FIFO Control Register).

Please ensure that IIS Controller is configured in one of the following modes.

- TX only mode
- TX/RX simultaneous mode

This can be done by programming the TXR bit in the I2SMOD Register (IIS Mode Register).

1. Then Program the following parameters according to the need

- MSS, RCLKSRC
- SDF
- BFS
- BLC
- LRP

For Programming, the above-mentioned fields please refer I2SMOD Register (IIS Mode Register).

2. Once ensured that the input clocks for IIS controller are up and running and step 1 and 2 have been completed we can write to TX FIFO.

The write to the TX FIFO has to be carried out thorough the I2STXD Register (IIS TX FIFO Register) This 32-bit data will occupy position 0 of the FIFO and any further data will be written to position 2, 3 and so on.

The Data is aligned in the TX FIFO for 8-bit/channel or 16-bit/channel BLC as shown in the [Figure 2-7](#).

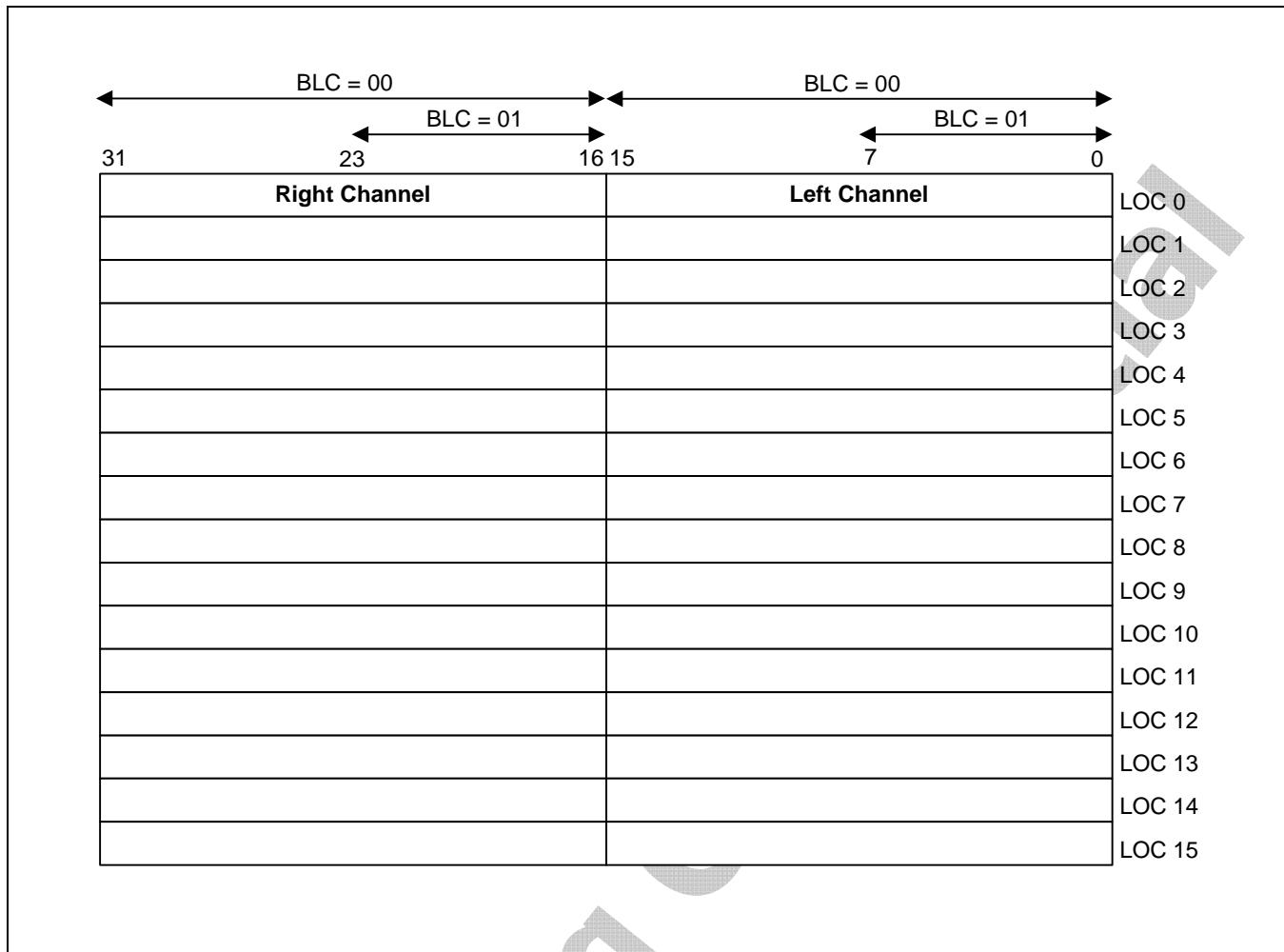


Figure 2-7 TX FIFO Structure for BLC = 00 or BLC = 01

The Data is aligned in the TX FIFO for 24-bit/channel BLC as shown in [Figure 2-8](#).

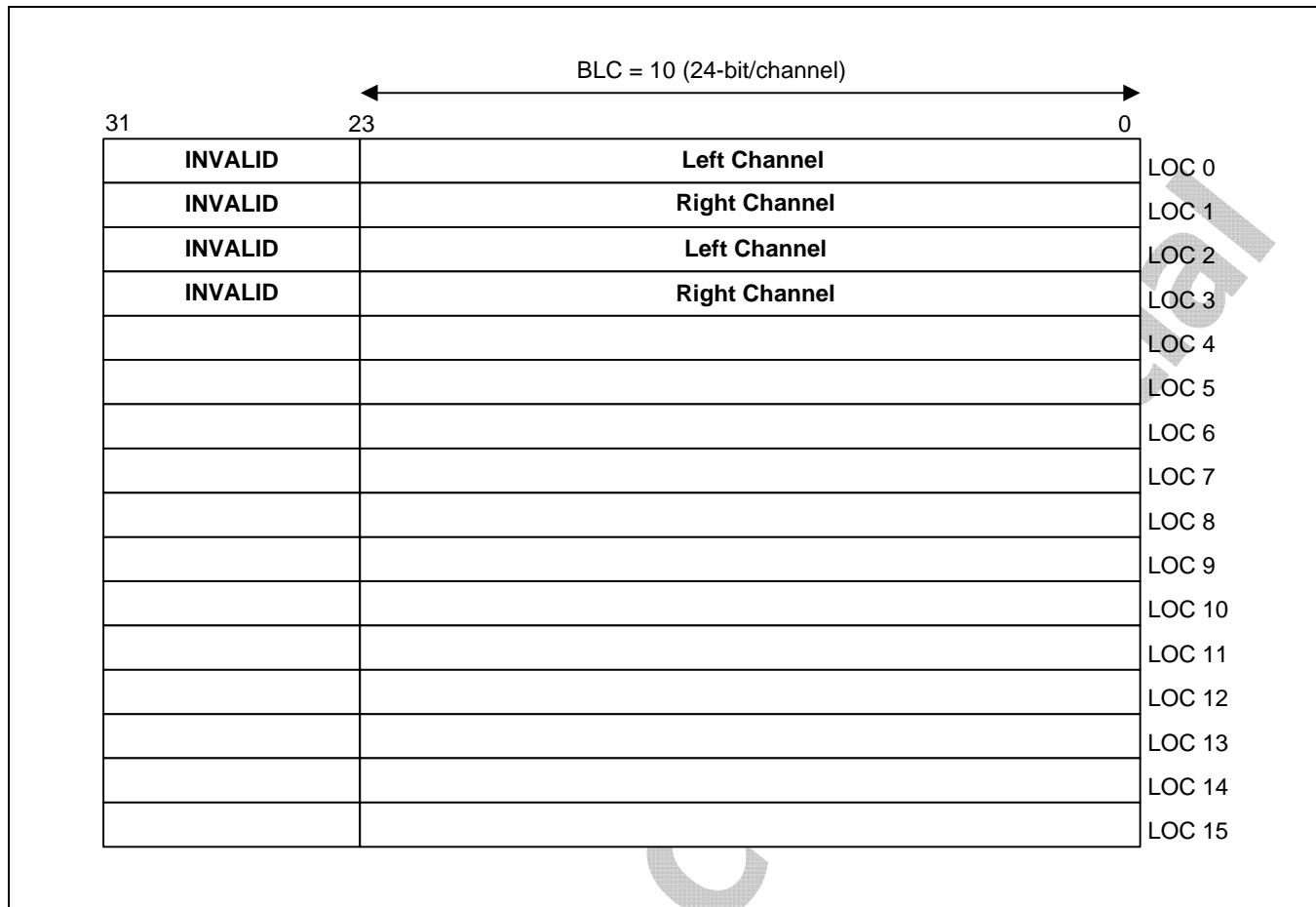


Figure 2-8 TX FIFO Structure for BLC = 10 (24-bit/channel)

Once the data is written to the TX FIFO the TX channel can be made active by enabling the I2SACTIVE bit in the I2SCON Register (IIS Control Register).

The data is then serially shifted out with respect to the bit clock SCLK and word select clock LRCLK.

The TXCHPAUSE in the I2SCON Register (IIS Control Register) can stop the serial data transmission on the I2SSDO. The transmission is stopped once the current Left/Right channel is transmitted.

If the control registers in the I2SCON Register (IIS Control Register) and I2SMOD Register (IIS Mode Register) are to be reprogrammed then it is advisable to disable the TX channel.

If the TX channel is enabled while the FIFO is empty, no samples are read from the FIFO.

The Status of TX FIFO can be checked by checking the bits in the I2SFIC Register (IIS FIFO Control Register).

2.7.4.2 RX Channel

The IIS RX channel provides a single stereo compliant output. The receive channel can operate in master or slave mode. Data is received from the input line and transferred into the RX FIFO. The processor can then read this data via an APB read or a DMA access can access this data.

RX Channel has 64 X 32-bit wide RX FIFO where the processor or DMA can read upto 16 left/right data samples after enabling the channel for reception.

An Example sequence is as follows:

Ensure the Audio bus clock and CDCLK are coming correctly to the IIS controller and FLUSH the RX FIFO using the RFLUSH bit in the I2SFIC Register (IIS FIFO Control Register) and the I2S controller is configured in any of the modes

- Receive only.
- Receive/Transmit simultaneous mode

This can be done by Programming the TXR bit in the I2SMOD Register (IIS Mode Register)

1. Then Program the following parameters according to the need

- MSS, RCLKSRC
- SDF
- BFS
- BLC
- LRP

For Programming, the above mentioned fields please refer I2SMOD Register (IIS Mode Register)

2. Once ensured that the input clocks for IIS controller are up and running and step 1 and 2 have been completed user must put the I2SACTIVE high to enable any reception of data, the IIS Controller receives data on the LRCLK change.

The Data must be read from the RX FIFO using the I2SRXD Register (IIS RX FIFO Register) only after looking at the RX FIFO count in the I2SFIC Register (IIS FIFO Control Register). The count would only increment once the complete left channel and right have been received. The Data is aligned in the RX FIFO for 8-bits/channel or 16-bits/channel BLC as shown in [Figure 2-9](#).

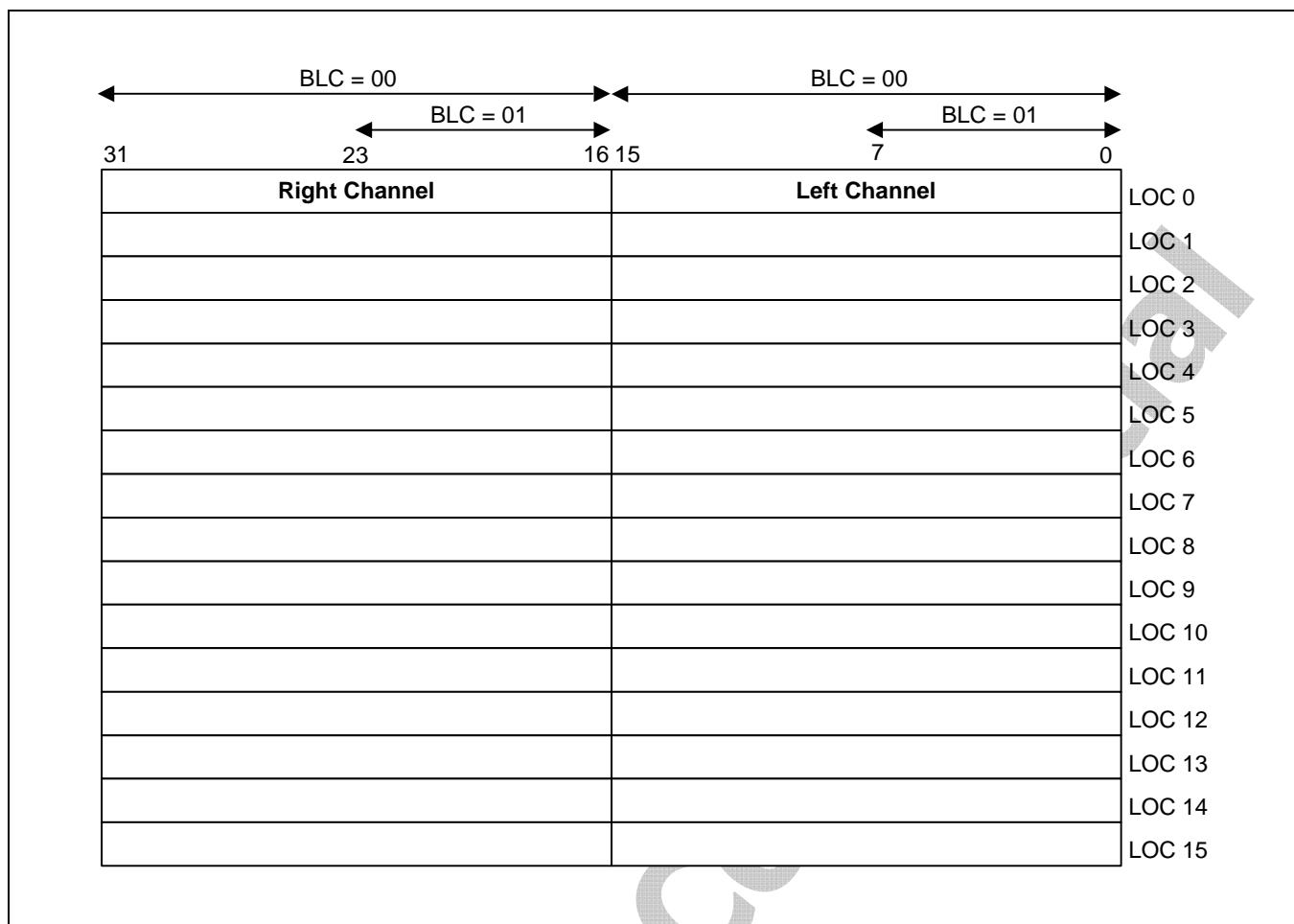


Figure 2-9 RX FIFO Structure for BLC = 00 or BLC = 01

The Data is aligned in the RX FIFO for 24-bit/channel BLC as shown in [Figure 2-10](#).

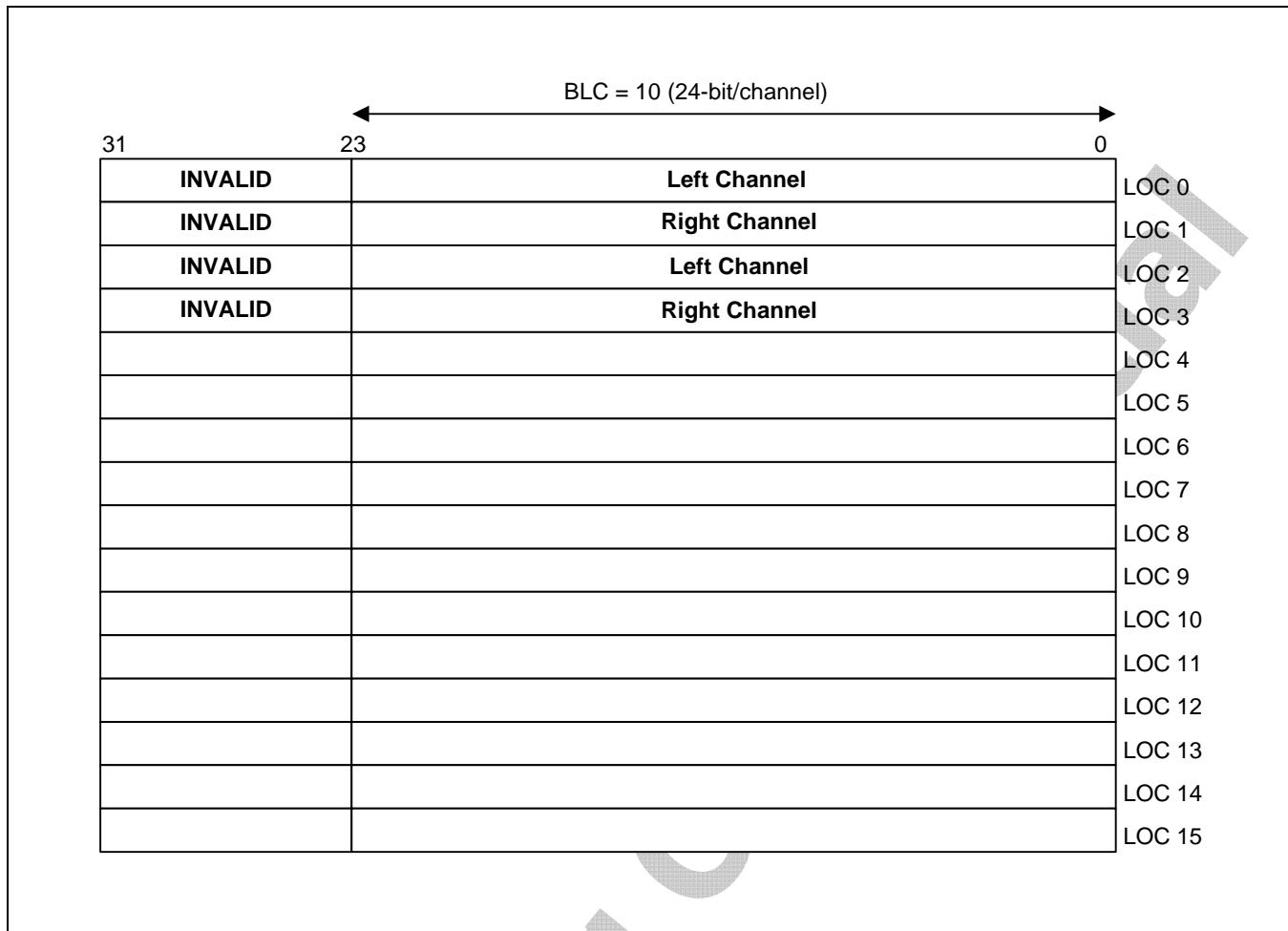


Figure 2-10 RX FIFO Structure for BLC = 10 (24-bit/channel)

The RXCHPAUSE in the I2SCON register can stop the serial data reception on the I2SSDI. The reception is stopped once the current Left/Right channel is received.

If the control registers in the I2SCON Register (IIS Control Register) and I2SMOD Register (IIS Mode Register) are to be reprogrammed then it is advisable to disable the RX channel.

The Status of RX FIFO can be checked by checking the bits in the I2SFIC Register (IIS FIFO Control Register).

2.8 IO DESCRIPTION

Signal	I/O	Description	Pad	Type
Xi2s0SCLK	I/O	IIS Serial clock (Bit clock)	Xi2s0SCLK	Dedicated
Xi2s0LRCLK	I/O	IIS LR channel clock	Xi2s0LRCLK	Dedicated
Xi2s0CDCLK	I/O	Auxiliary clock out for codec chip, IIS external clock input	Xi2s0CDCLK	Dedicated
Xi2s0SDI	I	IIS serial data input	Xi2s0SDI	Dedicated
Xi2s0SDO0	O	IIS serial data out 0	Xi2s0SDO0	Dedicated
Xi2s0SDO1	O	IIS serial data out 1	Xi2s0SDO1	Dedicated
Xi2s0SDO2	O	IIS serial data out 2	Xi2s0SDO2	Dedicated

2.9 REGISTER DESCRIPTION

2.9.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
IISCON	0xEEE3_0000	R/W	Specifies the IIS interface control register	0x000
IISMOD	0xEEE3_0004	R/W	Specifies the IIS interface mode register	0x0
IISFIC	0xEEE3_0008	R/W	Specifies the IIS interface primary Tx FIFO & Rx FIFO control register	0x0
IISPSR	0xEEE3_000C	R/W	Specifies the IIS interface clock divider control register	0x0
IISTXD	0xEEE3_0010	W	Specifies the IIS interface transmit primary sound data register	0x0
IISRXD	0xEEE3_0014	R	Specifies the IIS interface receive data register	0x0
IISFICS	0xEEE3_0018	R/W	Specifies the IIS interface secondary TXFIFO_S control register	0x0
IISTXDS	0xEEE3_001C	W	Specifies the IIS interface secondary transmit data register	0x0
IISAHB	0xEEE3_0020	R/W	Specifies the IIS AHB DMA control register	0x0
IISSTR0	0xEEE3_0024	R/W	Specifies the IIS AHB DMA start address0 register	0x0
IISSIZE	0xEEE3_0028	R/W	Specifies the IIS AHB DMA size register	0x7FFF_0000
IISTRNCNT	0xEEE3_002C	R	Specifies the IIS AHB DMA transfer count register	0x0
IISLVL0ADDR	0xEEE3_0030	R/W	Specifies the IIS AHA DMA Interrupt level 0 register	0x0000_0000
IISLVL1ADDR	0xEEE3_0034	R/W	Specifies the IIS AHA DMA Interrupt level 1 register	0x0000_0000
IISLVL2ADDR	0xEEE3_0038	R/W	Specifies the IIS AHA DMA Interrupt level 2 register	0x0000_0000
IISLVL3ADDR	0xEEE3_003C	R/W	Specifies the IIS AHA DMA Interrupt level 3 register	0x0000_0000
IISSTR1	0xEEE3_0040	R/W	Specifies the IIS AHB DMA start address1 register	0x0

NOTE: All registers of IIS interface are accessible by word unit with STR/LDR instructions.

2.9.1.1 IIS Interface Control Register (IISCON, R/W, Address = 0xEEE3_0000)

IISCON	Bit	Description	R/W	Initial State
SW_RST	[31]	IIS s/w reset control. This should be set to 1 after IIS clock is stable. 0 = Reset IIS module (default) 1 = Un-reset IIS module Before reading SFR of IIS, user must set this bit.	R/W	0
Reserved	[30:27]	-	R	0x0
FRXOFSTATUS	[26]	RX FIFO Over Flow Interrupt Status. And this is used by interrupt clear bit. When this is high, you can do interrupt clear by writing '1'. 0 = Interrupt does not occur. 1 = Interrupt occurs	R/W	0
FRXOFINTEN	[25]	Enables RX FIFO Overflow Interrupt 0 = Disables RXFIFO Overflow INT 1 = Enables RXFIFO Overflow INT	R/W	0
FTXSUR STATUS	[24]	Secondary TX FIFO_S under-run interrupt status. This is used by interrupt clear bit. When this is high, you can clear interrupt clear by writing '1'. 0 = Interrupt does not occur. 1 = Interrupt occurs.	R/W	0
FTXSURINTEN	[23]	Secondary TX FIFO_S Under-run Interrupt Enable 0 = TXFIFO_S Under-run INT disable 1 = TXFIFO_S Under-run INT enable	R/W	0
FTXSEMPY	[22]	Secondary TX FIFO_S empty Status Indication 0 = TX FIFO_S is not empty(Ready to transmit Data) 1 = TX FIFO_S is empty (Not Ready to transmit Data)	R	0
FTXSFULL	[21]	Secondary TX FIFO_S full Status Indication 0 = TX FIFO_S is not full 1 = TX FIFO_S is full	R	0
TXSDMAPAUSE	[20]	Tx External DMA operation for secondary TX FIFO_S pause command. Note that when this bit is activated, the External DMA request will be halted after current on-going External DMA transfer is completed. 0 = No pause External DMA operation for TX FIFO_S 1 = Pause External DMA operation for TX FIFO_S Note: IISDMAEN SFR performs Internal DMA stop control.	R/W	0
Reserved	[19]	Reserved. This value must be 0.	R/W	0
TXSDMACTIVE	[18]	Tx External DMA active for secondary TX FIFO_S (start External DMA request). Note that when this bit is set from high to low, the External DMA operation will be forced to stop immediately. 0 = Inactive 1 = Active	R/W	0

IISCON	Bit	Description	R/W	Initial State
FTXURSTATUS	[17]	Primary TX FIFOx under-run interrupt status. This is used by interrupt clear bit. When this is high, you can clear interrupt by writing '1'. 0 = Interrupt didn't be occurred. 1 = Interrupt was occurred.	R/W	0
FTXURINTEN	[16]	Primary TX FIFOx Under-run Interrupt Enable 0 = TXFIFO Under-run INT disable 1 = TXFIFO Under-run INT enable	R/W	0
FTX2EMPT	[15]	Primary TX FIFO2 empty Status Indication 0 = TX FIFO2 is not empty(Ready to transmit Data) 1 = TX FIFO2 is empty (Not Ready to transmit Data)	R	0
FTX1EMPT	[14]	Primary TX FIFO1 empty Status Indication 0 = TX FIFO1 is not empty (Ready to transmit Data) 1 = TX FIFO1 is empty (Not Ready to transmit Data)	R	0
FTX2FULL	[13]	Primary TX FIFO2 full Status Indication 0 = TX FIFO2 is not full 1 = TX FIFO2 is full	R	0
FTX1FULL	[12]	Primary TX FIFO1 full Status Indication 0 = TX FIFO1 is not full 1 = TX FIFO1 is full	R	0
LRI	[11]	Left/Right channel clock indication. Note that LRI meaning is dependent on the value of LRP bit of I2SMOD register. 0 = Left (when LRP bit is low) or right (when LRP bit is high) 1 = Right (when LRP bit is low) or left (when LRP bit is high)	R	0
FTX0EMPT	[10]	Primary Tx FIFO0 empty status indication. 0 = FIFO is not empty (ready for transmit data to channel) 1 = FIFO is empty (not ready for transmit data to channel)	R	0
FRXEMPT	[9]	Rx FIFO empty status indication. 0 = FIFO is not empty 1 = FIFO is empty	R	0
FTX0FULL	[8]	Primary Tx FIFO0 full status indication. 0 = FIFO is not full 1 = FIFO is full	R	0
FRXFULL	[7]	Rx FIFO full status indication. 0 = FIFO is not full (ready for receive data from channel) 1 = FIFO is full (not ready for receive data from channel)	R	0
TXDMAPAUSE	[6]	Tx DMA operation pause command for primary TX FIFOx. Note that when this bit is activated, the DMA request will be halted after current on-going DMA transfer is completed. 0 = No pause DMA operation for TX FIFOx 1 = Pause DMA operation for TX FIFOx	R/W	0

IISCON	Bit	Description	R/W	Initial State
RXDMAPAUSE	[5]	Rx DMA operation pause command. Note that when this bit is activated, the DMA request will be halted after current on-going DMA transfer is completed. 0 = No pause DMA operation 1 = Pause DMA operation	R/W	0
TXCHPAUSE	[4]	Tx channel operation pause command for primary TX FIFOx. Note that when this bit is activated, the channel operation will be halted after left-right channel data transfer is completed. 0 = No pause operation for TX FIFOx and TX_S FIFO 1 = Pause operation for TX FIFOx and TX_S FIFO	R/W	0
RXCHPAUSE	[3]	Rx channel operation pause command. Note that when this bit is activated, the channel operation will be halted after left-right channel data transfer is completed. 0 = No pause operation 1 = Pause operation	R/W	0
TXDMAACTIVE	[2]	Tx DMA active for primary TX FIFOx (start DMA request). Note that when this bit is set from high to low, the DMA operation will be forced to stop immediately. 0 = Inactive 1 = Active	R/W	0
RXDMAACTIVE	[1]	Rx DMA active (start DMA request). Note that when this bit is set from high to low, the DMA operation will be forced to stop immediately. 0 = Inactive 1 = Active	R/W	0
I2SACTIVE	[0]	IIS interface active (start operation). 0 = Inactive 1 = Active	R/W	0

2.9.1.2 IIS Interface Mode Register (IISMOD, R/W, Address = 0xEEE3_0004)

IISMOD	Bit	Description	R/W	Initial State
OP_CLK	[31:30]	Operation clock for IIS logic. 00 = Codec clock out 01 = Codec clock in 10 = Bit clock out 11 = Audio bus clock	R/W	00
Reserved	[29]	-	R	0
OP_MUX_SEL	[28]	Mux selection for secondary TX FIFO_S 0 = TX FIFO_S gets data from APB SFR interface 1 = TX FIFO_S gets data from internal DMA interface Before trying to change this field from 1 to 0, s/w must poll IISTRNCNT register to confirm that all the transfer is done according to internal DMA setting. There is no restriction on switching from 0 to 1.	R/W	0
BLC_S	[27:26]	Bit Length Control Bit which decides transmission of 8/16/24 bits per audio channel for Secondary TX FIFO_S 00 = 16 Bits per channel 01 = 8 Bits Per Channel 10 = 24 Bits Per Channel 11 = Reserved	R/W	00
BLC_P	[25:24]	Bit Length Control Bit Which decides transmission of 8/16/24 bits per audio channel for Primary TX FIFOx 00 = 16 Bits per channel 01 = 8 Bits Per Channel 10 = 24 Bits Per Channel 11 = Reserved	R/W	00
Reserved	[23:22]	-	R	00
CDD2	[21:20]	Channel-2 Data Discard. Discard means zero padding. It only supports 8/16 bit mode. 00 = No Discard 01 = I2STXD[15:0] Discard 10 = I2STXD[31:16] Discard 11 = Reserved	R/W	00
CDD1	[19:18]	Channel-1 Data Discard. Discard means zero padding. It only supports 8/16 bit mode. 00 = No Discard 01 = I2STXD[15:0] Discard 10 = I2STXD[31:16] Discard 11 = Reserved	R/W	00
DCE	[17:16]	Enables Data Channel. [17]: Enables SD2 channel [16]: Enables SD1 channel	R/W	00
Reserved	[15]	-	R	0

IISMOD	Bit	Description	R/W	Initial State
BLC	[14:13]	Bit Length Control Bit Which decides transmission of 8/16/24 bits per audio channel for final mixed sound Tx output or Rx input. 00 = 16 Bits per channel 01 = 8 Bits Per Channel 10 = 24 Bits Per Channel 11 = Reserved	R/W	00
CDCLKCON	[12]	Determine direction of codec clock source (I2S_CDCLK) 0 = Supply RCLK to I2S_CDCLK (external codec chip) 1 = Get clock (to CLKAUDIO) from I2S_CDCLK (external codec chip) (Refer to Figure 2-3)	R/W	0
MSS	[11]	Master or slave mode select 0 = Master mode 1 = Slave mode	R/W	0
RCLKSRC	[10]	Select RCLK clock source 0 = Using Audio bus clock 1 = Using I2SCLK (Refer to Figure 2-3)	R/W	0
TXR	[9:8]	Transmit or receive mode select. 00 = Transmit only mode 01 = Receive only mode 10 = Transmit and receive simultaneous mode 11 = Reserved	R/W	00
LRP	[7]	Left/Right channel clock polarity select. 0 = Low for left channel and high for right channel 1 = High for left channel and low for right channel	R/W	0
SDF	[6:5]	Serial data format. 00 = IIS format 01 = MSB-justified (left-justified) format 10 = LSB-justified (right-justified) format 11 = Reserved	R/W	00
RFS	[4:3]	IIS root clock (codec clock) frequency select. 00 = 256 fs, where fs is sampling frequency 01 = 512 fs 10 = 384 fs 11 = 768 fs Note: Even in the slave mode, this bit should be set for correct operation.	R/W	00
BFS	[2:1]	Bit clock frequency select. 00 = 32 fs, where fs is sampling frequency 01 = 48 fs 10 = 16 fs 11 = 24 fs Note: Even in the slave mode, this bit should be set for correct operation.	R/W	00

IISMOD	Bit	Description	R/W	Initial State
Reserved	[0]	-	R	0

Samsung Confidential

2.9.1.3 IIS Interface FIFO Control Register (IISFIC, R/W, Address = 0xEEE3_0008)

IISFIC	Bit	Description	R/W	Initial State
Reserved	[31]	-	W	0
FTX2CNT	[30:24]	Primary TX FIFO2 data count. FIFO has 64 depth, so value ranges from 0 to 64. N: Data count N of FIFO	R	0x00
Reserved	[23]	-	R	0
FTX1CNT	[22:16]	Primary TX FIFO1 data count. FIFO has 64 depth, so value ranges from 0 to 64. N: Data count N of FIFO	R	0x00
TFLUSH	[15]	Primary TX FIFO flush command. 0 = No flush 1 = Flush	R/W	0
FTX0CNT	[14:8]	Primary TX FIFO0 data count. FIFO has 64 dept, so value ranges from 0 to 64. N: Data count N of FIFO	R	0x00
RFLUSH	[7]	RX FIFO flush command. 0 = No flush 1 = Flush	R/W	0
FRXCNT	[6:0]	RX FIFO data count. FIFO has 64 dept, so value ranges from 0 to 64. N: Data count N of FIFO	R	0x00

2.9.1.4 IIS Interface Clock Divider Control Register (IISPSR, R/W, Address = 0xEEE3_000C)

IISPSR	Bit	Description	R/W	Initial State
Reserved	[31:16]	-	R	0x00
PSRAEN	[15]	Pre-scaler (Clock divider) a active. 0 = Inactive 1 = Active	R/W	0
Reserved	[14]	-	R	0
PSVALA	[13:8]	Pre-scaler (Clock divider) a division value. N: Division factor is N+1	R/W	0x00
Reserved	[7:0]	-	R	0x00

2.9.1.5 IIS Interface Transmit Data Register (IISTXD, W, Address = 0xEEE3_0010)

IISTXD	Bit	Description	R/W	Initial State
IISTXD	[31:0]	Primary TX FIFO write data. Note that the left/right channel data is allocated as the following bit fields. R[31:16], L[15:0] when 16-bit BLC R[23:16], L[7:0] when 8-bit BLC Refer Figure 10.2-7 when 24-bit BLC	W	0x00

2.9.1.6 IIS Interface Receive Data Register (IISRXD, R, Address = 0xEEE3_0014)

IISRXD	Bit	Description	R/W	Initial State
IISRXD	[31:0]	RX FIFO read data. Note that the left/right channel data is allocated as the following bit fields. R[31:16], L[15:0] when 16-bit BLC R[23:16], L[7:0] when 8-bit BLC Refer Figure 10.2-9 when 24-bit BLC	R	0x00

2.9.1.7 IIS Interface TXFIFO_S Control Register (IISFICS, R/W, Address = 0xEEE3_0018)

IISFICS	Bit	Description	R/W	Initial State
Reserved	[31:16]	-	R	0x00
TFLUSHS	[15]	Secondary TX FIFO_S flush command. 0 = No flush 1 = Flush	R/W	0
FTXSCNT	[14:8]	Secondary TX FIFO_S data count. FIFO has 64 depth, so value ranges from 0 to 64. N: Data count N of FIFO	R	0x00
Reserved	[7:0]	-	R	0x00

2.9.1.8 IIS Interface Transmit Data Register for TXFIFO_S (IISTXDS, W, Address = 0xEEE3_001C)

IISTXDS	Bit	Description	R/W	Initial State
IISTXDS	[31:0]	Secondary TX FIFO_S write data. Note that the left/right channel data is allocated as the following bit fields. R[31:16], L[15:0] when 16-bit BLC R[23:16], L[7:0] when 8-bit BLC Refer Figure 10.2-7 when 24-bit BLC	W	0x00

2.9.1.9 IIS AHB DMA Control Register (II SAHB, R/W, Address = 0xEEE3_0020)

II SAHB	Bit	Description	R/W	Initial State
Reserved	[31:28]	-	R	0x00
IISLVL3EN	[27]	Enables buffer level 3 interrupt. 0 = Disables IISLVL3INT. 1 = Enables IISLVL3INT.	R/W	0
IISLVL2EN	[26]	Enables buffer level 2 interrupt. 0 = Disables IISLVL2INT. 1 = Enables IISLVL2INT.	R/W	0
IISLVL1EN	[25]	Enable buffer level 1 interrupt. 0 = Disables IISLVL1INT. 1 = Enables IISLVL1INT.	R/W	0
IISLVL0EN	[24]	Enable buffer level 0 interrupt. 0 = Disables IISLVL0INT. 1 = Enables IISLVL0INT.	R/W	0
IISLVL3INT	[23]	Buffer level 3 interrupt status flag. During operation of DMA, when generated address in DMA matches with IISLVL3ADDR, this flag will be set. To clear this flag, use IISLVL3CLR field.	R	0
IISLVL2INT	[22]	Buffer level 2 interrupt status flag. During operation of DMA, when generated address in DMA matches with IISLVL2ADDR, this flag will be set. To clear this flag, use IISLVL2CLR field.	R	0
IISLVL1INT	[21]	Buffer level 1 interrupt status flag. During operation of DMA, when generated address in DMA matches with IISLVL1ADDR, this flag will be set. To clear this flag, use IISLVL1CLR field.	R	0
IISLVL0INT	[20]	Buffer level 0 interrupt status flag. During operation of DMA, when generated address in DMA matches with IISLVL0ADDR, this flag will be set. To clear this flag, use IISLVL0CLR field.	R	0
IISLVL3CLR	[19]	Clear IISLVL3INT flag When IISLVL3INT is set, setting IISLVL3CLR to 1 will clear IISLVL3INT to 0. Writing zero has no effect.	R/W	0

IISAHB	Bit	Description	R/W	Initial State
IISLVL2CLR	[18]	Clear IISLVL2INT flag When IISLVL2INT is set, setting IISLVL2CLR to 1 will clear IISLVL2INT to 0. Writing zero has no effect.	R/W	0
IISLVL1CLR	[17]	Clear IISLVL1INT flag When IISLVL1INT is set, setting IISLVL1CLR to 1 will clear IISLVL1INT to 0. Writing zero has no effect.	R/W	0
IISLVL0CLR	[16]	Clear IISLVL0INT flag When IISLVL0INT is set, setting IISLVL0CLR to 1 will clear IISLVL0INT to 0. Writing zero has no effect.	R/W	0
Reserved	[15:8]	-	R	0x00
IISDMA_STR ADDRST	[7]	DMA start address reset Before starting address toggle, write 1 to this bit. After reset, this bit is auto-cleared.	W	0x0
IISDMA_STR ADDRTOG	[6]	DMA start address toggle 0 = Disables start address toggling (IISSTR0 → IISSTR0 → ...) 1 = Enables start address toggling (IISSTR0 → IISSTR1 → IISSTR0 → IISSTR1 → ...)	R/W	0
IISDMARLD-	[5]	Auto-reload IIS internal DMA Configuration when DMA operation is done and re-start IIS internal DMA automatically. 0 = Disables auto-reload function 1 = Enables auto-reload function Before switching to 0 from 1, s/w must check if DMA_EN is set.	R/W	0
IISINTMASK	[3]	Disables interrupt request signal 0 = Enables interrupt request when DMA auto-reload is on. 1 = Disables interrupt request when DMA auto-reload is on. After DMA transfers all of data related to DMA configuration, interrupt signal will occur. If IISINTMASK bit is set, IISDMAINT & interrupt signal will NOT be set. IISINTMASK does NOT effect IISLVLxINT & under-run interrupt.	R/W	0
IISDMAINT	[2]	DMA interrupt status flag. After DMA operation is end, this flag will be set. To clear this flag, use IISDMACLR field. When ARM is used for decoder, do not use this interrupt as controlling timing of filling buffer. In that case, use level 0~3 interrupts. This interrupt is just used for ending condition.	R	0
IISDMACLR	[1]	Clear DMA interrupt status flag When IISINT is set, setting IISDMACLR to 1 will clear IISDMAINT to 0. Writing to zero is no meaning.	R/W	0

IISAHB	Bit	Description	R/W	Initial State
IISDMAEN	[0]	<p>Enable IIS internal DMA</p> <p>Users can use internal DMA in IIS after this bit field is ON. Internal DMA can issue 32-bit single read transaction for AHB and TXFIFO0 will hold data returned by DMA.</p> <p>Warning></p> <p>If IISDMARLD is set, IISDMAEN bit will be automatically cleared when reload operation is in progress. After auto-reload operation is done, IISDMAEN bit will be automatically set.</p> <p>When auto-reload operation is in progress, s/w intervention on this field will cause mal-function of internal DMA operations. To manipulate IISAHB register, s/w must check that IISDMAEN is in stable state.</p>	R/W	0

2.9.1.10 IIS AHB DMA Start Address0 Register (IISSTR0, R/W, Address = 0xEEE3_0024)

IISSTR0	Bit	Description	R/W	Initial State
IISSTR	[31:0]	<p>Start address0 of IIS internal DMA operation.</p> <p>When DMAEN is ON, internal DMA in IIS will start DMA operation based on IISSTR0 address.</p> <p>Internal DMA can handle word-aligned address only but to get best performance, IISSTR0 should be 64 word-aligned address.</p>	R/W	0x00

2.9.1.11 IIS AHB DMA Start Address1 Register (IISSTR1, R/W, Address = 0xEEE3_0040)

IISSTR1	Bit	Description	R/W	Initial State
IISSTR1	[31:0]	<p>Start address1 of IIS internal DMA operation.</p> <p>When DMAEN is ON, internal DMA in IIS will start DMA operation based on IISSTR1 address.</p> <p>Internal DMA can handle word-aligned address only, but to achieve best performance, IISSTR1 should be 64 word-aligned address.</p>	R/W	0x00

2.9.1.12 IIS AHB DMA Size Register (IISSIZE, R/W, Address = 0xEEE3_0028)

IISSIZE	Bit	Description	R/W	Initial State
TRNS_SIZE	[31:16]	Transfer block size for IIS internal DMA When IIS internal DMA is enabled, IIS internal DMA will transfer TRNS_SIZE word(s) data from memory before DMA done interrupt occurs. Valid ranges for TRNS_SIZE will be from 0x0001 to 0xA000. 0x0001 – 0xA000 : IMEM and DMEM at AUDIO Sub-System (160Kbytes)	R/W	0x7FFF
Reserved	[15:0]	-	R	0x0000

2.9.1.13 IIS AHB DMA Transfer Count Register (IISTRNCNT, R, Address = 0xEEE3_002C)

IISTRNCNT	Bit	Description	R/W	Initial State
Reserved	[31:24]	-	R	
IISTRNCNT	[23:0]	Number of transferred data using IIS internal DMA. (word unit) User program can terminate IIS internal DMA operation by turning DMA_EN off. After DMA_EN is 0, user program reads IISTRNCNT value to know where IIS internal DMA stops.	R	

2.9.1.14 IIS AHB DMA Level 0 Interrupt Address Register (IISLVL0ADDR, R/W, Address = 0xEEE3_0030)

IISLVL0ADDR	Bit	Description	R/W	Initial State
IISLVL0ADDR	[31:10]	AHB DMA level 0 interrupt address While IISLVL0EN in IISAHB register is set, AHB DMA is comparing this register to generated address in DMA. When two values match, IISLVL0INT in IISAHB will be set. Valid address range for IISLVL0ADDR is from 0xC000_0000 to 0xC01F_FFFF.	R	0x00
Reserved	[9:1]	-	R	0x00
IISLVL0STOP	[0]	Enables Precise stop 0 = Do not stop DMA operation 1 = Stop DMA operation when DMA working address is matched with IISLVL0ADDR. IISDMAEN in IISAHB will be turned off automatically.	R/W	0

2.9.1.15 IIS AHB DMA Level 1 Interrupt Address Register (IISLVL1ADDR, R/W, Address = 0xEEE3_0034)

IISLVL1ADDR	Bit	Description	R/W	Initial State
IISLVL1ADDR	[31:10]	AHB DMA level 1 interrupt address While IISLVL1EN in IISAHB register is set, AHB DMA is comparing this register to generated address in DMA. When two values match, IISLVL1INT in IISAHB will be set. Valid address range for IISLVL1ADDR is from 0xC000_0000 to 0xC01F_FFFF.	R	0x00
Reserved	[9:1]	-	R	0x00
IISLVL1STOP	[0]	Enables Precise stop 0 = Do not stop DMA operation 1 = Stop DMA operation when DMA working address is matched with IISLVL1ADDR. IISDMAEN in IISAHB will be turned off automatically.	R/W	0

2.9.1.16 IIS AHB DMA Level 2 Interrupt Address Register (IISLVL2ADDR, R/W, Address = 0xEEE3_0038)

IISLVL2ADDR	Bit	Description	R/W	Initial State
IISLVL2ADDR	[31:10]	AHB DMA level 2 interrupt address While IISLVL2EN in IISAHB register is set, AHB DMA is comparing this register to generated address in DMA. When two values match, IISLVL2INT in IISAHB will be set. Valid address range for IISLVL2ADDR is from 0xC000_0000 to 0xC01F_FFFF.	R	0x00
Reserved	[9:1]	-	R	0x00
IISLVL2STOP	[0]	Enables Precise stop 0 = Do not stop DMA operation 1 = Stop DMA operation when DMA working address is matched with IISLVL2ADDR. IISDMAEN in IISAHB will be turned off automatically.	R/W	0

2.9.1.17 IIS AHB DMA Level 3 Interrupt Address Register (IISLVL3ADDR, R/W, Address = 0xEEE3_003C)

IISLVL3ADDR	Bit	Description	R/W	Initial State
IISLVL3ADDR	[31:10]	AHB DMA level 3 interrupt address While IISLVL3EN in IISAHB register is set, AHB DMA is comparing this register to generated address in DMA. When two values match, IISLVL3INT in IISAHB will be set. Valid address range for IISLVL3ADDR is from 0xC000_0000 to 0xC01F_FFFF.	R	0x00
Reserved	[9:1]	-	R	0x00
IISLVL3STOP	[0]	Enables Precise stop 0 = Do not stop DMA operation 1 = Stop DMA operation when DMA working address is matched with IISLVL3ADDR. IISDMAEN in IISAHB will be turned off automatically.	R/W	0

3 IIS-BUS INTERFACE

3.1 OVERVIEW OF IIS-BUS INTERFACE

Inter-IC Sound (IIS) is one of the popular digital audio interface. The IIS bus handles audio data and other signals, namely, sub-coding and control, are transferred separately. It is possible to transmit data between two IIS bus. To minimize the number of pins required and to keep wiring simple, basically, a 3-line serial bus is used. This consists of a line for two time-multiplexed data channels, a word select line and a clock line.

IIS interface transmits or receives sound data from external stereo audio codec. To transmit and receive data, two 32x64 FIFOs (First-In-First-Out) data structures are included and DMA transfer mode to transmit and receive samples can be supported. IIS-specific clock can be supplied from internal system clock controller through IIS clock divider or direct clock source.

3.2 KEY FEATURES OF IIS-BUS INTERFACE

- 2-ports stereo(2ch) IIS-bus for audio interface with DMA-based operation
- Serial, 8/16/24-bit per channel data transfers
- Supports master/slave mode
- Supports IIS, MSB-justified and LSB-justified data format

3.3 BLOCK DIAGRAM OF IIS-BUS INTERFACE

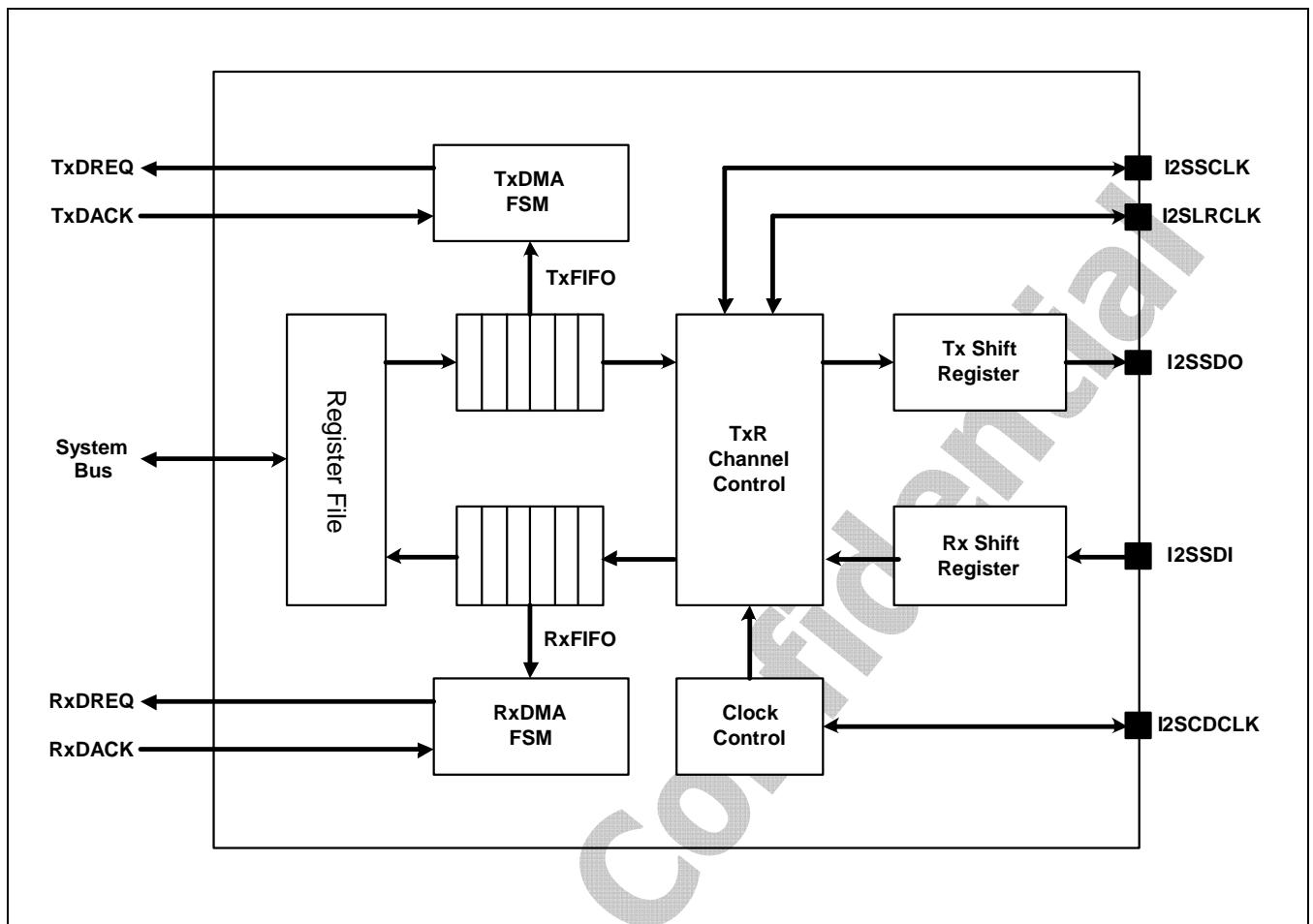


Figure 3-1 IIS-Bus Block Diagram

3.4 FUNCTIONAL DESCRIPTIONS

IIS interface consists of register bank, FIFOs, shift registers, clock control, DMA finite state machine, and channel control block as shown in [Figure 3-1](#). Note that each FIFO has 32-bit width and 64 depth structure, which contains left/right channel data. Thus, FIFO access and data transfer are handled with left/right pair unit. [Figure 3-1](#) shows the functional block diagram of IIS interface.

3.4.1 MASTER/SLAVE MODE

To select master or slave mode, set IMS bit of IISMOD register. In master mode, I2SSCLK and I2SLRCLK are generated internally and supplied to external device. Therefore, a root clock is required to generate I2SSCLK and I2SLRCLK. The IIS pre-scaler (clock divider) is employed to generate a root clock with divided frequency from internal system clock. In external master mode, the root clock can be directly fed from IIS external. The I2SSCLK and I2SLRCLK are supplied from the pin (GPIOs) in slave mode. That is, whatever source clock is, Only Master can generate I2SLRCLK and I2SSCLK.

Master/Slave mode is different compared to TX/RX. Master/Slave mode presents the direction of I2SLRCLK and I2SSCLK. The direction of I2SCDCLK (This is only auxiliary.) is not important. If IIS bus interface transmits clock signals to IIS codec, IIS bus is master mode. But if IIS bus interface receives clock signal from IIS codec, IIS bus is slave mode. TX/RX mode indicates the direction of data flow. If IIS bus interface transmits data to IIS codec, this indicates TX mode. Conversely, IIS bus interface receives data from IIS codec this indicates RX mode.

[Figure 3-2](#) shows the route of the root clock with internal master or external master mode setting in IIS clock control block and system controller. Note that RCLK indicates root clock and this clock can be supplied to external IIS codec chip in internal master mode.

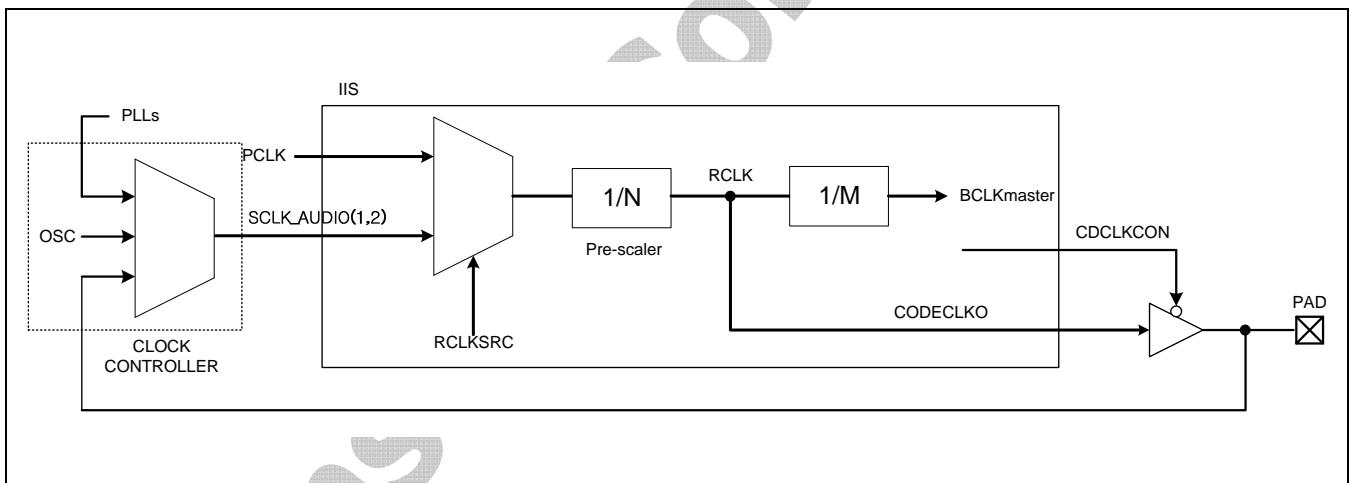


Figure 3-2 IIS Clock Control Block Diagram

3.4.2 DMA TRANSFER

In the DMA transfer mode, use external DMA controller to access the transmitter or receiver FIFO. The transmitter or receiver FIFO state activates the DMA service request internally. The FTXEMPT, FRXEMPT, FTXFULL, and FRXFULL bits of I2SCON register represent the transmitter or receiver FIFO data state. Especially, FTXEMPT and FRXFULL bit are the ready flag for DMA service request; the transmit DMA service request is activated when TXFIFO is not empty and the receiver DMA service request is activated when RXFIFO is not full.

The DMA transfer uses only handshaking method for single data. Note that during DMA acknowledge activation; the data read or write operation should be performed.

* Reference: DMA request point

- TX mode: (FIFO is not full) & (TXDMAACTIVE is active)
- RX mode: (FIFO is not empty) & (RXDMAACTIVE is active)

3.4.3 AUDIO SERIAL DATA FORMAT

3.4.3.1 IIS-bus Format

The IIS bus has four lines including serial data input I2SSDI, serial data output I2SSDO, left/right channel select clock I2SLRCLK, and serial bit clock I2SSCLK; master generates I2SLRCLK and I2SSCLK.

Serial data is transmitted in 2's complement with the MSB first with a fixed position, whereas the position of the LSB depends on the word length. The transmitter sends the MSB of the next word at one clock period after the I2SLRCLK is changed. Serial data sent by the transmitter can be synchronized either with the trailing or with the leading edge of the clock signal. However, the serial data must be latched into the receiver on the leading edge of the serial clock signal, and so there are some restrictions when transmitting data that is synchronized with the leading edge.

The LR channel select line indicates the channel being transmitted. I2SLRCLK may be changed either on a trailing or leading edge of the serial clock, but it does not need to be symmetrical. In the slave, this signal is latched on the leading edge of the clock signal. The I2SLRCLK line changes one clock period before the MSB is transmitted. This allows the slave transmitter to derive synchronous timing of the serial data that will be set up for transmission. Furthermore, it enables the receiver to store the previous word and clear the input for the next word.

3.4.3.2 MSB (Left) Justified

MSB-Justified (Left-Justified) format is similar to IIS bus format, except that in MSB-justified format, the transmitter always sends the MSB of the next word at the same time whenever the I2SLRCLK is changed.

3.4.3.3 LSB (Right) Justified

LSB-Justified (Right-Justified) format is opposite to the MSB-justified format. In other word, the transferring serial data is aligned with ending point of I2SLRCLK transition.

[Figure 3-3](#) shows the audio serial format of IIS, MSB-justified, and LSB-justified. Note that in this figure, the word length is 16-bit and I2SLRCLK makes transition every 24 cycle of I2SSCLK (BFS is 48 fs, where f_s is sampling frequency; I2SLRCLK frequency).

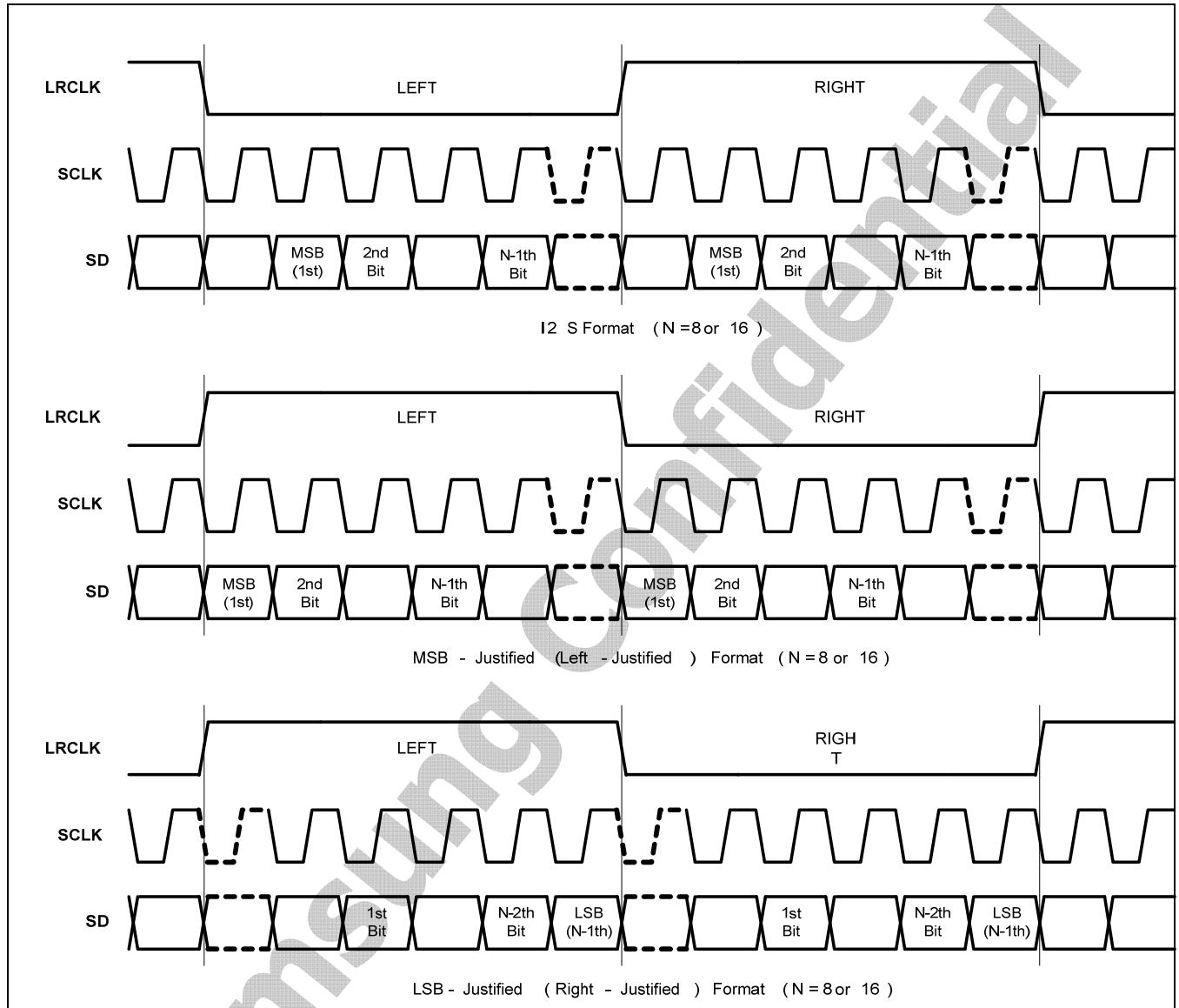


Figure 3-3 IIS Audio Serial Data Formats

3.4.3.4 Sampling Frequency and Master Clock

When IIS interface Controller operates as master, IIS interface Controller generates IISLRCLK and IISSCLK that are divided as Root Clock by RFS and BFS value. To decide Sampling Frequency – IISLRCLK -, BLC, BFS, and RFS are selected first. Optionally, IIS interface Controller clocks out Root clock as IISCDCLK for codec master clock (if source of root clock is not ISEXTCDCLK).

In slave mode, you must set the value of BLC, BFS and RFS similar to master (ex: Codec). Because IIS interface controller needs these value for correct operation.

3.4.4 PCM WORD LENGTH AND BFS DIVIDER

PCM Word Length (BLC) setting should be preceded before setting the BFS value. [Table 3-1](#) shows BFS available value as BLC.

Table 3-1 Allowed BFS Value as BLC

PCM Bit length(BLC)	8bit	16bit	24bit
Available BFS value	16fs, 24fs, 32fs, 48fs	32fs, 48fs	48fs

3.4.5 BFS DIVIDER AND RFS DIVIDER

RFS value is selected as BFS selected. [Table 3-2](#) shows RFS available value as BFS.

Table 3-2 Allowed RFS Value as BFS

BFS Divider	16fs, 32fs	24fs, 48fs
Available RFS value	256fs, 384fs, 512fs, 768fs.	384fs, 768fs.

3.4.6 RFS DIVIDER AND ROOT CLOCK

Root Clock is made for sampling frequency proper RFS value as shown in [Table 3-3](#). RCLK is clock divided by IIS pre-scaler(IISPSR) that is selected by IMS

Table 3-3 Root Clock Table (MHz)

IISLRCK RFS \	8.000 kHz	11.025 kHz	16.000 kHz	22.050 kHz	32.000 kHz	44.100 kHz	48.000 kHz	64.000 kHz	88.200 kHz	96.000 kHz
256fs	2.0480	2.8224	4.0960	5.6448	8.1920	11.2896	12.2880	16.3840	22.5792	24.5760
384fs	3.0720	4.2336	6.1440	8.4672	12.2880	16.9344	18.4320	24.5760	33.8688	36.8640
512fs	4.0960	5.6448	8.1920	11.2896	16.3840	22.5792	24.5760	32.7680	45.1584	49.1520
768fs	6.1440	8.4672	12.2880	16.9344	24.5760	33.8688	36.8640	49.1520	67.7376	73.7280

Root Clock Frequency = fs * (256, 384, 512 or 768)

3.5 PROGRAMMING GUIDE

The IIS bus interface can be accessed either by the processor using programmed I/O instructions or by the DMA controller.

3.5.1 INITIALIZATION

1. Before you use IIS bus interface, you must configure GPIOs to IIS mode, that is, I2SSDI is input and I2SSDO is output. I2SLRCLK, I2SSCLK and I2SCDCLK is inout-type.
2. Select clock source. S5PV210 has three clock sources, namely, PCLK, EPLL and external codec. For more information, refer [Figure 3-2](#).

3.5.2 PLAY MODE (TX MODE) WITH DMA

1. TXFIFO is flushed before operation. If you do not distinguish Master/Slave mode from TX/RX mode, you must study Master/Slave mode and TX/RX mode. Refer Master/Slave chapter.
2. Configure I2SMOD register and I2SPSR (IIS pre-scaler register).
3. To operate system in stability, the internal TXFIFO should be almost full before transmission. For TXFIFO to be almost full start DMA operation.
4. IIS bus does not support the interrupt. Therefore, you can only check state by polling through accessing SFR.
5. After TXFIFO is full, then I2SACTIVE must be asserted.

3.5.3 RECORDING MODE (RX MODE) WITH DMA

1. RXFIFO is flushed before operation. Also, if you don't distinguish between Master/Slave mode and TX/RX mode, you must study Master/Slave mode and TX/RX mode. Refer Master/Slave chapter.
2. Configure I2SMOD register and I2SPSR (IIS pre-scaler register).
3. To operate system in stability, the internal RXFIFO should have at least one data before DMA operation. You must assert I2SACTIVE before DMA operation.
4. Check RXFIFO state by polling through accessing SFR.
5. If RXFIFO is not empty, start RXDMACTIVE.

3.5.4 EXAMPLE CODE

3.5.4.1 Tx Channel

The I2S TX channel provides a single stereo compliant output. The transmit channel can operate in master or Slave mode. Data is transferred between the processor and the I2S controller via an APB access or a DMA access.

The processor must write words in multiples of two (i.e. for left and right audio sample). The words are serially shifted out timed with respect to the audio serial bitclk, SCLK and word select clock, LRCLK.

TX Channel has 64X32-bit wide FIFO where the processor or DMA can write upto 16 left/right data samples After enabling the channel for transmission.

An Example sequence is as follows:

Ensure the PCLK and CDCLK are coming correctly to the I2S controller and FLUSH the TX FIFO using the TFLUSH bit in the Please ensure that I2S Controller is configured in one of the following modes.

- TX only mode
- TX/RX simultaneous mode

The Data is aligned in the TX FIFO for 8-bits/channel or 16-bits/channel BLC as shown in [Figure 3-4](#).

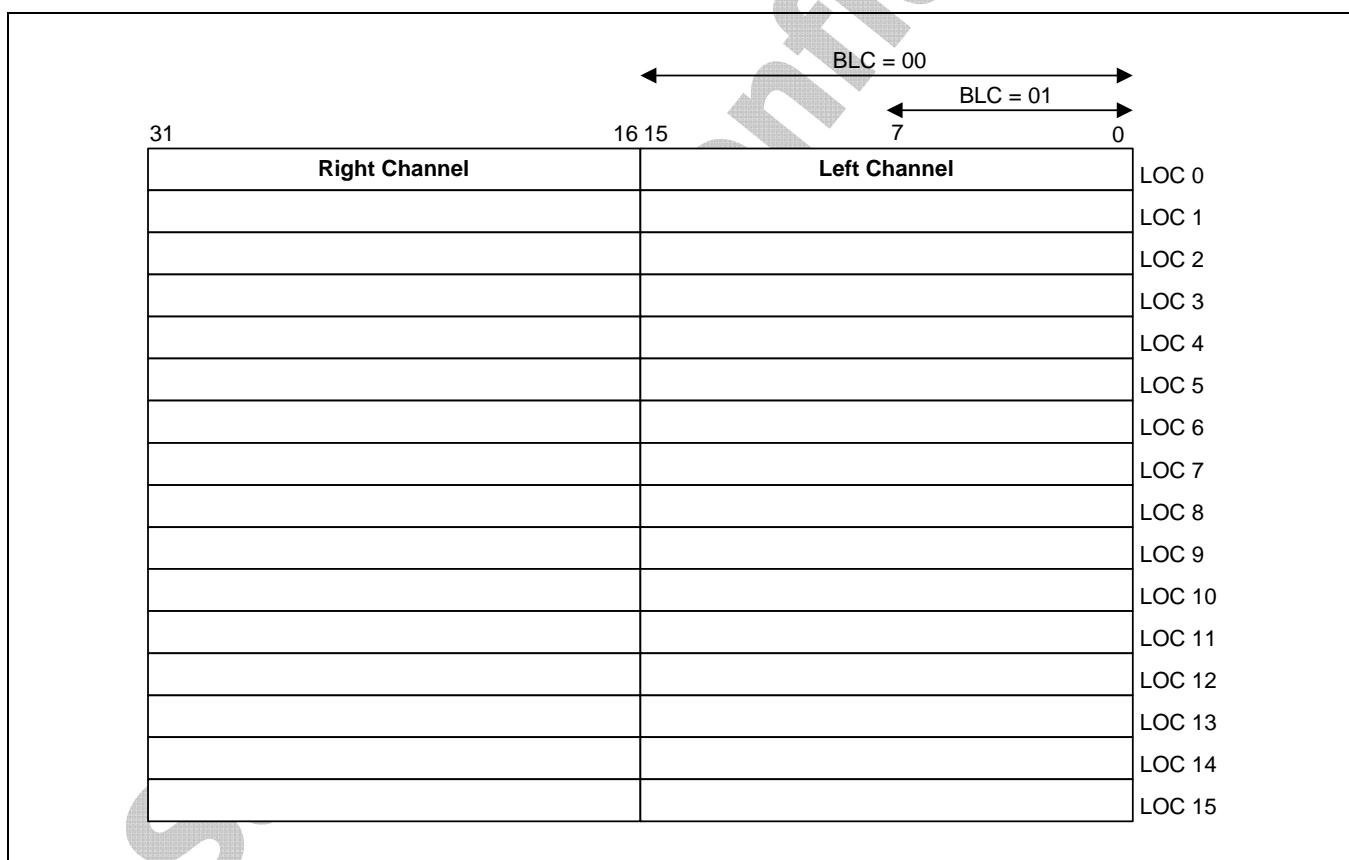


Figure 3-4 TX FIFO Structure for BLC = 00 or BLC = 01

The Data is aligned in the TX FIFO for 24-bit/channel BLC as shown in [Figure 3-5](#).

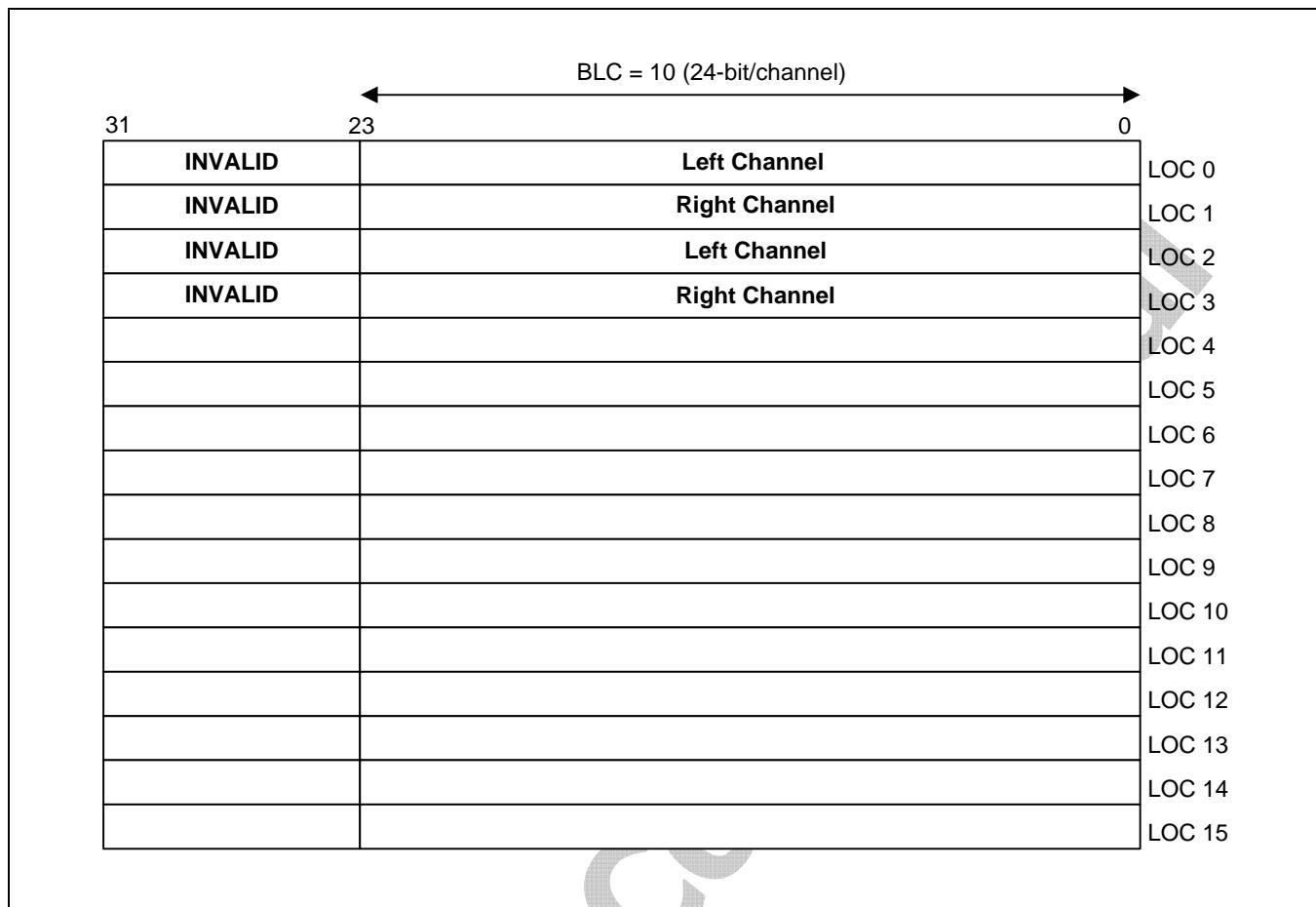


Figure 3-5 TX FIFO Structure for BLC = 10 (24-bit/channel)

Once the data is written to the TX FIFO the TX channel can be made active by enabling the I2SACTIVE bit in the I2SCON Register (I2S Control Register).

The data is then serially shifted out with respect to the serial bit clock SCLK and word select clock LRCLK.

The TXCHPAUSE in the I2SCON Register (I2S Control Register) can stop the serial data transmission on the I2SSDO. The transmission is stopped once the current Left/Right channel is transmitted.

If the control registers in the I2SCON Register (I2S Control Register) and I2SMOD Register (I2S Mode Register) are to be reprogrammed then it is advisable to disable the TX channel.

If the TX channel is enabled while the FIFO is empty, no samples are read from the FIFO.

The Status of TX FIFO can be checked by checking the bits in the I2SFIC Register (I2S FIFO Control Register).

3.5.4.2 RX Channel

The I2S RX channel provides a single stereo compliant output. The receive channel can operate in master or slave mode. Data is received from the input line and transferred into the RX FIFO. The processor can then read this data via an APB read or a DMA access can access this data.

RX Channel has a 64X32-bit wide RX FIFO where the processor or DMA can read UPTO 16 left/right data samples after enabling the channel for reception.

An Example sequence is as follows:

Ensure the PCLK and CDCLK are coming correctly to the I2S controller and FLUSH the RX FIFO using the RFLUSH bit in the I2SFIC Register (I2S FIFO Control Register) and the I2S controller is configured in any of the modes

- Receive only.
- Receive/Transmit simultaneous mode

This can be done by Programming the TXR bit in the I2SMOD Register (I2S Mode Register)

1. Then Program the following parameters according to the need
 - MMS, RCLKSRC
 - SDF
 - BFS
 - BLC
 - LRP

For Programming, the above mentioned fields please refer I2SMOD Register (I2S Mode Register)
2. Once ensured that the input clocks for I2S controller are up and running and step 1 and 2 have been completed user must put the I2SACTIVE high to enable any reception of data, the I2S Controller receives data on the LRCLK change.

Read the data from the RX FIFO using the I2SRXD Register (I2S RX FIFO Register) after looking at the RX FIFO count in the I2SFIC Register (I2S FIFO Control Register). The count would only increment once the complete left channel and right have been received.

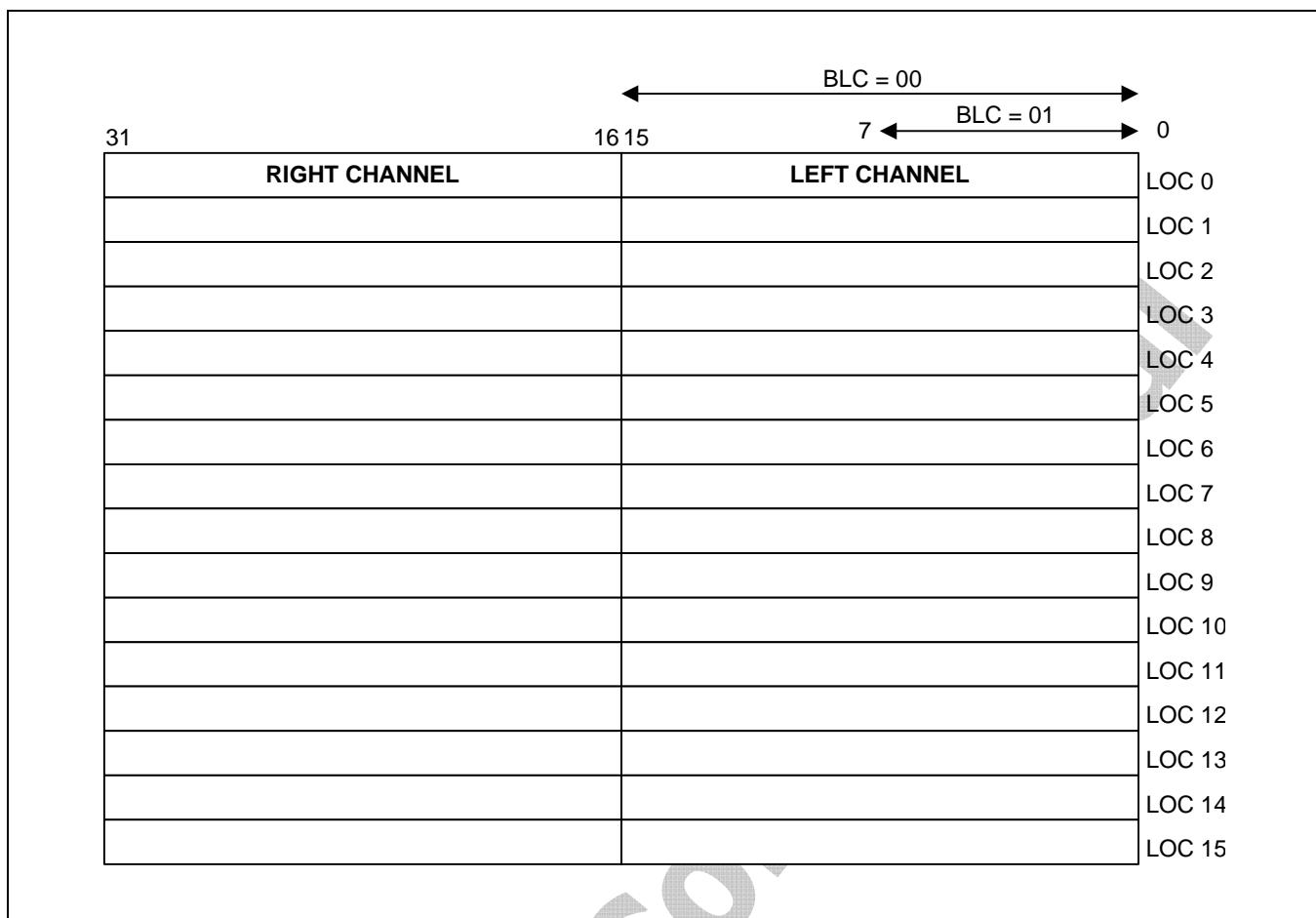


Figure 3-6 RX FIFO Structure for BLC = 00 or BLC = 01

The Data is aligned in the RX FIFO for 24-bit/channel BLC as shown [Figure 3-7](#).

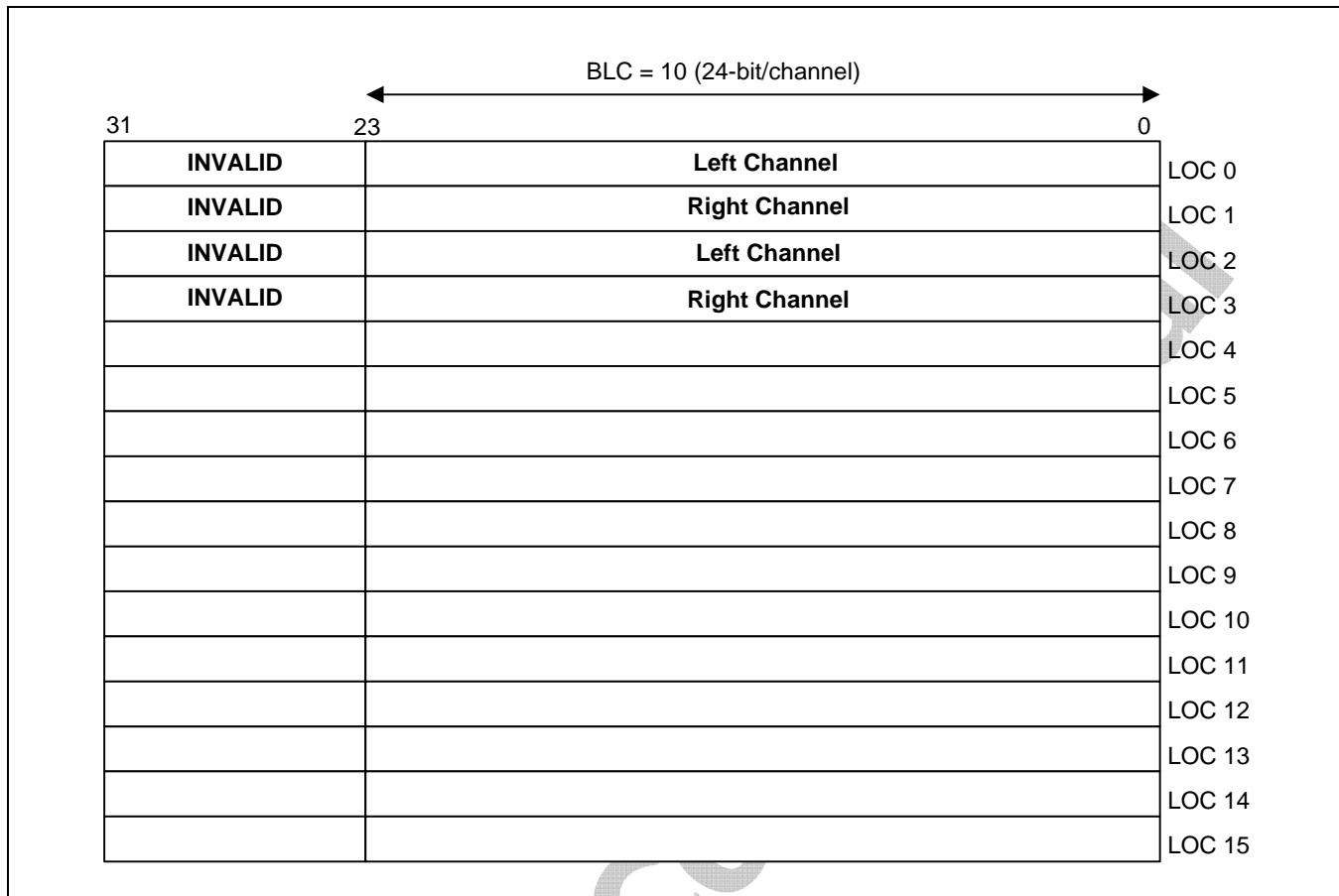


Figure 3-7 RX FIFO Structure for BLC = 10 (24-bits/channel)

The RXCHPAUSE in the I2SCON register can stop the serial data reception on the I2SSDI. The reception is stopped once the current Left/Right channel is received.

If the control registers in the I2SCON Register (I2S Control Register) and I2SMOD Register (I2S Mode Register) are to be reprogrammed then it is advisable to disable the RX channel.

Check the status of RX FIFO by checking the bits in the I2SFIC Register (I2S FIFO Control Register).

3.6 I/O DESCRIPTION

Each I2S (v3.2) external pads are shared with I2S and PCM. In order to use these pads for I2S, GPIO must be set before the I2S started. For mode information, refer to the GPIO chapter of this manual for proper GPIO setting

PAD Name	I/O	Description	Pad	Type
Xi2s1CDCLK, Xpcm2EXTCLK	I/O	I2S Codec clock input/output		dedicated
Xi2s1SCLK, Xpcm2SCLK	I/O	I2S Bit clock input/output		dedicated
Xi2s1LRCK, Xpcm2FSYNC	I/O	I2S LR channel clock input/output		dedicated
Xi2s1SDI, Xpcm2SIN	I	I2S serial data input		dedicated
Xi2s1SDO, Xpcm2SOUT	O	I2S serial data out		dedicated

3.7 REGISTER DESCRIPTION

3.7.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
I2S1				
IISCON	0xE210_0000	R/W	Specifies the IIS interface control register	0xE00
IISMOD	0xE210_0004	R/W	Specifies the IIS interface mode register	0x0
IISFIC	0xE210_0008	R/W	Specifies the IIS interface FIFO control register	0x0
IISPSR	0xE210_000C	R/W	Specifies the IIS interface clock divider control register	0x0
IISTXD	0xE210_0010	W	Specifies the IIS interface transmit data register	0x0
IISRXD	0xE210_0014	R	Specifies the IIS interface receive data register	0x0
I2S2				
IISCON	0xE2A0_0000	R/W	Specifies the IIS interface control register	0xE00
IISMOD	0xE2A0_0004	R/W	Specifies the IIS interface mode register	0x0
IISFIC	0xE2A0_0008	R/W	Specifies the IIS interface FIFO control register	0x0
IISPSR	0xE2A0_000C	R/W	Specifies the IIS interface clock divider control register	0x0
IISTXD	0xE2A0_0010	W	Specifies the IIS interface transmit data register	0x0
IISRXD	0xE2A0_0014	R	Specifies the IIS interface receive data register	0x0

NOTE: All registers of IIS interface are accessible by word unit with STR/LDR instructions.

3.7.1.1 IIS-BUS Interface Special Registers (IISCON)

- IISCON, R/W, Address = 0xE210_0000
- IISCON, R/W, Address = 0xE2A0_0000

IISCON	Bit	Description	R/W	Initial State
Reserved	[31:20]	Reserved. Program to zero.	R/W	12'b0
FRXOFSTATUS	[19]	RX FIFO OverFlow Interrupt Status. And this is used by interrupt clear bit. When this is high, you can do interrupt clear by writing '1'. 0 = Interrupt didn't be occurred. 1 = Interrupt was occurred.	R/W	1'b0
FRXOFINTEN	[18]	RX FIFO OverFlow Interrupt Enable 0 = RXFIFO Under-run INT disable 1 = RXFIFO Under-run INT enable	R/W	1'b0
FTXURSTATUS	[17]	TX FIFO under-run interrupt status. And this is used by interrupt clear bit. When this is high, you can do interrupt clear by writing '1'. 0 = Interrupt didn't be occurred. 1 = Interrupt was occurred.	R/W	1'b0
FTXURINTEN	[16]	TX FIFO Under-run Interrupt Enable 0 = TXFIFO Under-run INT disable 1 = TXFIFO Under-run INT enable	R/W	1'b0
Reserved	[15:12]	Reserved. Program to zero.	R/W	4'b0
LRI	[11]	Left/Right channel clock indication. Note that LRI meaning is dependent on the value of LRP bit of I2SMOD register. 0 = Left (when LRP bit is low) or right (when LRP bit is high) 1 = Right (when LRP bit is low) or left (when LRP bit is high)	R	1'b1
FTXEMPT	[10]	Tx FIFO empty status indication. 0 = FIFO is not empty (ready for transmit data to channel) 1 = FIFO is empty (not ready for transmit data to channel)	R	1'b1
FRXEMPT	[9]	Rx FIFO empty status indication. 0 = FIFO is not empty 1 = FIFO is empty	R	1'b1
FTXFULL	[8]	Tx FIFO full status indication. 0 = FIFO is not full 1 = FIFO is full	R	1'b0
FRXFULL	[7]	Rx FIFO full status indication. 0 = FIFO is not full (ready for receive data from channel) 1 = FIFO is full (not ready for receive data from channel)	R	1'b0

IISCON	Bit	Description	R/W	Initial State
TXDMAPAUSE	[6]	Tx DMA operation pause command. Note that when this bit is activated at any time, the DMA request will be halted after current on-going DMA transfer is completed. 0 = No pause DMA operation 1 = Pause DMA operation	R/W	1'b0
RXDMAPAUSE	[5]	Rx DMA operation pause command. Note that when this bit is activated at any time, the DMA request will be halted after current on-going DMA transfer is completed. 0 = No pause DMA operation 1 = Pause DMA operation	R/W	1'b0
TXCHPAUSE	[4]	Tx channel operation pause command. Note that when this bit is activated at any time, the channel operation will be halted after left-right channel data transfer is completed. 0 = No pause operation 1 = Pause operation	R/W	1'b0
RXCHPAUSE	[3]	Rx channel operation pause command. Note that when this bit is activated at any time, the channel operation will be halted after left-right channel data transfer is completed. 0 = No pause operation 1 = Pause operation	R/W	1'b0
TXDMACTIVE	[2]	Tx DMA active (start DMA request). Note that when this bit is set from high to low, the DMA operation will be forced to stop immediately. 0 = Inactive 1 = Active	R/W	1'b0
RXDMACTIVE	[1]	Rx DMA active (start DMA request). Note that when this bit is set from high to low, the DMA operation will be forced to stop immediately. 0 = Inactive 1 = Active	R/W	1'b0
I2SACTIVE	[0]	IIS interface active (start operation). 0 = Inactive 1 = Active	R/W	1'b0

3.7.1.2 IIS-BUS Interface Special Registers (IISMOD)

- IISMOD, R/W, Address = 0xE210_0004
- IISMOD, R/W, Address = 0xE2A0_0004

IISMOD	Bit	Description	R/W	Initial State
Reserved	[31:15]	Reserved. Program to zero.	R/W	1'b0
BLC	[14:13]	Bit Length Control Bit Which decides transmission of 8/16 bits per audio channel 00 = 16 Bits per channel 01 = 8 Bits Per Channel 10 = 24 Bits Per Channel 11 = Reserved	R/W	2'b00
CDCLKCON	[12]	Determine codec clock source 0 = Use internal codec clock source 1 = Get codec clock source from external codec chip * 0 means External CDCLK Input pad enable (Refer to Figure 3-2)	R/W	1'b0
MSS	[11]	IIS master or slave mode select. 0 = Master mode 1 = Slave mode	R/W	1'b0
RCLKSRC	[10]	Select RCLK clock source 0 = PCLK is internal source clock for IIS 1 = SCLK_AUDIO (SCLK_AUDIO1 for I2S1, SCLK_AUDIO2 for I2S2) (Refer to Figure 3-2)	R/W	1'b0
TXR	[9:8]	Transmit or receive mode select. 00 = Transmit only mode 01 = Receive only mode 10 = Transmit and receive simultaneous mode 11 = Reserved	R/W	2'b00
LRP	[7]	Left/Right channel clock polarity select. 0 = Low for left channel and high for right channel 1 = High for left channel and low for right channel	R/W	1'b0
SDF	[6:5]	Serial data format. 00 = IIS format 01 = MSB-justified (left-justified) format 10 = LSB-justified (right-justified) format 11 = Reserved	R/W	2'b00
RFS	[4:3]	IIS root clock (codec clock) frequency select. 00 = 256 fs, where fs is sampling frequency 01 = 512 fs 10 = 384 fs 11 = 768 fs	R/W	2'b00

IISMOD	Bit	Description	R/W	Initial State
BFS	[2:1]	Bit clock frequency select. 00 = 32 fs, where fs is sampling frequency 01 = 48 fs 10 = 16 fs 11 = 24 fs	R/W	2'b00
Reserved	[0]	Reserved. Program to zero.	R/W	1'b0

3.7.1.3 IIS-BUS Interface Special Registers (IISFIC)

- IISFIC, R/W, Address = 0xE210_0008
- IISFIC, R/W, Address = 0xE2A0_0008

IISFIC	Bit	Description	R/W	Initial State
Reserved	[31:16]	Reserved. Program to zero.	R/W	16'b0
TFLUSH	[15]	TX FIFO flush command. 0 = No flush 1 = Flush	R/W	1'b0
FTXCNT	[14:8]	TX FIFO data count. FIFO has 64 dept, so value ranges from 0 to 64. N: Data count N of FIFO	R	7'b0
RFLUSH	[7]	RX FIFO flush command. 0 = No flush 1 = Flush	R/W	1'b0
FRXCNT	[6:0]	RX FIFO data count. FIFO has 64 dept, so value ranges from 0 to 64. N: Data count N of FIFO	R	7'b0

3.7.1.4 IIS-BUS Interface Special Registers (IISPSR)

- IISPSR, R/W, Address = 0xE210_000C
- IISPSR, R/W, Address = 0xE2A0_000C

IISPSR	Bit	Description	R/W	Initial State
Reserved	[31:16]	Reserved. Program to zero.	R/W	16'b0
PSRAEN	[15]	Pre-scaler (Clock divider) A active. 0 = Inactive 1 = Active	R/W	1'b0
Reserved	[14]	Reserved. Program to zero.	R/W	1'b0
PSVALA	[13:8]	Pre-scaler (Clock divider) A division value. N: Division factor is N+1	R/W	6'b0
Reserved	[7:0]	Reserved. Program to zero.	R/W	8'b0

3.7.1.5 IIS-BUS Interface Special Registers (IISTXD)

- IISTXD, W, Address = 0xE210_0010
- IISTXD, W, Address = 0xE2A0_0010

IISTXD	Bit	Description	R/W	Initial State
IISTXD	[31:0]	TX FIFO write data. Note that the left/right channel data is allocated as the following bit fields. R[31:16], L[15:0] when 16-bit BLC R[23:16], L[7:0] when 8-bit BLC	W	32'b0

3.7.1.6 IIS-BUS Interface Special Registers (IISRXD)

- IISRXD, R, Address = 0xE210_0014
- IISRXD, R, Address = 0xE2A0_0014

IISRXD	Bit	Description	R/W	Initial State
IISRXD	[31:0]	RX FIFO read data. Note that the left/right channel data is allocated as the following bit fields. R[31:16], L[15:0] when 16-bit BLC R[23:16], L[7:0] when 8-bit BLC	R	32'b0

4 AC97 CONTROLLER

This chapter describes the functions and usage of AC97 Controller in S5PV210 RISC microprocessor.

4.1 OVERVIEW OF AC97 CONTROLLER

The AC97 Controller Unit in the S5PV210 supports the features of AC97 revision 2.0. AC97 Controller uses audio controller link (AC-link) to communicate with AC97 Codec. Controller sends the stereo PCM data to Codec. The external digital-to-analog converter (DAC) in the Codec converts the audio sample to an analog audio waveform. Controller receives the stereo PCM data and the mono Mic data from Codec then store in memories. This chapter describes the programming model for the AC97 Controller Unit. The prerequisite in this chapter requires an understanding of the AC97 revision 2.0 specifications.

4.2 KEY FEATURES OF AC97 CONTROLLER

The AC97 Controller includes the following features:

- Independent channels for stereo PCM In, stereo PCM Out, mono MIC In.
- DMA-based operation and interrupt based operation.
- All of the channels support only 16-bit samples.
- Variable sampling rate AC97 Codec interface (48 kHz and below)
- 16-bit, 16 entry FIFOs per channel
- Only primary Codec support

4.3 AC97 CONTROLLER OPERATION

This section explains the AC97 Controller operation, namely, AC-Link, Power-down sequence and Wake-up sequence.

4.3.1 BLOCK DIAGRAM OF AC97 CONTROLLER

Figure 4-1 shows the functional block diagram of S5PV210 AC97 Controller. The AC97 signals from the AC-link, which is a point-to-point synchronous serial inter-connecting that supports full-duplex data transfers. All digital audio streams and command/status information are communicated via AC-link.

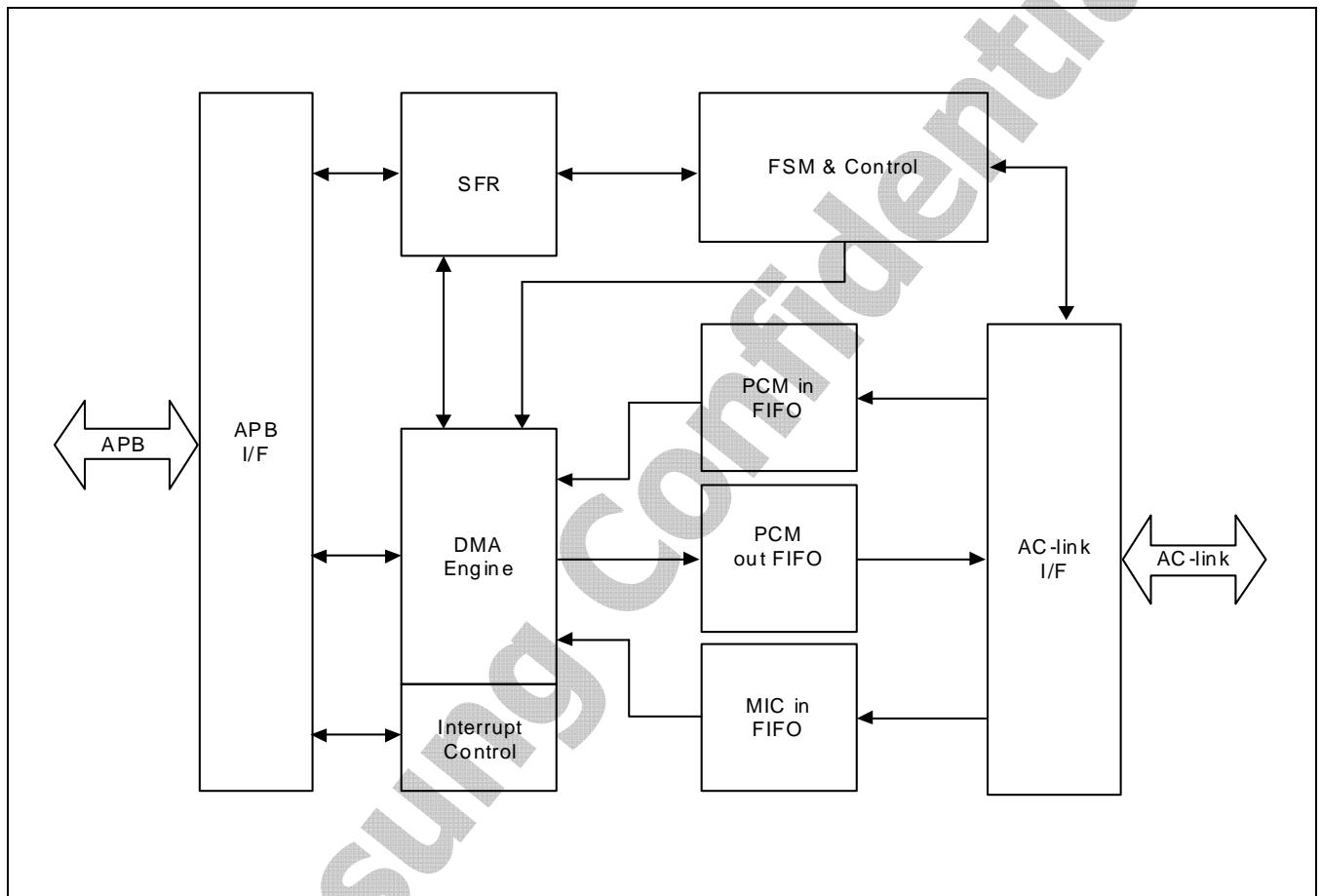


Figure 4-1 AC97 Block Diagram

4.3.2 INTERNAL DATA PATH

[Figure 4-2](#) shows the internal data path of S5PV210 AC97 Controller. It includes stereo Pulse Code Modulated (PCM) In, Stereo PCM Out and mono Mic-in buffers, which consist of 16-bit and 16 entries buffer. It also has 20-bit I/O shift register via AC-link.

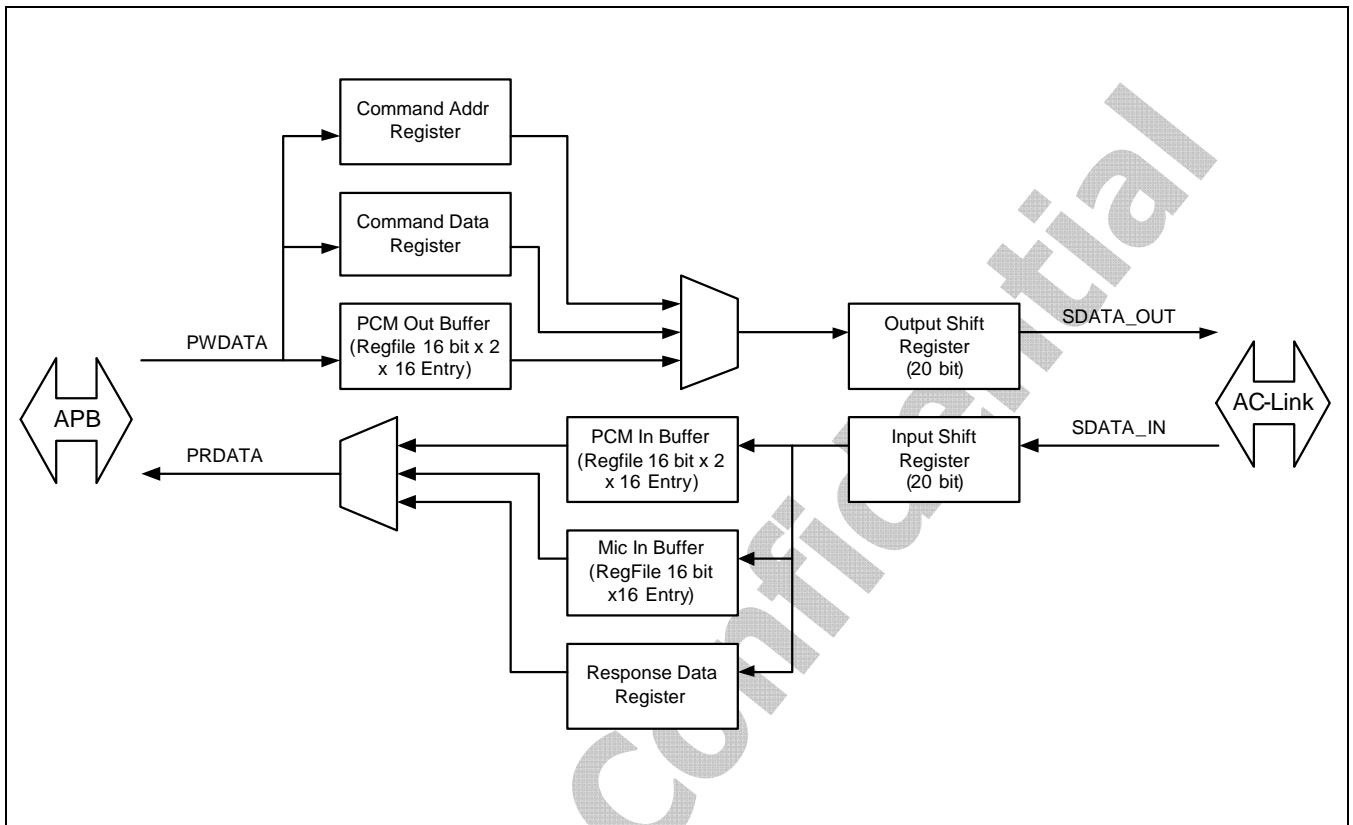


Figure 4-2 Internal Data Path

4.3.3 OPERATION FLOW CHART

When you initialize the AC97 controller, you must assert system reset or cold reset, because the previous state of the external AC97 audio-codec is not known. This assures that GPIO is already ready. Then you enable the codec ready interrupt. You can check codec ready interrupt by polling or interrupt. When interrupt occurs, you must de-assert codec ready interrupt. Use DMA or PIO (directly to write data to register) to transmit data from memory to register or from register to memory. If internal FIFOs (TX FIFO or RX FIFO) are not empty, then let data be transmitted.

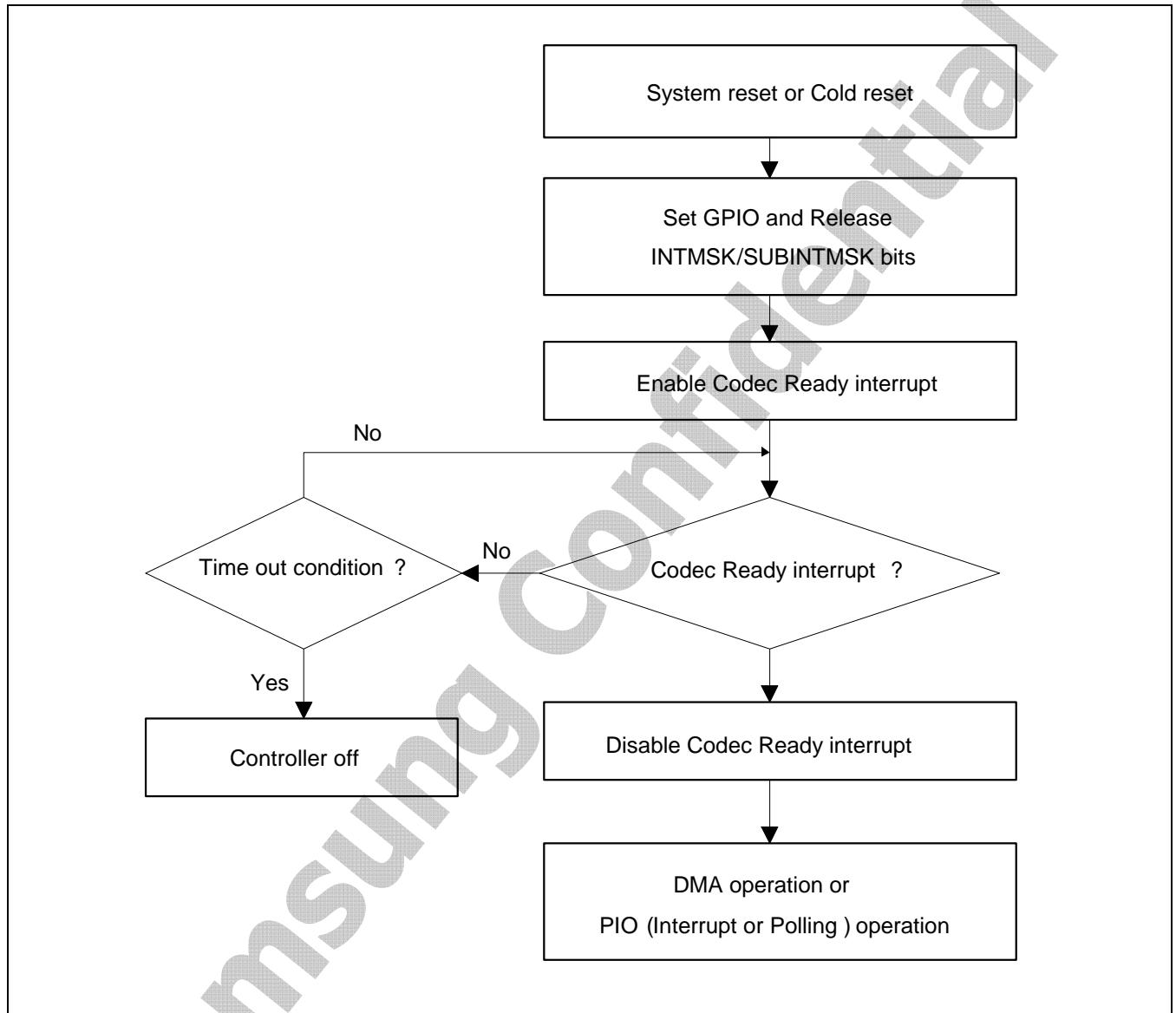


Figure 4-3 AC97 Operation Flow Chart

4.3.4 AC-LINK DIGITAL INTERFACE PROTOCOL

Each AC97 Codec incorporates a five-pin digital serial interface that links it to the S5PV210 AC97 Controller. AC-link is a full-duplex, fixed-clock and PCM digital stream. It employs a time division multiplexed (TDM) scheme to handle control register accesses and multiple input and output audio streams. The AC-link architecture divides each audio frame into 12 outgoing and 12 incoming data streams. Each stream has 20-bit sample resolution and requires a DAC and an analog-to-digital converter (ADC) with a minimum 16-bit resolution.

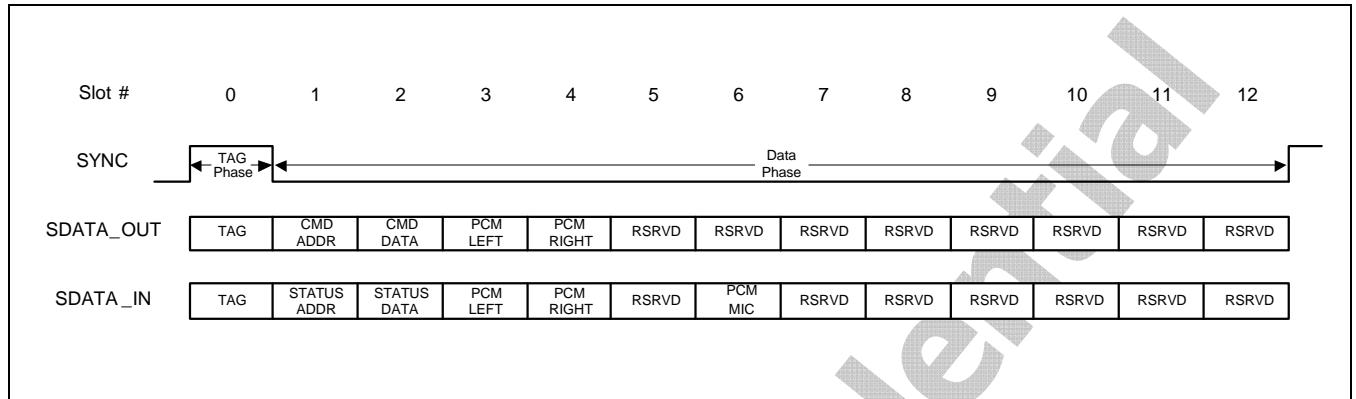


Figure 4-4 Bi-directional AC-link Frame with Slot Assignments

[Figure 4-4](#) shows the slot definitions supported by S5PV210 AC97 Controller. The S5PV210 AC97 Controller provides synchronization for all data transaction on the AC-link.

A data transaction is made up of 256 bits of information broken into groups of 13 time slots and is called a frame. Time slot 0 is called the Tag Phase and it is 16 bits long. The other 12 time slots are called the Data Phase. The Tag Phase contains one bit that identifies a valid frame and 12 bits that identify the time slots in the Data Phase that contain valid data. Each time slot in the Data Phase is 20 bits long. A frame begins when SYNC goes high. The amount of time that SYNC is high corresponds to the Tag Phase. AC97 frames occur at fixed 48 kHz intervals and are synchronous to the 12.288 MHz bit rate clock, BITCLK.

The controller and the Codec use the SYNC and BITCLK to determine when to send transmit data and when to sample received data. A transmitter transitions the serial data stream on each rising edge of BITCLK and a receiver samples the serial data stream on falling edges of BITCLK. The transmitter must tag the valid slots in its serial data stream. The valid slots are tagged in slot 0. Serial data on the AC-link is ordered most significant bit (MSB) to least significant bit (LSB). The Tag Phase's first bit is bit 15 and the first bit of each slot in Data Phase is bit 19. The last bit in any slot is bit 0.

4.3.4.1 AC-link Output Frame (SDATA_OUT)

Slot 0: Tag Phase

In slot 0, the first bit is a bit (SDATA_OUT, bit 15) which represents the validity of the entire frame. If bit 15 is 1, the current frame contains at least a valid time slot. The next 12-bit positions correspond each 12 time slot contains valid data. Bits 0 and 1 of slot 0 are used as CODEC IO bits for I/O reads and writes to the CODEC registers as described in the next section. In this way, data streams of differing sample rate can be transmitted across AC-link at its fixed 48 kHz audio frame rate.

Slot 1: Command Address Port

In slot 1, it communicates control register address and write/read command information to the AC97 controller. When software accesses the primary CODEC, the hardware configures the frame as follows:

- In slot 0, the valid bit for 1, 2 slots are set.
- In slot 1, bit 19 is set (read) or clear(write). Bits 18-12 (of slot 1) are configured to specify the index to the CODEC register. Others are filled with 0's(reserved).
- In slot 2, it configured with the data which is for writing because of output frame.

Slot 2: Command Data Port

In slot 2, this is the write data with 16-bit resolution ([19:4] is valid data)

Slot 3: PCM Playback Left channel

Slot 3 is audio output frame is the composite digital audio left stream. If a sample has a resolution that is less than 16 bits, the AC97 controller fills all training non-valid bit positions in the slot with zeroes.

Slot 4: PCM Playback Right channel

Slot 4 which is audio output frame is the composite digital audio right stream. If a sample has a resolution that is less than 16 bits, the AC97 controller fills all training non-valid bit positions in the slot with zeroes.

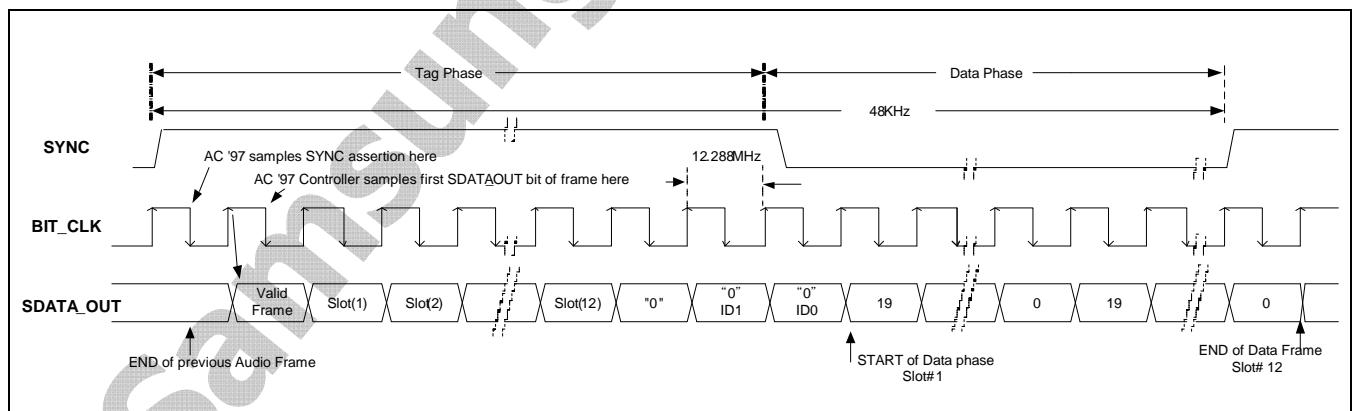


Figure 4-5 AC-link Output Frame

4.3.5 AC-LINK INPUT FRAME (SDATA_IN)

Slot 0: Tag Phase

In slot 0, the first bit (SDATA_OUT, bit 15) indicates whether the AC97 controller is in the CODEC ready state. If the CODEC Ready bit is 0, it means that the AC97 controller is not ready for normal operation. This condition is normal after the power is de-asserted on reset and the AC97 controller voltage references are settling.

Slot 1: Status Address Port/SLOTREQ bits

The status port monitors the status of the AC97 controller functions. It is not limited to mixer settings and power management. Audio input frame slot 1's stream echoes the control register index for the data to be returned in slot 2, if the controller tags slots 1 and 2 as valid during slot 0. The controller only accepts status data if the accompanying status address matches the last valid command address issued during the most recent read command. For multiple sample rate output, the CODEC examines its sample-rate control registers, its FIFOs' states, and the incoming SDATA_OUT tag bits at the beginning of each audio output frame to determine which SLOTREQ bits to set active (low). SLOTREQ bits asserted during the current audio input frame indicate which output slots require data from the controller in the next audio output frame. For fixed 48 kHz operation, the SLOTREQ bits are set active (low), and a sample is transferred in each frame. For multiple sample-rate input, the "tag" bit for each input slot indicates whether valid data is present.

Table 4-1 Input Slot 1 Bit Definitions

Bit	Description
19	Reserved (Filled with zero)
18-12	Control register index (Filled with zeroes if AC97 tags is invalid)
11	Slot 3 request: PCM Left channel
10	Slot 4 request: PCM Right channel
9	Slot 5 request: NA
8	Slot 6 request: MIC channel
7	Slot 7 request: NA
6	Slot 8 request: NA
5	Slot 9 request: NA
4	Slot 10 request: NA
3	Slot 11 request: NA
2	Slot 12 request: NA
1, 0	Reserved (Filled with zero)

Slot 2: Status Data Port

In slot 2, this is the status data with 16-bit resolution ([19:4] is valid data)

Slot 3: PCM Record Left channel

Slot 3 which is audio input frame is the left channel audio output of the AC97 Codec. If a sample has a resolution that is less than 16 bits, the AC97 Codec fills all training non-valid bit positions in the slot with zeroes.

Slot 4: PCM Record Right channel

Slot 4 which is audio input frame is the right channel audio output of the AC97 Codec. If a sample has a resolution that is less than 16 bits, the AC97 Codec fills all training non-valid bit positions in the slot with zeroes.

Slot 6: Microphone Record Data

The AC97 Controller supports 16-bit resolution for the MIC-in channel.

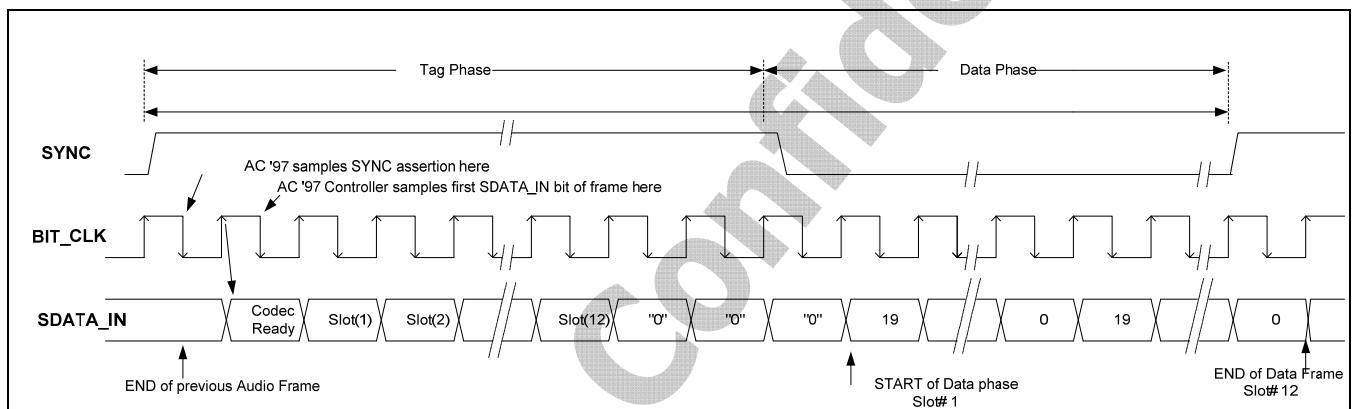


Figure 4-6 AC-link Input Frame

4.3.6 AC97 POWER-DOWN

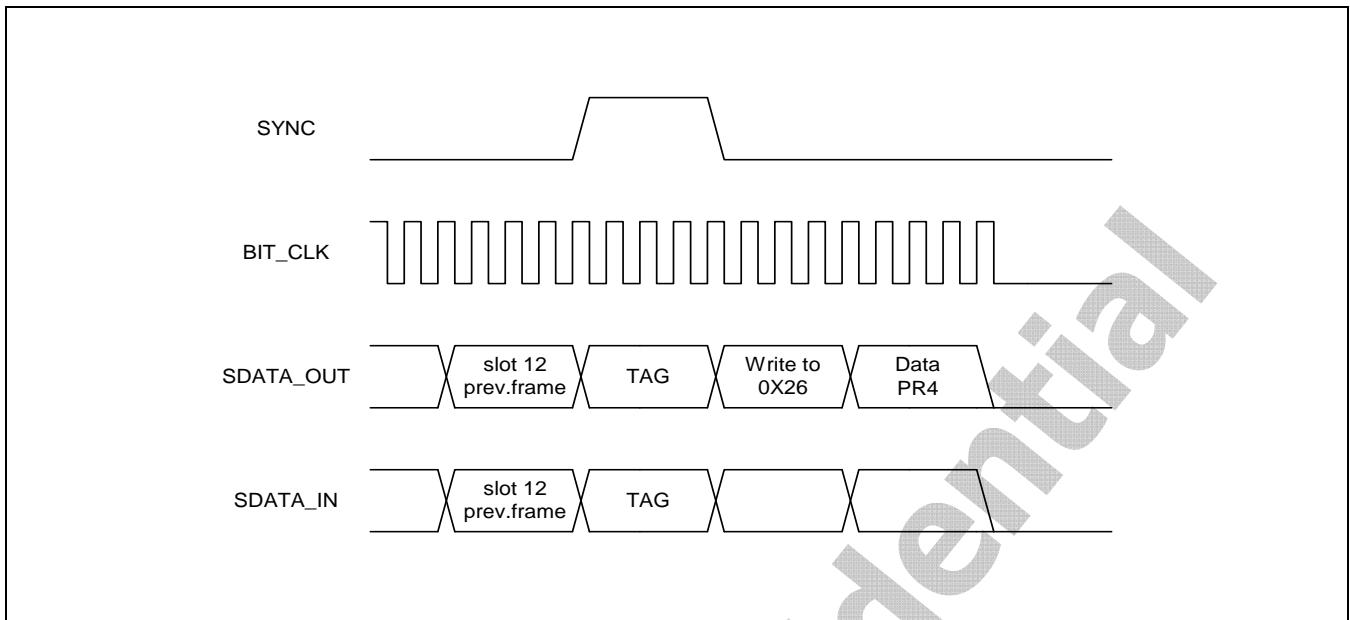


Figure 4-7 AC97 Power-down Timing

4.3.6.1 Powering Down the AC-link

The AC-link signals enter a low power mode when the AC97 Codec Power-down register (0x26) bit PR4 is set to 1 (by writing 0x1000). Then the Primary Codec drives both BITCLK and SDATA_IN to a logic low voltage level. The sequence follows the timing diagram as shown in [Figure 4-7](#).

The AC97 Controller transmits the write to Power-down register (0x26) via AC-link. Set up the AC97 Controller so that it does not transmit data to slots 3-12 when it writes to the Power-down register bit PR4 (data 0x1000), and it does not require the Codec to process other data when it receives a power down request. When the Codec processes the request it immediately transitions BITCLK and SDATA_IN to a logic low level. The AC97 Controller drives SYNC and SDATA_OUT to a logic low level after programming the AC_GLBCTRL register.

4.3.6.2 Waking up the AC-link - Wake Up Triggered by the AC97 Controller

AC-link protocol provides a cold AC97 reset and a warm AC97 reset. The current power-down state ultimately dictates which AC97 reset is used. Registers must stay in the same state during all power-down modes unless a cold AC97 reset is performed. In a cold AC97 reset, the AC97 registers are initialized to their default values. After a power down, the AC-link must wait for a minimum of four audio frame times after the frame in which the power down occurred before it can be reactivated by reasserting the SYNC signal. When AC-link powers up, Codec ready bit (input slot 0, bit 15) indicates that AC-link is ready for operation.

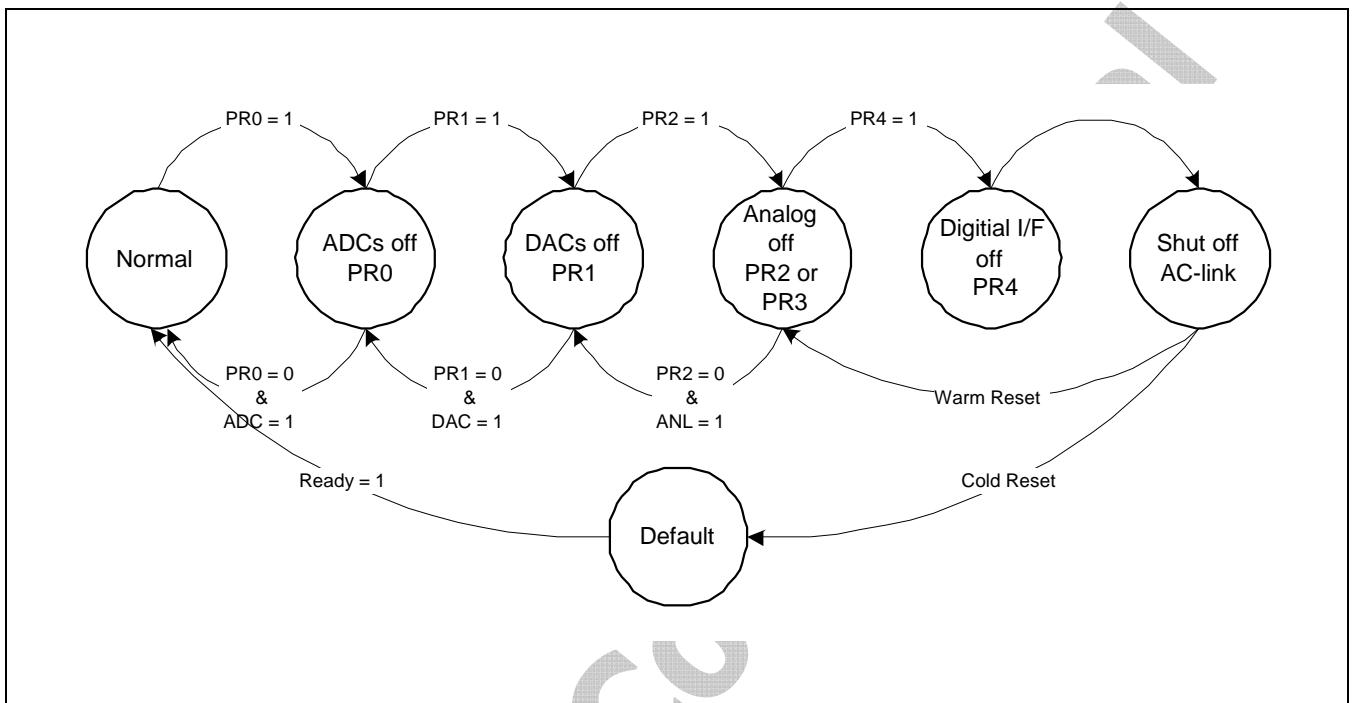


Figure 4-8 AC97 Power down/Power up Flow

4.3.6.3 Cold AC97 Reset

A cold reset is generated when the nRESET pin is asserted through the AC_GLBCTRL. Asserting and deasserting nRESET activates BITCLK and SDATA_OUT. All AC97 control registers are initialized to their default power on reset values. nRESET is an asynchronous AC97 input.

4.3.6.4 Warm AC97 Reset

A Warm AC97 reset reactivates the AC-link without altering the current AC97 register values. A warm reset is generated when BITCLK is absent and SYNC is driven high. In normal audio frames, SYNC is a synchronous AC97 input. When BITCLK is absent, SYNC is treated as an asynchronous input used to generate a warm reset to AC97. The AC97 Controller must not activate BITCLK until it samples SYNC low again. This prevents a new audio frame being falsely detected.

4.3.6.5 AC97 State Diagram

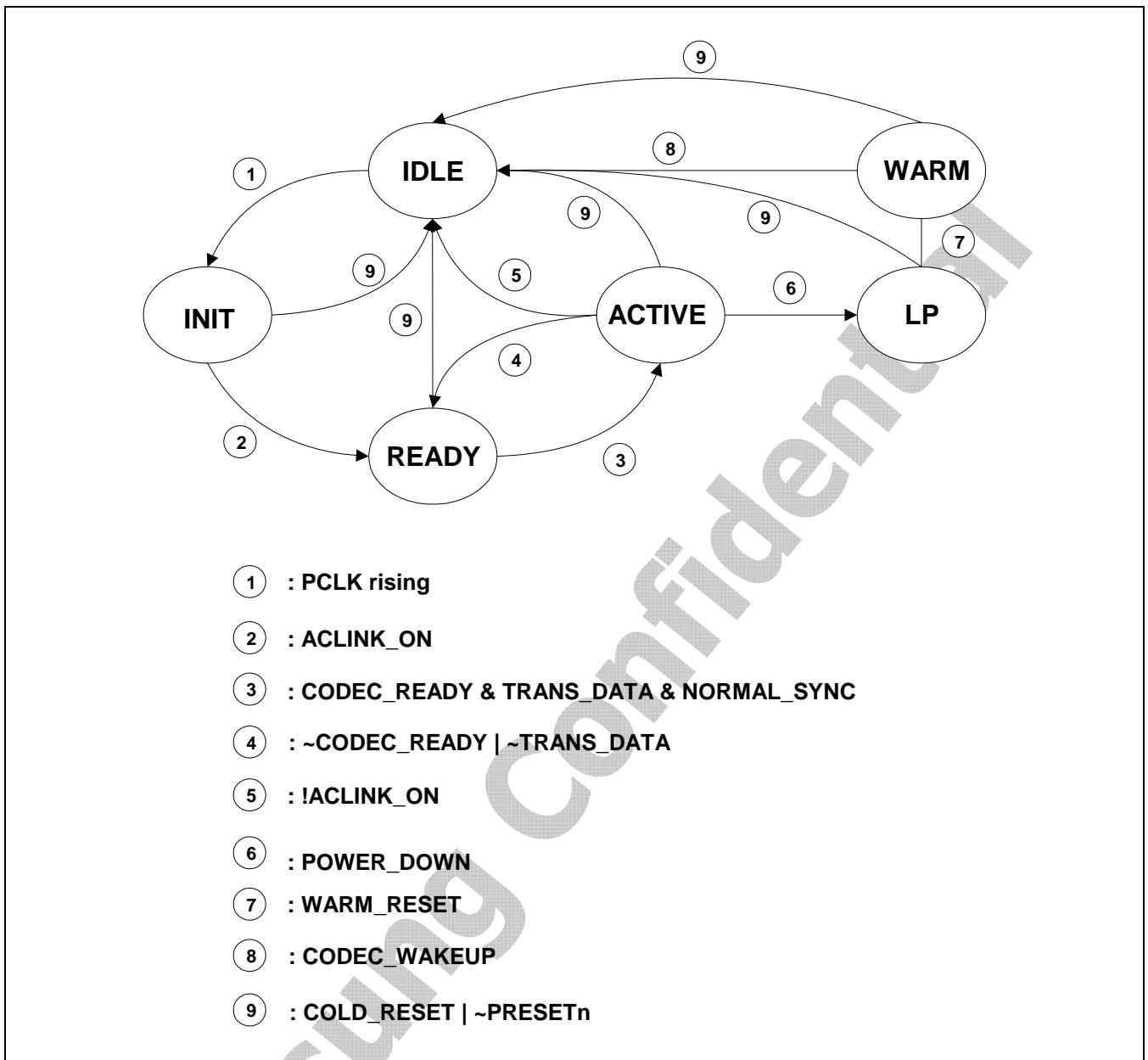


Figure 4-9 AC97 State Diagram

[Figure 4-9](#) shows the state diagram of AC97 controller. It is useful to check AC97 controller state machine. State machine shown in above figure is synchronized by peripheral clock (PCLK). Use AC_GLBSTAT register to monitor state.

4.4 I/O DESCRIPTION

AC97 external pads are shared with I2S. In order to use these pads for AC97, GPIO must be set before the AC97 starts. For mode information, refer to the GPIO chapter of this manual for exact GPIO setting

Signal	I/O	Description	Pad	Type
AC_nRESET	O	Active-low CODEC reset.	Xi2s1CDCLK	muxed
AC_BIT_CLK	I	12.288MHz bit-rate clock.	Xi2s1SCLK	muxed
AC_SYNC	O	48 kHz frame indicator and synchronizer	Xi2s1LRCK	muxed
AC_SDATA_OUT	O	Serial audio output data.	Xi2s1SDO	muxed
AC_SDATA_IN	I	Serial audio input data.	Xi2s1SDI	muxed

4.5 REGISTER DESCRIPTION

4.5.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
AC_GLBCTRL	0xE220_0000	R/W	Specifies the AC97 Global Control Register	0x00000000
AC_GLBSTAT	0xE220_0004	R	Specifies the AC97 Global Status Register	0x00000001
AC_CODEC_CMD	0xE220_0008	R/W	Specifies the AC97 Codec Command Register	0x00000000
AC_CODEC_STAT	0xE220_000C	R	Specifies the AC97 Codec Status Register	0x00000000
AC_PCMADDR	0xE220_0010	R	Specifies the AC97 PCM Out/In Channel FIFO Address Register	0x00000000
AC_MICADDR	0xE220_0014	R	Specifies the AC97 MIC In Channel FIFO Address Register	0x00000000
AC_PCMDATA	0xE220_0018	R/W	Specifies the AC97 PCM Out/In Channel FIFO Data Register	0x00000000
AC_MICDATA	0xE220_001C	R/W	Specifies the AC97 MIC In Channel FIFO Data Register	0x00000000

4.5.1.1 AC97 Global Control Register (AC_GLBCTRL, R/W, Address = 0xE220_0000)

This is the global register of the AC97 controller. There are interrupt control registers, DMA control registers, AC-Link control register, data transmission control register and related reset control register.

AC_GLBCTRL	Bit	Description	Initial State
-	[31]	Reserved.	0
Codec ready interrupt clear	[30]	1 = Interrupt clear(write only)	0
PCM out channel underrun interrupt clear	[29]	1 = Interrupt clear(write only)	0
PCM in channel overrun interrupt clear	[28]	1 = Interrupt clear(write only)	0
MIC in channel overrun interrupt clear	[27]	1 = Interrupt clear(write only)	0
PCM out channel threshold interrupt clear	[26]	1 = Interrupt clear(write only)	0
PCM in channel threshold interrupt clear	[25]	1 = Interrupt clear(write only)	0
MIC in channel threshold interrupt clear	[24]	1 = Interrupt clear(write only)	0
-	[23]	Reserved	0
Codec ready interrupt enable	[22]	0 = Disables 1 = Enables	0
PCM out channel underrun interrupt enable	[21]	0 = Disables 1 = Enables (FIFO is empty)	0
PCM in channel overrun interrupt enable	[20]	0 = Disables 1 = Enables (FIFO is full)	0
Mic in channel overrun interrupt enable	[19]	0 = Disables 1 = Enables (FIFO is full)	0
PCM out channel threshold interrupt enable	[18]	0 = Disables 1 = Enables (FIFO is half empty)	0
PCM in channel threshold interrupt enable	[17]	0 = Disables 1 = Enables (FIFO is half full)	0
MIC in channel threshold interrupt enable	[16]	0 = Disables 1 = Enables (FIFO is half full)	0
-	[15:14]	Reserved.	00
PCM out channel transfer mode	[13:12]	00 = Off 01 = PIO 10 = DMA 11 = Reserved	00
PCM in channel transfer mode	[11:10]	00 = Off 01 = PIO 10 = DMA 11 = Reserved	00
MIC in channel transfer mode	[9:8]	00 = Off	00

AC_GLBCTRL	Bit	Description	Initial State
		01 = PIO 10 = DMA 11 = Reserved	
-	[7:4]	Reserved.	0000
Transfer data enable using AC-link	[3]	0 = Disables 1 = Enables	0
AC-Link on	[2]	0 = Off 1 = SYNC signal transfer to Codec	0
Warm reset	[1]	0 = Normal 1 = Wake up codec from power down	0
Cold reset	[0]	0 = Normal 1 = Reset Codec and Controller logic Notes: 1. During Cold reset, writing to any AC97 Registers is not affected. 2. When recovering from Cold reset, writing to any AC97 Registers is not affected. Example: For consecutive Cold reset and Warm reset, first set AC_GLBCTRL=0x1 then set AC_GLBCTRL=0x0. After recovering from cold reset set AC_GLBCTRL=0x2 then AC_GLBCTRL=0x0.	0

4.5.1.2 AC97 Global Status Register (AC_GLBSTAT, R, Address = 0xE220_0004)

This is the status register. When the interrupt occurs, you can check the source of interrupt.

AC_GLBSTAT	Bit	Description	Initial State
-	[31:23]	Reserved.	0x00
Codec ready interrupt	[22]	0 = Not requested 1 = Requested	0
PCM out channel underrun interrupt	[21]	0 = Not requested 1 = Requested	0
PCM in channel overrun interrupt	[20]	0 = Not requested 1 = Requested	0
MIC in channel overrun interrupt	[19]	0 = Not requested 1 = Requested	0
PCM out channel threshold interrupt	[18]	0 = Not requested 1 = Requested	0
PCM in channel threshold interrupt	[17]	0 = Not requested 1 = Requested	0
MIC in channel threshold interrupt	[16]	0 = Not requested 1 = Requested	0
-	[15:3]	Reserved.	0x000
Controller main state	[2:0]	000 = Idle 001 = Init 010 = Ready 011 = Active 100 = LP 101 = Warm	001

4.5.1.3 AC97 Codec Command Register (AC_CODEC_CMD, R/W, Address = 0xE220_0008)

When you control writing or reading, you must set the Read enable bit. If you want to write data to the AC97 Codec, you set the index (or address) of the AC97 Codec and data.

AC_CODEC_CMD	Bit	Description	Initial State
-	[31:24]	Reserved	0x00
Read enable	[23]	0 = Command write ^(note) 1 = Status read	0
Address	[22:16]	Codec command address	0x00
Data	[15:0]	Codec command data	0x0000

NOTE: When the commands are written on the AC_CODEC_CMD register, it is recommended to have more than 1 / 48 kHz delay time between the command and the next command.

4.5.1.4 AC97 Codec Status Register (AC_CODEC_STAT, R, Address = 0xE220_000C)

If the Read enable bit is 1 and Codec command address is valid, Codec status data is also valid.

AC_CODEC_STAT	Bit	Description	Initial State
-	[31:23]	Reserved.	0x00
Address	[22:16]	Codec status address	0x00
Data	[15:0]	Codec status data	0x0000

NOTE: Steps to read data from AC97 codec register via the AC_CODEC_STAT register:

1. Write command address and data on the AC_CODEC_CMD register with Bit[23] =1.
2. Set a proper delay time. It depends on Codec type.
3. Read command address and data from AC_CODEC_STAT register.

4.5.1.5 AC97 PCM OUT/IN Channel FIFO Address Register (AC_PCMADDR, R, Address = 0xE220_0010)

To index the internal PCM FIFOs address.

AC_PCMADDR	Bit	Description	Initial State
-	[31:28]	Reserved.	0000
Out read address	[27:24]	PCM out channel FIFO read address	0000
-	[23:20]	Reserved.	0000
In read address	[19:16]	PCM in channel FIFO read address	0000
-	[15:12]	Reserved.	0000
Out write address	[11:8]	PCM out channel FIFO write address	0000
-	[7:4]	Reserved.	0000
In write address	[3:0]	PCM in channel FIFO write address	0000

4.5.1.6 AC97 MIC IN Channel FIFO Address Register (AC_MICADDR, R, Address = 0xE220_0014)

To index the internal MIC-in FIFO address.

AC_MICADDR	Bit	Description	Initial State
-	[31:20]	Reserved.	0000
Read address	[19:16]	MIC in channel FIFO read address	0000
-	[15:4]	Reserved.	0x000
Write address	[3:0]	MIC in channel FIFO write address	0000

4.5.1.7 AC97 PCM OUT/IN Channel FIFO Data Register (AC_PCMDATA, R/W, Address = 0xE220_0018)

This is PCM out/in channel FIFO data register.

AC_PCMDATA	Bit	Description	Initial State
Right data	[31:16]	PCM out/in right channel FIFO data Read = PCM in right channel Write = PCM out right channel	0x0000
Left data	[15:0]	PCM out/in left channel FIFO data Read = PCM in left channel Write = PCM out left channel	0x0000

4.5.1.8 AC97 MIC IN Channel FIFO Data Register (AC_MICDATA, R/W, Address = 0xE220_001C)

This is MIC-in channel FIFO data register.

AC_MICDATA	Bit	Description	Initial State
-	[31:16]	Reserved	0x0000
Mono data	[15:0]	MIC in mono channel FIFO data	0x0000

5

PCM AUDIO INTERFACE

5.1 OVERVIEW OF PCM AUDIO INTERFACE

The PCM Audio Interface module provides PCM bi-directional serial interface to an external Codec.

5.2 KEY FEATURES OF PCM AUDIO INTERFACE

The PCM Audio interface includes the following features:

- 16-bit PCM, 3 ports audio interface
- Supports only master mode.
- All PCM serial timings and strobes including the main shift clock, are based on an PCM_EXTCLK
- OSC, EPLL_FOUT or AUDIO_SCLK can be used as PCM_EXTCLK source clock
- Optional timing based on the internal APB PCLK
- Input (16-bit x 32depth) and output (16-bit x 32depth) FIFOs to buffer data
- Optional DMA interface for TX and/or RX

5.3 PCM AUDIO INTERFACE

The PCM Audio Interface provides a serial interface to an external Codec. The PCM module receives an input PCMCODEC_CLK to generate the serial shift timing. The PCM interface outputs a serial data out, a serial shift clock, and a sync signal. Data is received from the external Codec over a serial input line. The serial data in, serial data out, and sync signal are all synchronized to the serial shift clock.

The serial shift clock, PCMSCLK, is generated from a programmable divide of the input PCMCODEC_CLK. The sync signal, PCMSYNC, is generated based upon a programmable number of serial clocks and is one serial clock wide.

The PCM data words are 16-bit wide, serially shifted out 1-bit per PCMSCLK. Only one 16-bit word is shifted out for each PCMSYNC. The PCMSCLK will continue to toggle even after all 16-bit have been shifted out. The PCMSOUT data is not valid after the 16-bit word is complete. The next PCMSYNC will signal the start of the next PCM data word.

The TX FIFO provides the 16-bit data word to be serially shifted out. This data is serially shifted out MSB first, one bit per PCMSCLK. The PCM serial output data, PCMSOUT, is clocked out using the rising edge of the PCMSCLK. The MSB bit position relative to the PCMSYNC is programmable to either match the PCMSYNC or one PCMCLK later. After all 16-bit have been shifted out, to indicate the end of the transfer you can generate an interrupt.

At the same time data is being shifted out, the PCMSIN input is used to serially shift data from the external codec. The data is received MSB first and is clocked in on the falling edge of PCMSCLK. The position of the first bit is programmable to be coincident with the PCMSYNC or one PCMSCLK later.

The first 16-bit are serially shifted into the PCM_DATAIN register which is then loaded into the RX FIFO. Subsequent bits are ignored until the next PCMSYNC.

Various Interrupts are available to indicate the status of the RX and TX FIFO. Each FIFO has a programmable flag to indicate when the CPU needs to service the FIFO. In the RX FIFO, there is an interrupt, which will be raised when the FIFO exceeds a certain programmable ALMOST_FULL depth. Similarly there is a programmable ALMOST_EMPTY interrupt for the TX FIFO.

5.4 PCM TIMING

[Figure 5-1](#) shows the timing relationship for the PCM transfers.

Note in all cases, the PCM shift timing is derived by dividing the input clock, PCMCODEC_CLK. While the timing is based upon the PCMCODEC_CLK, there is no attempt to realign the rising edge of the output PCMSCLK with the original PCMCODEC_CLK input clock. These edges will be skewed by internal delay through the pads as well as the divider logic. This does not represent a problem because the actual shift clock, PCMSCLK, is synchronized with the data. Furthermore, even if the PCMSCLK output is not used, the skew will be significantly less than the period of the PCMCODEC_CLK and does not represent a problem since most PCM interfaces capture data on the falling edge of the clock.

[Figure 5-1](#) shows a PCM transfer with the MSB configured to be coincident with the PCMSYNC. This MSB positioning corresponds to setting the TX_MSB_POS and RX_MSB_POS bits in PCMCTL register to be 0.

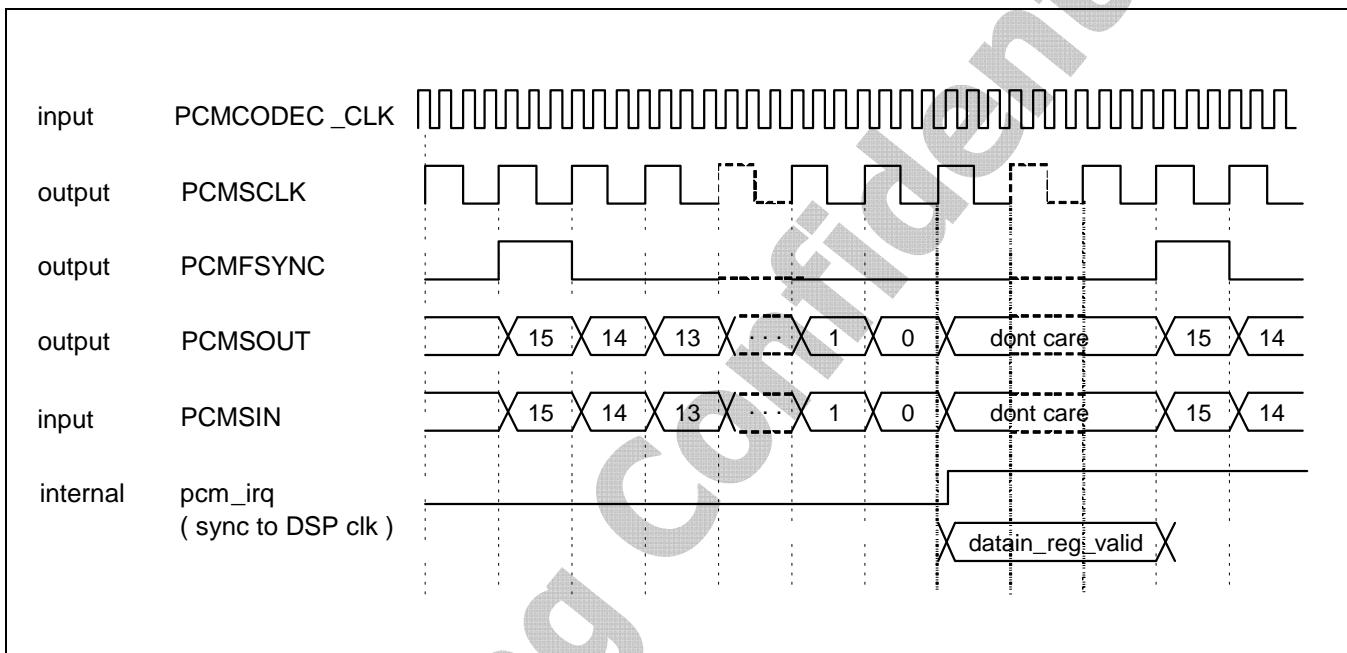


Figure 5-1 PCM timing, POS_MSB_WR/RD = 0

[Figure 5-2](#) shows a PCM transfer with the MSB configured one shift clock after the PCMSYNC. This MSB positioning corresponds to setting the TX_MSB_POS and RX_MSB_POS bits in PCMCTL register to be 1.

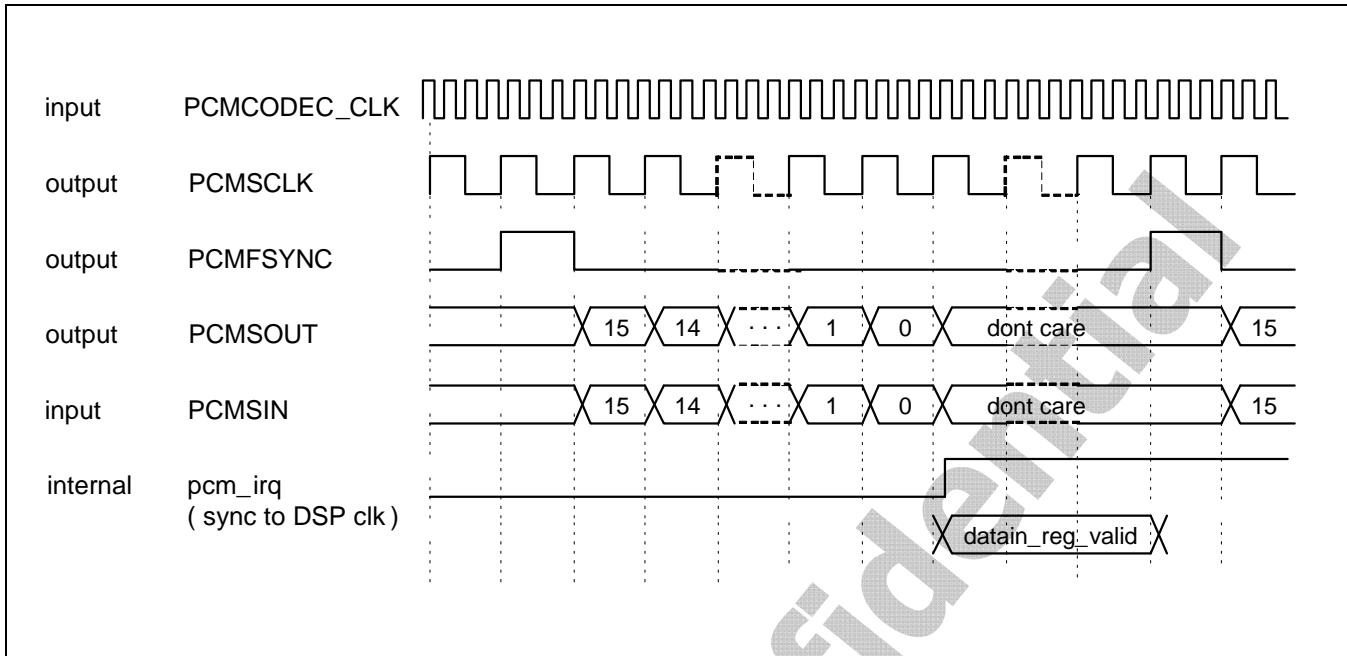


Figure 5-2 PCM timing, POS_MSB_WR/RD = 1

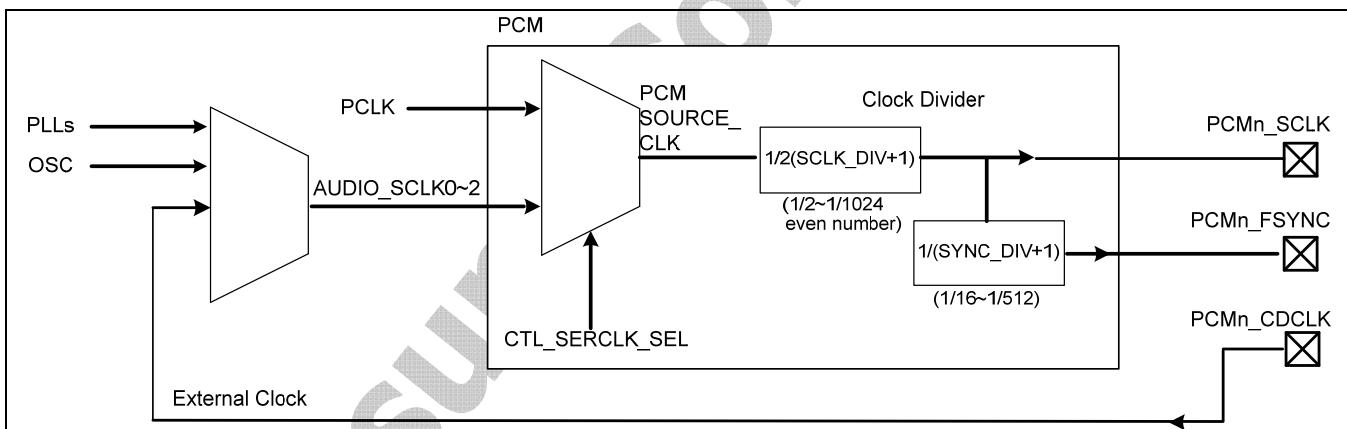


Figure 5-3 Input Clock Diagram for PCM

S5PV210 PCM can select clock either PCLK or External Clock. Refer [Figure 5-3](#). To enable clock gating, please refer to the SYSCON part (SCLKCON, PCLKCON)

5.5 I/O DESCRIPTION

Each PCM external pads are shared with I2S and PCM. In order to use these pads for I2S, GPIO must be set before the PCM started. For mode information, refer to the GPIO chapter of this manual for proper GPIO setting

PAD Name	I/O	Description	Pad	Type
Xi2s0CDCLK, Xi2s1CDCLK, Xpcm0EXTCLK	I/O	PCM Codec clock input/output		dedicated
Xi2s0SCLK, Xi2s1SCLK, Xpcm0SCLK	I/O	PCM Bit clock input/output		dedicated
Xi2s0LRCK, Xi2s1LRCK, Xpcm0FSYNC	I/O	PCM FSYNC channel clock input/output		dedicated
Xi2s0SDI, Xi2s1SDI, Xpcm0SIN	I	PCM serial data input		dedicated
Xi2s0SDO, Xi2s1SDO, Xpcm0SOUT	O	PCM serial data out		dedicated

5.6 REGISTER DESCRIPTION

5.6.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
PCM0				
PCM_CTL	0xE230_0000	R/W	Specifies the PCM Main Control	0x00000000
PCM_CLKCTL	0xE230_0004	R/W	Specifies the PCM Clock and Shift control	0x00000000
PCM_TXFIFO	0xE230_0008	R/W	Specifies the PCM TXFIFO write port	0x00010000
PCM_RXFIFO	0xE230_000C	R/W	Specifies the PCM RXFIFO read port	0x00010000
PCM_IRQ_CTL	0xE230_0010	R/W	Specifies the PCM Interrupt Control	0x00000000
PCM_IRQ_STAT	0xE230_0014	R	Specifies the PCM Interrupt Status	0x00000000
PCM_FIFO_STAT	0xE230_0018	R	Specifies the PCM FIFO Status	0x00000000
PCM_CLRINT	0xE230_0020	W	Specifies the PCM Interrupt Clear	-
PCM1				
PCM_CTL	0xE120_0000	R/W	Specifies the PCM Main Control	0x00000000
PCM_CLKCTL	0xE120_0004	R/W	Specifies the PCM Clock and Shift control	0x00000000
PCM_TXFIFO	0xE120_0008	R/W	Specifies the PCM TXFIFO write port	0x00010000
PCM_RXFIFO	0xE120_000C	R/W	Specifies the PCM RXFIFO read port	0x00010000
PCM_IRQ_CTL	0xE120_0010	R/W	Specifies the PCM Interrupt Control	0x00000000
PCM_IRQ_STAT	0xE120_0014	R	Specifies the PCM Interrupt Status	0x00000000
PCM_FIFO_STAT	0xE120_0018	R	Specifies the PCM FIFO Status	0x00000000
PCM_CLRINT	0xE120_0020	W	Specifies the PCM Interrupt Clear	-
PCM2				
PCM_CTL	0xE2B0_0000	R/W	Specifies the PCM Main Control	0x00000000
PCM_CLKCTL	0xE2B0_0004	R/W	Specifies the PCM Clock and Shift control	0x00000000
PCM_TXFIFO	0xE2B0_0008	R/W	Specifies the PCM TXFIFO write port	0x00010000
PCM_RXFIFO	0xE2B0_000C	R/W	Specifies the PCM RXFIFO read port	0x00010000
PCM_IRQ_CTL	0xE2B0_0010	R/W	Specifies the PCM Interrupt Control	0x00000000
PCM_IRQ_STAT	0xE2B0_0014	R	Specifies the PCM Interrupt Status	0x00000000
PCM_FIFO_STAT	0xE2B0_0018	R	Specifies the PCM FIFO Status	0x00000000
PCM_CLRINT	0xE2B0_0020	W	Specifies the PCM Interrupt Clear	-

5.6.1.1 PCM Control Register (PCM_CTL)

The PCM_CTL register is used to control the various aspects of the PCM module. It also provides a status bit for polling control instead of interrupt based control.

- PCM_CTL, R/W, Address = 0xE230_0000
- PCM_CTL, R/W, Address = 0xE120_0000
- PCM_CTL, R/W, Address = 0xE2B0_0000

The bit definitions for the PCM_CTL Control Register are described below:

PCM_CTL	Bit	Description	Initial State
Reserved	[31:19]	Reserved	
TXFIFO_DIPSTICK	[18:13]	<p>Determines when the ALMOST_FULL, ALMOST_EMPTY flags go active for the TXFIFO</p> <p>ALMOST_EMPTY: FIFO_depth < FIFO_dipstick ALMOST_FULL: FIFO_depth > (32 – FIFO_dipstick)</p> <p>Note: if FIFO_dipstick == 0 ALMOST_EMPTY, ALMOST_FULL are invalid</p> <p>Note: for DMA loading of TX FIFO TXFIFO_dipstick >= 2</p> <p>This is required since the PCM_TXDMA uses ALMOST_FULL as the DMA request (keep requesting data until the FIFO is almost full) In some circumstances, the DMA write one more word after the DMA_req fall to low. Thus the ALMOST_FULL flag must go active with at least space for one extra word in the FIFO</p>	0
RXFIFO_DIPSTICK	[12:7]	<p>Determines when the ALMOST_FULL, ALMOST_EMPTY flags go active for the RXFIFO</p> <p>ALMOST_EMPTY: FIFO_depth < FIFO_dipstick ALMOST_FULL: FIFO_depth > (32 – FIFO_dipstick)</p> <p>Note: if FIFO_dipstick == 0 ALMOST_EMPTY, ALMOST_FULL are invalid</p> <p>Note: for DMA, RXFIFO_DIPSTICK is a don't care DMA unloading of RX FIFO uses the RX_FIFO_EMPTY flag as the DMA request</p> <p>Note: non-DMA IRQ/polling RXFIFO_DIPSTICK should be 0x20 This will have the effect of RX_FIFO_ALMOST_FULL acting as an RX_FIFO_not_empty flag.</p>	0
PCM_TX_DMA_EN	[6]	Enables the DMA interface for the TXFIFO DMA_TX request will occur whenever the TXFIFO is not almost full	0
PCM_RX_DMA_EN	[5]	Enables the DMA interface for the RXFIFO DMA_RX request will occur whenever the RXFIFO is not empty.	0

PCM_CTL	Bit	Description	Initial State
TX_MSB_POS	[4]	Controls the position of the MSB bit in the serial output stream relative to the PCMSYNC signal 0 = MSB sent during the same clock that PCMSYNC is high 1 = MSB sent on the next PCMSCLK cycle after PCMSYNC is high	0
RX_MSB_POS	[3]	Controls the position of the MSB bit in the serial input stream relative to the PCMSYNC signal 0 = MSB is captured on the falling edge of PCMSCLK during the same cycle that PCMSYNC is high 1 = MSB is captured on the falling edge of PCMSCLK during the cycle after the PCMSYNC is high	0
PCM_TXFIFO_EN	[2]	Enables the TXFIFO When the enable is LOW the internal FIFOs will clear and reinitialize	0
PCM_RXFIFO_EN	[1]	Enables the RXFIFO When the enable is LOW the internal FIFOs will clear and reinitialize	0
PCM_PCM_ENABLE	[0]	PCM enable signal. Enables the serial shift state machines. The enable must be set HIGH for the PCM to operate. When the enable is LOW, the PCM outputs will not toggle (PCMSCLK, PCMSYNC, and PCMSOUT). Additionally when the enable is LOW, the internal divider-counters are held in reset.	0

5.6.1.2 PCM CLK Control Register (PCM_CLKCTL)

- PCM_CLKCTL, R/W, Address = 0xE230_0004
- PCM_CLKCTL, R/W, Address = 0xE120_0004
- PCM_CLKCTL, R/W, Address = 0xE2B0_0004

The bit definitions for the PCM_CTL Control Register are described below:

PCM_CLKCTL	Bit	Description	Initial State
Reserved	[31:20]	Reserved	0
CTL_SERCLK_EN	[19]	Enables the serial clock division logic. Must be HIGH for the PCM to operate	0
CTL_SERCLK_SEL	[18]	Selects the source of the serial clock 0 = SCLK_AUDIO0,1,2(PCM0, PCM1, PCM2) 1 = PCLK	0
SCLK_DIV	[17:9]	Controls the divider used to create the PCMSCLK based on the PCMCODEC_CLK Final clock will be source_clk / 2*(sclk_div+1)	000
SYNC_DIV	[8:0]	Controls the frequency of the PCMSYNC signal based on the PCMSCLK.	000

5.6.1.3 The PCM Tx FIFO Register (PCM_TXFIFO)

- PCM_TXFIFO, R/W, Address = 0xE230_0008
- PCM_TXFIFO, R/W, Address = 0xE120_0008
- PCM_TXFIFO, R/W, Address = 0xE2B0_0008

The bit definitions for the PCM_TXFIFO Register are described below:

PCM_TXFIFO	Bit	Description	Initial State
Reserved	[31:17]	Reserved	0
TXFIFO_DVALID	[16]	TXFIFO data is valid Write: Not valid Read: TXFIFO read data valid 1 = Valid 0 = Invalid (probably read an empty FIFO)	1
TXFIFO_DATA	[15:0]	TXFIFO DATA Write: TXFIFO_DATA is written into the TXFIFO Read: TXFIFO is read using the APB interface Note: reading the TXFIFO is meant to support debugging. Online the TXFIFO is read by the PCM serial shift engine, not the APB	0

5.6.1.4 PCM RX FIFO Register (PCM_RXFIFO)

- PCM_RXFIFO, R/W, Address = 0xE230_000C
- PCM_RXFIFO, R/W, Address = 0xE120_000C
- PCM_RXFIFO, R/W, Address = 0xE2B0_000C

The bit definitions for the PCM_RXFIFO Register are described below:

PCM_RXFIFO	Bit	Description	Initial State
Reserved	[31:17]	Reserved	0
RXFIFO_DVALID	[16]	RXFIFO data is valid Write: Not Valid Read: TXFIFO read data valid 1 = Valid 0 = Invalid (probably read an empty FIFO)	1
RXFIFO_DATA	[15:0]	RXFIFO DATA Write: RXFIFO_DATA is written into the RXFIFO Note: writing the RXFIFO is meant to support debugging. Online the RXFIFO is written by the PCM serial shift engine, not the APB Read: TXFIFO is read using the APB interface	0

5.6.1.5 PCM Interrupt Control Register (PCM_IRQ_CTL)

The PCM_IRQ_CTL register is used to control the various aspects of the PCM interrupts.

- PCM_IRQ_CTL, R/W, Address = 0xE230_0010
- PCM_IRQ_CTL, R/W, Address = 0xE120_0010
- PCM_IRQ_CTL, R/W, Address = 0xE2B0_0010

The bit definitions for the PCM_IRQ_CTL Control Register are described below:

PCM_IRQ_CTL	Bit	Description	Initial State
Reserved	[31:15]	Reserved	0
EN_IRQ_TO_ARM	[14]	Controls whether or not the PCM interrupt is sent to the ARM 1 = PCM IRQ is forwarded to the ARM subsystem 0 = PCM IRQ is NOT forwarded to the ARM subsystem	0
Reserved	[13]	Reserved	0
TRANSFER_DONE	[12]	Interrupt is generated every time the serial shift for a word completes 1 = IRQ source enabled 0 = IRQ source disabled	0
TXFIFO_EMPTY	[11]	Interrupt is generated whenever the TxFIFO is empty 1 = IRQ source enabled 0 = IRQ source disabled	0
TXFIFO_ALMOST_EMPTY	[10]	Interrupt is generated whenever the TxFIFO is ALMOST empty. Almost empty is defined as TX_FIFO_DEPTH < TX_FIFO_DIPSTICK 1 = IRQ source enabled 0 = IRQ source disabled	0
TXFIFO_FULL	[9]	Interrupt is generated whenever the TxFIFO is full 1 = IRQ source enabled 0 = IRQ source disabled	0
TXFIFO_ALMOST_FULL	[8]	Interrupt is generated whenever the TxFIFO is ALMOST full. Almost full is defined as TX_FIFO_DEPTH > (32 - TX_FIFO_DIPSTICK) 1 = IRQ source enabled 0 = IRQ source disabled	0
TXFIFO_ERROR_STARVE	[7]	Interrupt is generated for TxFIFO starve ERROR. This occurs whenever the TxFIFO is read when it is still empty. This is considered as an ERROR and will have unexpected results 1 = IRQ source enabled 0 = IRQ source disabled	0

PCM_IRQ_CTL	Bit	Description	Initial State
TXFIFO_ERROR_OVERFLOW	[6]	Interrupt is generated for TxFIFO overflow ERROR. This occurs whenever the TxFIFO is written when it is already full. This is considered as an ERROR and will have unexpected results 1 = IRQ source enabled 0 = IRQ source disabled	0
RXFIFO_EMPTY	[5]	Interrupt is generated whenever the RXFIFO is empty 1 = IRQ source enabled 0 = IRQ source disabled	0
RXFIFO_ALMOST_EMPTY	[4]	Interrupt is generated whenever the RXFIFO is ALMOST empty. Almost empty is defined as $RX_FIFO_DEPTH < RX_FIFO_DIPSTICK$ 1 = IRQ source enabled 0 = IRQ source disabled	0
RX_FIFO_FULL	[3]	Interrupt is generated whenever the RXFIFO is full 1 = IRQ source enabled 0 = IRQ source disabled	0
RX_FIFO_ALMOST_FULL	[2]	Interrupt is generated whenever the RXFIFO is ALMOST full. Almost full is defined as $RX_FIFO_DEPTH > (32 - RX_FIFO_DIPSTICK)$ 1 = IRQ source enabled 0 = IRQ source disabled	0
RXFIFO_ERROR_STARVE	[1]	Interrupt is generated for RXFIFO starve ERROR. This occurs whenever the RXFIFO is read when it is still empty. This is considered as an ERROR and will have unexpected results 1 = IRQ source enabled 0 = IRQ source disabled	0
RXFIFO_ERROR_OVERFLOW	[0]	Interrupt is generated for RXFIFO overflow ERROR. This occurs whenever the RXFIFO is written when it is already full. This is considered as an ERROR and will have unexpected results 1 = IRQ source enabled 0 = IRQ source disabled	0

5.6.1.6 PCM Interrupt Status Register (PCM_IRQ_STAT)

The PCM_IRQ_STAT register is used to report IRQ status.

- PCM_IRQ_STAT, R, Address = 0xE230_0014
- PCM_IRQ_STAT, R, Address = 0xE120_0014
- PCM_IRQ_STAT, R, Address = 0xE2B0_0014

The bit definitions for the PCM_IRQ_STATUS Register are described below:

PCM_IRQ_STAT	Bit	Description	Initial State
Reserved	[31:14]	Reserved	0
IRQ_PENDING	[13]	Monitoring PCM IRQ. 1 = PCM IRQ is occurred. 0 = PCM IRQ is not occurred.	0
TRANSFER_DONE	[12]	Interrupt is generated every time the serial shift for a word completes 1 = IRQ is occurred. 0 = IRQ is not occurred.	0
TXFIFO_EMPTY	[11]	Interrupt is generated whenever the TX FIFO is empty 1 = IRQ is occurred. 0 = IRQ is not occurred.	0
TXFIFO_ALMOST_EMPTY	[10]	Interrupt is generated whenever the TxFIFO is ALMOST empty. 1 = IRQ is occurred. 0 = IRQ is not occurred.	0
TXFIFO_FULL	[9]	Interrupt is generated whenever the TX FIFO is full 1 = IRQ is occurred. 0 = IRQ is not occurred.	0
TXFIFO_ALMOST_FULL	[8]	Interrupt is generated whenever the TX FIFO is ALMOST full. 1 = IRQ is occurred. 0 = IRQ is not occurred.	0
TXFIFO_ERROR_STARVE	[7]	Interrupt is generated for TX FIFO starve ERROR. This occurs whenever the TX FIFO is read when it is still empty. This is considered as an ERROR and will have unexpected results 1 = IRQ is occurred. 0 = IRQ is not occurred.	0
TXFIFO_ERROR_OVERFLOW	[6]	Interrupt is generated for TX FIFO overflow ERROR. This occurs whenever the TX FIFO is written when it is already full. This is considered as an ERROR and will have unexpected results 1 = IRQ is occurred. 0 = IRQ is not occurred.	0
RXFIFO_EMPTY	[5]	Interrupt is generated whenever the RX FIFO is empty	0

PCM_IRQ_STAT	Bit	Description	Initial State
		1 = IRQ is occurred. 0 = IRQ is not occurred.	
RXFIFO_ALMOST_EMPTY	[4]	Interrupt is generated whenever the RX FIFO is ALMOST empty. 1 = IRQ is occurred. 0 = IRQ is not occurred.	0
RX_FIFO_FULL	[3]	Interrupt is generated whenever the RX FIFO is full 1 = IRQ is occurred. 0 = IRQ is not occurred.	0
RX_FIFO_ALMOST_FULL	[2]	Interrupt is generated whenever the RX FIFO is ALMOST full. 1 = IRQ is occurred. 0 = IRQ is not occurred.	0
RXFIFO_ERROR_STARVE	[1]	Interrupt is generated for RX FIFO starve ERROR. This occurs whenever the RX FIFO is read when it is still empty. This is considered as an ERROR and will have unexpected results 1 = IRQ is occurred. 0 = IRQ is not occurred.	0
RXFIFO_ERROR_OVERFLOW	[0]	Interrupt is generated for RX FIFO overflow ERROR. This occurs whenever the RX FIFO is written when it is already full. This is considered as an ERROR and will have unexpected results 1 = IRQ is occurred. 0 = IRQ is not occurred.	0

5.6.1.7 PCM FIFO Status Register (PCM_FIFO_STAT)

The PCM_FIFO_STAT register is used to report FIFO status.

- PCM_FIFO_STAT, R, Address = 0xE230_0018
- PCM_FIFO_STAT, R, Address = 0xE120_0018
- PCM_FIFO_STAT, R, Address = 0xE2B0_0018

The bit definitions for the PCM_IRQ_STATUS Register are described below:

PCM_FIFO_STAT	Bit	Description	Initial State
Reserved	[31:20]	Reserved	0
TXFIFO_COUNT	[19:14]	TXFIFO data count (0~32).	0
TXFIFO_EMPTY	[13]	To indicate whether TXFIFO is empty.	0
TXFIFO_ALMOST_EMPTY	[12]	To indicate whether TXFIFO is almost empty.	0
TXFIFO_FULL	[11]	To indicate whether TXFIFO is full.	0
TXFIFO_ALMOST_FULL	[10]	To indicate whether TXFIFO is almost full.	0
RXFIFO_COUNT	[9:4]	RXFIFO data count (0~32).	0
RXFIFO_EMPTY	[3]	To indicate whether RXFIFO is empty.	0
RXFIFO_ALMOST_EMPTY	[2]	To indicate whether RXFIFO is almost empty.	0
RX_FIFO_FULL	[1]	To indicate whether RXFIFO is full.	0
RX_FIFO_ALMOST_FULL	[0]	To indicate whether RXFIFO is almost full.	0

5.6.1.8 PCM Interrupt Clear Register (PCM_CLRINT)

The PCM_CLRINT register is used to clear the interrupt. Interrupt service routine is responsible for clearing interrupt asserted. Writing any values on this register clears interrupts for both ARM and DSP. Reading this register is not allowed. Clearing interrupt must be prior to resolving the interrupt condition; else, another interrupt that would occur after this interrupt may be ignored.

- PCM_CLRINT, W, Address = 0xE230_0020
- PCM_CLRINT, W, Address = 0xE120_0020
- PCM_CLRINT, W, Address = 0xE2B0_0020

6 SPDIF TRANSMITTER

6.1 OVERVIEW OF SPDIF TRANSMITTER

SPDIF transmitter is based on IEC60958. This chapter describes a serial, un-directional, self-clocking interface to interconnect digital audio equipment in consumer and professional applications. If you use a consumer digital processing environment in SPDIF standard, the SPDIF interface is primarily intended to carry stereophonic programs, with a resolution of up to 20 bits per sample, an extension to 24 bits per sample being possible. If you use SPDIF in a broadcasting studio environment, the interface is primarily intended to carry monophonic or stereophonic programs, at a 48 kHz sampling frequency and with a resolution of up to 24 bits per sample; it can carry one or two signals sampled at 32 kHz.

In both cases, the clock references and auxiliary information are transmitted with the program. Provision in IEC60958 is made to allow the interface to carry software related data.

6.2 KEY FEATURES OF SPDIF TRANSMITTER

- SPDIFOUT module only supports the consumer application in S5PV210
- Supports linear PCM up to 24-bit per sample
- Supports Non-linear PCM formats such as AC3, MPEG1 and MPEG2
- 2 x 24-bit buffers which is alternately filled with data

6.3 BLOCK DIAGRAM OF SPDIF TRANSMITTER

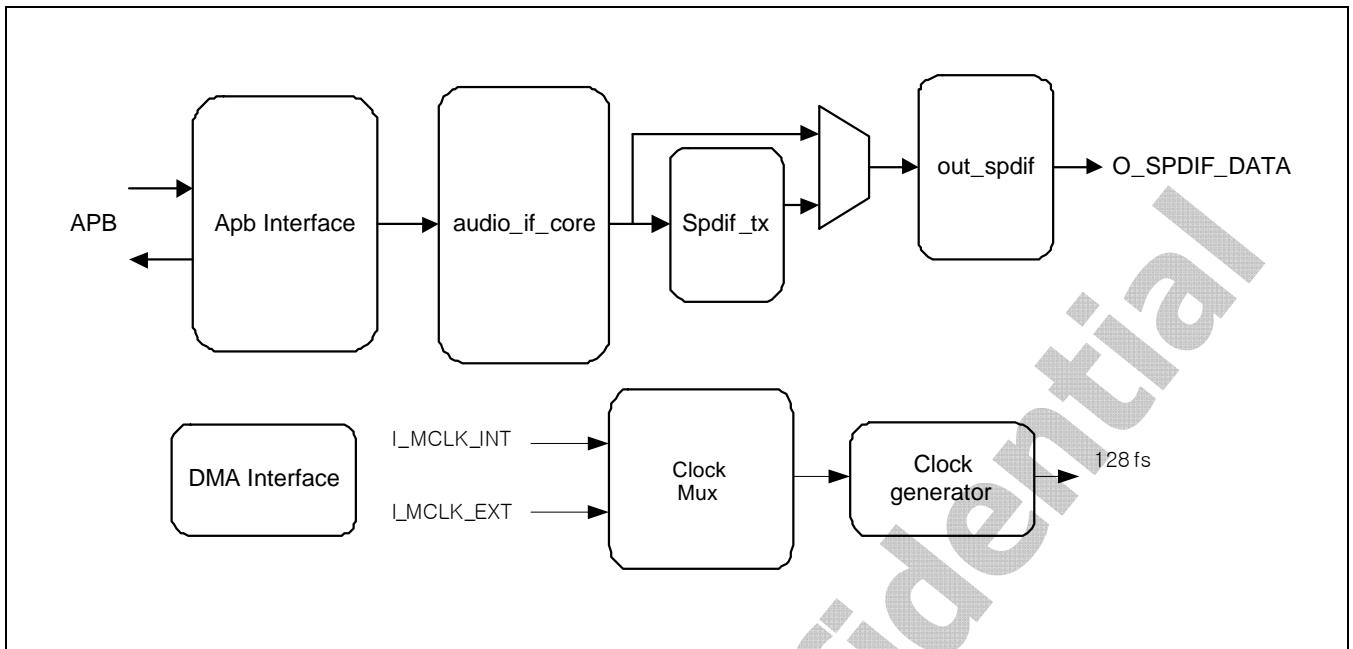


Figure 6-1 Block Diagram of SPDIFOUT

Components in SPDIF Transmitter:

- APB interface block: This block defines register banks to control the driving of SPDIFOUT module and data buffers to store linear or non-linear PCM data.
- DMA interface block: This block requests DMA service to IODMA depending on the status of data buffer in APB Interface block
- Clock Generator block: This block generates 128fs (sampling frequency) clock used in **out_spdif** block from system audio clock (MCLK)
- Clock Multiplex block: system audio clock (MCLK) can be selected as internal MCLK or external MCLK.
- Audio_if_core block: This block acts as interface block between data buffer and **out_spdif** block. Finite-state machine controls the flow of PCM data.
- **spdif_tx** block: This block inserts burst preamble and executes zero-stuffing in the nonlinear PCM stream. Linear PCM data are bypassed by **spdif_tx** module.
- **out_spdif** block: This block generates SPDIF format. It inserts 4-bit preamble, 16- or 20- or 24-bit data, user-data bit, validity bit, channel status bit and parity bit into the appropriate position of 32-bit word. It modulates each bit to bi-phase format.

6.4 FUNCTIONAL DESCRIPTIONS

6.4.1 DATA FORMAT OF SPDIF

6.4.1.1 Frame Format

A frame is uniquely composed of two sub-frames. The transmission rate of frames corresponds exactly to the source sampling frequency.

In the 2-channel operation mode, the time multiplexing transmits samples taken from both channels in consecutive sub-frames. Sub-frames related to channel 1(left or "A" channel in stereophonic operation and primary channel in monophonic operation) normally use preamble M. However, the preamble is changed to preamble B once every 192 frame. This unit composed of 192 frames defines the block structure used to organize the channel status information. Sub-frames of channel 2(right or "B" in stereophonic operation and secondary channel in monophonic operation) always use preamble W.

In the single channel operation mode in broadcasting studio environment the frame format is identical to the 2-channel mode. Data is carried only in channel 1. In the sub-frames allocated to channel 2, time slot 28 (validity flag) should be set to logical "1" ("1" means not valid).

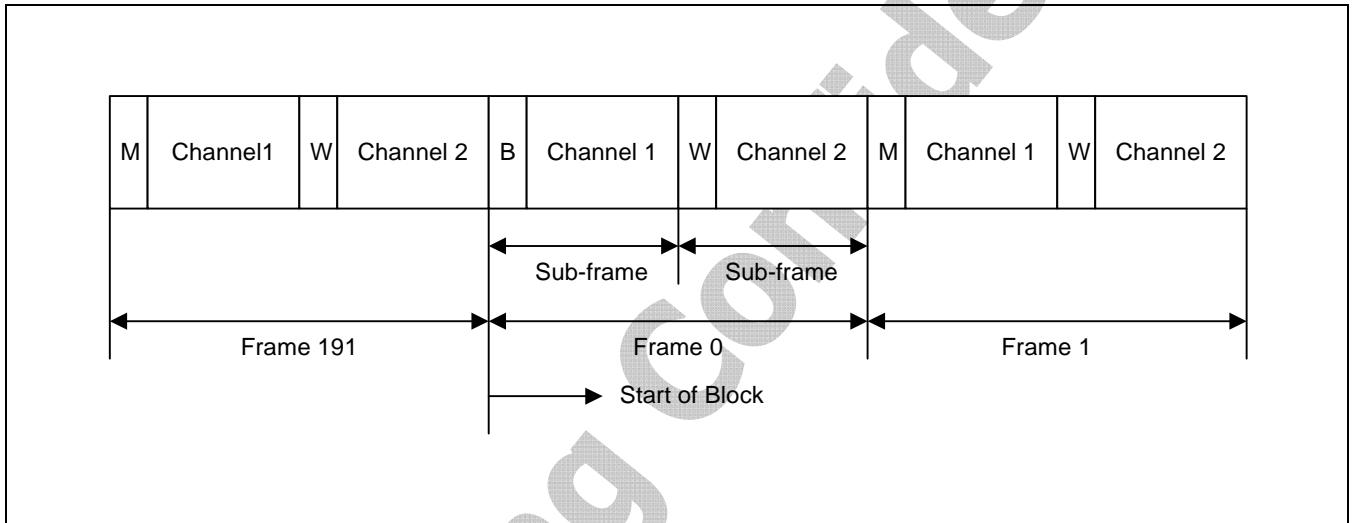


Figure 6-2 SPDIF Frame Format

6.4.1.2 Sub-frame Format (IEC 60958)

Each sub-frame is divided into 32 time slot, numbered from 0 to 31. Time slot 0 to 3 carries one of the three permitted preambles. These are used to affect synchronization of sub-frames, frames and blocks. Time slots 4 to 27 carry the audio sample word in linear 2's complement representation. The most significant bit is carried by time slot 27. When a 24-bit coding range is used, the least significant bit is in time slot 4. When a 20-bit coding range is sufficient, the least significant bit is in time slot 8 and time slot 4 to 7 may be used for other application. Under these circumstances, the bits in the time slot 4 to 7 are designated auxiliary sample bits.

If the source provides fewer bits than the interface allows (24 or 20), the unused least significant bits shall be set to a logical "0". This procedure supports to connect equipment using different numbers of bits. Time slot 28 carries the validity flag associated with the audio sample word. This flag is set to logical "0" if the audio sample is reliable. Time slot 29 carries one bit of the user data associated with the audio channel transmitted in the same sub-frame. The default value of the user bit is logical "0". Time slot 30 carries one bit of the channel status words associated with the audio channel transmitted in the same sub-frame. Time slot 31 carries a parity bit such that time slots 4 to 31 inclusive will carry an even number of ones and an even number of zeros.

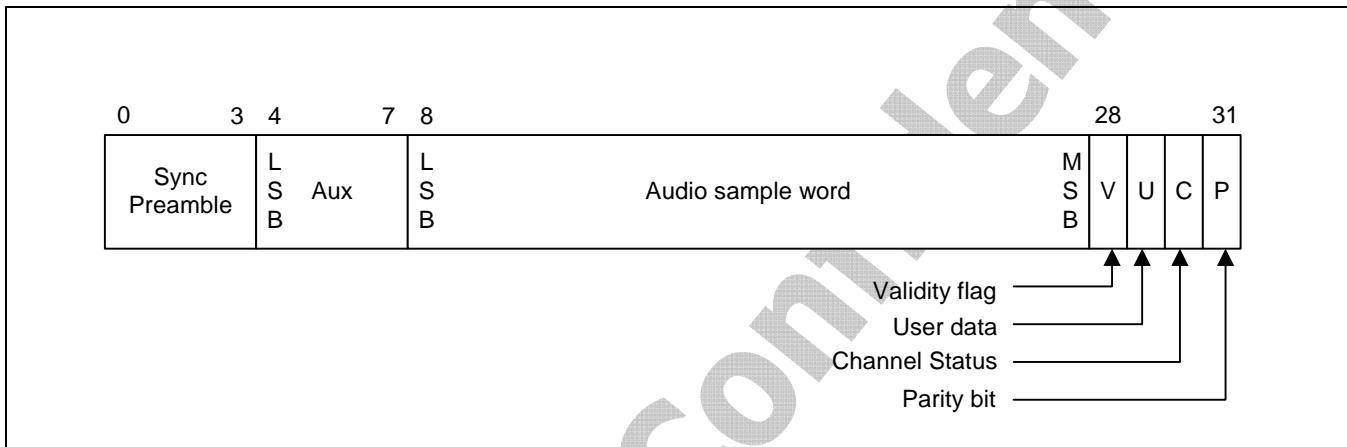


Figure 6-3 SPDIF Sub-frame Format

6.4.2 CHANNEL CODING

To minimize the dc component on the transmission line, to facilitate clock recovery from the data stream and to make the interface insensitive to the polarity of connections, time slots 4 to 31 are encoded in biphase-mark. A symbol comprising two consecutive binary states represent each bit to be transmitted. The first state of a symbol is always different from the second state of the previous symbol. The second state of the symbol is identical to the first if the bit to be transmitted is logical "0", is different from the first if the bit is logical "1".

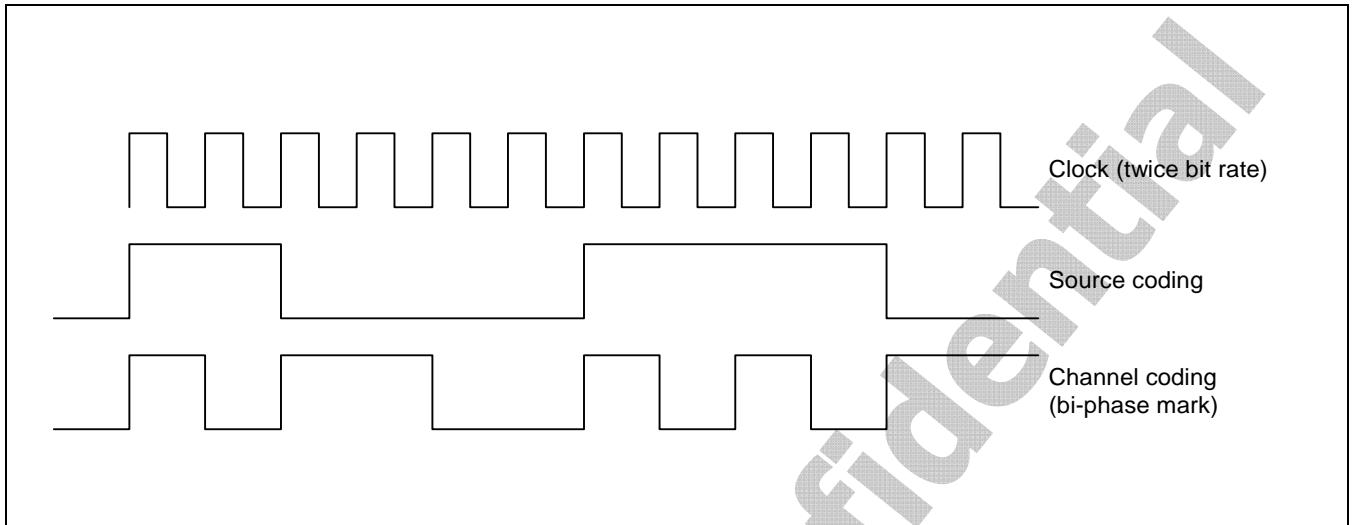


Figure 6-4 Channel Coding

6.4.3 PREAMBLE

Preambles are specific patterns providing synchronization and identification of the sub-frames and blocks. A set of three preambles (M, B and W) is used. These preambles are transmitted in the time allocated to four time slots (time slots 0 to 3) and are represented by eight successive states. The first state of the preamble is always different from the second state of the previous symbol.

Similar to bi-phase code, these preambles are dc free and provide clock recovery. They differ in minimum two states from any valid bi-phase sequence.

6.4.4 NON-LINEAR PCM ENCODED SOURCE (IEC 61937)

The non-linear PCM encoded audio bit stream is transferred using the basic 16-bit data area of the IEC 60958 sub frames, that is, in time slots 12 to 27. Each IEC 60958 frame can transfer 32 bits of the non-PCM data in consumer application mode.

When the SPDIF bit stream conveys linear PCM audio, the symbol frequency is 64 times the PCM sampling frequency (32 time slots per PCM sample time's two channels). When a non-linear PCM encoded audio bit stream is transmitted by the interface, the symbol frequency shall be 64 times the sampling rate of the encoded audio within that bit stream. If a non-linear PCM encoded audio bit stream is transmitted by the interface containing audio with low sampling frequency, the symbol frequency shall be 128 times the sampling rate of the encoded audio within that bit stream.

Each data burst contains a burst-preamble consisting of four 16-bit words (Pa, Pb, Pc, Pd), followed by the burst-payload which contains data of an encoded audio frame.

The burst-preamble consists of four mandatory fields. Pa and Pb represent a synchronization word; Pc gives information about the type of data and some information/control for the receiver; Pd gives the length of the burst-payload, limited to 216(=65,535) bits.

The four preamble words are contained in two sequential SPDIF frames. The frame beginning the data-burst contains preamble word Pa in subframe 1 and Pb in subframe 2. The next frame contains Pc in subframe 1 and Pd in subframe 2. When placed into a SPDIF subframe, the MSB of a 16-bit burst-preamble is placed into time slot 27 and the LSB is placed into time slot 12.

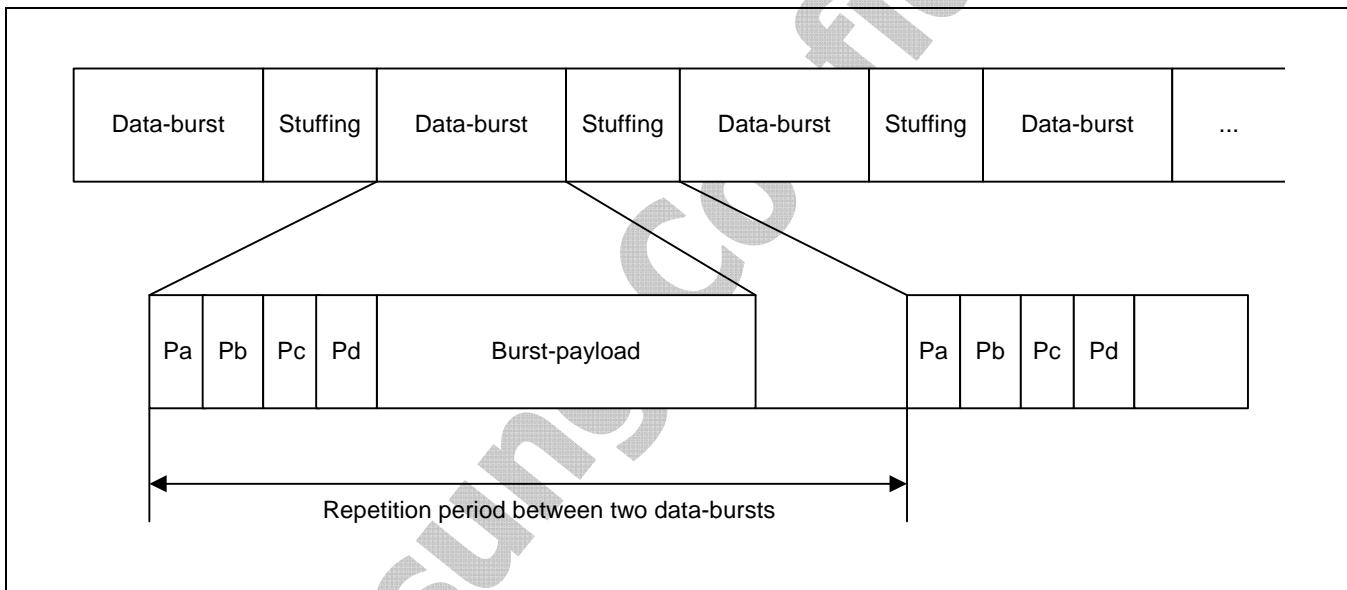


Figure 6-5 Format of Burst Payload

Table 6-1 Burst Preamble Words

Preamble word	Length of field	Contents	Value MSB.. LSB
Pa	16 bits	Sync word 1	0xF872
Pb	16 bits	Sync word 2	0x4E1F
Pc	16 bits	Burst-info	Refer to SPDBSTAS_SHD[15:0]
Pd	16 bits	Length-code	Refer to SPDBSTAS_SHD[31:16]

6.4.5 SPDIF OPERATION

Since the bit frequency of SPDIF is 128fs (fs: sampling frequency), divide audio main clock (MCLK) depending on the frequency of MCLK to make the main clock of SPDIF. MCLK is divided by 2 in case of 256fs, by 3 in case of 384fs and by 4 in case of 512fs.

SPDIF module in S5PV210 changes the audio sample data format to SPDIF. To change the format, SPDIF module inserts preamble data, channel status data, user data, error check bit and parity bit into the appropriate time slots. Preamble data are fixed in the module and inserted depending on subframe counter. Channel status data are set in the SPDCSTAS register and used by one bit per frame. User data always have zero values.

For non-linear PCM data, insert burst-preamble, which consists of Pa, Pb, Pc and Pd, before burst-payload and zero is padded from the end of burst-payload to the repetition count. Pa(=16'hF872) and Pb(=16'h4E1F) is fixed in the module and Pc and Pd is set in the register SPDBSTAS. To stuff zero, the end of burst-payload is calculated from Pd value and repetition count which depends on data type in the preamble. Pc is acquired from register SPDCNT.

Audio data are justified to the LSB. 16-, 20- or 24-bit PCM data and 16-bit stream data are supported. The unoccupied upper bits of 32-bit word are ignored.

Data are fetched via DMA request. If one of two data buffers is empty, DMA service is requested. Audio data stored in the data buffers are transformed into SPDIF format and output to the port. For non-linear PCM data, interrupt is generated after audio data are output up to the value specified in the SPDCNT register. Interrupt sets the registers such as SPDBSTAS and SPDCNT to new values, if data type of new bitstream is different from the previous one.

6.4.6 SHADOWED REGISTER

Both SPDBSTAS_SHD register and SPDCNT_SHD register are shadowed registers which are related to SPDBSTAS register and SPDCNT register, respectively. They are updated from related registers at every stream end interrupt signal. The usage of shadowed register is as follows.

1. Set burst status and repetition count information to their respective registers.
2. Turn on SPDIF module, and stream end interrupt is asserted immediately.
3. With stream end interrupt, shadowed registers are updated from their related registers and SPDIF starts to transfer data. Now next stream information (burst status and repetition count) can be written to SPDBSTAS and SPDCNT register because previous information is copied to their respective shadowed registers.
4. Set next stream information to SPDBSTAS and SPDCNT register.
5. Wait for stream end interrupt which signals the end of the first stream.
6. With stream end interrupt, the 2nd stream data will start to transfer. Set 3rd stream information to registers.

The usage of user bit registers is similar to stream information registers except that they are not related to SPDIF end interrupt but to user data interrupt. As soon as SPDIF is on, shadowed user bit registers are updated from their related registers and user bit starts to be shifted out with user data interrupt asserted. User can write the next user data to registers with this interrupt. After entire 96 user bits are shifted out, user data interrupt will be asserted again and 3rd user bits can be written to registers with 2nd user bits going out.

6.5 I/O DESCRIPTION

SPDIF external pads are shared with I2S and PCM. In order to use these pads for SPDIF, GPIO must be set before the SPDIF starts. For more information, refer to the GPIO chapter of this manual for exact GPIO setting

Signal	I/O	Description	Pad	Type
I_MLCK_EXT	I	Global audio main clock (External MCLK)	Xpcm0EXTCLK	muxed
O_SPDIF_DATA	O	SPDIFOUT data output	Xpcm0SCLK	muxed

6.6 REGISTER DESCRIPTION

6.6.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
SPDCLKCON	0xE110_0000	R/W	Specifies the Clock control register	0x0000_0002
SPDCON	0xE110_0004	R/W	Specifies the Control register	0x0000_0000
SPDBSTAS	0xE110_0008	R/W	Specifies the Burst status register	0x0000_0000
SPDCSTAS	0xE110_000C	R/W	Specifies the Channel status register	0x0000_0000
SPDDAT	0xE110_0010	W	Specifies the SPDIFOUT data buffer	0x0000_0000
SPDCNT	0xE110_0014	R/W	Specifies the Repetition count register	0x0000_0000
SPDBSTAS_SHD	0xE110_0018	R	Specifies the Shadowed Burst Status Register	0x0000_0000
SPDCNT_SHD	0xE110_001C	R	Specifies the Shadowed Repetition Count Register	0x0000_0000
USERBIT1	0xE110_0020	R/W	Specifies the Subcode Q1 ~ Q32	0x0000_0000
USERBIT2	0xE110_0024	R/W	Specifies the Subcode Q33 ~ Q64	0x0000_0000
USERBIT3	0xE110_0028	R/W	Specifies the Subcode Q65 ~ Q96	0x0000_0000
USERBIT1_SHD	0xE110_002C	R	Specifies the Shadowed Register Userbit1	0x0000_0000
USERBIT2_SHD	0xE110_0030	R	Specifies the Shadowed Register Userbit2	0x0000_0000
USERBIT3_SHD	0xE110_0034	R	Specifies the Shadowed Register Userbit3	0x0000_0000
VERSION_INFO	0xE110_0038	R	Specifies the RTL Version Information	0x0000_000C

6.6.1.1 SPDIFOUT Clock Control Register (SPDCLKCON, R/W, Address = 0XE110_0000)

SPDCLKCON	Bit	Description	Initial State
-	[31:3]	Reserved	0
Main Audio Clock Selection	[2]	0 = Internal clock (I_MCLK_INT) 1 = External clock (I_MCLK_EXT)	0
SPDIFOUT clock down ready (read only)	[1]	0 = Clock-down not ready 1 = Clock-down ready	1
SPDIFOUT power on	[0]	0 = Power off 1 = Power on	0

6.6.1.2 SPDIFOUT Control Register (SPDCON, R/W, Address = 0XE110_0004)

SPDCON	Bit	Description	Initial State
-	[31:27]	Reserved	0
FIFO Level	[26:22]	FIFO Level Monitoring (Read Only) FIFO depth is 16 *0 = Empty of FIFO Level, 16 = Full of FIFO Level	00000
FIFO Level Threshold	[21:19]	FIFO Threshold Level is controllable 000 = 0-FIFO Level 001 = 1-FIFO Level 010 = 4-FIFO Level 011 = 6-FIFO Level 100 = 10-FIFO Level 101 = 12-FIFO Level 110 = 14-FIFO Level 111 = 15-FIFO Level	000
FIFO transfer mode	[18:17]	00 = DMA transfer mode 01 = Polling mode 10 = Interrupt mode 11 = Reserved	00
FIFO_level Interrupt Status	[16]	Read Operation 0 = No interrupt pending. 1 = Interrupt pending. Write Operation 0 = No effect. 1 = Clear this flag.	0
FIFO_level Interrupt Enable	[15]	0 = Interrupt masked 1 = Interrupt enable	0
Endian format	[14:13]	00 = big endian 0_data = {in_data[23:0]} 01 = 4 byte swap 0_data={in_data[15:8], in_data[23:16], in_data[31:24]}	0

SPDCON	Bit	Description	Initial State
		10 = 3 byte swap o_data={in_data[7:0], in_data[15:8], in_data[23:16]} 11 = 2 byte swap o_data={0x00,in_data[7:0], in_data[15:8]} *in_data: BUS → in port of SPDIF o_data: in port of SPDIF → Logic	
user_data_attach	[12]	0 = User data is stored in USERBIT register. User data of subframe is out from USERBIT1,2,3 (96-bit) 1 = User data is stored in 23rd bit of audio data. User data is out in PCM data's 23th bit.	0
User Data Interrupt Status	[11]	Read Operation 0 = No interrupt pending. 1 = Interrupt pending when 96-bit of user data is out. Write Operation 0 = No effect. 1 = Clear this flag.	0
User Data Interrupt Enable	[10]	0 = Interrupt masked 1 = Interrupt enable	0
Buffer Empty Interrupt Status	[9]	Read Operation 0 = No interrupt pending. 1 = Interrupt pending. Write Operation 0 = No effect. 1 = Clear this flag.	0
Buffer Empty Interrupt Enable	[8]	0 = Interrupt masked 1 = Interrupt enable	0
Stream End Interrupt Status	[7]	Read Operation 0 = No interrupt pending. 1 = Interrupt pending when the number of output audio data reaches repetition count in SPDCNT register. Write Operation 0 = No effect. 1 = Clear this flag.	0
Stream End Interrupt Enable	[6]	0 = Interrupt masked 1 = Interrupt enable	0
software reset	[5]	0 = Normal operation 1 = Software reset Software reset is 1-cycle pulse (auto clear) Enable I_MCLK before software reset assertion because SPDIF uses synchronous reset	0
Main Audio Clock Frequency	[4:3]	00 = 256fs 01 = 384fs 10 = 512fs	0

SPDCON	Bit	Description	Initial State
		11 = Reserved If you want to use SPDIF on HDMI, 512fs should be selected. Because HDMI in S5PV210 accepts only 512fs or more frequency.	
PCM Data Size	[2:1]	00 = 16-bit 01 = 20-bit 10 = 24-bit 11 = Reserved	0
PCM or Stream	[0]	0 = Stream 1 = PCM	0

6.6.1.3 SPDIFOUT Burst Status Register (SPDBSTAS, R/W, Address = 0XE110_0008)

SPDBSTAS	Bit	Description	Initial State
Burst data length bit	[31:16]	ES size in bits (Burst Preamble Pd) ES size: Elementary Stream size This indicates Burst-payload length	0
Bitstream number	[15:13]	Bit_stream_number, shall be set to 0	0
Data type dependent info	[12:8]	Data type dependent information	0
Error flag	[7]	0 = Error flag indicates a valid burst_payload 1 = Error flag indicates that the burst payload may contain errors	0
-	[6:5]	Reserved	0
Compressed data type	[4:0]	00000 = Null Data 00001 = AC-3 00010 = Reserved 00011 = Pause 00100 = MPEG1 (layer1) 00101 = MPEG1 (layer2, 3), MPEG2-bc 00110 = MPEG2 – extension 00111 = Reserved 01000 = MPEG2 (layer1 – lsf) 01001 = MPEG2 (layer2, layer3 – lsf) Others = Reserved	0

6.6.1.4 SPDIFOUT Channel Status Register (SPDCSTAS, R/W, Address = 0XE110_000C)

SPDCSTAS	Bit	Description	Initial State
-	[31:30]	Reserved	0
Clock accuracy	[29:28]	10 = Level I, ± 50 ppm 00 = Level II, ± 1000 ppm 01 = Level III, variable pitch shifted	0
Sampling frequency	[27:24]	0000 = 44.1 kHz 0010 = 48 kHz 0011 = 32 kHz 1010 = 96 kHz	0
Channel number	[23:20]	Bit 20 is LSB	0
Source number	[19:16]	Bit 16 is LSB	0
Category code	[15:8]	Equipment type CD player = 0000_0001 DAT player = L000_0011 DCC player = L100_0011 Mini disc = L100_1001 (L: information about generation status of the material)	0
Channel status mode	[7:6]	00 = Mode 0 Others = Reserved	0
Emphasis	[5:3]	When bit1 = 0, 000 = 2 audio channels without pre-emphasis 001 = 2 audio channels with 50us / 15us pre-emphasis When bit1 = 1, 000 = default state	0
Copyright assertion	[2]	0 = Copyright 1 = No copyright	0
Audio sample word	[1]	0 = Linear PCM 1 = Non-linear PCM	0
Channel status block	[0]	0 = Consumer format 1 = Professional format	0

6.6.1.5 SPDIFOUT Data Buffer (SPDDAT, W, Address = 0XE110_0010)

SPDDAT	Bit	Description	Initial State
-	[31:24]	Reserved	0
SPDIFOUT data	[23:0]	PCM or stream data	0

6.6.1.6 SPDIFOUT Repetition Count Register (SPDCNT, R/W, Address = 0XE110_0014)

SPDCNT	Bit	Description	Initial State
-	[31:13]	Reserved	0
Stream repetition count	[12:0]	Repetition count according to data type. This bit is valid only for stream data.	0

6.6.1.7 Shadowed SPDIF Burst Status Register (SPDBSTAS_SHD, R, Address = 0XE110_0018)

SPDBSTAS	Bit	Description	Initial State
Burst Data Length Bit	[31:16]	ES size in bits (Burst Preamble Pd) ES size: Elementary Stream size This indicates Burst-payload length	0
Bitstream number	[15:13]	Bit_stream_number, shall be set to 0	0
Data Type Dependent Info	[12:8]	Data type dependent information	0
Error Flag	[7]	0 = Error flag indicating a valid burst_payload 1 = Error flag indicating that the burst payload may contain errors	0
-	[6:5]	Reserved	0
Compressed Data Type	[4:0]	00000 = Null Data 00001 = AC-3 00010 = Reserved 00011 = Pause 00100 = MPEG1 (layer1) 00101 = MPEG1 (layer2, 3), MPEG2-bc 00110 = MPEG2 – extension 00111 = Reserved 01000 = MPEG2 (layer1 – lsf) 01001 = MPEG2 (layer2, layer3 – lsf) Others = Reserved	0

6.6.1.8 Shadowed SPDIF Repetition Count Register (SPDCNT_SHD, R, Address = 0XE110_001C)

SPDCNT	Bit	Description	Initial State
-	[31:13]	Reserved	0
Stream Repetition Count	[12:0]	Repetition count according to data type. This bit is valid only for stream data.	0

6.6.1.9 User Data Register (USERBIT1~3)

- USERBIT1, R/W, Address = 0XE110_0020
- USERBIT2, R/W, Address = 0XE110_0024
- USERBIT3, R/W, Address = 0XE110_0028

USERBIT1~3	Bit	Description	Initial State
User Data Bit (subcode Q for CD)	[31:0]	USERBIT1: Q1 ~ Q32 USERBIT2: Q33 ~ Q64 USERBIT3: Q65 ~ Q96 User Data Bit has the Digital Audio Track information (Track NO, Play Time etc.). 1176 bits of these being taken out in a row.	0

6.6.1.10 Shadowed User Data Register (USERBIT_SHD)

- USERBIT1_SHD, R, Address = 0XE110_002C
- USERBIT2_SHD, R, Address = 0XE110_0030
- USERBIT3_SHD, R, Address = 0XE110_0034

USERBIT_SHD	Bit	Description	Initial State
-------------	-----	-------------	---------------

7

ADC & TOUCH SCREEN INTERFACE

This chapter describes the functions and usage of ADC and Touch Screen interface.

7.1 OVERVIEW OF ADC & TOUCH SCREEN INTERFACE

The 10-bit or 12-bit CMOS Analog to Digital Converter (ADC) comprises of 10-channel analog inputs. It converts the analog input signal into 10-bit or 12-bit binary digital codes at a maximum conversion rate of 1MSPS with 5MHz A/D converter clock. A/D converter operates with on-chip sample-and-hold function. ADC supports low power mode.

Touch Screen Interface can control input pads (XP, XM, YP, and YM) to obtain X/Y-position on the external touch screen device. Touch Screen Interface contains three main blocks, namely, touch screen pads control logic, ADC interface logic and interrupt generation logic. There are two set of touch screen interfaces, which share one ADC.

7.2 KEY FEATURES OF ADC & TOUCH SCREEN INTERFACE

The ADC & Touch Screen interface includes the following features:

- Resolution: 10-bit / 12-bit (optional)
- Differential Nonlinearity Error: ± 1.0 LSB (Max.)
- Integral Nonlinearity Error: ± 4.0 LSB (Max.)
- Maximum Conversion Rate: 1 MSPS
- Low Power Consumption
- Power Supply Voltage: 3.3V
- Analog Input Range: 0 ~ 3.3V
- On-chip sample-and-hold function
- Normal Conversion Mode
- Separate X/Y position conversion Mode
- Auto (Sequential) X/Y Position Conversion Mode
- Waiting for Interrupt Mode
- IDLE, DIDLE, STOP and DSTOP mode wakeup source
- Two touch screen interfaces

7.3 TOUCH SCREEN INTERFACE OPERATION

7.3.1 BLOCK DIAGRAM ADC & TOUCH SCREEN INTERFACE

Figure 7-1 is the functional block diagram of A/D converter and Touch Screen Interface.

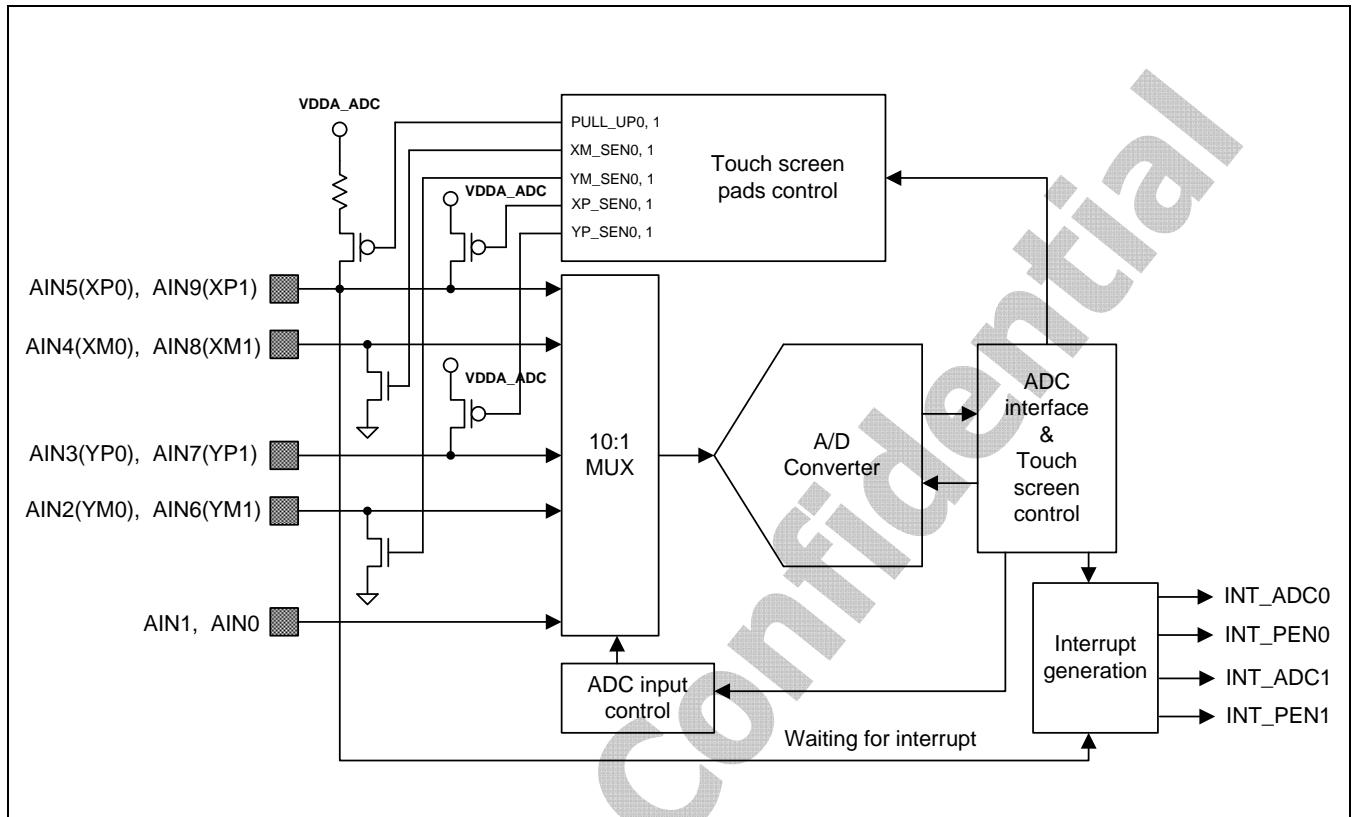


Figure 7-1 ADC and Touch Screen Interface Functional Block Diagram

NOTE: When Touch Screen device is not used, XM, XP, YM or YM can be connected to Analog Input Signal for Normal ADC conversion.

7.4 FUNCTION DESCRIPTIONS

7.4.1 A/D CONVERSION TIME

When the PCLK frequency is 66MHz and the prescaler value is 65, total 12-bit conversion time is as follows.

- A/D converter freq. = $66\text{MHz}/(65+1) = 1\text{MHz}$
- Conversion time = $1/(1\text{MHz} / 5\text{cycles}) = 1/200\text{kHz} = 5\text{us}$

NOTE: This A/D converter was designed to operate at maximum 5MHz clock, so the conversion rate can go up to 1MSPS.

7.4.2 TOUCH SCREEN INTERFACE MODE

1. Normal Conversion Mode (AUTO_PST = 0, XY_PST = 0)

The operation of this mode is same as AIN0~AIN9's. To initialize this mode, set the TSADCCON0 (ADC control register) and TSCONn (Touch screen control register). The switches and pull-up resistor should be turned off (all switches are turned off if TSCON0 and TSCON1 are set to 0x58). The converted data can be read out from TSDATX0 (ADC conversion data X register).

NOTE: TSADCCON1 register is useless in normal conversion mode. Therefore, TSSEL bit of TSADCCON0 register should be 0. TSADCCON1 register is meaningless if TSSEL bit is 0.

2. Separate X/Y Position Conversion Mode (AUTO_PST = 0, XY_PST = control)

This mode consists of two states, namely, X-position measurement state and Y-position measurement state.

Steps to operate X-position measurement state;

- Set '0x69' to TSCONn.
(XY_PST=1, AUTO_PST=0, PULL_UP disable, XP enable, XM enable, YP disable, YM disable)
- Start conversion by setting TSADCCONn.
- The end of X-position conversion can be notified by interrupt (INT_ADCn).
- Read out the converted data (X-position) from TSDATXn.

Steps to operate Y-position measurement state;

- Set '0x9a' to TSCONn.
(XY_PST=2, AUTO_PST=0, PULL_UP disable, XP disable, XM disable, YP enable, YM enable)
- Start conversion by setting TSADCCONn.
- The end of Y-position conversion can be notified by interrupt (INT_ADCn).
- Read out the converted data (Y-position) from TSDATYn.

7.4.2.1 Touch Screen0 pin Conditions in X/Y Position MEASUREMENT

State	XP0	XM0	YP0	YM0
TS0: X-position measurement	VDDA_ADC	VSSA_ADC	AIN3	Hi-z
TS0: Y-position measurement	AIN5	Hi-z	VDDA_ADC	VSSA_ADC

7.4.2.2 Touch Screen1 pin Conditions in X/Y Position MEASUREMENT

State	XP1	XM1	YP1	YM1
TS1: X-position measurement	VDDA_ADC	VSSA_ADC	AIN7	Hi-z
TS1: Y-position measurement	AIN9	Hi-z	VDDA_ADC	VSSA_ADC

3. Auto (Sequential) X/Y Position Conversion Mode (AUTO_PST = 1, XY_PST = 0)

Steps to operate Auto (Sequential) X/Y Position Conversion Mode:

- Set '0x5c' to TSCONn. (XY_PST=0, AUTO_PST=1, PULL_UP disable, XP disable, XM disable, YP disable, YM disable)
- Start conversion by setting TSADCCONn.
- Touch screen controller converts X-Position and writes it to TSDATXn.
- Touch screen controller converts Y-Position and writes it to TSDATYn.
- Touch screen interface generates interrupt (INT_ADCn). In other words, INT_ADCn is occurred only once, not twice.

4. Waiting for Interrupt Mode (TSCONn[7:0] = 0xd3)

Touch screen controller generates an interrupt signal (INT_PENn) when the stylus pen is down or up. The value of TSCONn[7:0] should be '0xd3', that is, pull-up enable, XP disable, XM disable, YP disable and YM enable. After touch screen controller generates interrupt signal (INT_PENn), waiting for interrupt Mode must be cleared (Set 0 to XY_PST).

7.4.2.3 Touch Screen0 Pin Conditions in Wait for Interrupt Mode

Mode	XP0	XM0	YP0	YM0
TS0: Waiting for Interrupt Mode	VDDA_ADC (Pull-up enable)	Hi-z	Hi-z	VSSA_ADC

7.4.2.4 Touch Screen1 Pin Conditions in Wait for Interrupt Mode

Mode	XP1	XM1	YP1	YM1
TS1: Waiting for Interrupt Mode	VDDA_ADC (Pull-up enable)	Hi-z	Hi-z	VSSA_ADC

7.4.3 STANDBY MODE

Standby mode is activated when TSSEL bit is '0' and STANDBY bit is '1' in TSADCCON0 register. In this mode, A/D conversion operation is halted and TSDATXn and TSDATYn registers hold their values.

7.4.4 TWO TOUCH SCREEN INTERFACES

There are two set of touch screen interfaces, namely, AIN[5] ~ AIN[2] for touch screen 0 and AIN[9] ~ AIN[6] for touch screen 1. There are separate switches for XP, XM, YP and YM control and separate registers to interface with two touch screens. They share one analog digital converter, so interfacing with the two touch screens should be performed in turn. TSSEL bit of TSADCCON0 register is used to select which touch screen is connected to the ADC. Therefore, you must set '1' to TSSEL before an access to TSADCCON1. Similarly, you must set '0' to TSSEL before an access to TSADCCON0.

An access to TSADCCON1 bits is prohibited when TSSEL bit is '0', and an access to TSADCCON0 bits except TSSEL is also prohibited when TSSEL bit is '1'. An access to TSSEL bit is always permitted.

Programming Notes

1. The A/D converted data can be accessed by means of interrupt or polling method. With interrupt method, the overall conversion time - from A/D converter start to converted data read - may be delayed because of the return time of interrupt service routine and data access time. With polling method, to determine the read time for TSDATXn or TSDATYn register, check the TSADCCONn[15] – end of conversion flag – bit.
2. A/D conversion can be activated in different way. After TSADCCONn[1] - A/D conversion start-by-read mode- is set to 1. A/D conversion starts simultaneously when converted data is read.

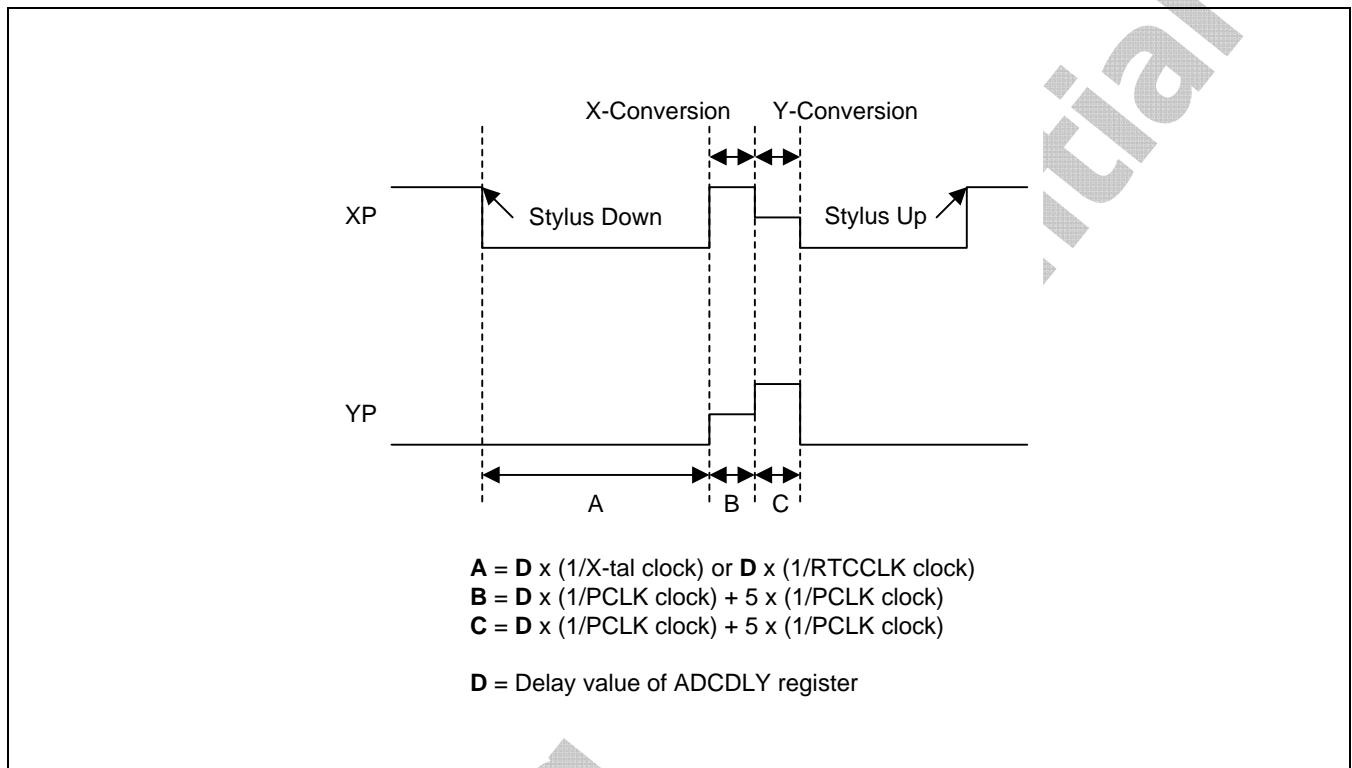


Figure 7-2 ADC and Touch Screen Operation Signal

3. If INT_PEN interrupt is used as an wakeup source in IDLE, DIDLE, STOP and DSTOP mode, XY_PST bit (TSCONn[1:0]) should be set to waiting for interrupt mode (2b'11). UD_SEN bit (TSCONn[8]) determines a stylus pen up wakeup or pen down wakeup.

7.5 ADC & TOUCH SCREEN INTERFACE INPUT CLOCK DIAGRAM

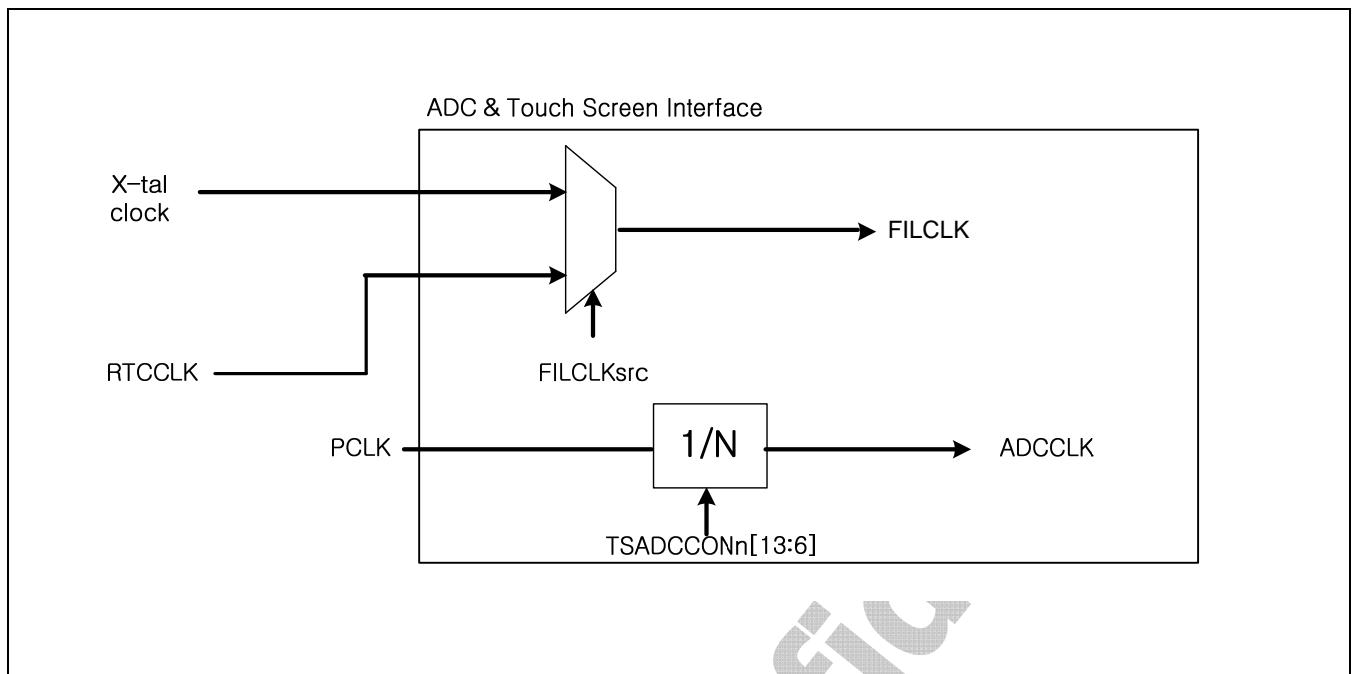


Figure 7-3 Input Clock Diagram for ADC & Touch Screen Interface

7.6 I/O DESCRIPTIONS

Signal	I/O	Description	Pad	Type
AIN[9]	Input	ADC Channel[9] Analog input	XadcAIN[9]	Analog
AIN[8]	Input	ADC Channel[8] Analog input	XadcAIN[8]	Analog
AIN[7]	Input	ADC Channel[7] Analog input	XadcAIN[7]	Analog
AIN[6]	Input	ADC Channel[6] Analog input	XadcAIN[6]	Analog
AIN[5]	Input	ADC Channel[5] Analog input	XadcAIN[5]	Analog
AIN[4]	Input	ADC Channel[4] Analog input	XadcAIN[4]	Analog
AIN[3]	Input	ADC Channel[3] Analog input	XadcAIN[3]	Analog
AIN[2]	Input	ADC Channel[2] Analog input	XadcAIN[2]	Analog
AIN[1]	Input	ADC Channel[1] Analog input	XadcAIN[1]	Analog
AIN[0]	Input	ADC Channel[0] Analog input	XadcAIN[0]	Analog

7.7 REGISTER DESCRIPTION

7.7.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
TSADCCON0	0xE170_0000	R/W	Specifies the TS0 - ADC Control Register	0x0000_3FC4
TSCON0	0xE170_0004	R/W	Specifies the TS0 - Touch Screen Control Register	0x0000_0058
TSDELAY0	0xE170_0008	R/W	Specifies the TS0 - ADC Start or Interval Delay Register	0x0000_00FF
TSDATX0	0xE170_000C	R	Specifies the TS0 - ADC Conversion Data X Register	-
TSDATY0	0xE170_0010	R	Specifies the TS0 - ADC Conversion Data Y Register	-
TSPENSTAT0	0xE170_0014	R/W	Specifies the TS0 - Pen0 Up or Down Status Register	0x0000_0000
CLRINTADC0	0xE170_0018	W	Specifies the TS0 - Clear ADC0 Interrupt	-
ADCMUX	0xE170_001C	R/W	Specifies the Analog input channel selection	0x0000_0000
CLRINTPEN0	0xE170_0020	W	Specifies the TS0 - Clear Pen0 Down/Up Interrupt	-
TSADCCON1	0xE170_1000	R/W	Specifies the TS1 - ADC Control Register	0x0000_3FC4
TSCON1	0xE170_1004	R/W	Specifies the TS1 - Touch Screen Control Register	0x0000_0058
TSDELAY1	0xE170_1008	R/W	Specifies the TS1 - ADC Start or Interval Delay Register	0x0000_00FF
TSDATX1	0xE170_100C	R	Specifies the TS1 - ADC Conversion Data X Register	-
TSDATY1	0xE170_1010	R	Specifies the TS1 - ADC Conversion Data Y Register	-
TSPENSTAT1	0xE170_1014	R/W	Specifies the TS1 - Pen1 Up or Down Status Register	0x0000_0000
CLRINTADC1	0xE170_1018	W	Specifies the TS1 - Clear ADC1 Interrupt	-
CLRINTPEN1	0xE170_1020	W	Specifies the TS1 - Clear Pen1 Up/Down Interrupt	-

7.7.1.1 ADC Control Register (TSADCCONn)

- TSADCCON0, R/W, Address = 0xE170_0000
- TSADCCON1, R/W, Address = 0xE170_1000

TSADCCONn	Bit	Description	Initial State
TSSEL	[17]	Touch screen selection 0 = Touch screen 0 (AIN2~AIN5) 1 = Touch screen 1 (AIN6~AIN9) This bit exists only in TSADCCON0. Note: An access to TSADCCON1 bits is prohibited when TSSEL bit is 0, and an access to TSADCCON0 bits except TSSEL is prohibited when TSSEL bit is 1. An access to TSSEL bit is always permitted.	0
RES	[16]	ADC output resolution selection 0 = 10bit A/D conversion 1 = 12bit A/D conversion	0
ECFLG	[15]	End of conversion flag(Read only) 0 = A/D conversion in process 1 = End of A/D conversion	0
PRSCEN	[14]	A/D converter prescaler enable 0 = Disable 1 = Enable	0
PRSCVL	[13:6]	A/D converter prescaler value Data value: 5 ~ 255 The division factor is (N+1) when the prescaler value is N. For example, ADC frequency is 3.3MHz if PCLK is 66MHz and the prescaler value is 19. Note: This A/D converter is designed to operate at maximum 5MHz clock, so the prescaler value should be set such that the resulting clock does not exceed 5MHz.	0xFF
Reserved	[5:3]	Reserved	0
STANDBY	[2]	Standby mode select 0 = Normal operation mode 1 = Standby mode Note: In standby mode, prescaler should be disabled to reduce more leakage power consumption.	1
READ_START	[1]	A/D conversion start by read 0 = Disables start by read operation 1 = Enables start by read operation	0
ENABLE_START	[0]	A/D conversion starts by enable. If READ_START is enabled, this value is not valid. 0 = No operation 1 = A/D conversion starts and this bit is automatically cleared after the start-up.	0

7.7.1.2 Touch Screen Control Register (TSCONn)

- TSCON0, R/W, Address = 0xE170_0004
- TSCON1, R/W, Address = 0xE170_1004

TSCONn	Bit	Description	Initial State
UD_SEN	[8]	Detect Pen Up or Down status. 0 = Detects pen down. 1 = Detects pen up.	0
YM_SEN	[7]	YM to GND Switch Enable 0 = Switch disable.(YM = Hi-z) 1 = Switch enable(YM = VSSA_ADC)	0
YP_SEN	[6]	YP to VDD Switch Enable 0 = Switch enable. (YP = VDDA_ADC) 1 = Switch disable.(YP = Hi-z)	1
XM_SEN	[5]	XM to GND Switch Enable 0 = Switch disable.(XM = Hi-z) 1 = Switch enable.(XM = VSSA_ADC)	0
XP_SEN	[4]	XP to VDD Switch Enable 0 = Switch enable.(XP = VDDA_ADC) 1 = Switch disable.(XP = Hi-z)	1
PULL_UP	[3]	Pull-up Switch Enable 0 = XP Pull-up Enable. 1 = XP Pull-up Disable.	1
AUTO_PST	[2]	Automatic sequencing conversion of X-Position and Y-Position 0 = Normal ADC conversion. 1 = Auto Sequential measurement of X-position, Y-position.	0
XY_PST	[1:0]	Manually measurement of X-Position or Y-Position. 00 = No operation mode 01 = X-position measurement 10 = Y-position measurement 11 = Waiting for Interrupt Mode	0

NOTE:

1. While waiting for touch screen Interrupt, XP_SEN bit must be set to '1', namely 'XP output disable' and PULL_UP bit must be set to '0', namely 'XP pull-up enable'.
2. AUTO_PST bit should be set '1' only in Automatic & Sequential X/Y Position conversion.

Touch screen0 pin conditions in X/Y position conversion.

	XP0	XM0	YP0	YM0	ADC ch. select
TS0: X Position	Vref	GND	AIN[3]	Hi-Z	YP0
TS0: Y Position	AIN[5]	Hi-Z	Vref	GND	XP0

Touch screen1 pin conditions in X/Y position conversion.

	XP1	XM1	YP1	YM1	ADC ch. select
TS1: X Position	Vref	GND	AIN[7]	Hi-Z	YP1
TS1: Y Position	AIN[9]	Hi-Z	Vref	GND	XP1

7.7.1.3 ADC Delay Register (TSDLYn)

- TSDLY0, R/W, Address = 0xE170_0008
- TSDLY1, R/W, Address = 0xE170_1008

TSDLYn	Bit	Description	Initial State
FILCLKsrc	[16]	Reference clock source for delay. 0 = X-tal clock. 1 = RTC clock.	0
DELAY	[15:0]	In case of ADC conversion mode (Normal, Separate, Auto conversion); ADC conversion is delayed by counting this value. Counting clock is PCLK. → ADC conversion delay value. In case of waiting for Interrupt mode: When stylus down occurs in waiting for interrupt mode, it generates interrupt signal (INT_PENn) at interval of several ms for Auto X/Y position conversion. If this interrupt occurs in STOP mode, it generates Wake-Up signal, having interval (several ms), for Exiting STOP MODE. Note: Do not use zero value(0x0000)	00ff

NOTE: Before ADC conversion, Touch screen uses X-tal clock.

During ADC conversion PCLK (Max. 66MHz) is used.

7.7.1.4 ADC Conversion Data X Register (TSDATXn)

- TSDATX0, R, Address = 0xE170_000C
- TSDATX1, R, Address = 0xE170_100C

TSDATXn	Bit	Description	Initial State
UPDOWN	[15]	Up or Down state of stylus pen at Waiting for Interrupt Mode. 0 = Pen down state. 1 = Pen up state.	-
AUTO_PST_VAL	[14]	Monitoring value of AUTO_PST field in TSCONn register. Read only. 0 = Normal ADC conversion. 1 = Sequencing measurement of X-position, Y-position.	-
XY_PST_VAL	[13:12]	Monitoring value of XY_PST field in TSCONn register. Read only. 00 = No operation mode 01 = X-position measurement 10 = Y-position measurement 11 = Waiting for Interrupt Mode	-
XPDATA (Normal ADC)	[11:0]	X-Position conversion data value (includes normal ADC conversion data value) Data value: 0x0 ~ 0xFFFF	-

7.7.1.5 ADC Conversion Data Y Register (TSDATYn)

- TSDATY0, R, Address = 0xE170_0010
- TSDATY1, R, Address = 0xE170_1010

TSDATYn	Bit	Description	Initial State
UPDOWN	[15]	Up or Down state of stylus pen at Waiting for Interrupt Mode. 0 = Pen down state. 1 = Pen up state.	-
AUTO_PST_VAL	[14]	Monitoring value of AUTO_PST field in TSCONn register. Read only. 0 = Normal ADC conversion. 1 = Sequencing measurement of X-position, Y-position.	-
XY_PST_VAL	[13:12]	Monitoring value of XY_PST field in TSCONn register. Read only. 00 = No operation mode 01 = X-position measurement 10 = Y-position measurement 11 = Waiting for Interrupt Mode	-
YPDATA	[11:0]	Y-Position conversion data value Data value: 0x0 ~ 0xFFFF	-

7.7.1.6 Pen Status Register (TSPENSTATn)

- TSPENSTAT0, R/W, Address = 0xE170_0014
- TSPENSTAT1, R/W, Address = 0xE170_1014

TSPENSTATn	Bit	Description	Initial State
TSC_UP	[1]	Pen up interrupt history. (after check, this bit should be cleared manually) 0 = No pen up state. 1 = Pen up interrupt has been occurred.	0
TSC_DN	[0]	Pen down interrupt history. (after check, this bit should be cleared manually) 0 = No pen down state. 1 = Pen down interrupt has been occurred.	0

7.7.1.7 ADC Interrupt Clear Register (CLRINTADCn)

- CLRINTADC0, W, Address = 0xE170_0018
- CLRINTADC1, W, Address = 0xE170_1018

These registers are used to clear the interrupts. Interrupt service routine is responsible to clear interrupts after the interrupt service is completed. Writing any values on this register will clear up the relevant interrupts asserted. When it is read, undefined value will be returned.

CLRINTADCn	Bit	Description	Initial State
INTADCCLR	[0]	INT_ADCn interrupt clear. Cleared if any value is written.	-

7.7.1.8 ADC Channel Mux Register (ADCMUX, R/W, Address = 0xE170_001C)

ADCMUX	Bit	Description	Initial State
SEL_MUX	[3:0]	Analog input channel select 0000 = AIN 0 0001 = AIN 1 0010 = AIN 2 (YM0) 0011 = AIN 3 (YP0) 0100 = AIN 4 (XM0) 0101 = AIN 5 (XP0) 0110 = AIN 6 (YM1) 0111 = AIN 7 (YP1) 1000 = AIN 8 (XM1) 1001 = AIN 9 (XP1)	0

NOTE:

1. When touch screen is not used, the touch screen ports (AIN2 ~ AIN9) can be used as analog input ports for ADC.
2. SEL_MUX value is invalid when TSADC is set as 1) separate X/Y position conversion mode or 2) auto (sequential) X/Y position conversion mode.

7.7.1.9 Pen Interrupt Clear Register (CLRINTPENn)

- CLRINTPEN0, W, Address = 0xE170_0020
- CLRINTPEN1, W, Address = 0xE170_1020

CLRINTPENn	Bit	Description	Initial State
INTPENCLR	[0]	INT_PENn interrupt clear. Cleared if any value is written.	-

8 KEYPAD INTERFACE

8.1 OVERVIEW OF KEYPAD INTERFACE

The Key Pad Interface block in S5PV210 facilitates communication with external keypad devices. The ports multiplexed with GPIO ports provide up to 14 rows and 8 columns. You can use keypad interface on port0 and port1. port0 is mapped for 8x8 key interface and port1 for 14x8. You can also make your own mapping that can be mix of port0 and port1. The events of key press or key release are delivered to the CPU by an interrupt. If one of the interrupt from row lines occurs, the software must scan the column lines using the proper procedure to detect one or multiple key press or release.

It provides interrupt status register bits at the time of key pressed or key released or both cases (when two interrupt conditions are enabled). To prevent the switching noises, keypad interface comprise of internal debouncing filter.

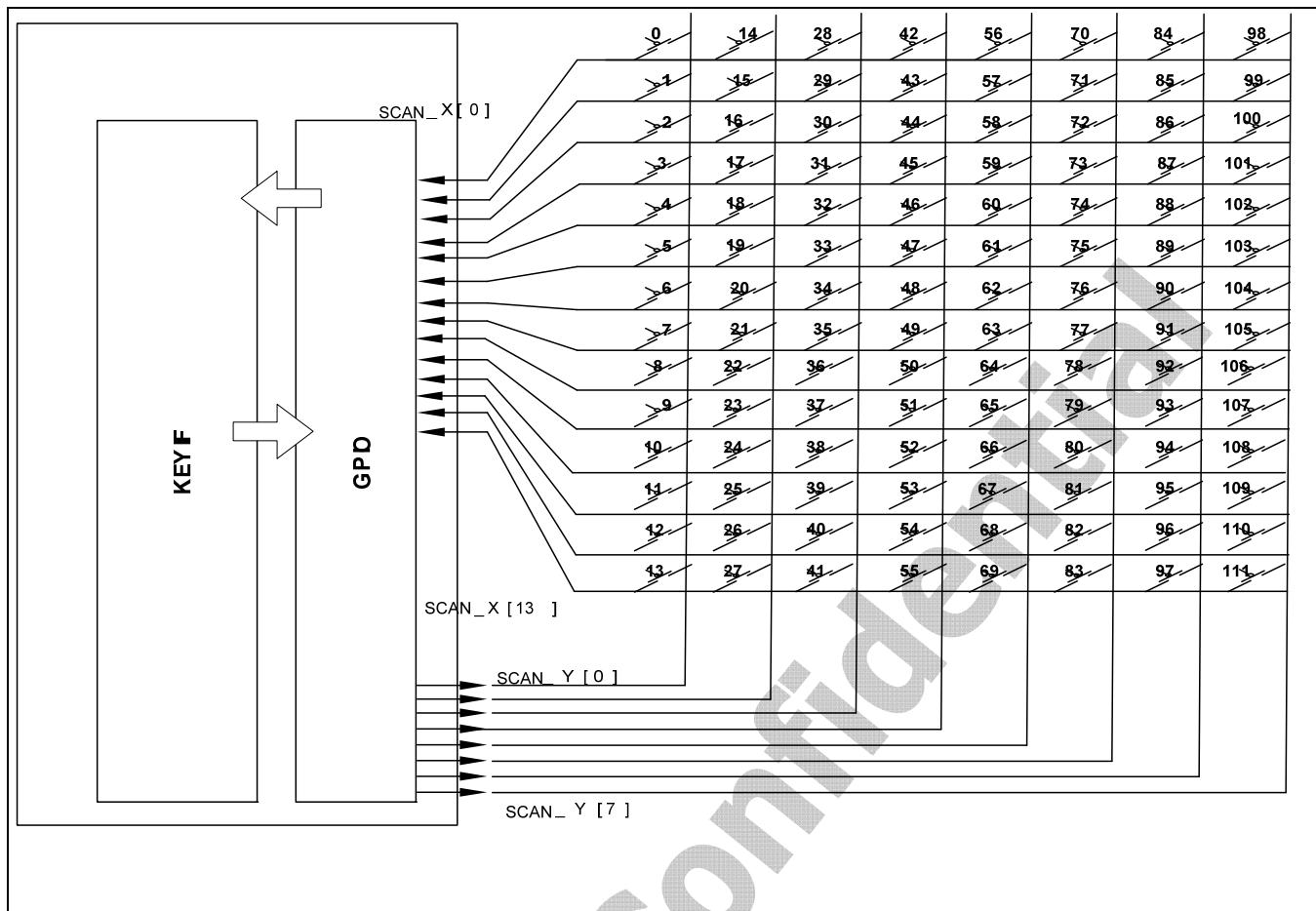


Figure 8-1 Key Matrix Interface External Connection Guide

8.2 DEBOUNCING FILTER

The debouncing filter is supported for keypad interrupt of any key input. The filtering width is approximately 62.5usec ("FCLK" two-clock, when the FCLK is 32kHz). The keypad interrupt (key pressed or key released) to the CPU is an ANDed signal of the all row input lines after filtering.

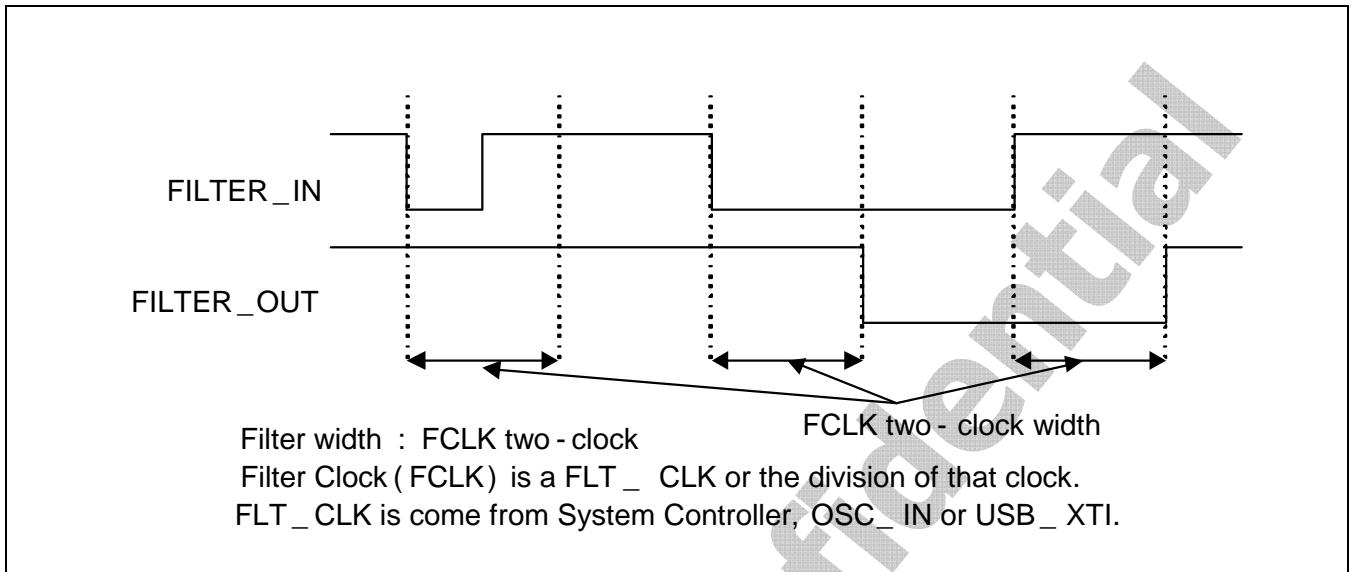


Figure 8-2 Internal Debouncing Filter Operation

8.3 FILTER CLOCK

KEYPAD interface debouncing filter clock (FCLK) is divided from FLT_CLK, that is OSC_IN. User can set compare value for 10-bit up-counter (KEYIFFC). When filter enable bit(FC_EN) is HIGH, filter clock divider is ON. The frequency of FCLK is frequency of FLT_CLK / ((KEYIFFC + 1) x 2). On the contrary, if FC_EN is Low, filter clock divider does not divide FLT_CLK.

8.4 WAKEUP SOURCE

KEYPAD inputs using Port0 can be used as a wakeup source. When the Key input is used for wakeup source from IDLE, STOP or SLEEP mode, KEYPAD interface register setting is not required. GPIO register setting (GPH2CON, GPH3CON) for KEYPAD interface and SYSCON register (PWR_CFG) for masking are required for wakeup. Therefore, to use 14x8 KEYIF which can be used by wakeup source, you must input mix rows of port0 and port1. For example, port0 for ROW[0:7], port1 for ROW[8:13]. In this case, the only keys using port0 can be wakeup source.

8.5 KEYPAD SCANNING PROCEDURE

At initial state, all column lines (outputs) are low level. But column data output tri-state enable bits are all high, so, when the tri-state enable mode is not used, these bits should be written to zeros. If state is no key pressed, all row lines (inputs) are high (used pull-up pads). If any key is pressed, the corresponding row and column lines are shortened together and a low level is driven on the corresponding row line, generating a keypad interrupt. The CPU (software) outputs a LOW on one column line and Hi-Z on the others by setting KEYIFCOLEN and KEYIFCOL fields in KEYIFCOL register. Each write time, the CPU reads the value of the KEYIFROW register and detects if one key of the corresponding column line is pressed. Because the KEYIF has pull-up PAD, each KEYIFROW bits will be read as HIGH, except pressed ROW bit. When the scanning procedure ends, the pressed key (one or more) can be detected.

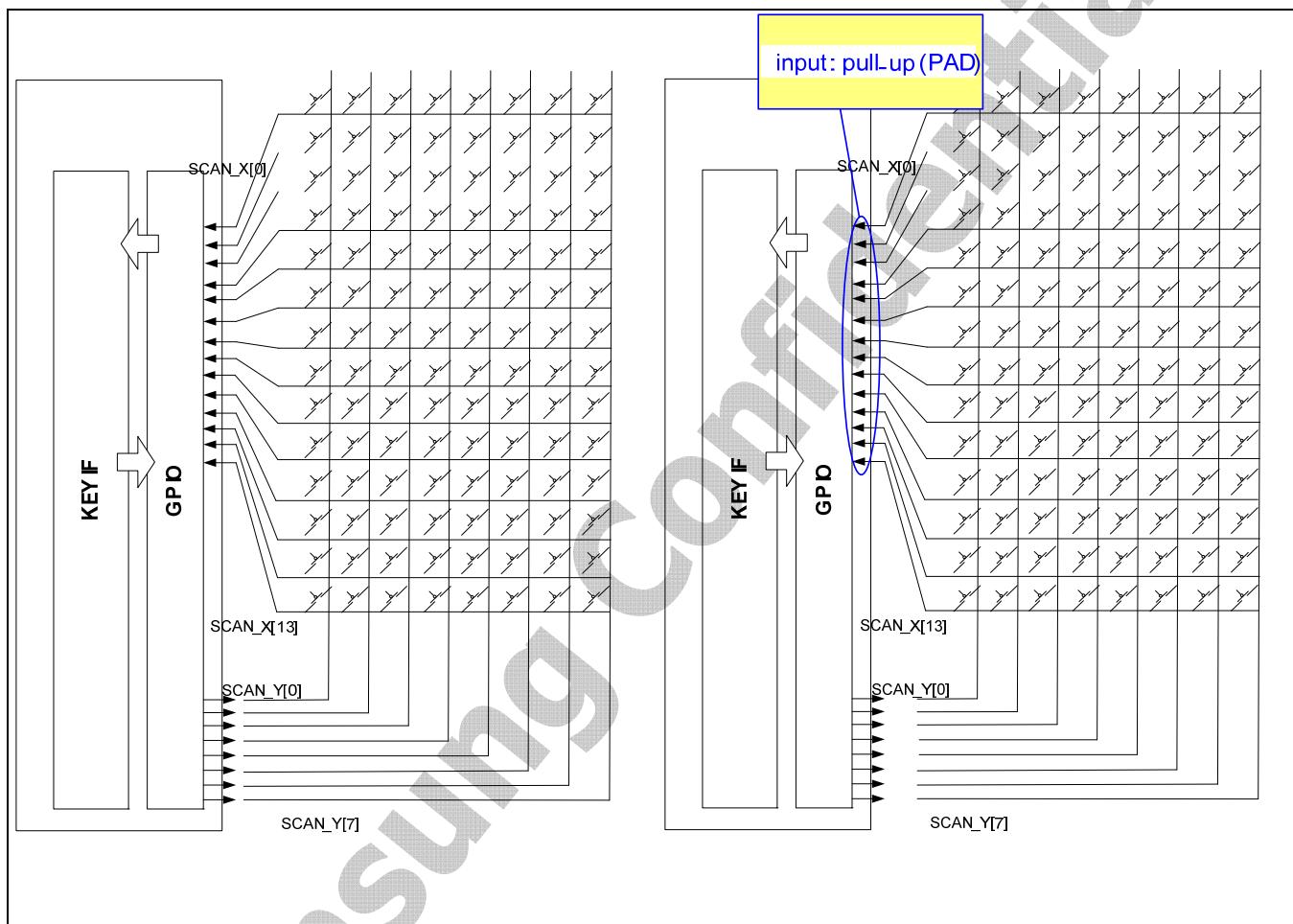


Figure 8-3 Keypad Scanning Procedure

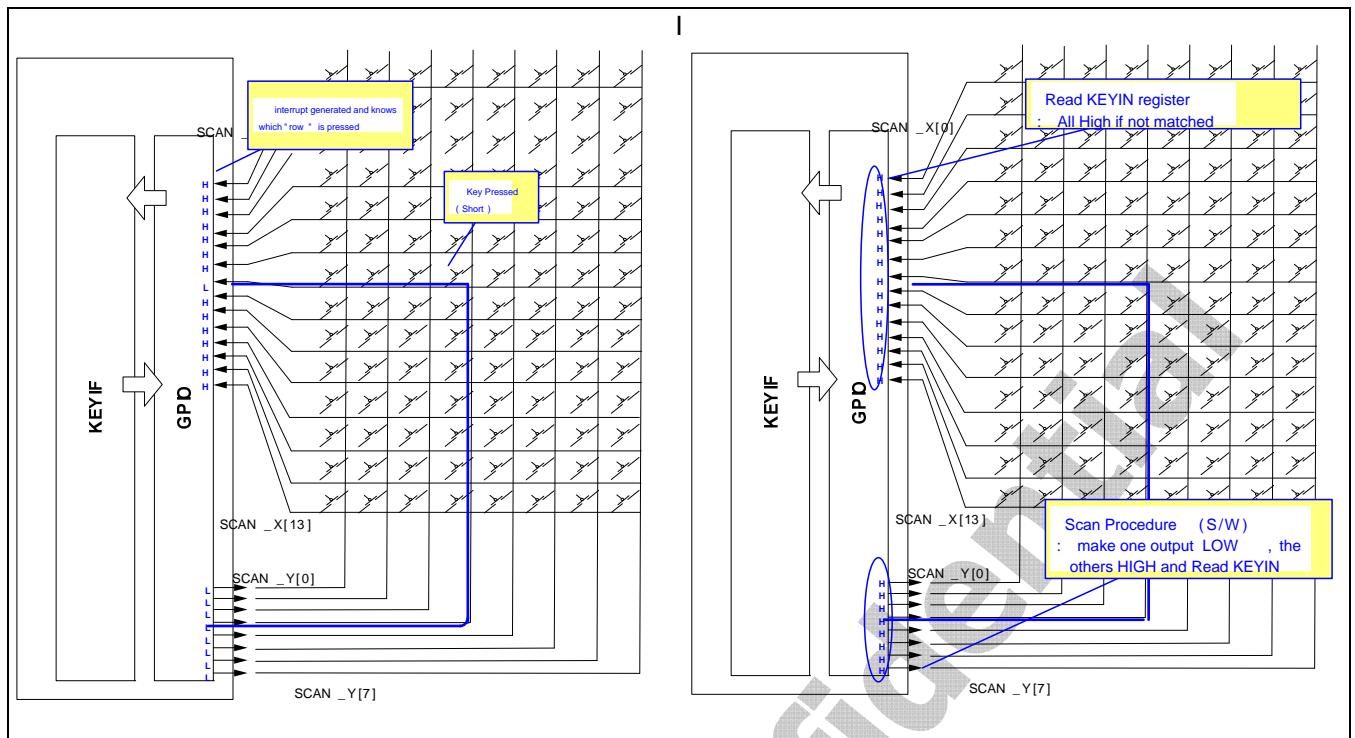


Figure 8-4 Keypad Scanning Procedure II

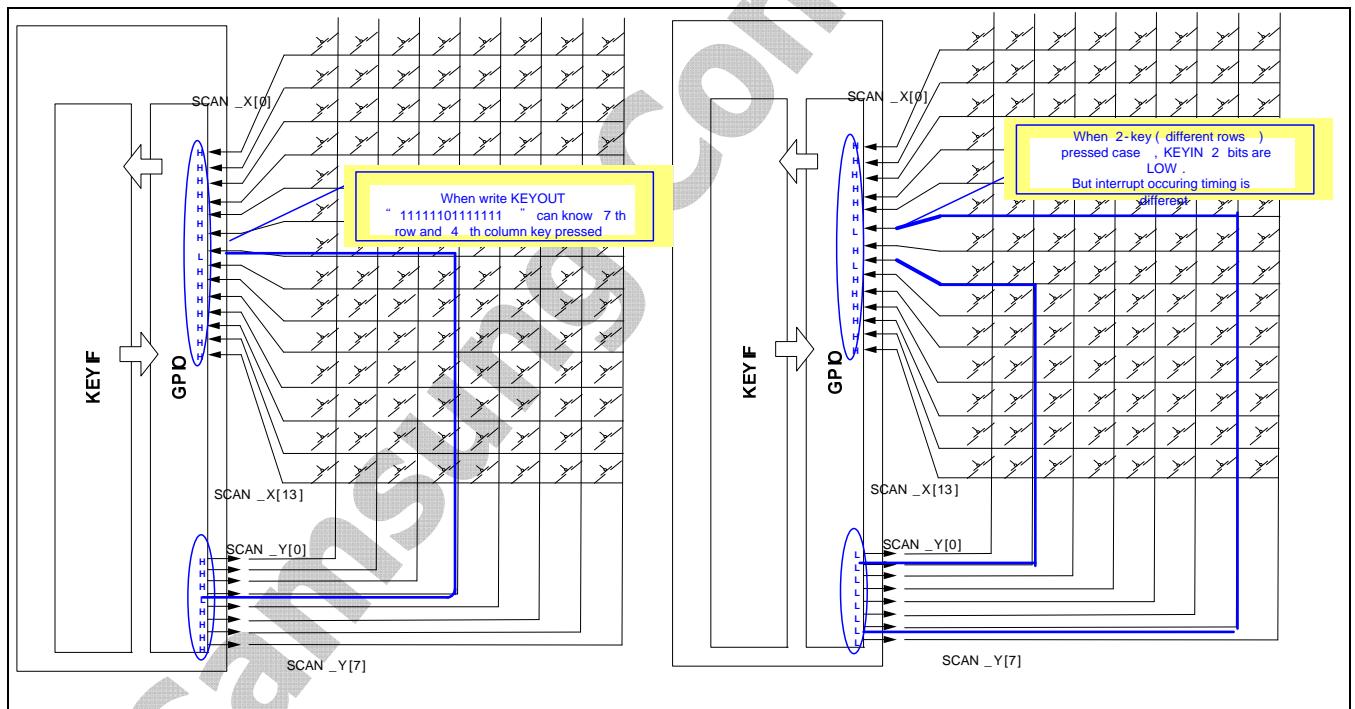


Figure 8-5 Keypad Scanning Procedure III

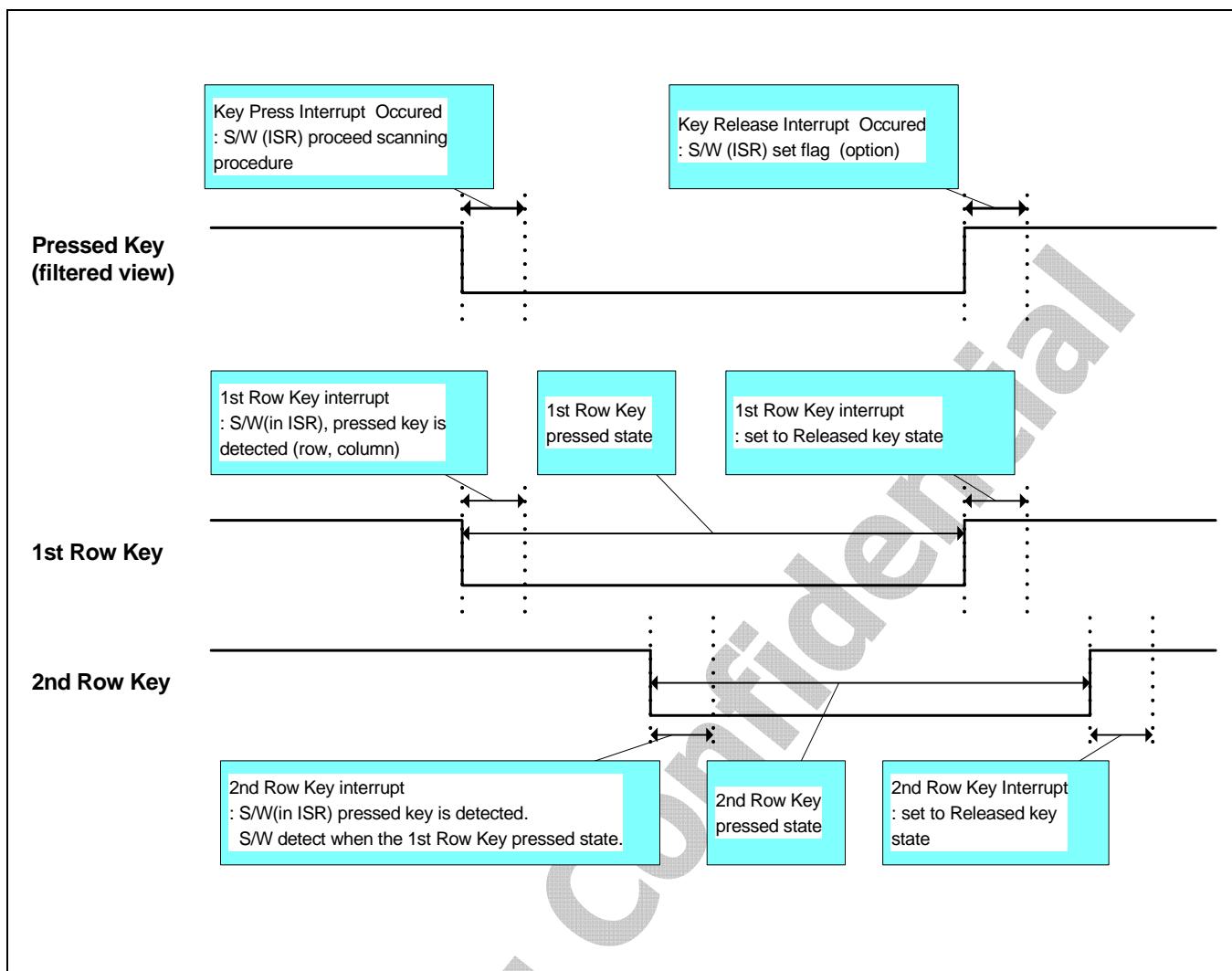


Figure 8-6 Keypad Scanning Procedure when the two-key Pressed with Different Row

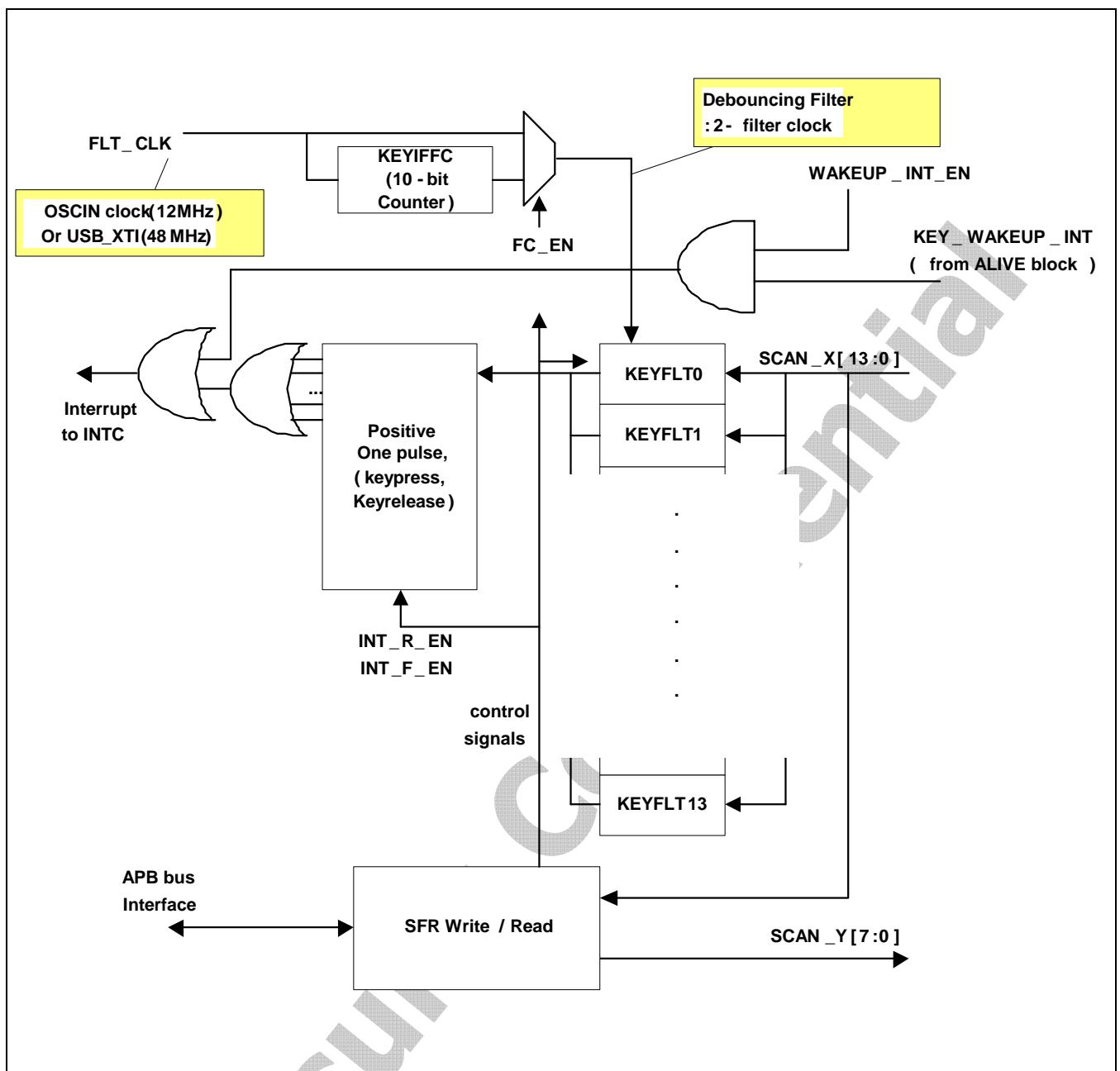


Figure 8-7 Keypad I/F Block Diagram

8.6 I/O DESCRIPTION

Table 8-1 Keypad interface I/O Description

Signal	I/O	Description	Pad		Type
			Port0	Port1	
ROW_IN[13]	I	KEYPAD Interface Row[13] Data		XmsmADVN (GPJ4[4])	muxed
ROW_IN[12]	I	KEYPAD Interface Row[12] Data		XmsmIRQn (GPJ4[3])	muxed
ROW_IN[11]	I	KEYPAD Interface Row[11] Data		XmsmRn (GPJ4[2])	muxed
ROW_IN[10]	I	KEYPAD Interface Row[10] Data		XmsmWEn (GPJ4[1])	muxed
ROW_IN[9]	I	KEYPAD Interface Row[9] Data		XmsmCSn (GPJ4[0])	muxed
ROW_IN[8]	I	KEYPAD Interface Row[8] Data		XmsmDATA[15] (GPJ3[7])	muxed
ROW_IN[7]	I	KEYPAD Interface Row[7] Data	XEINT[31] (GPH3[7])	XmsmDATA[14] (GPJ3[6])	muxed
ROW_IN[6]	I	KEYPAD Interface Row[6] Data	XEINT[30] (GPH3[6])	XmsmDATA[13] (GPJ3[5])	muxed
ROW_IN[5]	I	KEYPAD Interface Row[5] Data	XEINT[29] (GPH3[5])	XmsmDATA[12] (GPJ3[4])	muxed
ROW_IN[4]	I	KEYPAD Interface Row[4] Data	XEINT[28] (GPH3[4])	XmsmDATA[11] (GPJ3[3])	muxed
ROW_IN[3]	I	KEYPAD Interface Row[3] Data	XEINT[27] (GPH3[3])	XmsmDATA[10] (GPJ3[2])	muxed
ROW_IN[2]	I	KEYPAD Interface Row[2] Data	XEINT[26] (GPH3[2])	XmsmDATA[9] (GPJ3[1])	muxed
ROW_IN[1]	I	KEYPAD Interface Row[1] Data	XEINT[25] (GPH3[1])	XmsmDATA[8] (GPJ3[0])	muxed
ROW_IN[0]	I	KEYPAD Interface Row[0] Data	XEINT[24] (GPH3[0])	XmsmDATA[7] (GPJ2[0])	muxed
COL_OUT[7]	O	KEYPAD Interface Column[7] Data	XEINT[23] (GPH2[7])	XmsmDATA[6] (GPJ2[0])	muxed
COL_OUT [6]	O	KEYPAD Interface Column[6] Data	XEINT[22] (GPH2[6])	XmsmDATA[5] (GPJ2[0])	muxed
COL_OUT [5]	O	KEYPAD Interface Column[5] Data	XEINT[21] (GPH2[5])	XmsmDATA[4] (GPJ2[0])	muxed
COL_OUT [4]	O	KEYPAD Interface Column[4] Data	XEINT[20] (GPH2[4])	XmsmDATA[3] (GPJ2[0])	muxed

Signal	I/O	Description	Pad		Type
			Port0	Port1	
COL_OUT [3]	O	KEYPAD Interface Column[3] Data	XEINT[19] (GPH2[3])	XmsmDATA[2] (GPJ2[0])	Muxed
COL_OUT [2]	O	KEYPAD Interface Column[2] Data	XEINT[18] (GPH2[2])	XmsmDATA[1] (GPJ2[0])	muxed
COL_OUT [1]	O	KEYPAD Interface Column[1] Data	XEINT[17] (GPH2[1])	XmsmDATA[0] (GPJ2[0])	muxed
COL_OUT [0]	O	KEYPAD Interface Column[0] Data	XEINT[16] (GPH2[0])	XmsmADDR[13] (GPJ2[0])	muxed

8.7 REGISTER DESCRIPTION

8.7.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
KEYIFCON	0xE160_0000	R/W	Specifies the KEYPAD interface control register	0x00000000
KEYIFSTSCLR	0xE160_0004	R/W	Specifies the KEYPAD interface status and clear register	0x00000000
KEYIFCOL	0xE160_0008	R/W	Specifies the KEYPAD interface column data output register	0x0000FF00
KEYIFROW	0xE160_000C	R	Specifies the KEYPAD interface row data input register	Reflects input ports
KEYIFFC	0xE160_0010	R/W	Specifies the KEYPAD interface debouncing filter clock division register	0x00000000

8.7.1.1 KEYPAD Interface Control Registers (KEYIFCON, R/W, Address = 0xE160_0000)

KEYIFCON	Bit	Description	Initial State
Reserved	[31:5]	Reserved for future use	-
WAKEUPEN	[4]	KEYPAD input Stop / Idle mode wakeup enable. Wakeup signal is to System Controller. 0 = Disables 1 = Key input Low Level (while key-pressed) wakeup	1'b0
FC_EN	[3]	10-bit counter (for debouncing digital filter clock) enable 0 = Disables: No use division counter 1 = Enables: use division counter	1'b0
DF_EN	[2]	KEYPAD input port debouncing filter enable 0 = Disables 1 = Enables	1'b0
INT_R_EN	[1]	KEYPAD input port rising edge (key-released) interrupt 0 = Disables 1 = Enables	1'b0
INT_F_EN	[0]	KEYPAD input port falling edge (key-pressed) interrupt 0 = Disables 1 = Enables	1'b0

NOTE: Both edge interrupt is selected when both INT_F_EN and INT_R_EN are set.

8.7.1.2 KEYPAD Interrupt Status and Clear Register (KEYIFSTSCLR, R/W, Address = 0xE160_0004)

KEYIFSTSCLR	Bit	Description	Initial State
R_INT	[29:16]	KEYPAD input "release" interrupt (rising edge) status(read) and clear(write) Read: 1 = Released interrupt occurred 0 = Not occurred Write: Released interrupt is cleared when write '1' The R_INT[13:0] indicate that each key pressed from 0 to 13 has a dedicated interrupt from R_INT[16] to R_INT[29]	14'b0
P_INT	[13:0]	KEYPAD input "press" interrupt (falling edge) status(read) and clear(write) Read: 1 = Pressed interrupt occurred 0 = Not occurred Write: Pressed interrupt is cleared when write '1' The P_INT[13:0] indicate that each key released from 0 to 13 has a dedicated interrupt from P_INT[0] to P_INT[13]	14'b0

NOTE: Keypad wakeup interrupt is also cleared when the write access to the KEYIFSTSCLR.

8.7.1.3 KEYPAD Interface Column Data Output Register (KEYIFCOL, R/W, Address = 0xE160_0008)

KEYIFCOL	Bit	Description	Initial State
Reserved	[31:16]	Reserved for future use	-
KEYIFCOLEN	[15:8]	KEYPAD interface column data output tri-state enable register Each bit is for each KEYIFCOL bit. 0 = Output pad tri-state buffer enable(Normal output), 1 = Output pad Tri-state buffer disable(High-Z output) (@ reset)	8'b1111_1111
KEYIFCOL	[7:0]	KEYPAD interface column data output register	8'b0

8.7.1.4 KEYPAD Interface Row Data Input Register (KEYIFROW, R, Address = 0xE160_000C)

KEYIFROW	Bit	Description	Initial State
Reserved	[31:16]	Reserved for future use	-
KEYIFROW	[13:0]	KEYPAD interface row data input register (read only) This register values from input ports are not filtered data.	Reflects input ports

8.7.1.5 KEYPAD Interface Debouncing Filter Clock Division Register (KEYIFFC, R/W, Address = 0xE160_0010)

KEYIFFC	Bit	Description	Initial State
Reserved	[31:10]	Reserved for future use	-
KEYIFFC	[9:0]	KEYPAD interface debouncing filter clock division register. User can set compare value for 10-bit up-counter. This register value means when FC_EN bit is HIGH. $FCLK = FLT_CLK / (KEYIFFC[9:0] + 1)$ (FLT_CLK is from FINpll)	10'b0

Section 11

SECURITY

Table of Contents

1	Security-system.....	1-1
1.1	Overview of Security-system	1-1
2	Advanced Crypto Engine.....	2-1
2.1	Overview of Advanced Crypto Engine	2-1
2.1.1	KEY Features of SSS	2-3
2.2	Functional Description OF SSS	2-4
2.2.1	CPU Mode	2-4
2.2.2	FIFO Mode.....	2-4
2.2.3	Byte Swapping Options	2-8
2.3	Register Description.....	2-11
2.3.1	Register Map	2-11
2.3.2	TDES Control (TDES_OUTPUT_0, R, Address = 0xEA00_5038).....	2-33

List of Figures

Figure Number	Title	Page Number
Figure 2-1	Block Diagram of SSS	2-2
Figure 2-2	DES or 3DES Only Data Flow	2-4
Figure 2-3	AES and Hash parallel data flow	2-5
Figure 2-4	Data flow of AES and Hash with Shared Input.....	2-5
Figure 2-5	Data Flow of Hashing the Output of AES	2-6
Figure 2-6	FIFO and FIFO Interconnections	2-6
Figure 2-7	Interrupt Controller Scheme for one Interrupt Signal.....	2-7
Figure 2-8	AES Byte Swapping Scheme	2-8
Figure 2-9	DES Byte Swapping Scheme	2-9
Figure 2-10	Hash Byte Swapping Scheme	2-10
Figure 2-11	PKA Byte Swapping Scheme	2-10

List of Tables

Table Number	Title	Page Number
Table 1-1	Security Features of S5PV210.....	1-1

Samsung Confidential

1 SECURITY-SYSTEM

1.1 OVERVIEW OF SECURITY-SYSTEM

S5PV210 supports following security features

- Secure Booting
- Security Engines

Table 1-1 shows the detail security features of S5PV210.

S5PV210 provides on chip 64KB boot ROM, 96KB RAM and the 128-bit e-Fuse for secure booting. For more information, refer to 'Section 02.06 Booting sequence'.

Additionally, users can design security solution more conveniently using H/W security engine such as DES/ 3DES, AES, SHA-1, PRNG and PKA.

Table 1-1 Security Features of S5PV210

	Description
Secure Booting	On chip 64KB secure boot ROM On chip 96KB secure SRAM 128-bit e-fuse ROM for secure context
Security Engines	DES/ 3DES, AES, SHA-1, PRNG and PKA

2 ADVANCED CRYPTO ENGINE

2.1 OVERVIEW OF ADVANCED CRYPTO ENGINE

Security subsystem (SSS) represents a small system with internal buses and small security IPs that should be attached to a chip as an IP. The security IPs in SSS can process the independent security function.

SSS comprises of the following internal components:

- AES
- DES and 3DES
- SHA-1, MD5, HMAC, and PRNG
- Public Key Accelerator (PKA)
- Feed Controller (FeedCtrl)

FeedCtrl comprises of the following components:

- Block Cipher Receiving DMA (BRDMA)
- Block Cipher Transmission DMA (BTDMA)
- Hash Receiving DMA (HRDMA)
- PKA Bi-directional DMA (PKDMA)
- FIFO and FIFO Interconnections
- Interrupt Controller
- FIFO Controller

SSS comprises of the following external interfaces:

- One bus slave port (for SFR setting)
- Four bus master ports (for DMA operations)
- Two interrupts: MA interrupt (to notify the end of DMA operations) and Hash interrupt (to notify the end of Hash or PRNG operations)

Each security IP can be accessed through two access modes, namely:

- CPU mode
 - For AES, DES, 3DES, SHA-1, MD5, and PKA
 - Every input and output data should be carried out by the host processor.
- FIFO mode
 - For AES, DES, 3DES, SHA-1, and MD5
 - DMA supplies input data to each IP through FIFO.
 - DMA drains output data from each IP (except SHA-1 and MD5) through FIFO.
 - Block ciphers and hashes can share input data.
 - Output data of block ciphers can be used as input of the hash.

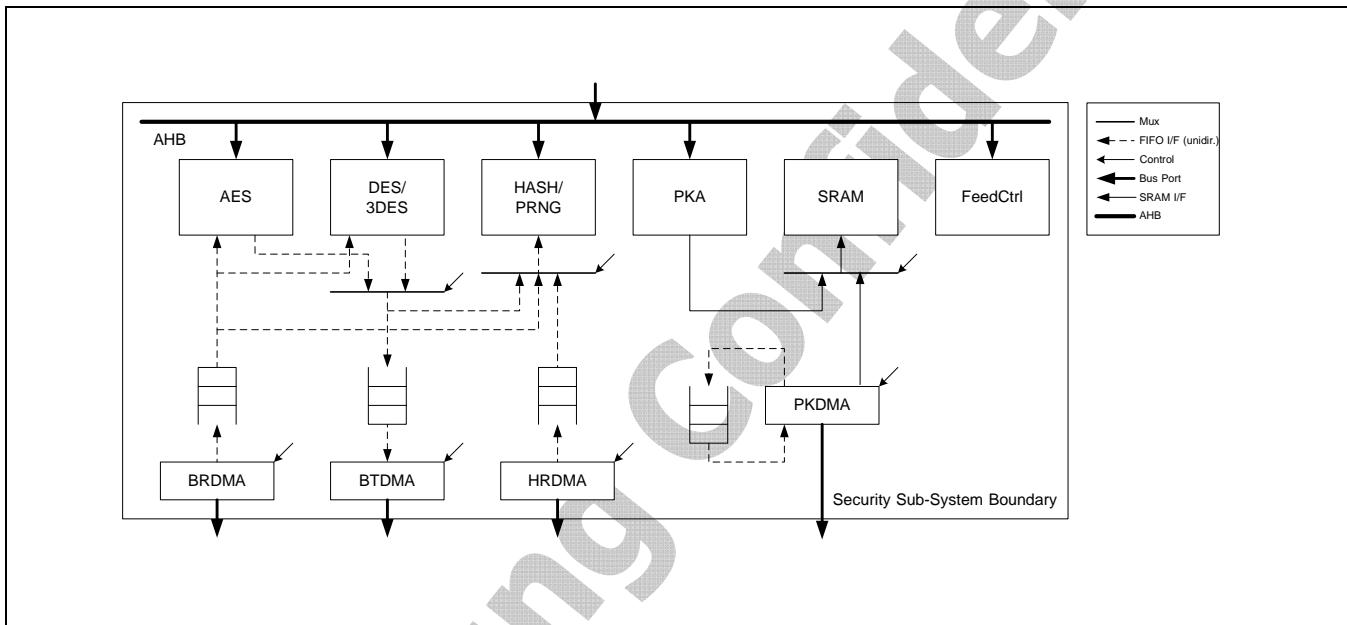


Figure 2-1 Block Diagram of SSS

2.1.1 KEY FEATURES OF SSS

The key features of SSS include:

- AES (ECB, CBC, and CTR modes)
- DES (ECB and CBC modes)
- 3DES (ECB, CBC, EDE, and EEE modes)
- SHA-1 (with hardware padding) and SHA-1 HMAC
- MD5 (w/ hardware padding) and MD5 HMAC
- Pseudo Random Number Generator (PRNG)
- Public Key Accelerator (PKA)
- DMA Support for AES, DES, 3DES, SHA-1, MD5, and PKA
- Block Ciphers combined with Hashing
 - Concurrent AES/ DES and SHA1/ MD5
 - SHA-1/ MD5 after AES/ DES

Samsung Confidential

2.2 FUNCTIONAL DESCRIPTION OF SSS

2.2.1 CPU MODE

Using the SFR, you can access the full functions of AES, DES, Hash, or PRNG. You can also supply the input data, trigger an operation, and extract the output data.

2.2.2 FIFO MODE

BRDMA supplies input data to block ciphers such as AES or DES, as shown in [Figure 2-2](#). On the other hand, BTDMA receives output data from AES or DES. Only one block (either AES or DES) can use the DMA. The other block that does not occupy the DMA can be used in CPU or buffered CPU modes.

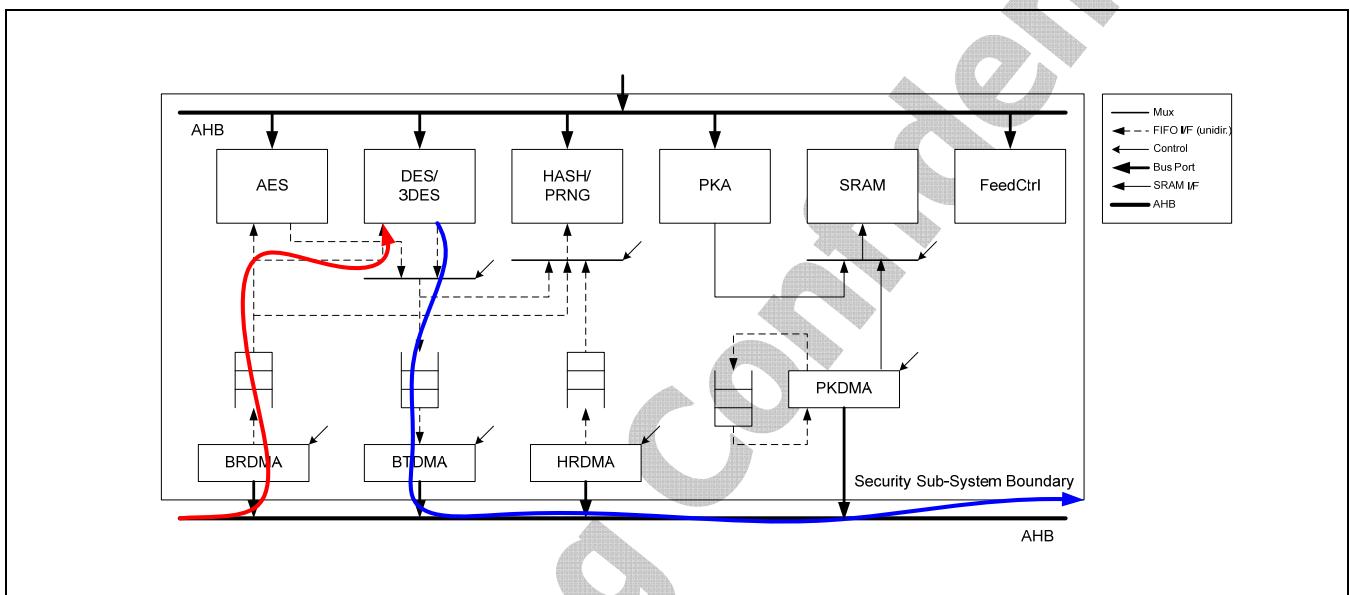


Figure 2-2 DES or 3DES Only Data Flow

[Figure 2-3](#) shows that the AES block uses BRDMA and BTDMA. The hash block uses HRDMA, which represents Hash Receiving DMA. HRDMA can work independently of BRDMA or BTDMA. In this case, the hash processes different data stream than block ciphers.

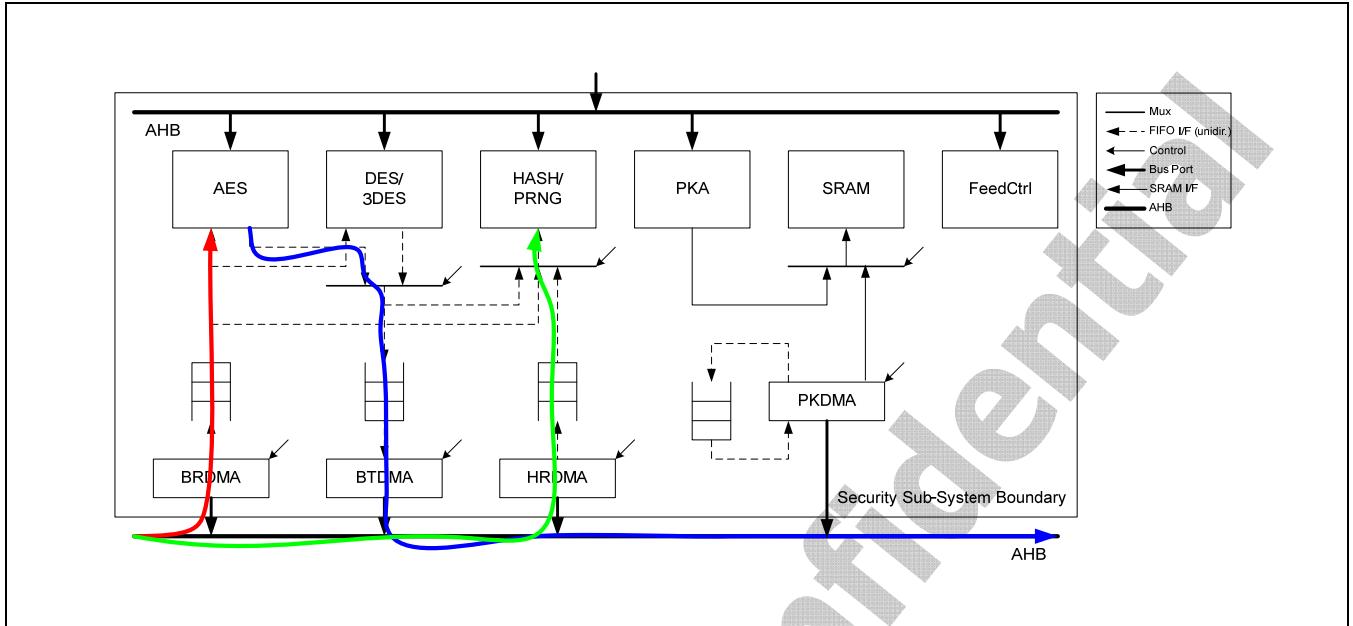


Figure 2-3 AES and Hash parallel data flow

[Figure 2-4](#) shows the configuration of AES and hash with shared input data from external memory.

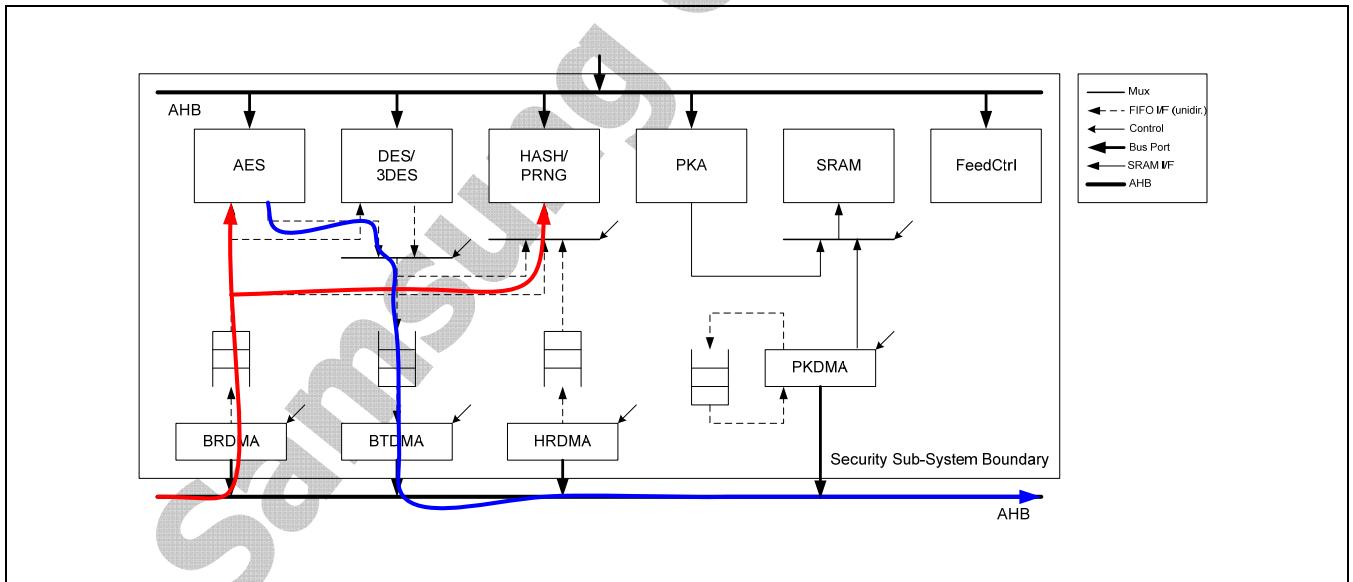


Figure 2-4 Data flow of AES and Hash with Shared Input

[Figure 2-5](#) shows the configuration of hash when it processes the output of AES.

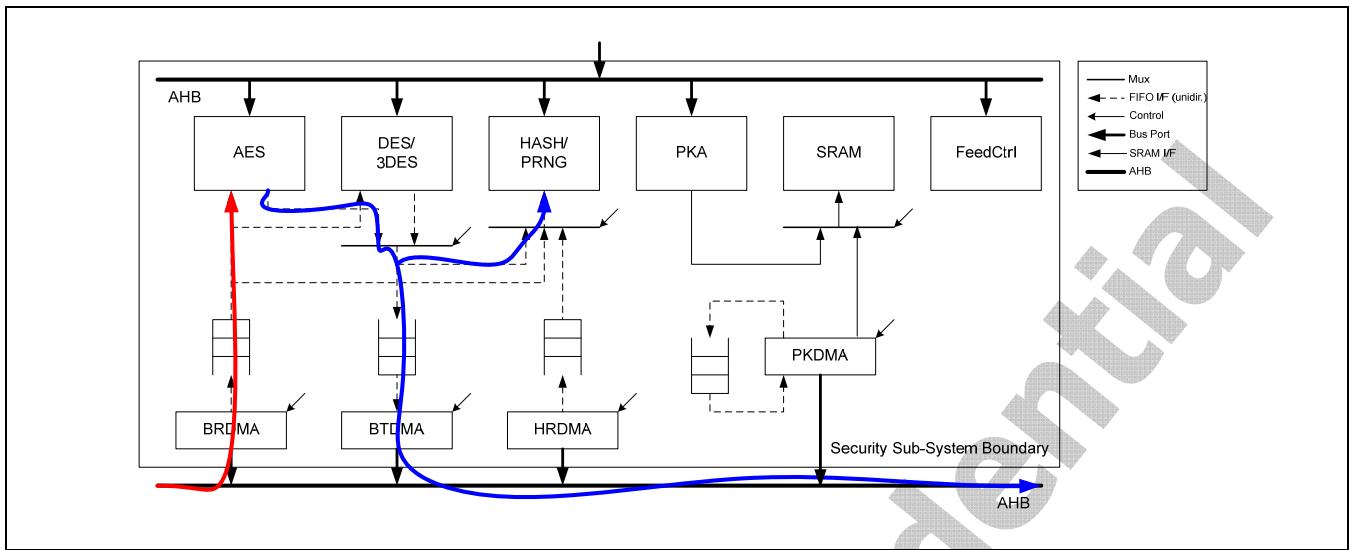


Figure 2-5 Data Flow of Hashing the Output of AES

For the above two cases, HRDMA cannot be used.

2.2.2.1 FIFO Configuration

The FCFIFOCTRL register affects FIFO configuration. The DESSEL bit of FCFIFOCTRL selects between DES and AES. Also, the HASHINSEL bits select hash input data from three possible inputs that comes from HRDMA, input of block cipher, and output of block cipher.

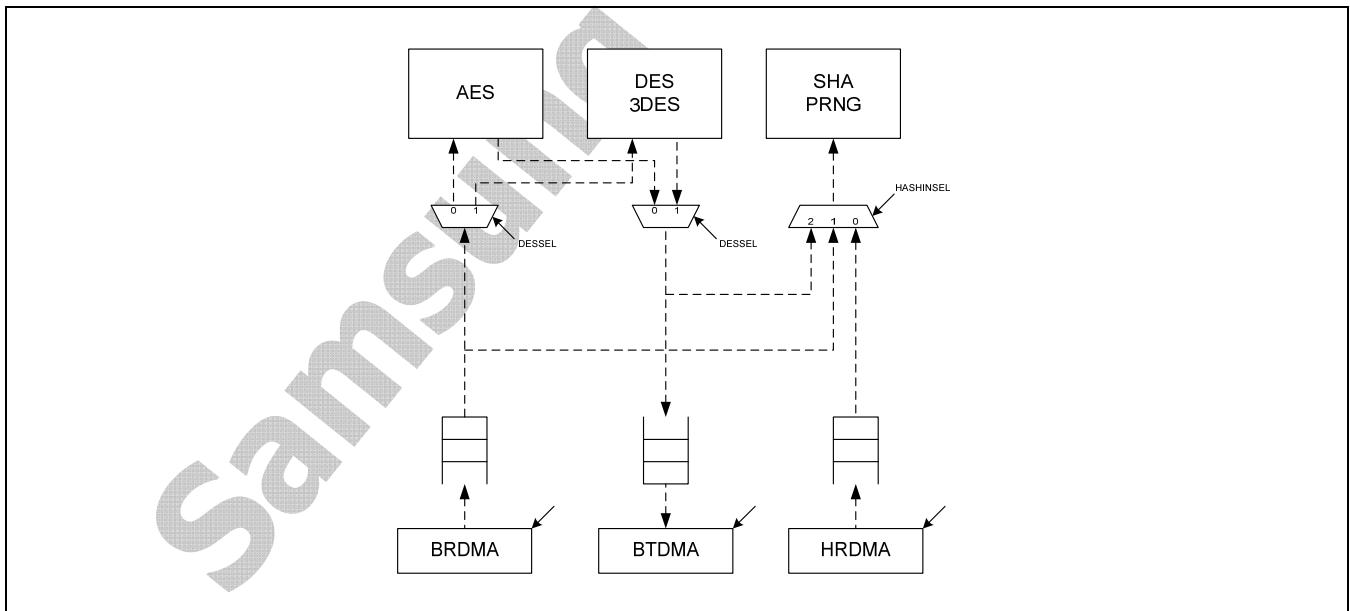


Figure 2-6 FIFO and FIFO Interconnections

2.2.2.2 DMA Configuration

Each DMA has three main parameters, namely:

- STARTADDR (32-bit): Specifies the start address of DMA. The address does not need to be aligned by 32-bit. Its value increases by four after every transaction.
- LENGTH (32-bit): Specifies the block length of DMA. The length need not be aligned by 32-bit. Its value decreases by four after every transaction.
- FLUSH (1-bit): If this bit is high, then data flushes out from FIFO and DMA. After flushing, the start address keeps the stopped address and the length is 0. The flushing state should be released by writing value '0' to this bit.

2.2.2.3 Interrupt Controller (for DMA Interrupt)

[Figure 2-7](#) shows the interrupt controller scheme for one interrupt signal. Each of the four DMA interrupt signals have the following control scheme, that is, each interrupt signal is generated by a DMA in pulse form and latched by the FCINTPEND register to form a level sensitive interrupt signal.

The latched signal is masked by FCINTENSET register in bit-by-bit form. Each bit in FCINTENSET can be set by writing '1' to the corresponding bit in FCINTENSET, and cleared by writing '1' to the corresponding bit in FCINTENCLR.

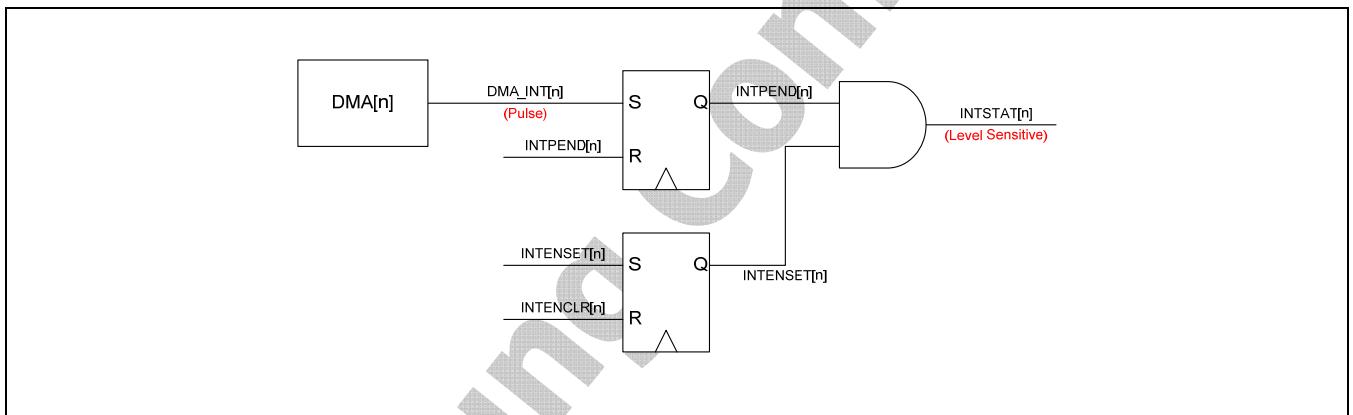


Figure 2-7 Interrupt Controller Scheme for one Interrupt Signal

2.2.3 BYTE SWAPPING OPTIONS

SSS supports byte-swapping options for various data. Byte swapping in this context means byte order reversion in a 32-bit word boundary.

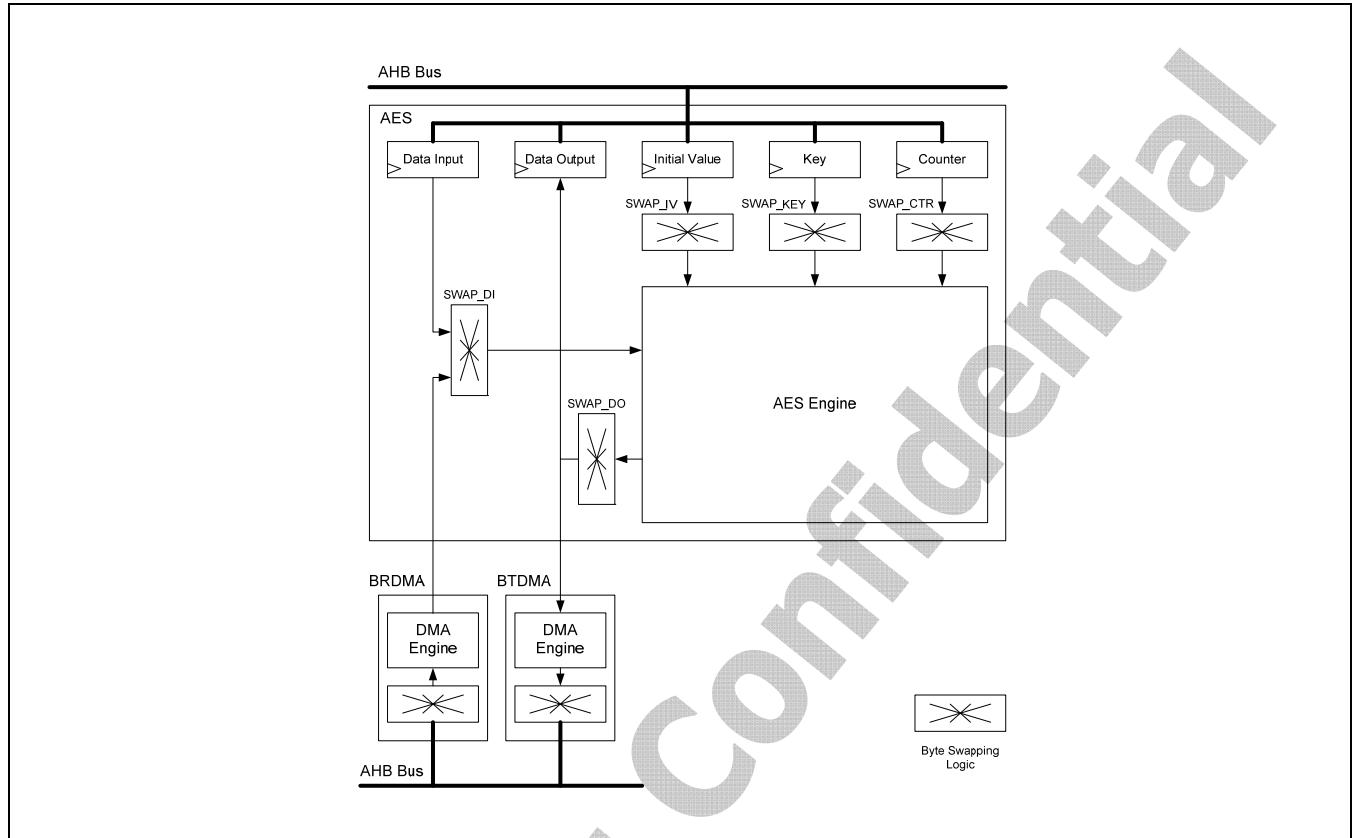


Figure 2-8 AES Byte Swapping Scheme

According to [Figure 2-8](#), AES has five swapping options for every data (be it data input, data output, initial value, key, and counter).

Moreover, all DMA (BRDMA and BTDMA) have their own swapping option. The byte-swapping option of DMA should follow the bus endian.

1. For little endian bus, the DMA should swap data.
2. For big endian bus, the DMA should not swap data.

The only reason why option 2 must be used is that S5PV210 supports little endian case.

In case DES, Hash, and PKA contain different numbers of data, the same scheme will be applied.

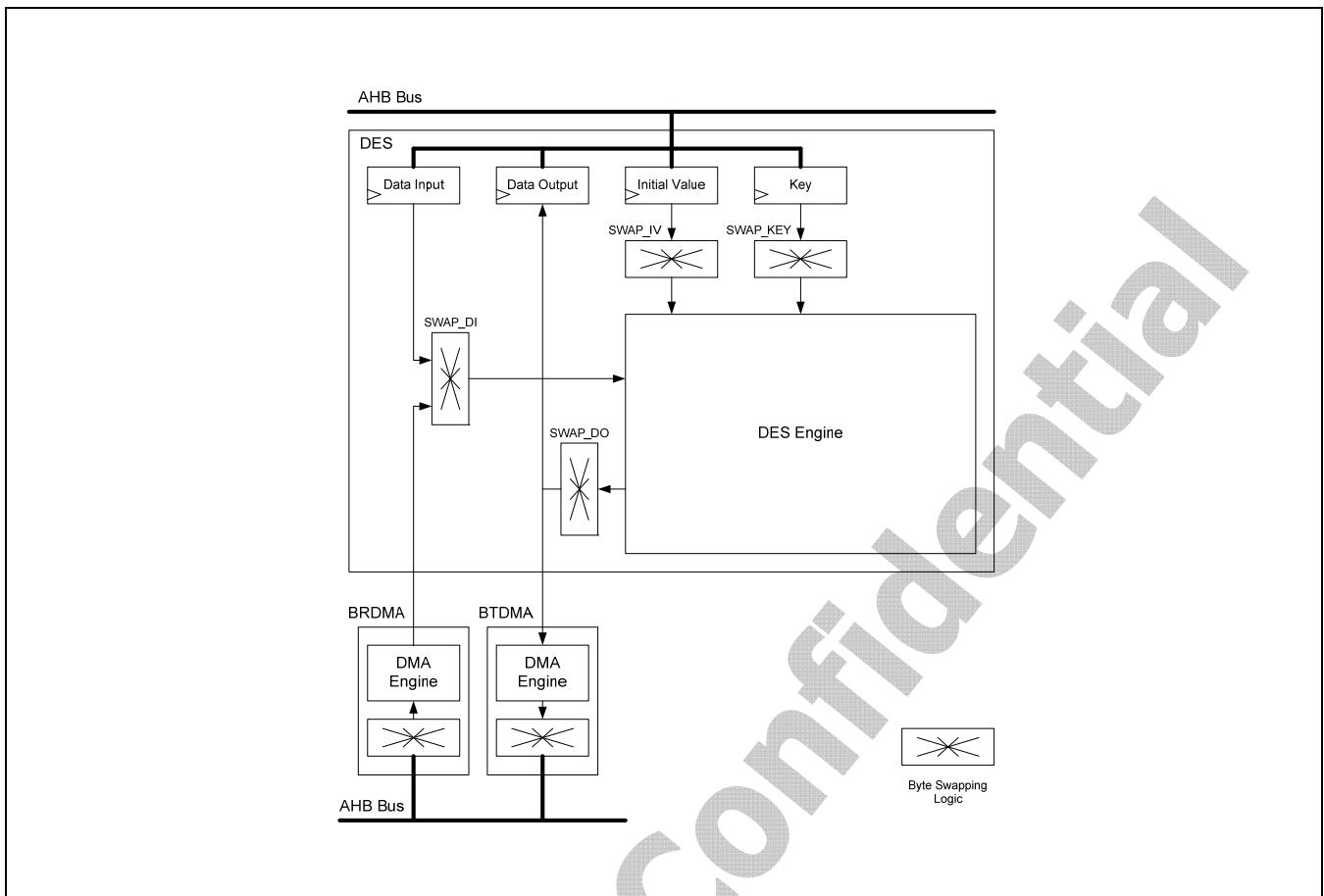


Figure 2-9 DES Byte Swapping Scheme

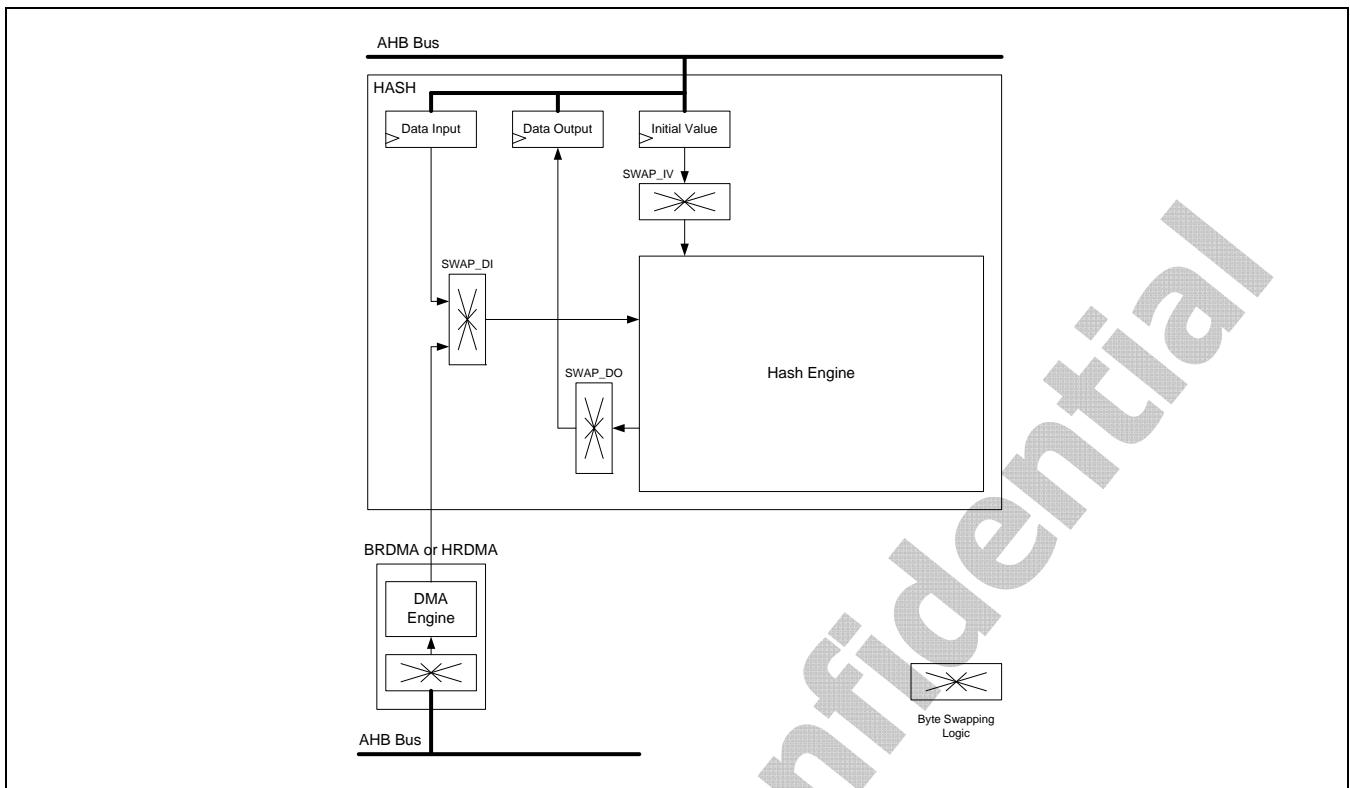


Figure 2-10 Hash Byte Swapping Scheme

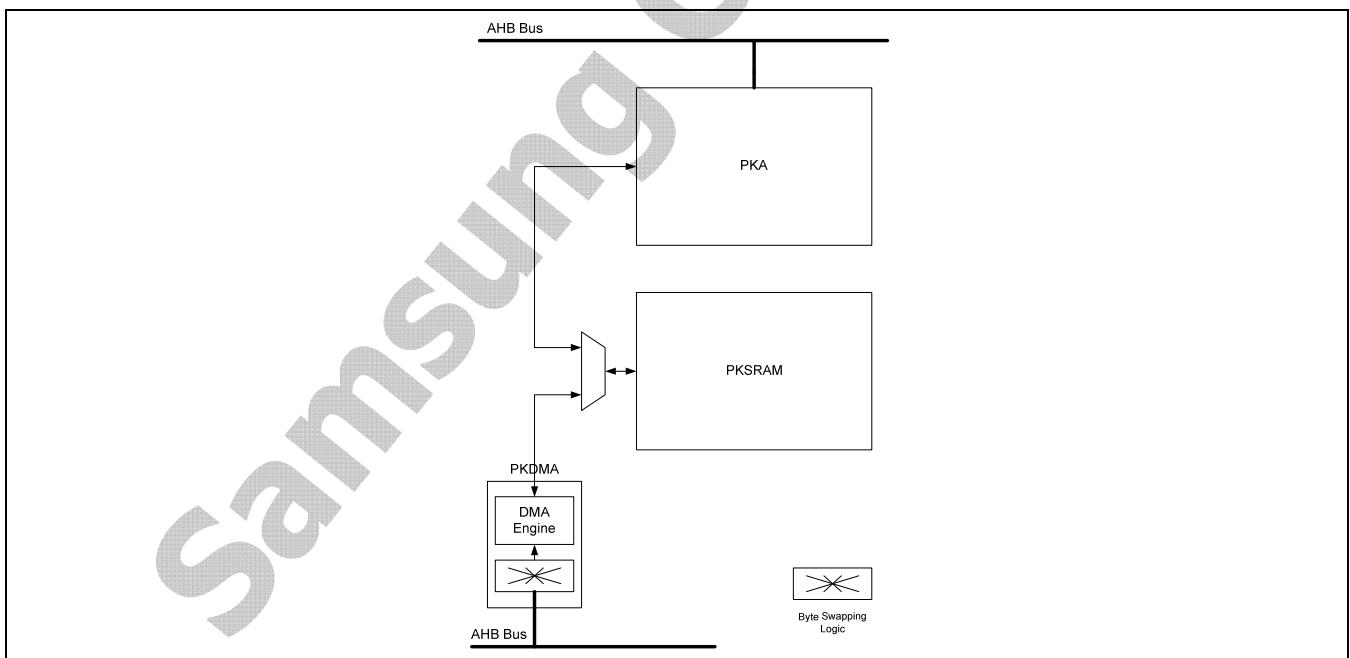


Figure 2-11 PKA Byte Swapping Scheme

2.3 REGISTER DESCRIPTION

2.3.1 REGISTER MAP

Register	Address	R/W	Description	Reset Value
Feed				
FCINTSTAT	0xEA00_0000	R	Specifies the interrupt status of feed control.	0x0000_0000
FCINTENSET	0xEA00_0004	R/W	Specifies the interrupt enable set register of feed control. Value '1' should be written to set the corresponding bit.	0x0000_0000
FCINTENCLR	0xEA00_0008	R/W	Specifies the interrupt enable clear register of feed control. Value '1' should be written to clear the corresponding bit.	0x0000_0000
FCINTPEND	0xEA00_000C	R/W	Specifies the pending interrupts of feed control.	0x0000_0000
FCFIFOSTAT	0xEA00_0010	R	Specifies the FIFO status of feed control.	0x0000_0055
FCFIFOCTRL	0xEA00_0014	R/W	Specifies the FIFO control of feed control.	0x0000_0000
FCBRDMAS	0xEA00_0020	R/W	Specifies the start address of block cipher receiving DMA.	0x0000_0000
FCBRDMAL	0xEA00_0024	R/W	Specifies the length of block cipher receiving DMA.	0x0000_0000
FCBRDMAC	0xEA00_0028	R/W	Specifies the control of block cipher receiving DMA.	0x0000_0000
FCBTDMAS	0xEA00_0030	R/W	Specifies the start address of block cipher transmitting DMA.	0x0000_0000
FCBTDMAL	0xEA00_0034	R/W	Specifies the length of block cipher transmitting DMA.	0x0000_0000
FCBTDMAC	0xEA00_0038	R/W	Specifies the control of block cipher transmitting DMA.	0x0000_0000
FCHRDMAS	0xEA00_0040	R/W	Specifies the start address of hash receiving DMA.	0x0000_0000
FCHRDMAL	0xEA00_0044	R/W	Specifies the length of hash receiving DMA.	0x0000_0000
FCHRDMAC	0xEA00_0048	R/W	Specifies the control of hash receiving DMA.	0x0000_0000
FCPKDMAS	0xEA00_0050	R/W	Specifies the start address of PKA DMA.	0x0000_0000
FCPKDMAL	0xEA00_0054	R/W	Specifies the length of PKA DMA.	0x0000_0000
FCPKDMAC	0xEA00_0058	R/W	Specifies the control of PKA DMA.	0x0000_0000
FCPKDMAO	0xEA00_005C	R/W	Specifies the offset in PKA SRAM.	0x0000_0000
AES				
AES_control	0xEA00_4000	R/W	Specifies the AES control register.	0x0000_0000
AES_status	0xEA00_4004	R/W	Specifies the AES status register.	0x0000_0002

Register	Address	R/W	Description	Reset Value
AES_indata_01	0xEA00_4010	W	Specifies the Input data to be used in encryption/decryption: [127:96].	0x0000_0000
AES_indata_02	0xEA00_4014	W	Specifies the Input data to be used in encryption/decryption: [95:64].	0x0000_0000
AES_indata_03	0xEA00_4018	W	Specifies the Input data to be used in encryption/decryption: [63:32].	0x0000_0000
AES_indata_04	0xEA00_401C	W	Specifies the Input data to be used in encryption/decryption: [31:0].	0x0000_0000
AES_outdata_01	0xEA00_4020	R	Specifies the Output data to be used in encryption/decryption: [127:96].	0x0000_0000
AES_outdata_02	0xEA00_4024	R	Specifies the Output data to be used in encryption/decryption: [95:64].	0x0000_0000
AES_outdata_03	0xEA00_4028	R	Specifies the Output data to be used in encryption/decryption: [63:32].	0x0000_0000
AES_outdata_04	0xEA00_402C	R	Specifies the Output data to be used in encryption/decryption: [31:0].	0x0000_0000
AES_ivdata_01	0xEA00_4030	W	Specifies the Initialization vector to be used in encryption/decryption: [127:96].	0x0000_0000
AES_ivdata_02	0xEA00_4034	W	Specifies the Initialization vector to be used in encryption/decryption: [95:64].	0x0000_0000
AES_ivdata_03	0xEA00_4038	W	Specifies the Initialization vector to be used in encryption/decryption: [63:32].	0x0000_0000
AES_ivdata_04	0xEA00_403C	W	Specifies the Initialization vector to be used in encryption/decryption: [31:0].	0x0000_0000
AES_cntdata_01	0xEA00_4040	W	Specifies the Counter data to be used in encryption/decryption: [127:96].	0x0000_0000
AES_cntdata_02	0xEA00_4044	W	Specifies the Counter data to be used in encryption/decryption: [95:64].	0x0000_0000
AES_cntdata_03	0xEA00_4048	W	Specifies the Counter data to be used in encryption/decryption: [63:32].	0x0000_0000
AES_cntdata_04	0xEA00_404C	W	Specifies the Counter data to be used in encryption/decryption: [31:0].	0x0000_0000
AES_keydata_01	0xEA00_4080	W	Specifies the Key data to be used in encryption/decryption: [255:224].	0x0000_0000
AES_keydata_02	0xEA00_4084	W	Specifies the Key data to be used in encryption/decryption: [223:192].	0x0000_0000
AES_keydata_03	0xEA00_4088	W	Specifies the Key data to be used in encryption/decryption: [191:160].	0x0000_0000
AES_keydata_04	0xEA00_408C	W	Specifies the Key data to be used in encryption/decryption: [159:128].	0x0000_0000
AES_keydata_05	0xEA00_4090	W	Specifies the Key data to be used in encryption/decryption: [127:96].	0x0000_0000

Register	Address	R/W	Description	Reset Value
AES_keydata_06	0xEA00_4094	W	Specifies the Key data to be used in encryption/decryption: [95:64].	0x0000_0000
AES_keydata_07	0xEA00_4098	W	Specifies the Key data to be used in encryption/decryption: [63:32].	0x0000_0000
AES_keydata_08	0xEA00_409C	W	Specifies the Key data to be used in encryption/decryption : [31:0].	0x0000_0000
TDES				
TDES_CONF	0xEA00_5000	R/W	Specifies the TDES configuration register.	0x0000_0000
TDES_STAT	0xEA00_5004	R/W	Specifies the TDES status register.	0x0000_0002
TDES_KEY1_0	0xEA00_5010	W	Specifies the TDES Input Key 1 [63:32].	0x0000_0000
TDES_KEY1_1	0xEA00_5014	W	Specifies the TDES Input Key 1 [31:0].	0x0000_0000
TDES_KEY2_0	0xEA00_5018	W	Specifies the TDES Input Key 2 [63:32].	0x0000_0000
TDES_KEY2_1	0xEA00_501C	W	Specifies the TDES Input Key 2 [31:0].	0x0000_0000
TDES_KEY3_0	0xEA00_5020	W	Specifies the TDES Input Key 3 [63:32].	0x0000_0000
TDES_KEY3_1	0xEA00_5024	W	Specifies the TDES Input Key 3 [31:0].	0x0000_0000
TDES_IV_0	0xEA00_5028	W	Specifies the TDES Initial vector [63:32].	0x0000_0000
TDES_IV_1	0xEA00_502C	W	Specifies the TDES Initial vector [31:0].	0x0000_0000
TDES_INPUT_0	0xEA00_5030	W	Specifies the TDES Input Data [63:32].	0x0000_0000
TDES_INPUT_1	0xEA00_5034	W	Specifies the TDES Input Data [31:0].	0x0000_0000
TDES_OUTPUT_0	0xEA00_5038	R	Specifies the TDES output Data [63:32].	0x0000_0000
TDES_OUTPUT_1	0xEA00_503C	R	Specifies the TDES output Data [31:0].	0x0000_0000
HASH and PRNG				
HASH_CONTROL_1	0xEA00_6000	R/W	Specifies the hash control register 1.	0x0000_0000
HASH_CONTROL_2	0xEA00_6004	W	Specifies the hash control register 2.	0x0000_0000
HASH_FIFO_MOD_E_EN	0xEA00_6008	R/W	Enables FIFO mode.	0x0000_0000
HASH_BYTE_SWA_P	0xEA00_600C	R/W	Specifies the byte swap configuration register.	0x0000_0000
HASH_STATUS	0xEA00_6010	R	Specifies the status register.	0x0000_0001
HASH_MSG_SIZE_LOW	0xEA00_6014	R/W	Specifies the message size in bytes (lower 32-bits).	0x0000_0000
HASH_MSG_SIZE_HIGH	0xEA00_6018	R/W	Specifies the message size in bytes (higher 32-bits).	0x0000_0000
HASH_DATA_IN_1	0xEA00_6020	W	Specifies the key/message input register 1. Only effective when the FIFO mode is disabled.	-

Register	Address	R/W	Description	Reset Value
HASH_DATA_IN_2	0xEA00_6024	W	Specifies the key/message input register 2. Only effective when the FIFO mode is disabled.	-
HASH_DATA_IN_3	0xEA00_6028	W	Specifies the key/message input register 3. Only effective when the FIFO mode is disabled.	-
HASH_DATA_IN_4	0xEA00_602C	W	Specifies the key/message input register 4. Only effective when the FIFO mode is disabled.	-
HASH_DATA_IN_5	0xEA00_6030	W	Specifies the key/message input register 5. Only effective when the FIFO mode is disabled.	-
HASH_DATA_IN_6	0xEA00_6034	W	Specifies the key/message input register 6. Only effective when the FIFO mode is disabled.	-
HASH_DATA_IN_7	0xEA00_6038	W	Specifies the key/message input register 7. Only effective when the FIFO mode is disabled.	-
HASH_DATA_IN_8	0xEA00_603C	W	Specifies the key/message input register 8. Only effective when the FIFO mode is disabled.	-
HASH_SEED_IN_1	0xEA00_6040	W	Specifies the PRNG seed data input 1.	-
HASH_SEED_IN_2	0xEA00_6044	W	Specifies the PRNG seed data input 2.	-
HASH_SEED_IN_3	0xEA00_6048	W	Specifies the PRNG seed data input 3.	-
HASH_SEED_IN_4	0xEA00_604C	W	Specifies the PRNG seed data input 4.	-
HASH_SEED_IN_5	0xEA00_6050	W	Specifies the PRNG seed data input 5.	-
HASH_RESULT_1	0xEA00_6060	R	Specifies the Hash/HMAC/Partial result 1.	0x6745_2301
HASH_RESULT_2	0xEA00_6064	R	Specifies the Hash/HMAC/Partial result 2.	0xefcd_ab89
HASH_RESULT_3	0xEA00_6068	R	Specifies the Hash/HMAC/Partial result 3.	0x98ba_dcfe
HASH_RESULT_4	0xEA00_606C	R	Specifies the Hash/HMAC/Partial result 4.	0x1032_5476
HASH_RESULT_5	0xEA00_6070	R	Specifies the Hash/HMAC/Partial result 5.	0xc3d2_e1f0
HASH_PRNG_1	0xEA00_6080	R	Specifies the PRNG output 1.	0x0000_0000
HASH_PRNG_2	0xEA00_6084	R	Specifies the PRNG output 2.	0x0000_0000
HASH_PRNG_3	0xEA00_6088	R	Specifies the PRNG output 3.	0x0000_0000
HASH_RESULT_4	0xEA00_606C	R	Specifies the Hash/HMAC/Partial result 4.	0x1032_5476
HASH_PRNG_5	0xEA00_6090	R	Specifies the PRNG output 5.	0x0000_0000
HASH_IV_1	0xEA00_60A0	W	Specifies the Custom IV input 1.	-
HASH_IV_2	0xEA00_60A4	W	Specifies the Custom IV input 2.	-
HASH_IV_3	0xEA00_60A8	W	Specifies the Custom IV input 3.	-
HASH_IV_4	0xEA00_60AC	W	Specifies the Custom IV input 4.	-

Register	Address	R/W	Description	Reset Value
HASH_IV_5	0xEA00_60B0	W	Specifies the Custom IV input 5.	-
HASH_PRE_MSG LENG_HIGH	0xEA00_60C0	W	Specifies the Pre-message length [63:32].	0
HASH_PRE_MSG LENG_LOW	0xEA00_60C4	W	Specifies the Pre-message length [31:0].	0
PKA				
PKA_SFR0	0xEA00_7000	R/W	CHNK_SZ / PREC_ID	0x0000_0000
PKA_SFR1	0xEA00_7004	R/W	PLDM_ON / EXEC_ON	0x0000_0000
PKA_SFR2	0xEA00_7008	R/W	SEG_ID (A, B, M, S)	0x0000_0000
PKA_SFR3	0xEA00_700C	R/W	SEG_SIGN	0x0000_0000
PKA_SFR4	0xEA00_7010	R/W	SEG_SIZE, FUNC_ID	0x0000_0000

2.3.1.1 Feed Control (FCINTSTAT, R, Address = 0xEA00_0000)

FCINTSTAT	Bit	Description	R/W	Initial State
Reserved	[31:4]	Reserved		0
BRDMAINT	[3]	Specifies the interrupt signal of block cipher receiving DMA.		0
BTDMAINT	[2]	Specifies the interrupt signal of block cipher transmitting DMA.		0
HRDMAINT	[1]	Specifies the interrupt signal of hash receiving DMA.		0
PKDMAINT	[0]	Specifies the interrupt signal of PKA DMA.		0

2.3.1.2 Feed Control (FCINTENSET, R/W, Address = 0xEA00_0004)

FCINTENSET	Bit	Description	R/W	Initial State
Reserved	[31:4]	Reserved		0
BRDMAINTENSET	[3]	Specifies the interrupt enable set signal of block cipher receiving DMA.		0
BTDMAINTENSET	[2]	Specifies the interrupt enable set signal of block cipher transmitting DMA.		0
HRDMAINTENSET	[1]	Specifies the interrupt enable set signal of hash receiving DMA.		0
PKDMAINTENSET	[0]	Specifies the interrupt enable signal of PKA DMA.		0

2.3.1.3 Feed Control (FCINTENCLR, R/W, Address = 0xEA00_0008)

FCINTENCLR	Bit	Description	R/W	Initial State
Reserved	[31:4]	Reserved		0
BRDMAINTENCLR	[3]	Specifies the interrupt enable clear signal of block cipher receiving DMA.		0
BTDMAINTENCLR	[2]	Specifies the interrupt enable clear signal of block cipher transmitting DMA.		0
HRDMAINTENCLR	[1]	Specifies the interrupt enable clear signal of hash receiving DMA.		0
PKDMAINTENCLR	[0]	Specifies the interrupt enable clear signal of PKA DMA.		0

2.3.1.4 Feed Control (FCINTPEND, R/W, Address = 0xEA00_000C)

FCINTPEND	Bit	Description	R/W	Initial State
Reserved	[31:4]	Reserved		0
BRDMAINTP	[3]	Specifies the Pending Interrupt signal of Block cipher Receiving DMA.		0
BTDMAINTP	[2]	Specifies the Pending Interrupt signal of Block cipher Transmitting DMA.		0
HRDMAINTP	[1]	Specifies the Pending Interrupt signal of Hash Receiving DMA.		0
PKDMAINTP	[0]	Specifies the Pending Interrupt signal of PKA DMA.		0

2.3.1.5 Feed Control (FCFIFOSTAT, R, Address = 0xEA00_0010)

FCFIFOSTAT	Bit	Description	R/W	Initial State
Reserved	[31:8]	Reserved		0
BRFIFOFUL	[7]	Specifies the Full state of Block cipher Receiving FIFO.		0
BRFIFOEMP	[6]	Specifies the Empty state of Block cipher Receiving FIFO.		1
BTIFOFUL	[5]	Specifies the Full state of Block cipher Transmitting FIFO.		0
BTIFOEMP	[4]	Specifies the Empty state of Block cipher Transmitting FIFO.		1
HRFIFOFUL	[3]	Specifies the Full state of Hash Receiving FIFO.		0
HRFIFOEMP	[2]	Specifies the Empty state of Hash Receiving FIFO.		1
PKFIFOFUL	[1]	Specifies the Full state of PKA FIFO.		0
PKFIFOEMP	[0]	Specifies the Empty state of PKA FIFO.		1

2.3.1.6 Feed Control (FCFIFOCTRL, R/W, Address = 0xEA00_0014)

FCFIFOCTRL	Bit	Description	R/W	Initial State
Reserved	[31:3]	Reserved		0
DESSEL	[2]	Specifies the Destination block cipher of FIFO. AES(=0)/DES(=1)		0
HASHINSEL	[1:0]	Specifies the following: Data from independent source (0) Data from block cipher input (1) Data from block cipher output (2) Reserved (3)		0

2.3.1.7 Feed Control (FCBRDMAS, R/W, Address = 0xEA00_0020)

FCBRDMAS	Bit	Description	R/W	Initial State
STARTADDR	[31:0]	Specifies the Start Address of DMA. The address does not to be aligned by 32-bit. Its value increases by 4 after every transaction.		0

2.3.1.8 Feed Control (FCBRDMAL, R/W, Address = 0xEA00_0024)

FCBRDMAL	Bit	Description	R/W	Initial State
LENGTH	[31:0]	Specifies the Block length of DMA. The length does not to be aligned by 32-bit. Its value decreases by 4 after every transaction.		0

2.3.1.9 Feed Control (FCBRDMAC, R/W, Address = 0xEA00_0028)

FCBRDMAC	Bit	Description	R/W	Initial State
Reserved	[31:2]	Reserved		0
BYTESWAP	[1]	If this bit is high, then the data read from the bus is byte-swapped in a word boundary. If this bit is low (default), then the data is handed over to the FIFO without byte-swap. For little endian bus, this bit should be '1'.		0
FLUSH	[0]	If this bit is high, then data flushes out from FIFO and DMA. After flushing, the start address keeps the stopped address, and the length is 0. The flushing state should be released by writing value '0' to this bit.		0

2.3.1.10 Feed Control (FCBTDMAS, R/W, Address = 0xEA00_0030)

FCBTDMAS	Bit	Description	R/W	Initial State
STARTADDR	[31:0]	Specifies the Start Address of DMA. The address needs not to be aligned by 32-bit. Its value increases by 4 after every transaction.		0

2.3.1.11 Feed Control (FCBTDMAL, R/W, Address = 0xEA00_0034)

FCBTDMAL	Bit	Description	R/W	Initial State
LENGTH	[31:0]	Specifies the Block length of DMA. The length needs not to be aligned by 32-bit. Its value decreases by 4 after every transaction.		0

2.3.1.12 Feed Control (FCBTDMAC, R/W, Address = 0xEA00_0038)

FCBTDMAC	Bit	Description	R/W	Initial State
Reserved	[31:2]	Reserved		0
BYTESWAP	[1]	If this bit is high, then the data written to the bus is byte-swapped in a word boundary. If this bit is low (default), then the data is handed over to the FIFO without byte-swap. For little endian bus, this bit should be '1'.		0
FLUSH	[0]	If this bit is high, then data flushes out from FIFO and DMA. After flushing, the start address keeps the stopped address, and the length is 0. The flushing state should be released by writing value '0' to this bit.		0

2.3.1.13 Feed Control (FCHRDMAS, R/W, Address = 0xEA00_0040)

FCHRDMAS	Bit	Description	R/W	Initial State
STARTADDR	[31:0]	Specifies the Start Address of DMA. The address does not need to be aligned by 32-bit. Its value increases by 4 after every transaction.		0

2.3.1.14 Feed Control (FCHRDMAL, R/W, Address = 0xEA00_0044)

FCHRDMAL	Bit	Description	R/W	Initial State
LENGTH	[31:0]	Specifies the Block length of DMA. The length needs not to be aligned by 32-bit. Its value decreases by 4 after every transaction.		0

2.3.1.15 Feed Control (FCHRDMA, R/W, Address = 0xEA00_0048)

FCHRDMA	Bit	Description	R/W	Initial State
Reserved	[31:2]	Reserved		0
BYTESWAP	[1]	If this bit is high, then the data read from the bus is byte-swapped in a word boundary. If this bit is low (default), then the data is handed over to the FIFO without byte-swap. For little endian bus, this bit should be '1'.		0
FLUSH	[0]	If this bit is high, then data flushes out from FIFO and DMA. After flushing, the start address keeps the stopped address, and the length is 0. The flushing state should be released by writing value '0' to this bit.		0

2.3.1.16 Feed Control (FCPKDMAS, R/W, Address = 0xEA00_0050)

FCPKDMAS	Bit	Description	R/W	Initial State
STARTADDR	[31:0]	Specifies the Start Address of DMA. The address needs to be aligned by 32-bit. Its value increases by 4 after every transaction.		0

2.3.1.17 Feed Control (FCPKDMAL, R/W, Address = 0xEA00_0054)

FCPKDMAL	Bit	Description	R/W	Initial State
LENGTH	[31:0]	Specifies the Block length of DMA. The length needs to be aligned by 32-bit. Its value decreases by 4 after every transaction.		0

2.3.1.18 Feed Control (FCPKDMAC, R/W, Address = 0xEA00_0058)

FCPKDMAC	Bit	Description	R/W	Initial State
Reserved	[31:4]	Reserved		0
BYTESWAP	[3]	If this bit is high, then the data read from or written to the bus is byte-swapped in a word boundary. If this bit is low (default), then the data is handed over to the FIFO without byte-swap.		0
DESCEND	[2]	If this bit is low (default), then offset value in FCPKDMAO increases by 4 after every transfer. If this bit is high, then offset value in FCPKDMAO decreases by 4 after every transfer.		0
TRANSMIT	[1]	Selects receiving (0) or transmitting (1).		0
FLUSH	[0]	If this bit is high, then data flushes out from FIFO and DMA. After flushing, the start address keeps the stopped address, and the length is 0. The flushing state should be released by writing value '0' to this bit.		0

2.3.1.19 Feed Control (FCPKDMAO, R/W, Address = 0xEA00_005C)

FCPKDMAO	Bit	Description	R/W	Initial State
Reserved	[31:12]	Reserved		0
SRAMOFFSET	[11:0]	Specifies the Address offset in PKA SRAM. The address needs to be aligned by 32-bit. Its value increases by 4 after every transfer (decreases by 4, if DESCEND in FCPKDMAC is high).		0

2.3.1.20 AES Control (AES_control, R/W, Address = 0xEA00_4000)

AES_control	Bit	Description	R/W	Initial State
Reserved	[31:12]	-	-	-
AES_BitSwap_DI	[11]	0 = Disables Input data byte swap 1 = Enables Input data byte swap	R/W	
AES_BitSwap_DO	[10]	0 = Disables Output data byte swap 1 = Enables Output data byte swap	R/W	
AES_BitSwap_IV	[9]	0 = Disables Initial value byte swap 1 = Enables Initial value byte swap	R/W	
AES_BitSwap_CNT	[8]	0 = Disables Counter data byte swap 1 = Enables Counter data byte swap	R/W	
AES_BitSwap_Key	[7]	0 = Disables Key byte swap 1 = Enables Key byte swap	R/W	
Key Change Mode	[6]	Specifies the AES key change mode selection signal. 0 = Key is not changed 1 = Key is changed		0
AES Key Size	[5:4]	Specifies the AES key size selection signal. 00 = 128-bit key 01 = 192-bit key 10 = 256-bit key		00
FIFO Mode	[3]	Specifies the ARM/ FIFO mode selection signal. 0 = ARM mode (ARM Slave) 1 = FIFO mode		0
AES Chain Mode	[2:1]	Specifies the AES chain mode selection signal. 00 = ECB mode 01 = CBC mode 10 = CTR mode		00
AES Mode	[0]	Specifies the Encryption/ Decryption mode selection signal. 0 = Encryption 1 = Decryption		0

NOTE:

1. AES_control[0]: In case of CTR mode, AES core should always work in encryption mode, even in decryption. Therefore, AES_control[0] should always be '0'.
2. AES_control[6]: The Key Change Mode Bit is used if the key is exactly the same as it was before decryption or encryption. If the AES_control[6] is high, this means the key changes for every block, which consumes double the time of decryption. If a new key should be applied, at least the first block should be processed with AES_control[6] high.

2.3.1.21 AES Control (AES_status, R/W, Address = 0xEA00_4004)

AES_status	Bit	Description	R/W	Initial State
Reserved	[31:3]	-	-	-
Busy	[2]	Specifies the AES busy signal. 0 = Idle 1 = Busy		0
Input Ready	[1]	Specifies the AES input ready signal. 0 = AES input buffer is not empty 1 = AES input buffer is empty, and the host is permitted to write the next block of data		1
Output Ready	[0]	Specifies the AES output ready signal. 0 = AES output is not available 1 = AES output is available to the host for retrieval		0

NOTE: To clear the Output Ready bit, write 0x1 at that bit, AES_status[0].

2.3.1.22 AES Control (AES_indata_01, W, Address = 0xEA00_4010)

AES_indata_01	Bit	Description	R/W	Initial State
AES_indata_01	[31:0]	Specifies the Input data [127:96].		0

2.3.1.23 AES Control (AES_indata_02, W, Address = 0xEA00_4014)

AES_indata_02	Bit	Description	R/W	Initial State
AES_indata_02	[31:0]	Specifies the Input data [95:64].		0

2.3.1.24 AES Control (AES_indata_03, W, Address = 0xEA00_4018)

AES_indata_03	Bit	Description	R/W	Initial State
AES_indata_03	[31:0]	Specifies the Input data [63:32].		0

2.3.1.25 AES Control (AES_indata_04, W, Address = 0xEA00_401C)

AES_indata_04	Bit	Description	R/W	Initial State
AES_indata_04	[31:0]	Specifies the Input data [31:0].		0

2.3.1.26 AES Control (AES_outdata_01, R, Address = 0xEA00_4020)

AES_outdata_01	Bit	Description	R/W	Initial State
AES_outdata_01	[31:0]	Specifies the Output data [127:96].		0

2.3.1.27 AES Control (AES_outdata_02, R, Address = 0xEA00_4024)

AES_outdata_02	Bit	Description	R/W	Initial State
AES_outdata_02	[31:0]	Specifies the Output data [95:64].		0

2.3.1.28 AES Control (AES_outdata_03, R, Address = 0xEA00_4028)

AES_outdata_03	Bit	Description	R/W	Initial State
AES_outdata_03	[31:0]	Specifies the Output data [63:32].		0

2.3.1.29 AES Control (AES_outdata_04, R, Address = 0xEA00_402C)

AES_outdata_04	Bit	Description	R/W	Initial State
AES_outdata_04	[31:0]	Specifies the Output data [31:0].		0

2.3.1.30 AES Control (AES_ivdata_01, W, Address = 0xEA00_4030)

AES_ivdata_01	Bit	Description	R/W	Initial State
AES_ivdata_01	[31:0]	Specifies the Initialization vector [127:96].		0

2.3.1.31 AES Control (AES_ivdata_02, W, Address = 0xEA00_4034)

AES_ivdata_02	Bit	Description	R/W	Initial State
AES_ivdata_02	[31:0]	Specifies the Initialization vector [95:64].		0

2.3.1.32 AES Control (AES_ivdata_03, W, Address = 0xEA00_4038)

AES_ivdata_03	Bit	Description	R/W	Initial State
AES_ivdata_03	[31:0]	Specifies the Initialization vector [63:32].		0

2.3.1.33 AES Control (AES_ivdata_04, W, Address = 0xEA00_403C)

AES_ivdata_04	Bit	Description	R/W	Initial State
AES_ivdata_04	[31:0]	Specifies the Initialization vector [31:0].		0

2.3.1.34 AES Control (AES_cntdata_01, W, Address = 0xEA00_4040)

AES_cntdata_01	Bit	Description	R/W	Initial State
AES_cntdata_01	[31:0]	Specifies the Counter data [127:96].		0

2.3.1.35 AES Control (AES_cntdata_02, W, Address = 0xEA00_4044)

AES_cntdata_02	Bit	Description	R/W	Initial State
AES_cntdata_02	[31:0]	Specifies the Counter data [95:64].		0

2.3.1.36 AES Control (AES_cntdata_03, W, Address = 0xEA00_4048)

AES_cntdata_03	Bit	Description	R/W	Initial State
AES_cntdata_03	[31:0]	Specifies the Counter data [63:32].		0

2.3.1.37 AES Control (AES_cntdata_04, W, Address = 0xEA00_404C)

AES_cntdata_04	Bit	Description	R/W	Initial State
AES_cntdata_04	[31:0]	Specifies the Counter data [31:0].		0

2.3.1.38 AES Control (AES_keydata_01, W, Address = 0xEA00_4080)

AES_keydata_01	Bit	Description	R/W	Initial State
AES_keydata_01	[31:0]	Specifies the Input key data [255:224].		0

2.3.1.39 AES Control (AES_keydata_02, W, Address = 0xEA00_4084)

AES_keydata_02	Bit	Description	R/W	Initial State
AES_keydata_02	[31:0]	Specifies the Input key data [223:192].		0

2.3.1.40 AES Control (AES_keydata_03, W, Address = 0xEA00_4088)

AES_keydata_03	Bit	Description	R/W	Initial State
AES_keydata_03	[31:0]	Specifies the Input key data [191:160].		0

2.3.1.41 AES Control (AES_keydata_04, W, Address = 0xEA00_408C)

AES_keydata_04	Bit	Description	R/W	Initial State
AES_keydata_04	[31:0]	Specifies the Input key data [159:128].		0

2.3.1.42 AES Control (AES_keydata_05, W, Address = 0xEA00_4090)

AES_keydata_05	Bit	Description	R/W	Initial State
AES_keydata_05	[31:0]	Specifies the Input key data [127:96].		0

2.3.1.43 AES Control (AES_keydata_06, W, Address = 0xEA00_4094)

AES_keydata_06	Bit	Description	R/W	Initial State
AES_keydata_06	[31:0]	Specifies the Input key data [95:64].		0

2.3.1.44 AES Control (AES_keydata_07, W, Address = 0xEA00_4098)

AES_keydata_07	Bit	Description	R/W	Initial State
AES_keydata_07	[31:0]	Specifies the Input key data [63:32].		0

2.3.1.45 AES Control (AES_keydata_08, W, Address = 0xEA00_409C)

AES_keydata_08	Bit	Description	R/W	Initial State
AES_keydata_08	[31:0]	Specifies the Input key data [31:0].		0

2.3.1.46 TDES Control (TDES_CONF, R/W, Address = 0xEA00_5000)

TDES_CONF	Bit	Description	R/W	Initial State
Reserved	[31:10]	-	-	-
TDES_BitSwap_DI	[9]	0 = Disables data input byte swap 1 = Enables data input byte swap	R/W	0
TDES_BitSwap_DO	[8]	0 = Disables data output byte swap 1 = Enables data output byte swap	R/W	0
TDES_BitSwap_IV	[7]	0 = Disables initial value byte swap 1 = Enables initial value byte swap	R/W	0
TDES_BitSwap_Key	[6]	0 = Disables key byte swap 1 = Enables key byte swap	R/W	0
TDES_FiFo	[5]	0 = CPU 1 = FiFo	R/W	0
TDES_EEE	[4]	0 = TDES EDE mode 1 = TDES EEE mode	R/W	0
TDES_Select	[3]	0 = DES 1 = TDES	R/W	0
Reseved	[2]	-	-	-
TDES_Mode	[1]	0 = ECB mode 1 = CBC mode	R/W	0
TDES_Enc	[0]	0 = Encryption 1 = Decryption	R/W	0

2.3.1.47 TDES Control (TDES_STAT, R/W, Address = 0xEA00_5004)

TDES_CONF	Bit	Description	R/W	Initial State
Reserved	[31:3]	-	-	-
TDES_Busy	[2]	0 = Idle 1 = Busy	R	0
TDES_Ready	[1]	0 = Input buffer full 1 = Input buffer empty	R	1
TDES_Valid	[0]	0 = Output Invalid 1 = Output Valid	R/W	0

NOTE: To clear the TDES_Valid bit, write 0x1 at that bit, TDES_STAT[0].

2.3.1.48 TDES Control (TDES_KEY1_0, W, Address = 0xEA00_5010)

TDES_KEY1_0	Bit	Description	R/W	Initial State
TDES_KEY1_0	[31:0]	Specifies the Input key 1 [63:32].		0

2.3.1.49 TDES Control (TDES_KEY1_1, W, Address = 0xEA00_5014)

TDES_KEY1_1	Bit	Description	R/W	Initial State
TDES_KEY1_1	[31:0]	Specifies the Input key 1 [31:0].		0

2.3.1.50 TDES Control (TDES_KEY2_0, W, Address = 0xEA00_5018)

TDES_KEY2_0	Bit	Description	R/W	Initial State
TDES_KEY2_0	[31:0]	Specifies the Input key 2 [63:32].	W	0

2.3.1.51 TDES Control (TDES_KEY2_1, W, Address = 0xEA00_501C)

TDES_KEY2_1	Bit	Description	R/W	Initial State
TDES_KEY2_1	[31:0]	Input key 2 [31:0]	W	0

2.3.1.52 TDES Control (TDES_KEY3_0, W, Address = 0xEA00_5020)

TDES_KEY3_0	Bit	Description	R/W	Initial State
TDES_KEY3_0	[31:0]	Specifies the Input key 3 [63:32]	W	0

2.3.1.53 TDES Control (TDES_KEY3_1, W, Address = 0xEA00_5024)

TDES_KEY3_1	Bit	Description	R/W	Initial State
TDES_KEY3_1	[31:0]	Specifies the Input key 3 [31:0].	W	0

2.3.1.54 TDES Control (TDES_IV_0, W, Address = 0xEA00_5028)

TDES_IV_0	Bit	Description	R/W	Initial State
TDES_IV_0	[31:0]	Specifies the Input Initial vector [63:32].	W	0

2.3.1.55 TDES Control (TDES_IV_1, W, Address = 0xEA00_502C)

TDES_IV_1	Bit	Description	R/W	Initial State
TDES_IV_1	[31:0]	Specifies the Input Initial vector [31:0].	W	0

2.3.1.56 TDES Control (TDES_INPUT_0, W, Address = 0xEA00_5030)

TDES_INPUT_0	Bit	Description	R/W	Initial State
TDES_INPUT_0	[31:0]	Specifies the Input data [63:32].	W	0

2.3.1.57 TDES Control (TDES_INPUT_1, W, Address = 0xEA00_5034)

TDES_INPUT_1	Bit	Description	R/W	Initial State
TDES_INPUT_1	[31:0]	Specifies the Input data [31:0].	W	0

2.3.2 TDES CONTROL (TDES_OUTPUT_0, R, ADDRESS = 0xEA00_5038)

TDES_OUTPUT_0	Bit	Description	R/W	Initial State
TDES_OUTPUT_0	[31:0]	Specifies the Output data [63:32].	R	0

2.3.2.1 TDES Control (TDES_OUTPUT_1, R, Address = 0xEA00_503C)

TDES_OUTPUT_1	Bit	Description	R/W	Initial State
TDES_OUTPUT_1	[31:0]	Specifies the Output data [31:0].	R	0

2.3.2.2 HASH and PRNG Control (HASH_CONTROL_1, R/W, Address = 0xEA00_6000)

HASH_CONTROL_1	Bit	Description	R/W	Initial State
Engine_Selection	[3:0]	4'b0000: SHA1_HASH 4'b0001: SHA1_HMAC_INNER 4'b1001: SHA1_HMAC_OUTER 4'b0010: MD5_HASH 4'b0011: MD5_HMAC_INNER 4'b1011: MD5_HMAC_OUTER 4'b0100: PRNG		0000
START_INIT_BIT	[4]	Starts/initializes the hash/HMAC/PRNG (software reset). Automatically cleared by hardware.		0
USER_IV_EN	[5]	Uses customized IV. Automatically cleared by hardware.		0

2.3.2.3 HASH and PRNG Control (HASH_CONTROL_2, W, Address = 0xEA00_6004)

HASH_CONTROL_2	Bit	Description	R/W	Initial State
HASH_PAUSE	[3]	Pauses a hash operation. Automatically cleared by hardware.		0

2.3.2.4 HASH and PRNG Control (HASH_FIFO_MODE_EN, R/W, Address = 0xEA00_6008)

HASH_FIFO_MODE_EN	Bit	Description	R/W	Initial State
HASH_FIFO_MODE_EN	[0]	0 = Disables FIFO mode (default) 1 = Enables FIFO mode		0

2.3.2.5 HASH and PRNG Control (HASH_BYTE_SWAP, R/W, Address = 0xEA00_600C)

HASH_BYTE_SWAP	Bit	Description	R/W	Initial State
HASH_SWAP_DI	[3]	Specifies the Byte swap of data input. 0 = Does not swap (default) 1 = Swap		0
HASH_SWAP_DO	[2]	Specifies the Byte swap of data output (hash result). 0 = Does not swap (default) 1 = Swap		0
HASH_SWAP_IV	[1]	Specifies the Byte swap of custom IVs. 0 = Does not swap (default) 1 = Swap		0

NOTE:

1. If HASH_SWAP_DI or HASH_SWAP_IV is 0, data will enter the hash core in the same order as HRDATA [31:0]. Otherwise, the 32-bit word is byte-swapped before entering the hash core. Note that the hash core is designed with "big endian" in mind, so you should turn on byte swapping if the bus is little endian.
2. SHA1(abcd) = 81fe8bfe_87576c3e_cb22426f_8e578473_82917acf
 READ(HASH_RESULT_1) → HRDATA = 0x81fe8bfe (when HASH_SWAP_DO = 0)
 READ(HASH_RESULT_1) → HRDATA = 0xfe8bfe81 (when HASH_SWAP_DO = 1)
 MD5(abcd) = e2fc714c_4727ee93_95f324cd_2e7f331f
 READ(HASH_RESULT_1) → HRDATA = 0xe2fc714c (when HASH_SWAP_DO = 0)
 READ(HASH_RESULT_1) → HRDATA = 0x4c71fce2 (when HASH_SWAP_DO = 1)
3. You must correctly configure the byte swapping before starting a hash/HMAC operation. (This step is omitted in the example code for simplicity.)

2.3.2.6 HASH and PRNG Control (HASH_STATUS, R, Address = 0xEA00_6010)

HASH_STATUS	Bit	Description	R/W	Initial State
BUFFER_READY	[0]	Specifies the status of the internal SHA1 buffer. 0 = Buffer is full. 1 = Buffer still has empty spaces (not full).		1
SEED_SETTING_DONE	[1]	1 = Seed setup is done. 0 = Seed setup is not done.		0
PRNG_BUSY	[2]	Specifies the PRNG engine status. 1 = Busy 0 = Idle		0
PARTIAL_DONE	[4]	[R/W] The partial result is done Write "1" in this bit to clear it.		0
PRNG_DONE	[5]	[R/W] PRNG is done Write "1" in this bit to clear it.		0
MSG_DONE	[6]	[R/W] Hash/HMAC is done Write "1" in this bit to clear it.		0
PRNG_ERROR	[7]	Specifies the PRNG error bit. This bit goes HIGH if a PRNG request occurs without a complete seed setup. In order to clear this bit, you must perform a complete seed setup operation.		0

2.3.2.7 HASH and PRNG Control (HASH_MSG_SIZE_LOW, R/W, Address = 0xEA00_6014)

HASH_MSG_SIZE_LOW	Bit	Description	R/W	Initial State
HASH_MSG_SIZE_LOW	[31:0]	Specifies the message size in bytes (lower 32-bits).		0

2.3.2.8 HASH and PRNG Control (HASH_MSG_SIZE_HIGH, R/W, Address = 0xEA00_6018)

HASH_MSG_SIZE_HIGH	Bit	Description	R/W	Initial State
HASH_MSG_SIZE_H	[31:0]	Specifies the message size in bytes (higher 32-bits).		0

2.3.2.9 HASH and PRNG Control (HASH_DATA_IN, W, Address = 0xEA00_6020~0xEA00_603C)

HASH_DATA_IN	Bit	Description	R/W	Initial State
HASH_DATA_IN	[31:0]	Specifies the key/message input register 1 ~ 8. Only effective when the FIFO mode is disabled. Supports burst up to 8 words.		-

2.3.2.10 HASH and PRNG Control (HASH_SEED_IN_1, W, Address = 0xEA00_6040)

HASH_SEED_IN_1	Bit	Description	R/W	Initial State
HASH_SEED_IN_1	[31:0]	Specifies the PRNG seed buffer [159:128].		-

2.3.2.11 HASH and PRNG Control (HASH_SEED_IN_2, W, Address = 0xEA00_6044)

HASH_SEED_IN_2	Bit	Description	R/W	Initial State
HASH_SEED_IN_2	[31:0]	Specifies the PRNG seed buffer [127:96].		-

2.3.2.12 HASH and PRNG Control (HASH_SEED_IN_3, W, Address = 0xEA00_6048)

HASH_SEED_IN_3	Bit	Description	R/W	Initial State
HASH_SEED_IN_3	[31:0]	Specifies the PRNG seed buffer [95:64].		-

2.3.2.13 HASH and PRNG Control (HASH_SEED_IN_4, W, Address = 0xEA00_604C)

HASH_SEED_IN_4	Bit	Description	R/W	Initial State
HASH_SEED_IN_4	[31:0]	Specifies the PRNG seed buffer [63:32].		-

2.3.2.14 HASH and PRNG Control (HASH_SEED_IN_5, W, Address = 0xEA00_6050)

HASH_SEED_IN_5	Bit	Description	R/W	Initial State
HASH_SEED_IN_5	[31:0]	Specifies the PRNG seed buffer [31:0].		-

2.3.2.15 HASH and PRNG Control (HASH_RESULT_1, R, Address = 0xEA00_6060)

HASH_RESULT_1	Bit	Description	R/W	Initial State
HASH_RESULT_1	[31:0]	Specifies the Hash/HMAC/Partial result 1.		0x6745_2301

2.3.2.16 HASH and PRNG Control (HASH_RESULT_2, R, Address = 0xEA00_6064)

HASH_RESULT_2	Bit	Description	R/W	Initial State
HASH_RESULT_2	[31:0]	Specifies the Hash/HMAC/Partial result 2.		0xefcd_ab89

2.3.2.17 HASH and PRNG Control (HASH_RESULT_3, R, Address = 0xEA00_6068)

HASH_RESULT_3	Bit	Description	R/W	Initial State
HASH_RESULT_3	[31:0]	Specifies the Hash/HMAC/Partial result 3.		0x98ba_dcfe

2.3.2.18 HASH and PRNG Control (HASH_RESULT_4, R, Address = 0xEA00_606C)

HASH_RESULT_4	Bit	Description	R/W	Initial State
HASH_RESULT_4	[31:0]	Specifies the Hash/HMAC/Partial result 4.		0x1032_5476

2.3.2.19 HASH and PRNG Control (HASH_RESULT_5, R, Address = 0xEA00_6070)

HASH_RESULT_5	Bit	Description	R/W	Initial State
HASH_RESULT_5	[31:0]	Specifies the Hash/HMAC/Partial result 5.		0xc3d2_e1f0

2.3.2.20 HASH and PRNG Control (HASH_PRNG_1, R, Address = 0xEA00_6080)

HASH_PRNG_1	Bit	Description	R/W	Initial State
HASH_PRNG_1	[31:0]	Specifies the PRNG output 1.		0

2.3.2.21 HASH and PRNG Control (HASH_PRNG_2, R, Address = 0xEA00_6084)

HASH_PRNG_2	Bit	Description	R/W	Initial State
HASH_PRNG_2	[31:0]	Specifies the PRNG output 2.		0

2.3.2.22 HASH and PRNG Control (HASH_PRNG_3, R, Address = 0xEA00_6088)

HASH_PRNG_3	Bit	Description	R/W	Initial State
HASH_PRNG_3	[31:0]	Specifies the PRNG output 3.		0

2.3.2.23 HASH and PRNG Control (HASH_PRNG_4, R, Address = 0xEA00_608C)

HASH_PRNG_4	Bit	Description	R/W	Initial State
HASH_PRNG_4	[31:0]	Specifies the PRNG output 4.		0

2.3.2.24 HASH and PRNG Control (HASH_PRNG_5, R, Address = 0xEA00_6090)

HASH_PRNG_5	Bit	Description	R/W	Initial State
HASH_PRNG_5	[31:0]	Specifies the PRNG output 5.		0

2.3.2.25 HASH and PRNG Control (HASH_IV_1, W, Address = 0xEA00_60A0)

HASH_IV_1	Bit	Description	R/W	Initial State
HASH_IV_1	[31:0]	Specifies the Custom IV input 1.		-

2.3.2.26 HASH and PRNG Control (HASH_IV_2, W, Address = 0xEA00_60A4)

HASH_IV_2	Bit	Description	R/W	Initial State
HASH_IV_2	[31:0]	Specifies the Custom IV input 2.		-

2.3.2.27 HASH and PRNG Control (HASH_IV_3, W, Address = 0xEA00_60A8)

HASH_IV_3	Bit	Description	R/W	Initial State
HASH_IV_3	[31:0]	Specifies the Custom IV input 3.		-

2.3.2.28 HASH and PRNG Control (HASH_IV_4, W, Address = 0xEA00_60AC)

HASH_IV_4	Bit	Description	R/W	Initial State
HASH_IV_4	[31:0]	Specifies the Custom IV input 4.		-

2.3.2.29 HASH and PRNG Control (HASH_IV_5, W, Address = 0xEA00_60B0)

HASH_IV_5	Bit	Description	R/W	Initial State
HASH_IV_5	[31:0]	Specifies the Custom IV input 5.		-

2.3.2.30 HASH and PRNG Control (HASH_PRE_MSG_LENH_HIGH, W, Address = 0xEA00_60C0)

HASH_PRE_MSG_LENH_HIGH	Bit	Description	R/W	Initial State
HASH_PRE_MSG_LENH_HIGH	[31:0]	Specifies the pre-message length [63:32].		0

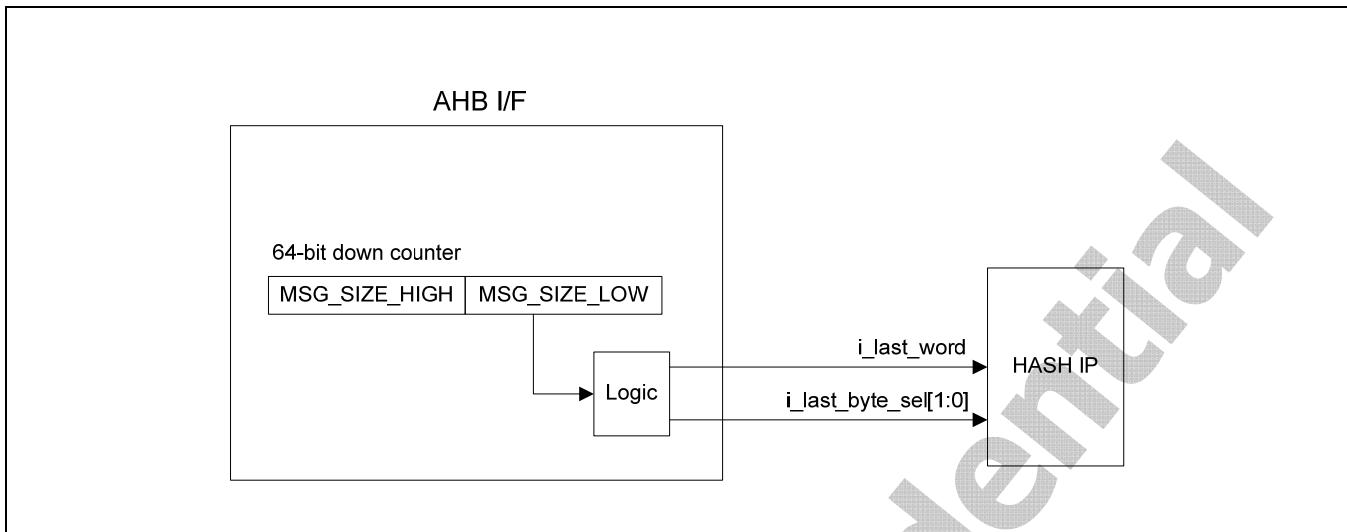
2.3.2.31 HASH and PRNG Control (HASH_PRE_MSG_LENL_LOW, W, Address = 0xEA00_60C4)

HASH_PRE_MSG_LENL_LOW	Bit	Description	R/W	Initial State
HASH_PRE_MSG_LENL_LOW	[31:0]	Specifies the pre-message length [31:0].		0

NOTE:

1. **HASH_CONTROL_1**
ENGINE_SELECTION, START_INIT_BIT and USER_IV_EN can be set at the same time.
If custom IVs are used, HASH_IV_1 ~ HASH_IV_5 must be initialized before USER_IV_EN is set.
2. **HASH_CONTROL_2**
LAST_WORD and LAST_BYTE_SEL[1:0] are removed from the register map. Instead, HASH_MSG_SIZE_LOW and HASH_MSG_SIZE_HIGH are added (see below).

HASH_MSG_SIZE_LOW & HASH_MSG_SIZE_HIGH



As shown in the above diagram, the two registers form a 64-bit counter. When you write values into them, they are initialized with HWDATA. As data words are written through AHB or FIFO, the counter decreases by itself. When the counter is about to become zero, the internal logic generates correct “*i_last_word*” and “*i_last_byte_sel*” signals for the IP.

Note that the unit of this counter is byte. The maximum counting range is $(264 - 1)$ bytes, which is large than specified in the SHA1 specification.

In certain cases, you can use HASH_MSG_SIZE_HIGH and HASH_MSG_SIZE_LOW. A typical example would be multi-part hashing (partial result is involved) without knowing the total message size in advance. In this case, you can initialize the counter with a “big” number (such as 64'h80000000_00000000) for all the parts except the last one. While processing the last part, through which the message will be known, you should initialize this counter with the real message size.

HASH_IV_1 ~ HASH_IV_5

The values in these five registers are sampled and saved by the hardware only when both USER_IV_EN (HASH_CONTROL_1[5]) and START_INIT_BIT (HASH_CONTROL_1[4]) are high. Since USER_IV_EN and START_INIT_BIT are automatically cleared by hardware, these five registers do not need to be cleared after they are used.

HASH_PRE_MSG LENG HIGH & HASH_PRE_MSG LENG LOW

In contrast to HASH_IV_1 ~ HASH_IV_5, these two registers always affect the hardware. Therefore, they must be set to zero when “Pre-message length” is not used.

NOTE: The unit is bit.

2.3.2.32 PKA Control (PKA_SFR0, R/W, Address = 0xEA00_7000)

PKA_SFR0	Bit	Description	R/W	Initial State
PREC_ID	[1:0]	Sets the precision. 00 = Single precision, that is, x1 [Default] 01 = Double precision, that is, x2 10 = Triple precision, that is, x3 11 = Quadruple precision, that is, x4		00
Reserved	[2]	-		-
CHNK_SZ	[6:3]	Sets the chunk size. 0000 = (don't use) [Default] 0001 = (don't use) 0010 = (don't use) 0011 = 128-bits 0100 = 160-bits 0101 = 192-bits 0110 = 224-bits 0111 = 256-bits 1000 = 288-bits 1001 = 320-bits 1010 = 352-bits 1011 = 384-bits 1100 = 416-bits 1101 = 448-bits 1110 = 480-bits 1111 = 512-bits		0000
Reserved	[31:7]	-		-

NOTE: Operand's bit length = (Chunk's size)*(Precision)

For example, (160-bits)*Single=160-bits, (512-bits)*Double=1024-bits, (512-bits)*Quadruple=2048-bits

2.3.2.33 PKA Control (PKA_SFR1, R/W, Address = 0xEA00_7004)

PKA_SFR1	Bit	Description	R/W	Initial State
EXEC_ON	[0]	Controls and monitors the execution of PKA 0 = PKA stays in idle state [Default] 1 = PKA starts to run and continues running		0
Reserved	[2:1]	-		-
PLDM_ON ^(note)	[3]	Controls the pre-loading of the least significant chunk of modulus M 0 = Does not pre-load the least significant chunk of modulus M [Default] 1 = Pre-loads the least significant chunk of modulus M		0

NOTE: If PLDM_ON is set to '1', PKA loads modulus M's least significant chunk data from memory to PKA's internal register at the initial time of multiplication. For the whole modular exponentiation, only the first modular multiplication needs to pre-load the least significant chunk of modulus M. When PKA performs a number of modular multiplications except the first one, the least significant chunk of modulus M is pre-loaded during the previous modular multiplication. When PLDM_ON is '0', the processing time for a multiplication becomes shorter. The saved clock cycles after setting PLDM_ON to '0' is $c/32 + 5$.

2.3.2.34 PKA Control (PKA_SFR2, R/W, Address = 0xEA00_7008)

PKA_SFR2	Bit	Description	R/W	Initial State
S_SEG_ID	[4:0]	Memory Segment ID for input operand S in PKA-2 mode 00000 = Segment 0 [Default] 00001 = Segment 1 00010 = Segment 2 00011 = Segment 3 00100 = Segment 4 00101 = Segment 5 00110 = Segment 6 00111 = Segment 7 01000 = Segment 8 01001 = Segment 9 01010 = Segment 10 01011 = Segment 11 01100 = Segment 12 01101 = Segment 13 01110 = Segment 14 01111 = Segment 15 10000 = Segment 16 10001 = Segment 17 10010 = Segment 18 10011 = Segment 19 10100 = Segment 20 10101 = Segment 21 10110 = Segment 22 10111 = Segment 23 11000 = Segment 24 11001 = Segment 25 11010 = Segment 26 11011 = Segment 27 11100 = Segment 28 11101 = Segment 29 11110 = (dedicated to the hardware's internal usage) 11111 = (dedicated to the hardware's internal usage)		00000
Reserved	[7:5]	-		-

2.3.2.35 PKA Control (PKA_SFR3, R/W, Address = 0xEA00_700C)

PKA_SFR3	Bit	Description	R/W	Initial State
Reserved	[31:30]			-
SEG_SIGN	[29:0]	<p>Specifies the signs of numbers stored in the segments.</p> <ul style="list-style-type: none"> - 0 at the ith bit: The number in the ith segment is positive [Default] - 1 at the ith bit: The number in the ith segment is negative <p>xx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_xxx0: Segment 0 is positive</p> <p>xx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_xx1: Segment 0 is negative</p> <p>xx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_xx0x: Segment 1 is positive</p> <p>xx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_xx1x: Segment 1 is negative</p> <p>xx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_x0xx: Segment 2 is positive</p> <p>xx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_x1xx: Segment 2 is negative</p> <p>xx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_0xxx: Segment 3 is positive</p> <p>xx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_1xxx: Segment 3 is negative</p> <p>xx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_xx0_xxxx: Segment 4 is positive</p> <p>xx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_xxx1_xxxx: Segment 4 is negative</p> <p>xx_xxxx_xxxx_xxxx_xxxx_xxxx_xx0x_xxxx: Segment 5 is positive</p> <p>xx_xxxx_xxxx_xxxx_xxxx_xxxx_xx1x_xxxx: Segment 5 is negative</p> <p>xx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_x0xx_xxxx: Segment 6 is positive</p> <p>xx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_x1xx_xxxx: Segment 6 is negative</p> <p>xx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_0xxx_xxxx: Segment 7 is positive</p> <p>xx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_1xxx_xxxx: Segment 7 is negative</p> <p>xx_xxxx_xxxx_xxxx_xxxx_xxxx_xx0_xxxx_xxxx: Segment 8 is positive</p> <p>xx_xxxx_xxxx_xxxx_xxxx_xxx1_xxxx_xxxx: Segment 8 is negative</p> <p>xx_xxxx_xxxx_xxxx_xxxx_xx0x_xxxx_xxxx: Segment 9 is positive</p> <p>xx_xxxx_xxxx_xxxx_xxxx_xx1x_xxxx_xxxx: Segment 9 is negative</p>		0

PKA_SFR3	Bit	Description	R/W	Initial State
		xx_xxxx_xxxx_xxxx_xxxx_x0xx_xxxx_xxxx: Segment 10 is positive xx_xxxx_xxxx_xxxx_xxxx_x1xx_xxxx_xxxx: Segment 10 is negative xx_xxxx_xxxx_xxxx_xxxx_0xxx_xxxx_xxxx: Segment 11 is positive xx_xxxx_xxxx_xxxx_xxxx_1xxx_xxxx_xxxx: Segment 11 is negative xx_xxxx_xxxx_xxxx_xxx0_xxxx_xxxx_xxxx: Segment 12 is positive xx_xxxx_xxxx_xxxx_xxx1_xxxx_xxxx_xxxx: Segment 12 is negative xx_xxxx_xxxx_xxxx_xx0x_xxxx_xxxx_xxxx: Segment 13 is positive xx_xxxx_xxxx_xxxx_xx1x_xxxx_xxxx_xxxx: Segment 13 is negative xx_xxxx_xxxx_xxxx_x0xx_xxxx_xxxx_xxxx: Segment 14 is positive xx_xxxx_xxxx_xxxx_x1xx_xxxx_xxxx_xxxx: Segment 14 is negative xx_xxxx_xxxx_xxxx_0xxx_xxxx_xxxx_xxxx: Segment 15 is positive xx_xxxx_xxxx_xxxx_1xxx_xxxx_xxxx_xxxx: Segment 15 is negative xx_xxxx_xxxx_xxx0_xxxx_xxxx_xxxx_xxxx: Segment 16 is positive xx_xxxx_xxxx_xxx1_xxxx_xxxx_xxxx_xxxx: Segment 16 is negative xx_xxxx_xxxx_xx0x_xxxx_xxxx_xxxx_xxxx: Segment 17 is positive xx_xxxx_xxxx_xx1x_xxxx_xxxx_xxxx_xxxx: Segment 17 is negative xx_xxxx_xxxx_x0xx_xxxx_xxxx_xxxx_xxxx: Segment 18 is positive xx_xxxx_xxxx_x1xx_xxxx_xxxx_xxxx_xxxx: Segment 18 is negative xx_xxxx_xxxx_0xxx_xxxx_xxxx_xxxx_xxxx: Segment 19 is positive xx_xxxx_xxxx_1xxx_xxxx_xxxx_xxxx_xxxx: Segment 19 is negative xx_xxxx_xxx0_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 20 is positive xx_xxxx_xxx1_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 20 is negative xx_xxxx_xx0x_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 21 is positive xx_xxxx_xx1x_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 21		

PKA_SFR3	Bit	Description	R/W	Initial State
		is negative xx_xxxx_x0xx_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 22 is positive xx_xxxx_x1xx_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 22 is negative xx_xxxx_0xxx_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 23 is positive xx_xxxx_1xxx_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 23 is negative xx_xxx0_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 24 is positive xx_xxx1_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 24 is negative xx_xx0x_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 25 is positive xx_xx1x_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 25 is negative xx_x0xx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 26 is positive xx_x1xx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 26 is negative xx_0xxx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 27 is positive xx_1xxx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 27 is negative x0_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 28 is positive x1_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 28 is negative 0x_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 29 is positive 1x_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx_xxxx: Segment 29 is negative		

2.3.2.36 PKA Control (PKA_SFR4, R/W, Address = 0xEA00_7010)

PKA_SFR4	Bit	Description	R/W	Initial State
FUNC_ID	[0]	Selects the function to be executed. 0 = Montgomery multiplication (A by B) [Default] 1 = Montgomery multiplication (A by 1)		0
Reserved	[4:1]	-		-
SEG_SIZE	[6:5]	Size of the memory segments. 00 = Full-size (that is, 256 bytes) [Default] 01 = Half-size (that is, 128 bytes) 10 = Quarter-size (that is, 64 bytes)		00
Reserved	[31:7]	-		-

NOTE: Selecting the half-size and quarter-size segments is only possible in PKA-2 mode.

Section 12

ETC
Samsung Confidential

Table of Contents

1 Electrical Data.....	1-1
1.1 Absolute Maximum Ratings	1-1
1.2 Recommended Operating Conditions.....	1-3
1.3 D.C. Electrical Characteristics	1-6
1.4 CLK Alternating-Current Electrical Characteristics	1-8
1.5 ROM/ SRAM AC Electrical Characteristics.....	1-11
1.6 Onenand AC Electrical Characteristics.....	1-13
1.7 NFCON AC Electrical Characteristics.....	1-16
1.8 LPDDR1 (mDDR) SDRAM Electrical Characteristics	1-18
1.9 Modemif AC Electrical Characteristics.....	1-20
1.10 LCD Controller AC Electrical Characteristics.....	1-21
1.11 Camera Interface AC Electrical Characteristics.....	1-24
1.12 SDMMC AC Electrical Characteristics.....	1-27
1.13 SPI AC Electrical Characteristics.....	1-28
1.14 I2C AC Electrical Characteristics	1-31
1.15 TSI AC Electrical Characteristics	1-32

List of Figures

Figure Number	Title	Page Number
Figure 1-1	XTIpll Clock Timing	1-8
Figure 1-2	EXTCLK Clock Input Timing	1-8
Figure 1-3	Manual Reset Input Timing	1-9
Figure 1-4	Power-On Reset Sequence	1-10
Figure 1-5	ROM/ SRAM Timing (Tacs = 0, Tcos = 0, Tacc = 2, Tcoh = 0, Tcah = 0, PMC = 0, ST = 0, DW = 16-bit)	1-11
Figure 1-6	OneNand Flash Timing	1-13
Figure 1-7	NAND Flash Timing	1-16
Figure 1-8	LPDDR1 SDRAM Read / Write Timing (Trp = 2, Trcd = 2, Tcl = 2, DW = 16-bit)	1-18
Figure 1-9	LCD Controller Timing	1-21
Figure 1-10	LCD I80 Interface Timing	1-23
Figure 1-11	Camera Interface VSYNC Timing	1-24
Figure 1-12	Camera Interface HREF Timing	1-24
Figure 1-13	Camera Interface Data Timing	1-25
Figure 1-14	Camera Interface PCLK timingTable 1-14 Camera Controller Module Signal Timing Constants	1-25
Table 1-14	Camera Controller Module Signal Timing Constants	1-26
Figure 1-15	High Speed SDMMC Interface Timing	1-27
Figure 1-16	SPI Interface Timing (CPHA = 0, CPOL = 1)	1-28
Figure 1-17	IIC Interface Timing	1-31
Figure 1-18	Transport Stream Interface Timing	1-32

List of Tables

Table Number	Title	Page Number
Table 1-1	Absolute Maximum Rating	1-2
Table 1-2	Recommended Operating Conditions.....	1-3
Table 1-3	I/O DC Electrical Characteristics.....	1-6
Table 1-4	RTC OSC Electrical Characteristics	1-7
Table 1-5	USB DC Electrical Characteristics.....	1-7
Table 1-6	Clock Timing Constants	1-9
Table 1-7	Power on Reset Timing Specifications	1-10
Table 1-8	ROM/SRAM Bus Timing Constants	1-12
Table 1-9	OneNAND Bus Timing Constants.....	1-14
Table 1-10	NFCON Bus Timing Constants	1-17
Table 1-11	Memory Port 1, 2 Interface Timing Constants (LPDDR1 SDRAM)	1-19
Table 1-12	TFT LCD Controller Module Signal Timing Constants	1-22
Table 1-13	LCD I80 Interface Signal Timing Constants	1-23
Table 1-15	High Speed SDMMC Interface Transmit/Receive Timing Constants	1-27
Table 1-16	SPI Interface Transmit/ Receive Timing Constants	1-29
Table 1-17	IIC BUS Controller Module Signal Timing	1-31
Table 1-18	Transport Stream Interface Timing Constants.....	1-32

1

ELECTRICAL DATA

1.1 ABSOLUTE MAXIMUM RATINGS

Any Stress beyond “Absolute Maximum Ratings” listed in [Table 1-1](#) can cause permanent damage to the device. These are stress ratings only and functional operation of the device at these or any other conditions beyond those indicated under “Recommended Operating Conditions” are not implied. Exposure to absolute-maximum rated conditions for extended periods can affect device reliability.



Table 1-1 Absolute Maximum Rating

Parameter	Symbol	Minimum	Maximum	Unit
DC Supply Voltage	VDD_ALIVE,VDD_APPLL,VDD_MPPLL, VDD_EPLL,VDD_VPLL,VDD_ARM, VDD_INT,VDD_UOTG_D, VDD_UHOST_D,VDD_HDMI,VDD_HDMI_PLL, VDD_MIPI_D,VDD_MIPI_PLL	-0.5	1.8	V
	VDD_M1,VDD_M2, VDD_MIPI_A	-0.5	2.5	
	VDD_M0,VDD_SYS0,VDD_SYS1, VDD_EXT0,VDD_EXT1,VDD_EXT2, VDD_CKO,VDD_RTC,VDD_LCD, VDD_CAM,VDD_AUD,VDD_MODEM, VDD_KEY,VDD_DAC_A,VDD_DAC,, VDD_HDMI_OSC, VDD_UOTG_A	-0.5	4.6	
DC Input Voltage	1.1V Input buffer	-0.5	1.8	V
	1.8V Input buffer	-0.5	2.5	
	3.3V Input buffer	-0.5	4.6	
	5.0V Input buffer	-0.5	8.0	
DC Output Voltage	1.1V Input buffer	-0.5	1.8	
	1.8V Input buffer	-0.5	2.5	
	3.3V Input buffer	-0.5	4.6	
DC In/Out Current	-	± 20		mA
Storage Temperature	TSTG	- 40 to 150		°C

NOTE: Absolute Maximum Ratings are those values beyond which damage to the device may occur. Functional operation under these conditions or at any other condition beyond those indicated in the operational sections of this standard is not implied. And VDD_ODRAMA_IO, VDD_ODRAMB_IO, VDD_MDDR_IO, VDD_ONAND_IO absolute maximum rate is written in each MCP spec. Please refer those.

1.2 RECOMMENDED OPERATING CONDITIONS

Operate S5PV210 based on the operating conditions listed in [Table 1-2](#).

Table 1-2 Recommended Operating Conditions

Parameter	Pin Name	Minimum	Typical	Maximum	Unit
DC Supply Voltage for Alive Block ⁽¹⁾	VDD_ALIVE	1.045	1.1	1.26	V
DC Supply Voltage for Core Block ⁽¹⁾	VDD_APPLL	1.045	1.1	1.26	
	VDD_MPPLL				
	VDD_EPPLL				
	VDD_VPPLL				
	VDD_ARM	0.9	1.25 ⁽²⁾	1.31	
DC Supply Voltage for Memory Interface0 (NOR/NAND/OneNAND)	VDD_M0	1.7	1.8/2.5/3.3	3.6	
DC Supply Voltage for Memory Interface1 (DRAM) ⁽³⁾	VDD_M1	1.7	1.8	1.9	
DC Supply Voltage for Memory Interface2 (DRAM) ⁽³⁾	VDD_M2	1.7	1.8	1.9	
DC Supply Voltage for SYS0 Block (XEINT0~7, XOM, XnRESET, XusbXtal, JTAG)	VDD_SYS0	1.7	1.8/2.5/3.0	3.6	
DC Supply Voltage for SYS1 Block (XEINT8~15)	VDD_SYS1	1.7	1.8/2.5/3.0	3.6	
DC Supply Voltage for EXT0	VDD_EXT0	1.7	1.8/2.5/3.0	3.6	
DC Supply Voltage for EXT1	VDD_EXT1	1.7	1.8/2.5/3.0	3.6	
DC Supply Voltage for EXT2	VDD_EXT2	1.7	1.8/2.5/3.0	3.6	
DC Supply Voltage for CKO	VDD_CKO	1.7	2.5/3.0	3.6	
DC Supply Voltage for RTC	VDD_RTC	1.7	2.5/3.0	3.6	
DC Supply Voltage for LCD	VDD_LCD	1.7	2.5/3.0	3.6	
DC Supply Voltage for CAM	VDD_CAM	1.7	2.5/3.0	3.6	
DC Supply Voltage for AUD	VDD_AUD	1.7	2.5/3.0	3.6	
DC Supply Voltage for MODEM	VDD_MODEM	1.7	2.5/3.0	3.6	
DC Supply Voltage for KEY	VDD_KEY	1.7	2.5/3.0	3.6	
DC Supply Voltage for ADAC	VDD_DAC_A	3.0	3.3	3.6	
DC Supply Voltage for DAC	VDD_DAC	3.0	3.3	3.6	
DC Supply Voltage for HDMI Core	VDD_HDMI	1.05	1.1	1.15	
DC Supply Voltage for HDMI PLL	VDD_HDMI_PLL	1.05	1.1	1.15	

Parameter	Pin Name	Minimum	Typical	Maximum	Unit
DC Supply Voltage for HDMI OSC	VDD_HDMI_OSC	3.0	3.3	3.6	
DC Supply Voltage for MIPI I/O	VDD_MIPI_A	1.7	1.8	1.9	
DC Supply Voltage for MIPI Core	VDD_MIPI_D	1.05	1.1	1.15	
DC Supply Voltage for MIPI PLL	VDD_MIPI_PLL	1.05	1.1	1.15	
DC Supply Voltage for USB OTG I/O	VDD_UOTG_A	3.0	3.3	3.6	
DC Supply Voltage for USB OTG Core	VDD_UOTG_D	1.05	1.1	1.15	
DC Supply Voltage for USB HOST I/O	VDD_UHOST_A	3.0	3.3	3.6	
DC Supply Voltage for USB HOST Core	VDD_UHOST_D	1.05	1.1	1.15	
DC Supply Voltage for ADC	VDD_ADC	3.0	3.3	3.6	
DC Supply Voltage for OneDRAM™ A-port IO of MCP	VDD_ODRAMA_IO ⁽⁴⁾	1.7	1.8	1.9	
DC Supply Voltage for OneDRAM™ B-port IO of MCP	VDD_ODRAMB_IO ⁽⁴⁾	1.7	1.8	1.9	
DC Supply Voltage for OneDRAM™ Core of MCP	VDD_ODRAM ⁽⁵⁾	1.7	1.8	1.9	
DC Supply Voltage for Mobile DRAM IO of MCP	VDD_MDDR_IO ⁽⁶⁾	1.7	1.8	1.9	
DC Supply Voltage for Mobile DRAM Core of MCP	VDD_MDDR ⁽⁷⁾	1.7	1.8	1.9	
DC Supply Voltage for OneNAND IO of MCP	VDD_ONAND_IO ⁽⁸⁾	1.7	1.8	1.9	
DC Supply Voltage for OneNAND Core of MCP	VDD_ONAND ⁽⁹⁾	1.7	1.8	1.9	
Operating Temperature	TA	-25 to 85			°C

NOTE:

- 1 In case of Engineering sample, the typical voltage is 1.2 V.
- 2 The recommendatory voltage level to apply DVFS(Dynamic Voltage Frequency Scaling);

Operation Freq.	ARM	1GHz	800MHz	400MHz	200MHz	100MHz
	BUS	200MHz	200MHz	200MHz	200MHz	100MHz
VDD_ARM	1.25V±5%	1.2V±5%	1.05V±5%	0.95V±5%	0.95V±5%	
VDD_INT	1.1V±5%	1.1V±5%	1.1V±5%	1.1V±5%	1.0V±5%	

Caution: In case over 680MHz, VDD_ARM should be higher than VDD_INT.
In DDR2 memory, DVFS does not support because of the DLL.

- 3 VDD_M1/M2 power depends on MCP voltage

- 4 In the MCP, pin Name is "VDDQa" & "VDDQb"
- 5 In the MCP, pin Name is "VDD"
- 6 In the MCP, pin Name is "VDDQd"
- 7 In the MCP, pin Name is "VDD"
- 8 In the MCP, pin Name is "VCCQo"
- 9 In the MCP, pin Name is "VCCo"

Samsung Confidential

1.3 D.C. ELECTRICAL CHARACTERISTICS

The entire DC characteristics listed in [Table 1-3](#) for each pin include input sense levels, output drive levels, and currents.

Use these parameters to determine maximum DC loading and to determine maximum transition times for a given load. [Table 1-3](#) shows the DC operating conditions for the high- and low-strength input, output, and I/O pins.

Table 1-3 I/O DC Electrical Characteristics

VDD = 1.65V~3.60V, T_j = -25 to 85°C (Junction temperature)

Parameter		Condition	Minimum	Typical	Maximum	Unit
Vtol	Tolerant external voltage**	VDD Power Off			3.6	V
		VDD Power On	VDD=3.3V		3.6	V
			VDD=1.8V		3.6	
Vih	High Level Input Voltage					
	LVC MOS Interface		0.7VDD		VDD+0.3V	V
Vil	Low Level Input Voltage					
	LVC MOS Interface		-0.3V		0.3VDD	V
ΔV	Hysteresis Voltage		0.1VDD			V
Iih	High Level Input Current					
	Input Buffer	Vin=VDD (VDD=min)	-10		10	uA
			45 ⁽¹⁾			uA
Iil	Low Level Input Current					
	Input Buffer	Vin=VSS (Vss=min)	-10		10	uA
					-45 ⁽¹⁾	uA
Voh	Output high voltage ⁽²⁾	Driver Strength	0.8xVDD			V
Vol	Output Low voltage ⁽²⁾	Driver Strength			0.2xVDD	V
Ioz	Tri-State Output Leakage Current	Vout=VSS or VDD	-10		10	uA
CIN	Input capacitance	Any input and Bidirectional buffers			5	pF
COUT	Output capacitance	Any output buffer			5	pF

NOTE:

1. The values of Ioh & Iol are valid only for 3.3V range.
2. The value of IOH and IOL is for min. driver strength.

Table 1-4 RTC OSC Electrical Characteristics

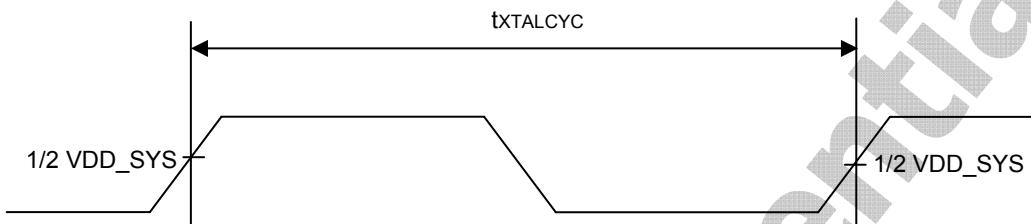
Symbol	Parameter	Minimum	Typical	Maximum	Unit
VIH	DC input logic high	0.7*VDDrtc			V
VIL	DC input logic low			0.3*VDDrtc	V
IIH	High level input current	-10		10	µA
IIL	Low level input current	-10		10	µA

Table 1-5 USB DC Electrical Characteristics

Symbol	Parameter	Condition	Minimum	Maximum	Unit
VIH	High level input voltage		2.0		V
VIL	Low level input voltage			0.8	V
IIH	High level input current	Vin = 3.3v	-10	10	µA
IIL	Low level input current	Vin = 0.0v	-10	10	µA
VOH	Static Output High	14.25Kohm to GND	2.8	3.6	V
VOL	Static Output Low	1.425Kohm to 3.6V		0.3	V
VBUS	Valid level voltage		4.4	5.25	V

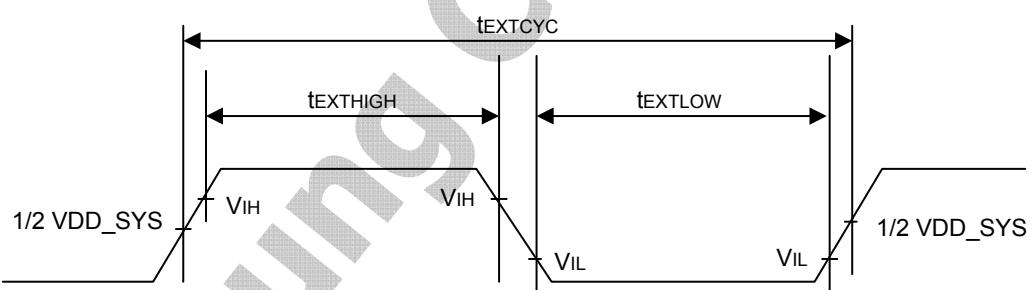
1.4 CLK ALTERNATING-CURRENT ELECTRICAL CHARACTERISTICS

Pin's Alternating-Current (AC) characteristics include input and output capacitance. These factors determine the loading for external drivers and other load analyses.



NOTE: The clock input from the XTalII pin.

Figure 1-1 XTalII Clock Timing



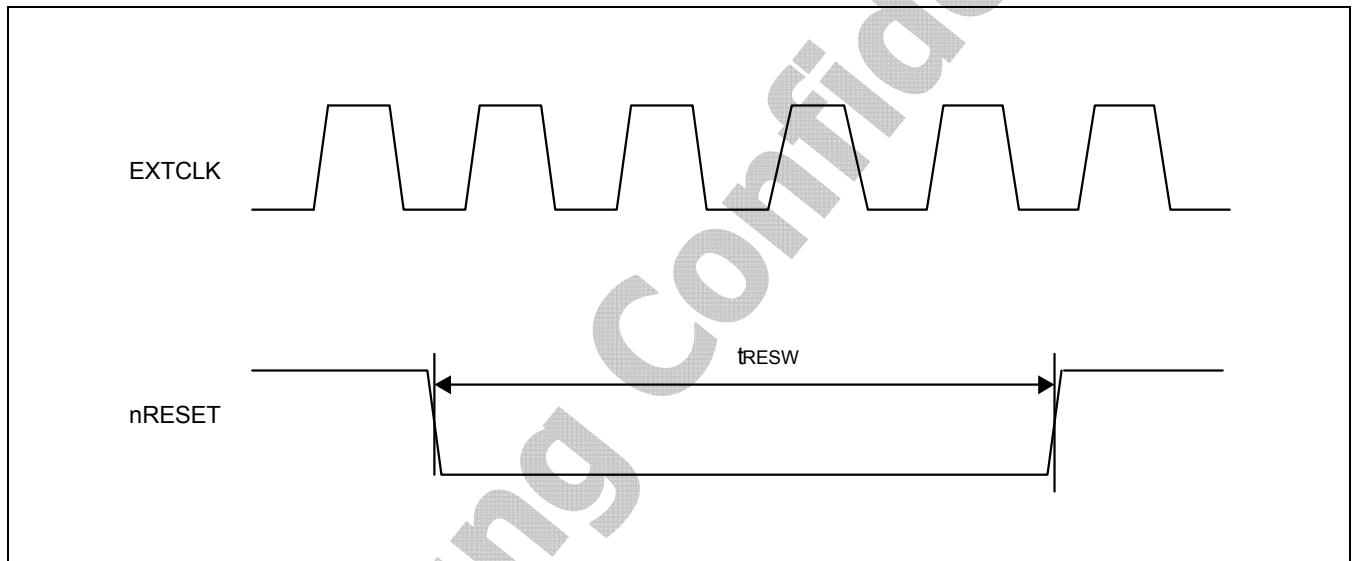
NOTE: The clock input from the EXTCLK pin.

Figure 1-2 EXTCLK Clock Input Timing

Table 1-6 Clock Timing Constants(VDDINT = 1.1V \pm 5%, TA = -25 to 85°C, VDDSYS = 3.3V \pm 5%, 2.5V \pm 5%, 1.8V \pm 5%)

Parameter	Symbol	Minimum	Typical	Maximum	Unit
Duration for crystal oscillator clock input	$t_{XTALCYC}$	20			ns
High width for external clock input	$t_{EXTHIGH}$	10			ns
Low width for external clock input	t_{EXTLOW}	10			ns
APLL lock time	t_{APLL}			100	usec
MPLL lock time	t_{MPLL_LT}			400	Fref*
EPLL lock time	t_{EPLL_LT}			3000	Fref
VPLL lock time	t_{VPLL_LT}			400	Fref

* Fref = Fin/P(refer to CMU manual)

**Figure 1-3 Manual Reset Input Timing**

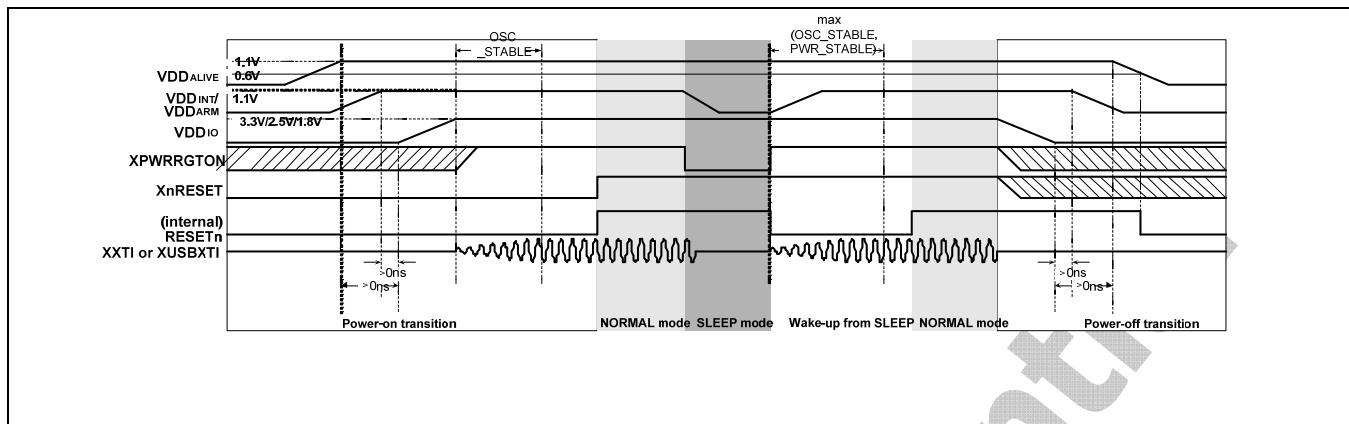


Figure 1-4 Power-On Reset Sequence

OSC STABLE in Figure 1-4 indicates the time required for the oscillator pad to be stabilized.

Table 1-7 Power on Reset Timing Specifications

($VDDINT = 1.1V \pm 5\%$, $TA = -25$ to $85^\circ C$, $VDDSYS = 3.3V \pm 5\%$, $2.5V \pm 5\%$, $1.8V \pm 5\%$)

Parameter	Symbol	Minimum	Typical	Maximum	Unit
Reset assert time after clock stabilization	t_{RESW}	4		-	XTIPLL or EXTCLK

1.5 ROM/ SRAM AC ELECTRICAL CHARACTERISTICS

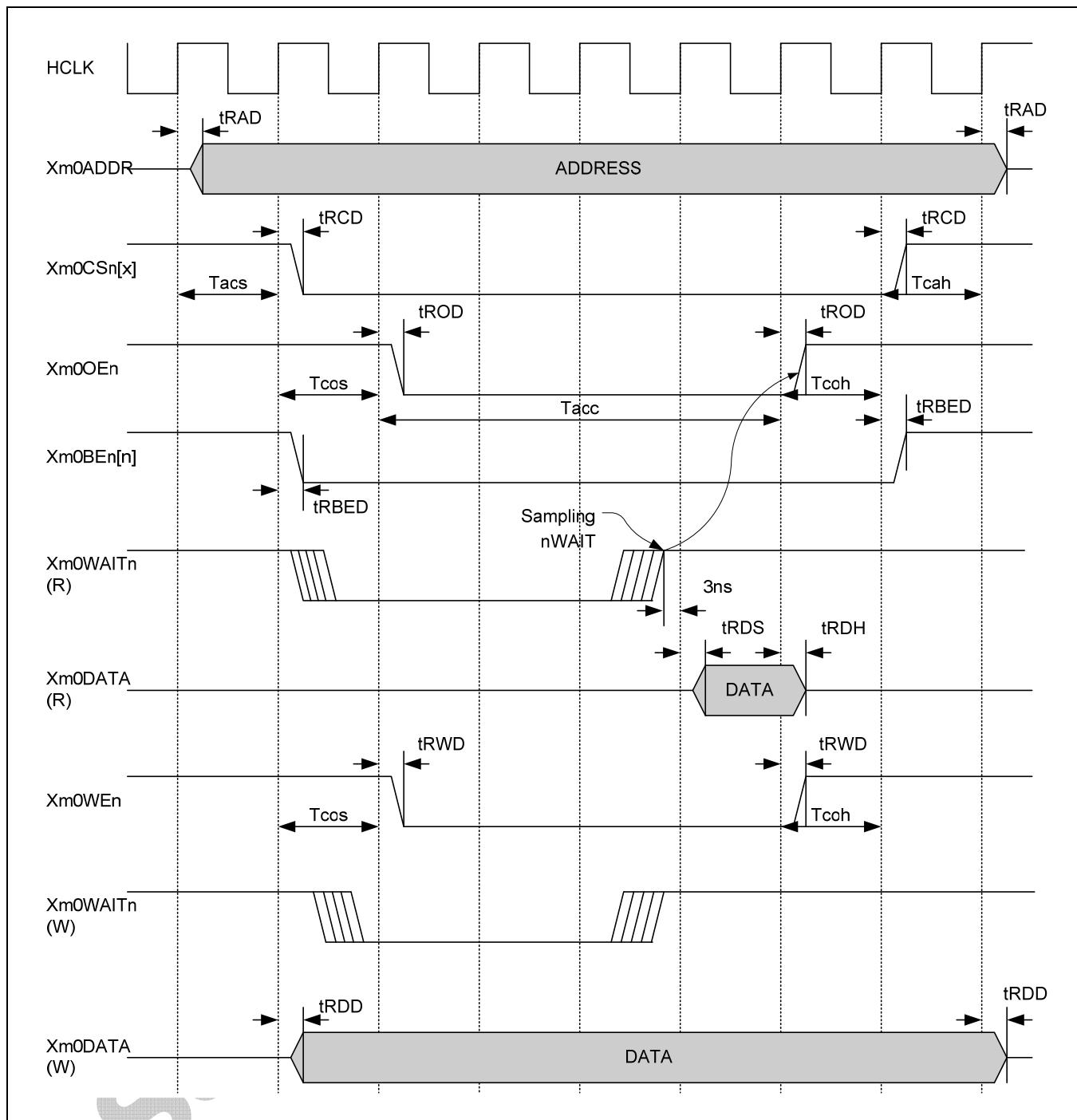


Figure 1-5 ROM/ SRAM Timing
 (Tacs = 0, Tcos = 0, Tacc = 2, Tcoh = 0, Tcah = 0, PMC = 0, ST = 0, DW = 16-bit)

Table 1-8 ROM/SRAM Bus Timing Constants(VDDINT = 1.1V \pm 5%, TA = -25 to 85°C, VDDM0 = 1.7V - 3.6V)

Parameter	Symbol	Minimum	Maximum	Unit
ROM/SRAM Address Delay	t_{RAD}	1.77	5.22	ns
ROM/SRAM Chip Select 0 Delay	t_{RCD}	3.39	8.06	ns
ROM/SRAM Chip Select 1 Delay	t_{RCD}	3.48	8.33	ns
ROM/SRAM Chip Select 2 Delay	t_{RCD}	4.13	7.29	ns
ROM/SRAM Chip Select 3 Delay	t_{RCD}	3.14	7.22	ns
ROM/SRAM Chip Select 4 Delay	t_{RCD}	2.16	4.21	ns
ROM/SRAM Chip Select 5 Delay	t_{RCD}	2.30	4.53	ns
ROM/SRAM nOE(Output Enable) Delay	t_{ROD}	2.78	5.46	ns
ROM/SRAM nWE(Write Enable) Delay	t_{RWD}	2.02	3.96	ns
ROM/SRAM Byte Enable Delay	t_{RBED}	2.95	6.52	ns
ROM/SRAM Output Data Delay	t_{RDD}	3.50	7.41	ns
ROM/SRAM Read Data Setup Time	t_{RDS}	2.00	-	ns
ROM/SRAM Write Data Hold Time	t_{RDH}	1.00	-	ns

1.6 ONENAND AC ELECTRICAL CHARACTERISTICS

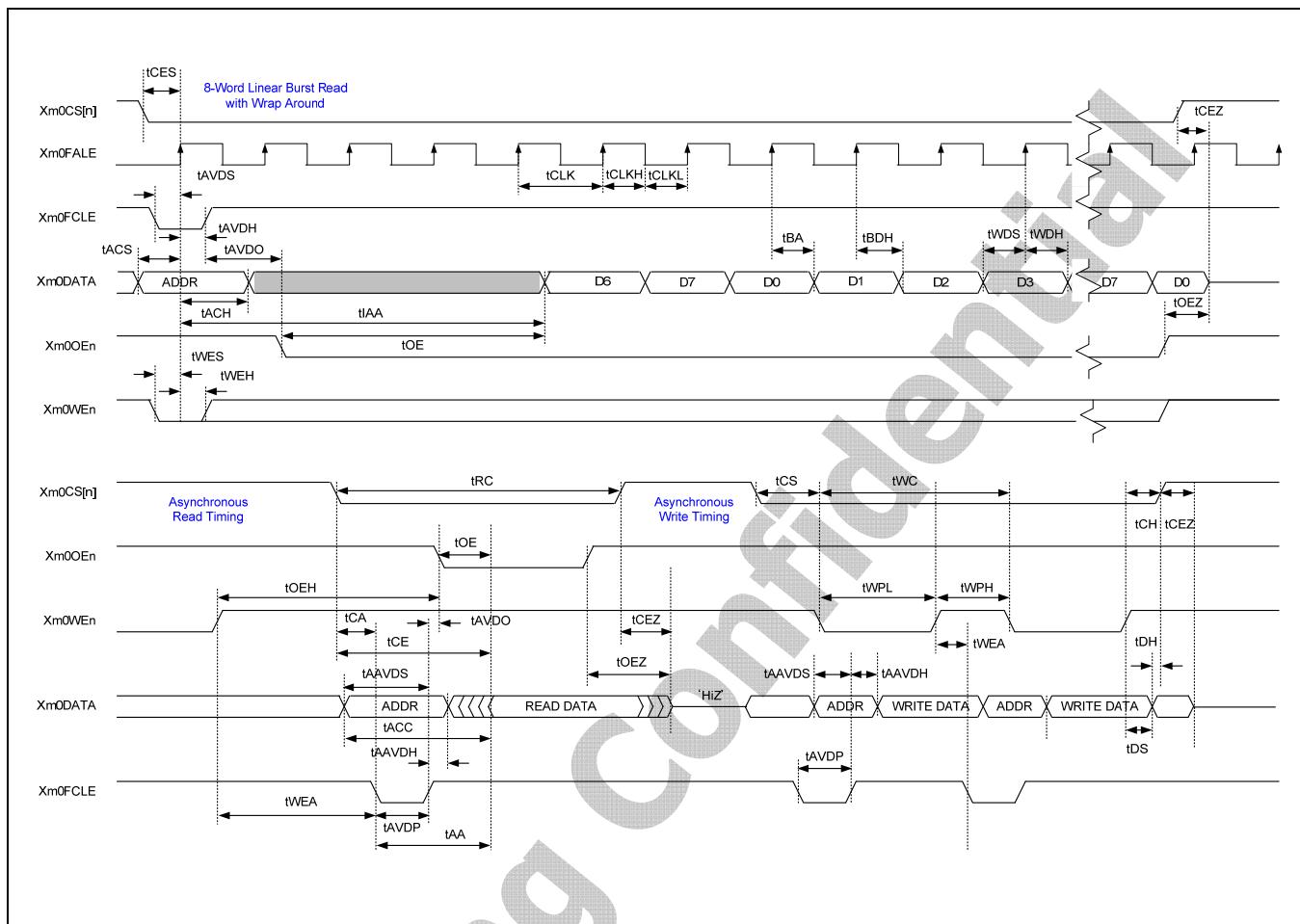


Figure 1-6 OneNand Flash Timing

Table 1-9 OneNAND Bus Timing Constants

(VDDINT = 1.1V ± 5%, TA = -25 to 85°C, VDDM0 = 1.7V - 1.9V)

Parameter	Symbol	Minimum	Maximum	Unit
OneNAND SMCLK cycle	t_{CLK}	12	-	ns
OneNAND Clock High time	t_{CLKH}	5	-	ns
OneNAND Clock Low time	t_{CLKL}	5	-	ns
OneNAND CSn Setup time to SMCLK	t_{CES}	4.5	-	ns
OneNAND Initial Access time	t_{IAA}	-	70	ns
OneNAND Burst Access time valid SMCLK to Output delay	t_{BA}	-	9	ns
OneNAND Data Hold time from next clock cycle	t_{BDH}	2	-	ns
OneNAND Output Enable to Data	t_{OE}	-	20	ns
OneNAND CSn Disable to Output High Z	t_{CEZ}	-	20	ns
OneNAND OEn Disable to Output High Z	t_{OEZ}	-	15	ns
OneNAND Address Setup time to SMCLK	t_{ACS}	4	-	ns
OneNAND Address Hold time to SMCLK	t_{ACH}	6	-	ns
OneNAND ADRVALID Setup time to SMCLK	t_{AVDS}	4	-	ns
OneNAND ADRVALID Hold time to SMCLK	t_{AVDH}	6	-	ns
OneNAND Write Data Setup time to SMCLK	t_{WDS}	4	-	ns
OneNAND Write Data Hold time to SMCLK	t_{WDH}	2	-	ns
OneNAND WEn Setup time to SMCLK	t_{WES}	4	-	ns
OneNAND WEn Hold time to SMCLK	t_{WEH}	6	-	ns
OneNAND ADRVALID high to OEn low	t_{AVDO}	0	-	ns
OneNAND Access time from CSn low	t_{CE}	-	76	ns
OneNAND Asynchronous Access time from ADRVALID low	t_{AA}	-	76	ns
OneNAND Asynchronous Access time from address valid	t_{ACC}	-	76	ns
OneNAND Read Cycle time	t_{RC}	76	-	ns
OneNAND ADRVALID low pulse width	t_{AVDP}	12	-	ns
OneNAND Address Setup to rising edge of ADRVALID	t_{AAVDS}	5	-	ns
OneNAND Address Hold to rising edge of ADRVALID	t_{AAVDH}	7	-	ns
OneNAND CSn Setup to ADRVALID falling edge	t_{CA}	0	-	ns
OneNAND WEn Disable to ADRVALID enable	t_{WEA}	15	-	ns
OneNAND Address to OEn low	t_{ASO}	10	-	ns
OneNAND WEn Cycle time	t_{WC}	70	-	ns
OneNAND Data Setup time	t_{DS}	30	-	ns

Parameter	Symbol	Minimum	Maximum	Unit
OneNAND Data Hold time	t_{DH}	0	-	ns
OneNAND CSn Setup time	t_{CS}	0	-	ns
OneNAND CSn Hold time	t_{CH}	0	-	ns
OneNAND WEn Pulse width low	t_{WPL}	40	-	ns
OneNAND WEn Pulse width high	t_{WPH}	30	-	ns

1.7 NFCON AC ELECTRICAL CHARACTERISTICS

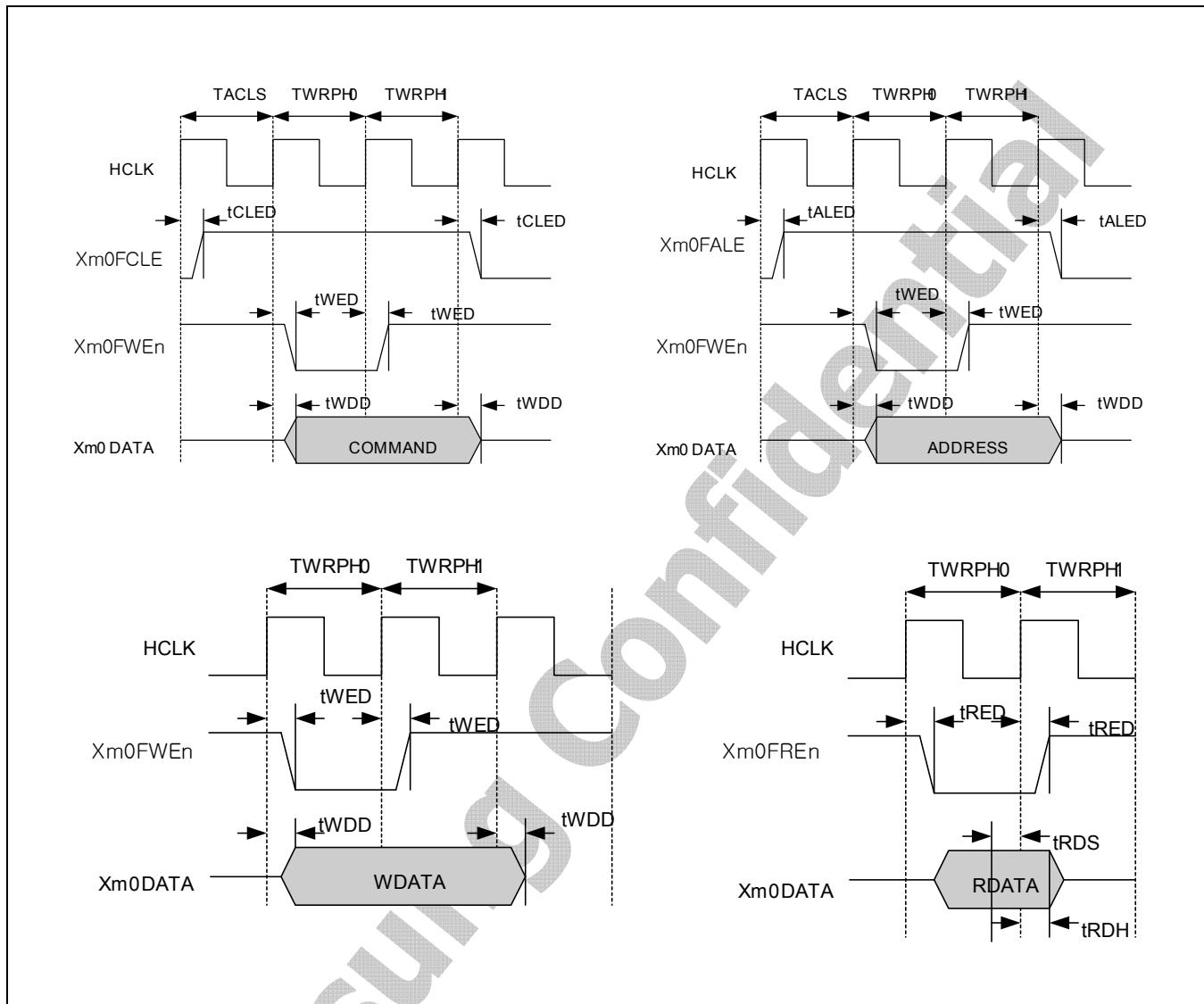


Figure 1-7 NAND Flash Timing

Table 1-10 NFCON Bus Timing Constants(VDDINT = 1.1V \pm 5%, TA = -25 to 85°C, VDDM0 = 1.7V - 3.6V)

Parameter	Symbol	Minimum	Maximum	Unit
NFCON Chip Enable delay	t_{CED}	-	8.40	ns
NFCON CLE delay	t_{CLED}	-	4.45	ns
NFCON ALE delay	t_{ALED}	-	5.36	ns
NFCON Write Enable delay	t_{WED}	-	8.15	ns
NFCON Read Enable delay	t_{RED}	-	8.21	ns
NFCON Write Data delay	t_{WDD}	-	5.39	ns
NFCON Read Data Setup requirement time	t_{RDS}	1.00	-	ns
NFCON Read Data Hold requirement time	t_{RDH}	0.20	-	ns

1.8 LPDDR1 (MDDR) SDRAM ELECTRICAL CHARACTERISTICS

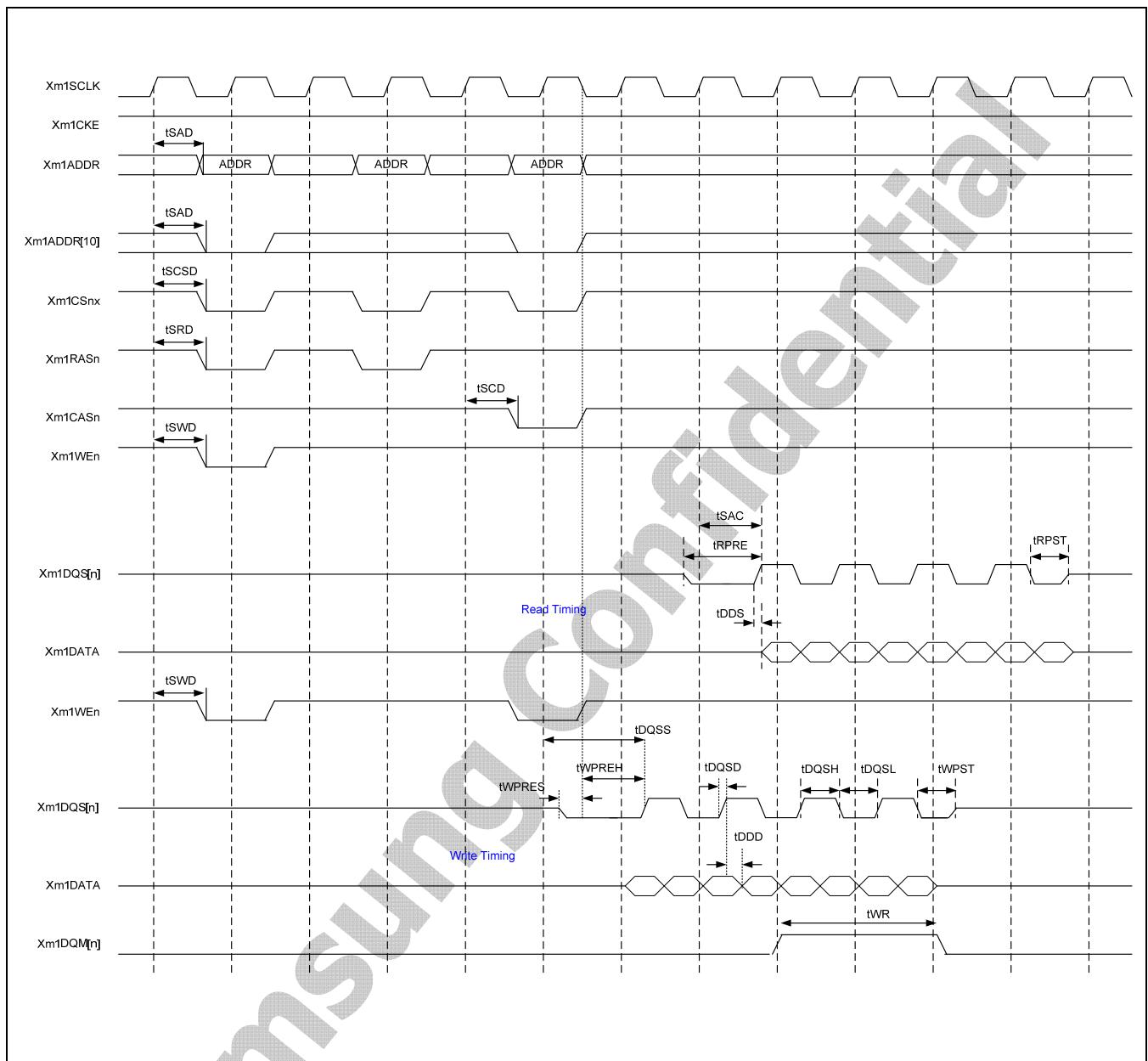


Figure 1-8 LPDDR1 SDRAM Read / Write Timing (Trp = 2, Trcd = 2, TcI = 2, DW = 16-bit)

Table 1-11 Memory Port 1, 2 Interface Timing Constants (LPDDR1 SDRAM)

(VDDINT = 1.1V ± 5%, TA = -25 to 85°C, VDDM1,VDDM2 = 1.7V – 1.9V)

Parameter	Symbol	Minimum	Maximum	Unit
DDR SDRAM Address Delay	t_{SAD}	2.48	3.51	ns
DDR SDRAM Chip Select Delay	t_{SCSD}	2.62	3.72	ns
DDR SDRAM Row active Delay	t_{SRD}	2.63	3.73	ns
DDR SDRAM Column active Delay	t_{SCD}	2.62	3.72	ns
DDR SDRAM Write enable Delay	t_{SWD}	2.62	3.73	ns
DDR SDRAM Output data access time from CK	t_{SAC}	2.00	5.50	ns
DDR SDRAM Row Precharge time	t_{RP}	18.00	-	ns
DDR SDRAM RAS to CAS delay	t_{RCD}	18.00	-	ns
DDR SDRAM Write recovery time	t_{WR}	12.00	-	ns
DDR SDRAM Clock low level width	t_{CL}	0.45	0.55	tCK
DDR SDRAM Read Preamble	t_{RPRE}	0.90	1.10	tCK
DDR SDRAM Read Postamble	t_{RPST}	0.40	0.60	tCK
DDR SDRAM Write Postamble time	t_{WPST}	0.40	0.60	tCK
DDR SDRAM Clock to valid DQS-In	t_{DQSS}	0.75	1.25	tCK
DDR SDRAM DQS-In Setup time	t_{WPRES}	1.30	-	ns
DDR SDRAM DQS-In Hold time	t_{WPREH}	1.30	-	ns
DDR SDRAM DQS-In high level width	t_{DQSH}	0.35	0.60	tCK
DDR SDRAM DQS-In low level width	t_{DQSL}	0.35	0.60	tCK
DDR SDRAM read Data Setup time	t_{DDS}	-	0.50	ns

Load Capacitance	
Xm1,Xm2 signal line	< 15pF

1.9 MODEMIF AC ELECTRICAL CHARACTERISTICS

For more information, Refer to Section 8-6 S5PV210_Modem Interface user's manual.

Samsung Confidential

1.10 LCD CONTROLLER AC ELECTRICAL CHARACTERISTICS

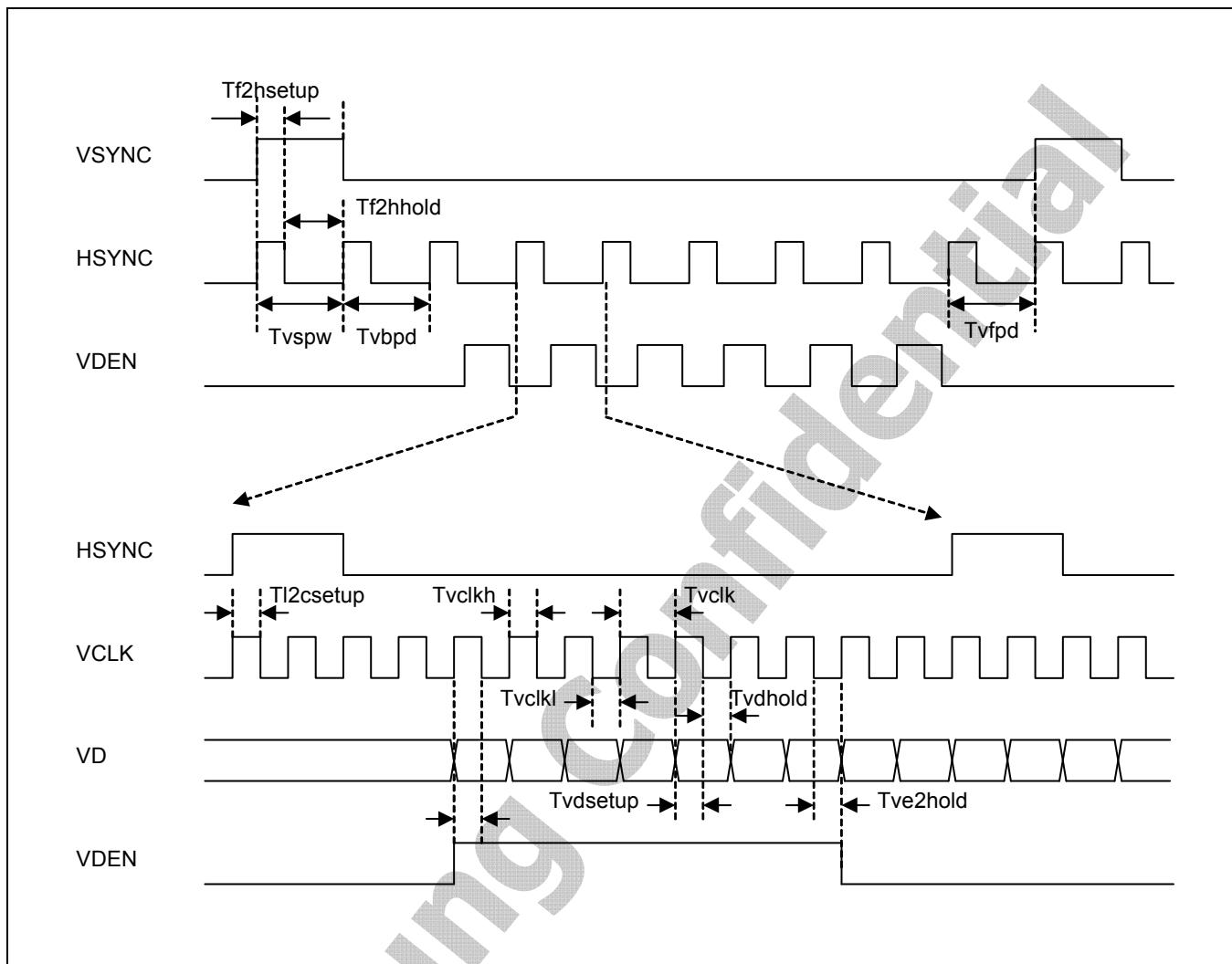


Figure 1-9 LCD Controller Timing

Table 1-12 TFT LCD Controller Module Signal Timing Constants

(VDDINT = 1.1V ± 5%, TA = -25 to 85°C, VDDlcd = 1.7V - 3.6V)

Parameter	Symbol	Minimum	Typical	Maximum	Unit
VCLK pulse width	Tvclk	12	-	-	ns
VCLK pulse width high	Tvclkh	0.3	-	-	Pvclk ⁽¹⁾
VCLK pulse width low	Tvclkl	0.3	-	-	Pvclk
Vertical sync pulse width	Tvspw	VSPW + 1	-	-	Phclk ⁽²⁾
Vertical back porch delay	Tvbpd	VBPD+1	-	-	Phclk
Vertical front porch delay	Tvfpd	VFPD+1	-	-	Phclk
Hsync setup to VCLK falling edge	Tl2csetup	0.3	-	-	Pvclk
VDEN setup to VCLK falling edge	Tde2csetup	0.3	-	-	Pvclk
VDEN hold from VCLK falling edge	Tde2chold	0.3	-	-	Pvclk
VD setup to VCLK falling edge	Tvd2csetup	0.3	-	-	Pvclk
VD hold from VCLK falling edge	Tvd2chold	0.3	-	-	Pvclk
VSYNC setup to HSYNC falling edge	Tf2hsetup	HSPW + 1	-	-	Pvclk
VSYNC hold from HSYNC falling edge	Tf2hhold	HBPD + HFDPD + HOZVAL + 3	-	-	Pvclk

NOTE:

1. VCLK period
2. HSYNC period

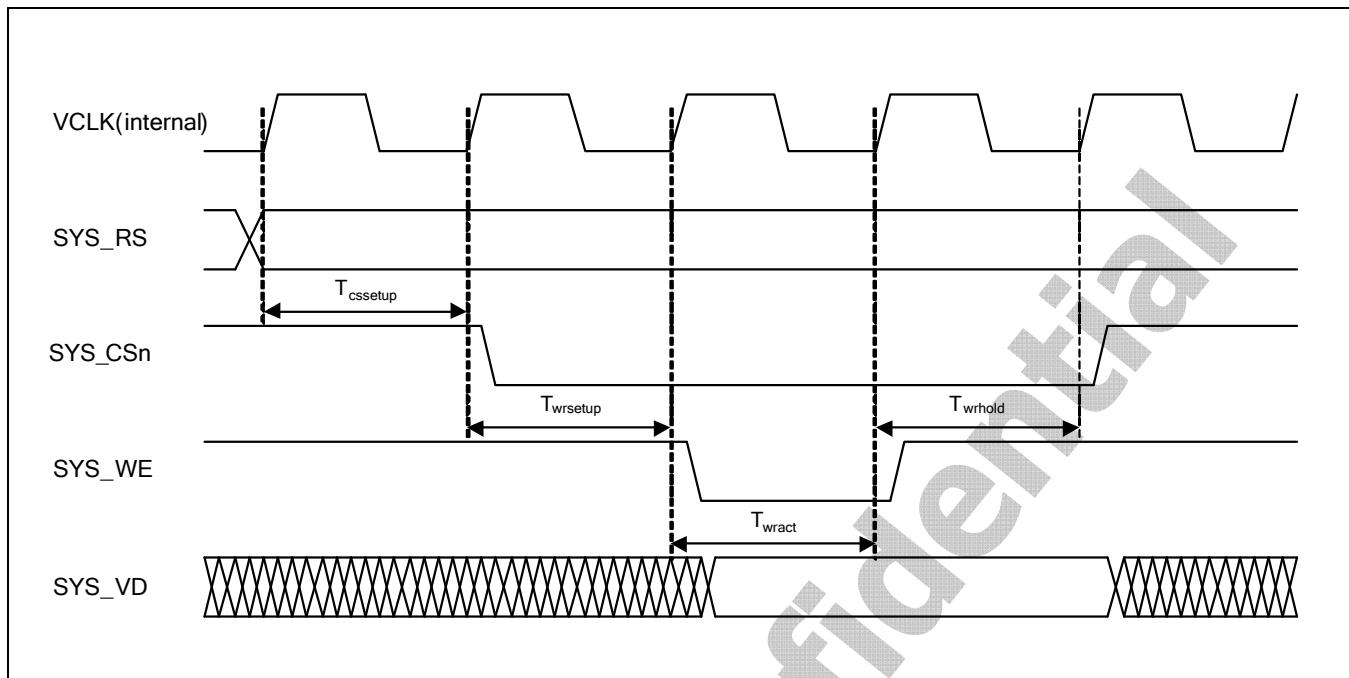


Figure 1-10 LCD I80 Interface Timing

Table 1-13 LCD I80 Interface Signal Timing Constants

(VDDINT = 1.1V ± 5%, TA = -25 to 85°C, VDDIcd = 1.7V - 3.6V)

Parameter	Symbol	Minimum	Typical	Maximum	Unit
SYS_RS to SYS_CSn Low	T _{cssetup}	-	LCD_CS_SETUP + 1	-	Pvclk*
SYS_CSn Low to SYS_WR Low	T _{wrsetup}	-	LCD_WR_SETUP	-	Pvclk
SYS_WE Pulse Width	T _{wract}	-	LCD_WR_ACT	-	Pvclk
SYS_WE High to SYS_CSn High	T _{wrhold}	-	LCD_WR_HOLD	-s	Pvclk

NOTE: Internal VCLK period

1.11 CAMERA INTERFACE AC ELECTRICAL CHARACTERISTICS

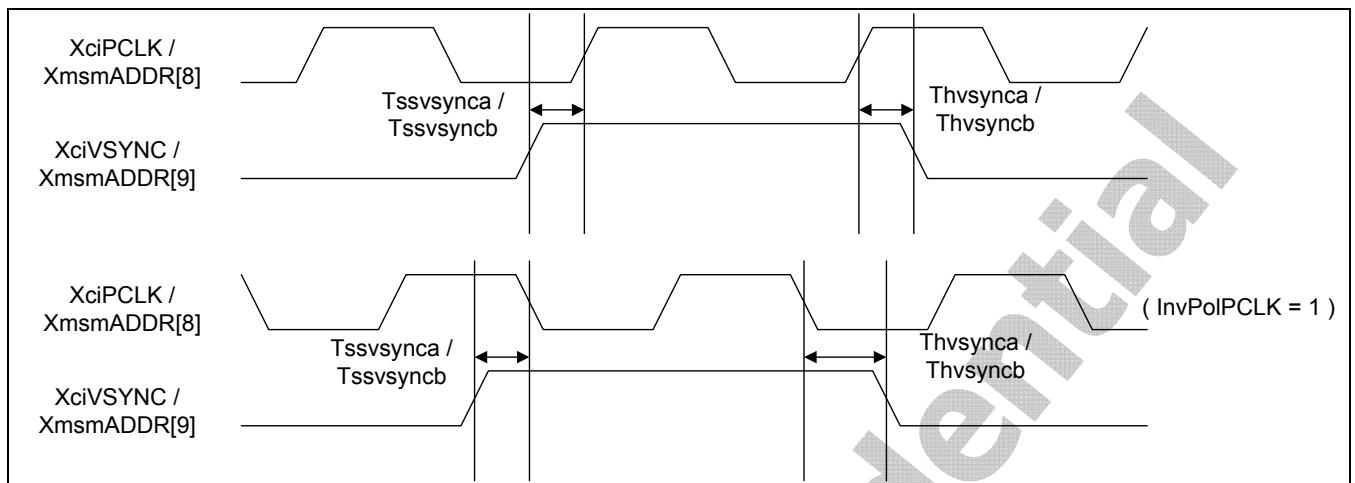


Figure 1-11 Camera Interface VSYNC Timing

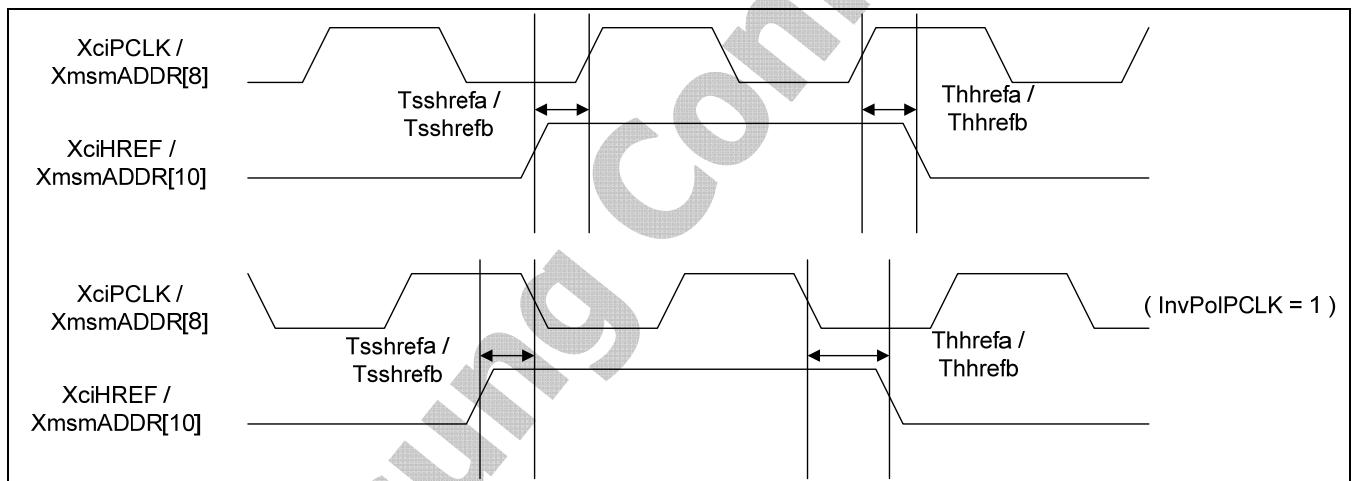


Figure 1-12 Camera Interface HREF Timing

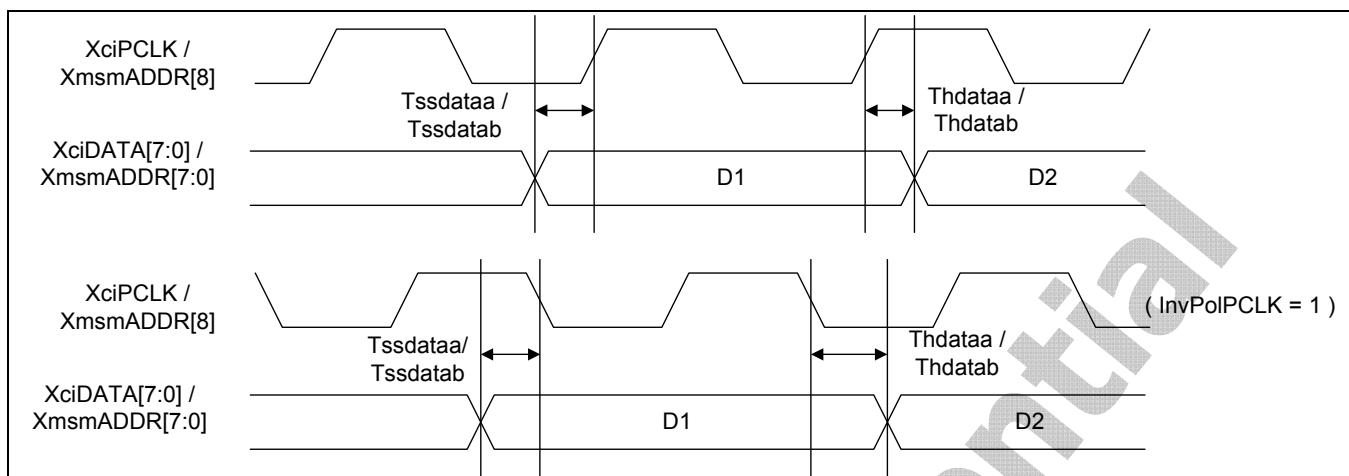


Figure 1-13 Camera Interface Data Timing

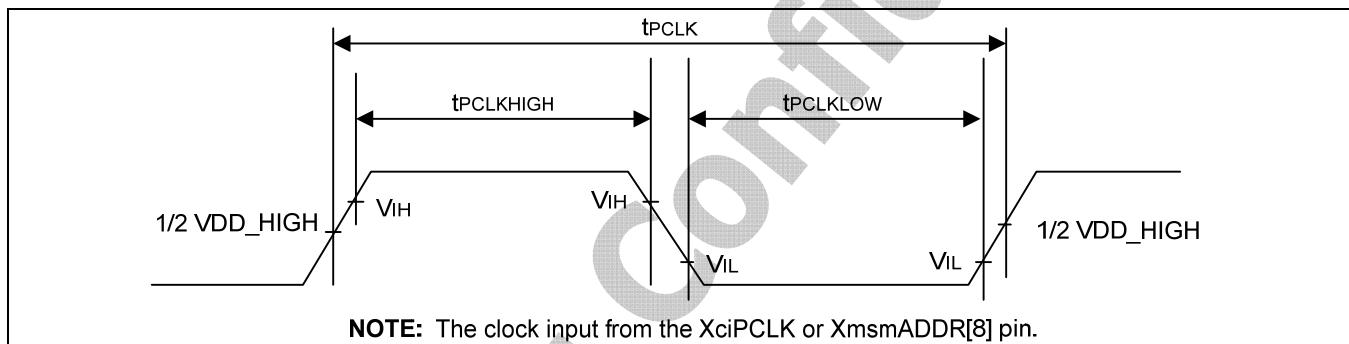


Figure 1-14 Camera Interface PCLK timing

Table 1-14 Camera Controller Module Signal Timing Constants

(VDDINT= 1.1V ± 5%, TA = -25 to 85°C, VDDext = 1.7V - 3.6V)

Parameter	Symbol	Minimum	Typ	Maximum	Unit
XciVSYNC(CAM_VSYNC_A) input Setup time	Tssvsynca	0	-	PA - 0.26	ns
XciVSYNC(CAM_VSYNC_A) input Hold time	Thvsynca	0.26	-	PA	ns
XciHREF(CAM_HREF_A) input Setup time	Tsshrefa	0.35	-	PA	ns
XciHREF(CAM_HREF_A) input Hold time	Thhrefa	0	-	PA - 0.35	ns
XciDATA(CAM_DATA_A) input Setup time	Tssdataa	0	-	PA - 4.8	ns
XciDATA(CAM_DATA_A) input Hold time	Thdataa	4.8	-	PA	ns

NOTE: PA denotes period (ns) of CAM_PCLK_A

Parameter	Symbol	Minimum	Typ	Maximum	Unit
XmsmADDR[9](CAM_VSYNC_B) input Setup time	Tssvsyncb	0	-	PB	ns
XmsmADDR[9] (CAM_VSYNC_B) input Hold time	Thvsyncb	0	-	PB	ns
XmsmADDR[10] (CAM_HREF_B) input Setup time	Tsshrefb	0.08	-	PB	ns
XmsmADDR[10] (CAM_HREF_B) input Hold time	Thhrefb	0	-	PB - 0.08	ns
XmsmADDR[7:0] (CAM_DATA_B) input Setup time	Tssdatab	0	-	PB - 4.93	ns
XmsmADDR[7:0] (CAM_DATA_B) input Hold time	Thdatab	4.93	-	PB	ns

NOTE: PB denotes period (ns) of CAM_PCLK_B

Parameter	Symbol	Minimum	Typ	Maximum	Unit
PCLK period	tPCLK	12	-	-	ns
PCLK input high level pulse width	tPCLKHIGH	3	-	-	ns
PCLK input low level pulse width	tPCLKLOW	3	-	-	ns

1.12 SDMMC AC ELECTRICAL CHARACTERISTICS

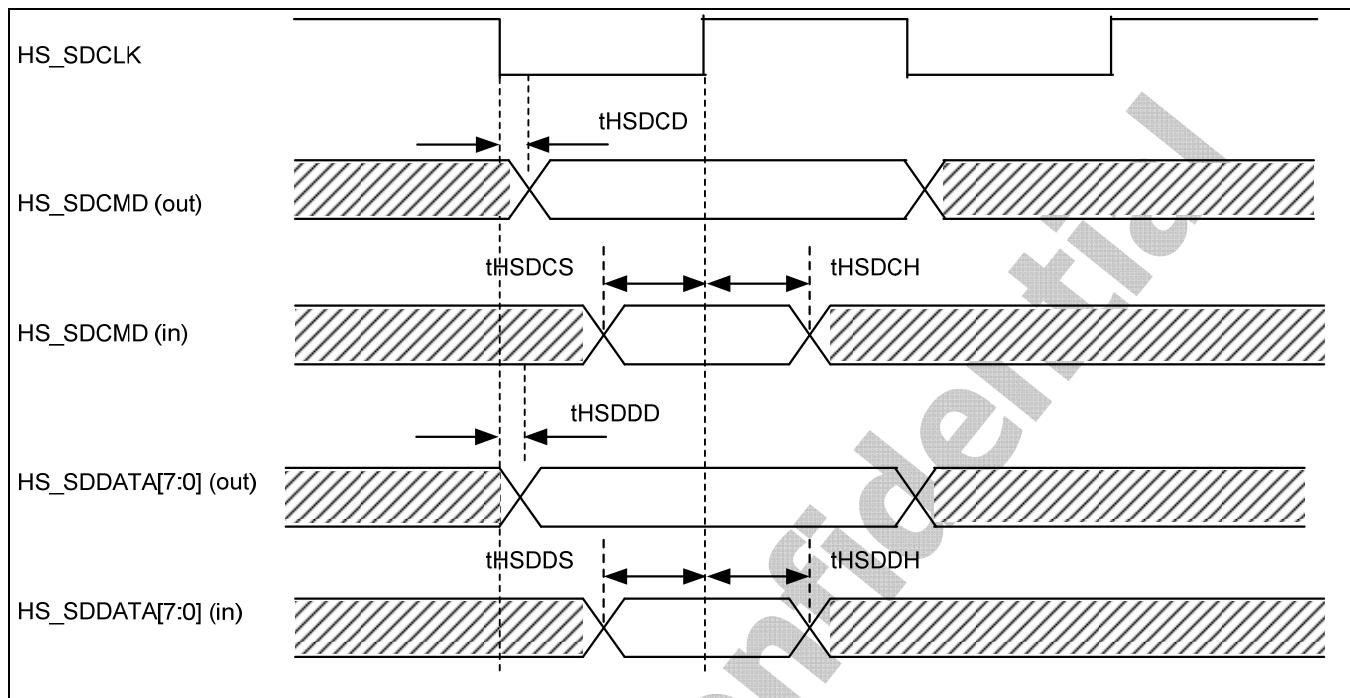


Figure 1-15 High Speed SDMMC Interface Timing

Table 1-15 High Speed SDMMC Interface Transmit/Receive Timing Constants

(VDDINT = 1.1V ± 5%, TA = -25 to 85°C, VDDmmc = 3.3V ± 5%, 2.5V ± 5%, 1.8V ± 5%)

Parameter	Symbol	Minimum	Typical	Maximum	Unit
SD Command output Delay time	t_{SDCD}	-	-	4.0	ns
SD Command input Setup time	t_{SDCS}	4.0	-	-	ns
SD Command input Hold time	t_{SDCH}	0	-	-	ns
SD Data output Delay time	t_{SDDD}	-	-	4.0	ns
SD Data input Setup time	t_{SDDS}	4.0	-	-	ns
SD Data input Hold time	t_{SDDH}	0	-	-	ns

1.13 SPI AC ELECTRICAL CHARACTERISTICS

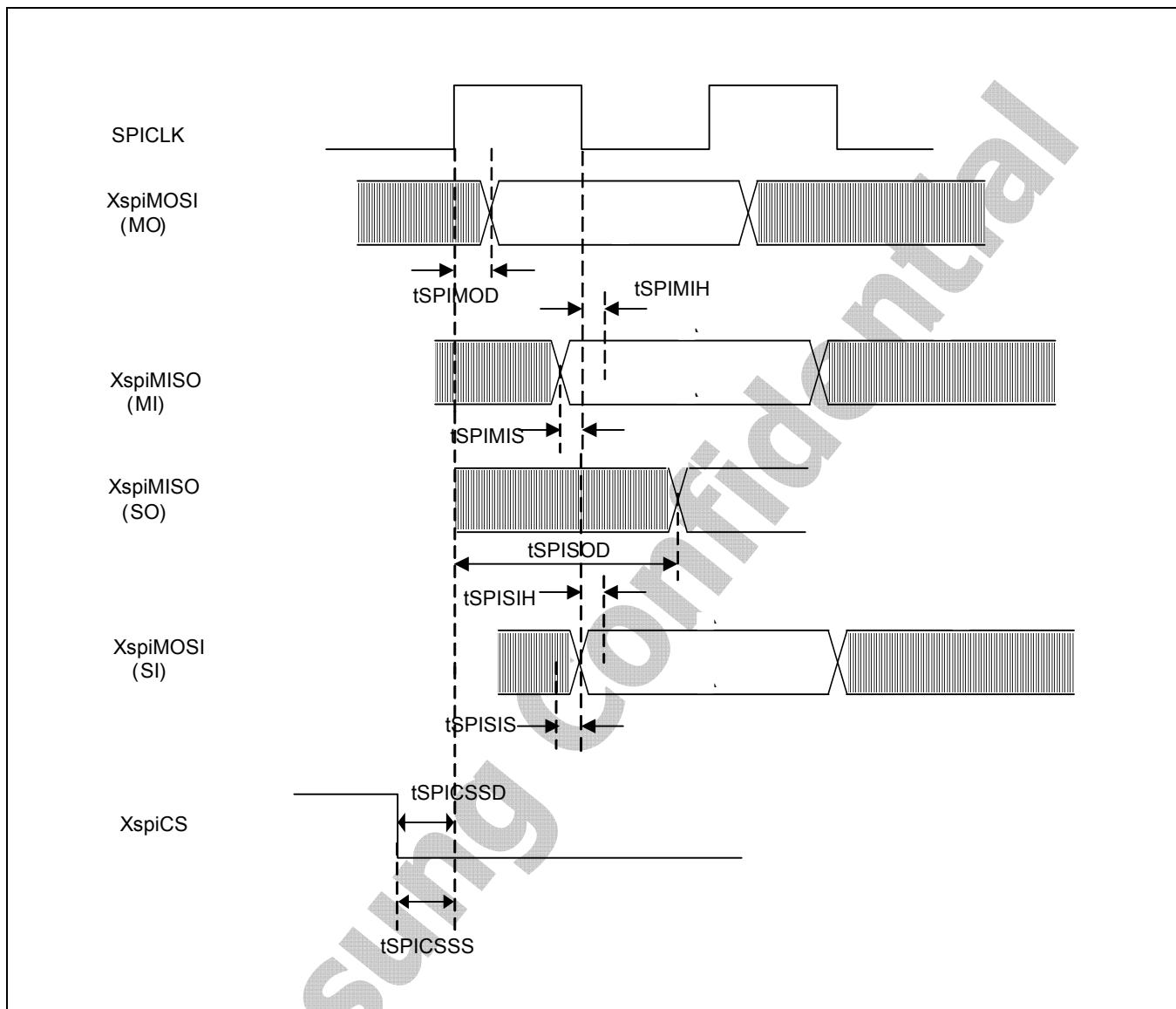


Figure 1-16 SPI Interface Timing (CPHA = 0, CPOL = 1)

Table 1-16 SPI Interface Transmit/ Receive Timing Constants

(VDDINT = 1.1V ± 5%, TA = -25 to 85°C, VDDext = 1.8V ± 10%, load = 15pF)

Parameter	Symbol	Minimum	Typical	Maximum	Unit
SPI MOSI Master Output Delay time	t_{SPIMOD}	-	-	5	ns
SPI MISO Master Input Setup time (FB_CLK_SEL = 00)	t_{SPIMIS}	12	-	-	ns
SPI MISO Master Input Setup time (FB_CLK_SEL = 01)		7	-	-	ns
SPI MISO Master Input Setup time (FB_CLK_SEL = 10)		2	-	-	ns
SPI MISO Master Input Setup time (FB_CLK_SEL = 11)		-3	-	-	ns
SPI MISO Master Input Hold time	t_{SPIMIH}	5	-	-	ns
SPI MOSI Slave Input Setup time	t_{SPISIS}	2	-	-	ns
SPI MOSI Slave Input Hold time	t_{SPISIH}	5	-	-	ns
SPI MISO Slave Output Delay time	t_{SPISOD}	-	-	17	ns
SPI nSS Master Output Delay time	$t_{SPICSSD}$	7	-	-	ns
SPI nSS Slave Input Setup time	$t_{SPICSSS}$	5	-	-	ns
SPI MOSI Master Output Delay time	t_{SPIMOD}	-	-	4	ns
SPI MISO Master Input Setup time (FB_CLK_SEL = 00)	t_{SPIMIS}	13	-	-	ns
SPI MISO Master Input Setup time (FB_CLK_SEL = 01)		8	-	-	ns
SPI MISO Master Input Setup time (FB_CLK_SEL = 10)		3	-	-	ns
SPI MISO Master Input Setup time (FB_CLK_SEL = 11)		-2	-	-	ns
SPI MISO Master Input Hold time	t_{SPIMIH}	5	-	-	ns
SPI MOSI Slave Input Setup time	t_{SPISIS}	3	-	-	ns
SPI MOSI Slave Input Hold time	t_{SPISIH}	5	-	-	ns
SPI MISO Slave Output Delay time	t_{SPISOD}	-	-	18	ns
SPI nSS Master Output Delay time	$t_{SPICSSD}$	7	-	-	ns
SPI nSS Slave Input Setup time	$t_{SPICSSS}$	5	-	-	ns

NOTE: SPICLKout = 50MHz

$$t_{SPIMIS,CH0} = 12 - (\text{cycle period} / 4) \times \text{FB_CLK_SEL}$$

$$t_{SPIMIS,CH1} = 13 - (\text{cycle period} / 4) \times \text{FB_CLK_SEL}$$

(VDDINT = 1.1V ± 5%, TA = -25 to 85°C, VDDext = 3.3V ± 10%, load = 30pF)

Parameter		Symbol	Minimum	Typical	Maximum	Unit
Ch 0	SPI MOSI Master Output Delay time	t_{SPIMOD}	-	-	6	ns
	SPI MISO Master Input Setup time (FB_CLK_SEL = 00)	t_{SPIMIS}	13	-	-	ns
	SPI MISO Master Input Setup time (FB_CLK_SEL = 01)		8	-	-	ns
	SPI MISO Master Input Setup time (FB_CLK_SEL = 10)		3	-	-	ns
	SPI MISO Master Input Setup time (FB_CLK_SEL = 11)		-2	-	-	ns
	SPI MISO Master Input Hold time	t_{SPIMIH}	5	-	-	ns
	SPI MOSI Slave Input Setup time	t_{SPISIS}	4	-	-	ns
	SPI MOSI Slave Input Hold time	t_{SPISIH}	5	-	-	ns
	SPI MISO Slave Output Delay time	t_{SPISOD}	-	-	18	ns
	SPI nSS Master Output Delay time	$t_{SPICSSD}$	8	-	-	ns
Ch 1	SPI MOSI Master Output Delay time	t_{SPIMOD}	-	-	5	ns
	SPI MISO Master Input Setup time (FB_CLK_SEL = 00)	t_{SPIMIS}	14	-	-	ns
	SPI MISO Master Input Setup time (FB_CLK_SEL = 01)		9	-	-	ns
	SPI MISO Master Input Setup time (FB_CLK_SEL = 10)		4	-	-	ns
	SPI MISO Master Input Setup time (FB_CLK_SEL = 11)		-1	-	-	ns
	SPI MISO Master Input Hold time	t_{SPIMIH}	5	-	-	ns
	SPI MOSI Slave Input Setup time	t_{SPISIS}	4	-	-	ns
	SPI MOSI Slave Input Hold time	t_{SPISIH}	5	-	-	ns
	SPI MISO Slave Output Delay time	t_{SPISOD}	-	-	19	ns
	SPI nSS Master Output Delay time	$t_{SPICSSD}$	8	-	-	ns
	SPI nSS Slave Input Setup time	$t_{SPICSSS}$	6	-	-	ns

NOTE: SPICLKout = 50MHz
 $t_{SPIMIS,CH0} = 12 - (\text{cycle period} / 4) \times \text{FB_CLK_SEL}$
 $t_{SPIMIS,CH1} = 13 - (\text{cycle period} / 4) \times \text{FB_CLK_SEL}$

1.14 I2C AC ELECTRICAL CHARACTERISTICS

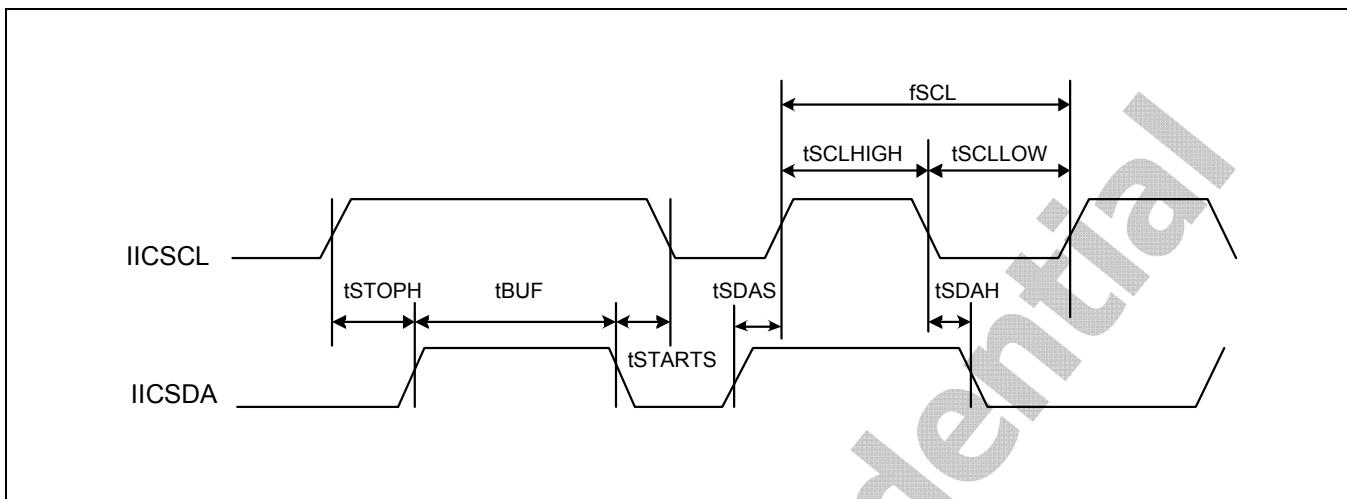


Figure 1-17 IIC Interface Timing

Table 1-17 IIC BUS Controller Module Signal Timing

(VDDINT = 1.1V ± 5%, TA = -25 to 85°C, VDDext = 3.3V ± 10%)

Parameter	Symbol	Minimum	Typical	Maximum	Unit
SCL clock frequency	f_{SCL}	-	-	std. 100 fast 400	kHz
SCL high level pulse width	$t_{SCLHIGH}$	std. 4.0 fast 0.6	-	-	us
SCL low level pulse width	t_{SCLLOW}	std. 4.7 fast 1.3	-	-	us
Bus free time between STOP and START	t_{BUF}	std 4.7 fast 1.3	-	-	us
START hold time	t_{STARTS}	std. 4.0 fast 0.6	-	-	us
SDA hold time	t_{SDAH}	std. 0 fast 0	-	std.-fast 0.9	us
SDA setup time	t_{SDAS}	std. 250 fast 100	-	-	ns
STOP setup time	t_{STOPH}	std. 4.0 fast 0.6	-	-	us

NOTE: std. refers to Standard Mode and fast refers to Fast Mode.

1. The IIC data hold time (t_{SDAH}) is minimum 0ns.
(IIC data hold time is minimum 0ns for standard/ fast bus mode IIC specification v2.1)
Check whether the data hold time of your IIC device is 0 ns or not.
2. The IIC controller supports IIC bus device only (standard/fast bus mode), and does not support C bus device.

1.15 TSI AC ELECTRICAL CHARACTERISTICS

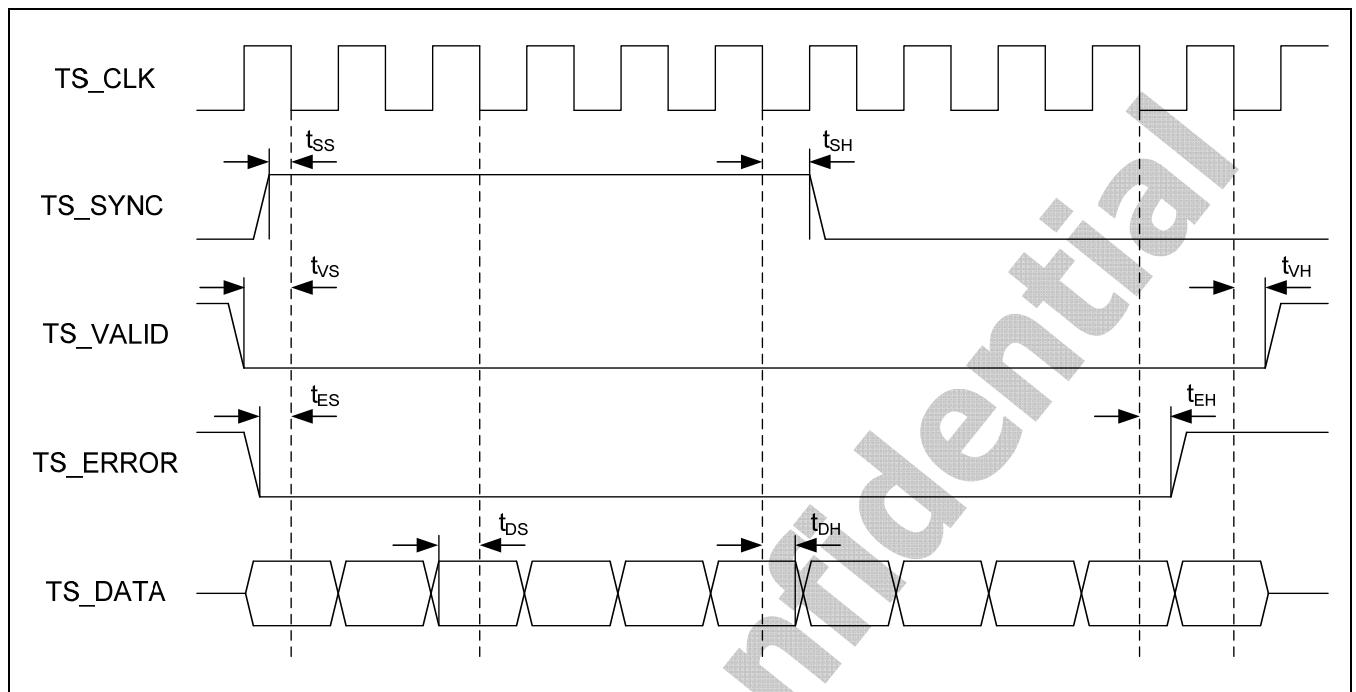


Figure 1-18 Transport Stream Interface Timing

Table 1-18 Transport Stream Interface Timing Constants

(VDDINT = 1.1V ± 5%, TA = -25 to 85°C, VDDext = 1.8V ± 10%, load = 35pF)

Parameter	Symbol	Minimum	Maximum	Unit
TSI synchronization signal setup time	t _{SS}	3	-	ns
TSI synchronization signal hold time	t _{SH}	3	-	ns
TSI valid signal setup time	t _{VS}	3	-	ns
TSI valid signal hold time	t _{VH}	3	-	ns
TSI error signal setup time	t _{ES}	3	-	ns
TSI error signal hold time	t _{EH}	3	-	ns
TSI input data setup time	t _{DS}	3	-	ns
TSI input data hold time	t _{DH}	3	-	ns