

Offset: 0x000C			Register Name: UART_LCR
Bit	Read/Write	Default/Hex	Description
1:0	R/W	0x0	<p>DLS Data Length Select</p> <p>It is writable only when UART is not busy (UART_USR[0] is 0) and always readable. This is used to select the count of bits in a transmitted or received frame.</p> <p>00: 5 bits 01: 6 bits 10: 7 bits 11: 8 bits</p>

9.2.6.9 0x0010 UART Modem Control Register (Default Value: 0x0000\_0000)

Offset: 0x0010			Register Name: UART_MCR
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:6	R/W	0x0	<p>UART_FUNCTION Select IrDA or RS485</p> <p>00: UART Mode 01: IrDA SIR Mode 10: RS485 Mode 11: Reserved</p>
5	R/W	0x0	<p>AFCE Auto Flow Control Enable</p> <p>When FIFOs are enabled and the AFCE bit is set, the AutoFlow Control is enabled.</p> <p>0: Auto Flow Control mode disabled 1: Auto Flow Control mode enabled</p>

Offset: 0x0010			Register Name: UART_MCR
Bit	Read/Write	Default/Hex	Description
4	R/W	0x0	<p>LOOP</p> <p>Loop Back Mode</p> <p>0: Normal Mode</p> <p>1: Loop Back Mode</p> <p>This is used to put the UART into a diagnostic mode for test purposes. If operating in UART mode (SIR_MODE != Enabled or not active, UART_MCR[6] is set to 0), the data on the SOUT line is held high, while serial data output is looped back to the sin line, internally. In this mode, all the interrupts are fully functional. Also, in loopback mode, the modem control inputs (dsr_n, cts_n, ri_n, dcd_n) are disconnected and the modem control outputs (dtr_n, rts_n, out1_n, out2_n) are looped back to the inputs, internally. If operating in infrared mode (SIR_MODE == Enabled AND active, UART_MCR[6] is set to 1), the data on the sir_out_n line is held low, while serial data output is inverted and looped back to the sir_in line.</p>
3:2	/	/	/
1	R/W	0x0	<p>RTS</p> <p>Request to Send</p> <p>This is used to directly control the Request to Send (rts_n) output. The RTS (rts_n) output is used to inform the modem or data set that the UART is ready to exchange data. When Auto RTS Flow Control is not enabled (UART_MCR[5] is set to 0), the rts_n signal is set low by programming UART_MCR[1] (RTS) to a high. In Auto Flow Control, AFCE_MODE == Enabled and active (UART_MCR[5] is set to 1) and FIFOs enable (UART_FCR[0] is set to 1), the rts_n output is controlled in the same way, but is also gated with the receiver FIFO threshold trigger (rts_n is inactive high when above the threshold). The rts_n signal is de-asserted when UART_MCR[1] is set low.</p> <p>0: rts_n de-asserted (logic 1)</p> <p>1: rts_n asserted (logic 0)</p> <p>Note that in Loopback mode (UART_MCR[4] is set to 1), the rts_n output is held inactive high while the value of this location is internally looped back to an input.</p>

Offset: 0x0010			Register Name: UART_MCR
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	<p>DTR Data Terminal Ready</p> <p>This is used to directly control the Data Terminal Ready (dtr_n) output. The value written to this location is inverted and driven out on dtr_n.</p> <p>0: dtr_n de-asserted (logic 1) 1: dtr_n asserted (logic 0)</p> <p>The DTR output is used to inform the modem or data set that the UART is ready to establish communications.</p> <p>Note that in Loopback mode (UART_MCR[4] is set to 1), the dtr_n output is held inactive high while the value of this location is internally looped back to an input.</p>

9.2.6.10 0x0014 UART Line Status Register (Default Value: 0x0000\_0060)

Offset:0x0014			Register Name: UART_LSR
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R	0x0	<p>FIFOERR RX Data Error in FIFO</p> <p>When FIFOs are disabled, this bit is always 0. When FIFOs are enabled, this bit is set to "1" when there is at least one PE, FE, or BI in the RX FIFO. It is cleared by reading from the UART_LSR register, there are no subsequent errors in the FIFO.</p>
6	R	0x1	<p>TEMT Transmitter Empty</p> <p>If the FIFOs are disabled, this bit is set to "1" whenever the TX Holding Register (UART_THR) and the TX Shift Register are empty. If the FIFOs are enabled, this bit is set whenever the TX FIFO and the TX Shift Register are empty. In both cases, this bit is cleared when a byte is written to the TX data channel.</p>

Offset:0x0014			Register Name: UART_LSR
Bit	Read/Write	Default/Hex	Description
5	R	0x1	<p>THRE</p> <p>TX Holding Register Empty</p> <p>If the FIFOs are disabled, this bit is set to "1" when the TX Holding Register (UART_THR) is empty and ready to accept new data and it is cleared when the CPU writes to the TX Holding Register.</p> <p>If the FIFOs are enabled, this bit is set to "1" whenever the TX FIFO is empty and it is cleared when at least one byte is written to the TX FIFO.</p>
4	R	0x0	<p>BI</p> <p>Break Interrupt</p> <p>This is used to indicate the detection of a break sequence on the serial input data.</p> <p>If in UART mode (SIR_MODE == Disabled), it is set when the serial input, <i>sir_in</i>, is held in a logic '0' state for longer than the sum of <i>start time + data bits + parity + stop bits</i>.</p> <p>If in infrared mode (SIR_MODE == Enabled), it is set when the serial input, <i>sir_in</i>, is continuously pulsed to logic '0' for longer than the sum of <i>start time + data bits + parity + stop bits</i>. A break condition on serial input causes one and only one character, consisting of all zeros, to be received by the UART.</p> <p>In the FIFO mode, the character associated with the break condition is carried through the FIFO and is revealed when the character is at the top of the FIFO. Reading the UART_LSR clears the BI bit. In the non-FIFO mode, the BI indication occurs immediately and persists until the UART_LSR is read.</p>

Offset:0x0014			Register Name: UART_LSR
Bit	Read/Write	Default/Hex	Description
3	RC	0x0	<p>FE</p> <p>Framing Error</p> <p>This is used to indicate the occurrence of a framing error in the receiver. A framing error occurs when the receiver does not detect a valid STOP bit in the received data.</p> <p>In the FIFO mode, since the framing error is associated with a character received, it is revealed when the character with the framing error is at the top of the FIFO. When a framing error occurs, the UART tries to resynchronize. It does this by assuming that the error was due to the start bit of the next character and then continues receiving the other bit i.e. data, and/or parity and stop. It should be noted that the Framing Error (FE) bit (UART_LSR[3]) is set if a break interrupt has occurred, as indicated by Break Interrupt (BI) bit (UART_LSR[4]).</p> <p>0: no framing error 1: framing error</p> <p>Reading the UART_LSR clears the FE bit.</p>
2	RC	0x0	<p>PE</p> <p>Parity Error</p> <p>This is used to indicate the occurrence of a parity error in the receiver if the Parity Enable (PEN) bit (UART_LCR[3]) is set. In the FIFO mode, since the parity error is associated with a character received, it is revealed when the character with the parity error arrives at the top of the FIFO. It should be noted that the Parity Error (PE) bit (UART_LSR[2]) is set if a break interrupt has occurred, as indicated by Break Interrupt (BI) bit (UART_LSR[4]).</p> <p>0: no parity error 1: parity error</p> <p>Reading the UART_LSR clears the PE bit.</p>

Offset:0x0014			Register Name: UART_LSR
Bit	Read/Write	Default/Hex	Description
1	RC	0x0	<p>OE Overrun Error</p> <p>This occurs if a new data character was received before the previous data was read. In the non-FIFO mode, the OE bit is set when a new character arrives in the receiver before the previous character was read from the UART_RBR. When this happens, the data in the UART_RBR is overwritten. In the FIFO mode, an overrun error occurs when the FIFO is full and a new character arrives at the receiver. The data in the FIFO is retained and the data in the receive shift register is lost.</p> <p>0: no overrun error 1: overrun error</p> <p>Reading the UART_LSR clears the OE bit.</p>
0	R	0x0	<p>DR Data Ready</p> <p>This is used to indicate that the receiver contains at least one character in the UART_RBR or the receiver FIFO.</p> <p>0: no data ready 1: data ready</p> <p>This bit is cleared when the UART_RBR is read in non-FIFO mode, or when the receiver FIFO is empty, in FIFO mode.</p> <p><b>Note: Not use when the RXDMA master is enabled (rxdma_ctrl[0] is set to 1).</b></p>

9.2.6.11 0x0018 UART Modem Status Register (Default Value: 0x0000\_0000)

Offset: 0x0018			Register Name: UART_MSR
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R	0x0	<p>DCD Line State of Data Carrier Detect</p> <p>This is used to indicate the current state of the modem control line dcd_n. This bit is the complement of dcd_n. When the Data Carrier Detect input (dcd_n) is asserted it is an indication that the carrier has been detected by the modem or data set.</p> <p>0: dcd_n input is de-asserted (logic 1) 1: dcd_n input is asserted (logic 0)</p>

Offset: 0x0018			Register Name: UART_MSR
Bit	Read/Write	Default/Hex	Description
6	R	0x0	<p>RI Line State of Ring Indicator</p> <p>This is used to indicate the current state of the modem control line ri_n. This bit is the complement of ri_n. When the Ring Indicator input (ri_n) is asserted it is an indication that a telephone ringing signal has been received by setting the modem or data.</p> <p>0: ri_n input is de-asserted (logic 1) 1: ri_n input is asserted (logic 0)</p>
5	R	0x0	<p>DSR Line State of Data Set Ready</p> <p>This is used to indicate the current state of the modem control line dsr_n. This bit is the complement of the dsr_n. When the Data Set Ready input (dsr_n) is asserted, it is an indication that the modem or data set is ready to establish communication with UART.</p> <p>0: dsr_n input is de-asserted (logic 1) 1: dsr_n input is asserted (logic 0)</p> <p>In Loopback Mode (UART_MCR[4] is set to 1), the DSR is the same as the DTR (UART_MCR[0]).</p>
4	R	0x0	<p>CTS Line State of Clear To Send</p> <p>This is used to indicate the current state of the modem control line cts_n. This bit is the complement of cts_n. When the Clear to Send input (cts_n) is asserted, it is an indication that the modem or data set is ready to exchange data with UART.</p> <p>0: cts_n input is de-asserted (logic 1) 1: cts_n input is asserted (logic 0)</p> <p>In Loopback Mode (UART_MCR[4] = 1), the CTS is the same as the RTS (UART_MCR[1]).</p>

Offset: 0x0018			Register Name: UART_MSR
Bit	Read/Write	Default/Hex	Description
3	RC	0x0	<p>DDCD Delta Data Carrier Detect</p> <p>This is used to indicate that the modem control line dcd_n has changed since the last time the UART_MSR was read.</p> <p>0: no change on dcd_n since the last read of UART_MSR 1: change on dcd_n since the last read of UART_MSR</p> <p>Reading the UART_MSR clears the DDCD bit.</p> <p><b>Note: If the DDCD bit is not set and the dcd_n signal is asserted (low) and a reset occurs, then the DDCD bit is set when the reset is removed if the dcd_n signal remains asserted.</b></p>
2	RC	0x0	<p>TERI Trailing Edge Ring Indicator</p> <p>This is used to indicate that a change in the input ri_n (from an active-low to an inactive-high state) has occurred since the last time the UART_MSR was read.</p> <p>0: no change on ri_n since the last read of UART_MSR 1: change on ri_n since the last read of UART_MSR</p> <p>Reading the UART_MSR clears the TERI bit.</p>
1	RC	0x0	<p>DDSR Delta Data Set Ready</p> <p>This is used to indicate that the modem control line dsr_n has changed since the last time the UART_MSR was read.</p> <p>0: no change on dsr_n since the last read of UART_MSR 1: change on dsr_n since the last read of UART_MSR</p> <p>Reading the UART_MSR clears the DDSR bit. In Loopback Mode (UART_MCR[4] = 1), the DDSR reflects changes on the DTR (UART_MCR[0]).</p> <p><b>Note: If the DDSR bit is not set and the dsr_n signal is asserted (low) and a reset occurs, then the DDSR bit is set when the reset is removed if the dsr_n signal remains asserted.</b></p>



Offset: 0x0018			Register Name: UART_MSR
Bit	Read/Write	Default/Hex	Description
0	RC	0x0	<p>DCTS Delta Clear to Send</p> <p>This is used to indicate that the modem control line cts_n has changed since the last time the UART_MSR was read.</p> <p>0: no change on ctsdsr_n since the last read of UART_MSR 1: change on ctsdsr_n since the last read of UART_MSR</p> <p>Reading the UART_MSR clears the DCTS bit. In Loopback Mode (UART_MCR[4] = 1), the DCTS reflects changes on the RTS (UART_MCR[1]).</p> <p><b>Note: If the DCTS bit is not set and the cts_n signal is asserted (low) and a reset occurs, then the DCTS bit is set when the reset is removed if the cts_n signal remains asserted.</b></p>

9.2.6.12 0x001C UART Scratch Register (Default Value: 0x0000\_0000)

Offset: 0x001C			Register Name: UART_SCH
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0x0	<p>SCRATCH_REG Scratch Register</p> <p>This register is for programmers to use as a temporary storage space. It has no defined purpose in the UART.</p>

9.2.6.13 0x007C UART Status Register (Default Value: 0x0000\_0006)

Offset: 0x007C			Register Name: UART_USR
Bit	Read/Write	Default/Hex	Description
31:5	/	/	/
4	R	0x0	<p>RFF RX FIFO Full</p> <p>This is used to indicate that the RX FIFO is completely full.</p> <p>0: RX FIFO not full 1: RX FIFO Full</p> <p>This bit is cleared when the RX FIFO is no longer full.</p>

Offset: 0x007C			Register Name: UART_USR
Bit	Read/Write	Default/Hex	Description
3	R	0x0	<p>RFNE RX FIFO Not Empty</p> <p>This is used to indicate that the RX FIFO contains one or more entries.</p> <p>0: RX FIFO is empty 1: RX FIFO is not empty</p> <p>This bit is cleared when the RX FIFO is empty.</p>
2	R	0x1	<p>TFE TX FIFO Empty</p> <p>This is used to indicate that the TX FIFO is completely empty.</p> <p>0: TX FIFO is not empty 1: TX FIFO is empty</p> <p>This bit is cleared when the TX FIFO is no longer empty.</p>
1	R	0x1	<p>TFNF TX FIFO Not Full</p> <p>This is used to indicate that the TX FIFO is not full.</p> <p>0: TX FIFO is full 1: TX FIFO is not full</p> <p>This bit is cleared when the TX FIFO is full.</p>
0	R	0x0	<p>BUSY UART Busy Bit</p> <p>0: Idle or inactive 1: Busy</p>

**9.2.6.14 0x0080 UART Transmit FIFO Level Register (Default Value: 0x0000\_0000)**

Offset: 0x0080			Register Name: UART_TFL
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8:0	R	0x0	<p>TFL TX FIFO Level</p> <p>The bit indicates the number of data entries in the TX FIFO.</p>

9.2.6.15 0x0084 UART Receive FIFO Level Register (Default Value: 0x0000\_0000)

Offset: 0x0084			Register Name: UART_RFL
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8:0	R	0x0	RFL RX FIFO Level The bit indicates the number of data entries in the RX FIFO. <b>Note: Not use when the RXDMA master is enabled (UART_RXDMA_CTRL[0] is set to 1).</b>

9.2.6.16 0x0088 UART DMA Handshake Configuration Register (Default Value: 0x0000\_00A5)

Offset: 0x0088			Register Name: UART_HSK
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0xA5	Handshake configuration 0xA5: DMA wait cycle mode 0xE5: DMA handshake mode

9.2.6.17 0x008C UART DMA Request Enable Register(Default Value: 0x0000\_0003)

Offset: 0x008C			Register Name: UART_DMA_REQ_EN
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/
2	R/W	0x0	DMA Timeout Enable 0: Disable 1: Enable
1	R/W	0x1	DMA TX REQ Enable 0: Disable 1: Enable
0	R/W	0x1	DMA RX REQ Enable 0: Disable 1: Enable

9.2.6.18 0x00A4 UART Halt TX Register (Default Value: 0x0000\_0000)

Offset: 0x00A4			Register Name: UART_HALT
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	<p>PTE</p> <p>The sending of TX_REQ</p> <p>In DMA1 mode (FIFO on), if the PTE is set to 1 when the TFL in UART_TFL is less than or equal to the trigger value, the controller sends the DMA request. If the PTE is set to 0, when FIFO is empty, the controller sends the DMA request. The DMA request will stop when FIFO is full.</p> <p>In DMA0 mode, if the PTE is set to 1 and FIFO is on, when the TFL in UART_TFL is less than or equal to the trigger value, the controller sends DMA request. If the PTE is set to 1 and FIFO off, when the THR in UART_THR is empty, the controller sends DMA request. If the PTE is set to 0, when FIFO(FIFO Enable) or THR(FIFO Enable) is empty, the controller sends DMA request. Otherwise, the DMA request is cleared.</p>
6	R/W	0x0	<p>DMA_PTE_RX</p> <p>The Transmission of RX_DRQ</p> <p>In DMA1 mode, when RFL is more than or equal to the trigger value, or a receive timeout has occurred, the controller sends DRQ.</p> <p>In DMA0 mode, when DMA_PTE_RX = 1 and FIFO is on, if RFL is more than or equal to trig, the controller sends DRQ, else DRQ is cleared. In other cases, once the received data is valid, the controller sends DRQ.</p>
5	R/W	0x0	<p>SIR_RX_INVERT</p> <p>SIR RX Pulse Polarity Invert</p> <p>0: Not invert receiver signal</p> <p>1: Invert receiver signal</p>
4	R/W	0x0	<p>SIR_TX_INVERT</p> <p>SIR TX Pulse Polarity Invert</p> <p>0: Not invert transmit pulse</p> <p>1: Invert transmit pulse</p>
3	/	/	/

Offset: 0x00A4			Register Name: UART_HALT
Bit	Read/Write	Default/Hex	Description
2	R/WAC	0x0	<p>CHANGE_UPDATE</p> <p>After the user uses UART_HALT[1] to change the baud rate or LCR configuration, write 1 to update the configuration and wait this bit to self-clear to 0 to finish update process. Writing 0 to this bit has no effect.</p> <p>1: Update trigger, self-clear to 0 when finish update.</p>
1	R/W	0x0	<p>CHCFG_AT_BUSY</p> <p>This is an enable bit for the user to change LCR register configuration and baud rate register (UART_DLH and UART_DLL) when the UART is busy.</p> <p>1: Enable change when busy</p>
0	R/W	0x0	<p>HALT_TX</p> <p>Halt TX</p> <p>This register is used to halt transmissions for testing, so that the transmit FIFO can be filled by the master when FIFOs are implemented and enabled.</p> <p>0 : Halt TX disabled 1 : Halt TX enabled</p> <p><b>Note: If FIFOs are not enabled, the setting has no effect on operation.</b></p>

9.2.6.19 0x00B0 UART DBG DLL Register (Default Value: 0x0000\_0000)

Offset: 0x00B0			Register Name: UART_DBG_DLL
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R	0x0	DEBUG DLL

9.2.6.20 0x00B4 UART DBG DLH Register (Default Value: 0x0000\_0000)

Offset: 0x00B4			Register Name: UART_DBG_DLH
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R	0x0	DEBUG DLH

9.2.6.21 0x00F0 UART FIFO Clock Control Register (Default Value: 0x0000\_0003)

Offset: 0x00F0			Register Name: UART_FCC
Bit	Read/Write	Default/Hex	Description
31:8	R	0x0	FIFO Depth Indicates the depth of TX/RX FIFO
7:3	/	/	/
2	R/W	0x0	RXFIFO Clock Mode 0: Sync mode, writing/reading clocks use apb clock 1: Sync mode, writing clock uses apb clock, reading clock uses ahb clock
1	R/W	0x1	TX FIFO Clock Enable 0: Clock disable 1: Clock enable
0	R/W	0x1	RX FIFO Clock Enable 0: Clock disable 1: Clock enable

9.2.6.22 0x0100 UART RXDMA Control Register (Default Value: 0x0000\_0000)

Offset: 0x0100			Register Name: UART_RXDMA_CTRL
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:8	R/W	0x0	RXDMA Timeout Threshold Unit is 1 UART bit time Note that this field is only configurable when RXDMA Busy is 0.
7	/	/	/
6	R/W	0x0	RXDMA Timeout Enable Once enable, the DMA starts a transfer even the data entries in RX FIFO do not reach BLK_SIZE. Note that this field is only configurable when RXDMA Busy is 0.

Offset: 0x0100			Register Name: UART_RXDMA_CTRL
Bit	Read/Write	Default/Hex	Description
5:4	R/W	0x0	<p>RXDMA AHB Burst Mode</p> <p>Set for AHB port burst supported</p> <p>INCR8 is recommended, while INCR16 may be unsupported due to the system bus.</p> <p>00: SIGNLE</p> <p>01: INCR4</p> <p>10: INCR8</p> <p>11: INCR16</p> <p>Note that this field is only configurable while RXDMA Busy is 0.</p>
3:2	R/W	0x0	<p>RXDMA BLK SIZE</p> <p>Every time when data entries in RX FIFO reach BLK_SIZE, start a DMA block transfer. It is recommended that the block size no more than RX FIFO Depth.</p> <p>00: 8 bytes</p> <p>01: 16 bytes</p> <p>10: 32 bytes</p> <p>11: 64 bytes</p> <p>Note that this field is only configurable while RXDMA Busy is 0.</p>
1	R/W	0x0	<p>RXDMA Mode</p> <p>0: Continuous</p> <p>1: Limited</p> <p>When data transferred reaches the limited count set in RXDMA LIMIT, the DMA stops and the RXDMA Start bit is cleared automatically.</p> <p>Note that this field is only configurable while RXDMA Busy is 0.</p>
0	R/W	0x0	<p>RXDMA Enable</p> <p>0: RXDMA Disable</p> <p>1: RXDMA Enable</p> <p>Note that if the software turns off this bit, the RXDMA will stop after the current block transfer completes, then the software should do a reset to the RX FIFO before re-enable.</p>

9.2.6.23 0x0104 UART RXDMA Start Register (Default Value: 0x0000\_0000)

Offset: 0x0104			Register Name: UART_RXDMA_STR
Bit	Read/Write	Default/Hex	Description
31:2	/	/	/
0	R/WAC	0x0	RXDMA Start Only valid when RXDMA mode is set to 1, it is auto cleared when data transferred reaches the RXDMA Limit Size.

9.2.6.24 0x0108 UART RXDMA Status Register (Default Value: 0x0000\_0000)

Offset: 0x0108			Register Name: UART_RXDMA_STA
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
1	R	0x0	Buffer Read Address Updating 0: Buffer Read Address Register is ready for updating 1: Buffer Read Address Register is busy for updating The software should not update Buffer Read Address Register until this bit is 0.
0	R	0x0	RXDMA BUSY 0: RXDMA is idle 1: RXDMA is busy

9.2.6.25 0x010C UART RXDMA Limit Register (Default Value: 0x0000\_0000)

Offset: 0x010C			Register Name: UART_RXDMA_LMT
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x0	RXDMA Limit Size Only valid when RXDMA Mode is set to 1, and the unit is byte. Note that this field is only configurable while RXDMA Busy is 0.



9.2.6.26 0x0110 UART RXDMA Buffer Start Address Low Register (Default Value: 0x0000\_0000)

Offset: 0x0110			Register Name: UART_RXDMA_SADDRL
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	RXDMA Buffer Start Address [31:0] Byte address Note that this field is only configurable while RXDMA Busy is 0.

9.2.6.27 0x0114 UART RXDMA Buffer Start Address High Register (Default Value: 0x0000\_0000)

Offset: 0x0114			Register Name: UART_RXDMA_SADDRH
Bit	Read/Write	Default/Hex	Description
31:2	/	/	/
1:0	R/W	0x0	RXDMA Buffer Start Address [33:32] Byte address Note that this field is only configurable while RXDMA Busy is 0.

9.2.6.28 0x0118 UART RXDMA Buffer Length Register (Default Value: 0x0000\_0000)

Offset: 0x0118			Register Name: UART_RXDMA_BL
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x0	RXDMA Buffer Length Unit is byte Note that this field is only configurable while RXDMA Busy is 0.

9.2.6.29 0x0120 UART RXDMA Interrupt Enable Register (Default Value: 0x0000\_0000)

Offset: 0x0120			Register Name: UART_RXDMA_IE
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
3	R/W	0x0	RXDMA Buffer Overrun RXDMA Buffer Overrun Interrupt Enable

Offset: 0x0120			Register Name: UART_RXDMA_IE
Bit	Read/Write	Default/Hex	Description
2	R/W	0x0	RXDMA Timeout Done RXDMA Timeout Done Interrupt Enable
1	R/W	0x0	RXDMA BLK Done RXDMA BLK Done Interrupt Enable
0	R/W	0x0	RXDMA Limit Done RXDMA Limit Done Interrupt Enable

**9.2.6.30 0x0124 UART RXDMA Interrupt Status Register (Default Value: 0x0000\_0000)**

Offset: 0x0124			Register Name: UART_RXDMA_IS
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
3	R/W1C	0x0	RXDMA Buffer Overrun Asserted when the RXDMA buffer is overflow.
2	R/W1C	0x0	RXDMA Timeout Done Asserted when a DMA transfer caused by timeout is done.
1	R/W1C	0x0	RXDMA BLK Done Asserted when a DMA block transfer is done.
0	R/W1C	0x0	RXDMA Limit Done Asserted when data transferred reaches limit size in RXDMA Limit Mode.

**9.2.6.31 0x0128 UART RXDMA Write Address Low Register (Default Value: 0x0000\_0000)**

Offset: 0x0128			Register Name: UART_RXDMA_WADDRL
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	RXDMA Current Write Address[31:0] Updated when every DMA transfer is done It is byte address.

9.2.6.32 0x012C UART RXDMA Write Address High Register (Default Value: 0x0000\_0000)

Offset: 0x012C			Register Name: UART_RXDMA_WADDRH
Bit	Read/Write	Default/Hex	Description
31:2	/	/	/
1:0	R	0x0	RXDMA Current Write Address[33:32] Updated when every DMA transfer is done It is byte address.

9.2.6.33 0x0130 UART RXDMA Read Address Low Register (Default Value: 0x0000\_0000)

Offset: 0x0130			Register Name: UART_RXDMA_RADDRL
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	RXDMA Current Read Address[31:0] Software should update this register after reading data in RXDMA Buffer in time It is byte address. The software should not update Buffer Read Address Register until UART_RXDMA_STA[1] is 0. The software should update Read Address High Register first, and then Read Address Low Register, even there is no change on Read Address High Register.

9.2.6.34 0x0134 UART RXDMA Read Address High Register (Default Value: 0x0000\_0000)

Offset: 0x0134			Register Name: UART_RXDMA_RADDRH
Bit	Read/Write	Default/Hex	Description
31:2	/	/	/
1:0	R/W	0x0	RXDMA Current Read Address[33:32] Software should update this register after reading data in RXDMA Buffer in time. It is byte address. The software should not update Buffer Read Address Register until until UART_RXDMA_STA[1] is 0. The software should update Read Address High Register first and then Read Address Low Register , even there is no change on Read Address High Register.

9.2.6.35 0x0138 UART RXDMA Data Count Register (Default Value: 0x0000\_0000)

Offset: 0x0138			Register Name: UART_RXDMA_DCNT
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R	0x0	RXDMA Data Count Only valid while RXDMA Mode is set to 1, it is used for counting the data transferred by RXDMA, and is cleared when reaches RXDMA Limit Size. Its unit is byte.



## 9.3 SPI

### 9.3.1 Overview

The Serial Peripheral Interface (SPI) is a full-duplex, synchronous, four-wire serial communication interface between a CPU and SPI-compliant external devices. The SPI controller contains a 64 x 8 bits receiver buffer (RXFIFO) and a 64 x 8 bits transmit buffer (TXFIFO). It can work in master mode and slave mode.

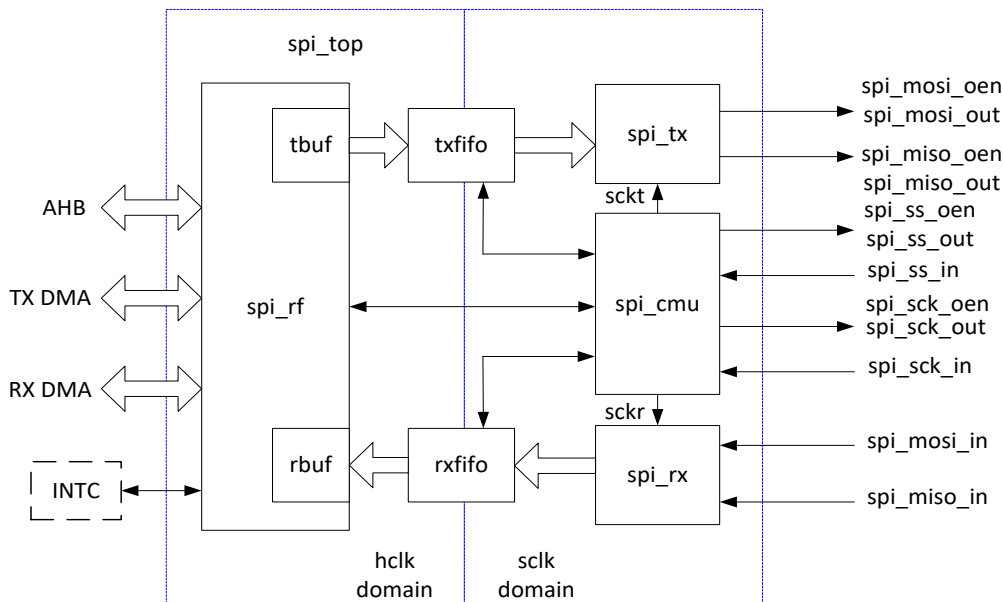
The SPI has the following features:

- Full-duplex synchronous serial interface
- Master/slave configurable
- 8-bit wide by 64-entry FIFO for both transmit and receive data
- Polarity and phase of the Chip Select (SPI\_SS) and SPI Clock (SPI\_SCLK) are configurable
- Supports interrupts and DMA
- Supports mode0, mode1, mode2, and mode3
- Supports 3-wire/4-wire SPI
- Supports programmable serial data frame length: 1 bit to 32 bits
- Supports standard SPI, dual-output/dual-input SPI, dual I/O SPI, quad-output/quad-input SPI
- Supports maximum IO rate of the mass production: 100 MHz

### 9.3.2 Block Diagram

Figure 9-20 shows a block diagram of the SPI.

Figure 9-20 SPI Block Diagram



SPI contains the following sub-blocks:

Table 9-6 SPI Sub-blocks

Sub-block	Description
spi_rf	Responsible for implementing the internal register, interrupt, and DMA Request.
spi_tbuf	The data length transmitted from AHB to TXFIFO is converted into 8 bits, then the data is written into the RXFIFO.
spi_rbuf	The block is used to convert the RXFIFO data into the reading data length of AHB.
txfifo, rxfifo	The data transmitted from the SPI to the external serial device is written into the TXFIFO; the data received from the external serial device into SPI is pushed into the RXFIFO.
spi_cmdu	Responsible for implementing SPI bus clock, chip select, internal sample, and the generation of transfer clock.
spi_tx	Responsible for implementing SPI data transfer, the interface of the internal TXFIFO, and status register.
spi_rx	Responsible for implementing SPI data receive, the interface of the internal RXFIFO, and status register.

### 9.3.3 Functional Description

#### 9.3.3.1 External Signals

The following table describes the external signals of SPI. The MOSI and MISO are bidirectional I/O, when SPI is as a master device, the CLK and CS are the output pin; when SPI is as a slave device, the CLK and CS are the input pin. When using SPI, the corresponding PADS are selected as SPI function via section 9.7 “GPIO”.

**Table 9-7 SPI External Signals**

Signal	Description	Type
SPI0-CS	SPI0 chip select signal, low active When the device is not selected, data will not be accepted via the SI pin, and the SO pin will stop transmission.	I/O
SPI0-CLK	SPI0 clock signal This pin is used to provide a clock to the device and is used to control the flow of data to and from the device.	I/O
SPI0-MOSI	SPI0 master data out, slave data in	I/O
SPI0-MISO	SPI0 master data in, slave data out	I/O
SPI0-WP	Write protection and low active It also can be used for serial data input and output for SPI Quad Input or Quad Output mode.	I/O
SPI0-HOLD	When the device is selected and a serial sequence is underway, the HOLD pin can be used to temporarily pause the serial communication with the master device without deselecting or resetting the serial sequence. While the HOLD pin is asserted, the SO pin is at high impedance, and all transitions on the SCK pin and data on the SI pin are ignored. It also can be used for serial data input and output for SPI Quad Input or Quad Output mode.	I/O

#### 9.3.3.2 Clock Sources

The SPI controller gets 5 different clock sources, users can select one of them to make SPI clock source. The following table describes the clock sources for SPI. For more details on the clock setting, configuration, and gating information, see section 3.3 “CCU”.

**Table 9-8 SPI Clock Sources**

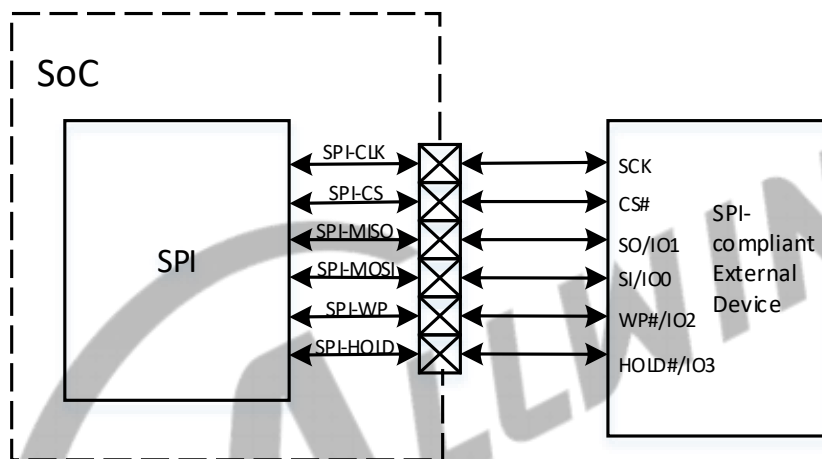
Clock Sources	Description
HOSC	24 MHz Crystal

Clock Sources	Description
PLL_PERI(1X)	Peripheral Clock, default value is 600 MHz
PLL_PERI(2X)	Peripheral Clock, default value is 1200 MHz
PLL_AUDIO0(DIV2)	Audio Clock, the default value is 1536 MHz
PLL_AUDIO0(DIV5)	Audio Clock, the default value is 614.4 MHz

### 9.3.3.3 Typical Application

Figure 9-21 shows the application block diagram when the SPI master device is connected to a slave device.

Figure 9-21 SPI Application Block Diagram



### 9.3.3.4 SPI Transmit Format

The SPI supports 4 different formats for data transfer. The software can select one of the four modes in which the SPI works by setting the bit1 (Polarity) and bit0 (Phase) of [SPI\\_TCR](#). The SPI controller master uses the SPI\_SCLK signal to transfer data in and out of the shift register. Data is clocked using any one of four programmable clock phase and polarity combinations.

The CPOL ([SPI\\_TCR\[1\]](#)) defines the polarity of the clock signal (SPI\_SCLK). The SPI\_SCLK is a high level when CPOL is '1' and it is a low level when CPOL is '0'. The CPHA ([SPI\\_TCR\[0\]](#)) decides whether the leading edge of SPI\_SCLK is used to setup or sample data. The leading edge is used to setup data when CPHA is '1', and sample data when CPHA is '0'. The following table lists the four modes.

Table 9-9 SPI Transmit Format

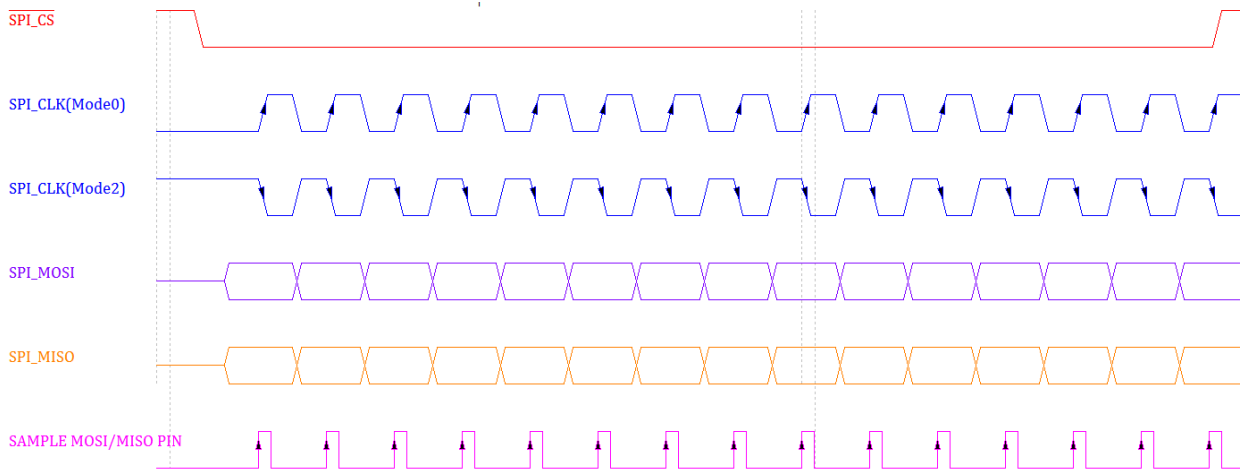
Mode	Polarity (CPOL)	Phase (CPHA)	Leading Edge	Trailing Edge
Mode0	0	0	Sample on the rising edge	Setup on the falling edge



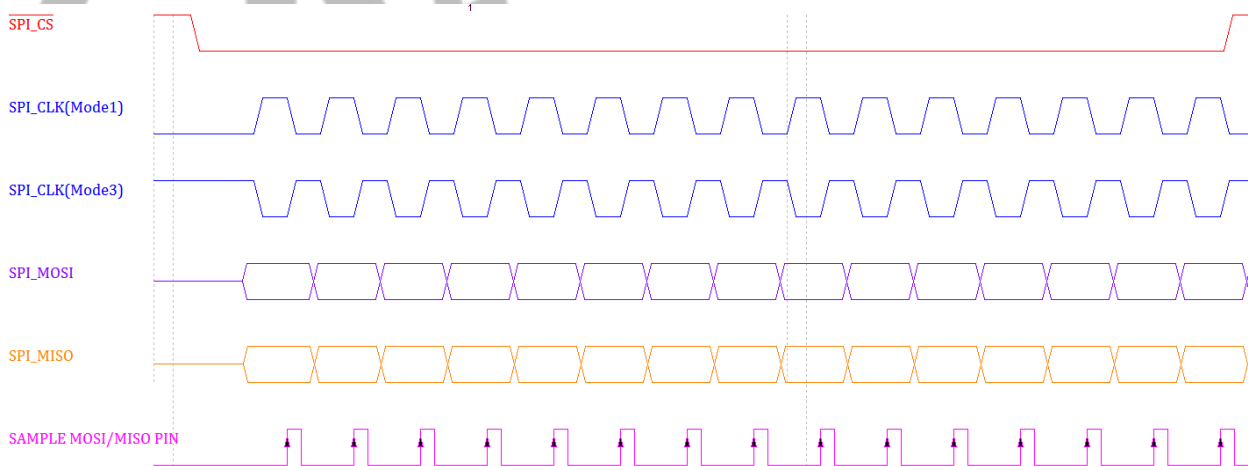
Mode	Polarity (CPOL)	Phase (CPHA)	Leading Edge	Trailing Edge
Mode1	0	1	Setup on the rising edge	Sample on the falling edge
Mode2	1	0	Sample on the falling edge	Setup on the rising edge
Mode3	1	1	Setup on the falling edge	Sample on the rising edge

Figure 9-22 and Figure 9-23 describe four waveforms for SPI\_SCLK.

**Figure 9-22 SPI Phase 0 Timing Diagram**



**Figure 9-23 SPI Phase 1 Timing Diagram**



### 9.3.3.5 SPI Master and Slave Mode

The SPI controller can be configured to a master or slave device. The master mode is selected by setting the MODE bit ([SPI\\_GCR\[1\]](#)); the slave mode is selected by clearing the MODE bit.

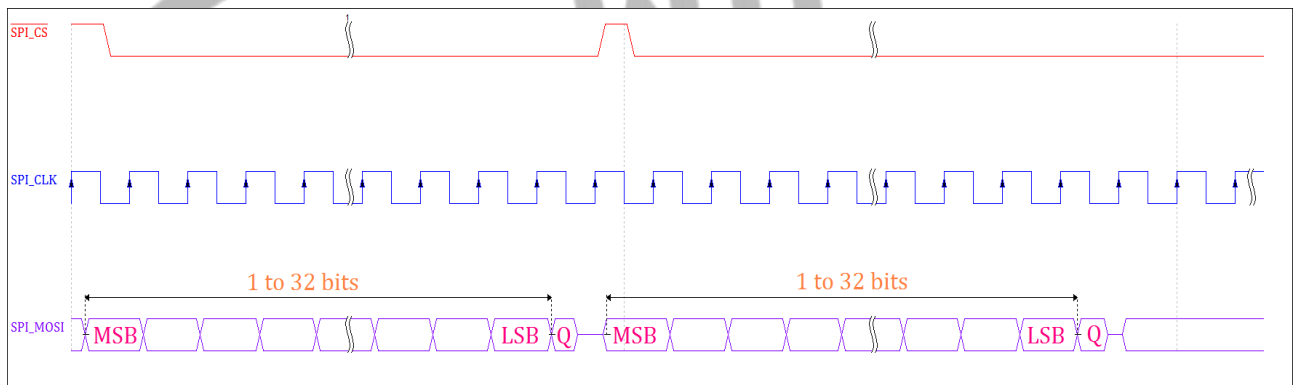
In master mode, the SPI\_CLK is generated and transmitted to the external device, and the data from the TX FIFO is transmitted on the MOSI pin, the data from the slave device is received on the MISO pin and sent to RX FIFO. The Chip Select (SPI\_SS) is an active low signal, and it must be set low before the data are transmitted or received. The SPI\_SS can be selected the auto control mode or software manual control mode. When using the auto control, the SS\_OWNER ([SPI\\_TCR\[6\]](#)) must be cleared (default value is 0); when using the manual control, the SS\_OWNER must be set. And the level of SPI\_SS is controlled by SS\_LEVEL ([SPI\\_TCR\[7\]](#)).

In slave mode, after the software selects the MODE bit ([SPI\\_GCR\[1\]](#)) to '0', it waits for master initiate a transaction. When the master asserts SPI\_SS, then SPI\_CLK is transmitted to the slave device, the slave data is transmitted from TX FIFO on the MISO pin and the data from the MOSI pin is received in RX FIFO.

### 9.3.3.6 SPI 3-Wire Mode

The SPI 3-wire mode is only valid when the SPI controller work in master mode, and is selected when the Work Mode Select bit ([SPI\\_BATC\[1:0\]](#)) is equal to 0x2. And in the 3-wire mode, the input data and the output data use the same single data line. The following figure describes the 3-wire mode.

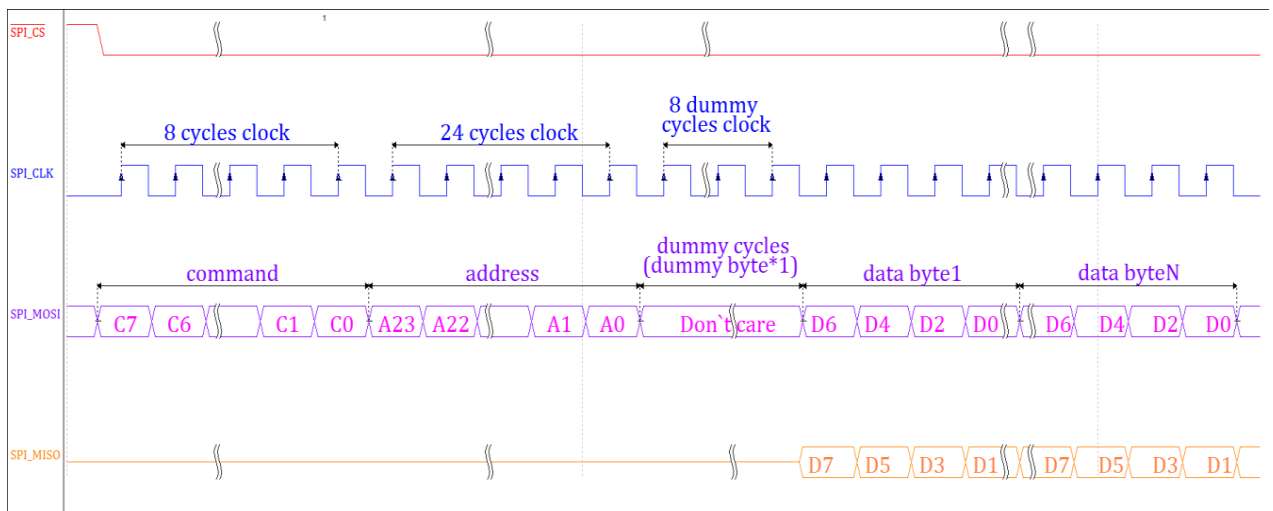
Figure 9-24 SPI 3-Wire Mode



### 9.3.3.7 SPI Dual-Input/Dual-Output and Dual I/O Mode

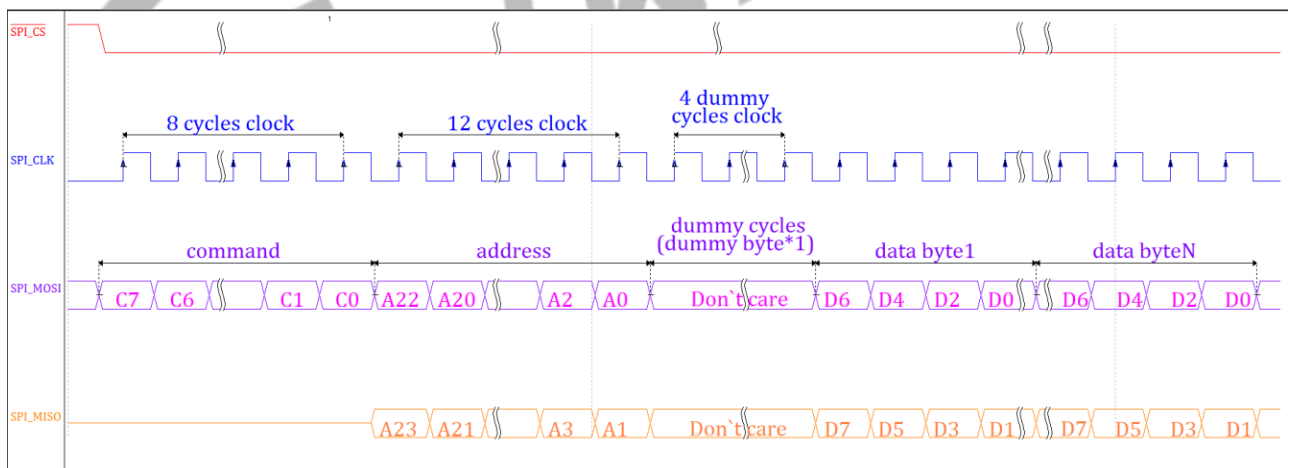
The dual read mode (SPI x2) is selected when the DRM is set in [SPI\\_BCC\[28\]](#). Using the dual mode allows data to be transferred to or from the device at double the rate of standard single mode, the data can be read at fast speed using two data bits (MOSI and MISO) at a time. The following figure describes the dual-input/dual-output SPI (Figure 9-25) and the dual I/O SPI (Figure 9-26).

Figure 9-25 SPI Dual-Input/Dual-Output Mode



In the dual-input/dual-output SPI mode, the command, address, and the dummy bytes output in a unit of a single bit in serial mode through the SPI\_MOSI line, only the data bytes are output (write) and input (read) in a unit of dual bits through the SPI\_MOSI and SPI\_MISO.

Figure 9-26 SPI Dual I/O Mode

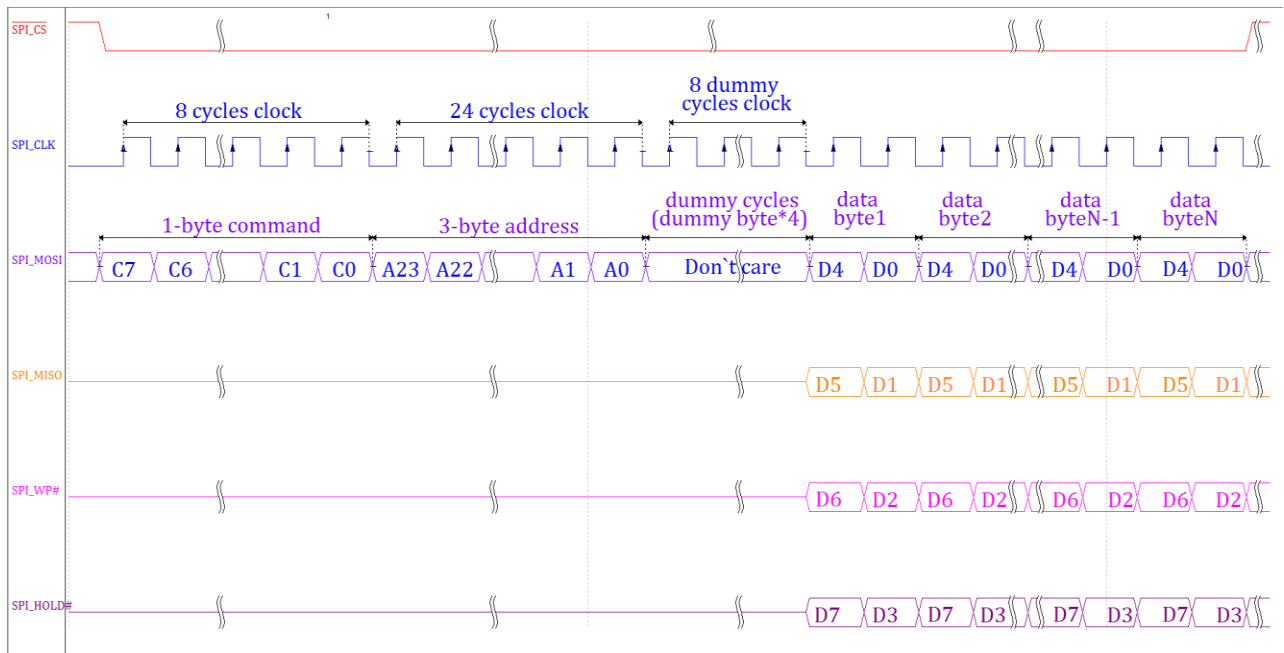


In the dual I/O SPI mode, only the command bytes output in a unit of a single bit in serial mode through the SPI\_MOSI line. The address bytes and the dummy bytes output in a unit of dual bits through the SPI\_MOSI and SPI\_MISO. And the data bytes output (write) and input (read) in a unit of dual bits through the SPI\_MOSI and SPI\_MISO.

### 9.3.3.8 SPI Quad-Input/Quad-Output Mode

The quad read mode (SPI x4) is selected when the Quad\_EN is set in [SPI\\_BCC\[29\]](#). Using the quad mode allows data to be transferred to or from the device at 4 times the rate of standard single mode, the data can be read at fast speed using four data bits (MOSI, MISO, IO2 (WP#) and IO3 (HOLD#)) at the same time. The following figure describes the quad-input/quad-output SPI.

Figure 9-27 SPI Quad-Input/Quad-Output Mode



In the quad-input/quad-output SPI mode, the command, address, and the dummy bytes output in a unit of a single bit in serial mode through SPI\_MOSI line. Only the data bytes output (write) and input (read) in a unit of quad bits through the SPI\_MOSI, SPI\_MISO, SPI\_WP#, and SPI\_HOLD#.

### 9.3.3.9 Transmission/Reception Bursts in Master Mode

In SPI master mode, the transmission and reception bursts (byte in unit) are configured before the SPI transfers the serial data between the processor and external device. The transmission bursts are written in MWTC (bit[23:0]) of the [SPI Master Transmit Counter Register](#). The transmission bursts in single mode before automatically sending dummy bursts are written in STC (bit[23:0]) of the [SPI Master Transmit Counter Register](#). For dummy data, the SPI controller can automatically send before receiving by writing DBC (bit[27:24]) in the [SPI Master Transmit Counter Register](#). If users do not use the SPI controller to send dummy data automatically, then the dummy bursts are used as the transmission counters to write together in MWTC (bit[23:0]) of the [SPI Master Transmit Counter Register](#). In master mode, the total burst numbers are written in MBC (bit[23:0]) of the [SPI Master Transmit Counter Register](#). When all transmission and reception bursts are transferred, the SPI controller will send a completed interrupt, at the same time, the SPI controller will clear [DBC](#), [MWTC](#), and [MBC](#).

### 9.3.3.10 SPI Sample Mode and Run Clock Configuration

The SPI controller runs at 3 kHz–100 MHz at its interface to external SPI devices. The internal SPI clock should run at the same frequency as the outgoing clock in the master mode. The SPI clock is selected from different clock sources, the SPI must configure different work mode. There are three work modes: normal sample mode, delay half-cycle sample mode, delay one-cycle sample mode. The Delay half-cycle sample mode is the default mode of the SPI controller. When the SPI runs at 40 MHz or below 40 MHz, the SPI can work at normal sample mode or delay half-cycle sample mode. When the SPI runs over 80 MHz, setting the SDC bit in the [SPI Transfer Control Register](#) to ‘1’ makes the internal read sample point with a half-cycle delay of SPI\_CLK, which is used in high speed read operation to reduce the error caused by the time delay of SPI\_CLK between master and slave. Table 9-10 and Table 9-11 show the different configurations of the SPI sample mode.

**Table 9-10 SPI Old Sample Mode and Run Clock**

SPI Sample Mode	SDM(bit13)	SDC(bit11)	Run Clock
normal sample	1	0	<=24 MHz
delay half cycle sample	0	0	<=40 MHz
delay one cycle sample	0	1	>=80 MHz



The remaining spectrum is not recommended. Because when the output delay of SPI flash (refer to the datasheet of the manufactures for the specific delay time) is the same with the half-cycle time of SPI working clock, the variable edge of the output data for the device bumps into the clock sampling edge of the controller, so setting 1 cycle of sampling delay would cause stability problem.

**Table 9-11 SPI New Sample Mode**

SPI Sample Mode	SDM (bit13)	SDC (bit11)	SDC1 (bit15)
normal sample	1	0	0
delay half cycle sample	0	0	0
delay one cycle sample	0	1	0
delay 1.5 cycle sample	1	1	0
delay 2 cycle sample	1	0	1
delay 2.5 cycle sample	0	0	1
delay 3 cycle sample	0	1	1

### 9.3.3.11 SPI Error Conditions

If any error conditions occur, the hardware will set the corresponding status bits in the [SPI Interrupt Status Register](#) and stop the transfer. For the SPI controller, the following error scenarios can happen.

#### 1. TX\_FIFO Underrun

The TX\_FIFO underrun happens when the CPU/DMA reads data from TX FIFO when it is empty. In the case, the SPI controller will end the transaction and flag the error bit along with the TF\_UDF bit in the [SPI Interrupt Status Register](#). The SPI controller will generate an interrupt if interrupts are enabled. The software has to clear the error bit and the TF\_UDF bit. To start a new transaction, the software has to reset the FIFO by writing to the SRST (soft reset) bit in the [SPI Global Control Register](#).

#### 2. TX\_FIFO Overflow

The TX\_FIFO overflow happens when the CPU/DMA writes data into the TX FIFO when it is full. In the case, the SPI controller will end the transaction and flag the error bit along with the TF\_OVF bit in the [SPI Interrupt Status Register](#). The SPI controller will generate an interrupt if interrupts are enabled. The software has to clear the error bit and the TF\_OVF bit. To start a new transaction, the software has to reset the FIFO by writing to the SRST (soft reset) bit in the [SPI Global Control Register](#).

#### 3. RX\_FIFO Underrun

The RX\_FIFO underrun happens when the CPU/DMA reads data from RX FIFO when it is empty. In the case, the SPI controller will end the transaction and flag the error bit along with the RF\_UDF bit in the [SPI Interrupt Status Register](#). The SPI controller will generate an interrupt if interrupts are enabled. The software has to clear the error bit and the RF\_UDF bit. To start a new transaction, the software has to reset the fifo by writing to the SRST (soft reset) bit in the [SPI Global Control Register](#).

#### 4. RX\_FIFO Overflow

The RX\_FIFO overflow happens when the CPU/DMA writes data into the RX FIFO when it is full. In the case, the SPI controller will end the transaction and flag the error bit along with the RF\_OVF bit in the [SPI Interrupt Status Register](#). The SPI controller will generate an interrupt if interrupts are enabled. The software has to clear the error bit and the RF\_OVF bit. To start a new transaction, the software has to reset the FIFO by writing to the SRST (soft reset) bit in the [SPI Global Control Register](#).

### 9.3.4 Programming Guidelines

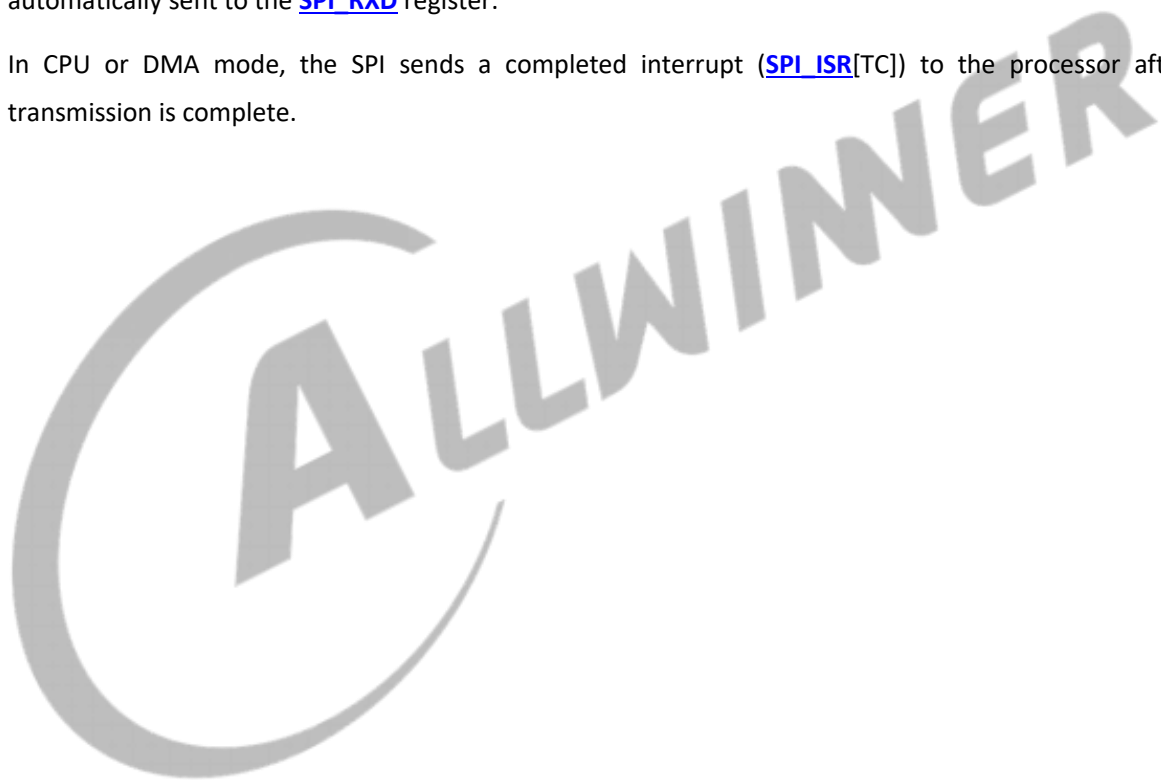
#### 9.3.4.1 Writing/Reading Data Process

The SPI transfers serial data between the processor and the external device. The CPU mode and DMA mode are the two main operational modes for SPI. For each SPI, the data is simultaneously transmitted (shifted out serially) and received (shifted in serially). The SPI has 2 channels, including the TX channel and RX channel. The TX channel has the path from TX FIFO to the external device. The RX channel has the path from the external device to RX FIFO.

**Write Data:** The CPU or DMA must write data on the [SPI\\_TXD](#) register, the data on the register are automatically moved to TX FIFO.

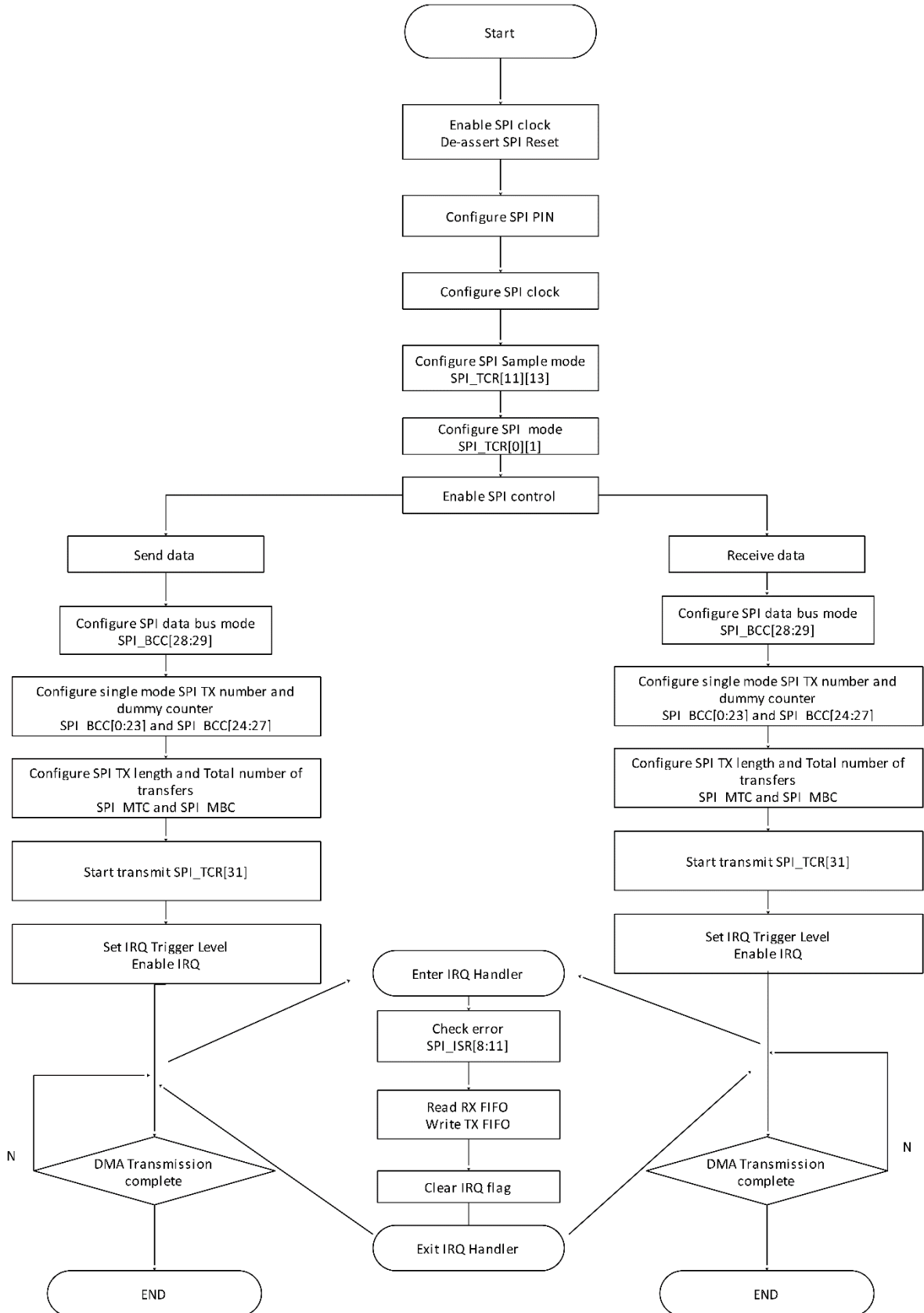
**Read Data:** To read data from RX FIFO, the CPU or DMA must access the [SPI\\_RXD](#) register and the data are automatically sent to the [SPI\\_RXD](#) register.

In CPU or DMA mode, the SPI sends a completed interrupt ([SPI\\_ISR\[TC\]](#)) to the processor after each transmission is complete.



CPU Mode

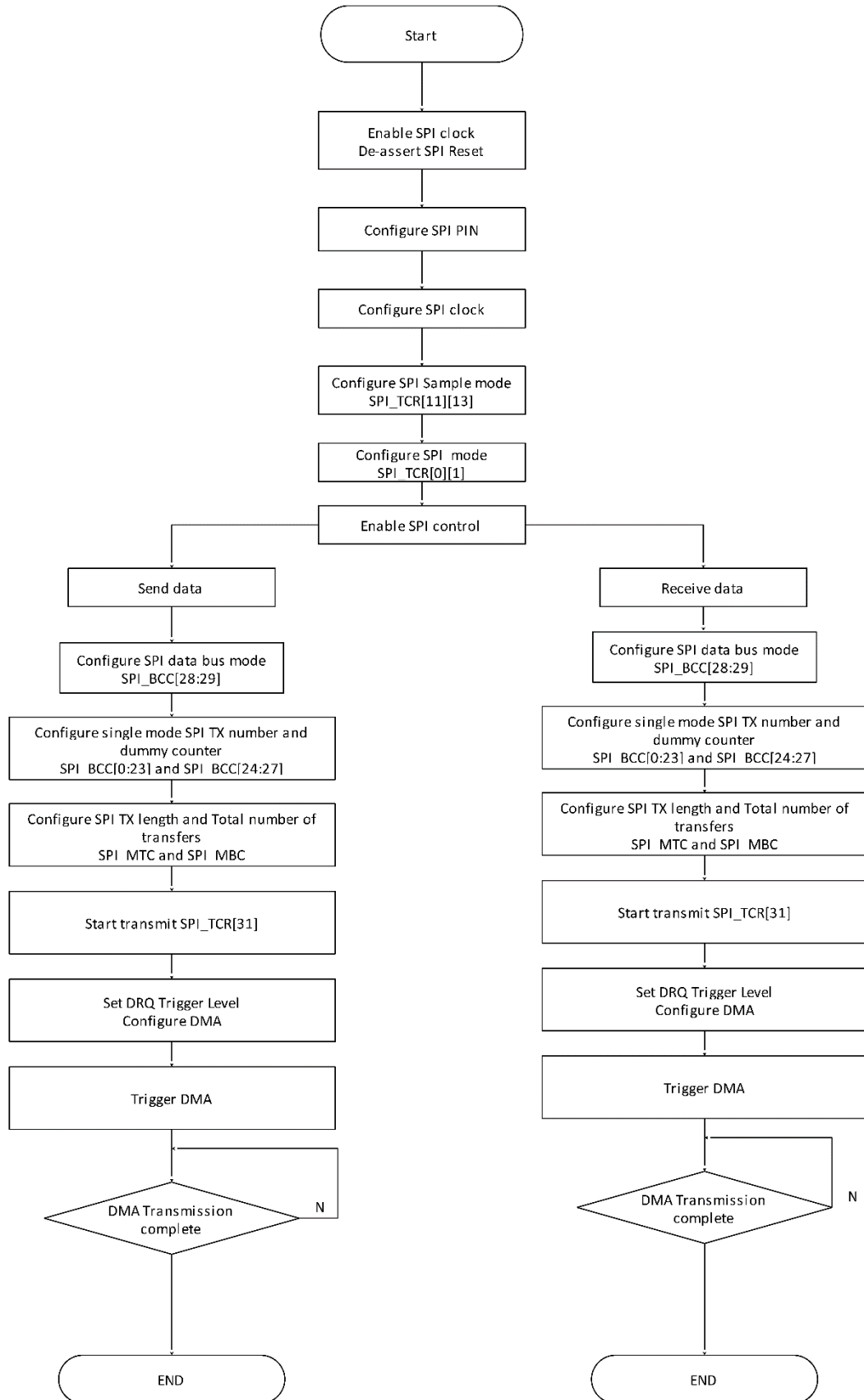
Figure 9-28 SPI Write/Read Data in CPU Mode





DMA Mode

Figure 9-29 SPI Write/Read Data in DMA Mode



### 9.3.4.2 Calibrate Delay Chain

The SPI has one delay chain which is used to generate delay to make proper timing between the internal SPI clock signal and data signals. Delay chain is made up of 64 delay cells. The delay time of one delay cell can be estimated through delay chain calibration.

The steps to calibrate delay chain are as follows:

- Step 1** Enable SPI. To calibrate the delay chain by the operation registers in SPI, the SPI must be enabled through AHB reset and AHB clock gating control registers.
- Step 2** Configure a proper clock for SPI. The calibration delay chain is based on the clock for SPI from CCU.
- Step 3** Set proper initial delay value. Write 0xA0 to the [SPI Sample Delay Control Register](#) to set initial delay value 0x20 to delay chain. Then write 0x0 to the [SPI Sample Delay Control Register](#) to clear this value.
- Step 4** Write 0x8000 to the [SPI Sample Delay Control Register](#) to start to calibrate the delay chain.
- Step 5** Wait until the flag (bit14 in the [SPI Sample Delay Control Register](#)) of calibration done is set. The number of delay cells is shown at the bit[13:8] of the [SPI Sample Delay Control Register](#). The delay time generated by these delay cells is equal to the cycle of the SPI clock nearly. This value is the result of calibration.
- Step 6** Calculate the delay time of one delay cell according to the cycle of the SPI clock and the result of calibration.

### 9.3.5 Register List

Module Name	Base Address
SPIO	0x04025000

Register Name	Offset	Description
SPI_GCR	0x0004	SPI Global Control Register
SPI_TCR	0x0008	SPI Transfer Control Register
SPI_IER	0x0010	SPI Interrupt Control Register
SPI_ISR	0x0014	SPI Interrupt Status Register
SPI_FCR	0x0018	SPI FIFO Control Register
SPI_FSR	0x001C	SPI FIFO Status Register
SPI_WCR	0x0020	SPI Wait Clock Register
SPI_SAMP_DL	0x0028	SPI Sample Delay Control Register
SPI_MBC	0x0030	SPI Master Burst Counter Register

Register Name	Offset	Description
SPI_MTC	0x0034	SPI Master Transmit Counter Register
SPI_BCC	0x0038	SPI Master Burst Control Register
SPI_BATCR	0x0040	SPI Bit-Aligned Transfer Configure Register
SPI_BA_CCR	0x0044	SPI Bit-Aligned Clock Configuration Register
SPI_TBR	0x0048	SPI TX Bit Register
SPI_RBR	0x004C	SPI RX Bit Register
SPI_NDMA_MODE_CTL	0x0088	SPI Normal DMA Mode Control Register
SPI_TXD	0x0200	SPI TX Data Register
SPI_RXD	0x0300	SPI RX Data Register

### 9.3.6 Register Description

#### 9.3.6.1 0x0004 SPI Global Control Register (Default Value: 0x0000\_0080)

Offset:0x0004			Register Name: SPI_GCR
Bit	Read/Write	Default/Hex	Description
31	R/WAC	0x0	SRST Soft reset Writing '1' to this bit will clear the SPI controller, and auto clear to '0' when reset operation completes. Writing '0' to this bit has no effect.
30:8	/	/	/
7	R/W	0x1	TP_EN Transmit Pause Enable In master mode, it is used to control the transmit state machine to stop smart burst sending when RX FIFO is full. 0: Normal operation, ignore RXFIFO status 1: Stop transmit data when RXFIFO full Cannot be written when XCH=1.
6:3	/	/	/
2	R/W	0x0	MODE_SELEC Sample Timing Mode Select 0: Old mode of Sample Timing 1: New mode of Sample Timing Cannot be written when XCH=1.

Offset:0x0004			Register Name: SPI_GCR
Bit	Read/Write	Default/Hex	Description
1	R/W	0x0	<p>MODE</p> <p>SPI Function Mode Select</p> <p>0: Slave mode</p> <p>1: Master mode</p> <p>Cannot be written when XCH=1.</p>
0	R/W	0x0	<p>EN</p> <p>SPI Module Enable Control</p> <p>0: Disable</p> <p>1: Enable</p> <p>After transforming from bit_mode to byte_mode, it must enable the SPI module again.</p>

9.3.6.2 0x0008 SPI Transfer Control Register (Default Value: 0x0000\_0087)

Offset: 0x0008			Register Name: SPI_TCR
Bit	Read/Write	Default/Hex	Description
31	R/WAC	0x0	<p>XCH</p> <p>Exchange Burst</p> <p>In master mode, it is used to start SPI burst.</p> <p>0: Idle</p> <p>1: Initiates exchange.</p> <p>Writing "1" to this bit will start the SPI burst, and will auto-clear after finishing the bursts transfer specified by SPI_MBC. Writing "1" to SRST (SPI_GCR[31]) will also clear this bit. Writing '0' to this bit has no effect.</p> <p>Cannot be written when XCH=1.</p>
30:16	/	/	/
15	R/W	0x0	<p>SDC1</p> <p>Master Sample Data Control register1</p> <p>Set this bit to '1' to make the internal read sample point with a delay of half-cycle of SPI_CLK. It is used in high speed read operation to reduce the error caused by the time delay of SPI_CLK between master and slave.</p> <p>0: normal operation, do not delay the internal read sample point</p> <p>1: delay the internal read sample point</p> <p>Cannot be written when XCH=1.</p>

Offset: 0x0008			Register Name: SPI_TCR
Bit	Read/Write	Default/Hex	Description
14	R/W	0x0	<p>SDDM Sending Data Delay Mode 0: Normal sending 1: Delay sending</p> <p>Set the bit to "1" to make the data that should be sent with a delay of half-cycle for SPI_CLK in dual IO mode of SPI mode0. Cannot be written when XCH=1.</p>
13	R/W	0x0	<p>SDM Master Sample Data Mode 0: Delay sample mode 1: Normal sample mode</p> <p>In normal sample mode, the SPI master samples the data at the correct edge for each SPI mode; In delay sample mode, the SPI master samples data at the edge that is half cycle delayed by the correct edge defined in respective SPI mode. Cannot be written when XCH=1.</p>
12	R/W	0x0	<p>FBS First Transmit Bit Select 0: MSB first 1: LSB first</p> <p>Cannot be written when XCH=1.</p>
11	R/W	0x0	<p>SDC Master Sample Data Control</p> <p>Set this bit to '1' to make the internal read sample point with a delay of half-cycle for SPI_CLK. It is used in high speed read operation to reduce the error caused by the time delay of SPI_CLK between master and slave. 0: Normal operation, do not delay the internal read sample point 1: Delay the internal read sample point Cannot be written when XCH=1.</p>
10	R/W	0x0	<p>RPSM Rapids Mode Select</p> <p>Select rapid mode for high speed write. 0: Normal write mode 1: Rapid write mode Cannot be written when XCH=1.</p>

Offset: 0x0008			Register Name: SPI_TCR
Bit	Read/Write	Default/Hex	Description
9	R/W	0x0	<p>DDB Dummy Burst Type</p> <p>0: The bit value of dummy SPI burst is zero 1: The bit value of dummy SPI burst is one</p> <p>Cannot be written when XCH=1.</p>
8	R/W	0x0	<p>DHB Discard Hash Burst</p> <p>In master mode, it controls whether discarding unused SPI bursts</p> <p>0: Receiving all SPI bursts in the BC period 1: Discard unused SPI bursts, only fetching the SPI bursts during the dummy burst period. The burst number is specified by TC.</p> <p>Cannot be written when XCH=1.</p>
7	R/W	0x1	<p>SS_LEVEL</p> <p>When control SS signal manually (SS_OWNER (SPI_TCR[6])=1), set this bit to '1' or '0' to control the level of SS signal.</p> <p>0: Set SS to low 1: Set SS to high</p> <p>Cannot be written when XCH=1.</p>
6	R/W	0x0	<p>SS_OWNER</p> <p>SS Output Owner Select</p> <p>Usually, the controller sends the SS signal automatically with data together. When this bit is set to 1, the software must manually write SS_LEVEL (SPI_TCR[7]) to 1 or 0 to control the level of the SS signal.</p> <p>0: SPI controller 1: Software</p> <p>Cannot be written when XCH=1.</p>
5:4	R/W	0x0	<p>SS_SEL</p> <p>SPI Chip Select</p> <p>Select one of four external SPI Master/Slave Devices</p> <p>00: SPI_SS0 will be asserted 01: SPI_SS1 will be asserted 10: SPI_SS2 will be asserted 11: SPI_SS3 will be asserted</p> <p>Cannot be written when XCH=1.</p>

Offset: 0x0008			Register Name: SPI_TCR
Bit	Read/Write	Default/Hex	Description
3	R/W	0x0	<p>SSCTL</p> <p>In master mode, this bit selects the output waveform for the SPI_SSx signal. Only valid when SS_OWNER (SPI_TCR[6])= 0.</p> <p>0: SPI_SSx remains asserted between SPI bursts</p> <p>1: Negate SPI_SSx between SPI bursts</p> <p>Cannot be written when XCH=1.</p>
2	R/W	0x1	<p>SPOL</p> <p>SPI Chip Select Signal Polarity Control</p> <p>0: Active high polarity (0 = Idle)</p> <p>1: Active low polarity (1 = Idle)</p> <p>Cannot be written when XCH=1.</p>
1	R/W	0x1	<p>CPOL</p> <p>SPI Clock Polarity Control</p> <p>0: Active high polarity (0 = Idle)</p> <p>1: Active low polarity (1 = Idle)</p> <p>Cannot be written when XCH=1.</p>
0	R/W	0x1	<p>CPHA</p> <p>SPI Clock/Data Phase Control</p> <p>0: Phase 0 (Leading edge for sample data)</p> <p>1: Phase 1 (Leading edge for setup data)</p> <p>Cannot be written when XCH=1.</p>

**9.3.6.3 0x0010 SPI Interrupt Control Register (Default Value: 0x0000\_0000)**

Offset: 0x0010			Register Name: SPI_IER
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13	R/W	0x0	<p>SS_INT_EN</p> <p>SSI Interrupt Enable</p> <p>Chip select signal (SSx) from the valid state to the invalid state</p> <p>0: Disable</p> <p>1: Enable</p>

Offset: 0x0010			Register Name: SPI_IER
Bit	Read/Write	Default/Hex	Description
12	R/W	0x0	TC_INT_EN Transfer Completed Interrupt Enable 0: Disable 1: Enable
11	R/W	0x0	TF_UDR_INT_EN TXFIFO Underrun Interrupt Enable 0: Disable 1: Enable
10	R/W	0x0	TF_OVF_INT_EN TX FIFO Overflow Interrupt Enable 0: Disable 1: Enable
9	R/W	0x0	RF_UDR_INT_EN RXFIFO Underrun Interrupt Enable 0: Disable 1: Enable
8	R/W	0x0	RF_OVF_INT_EN RX FIFO Overflow Interrupt Enable 0: Disable 1: Enable
7	/	/	/
6	R/W	0x0	TF_FUL_INT_EN TX FIFO Full Interrupt Enable 0: Disable 1: Enable
5	R/W	0x0	TX_EMP_INT_EN TX FIFO Empty Interrupt Enable 0: Disable 1: Enable
4	R/W	0x0	TX_ERQ_INT_EN TX FIFO Empty Request Interrupt Enable 0: Disable 1: Enable
3	/	/	/



Offset: 0x0010			Register Name: SPI_IER
Bit	Read/Write	Default/Hex	Description
2	R/W	0x0	RF_FUL_INT_EN RX FIFO Full Interrupt Enable 0: Disable 1: Enable
1	R/W	0x0	RX_EMP_INT_EN RX FIFO Empty Interrupt Enable 0: Disable 1: Enable
0	R/W	0x0	RF_RDY_INT_EN RX FIFO Ready Request Interrupt Enable 0: Disable 1: Enable

9.3.6.4 0x0014 SPI Interrupt Status Register (Default Value: 0x0000\_0032)

Offset: 0x0014			Register Name: SPI_ISR
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13	R/W1C	0x0	SSI SS Invalid Interrupt When SSI is 1, it indicates that SPI_SS has changed from valid state to invalid state. Writing 1 to this bit clears it.
12	R/W1C	0x0	TC Transfer Completed In master mode, it indicates that all bursts specified by SPI_MBC have been exchanged. In other conditions, when setting, this bit indicates that all the data in TXFIFO has been loaded in the Shift register, and the Shift register has shifted out all the bits. Writing 1 to this bit clears it. 0: Busy 1: Transfer completed

Offset: 0x0014			Register Name: SPI_ISR
Bit	Read/Write	Default/Hex	Description
11	R/W1C	0x0	<p>TF_UDF TXFIFO Underrun</p> <p>This bit is set when the TXFIFO is underrun. Writing 1 to this bit clears it.</p> <p>0: TXFIFO is not underrun 1: TXFIFO is underrun</p>
10	R/W1C	0x0	<p>TF_OVF TXFIFO Overflow</p> <p>This bit is set when the TXFIFO is overflowed. Writing 1 to this bit clears it.</p> <p>0: TXFIFO is not overflowed 1: TXFIFO is overflowed</p>
9	R/W1C	0x0	<p>RX_UDF RXFIFO Underrun</p> <p>When set, this bit indicates that RXFIFO is underrun. Writing 1 to this bit clears it.</p> <p>0: RXFIFO is not underrun 1: RXFIFO is underrun</p>
8	R/W1C	0x0	<p>RX_OVF RXFIFO Overflow</p> <p>When set, this bit indicates that RXFIFO has overflowed. Writing 1 to this bit clears it.</p> <p>0: RXFIFO is not overflowed 1: RXFIFO is overflowed</p>
7	/	/	/
6	R/W1C	0x0	<p>TX_FULL TXFIFO Full</p> <p>This bit is set when the TXFIFO is full. Writing 1 to this bit clears it.</p> <p>0: TXFIFO is not Full 1: TXFIFO is Full</p>
5	R/W1C	0x1	<p>TX_EMP TXFIFO Empty</p> <p>This bit is set when the TXFIFO is empty. Writing 1 to this bit clears it.</p> <p>0: TXFIFO contains one or more words 1: TXFIFO is empty</p>

Offset: 0x0014			Register Name: SPI_ISR
Bit	Read/Write	Default/Hex	Description
4	R/W1C	0x1	<p>TX_READY TXFIFO Ready</p> <p>0: TX_WL &gt; TX_TRIG_LEVEL 1: TX_WL &lt;= TX_TRIG_LEVEL</p> <p>This bit will be immediately set to 1 if TX_WL &lt;= TX_TRIG_LEVEL. Writing "1" to this bit clears it. The TX_WL is the water level of TXFIFO.</p>
3	/	/	/
2	R/W1C	0x0	<p>RX_FULL RXFIFO Full</p> <p>This bit is set when the RXFIFO is full. Writing 1 to this bit clears it.</p> <p>0: Not Full 1: Full</p>
1	R/W1C	0x1	<p>RX_EMP RXFIFO Empty</p> <p>This bit is set when the RXFIFO is empty. Writing 1 to this bit clears it.</p> <p>0: Not empty 1: empty</p>
0	R/W1C	0x0	<p>RX_RDY RXFIFO Ready</p> <p>0: RX_WL &lt; RX_TRIG_LEVEL 1: RX_WL &gt;= RX_TRIG_LEVEL</p> <p>This bit is will be immediately set to 1 if RX_WL &gt;= RX_TRIG_LEVEL. Writing "1" to this bit clears it. The RX_WL is the water level of RXFIFO.</p>

**9.3.6.5 0x0018 SPI FIFO Control Register (Default Value: 0x0040\_0001)**

Offset: 0x0018			Register Name: SPI_FCR
Bit	Read/Write	Default/Hex	Description
31	R/WAC	0x0	<p>TX_FIFO_RST TX FIFO Reset</p> <p>Writing '1' to this bit will reset the control portion of the TX FIFO and auto clear to '0' when completing reset operation, writing to '0' has no effect.</p>

Offset: 0x0018			Register Name: SPI_FCR
Bit	Read/Write	Default/Hex	Description
30	R/W	0x0	<p>TF_TEST_ENB TX Test Mode Enable</p> <p>0: Disable 1: Enable</p> <p>In normal mode, the TXFIFO can only be read by the SPI controller, writing '1' to this bit will switch the read and write function of the TXFIFO to AHB bus. This bit is used to test the TXFIFO, do not set in normal operation, and do not set RF_TEST and TF_TEST at the same time.</p>
29:25	/	/	/
24	R/W	0x0	<p>TF_DRQ_EN TXFIFO DMA Request Enable</p> <p>0: Disable 1: Enable</p>
23:16	R/W	0x40	<p>TX_TRIG_LEVEL TXFIFO Empty Request Trigger Level</p>
15	R/WAC	0x0	<p>RF_RST RXFIFO Reset</p> <p>Writing '1' to this bit will reset the control portion of the receiver FIFO, and auto clear to '0' when completing reset operation, writing '0' to this bit has no effect.</p>
14	R/W	0x0	<p>RF_TEST RX Test Mode Enable</p> <p>0: Disable 1: Enable</p> <p>In normal mode, the RXFIFO can only be written by the SPI controller, writing '1' to this bit will switch the read and write function of RXFIFO to AHB bus. This bit is used to test the RXFIFO, do not set in normal operation, and do not set RF_TEST and TF_TEST at the same time.</p>
13:9	/	/	/
8	R/W	0x0	<p>RF_DRQ_EN RX FIFO DMA Request Enable</p> <p>0: Disable 1: Enable</p>
7:0	R/W	0x1	<p>RX_TRIG_LEVEL RX FIFO Ready Request Trigger Level</p>

9.3.6.6 0x001C SPI FIFO Status Register (Default Value: 0x0000\_0000)

Offset: 0x001C			Register Name: SPI_FSR
Bit	Read/Write	Default/Hex	Description
31	R	0x0	TB_WR TX FIFO Write Buffer Write Enable
30:28	R	0x0	TB_CNT TX FIFO Write Buffer Counter These bits indicate the number of words in TX FIFO Write Buffer
27:24	/	/	/
23:16	R	0x0	TF_CNT TX FIFO Counter These bits indicate the number of words in TX FIFO 0: 0 byte in TX FIFO 1: 1 byte in TX FIFO ... 64: 64 bytes in TX FIFO other: Reserved
15	R	0x0	RB_WR RX FIFO Read Buffer Write Enable
14:12	R	0x0	RB_CNT RX FIFO Read Buffer Counter These bits indicate the number of words in RX FIFO Read Buffer
11:8	/	/	/
7:0	R	0x0	RF_CNT RX FIFO Counter These bits indicate the number of words in RX FIFO 0: 0 byte in RX FIFO 1: 1 byte in RX FIFO ... 64: 64 bytes in RX FIFO other: Reserved

9.3.6.7 0x0020 SPI Wait Clock Register (Default Value: 0x0000\_0000)

Offset: 0x0020			Register Name: SPI_WCR
Bit	Read/Write	Default/Hex	Description
31:20	/	/	/
19:16	R/W	0x0	<p>SWC</p> <p>Dual mode direction switch wait clock counter (for master mode only).</p> <p>These bits control the number of wait states to be inserted before starting dual data transfer in dual SPI mode. The SPI module counts SPI_SCLK by SWC for delaying the next word data transfer.</p> <p>0: No wait states inserted</p> <p>n: n SPI_SCLK wait states inserted</p> <p>Cannot be written when XCH=1.</p>
15:0	R/W	0x0	<p>WCC</p> <p>Wait Clock Counter (In master mode)</p> <p>These bits control the number of wait states to be inserted in data transfers. The SPI module counts SPI_SCLK by WCC for delaying the next word data transfer.</p> <p>0: No wait states inserted</p> <p>n: n SPI_SCLK wait states inserted</p> <p>Cannot be written when XCH=1.</p>

9.3.6.8 0x0028 SPI Sample Delay Control Register (Default Value: 0x0000\_2000)

Offset: 0x0028			Register Name: SPI_SAMP_DL
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15	R/W	0x0	<p>SAMP_DL_CAL_START</p> <p>Sample Delay Calibration Start</p> <p>When set, the sample delay chain calibration is started.</p> <p>Cannot be written when XCH=1.</p>
14	R	0x0	<p>SAMP_DL_CAL_DONE</p> <p>Sample Delay Calibration Done</p> <p>When set, it means that sample delay chain calibration is done and the result of calibration is shown in SAMP_DL.</p> <p>Cannot be written when XCH=1.</p>

Offset: 0x0028			Register Name: SPI_SAMP_DL
Bit	Read/Write	Default/Hex	Description
13:8	R	0x20	<p>SAMP_DL</p> <p>Sample Delay</p> <p>It indicates the number of delay cells corresponding to the current card clock. The delay time generated by these delay cells is equal to the cycle of the card clock nearly.</p> <p>Generally, it is necessary to do drive delay calibration when the card clock is changed.</p> <p>This bit is valid only when SAMP_DL_CAL_DONE is set.</p> <p>Cannot be written when XCH=1.</p>
7	R/W	0x0	<p>SAMP_DL_SW_EN</p> <p>Sample Delay Software Enable</p> <p>When set, it indicates that enable sample delay specified at SAMP_DL_SW.</p> <p>Cannot be written when XCH=1.</p>
6	/	/	/
5:0	R/W	0x0	<p>SAMP_DL_SW</p> <p>Sample Delay Software</p> <p>The relative delay between the clock line and command line, data lines.</p> <p>It can be determined according to the value of SAMP_DL, the cycle of the card clock, and the input timing requirement of the device.</p> <p>Cannot be written when XCH=1.</p>

**9.3.6.9 0x0030 SPI Master Burst Counter Register (Default Value: 0x0000\_0000)**

Offset: 0x0030			Register Name: SPI_MBC
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/

Offset: 0x0030			Register Name: SPI_MBC
Bit	Read/Write	Default/Hex	Description
23:0	R/W	0x0	<p>MBC Master Burst Counter</p> <p>In master mode, this field specifies the total burst number. The total transfer data include the TXD, RXD, and dummy burst.</p> <p>0: 0 burst 1: 1 burst ... N: N bursts</p> <p>Cannot be written when XCH=1.</p>

9.3.6.10 0x0034 SPI Master Transmit Counter Register (Default Value: 0x0000\_0000)

Offset: 0x0034			Register Name: SPI_MTC
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:0	R/W	0x0	<p>MWTC Master Write Transmit Counter</p> <p>In master mode, this field specifies the burst number that should be sent to TXFIFO before automatically sending dummy bursts. For saving bus bandwidth, the dummy bursts (all zero bits or all one bits) are sent by SPI Controller automatically.</p> <p>0: 0 burst 1: 1 burst ... N: N bursts</p> <p>Cannot be written when XCH=1.</p>

9.3.6.11 0x0038 SPI Master Burst Control Counter Register (Default Value: 0x0000\_0000)

Offset: 0x0038			Register Name: SPI_BCC
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/



Offset: 0x0038			Register Name: SPI_BCC
Bit	Read/Write	Default/Hex	Description
29	R/W	0x0	<p>Quad_EN</p> <p>Quad Mode Enable</p> <p>The quad mode includes Quad-Input and Quad-Output.</p> <p>0: Quad mode disable</p> <p>1: Quad mode enable</p> <p>Cannot be written when XCH=1.</p>
28	R/W	0x0	<p>DRM</p> <p>Master Dual Mode RX Enable</p> <p>It is only valid when Quad_Mode_EN=0.</p> <p>0: RX uses the single-bit mode</p> <p>1: RX uses the dual-bit mode</p> <p>Cannot be written when XCH=1.</p>
27:24	R/W	0x0	<p>DBC</p> <p>Master Dummy Burst Counter</p> <p>In master mode, this field specifies the burst number that should be sent before receiving in dual SPI mode. The data does not care by the device.</p> <p>0: 0 burst</p> <p>1: 1 burst</p> <p>...</p> <p>N: N bursts</p> <p>Cannot be written when XCH=1.</p>
23:0	R/W	0x0	<p>STC</p> <p>Master Single Mode Transmit Counter</p> <p>In master mode, this field specifies the burst number that should be sent in the single mode before automatically sending dummy bursts. This is the first transmit counter in all bursts.</p> <p>0: 0 burst</p> <p>1: 1 burst</p> <p>...</p> <p>N: N bursts</p> <p>Cannot be written when XCH=1.</p>

9.3.6.12 0x0040 SPI Bit-Aligned Transfer Configure Register (Default Value: 0x0000\_00A0)

Offset: 0x0040			Register Name: SPI_BATC
Bit	Read/Write	Default/Hex	Description
31	R/WAC	0x0	<p>TCE</p> <p>Transfer Control Enable</p> <p>In master mode, it is used to start to transfer the serial bit frame, it is only valid when Work Mode Select==0x10/0x11.</p> <p>0: Idle</p> <p>1: Initiates transfer</p> <p>Writing “1” to this bit will start to transfer serial bit frame (the value comes from the SPI TX Bit Register or SPI RX Bit Register), and will auto-clear after the bursts transfer completely. Writing ‘0’ to this bit has no effect.</p>
30	R/W	0x0	<p>MSMS</p> <p>Master Sample Standard</p> <p>0: Delay Sample Mode</p> <p>1: Standard Sample Mode</p> <p>In Standard Sample Mode, the SPI master samples the data at the standard rising edge of SCLK for each SPI mode;</p> <p>In Delay Sample Mode, the SPI master samples data at the edge that is half cycle delayed by the standard rising edge of SCLK defined in respective SPI mode.</p>
29:26	/	/	/
25	R/W1C	0x0	<p>TBC</p> <p>Transfer Bits Completed</p> <p>When set, this bit indicates that the last bit of the serial data frame in SPI TX Bit Register (or SPI RX Bit Register) has been transferred completely. Writing 1 to this bit clears it.</p> <p>0: Busy</p> <p>1: Transfer Completed</p> <p>It is only valid when Work Mode Select==0x10/0x11.</p>
24	R/W	0x0	<p>TBC_INT_EN</p> <p>Transfer Bits Completed Interrupt Enable</p> <p>0: Disable</p> <p>1: Enable</p> <p>It is only valid when Work Mode Select==0x10/0x11.</p>
23:22	/	/	/

Offset: 0x0040			Register Name: SPI_BATC
Bit	Read/Write	Default/Hex	Description
21:16	R/W	0x00	Configure the length of serial data frame (burst) of RX 000000: 0 bit 000001: 1 bit ... 100000: 32 bits Other values: reserved It is only valid when Work Mode Select==0x10/0x11, and cannot be written when TCE (bit31) is 1.
15:14	/	/	/
13:8	R/W	0x00	<b>TX_FRM_LEN</b> Configure the length of serial data frame (burst) of TX 000000: 0 bit 000001: 1 bit ... 100000: 32 bits Other values: reserved It is only valid when Work Mode Select==0x10/0x11, and cannot be written when TCE (bit31) is 1.
7	R/W	0x1	<b>SS_LEVEL</b> When control the SS signal manually, set this bit to '1' or '0' to control the level of SS signal. 0: Set SS to low 1: Set SS to high It is only valid when Work Mode Select==0x10/0x11, and only work in Mode0, cannot be written when TCE (bit31) is 1.
6	R/W	0x0	<b>SS_OWNER</b> SS Output Owner Select Usually, the controller sends the SS signal automatically with data together. When this bit is set to 1, the software must manually write SS_LEVEL to 1 or 0 to control the level of the SS signal. 0: SPI controller 1: Software It is only valid when Work Mode Select==0x10/0x11, and only work in Mode0, cannot be written when TCE (bit31) is 1.

Offset: 0x0040			Register Name: SPI_BATC
Bit	Read/Write	Default/Hex	Description
5	R/W	0x1	<p>SPOL</p> <p>SPI Chip Select Signal Polarity Control</p> <p>0: Active high polarity (0 = Idle)</p> <p>1: Active low polarity (1 = Idle)</p> <p>It is only valid when Work Mode Select==0x10/0x11, and only work in Mode0, cannot be written when TCE (bit31) is 1.</p>
4	/	/	/
3:2	R/W	0x0	<p>SS_SEL</p> <p>SPI Chip Select</p> <p>Select one of four external SPI Master/Slave Devices</p> <p>00: SPI_SS0 will be asserted</p> <p>01: SPI_SS1 will be asserted</p> <p>10: SPI_SS2 will be asserted</p> <p>11: SPI_SS3 will be asserted</p> <p>It is only valid when Work Mode Select= =0x10/0x11, and only work in Mode0, cannot be written when TCE (bit31) is 1.</p>
1:0	R/W	0x0	<p>Work Mode Select</p> <p>00: Data frame is byte aligned in standard SPI, dual-output/dual input SPI, dual IO SPI, and quad-output/quad-input SPI</p> <p>01: Reserved</p> <p>10: Data frame is bit aligned in 3-wire SPI</p> <p>11: Data frame is bit aligned in standard SPI</p>

**9.3.6.13 0x0044 SPI Bit-Aligned CLOCK Configuration Register (Default Value: 0x0000\_0000)**

Offset: 0x0044			Register Name: SPI_BA_CCR
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0x0	<p>CDR_N</p> <p>Clock Divide Rate (Master Mode Only)</p> <p>The SPI_SCLK is determined according to the following equation:  <math>SPI\_CLK = Source\_CLK / (2 * (CDR\_N + 1))</math>.</p> <p>This register is only valid when Work Mode Select==0x10/0x11.</p>

9.3.6.14 0x0048 SPI TX Bit Register (Default Value: 0x0000\_0000)

Offset: 0x0048			Register Name: SPI_TBR
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	<p>VTB</p> <p>The Value of the Transmit Bits</p> <p>This register is used to store the value of the transmitted serial data frame.</p> <p>In the process of transmission, the LSB is transmitted first.</p> <p>This register is only valid when Work Mode Select==0x10/0x11.</p>

9.3.6.15 0x004C SPI RX Bit Register (Default Value: 0x0000\_0000)

Offset: 0x004C			Register Name: SPI_RBR
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	<p>VRB</p> <p>The Value of the Receive Bits</p> <p>This register is used to store the value of the received serial data frame.</p> <p>In the process of transmission, the LSB is transmitted first.</p> <p>This register is only valid when Work Mode Select==0x10/0x11.</p>

9.3.6.16 0x0088 SPI Normal DMA Mode Control Register (Default Value: 0x0000\_00E5)

Offset: 0x0088			Register Name: SPI_NDMA_MODE_CTL
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:6	R/W	0x11	<p>SPI_ACT_M</p> <p>SPI NDMA Active Mode</p> <p>00: dma_active is low</p> <p>01: dma_active is high</p> <p>10: dma_active is controlled by dma_request (DRQ)</p> <p>11: dma_active is controlled by controller</p>
5	R/W	0x1	<p>SPI_ACK_M</p> <p>SPI NDMA Acknowledge Mode</p> <p>0: active fall do not care ack</p> <p>1: active fall must after detect ack is high</p>

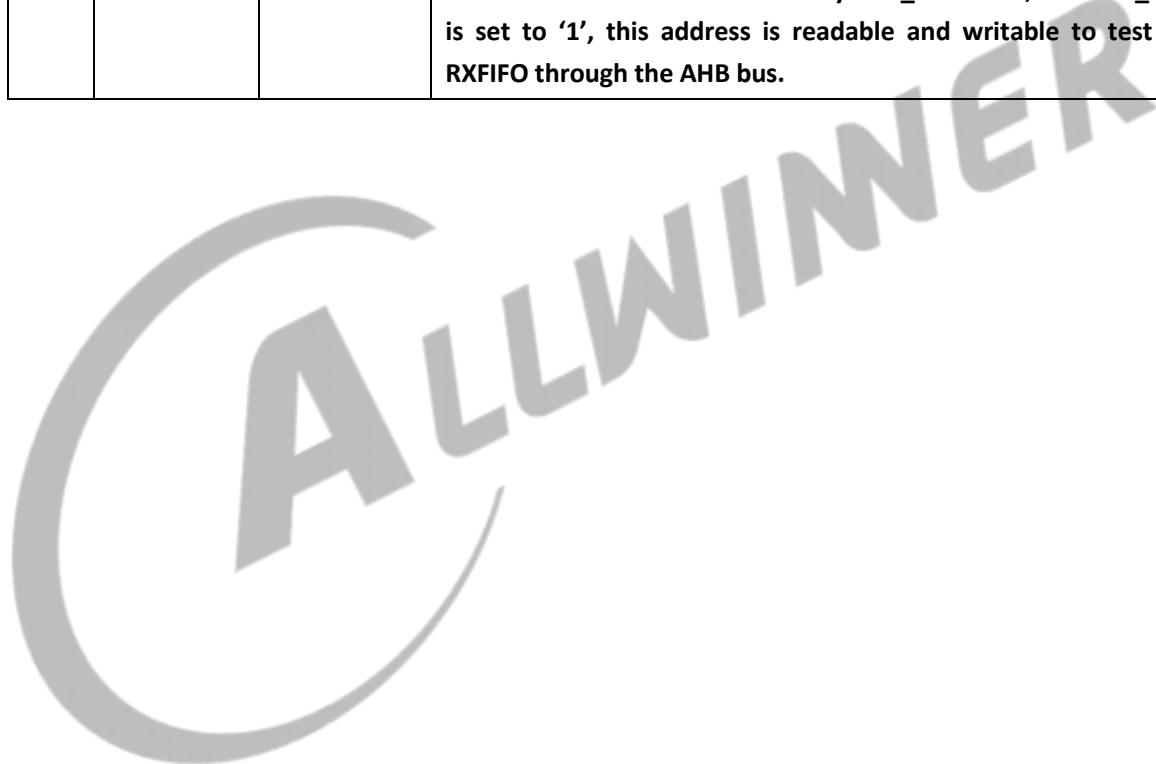
Offset: 0x0088			Register Name: SPI_NDMA_MODE_CTL
Bit	Read/Write	Default/Hex	Description
4:0	R/W	0x05	SPI_DMA_WAIT The counts of hold cycles from DMA last signal high to dma_active high

9.3.6.17 0x0200 SPI TX Data Register (Default Value: 0x0000\_0000)

Offset: 0x0200			Register Name: SPI_TXD
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	<p>TDATA Transmit Data</p> <p>This register can be accessed in the byte, half-word, or word unit by AHB. In the byte accessing method, if there are rooms in TXFIFO, one burst data is written to TXFIFO and the depth is increased by 1. In the half-word accessing method, two SPI burst data are written and the TXFIFO depth is increased by 2. In the word accessing method, four SPI burst data are written and the TXFIFO depth is increased by 4.</p> <p><b>Note: This address is writable-only if TF_TEST is '0', and if TF_TEST is set to '1', this address is readable and writable to test the TXFIFO through the AHB bus.</b></p>

9.3.6.18 0x0300 SPI RX Data Register (Default Value: 0x0000\_0000)

Offset: 0x0300			Register Name: SPI_RXD
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	<p>RDATA Receive Data</p> <p>This register can be accessed in the byte, half-word, or word unit by AHB. In the byte accessing method, if there are data in RXFIFO, the top word is returned and the RXFIFO depth is decreased by 1. In the half-word accessing method, two SPI bursts are returned and the RXFIFO depth is decreased by 2. In the word accessing method, the four SPI bursts are returned and the RXFIFO depth is decreased by 4.</p> <p><b>Note:</b> This address is readable-only if RF_TEST is '0', and if RF_TEST is set to '1', this address is readable and writable to test the RXFIFO through the AHB bus.</p>



## 9.4 SPI\_DBI

### 9.4.1 Overview

The D1s provides a 3/4 line SPI display bus interface (SPI\_DBI) for video data transmission. It supports DBI mode or SPI mode. The DBI mode is compatible with multiple video data formats at the same time. The SPI mode is used for low-cost display schemes.

The SPI mode has the following features:

- Full-duplex synchronous serial interface
- Master/slave configurable
- 8-bit wide by 64-entry FIFO for both transmit and receive data
- Polarity and phase of the Chip Select (SPI\_SS) and SPI Clock (SPI\_SCLK) are configurable
- Supports interrupts and DMA
- Supports mode0, mode1, mode2, and mode3
- Supports 3-wire/4-wire SPI
- Supports programmable serial data frame length: 1 bit to 32 bits
- Supports standard SPI, dual-output/dual-input SPI, dual I/O SPI, quad-output/quad-input SPI
- Supports maximum IO rate of the mass production: 100 MHz

The DBI mode has the following features:

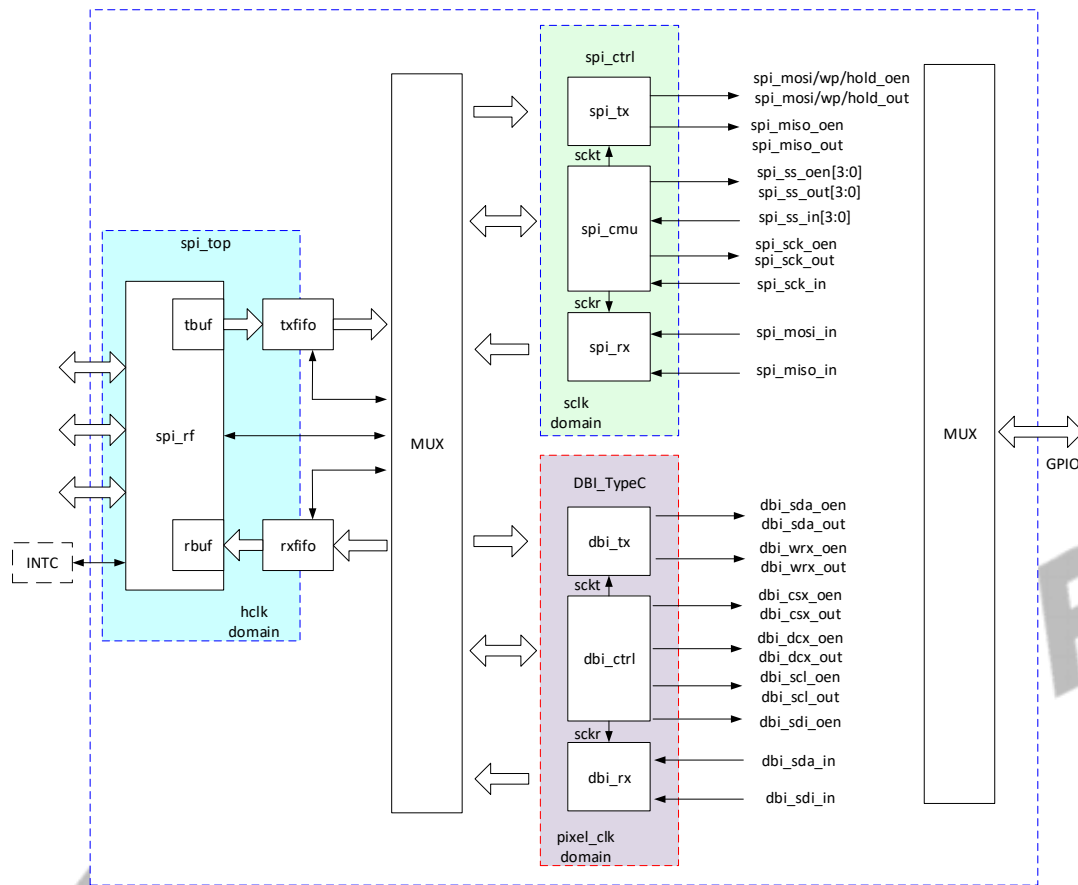
- Supports DBI Type C 3 Line/4 Line Interface Mode
- Supports 2 Data Lane Interface Mode
- Supports data source from CPU or DMA
- Supports RGB111/444/565/666/888 video format
- Maximum resolution of RGB666 240 x 320@30Hz with single data lane
- Maximum resolution of RGB888 240 x 320@60Hz or 320 x 480@30Hz with dual data lane
- Supports Tearing effect
- Supports software flexible control video frame rate

### 9.4.2 Block Diagram

Figure 9-30 shows a block diagram of the SPI\_DBI.



Figure 9-30 SPI\_DBI Block Diagram



SPI\_DBI contains the following sub-blocks:

Table 9-12 SPI\_DBI Sub-blocks

Sub-block	Description
spi_rf	Responsible for implementing the internal register, interrupt, and DMA Request.
spi_tbuf	The data length transmitted from AHB to TXFIFO is converted into 8 bits, then the data is written into the RXFIFO.
spi_rbuf	The block is used to convert the RXFIFO data into the reading data length of AHB.
txfifo, rxfifo	The data transmitted from the SPI to the external serial device is written into the TXFIFO; the data received from the external serial device into SPI is pushed into the RXFIFO.
spi_cmu	Responsible for implementing SPI bus clock, chip select, internal sample, and the generation of transfer clock.
spi_tx	Responsible for implementing SPI data transfer, the interface of the internal TXFIFO, and status register.

Sub-block	Description
spi_rx	Responsible for implementing SPI data receive, the interface of the internal RXFIFO, and status register.
dbi_ctrl	Responsible for implementing DBI bus clock, chip select, data command select, RGB format reshape.
dbi_tx	Responsible for implementing DBI data transfer, the interface of the internal TXFIFO, and status register.
dbi_rx	Responsible for implementing DBI data receive, the interface of the internal RXFIFO, and status register.

### 9.4.3 Functional Description

#### 9.4.3.1 External Signals

The following table describes the external signals of SPI\_DBI. When using SPI\_DBI, the corresponding PADs are selected as SPI\_DBI function via section 9.7 “[GPIO](#)”.

Table 9-13 SPI\_DBI External Signals

External Signal	Description	Type	
DBI Mode	DBI-CSX	Chip select signal, low active	I/O
	DBI-SCLK	Serial clock signal	I/O
	DBI-SDO	Data output signal	I/O
	DBI-SDI	Data input signal, the data is sampled on the rising edge and the falling edge	I/O
	DBI-TE	Tearing effect input, it is used to capture the external TE signal edge.	I/O
	DBI-DCX	DCX pin is the select output signal of data and command. DCX = 0: register command; DCX = 1: data or parameter.	I/O
	DBI-WRX	When DBI operates in dual data lane format, the RGB666 format 2 can use WRX to transfer data	I/O
SPI Mode	SPI1-CS	SPI1 chip select signal, low active When the device is not selected, data will not be accepted via the SI pin, and the SO pin will stop transmission.	I/O
	SPI1-CLK	SPI1 clock signal This pin is used to provide a clock to the device and is used to control the flow of data to and from the device.	I/O
	SPI1-MOSI	SPI1 master data out, slave data in	I/O

External Signal		Description	Type
	SPI1-MISO	SPI1 master data in, slave data out	I/O
	SPI1-WP	Write protection and active low It also can be used for serial data input and output for SPI Quad Input or Quad Output mode.	I/O
	SPI1-HOLD	When the device is selected and a serial sequence is underway, the HOLD pin is can be used to temporarily pause the serial communication with the master device without deselecting or resetting the device serial sequence. While the HOLD pin is asserted, the SO pin is at high impedance, and all transitions on the SCK pin and data on the SI pin are ignored. It also can be used for serial data input and output for SPI Quad Input or Quad Output mode.	I/O

### 9.4.3.2 Clock Sources

The SPI\_DBI controller gets 5 different clock sources, users can select one of them to make SPI\_DBI clock source. The following table describes the clock sources for SPI\_DBI. For more details on the clock setting, configuration, and gating information, see section 3.3 "[CCU](#)".

**Table 9-14 SPI\_DBI Clock Sources**

Clock Sources	Description
HOSC	24 MHz Crystal
PLL_PERI(1X)	Peripheral Clock, the default value is 600 MHz
PLL_PERI(2X)	Peripheral Clock, the default value is 1200 MHz
PLL_AUDIO0(DIV2)	Audio Clock, the default value is 1536 MHz
PLL_AUDIO0(DIV5)	Audio Clock, the default value is 614.4 MHz

### 9.4.3.3 Typical Application

Figure 9-31 shows the application block diagram when the SPI master device is connected to a slave device.

Figure 9-31 SPI Application Block Diagram

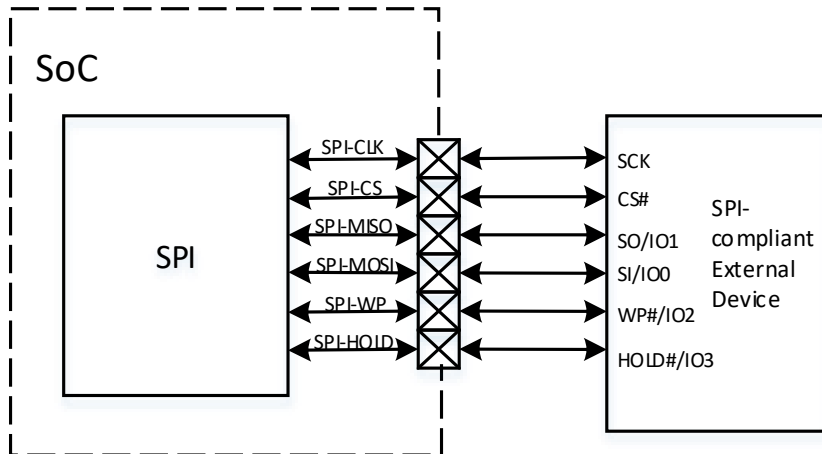
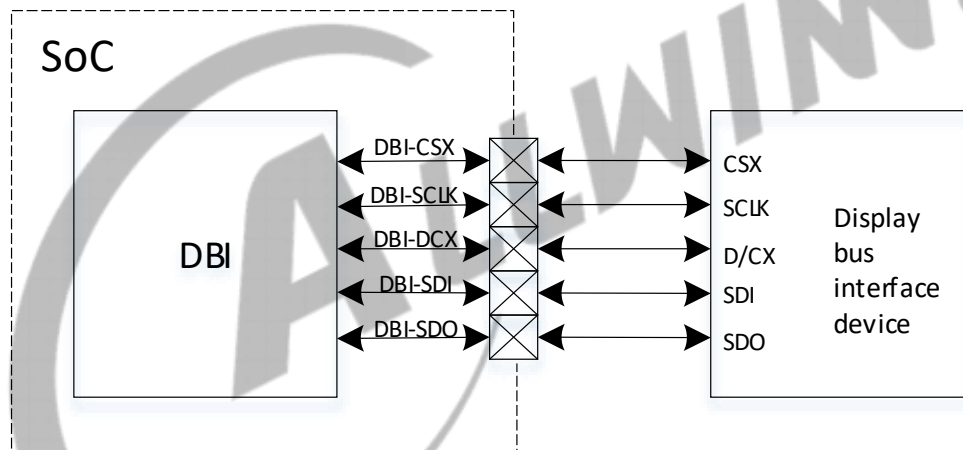


Figure 9-32 shows the application block diagram when the DBI master device is connected to a display bus interface device.

Figure 9-32 DBI Application Block Diagram



#### 9.4.3.4 SPI Transmission Format

The SPI supports 4 different formats for data transmission. The software can select one of the four modes in which the SPI works by setting the bit1 (Polarity) and bit0 (Phase) of [SPI\\_TCR](#). The SPI controller master uses the SPI\_SCLK signal to transfer data in and out of the shift register. Data is clocked using any one of four programmable clock phase and polarity combinations.

The CPOL ([SPI\\_TCR\[1\]](#)) defines the polarity of the clock signal (SPI\_SCLK). The SPI\_SCLK is a high level when CPOL is '1' and it is a low level when CPOL is '0'. The CPHA ([SPI\\_TCR\[0\]](#)) decides whether the leading edge of SPI\_SCLK is used to setup or sample data. The leading edge is used to setup data when CPHA is '1', and sample data when CPHA is '0'. The following table lists the four modes.

Table 9-15 SPI Transmit Format

SPI Mode	Polarity (CPOL)	Phase (CPHA)	Leading Edge	Trailing Edge
mode0	0	0	Sample on the rising edge	Setup on the falling edge
mode1	0	1	Setup on the rising edge	Sample on the falling edge
mode2	1	0	Sample on the falling edge	Setup on the rising edge
mode3	1	1	Setup on the falling edge	Sample on the rising edge

Figure 9-33 and Figure 9-34 describe four waveforms for SPI\_SCLK.

Figure 9-33 SPI Phase 0 Timing Diagram

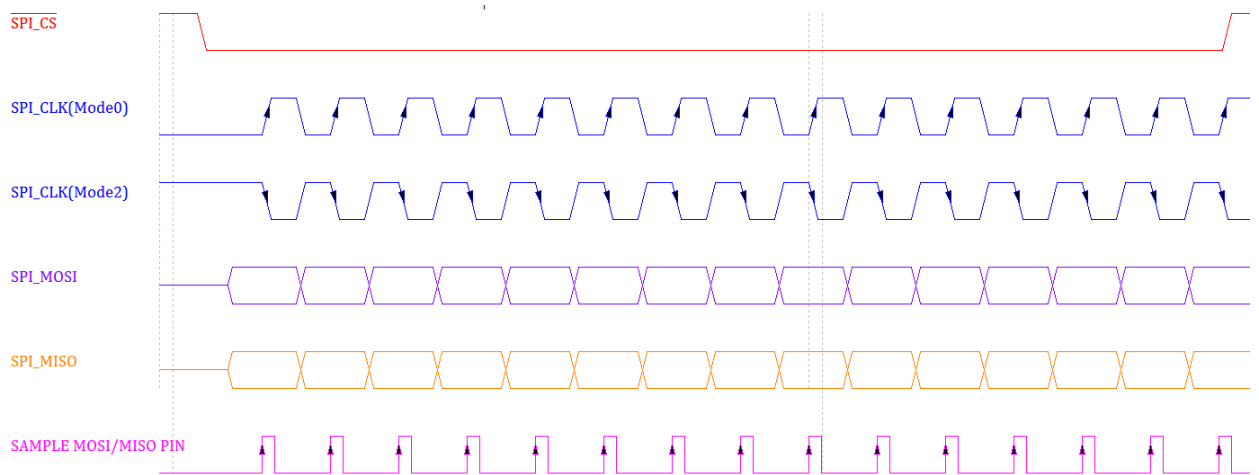
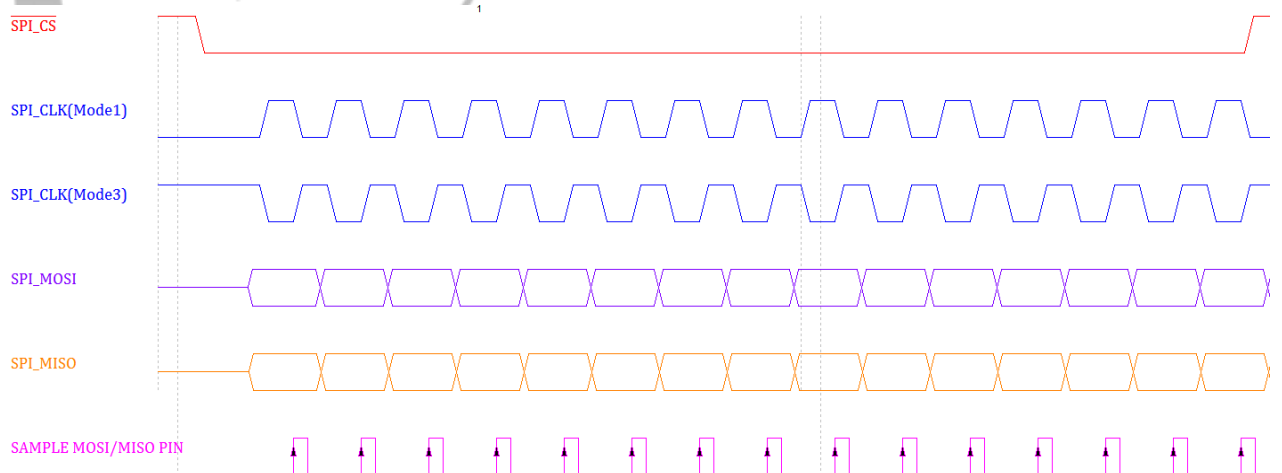


Figure 9-34 SPI Phase 1 Timing Diagram



### 9.4.3.5 SPI Master and Slave Mode

The SPI controller can be configured to a master or slave device. The master mode is selected by setting the MODE bit ([SPI\\_GCR\[1\]](#)); the slave mode is selected by clearing the MODE bit.

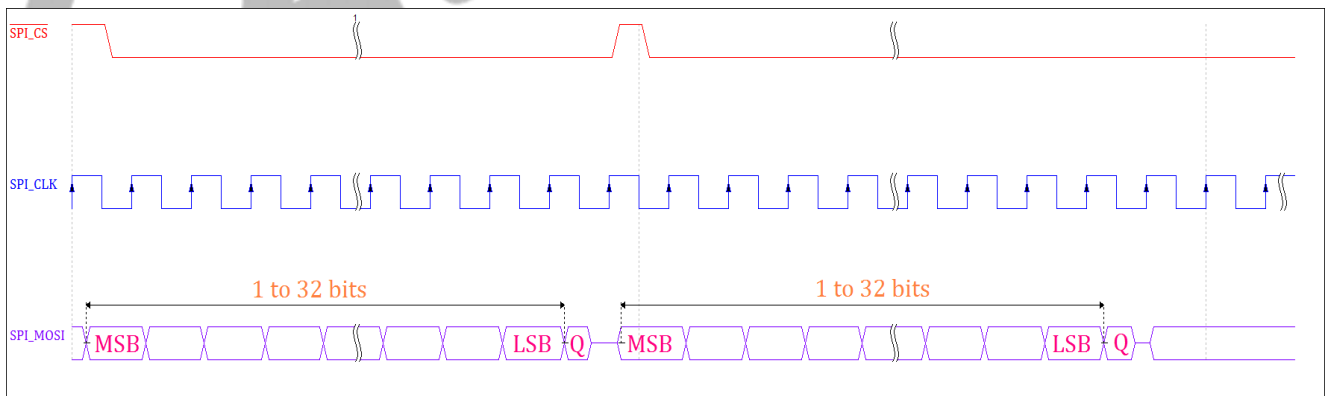
In master mode, the SPI\_CLK is generated and transmitted to the external device, and the data from the TX FIFO is transmitted on the MOSI pin, the data from the slave is received on the MISO pin and sent to RX FIFO. The Chip Select (SPI\_SS) is an active low signal, and it must be set low before the data are transmitted or received. The SPI\_SS can be selected the auto control mode or the software manual control mode. When using auto control, the SS\_OWNER ([SPI\\_TCR\[6\]](#)) must be cleared (default value is 0); when using manual control, the SS\_OWNER must be set. And the level of SPI\_SS is controlled by SS\_LEVEL ([SPI\\_TCR\[7\]](#)).

In slave mode, after the software selects the MODE bit ([SPI\\_GCR\[1\]](#)) to '0', it waits for master initiate a transaction. When the master asserts SPI\_SS, and SPI\_CLK is transmitted to the slave, the slave data is transmitted from TX FIFO on the MISO pin, and the data from the MOSI pin is received in RX FIFO.

### 9.4.3.6 SPI 3-Wire Mode

The SPI 3-wire mode is only valid when the SPI controller work in master mode, and is selected when the Work Mode Select bit ([SPI\\_BATC\[1:0\]](#)) is equal to 0x2. And in the 3-wire mode, the input data and the output data use the same single data line. The following figure describes the 3-wire mode.

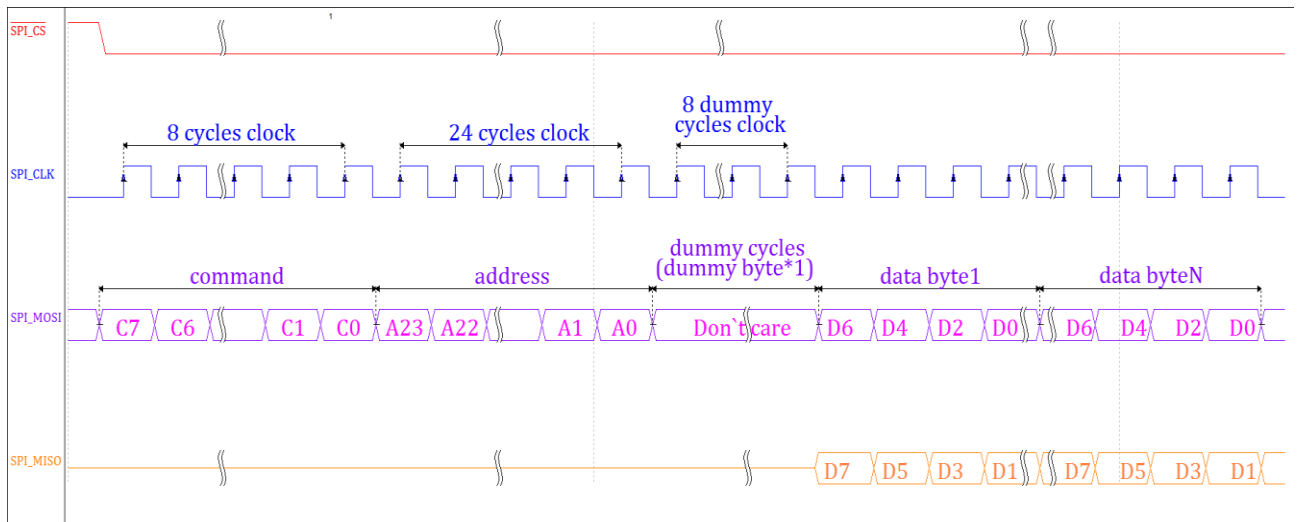
Figure 9-35 SPI 3-Wire Mode



### 9.4.3.7 SPI Dual-Input/Dual-Output and Dual I/O Mode

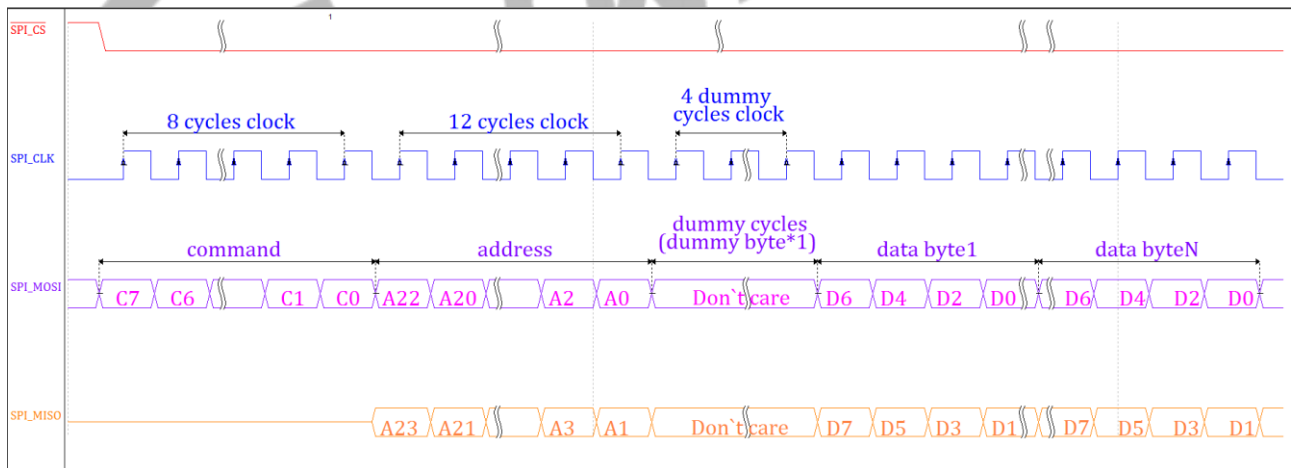
The dual read mode (SPI x2) is selected when the DRM is set in [SPI\\_BCC\[28\]](#). Using the dual mode allows data to be transferred to or from the device at double the rate of standard single mode SPI devices, the data can be read at fast speed using two data bits (MOSI and MISO) at a time. The following figure describes the dual-input/dual-output SPI (Figure 9-36) and the dual I/O SPI (Figure 9-37).

Figure 9-36 SPI Dual-Input/Dual-Output Mode



In the dual-input/dual-output SPI mode, the command, address, and the dummy bytes output in a unit of a single bit in serial mode through the SPI\_MOSI line, only the data bytes are output (write) and input (read) in a unit of dual bits through the SPI\_MOSI and SPI\_MISO.

Figure 9-37 SPI Dual I/O Mode

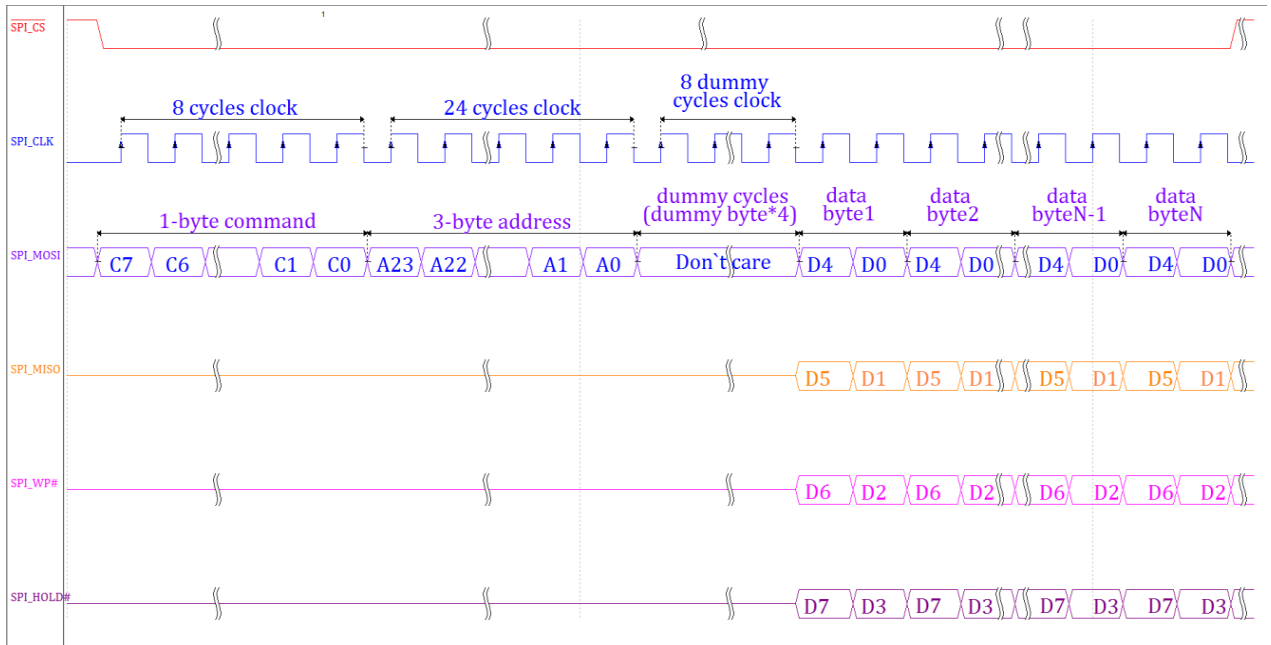


In the dual I/O SPI mode, only the command bytes output in a unit of a single bit in serial mode through the SPI\_MOSI line. The address bytes and the dummy bytes output in a unit of dual bits through the SPI\_MOSI and SPI\_MISO. And the data bytes output (write) and input (read) in a unit of dual bits through the SPI\_MOSI and SPI\_MISO.

### 9.4.3.8 SPI Quad-Input/Quad-Output Mode

The quad read mode (SPI x4) is selected when the Quad\_EN is set in [SPI\\_BCC\[29\]](#). Using the quad mode allows data to be transferred to or from the device at 4 times the rate of standard single mode SPI devices, the data can be read at fast speed using four data bits (MOSI, MISO, IO2 (WP#) and IO3 (HOLD#)) at the same time. The following figure describes the quad-input/quad-output SPI.

Figure 9-38 SPI Quad-Input/Quad-Output Mode



In the quad-input/quad-output SPI mode, the command, address, and the dummy bytes output in a unit of a single bit in serial mode through the SPI\_MOSI line. Only the data bytes output (write) and input (read) in a unit of quad bits through the SPI\_MOSI, SPI\_MISO, SPI\_WP#, and SPI\_HOLD#.

### 9.4.3.9 Transmission/Reception Bursts in Master Mode

In SPI master mode, the transmission and reception bursts (byte in unit) are configured before the SPI transfers serial data between the processor and external device. The transmission bursts are written in MWTC (bit[23:0]) of the [SPI Master Transmit Counter Register](#). The transmission bursts in single mode before automatically sending dummy bursts are written in STC (bit[23:0]) of the [SPI Master Transmit Counter Register](#). For dummy data, the SPI controller can automatically send before receiving by writing DBC (bit[27:24]) in the [SPI Master Transmit Counter Register](#). If users do not use the SPI controller to send dummy data automatically, then the dummy bursts are used as the transmission counters to write together in MWTC (bit[23:0]) of the [SPI Master Transmit Counter Register](#). In master mode, the total burst numbers are written in MBC (bit[23:0]) of the [SPI Master Transmit Counter Register](#). When all transmission and reception bursts are transferred, the SPI controller will send a completed interrupt, at the same time, the SPI controller will clear DBC, MWTC, and MBC.



### 9.4.3.10 SPI Sample Mode and Run Clock Configuration

The SPI controller runs at 3 kHz–100 MHz at its interface to external SPI devices. The internal SPI clock should run at the same frequency as the outgoing clock in the master mode. The SPI clock is selected from different clock sources, the SPI must configure different work mode. There are three work modes: normal sample mode, delay half-cycle sample mode, delay one-cycle sample mode. Delay half-cycle sample mode is the default mode of the SPI controller. When the SPI runs at 40 MHz or below 40 MHz, the SPI can work at normal sample mode or delay half-cycle sample mode. When the SPI runs over 80 MHz, setting the SDC bit in the [SPI Transfer Control Register](#) to ‘1’ makes the internal read sample point with a half-cycle delay of SPI\_CLK, which is used in high speed read operation to reduce the error caused by the time delay of SPI\_CLK between master and slave. Table 9-16 and Table 9-17 show the different configurations of the SPI sample mode.

**Table 9-16 SPI Old Sample Mode and Run Clock**

SPI Sample Mode	SDM(bit13)	SDC(bit11)	Run Clock
normal sample	1	0	<=24 MHz
delay half cycle sample	0	0	<=40 MHz
delay one cycle sample	0	1	>=80 MHz



The remaining spectrum is not recommended. Because when the output delay of SPI flash (refer to the datasheet of the manufactures for the specific delay time) is the same with the half-cycle time of SPI working clock, the variable edge of the output data for the device bumps into the clock sampling edge of the controller, so setting 1 cycle of sampling delay would cause stability problem.

**Table 9-17 SPI New Sample Mode**

SPI Sample Mode	SDM (bit13)	SDC (bit11)	SDC1 (bit15)
normal sample	1	0	0
delay half cycle sample	0	0	0
delay one cycle sample	0	1	0
delay 1.5 cycle sample	1	1	0
delay 2 cycle sample	1	0	1
delay 2.5 cycle sample	0	0	1
delay 3 cycle sample	0	1	1

### 9.4.3.11 DBI 3-Line Interface Writing and Reading Timing

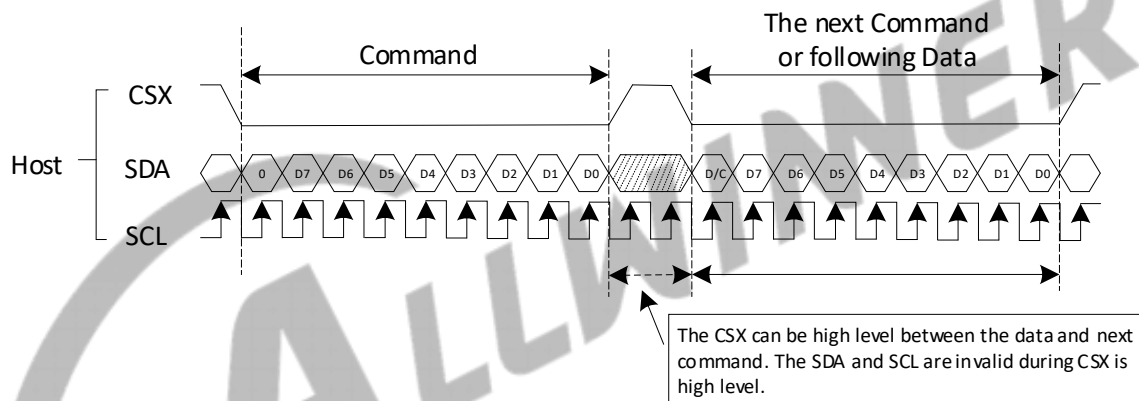
The 3-line DBI Interface I contains CSX, SDA, and SCL, where SDA shares this port for bidirectional port data input and output.

The 3-line DBI Interface II contains CSX, SDA, SCL, and SDI; Data input and output ports are independent of each other.

Since the 3-line display bus mode has no Data/Command data line indicating whether Data or Command is currently being transmitted, an extra bit is added to the data-stream before MSB to indicate whether Data or Command is currently being transmitted. (0: Command, 1: Data)

The following figure shows the writing operation format of 3-line DBI Interface I and Interface II.

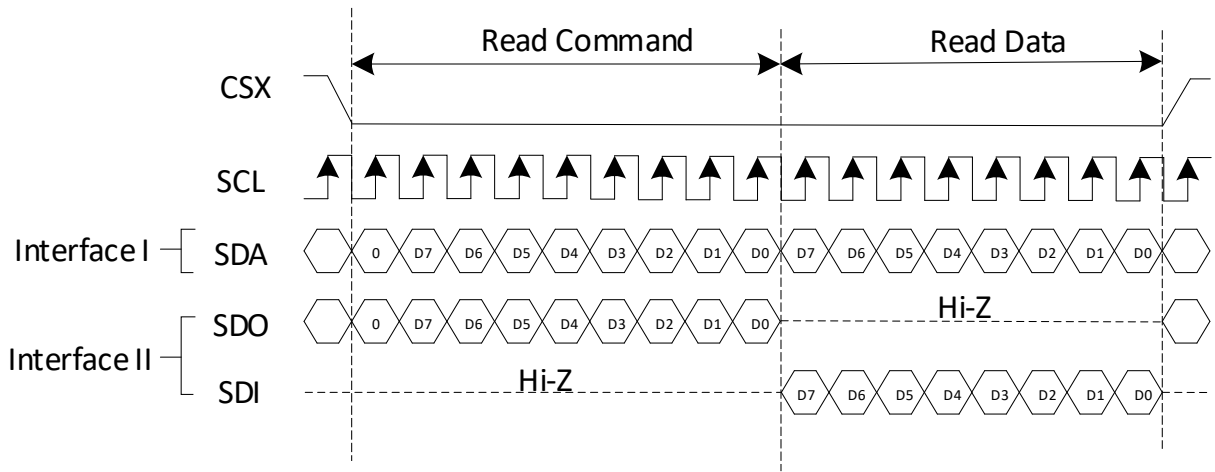
**Figure 9-39 DBI 3-Line Display Bus Serial Interface Writing Operation Format**



The 3-line DBI Interface I uses the SDA port as bidirectional data input and output port. There are only three cases of data reading volume, 8bits/24bits/32bits, and the first data sampled is high.

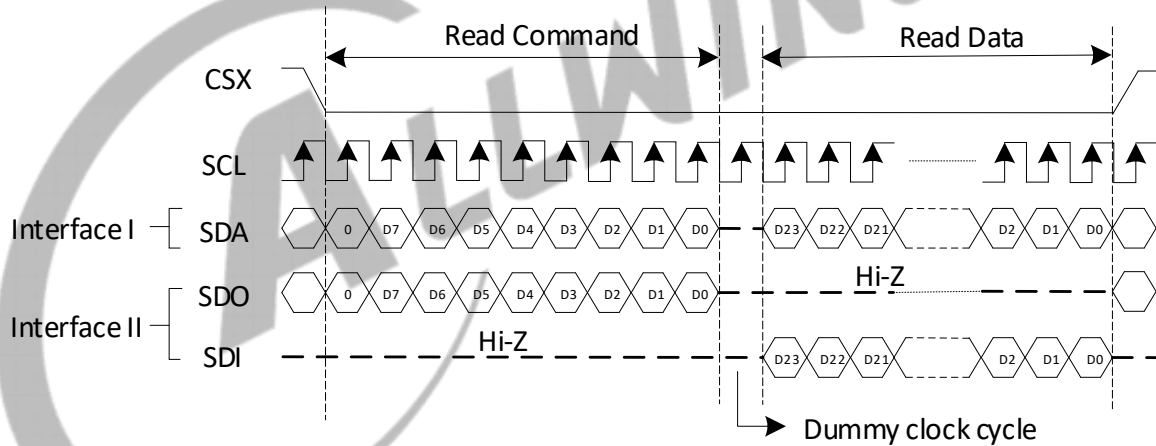
The following figure shows the 8 bits reading operation format of 3-line DBI Interface I and Interface II. After the read command is transmitted, the data is read immediately with on dummy period.

Figure 9-40 DBI 3-Line Display Bus Serial Interface 8-bit Reading Operation Format



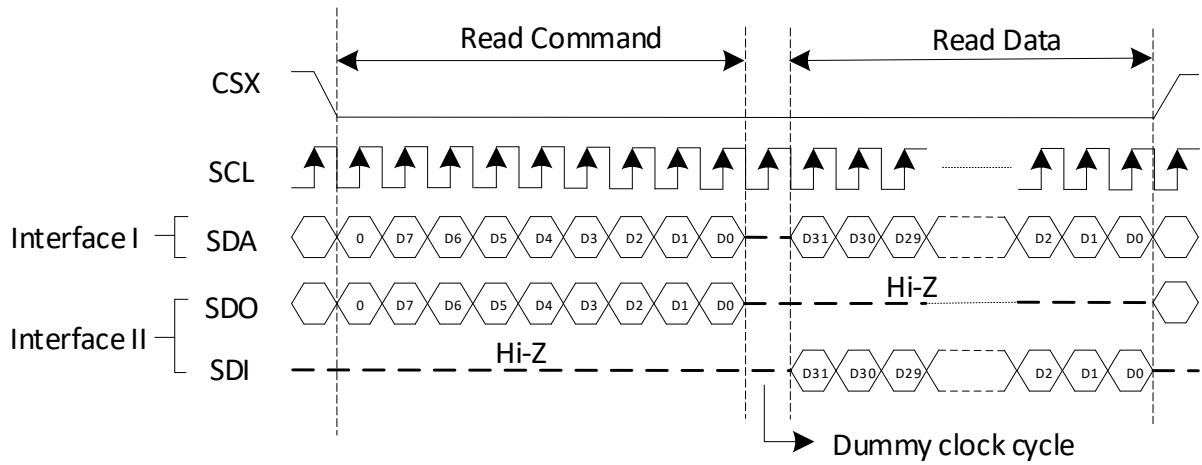
The following figure shows the 24 bits reading operation format of 3-line DBI Interface I and Interface II. After the read command is transmitted, the data is read after waiting for the dummy clock cycle.

Figure 9-41 DBI 3-Line Display Bus Serial Interface 24-bit Reading Operation Format



The following figure shows the 32 bits reading operation format of 3-line DBI Interface I and Interface II. After the read command is transmitted, the data is read after waiting for the dummy clock cycle.

Figure 9-42 DBI 3-Line Display Bus Serial Interface 32-bit Reading Operation Format



#### 9.4.3.12 DBI 4-Line Interface Writing and Reading Timing

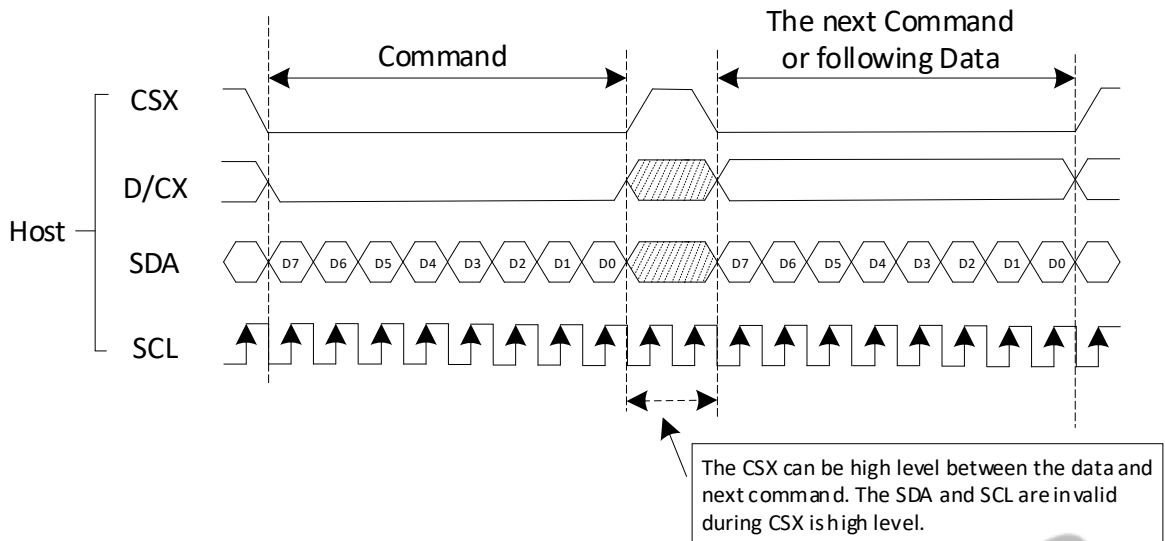
The 4-line DBI Interface I contains CSX, D/CX, SDA, and SCL, where SDA shares this port for bidirectional port data input and output.

The 4-line DBI Interface II contains CSX, D/CX, SDA, SCL, and SDI; Data input and output ports are independent of each other.

Since the 4-line display bus mode has a Data/Command data line indicating whether Data or Command is currently being transmitted (0: Command, 1: Data). So there is no need to add an extra bit to data-stream before MSB like the 3-line DBI.

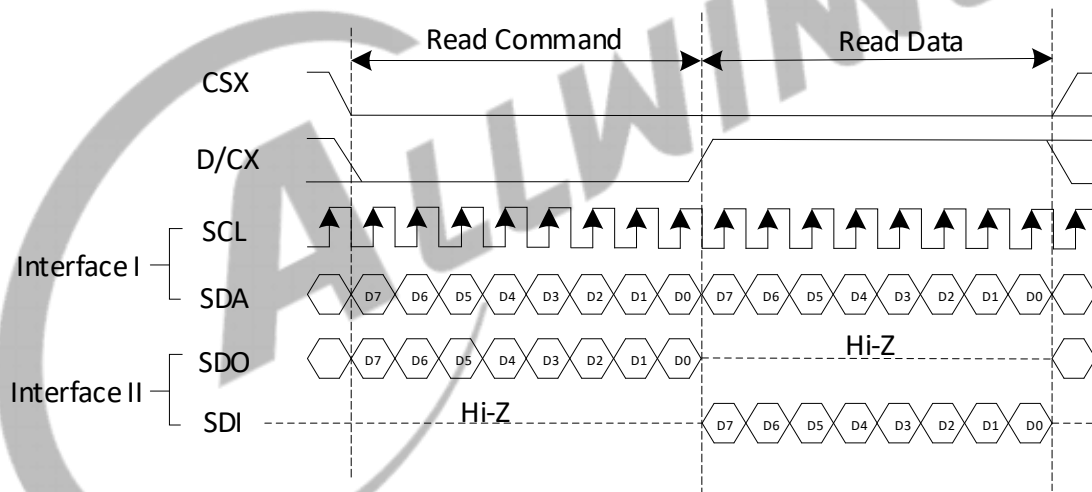
The following figure shows the writing operation format of 4-line DBI Interface I and Interface II.

Figure 9-43 DBI 4-Line Display Bus Serial Interface Writing Operation Format



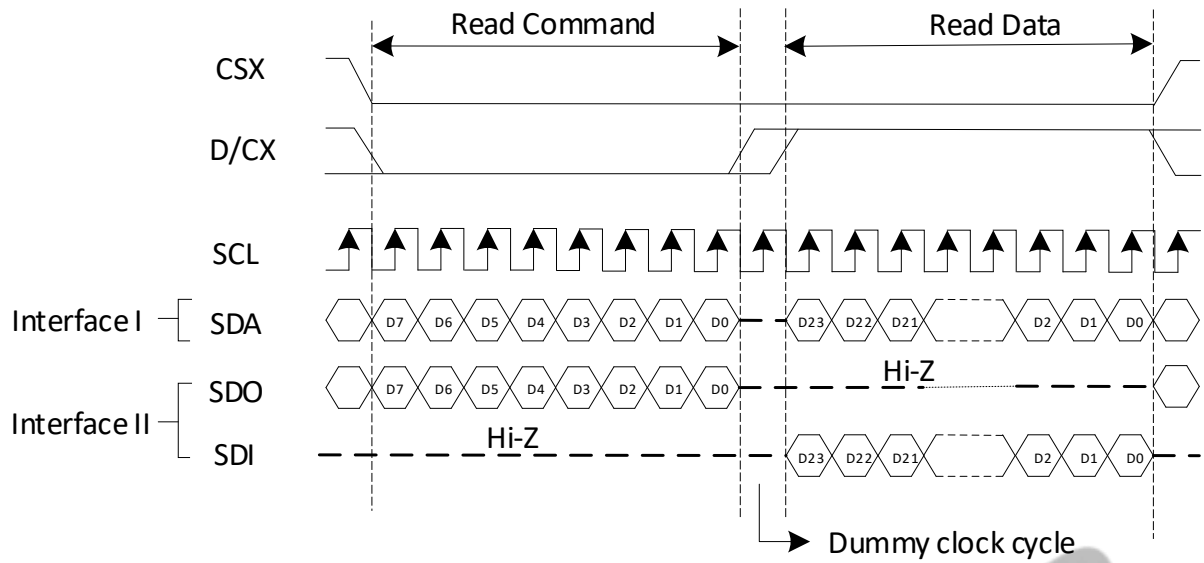
The following figure shows the 8 bits reading operation format of 4-line DBI Interface I and Interface II.

Figure 9-44 DBI 4-Line Display Bus Serial Interface 8-bit Reading Operation Format



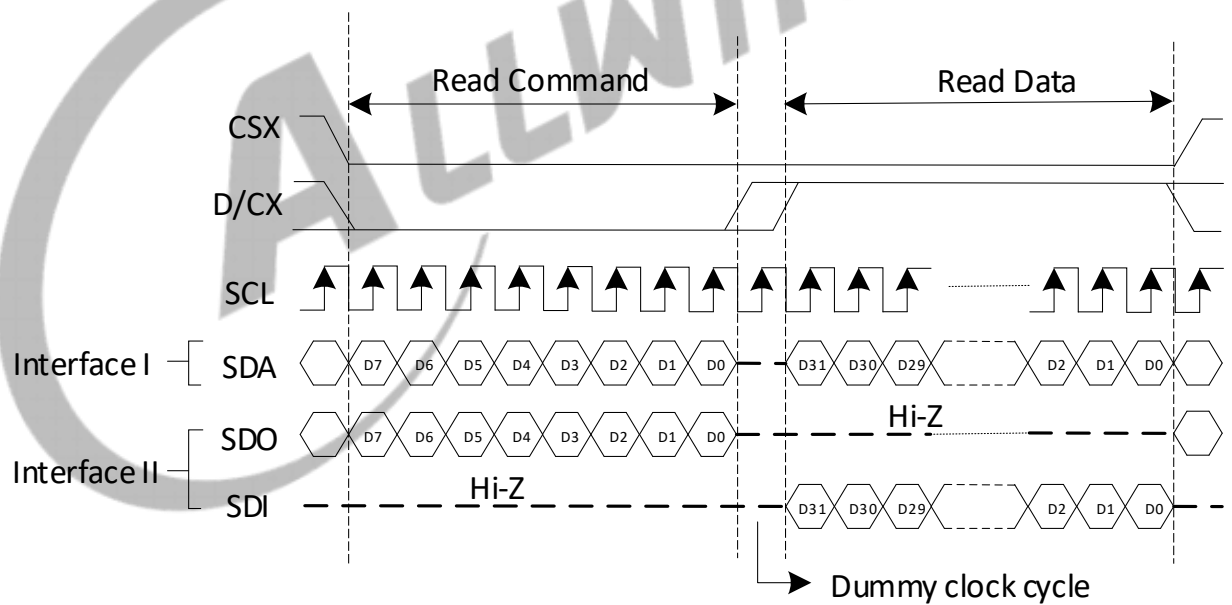
The following figure shows the 24 bits reading operation format of 4-line DBI Interface I and Interface II.

Figure 9-45 DBI 4-Line Display Bus Serial Interface 24-bit Reading Operation Format



The following figure shows the 32 bits reading operation format of 4-line DBI Interface I and Interface II.

Figure 9-46 DBI 4-Line Display Bus Serial Interface 32-bit Reading Operation Format



9.4.3.13 DBI 3-Line Interface Transmit Video Format

Figure 9-47 RGB111 3-Line Interface Transmit Video Format

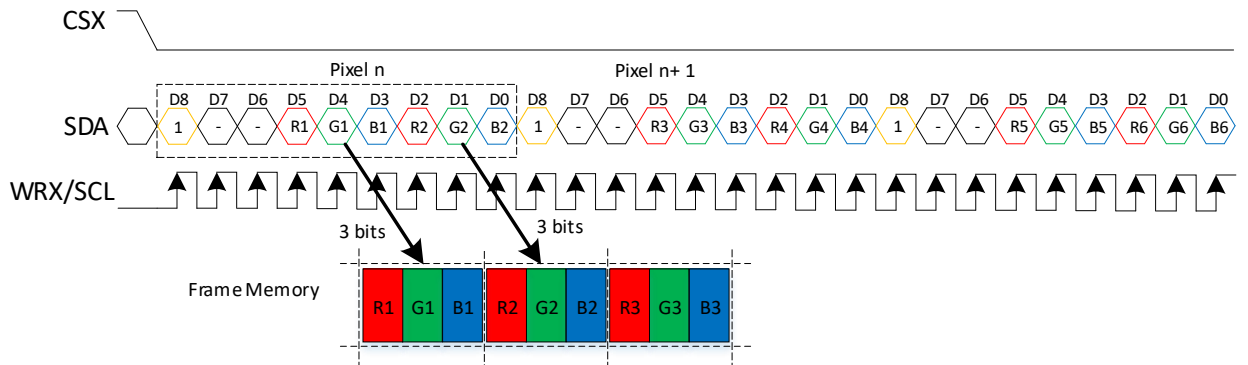
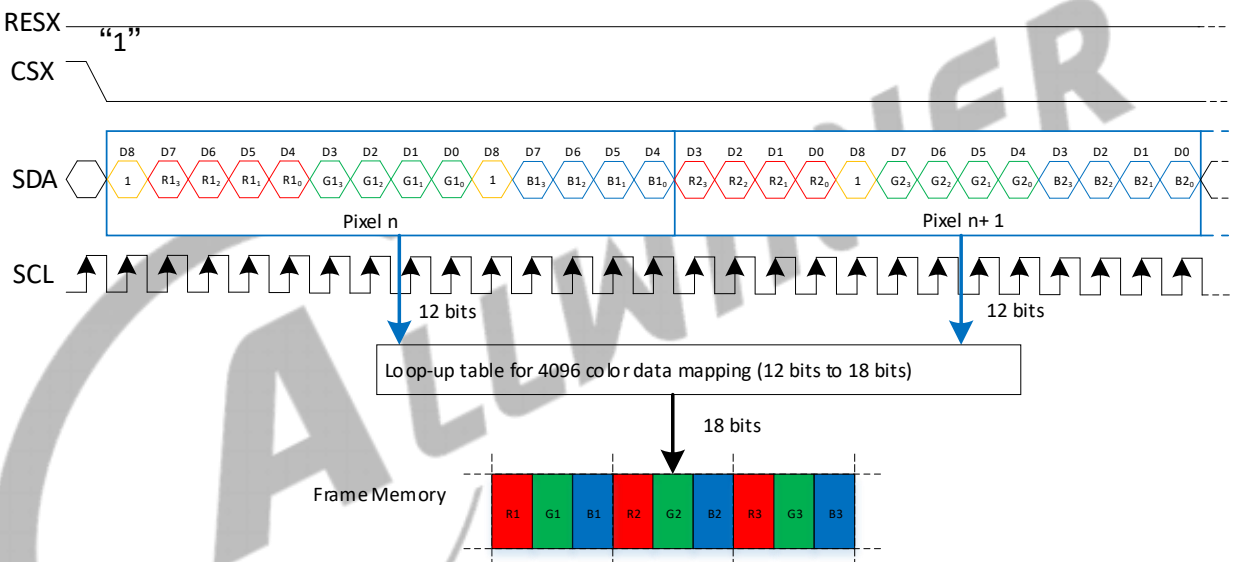


Figure 9-48 RGB444 3-Line Interface Transmit Video Format



- Note 1. Pixel data with 12-bit color depth information
- Note 2. The most significant bits are: Rx3, Gx3 and Bx3
- Note 3. The least significant bits are: Rx0, Gx0 and Bx0

Figure 9-49 RGB565 3-Line Interface Transmit Video Format

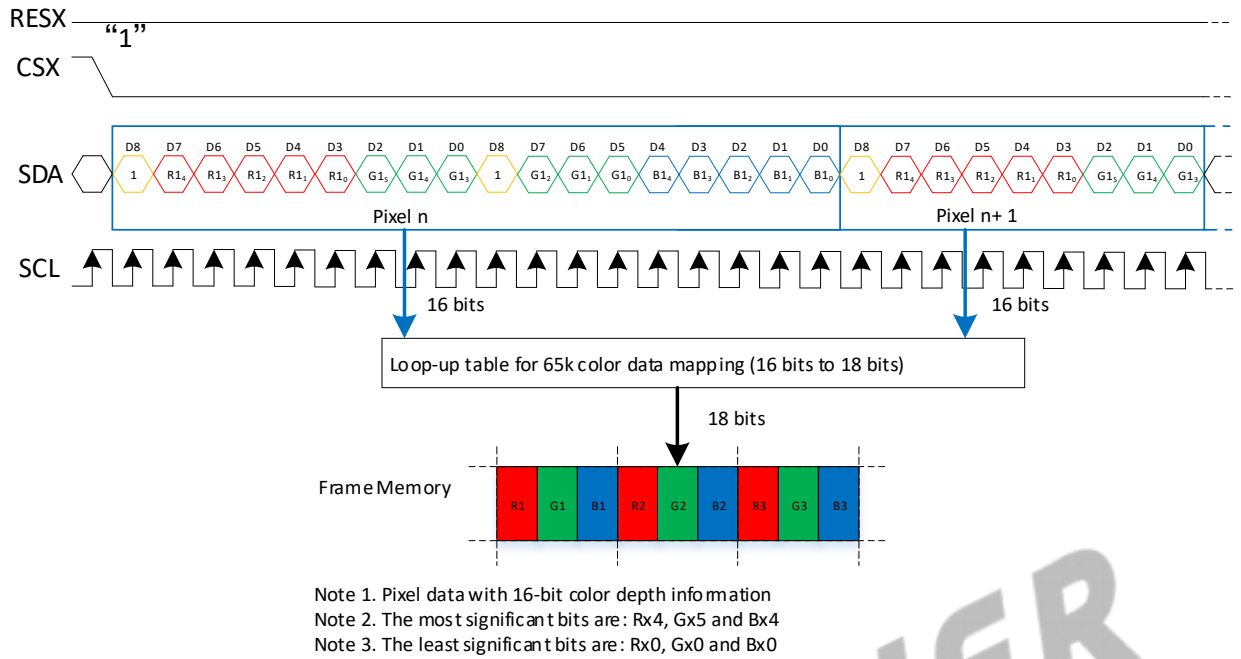
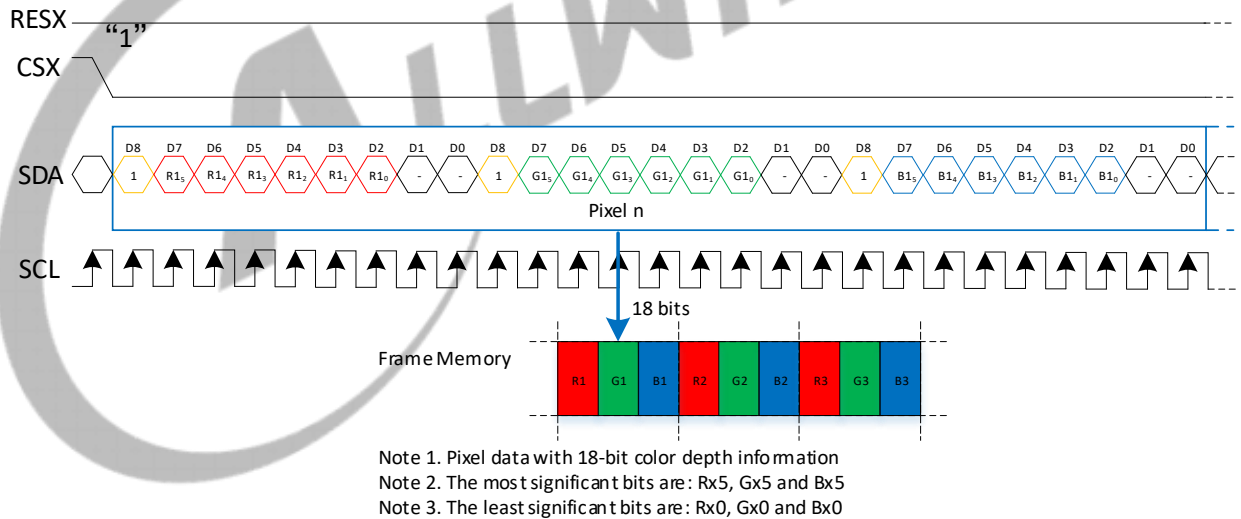


Figure 9-50 RGB666 3-Line Interface Transmit Video Format





9.4.3.14 DBI 4-Line Interface Transmit Video Format

Figure 9-51 RGB111 4-Line Interface Transmit Video Format

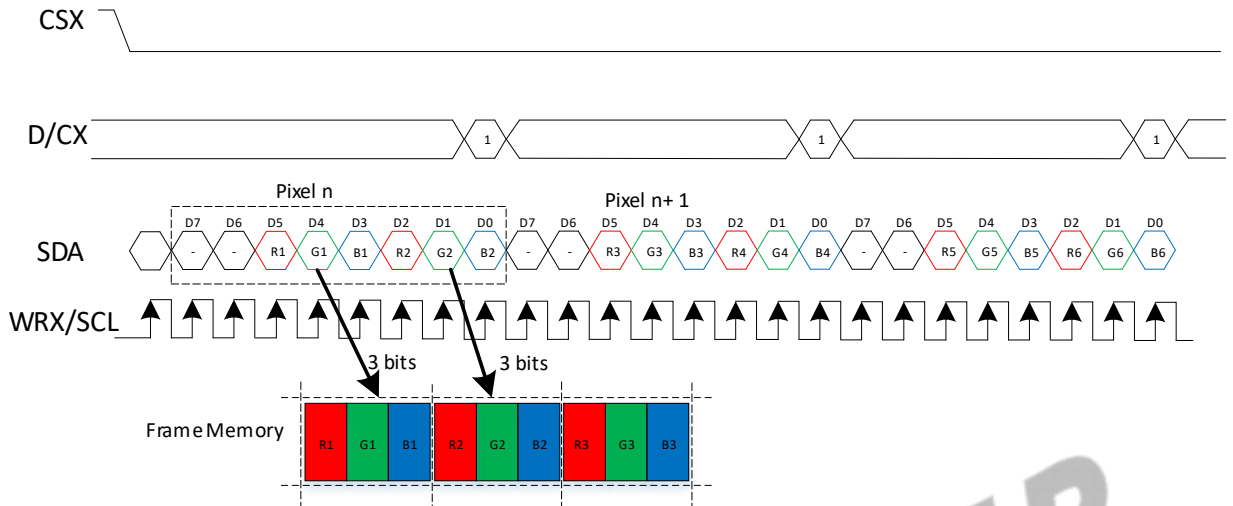
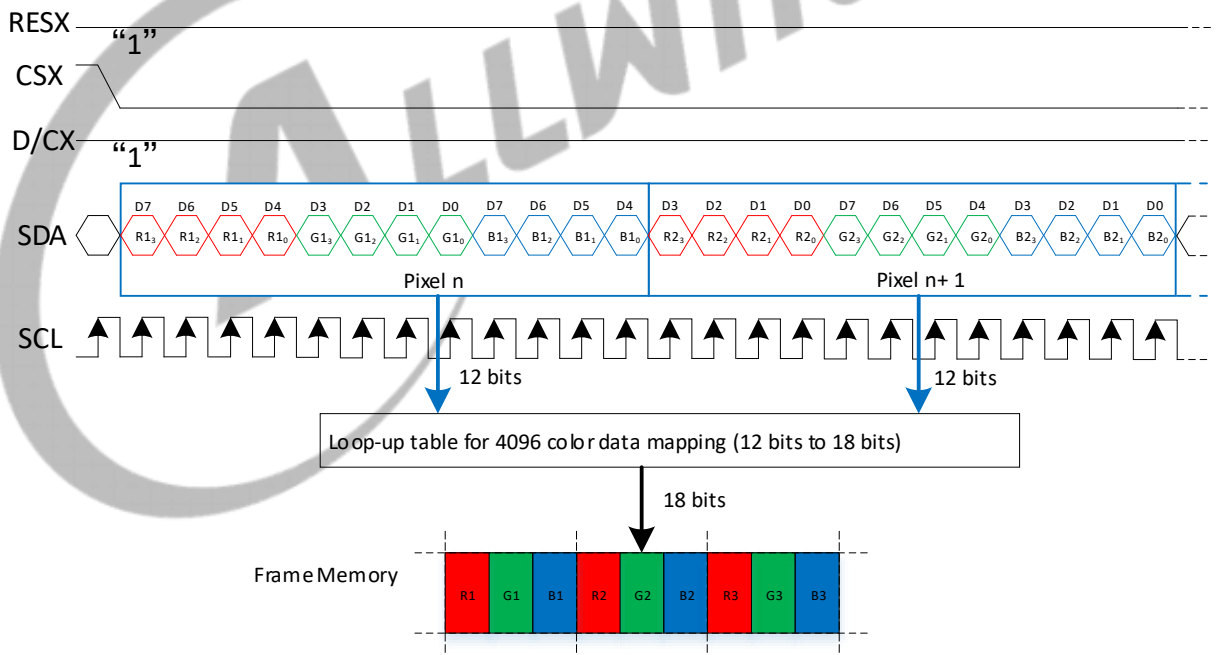


Figure 9-52 RGB444 4-Line Interface Transmit Video Format



- Note 1. Pixel data with 12-bit color depth information
- Note 2. The most significant bits are: Rx3, Gx3 and Bx3
- Note 3. The least significant bits are: Rx0, Gx0 and Bx0

Figure 9-53 RGB565 4-Line Interface Transmit Video Format

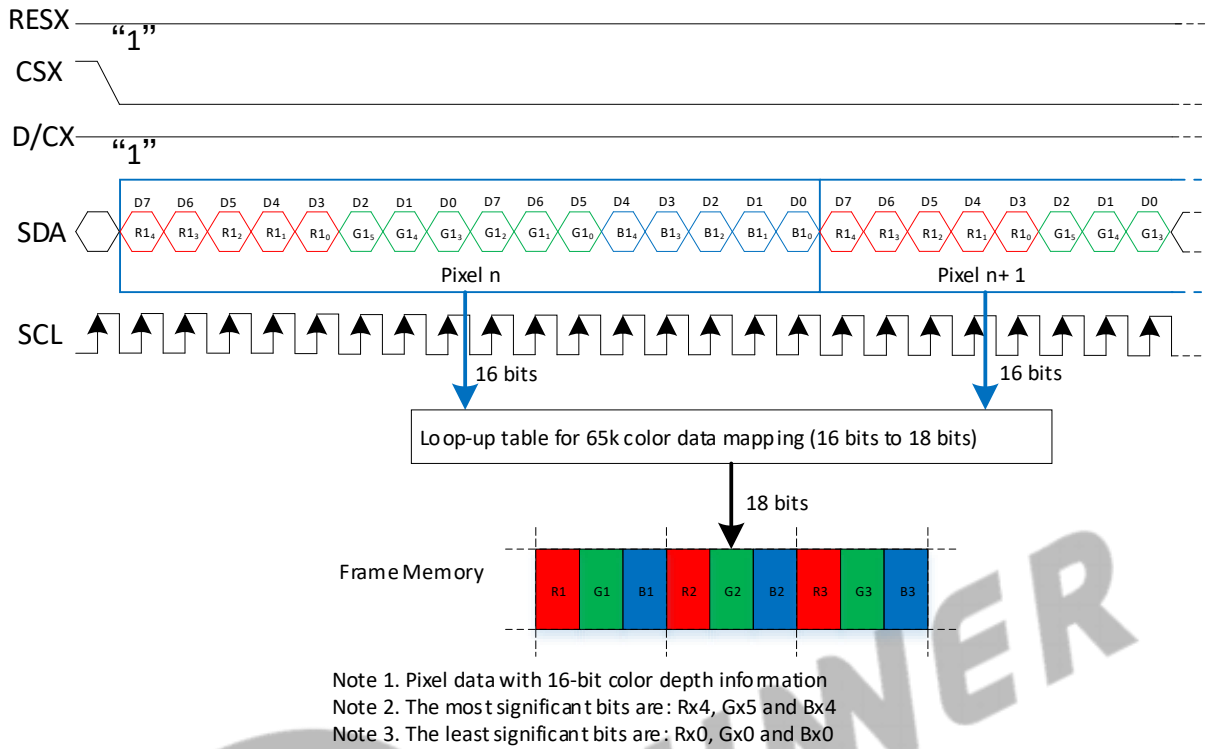
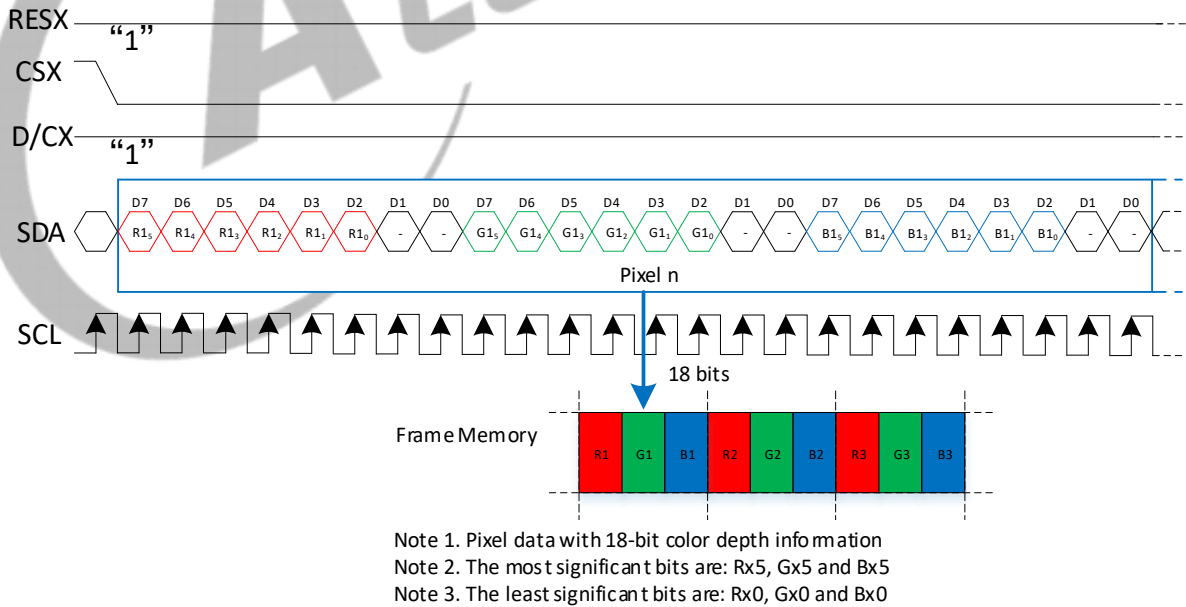


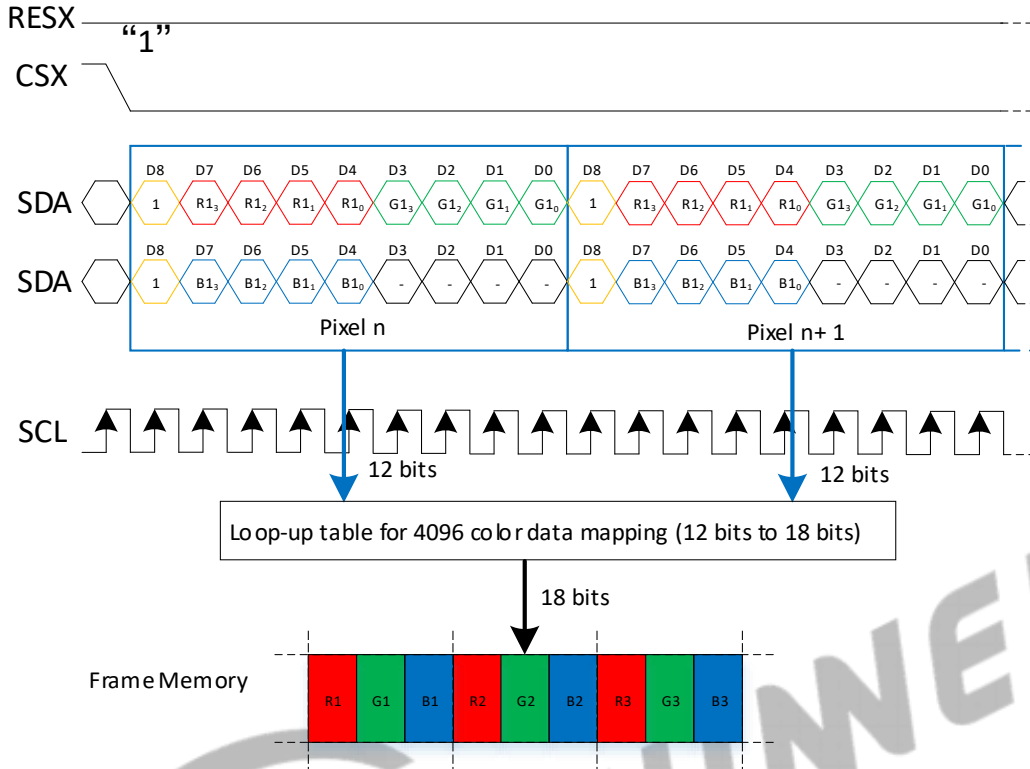
Figure 9-54 RGB666 4-Line Interface Transmit Video Format



9.4.3.15 DBI 2 Data Lane Interface Transmit Video Format

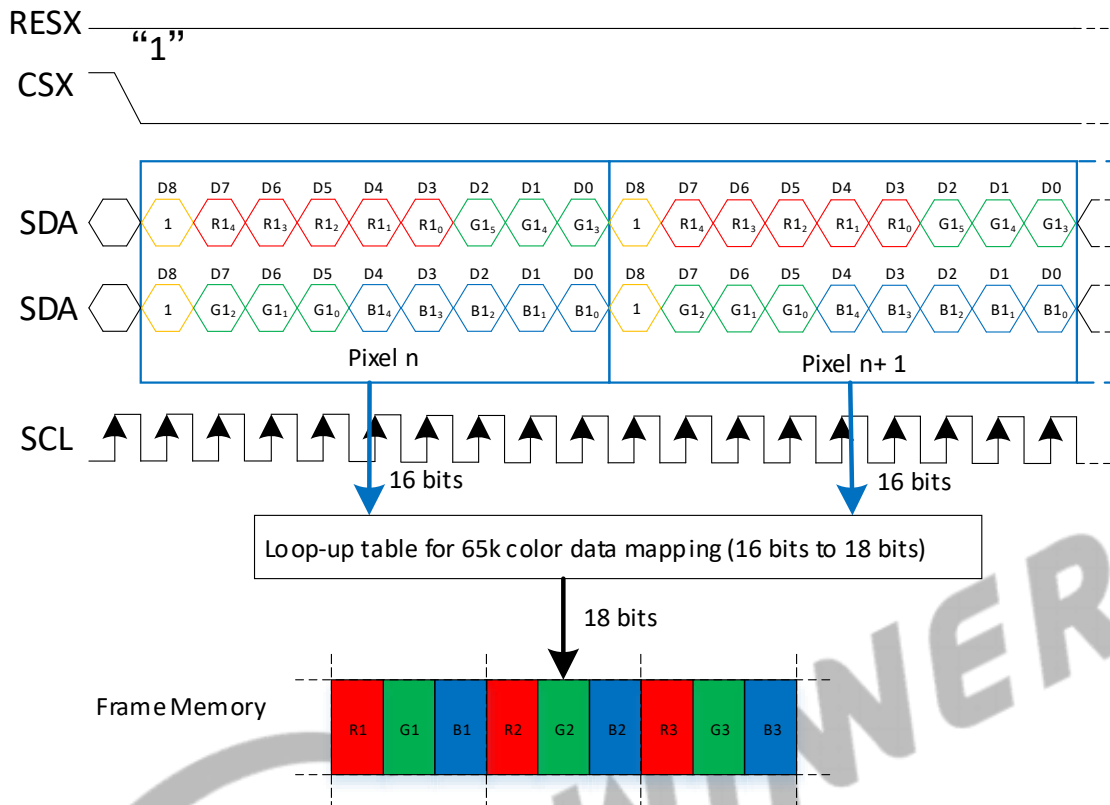
For RGB444:

Figure 9-55 RGB444 2 Data Lane Interface Transmit Video Format



- Note 1. Pixel data with 12-bit color information
- Note 2. The most significant bits are: R<sub>x3</sub>, G<sub>x3</sub> and B<sub>x3</sub>
- Note 3. The least significant bits are: R<sub>x0</sub>, G<sub>x0</sub> and B<sub>x0</sub>

Figure 9-56 RGB565 2 Data Lane Interface Transmit Video Format



- Note 1. Pixel data with 16-bit color information
- Note 2. The most significant bits are: Rx4, Gx5 and Bx4
- Note 3. The least significant bits are: Rx0, Gx0 and Bx0

Figure 9-57 RGB666 2 Data Lane Interface Transmit Video Format 0

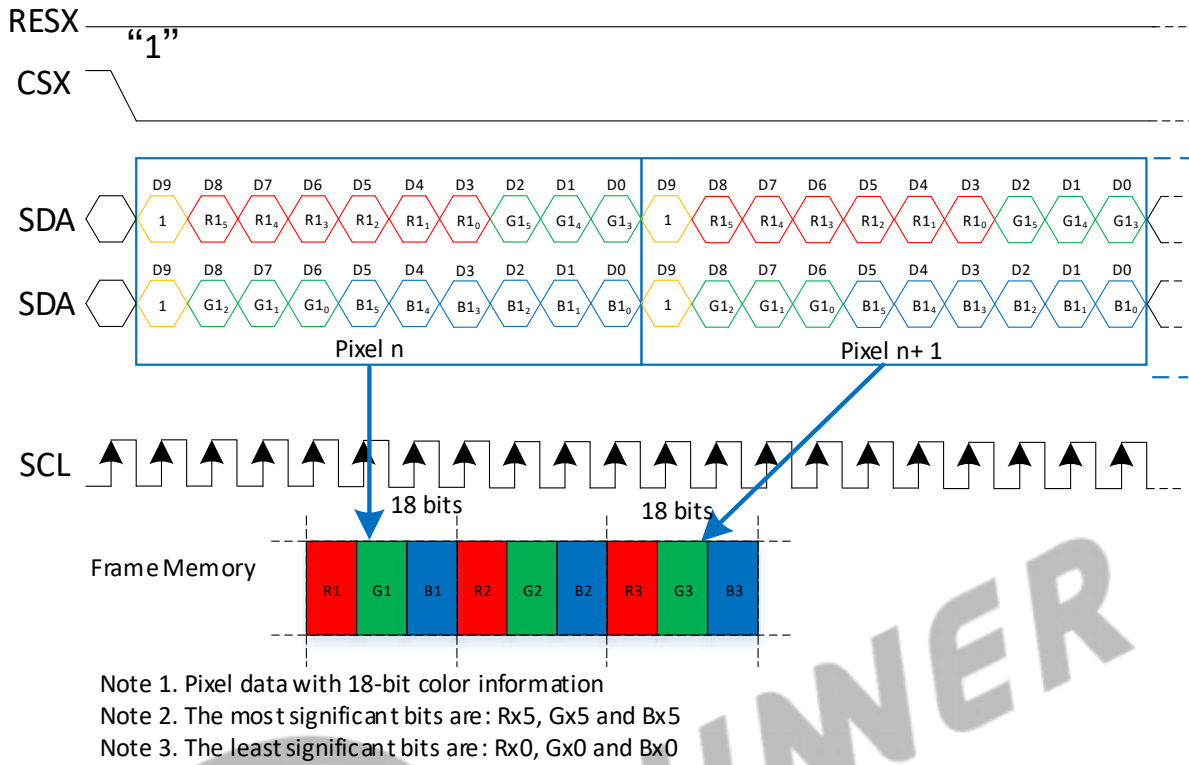


Figure 9-58 RGB666 2 Data Lane Interface Transmit Video Format 1 (ilitek)

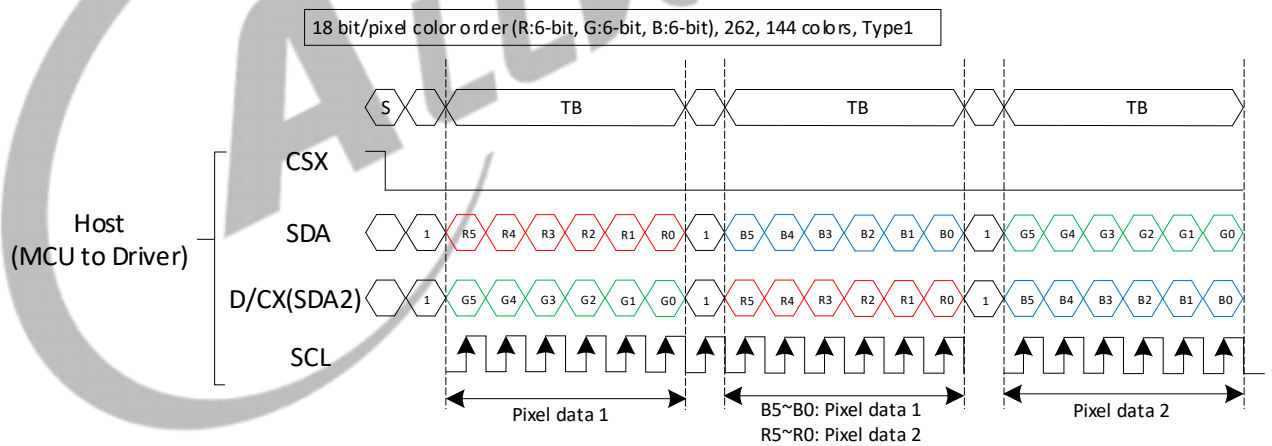


Figure 9-59 RGB666 2 Data Lane Interface Transmit Video Format 2 (New vision)

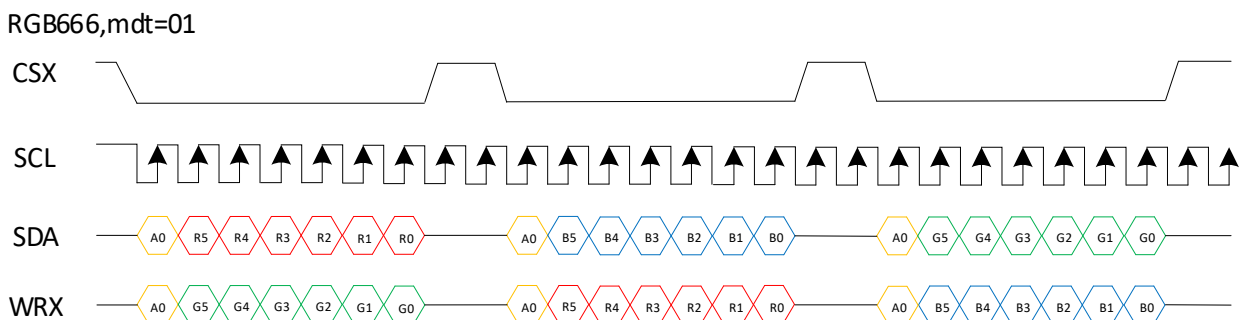
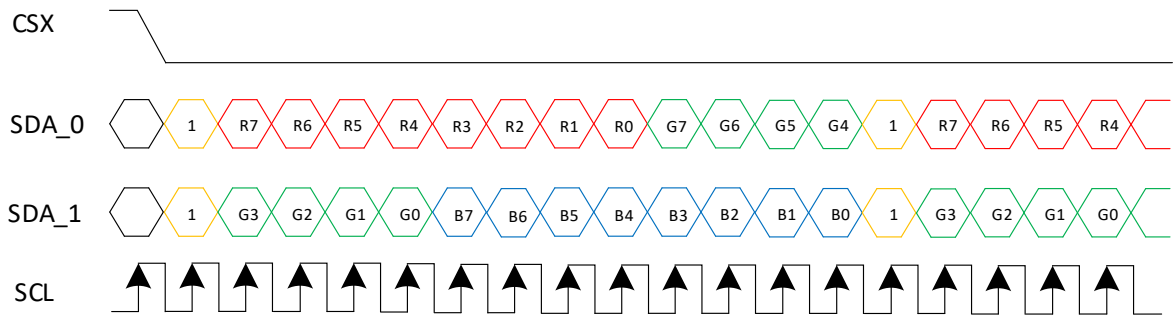


Figure 9-60 RGB888 2 Data Lane Interface Transmit Video Format

RGB888



- Note 1. Pixel data with 24-bit color information
- Note 2. The most significant t bits are: R7, G7 and B7
- Note 3. The least significant t bits are: R0, G0 and B0

### 9.4.4 Programming Guidelines

#### 9.4.4.1 Writing/Reading Data Process Using SPI Mode

The SPI transfers serial data between the processor and the external device. CPU and DMA are the two main operational modes for SPI. For each SPI, the data is simultaneously transmitted (shifted out serially) and received (shifted in serially). The SPI has 2 channels, including the TX channel and RX channel. The TX channel has the path from TX FIFO to the external device. The RX channel has the path from the external device to RX FIFO.

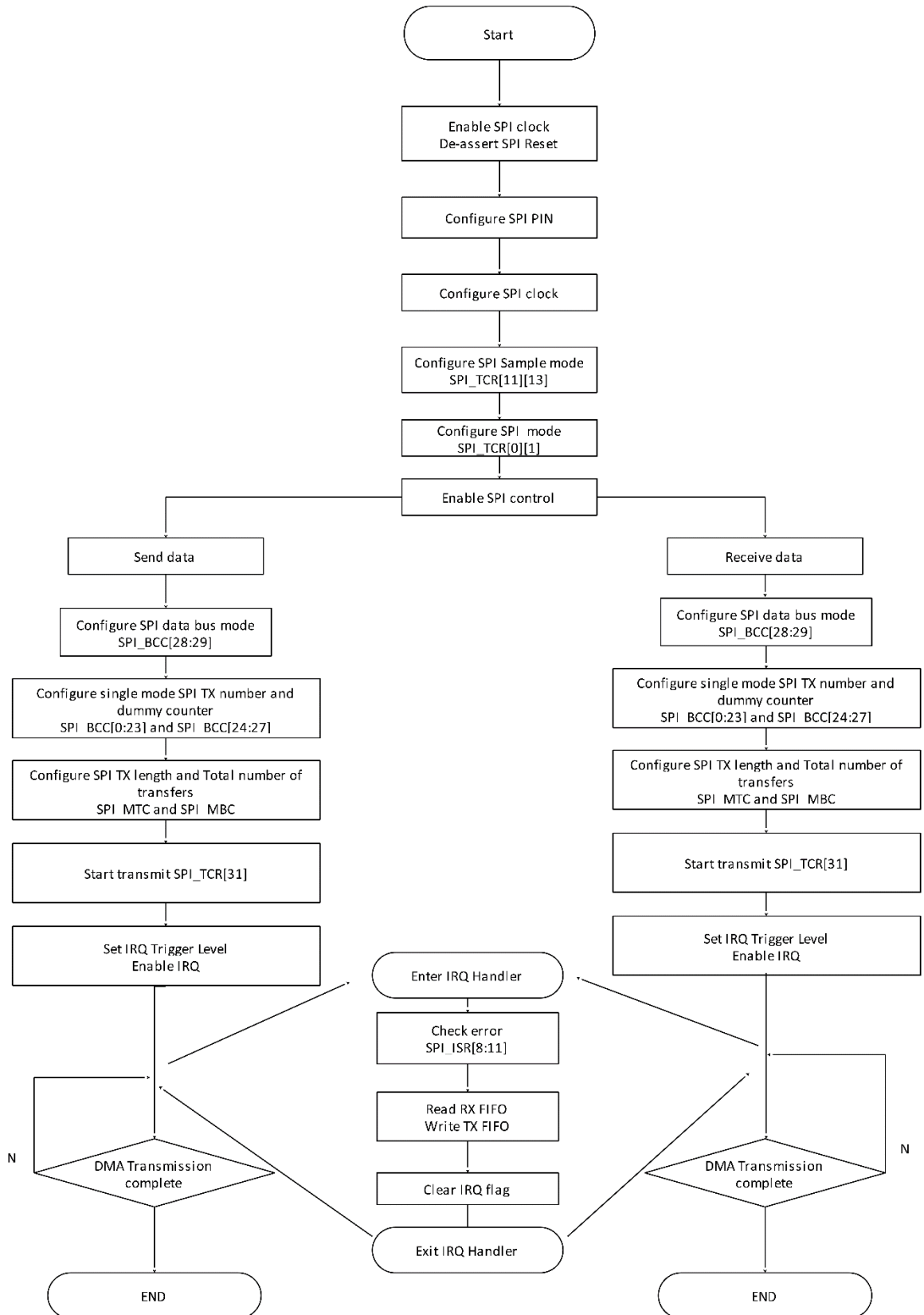
**Write Data:** CPU or DMA must write data on the [SPI\\_TXD](#) register, the data on the register are automatically moved to TX FIFO.

**Read Data:** To read data from RX FIFO, CPU or DMA must access the register [SPI\\_RXD](#) and data are automatically sent to the register [SPI\\_RXD](#).

In CPU or DMA mode, the SPI sends a completed interrupt ([SPI\\_ISR\[TC\]](#)) to the processor at the end of each transfer.

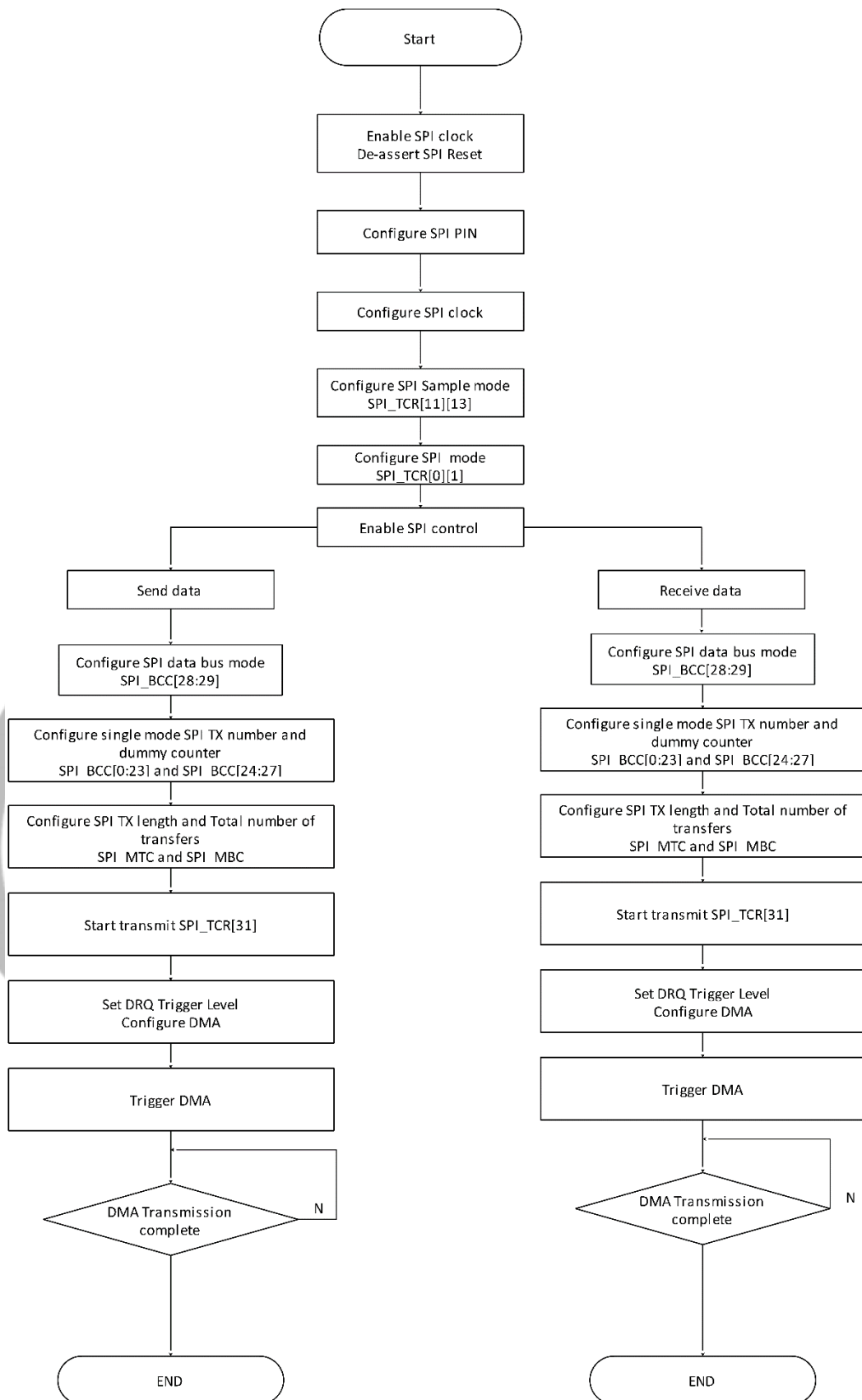
CPU Mode

Figure 9-61 SPI Write/Read Data in CPU Mode



DMA Mode

Figure 9-62 SPI Write/Read Data in DMA Mode





#### 9.4.4.2 Calibrate Delay Chain Using SPI Mode

The SPI has one delay chain which is used to generate delay to make proper timing between the internal SPI clock signal and data signals. Delay chain is made up of 64 delay cells. The delay time of one delay cell can be estimated through delay chain calibration.

The steps to calibrate delay chain are as follows:

- Step 1** Enable SPI. To calibrate the delay chain by operation registers in SPI, the SPI must be enabled through AHB reset and AHB clock gating control registers.
- Step 2** Configure a proper clock for SPI. The calibration delay chain is based on the clock for SPI from CCU.
- Step 3** Set proper initial delay value. Write 0xA0 to the [SPI Sample Delay Control Register](#) to set initial delay value 0x20 to delay chain. Then write 0x0 to the [SPI Sample Delay Control Register](#) to clear this value.
- Step 4** Write 0x8000 to the [SPI Sample Delay Control Register](#) to start to calibrate the delay chain.
- Step 5** Wait until the flag (Bit14 in the [SPI Sample Delay Control Register](#)) of calibration done is set. The number of delay cells is shown at Bit[13:8] in [SPI Sample Delay Control Register](#). The delay time generated by these delay cells is equal to the cycle of SPI's clock nearly. This value is the result of calibration.
- Step 6** Calculate the delay time of one delay cell according to the cycle of the SPI clock and the result of calibration.

#### 9.4.4.3 Transmitting Write Command Using DBI Mode

- Step 1** Set the [SPI DBI MODE SEL](#) (bit3) of [SPI\\_GCR](#) (0x0004) to 1 to select DBI mode.
- Step 2** Set the [DBI EN MODE SEL](#) (bit[30:29]) of [DBI\\_CTL\\_1](#) (0x0104) to 0 to select the trigger mode of DBI.
- Step 3** Configure the [DBI\\_CTL\\_0](#) (0x0100).
  - Set [DBI\\_CTL\\_0](#)[Command Type] (bit31) to 0 to configure the writing command.
  - Set [DBI\\_CTL\\_0](#)[Write Command Dummy Cycles] (bit[30:20]) to configure the number of dummy cycles between commands.
  - Set [DBI\\_CTL\\_0](#)[Output Data Sequence] (bit19) to select the MSB or LSB.
  - Set [DBI\\_CTL\\_0](#)[Transmit Mode] (bit15) to 0 to select the command path.
  - Set [DBI\\_CTL\\_0](#)[Output Data Format] (bit[14:12]) to 0 to transmit the command.
  - Set [DBI\\_CTL\\_0](#)[DBI interface Select] (bit[10:8]) to select the DBI interface type.
  - The remaining values of the [DBI\\_CTL\\_0](#) register remain the default value.

- Step 4** Set **DBI\_CTL\_1**[DCX\_DATA] (bit22) to 0 to send the command.
- Step 5** DMA Path: Configure the **SPI\_FCR** register (0x0018).
- Set **SPI\_FCR**[TF\_DRQ\_EN] (bit24) to 1 to enable TXFIFO DMA.
  - Set **SPI\_FCR**[TX\_TRIG\_LEVEL] (bit[23:16]) to 255. It indicates the controller requests data from DMA if the remaining space of TX FIFO is greater than 255.
- CPU Path: Write the command to be sent to the 0x200 address.
- Step 6** Set **SPI\_GCR**[DBI\_EN] (bit4) to 1 to start transmitting the command.
- Step 7** Wait until the TX FIFO underrun interrupt (**SPI\_ISR**[TF\_UDF]) is 1. It indicates that the command written to the TX FIFO is transmitted completely.

#### 9.4.4.4 Transmitting Parameter Using DBI Mode

- Step 1** Set the **SPI\_DBI\_MODE\_SEL** (bit3) of **SPI\_GCR** (0x0004) to 1 to select DBI mode.
- Step 2** Set the **DBI\_EN\_MODE\_SEL** (bit[30:29]) of **DBI\_CTL\_1** (0x0104) to 0 to select the trigger mode of DBI.
- Step 3** Configure the **DBI\_CTL\_0** register (0x0100).
- Set **DBI\_CTL\_0**[Command Type] (bit31) to 0 to configure the writing command.
  - Set **DBI\_CTL\_0**[Write Command Dummy Cycles] (bit[30:20]) to configure the number of dummy cycles between commands.
  - Set **DBI\_CTL\_0**[Output Data Sequence] (bit19) to select the MSB or LSB.
  - Set **DBI\_CTL\_0**[Transmit Mode] (bit15) to 0 to select the command path.
  - Set **DBI\_CTL\_0**[Output Data Format] (bit[14:12]) to 0 to transmit the command.
  - Set **DBI\_CTL\_0**[DBI interface Select] (bit[10:8]) to select the DBI interface type.
  - The remaining values of the **DBI\_CTL\_0** register remain the default value.
- Step 4** Set **DBI\_CTL\_1**[DCX\_DATA] (bit22) to 1 to send the parameter.
- Step 5** DMA Path: Configure the register (0x0018).
- Set **SPI\_FCR**[TF\_DRQ\_EN] (bit24) to 1 to enable TXFIFO DMA.
  - Set **SPI\_FCR**[TX\_TRIG\_LEVEL] (bit[23:16]) to 255. It indicates the controller requests data from DMA if the remaining space of TX FIFO is greater than 255.
- CPU Path: Write the command to be sent to the 0x0200 address.
- Step 6** Set **SPI\_GCR**[DBI\_EN] (bit4) to 1 to start transmitting the command.

**Step 7** Wait until the TX FIFO underrun interrupt ([SPI\\_ISR](#)[TF\_UDF]) is 1. It indicates that the command written to the TX FIFO is transmitted completely.

#### 9.4.4.5 Transmitting Video Using DBI Mode

Set the [SPI DBI MODE SEL](#) (bit3) of [SPI\\_GCR](#) (0x0004) to 1 to select DBI mode.

If the data is from the CPU path, the controller writes the command to be sent to the 0x0200 address by the AHB bus.

If the data is from the DMA path, configure [DBI\\_CTL\\_1](#)[DBI\_FIFO\_DRQ\_EN] (bit15) to 1 and [DBI\\_CTL\\_1](#)[TX\_TRIG\_LEVEL] (bit[14:8]) to 64, which indicates the controller requests data from DMA if the remaining space of TX FIFO is greater than 64.

#### Software Trigger Mode

The software enables DBI\_en\_trigger when the edge interrupt of TE is detected.

After transmitting each frame image, the controller clears automatically the line\_cnt, pixel\_cnt and stops transmitting data.

Wait for the edge interrupt of TE, the software needs to enable DBI\_en\_trigger, in circulation.

The operation process is as follows.

**Step 1** Set the [SPI DBI MODE SEL](#) (bit3) of [SPI\\_GCR](#) (0x0004) to 1 to select DBI mode.

**Step 2** Set the [DBI\\_EN\\_MODE\\_SEL](#) (bit[30:29]) of [DBI\\_CTL\\_1](#) (0x0104) to 1 to select the software trigger mode.

**Step 3** Configure the [DBI\\_CTL\\_0](#) register (0x0100).

- Set [DBI\\_CTL\\_0](#)[Command Type] (bit31) to 0 to set the writing command.
- Set [DBI\\_CTL\\_0](#)[Write Command Dummy Cycles] (bit[30:20]) to configure the number of dummy cycles between commands.
- Set [DBI\\_CTL\\_0](#)[Output Data Sequence] (bit19) to select the MSB or LSB.
- Set [DBI\\_CTL\\_0](#)[Transmit Mode] (bit15) to 1 to select the image path.
- Set [DBI\\_CTL\\_0](#)[Output Data Format] (bit[14:12]) to select RGB111//444/565/666/888.
- Set [DBI\\_CTL\\_0](#)[DBI interface Select] (bit[10:8]) to select the DBI interface type.
- The remaining values of the [DBI\\_CTL\\_0](#) register remain the default value.

**Step 4** Set [DBI\\_CTL\\_1](#)[DCX\_DATA] (bit22) to 0 to send the image data.

**Step 5** Configure [DBI\\_Video\\_Size](#) (0x110) according to the sent image size.

- Step 6** Configure [DBI\\_CTL\\_2](#) (0x0108) to set the TE-related parameter.
- Step 7** Detect the TE interrupt of the [DBI\\_INT](#) (0x0120) register.
- Step 8** Configure [DBI\\_CTL\\_1](#)[DBI\_soft\_trigger] to 1.

### Timer Trigger Mode

The software configures timer\_en to enable timer counting, and when the counter reaches the specified value, the DBI\_EN automatically can be enabled to start transmitting data.

After transmitting each frame image, the controller clears automatically the line\_cnt, pixel\_cnt, and stops transmitting data.

The timer starts counting again. When the counter reaches the specified value, the controller automatically enables DBI\_EN, and in circulation until the software turns off the timer\_en.

The operation process is as follows.

- Step 1** Set the [SPI\\_DBI\\_MODE\\_SEL](#) (bit3) of [SPI\\_GCR](#) (0x0004) to 1 to select DBI mode.
- Step 2** Set the [DBI\\_EN\\_MODE\\_SEL](#) (bit30:29) of [DBI\\_CTL\\_1](#) (0x0104) to 2 to select the timer trigger mode.
- Step 3** Configure the [DBI\\_CTL\\_0](#) register (0x0100).
  - Set [DBI\\_CTL\\_0](#)[Command Type] (bit31) to 0 to set the writing command.
  - Set [DBI\\_CTL\\_0](#)[Write Command Dummy Cycles] (bit[30:20]) to configure the number of dummy cycles between commands.
  - Set [DBI\\_CTL\\_0](#)[Output Data Sequence] (bit19) to select the MSB or LSB.
  - Set [DBI\\_CTL\\_0](#)[Transmit Mode] (bit15) to 1 to select the image path.
  - Set [DBI\\_CTL\\_0](#)[Output Data Format] (bit[14:12]) to select RGB111/444/565/666/888.
  - Set [DBI\\_CTL\\_0](#)[DBI interface Select] (bit[10:8]) to select the DBI interface type.
  - The remaining values of the [DBI\\_CTL\\_0](#) register remain the default value.
- Step 4** Set [DBI\\_CTL\\_1](#)[DCX\_DATA] (bit22) to 0 to send the image data.
- Step 5** Configure [DBI\\_Video\\_Size](#) (0x110) to transmit the image size.
- Step 6** Configure the related parameter of [DBI\\_Timer](#) (0x10C).

### TE Trigger Mode

When the edge changes of the TE are detected (The rising and falling edges are optional), the DBI\_EN automatically can be enabled to start transmitting data.

After transmitting each frame image, the controller clears automatically the line\_cnt, pixel\_cnt, and stops transmitting data.

When the edge changes of the TE are detected (The rising and falling edges are optional), the DBI\_EN automatically can be enabled to start transmitting data until the software shuts down TE\_EN or the screen no longer sends TE signals.

The operation process is as follows.

**Step 1** Set the [SPI DBI MODE SEL](#) (bit3) of [SPI\\_GCR](#) (0x0004) to 1 to select DBI mode.

**Step 2** Set the [DBI EN MODE SEL](#) (bit30:29) of [DBI\\_CTL\\_1](#) (0x0104) to 3 to select the TE trigger mode.

**Step 3** Configure the [DBI\\_CTL\\_0](#) register (0x0100).

- Set [DBI\\_CTL\\_0](#)[Command Type] (bit31) to 0 to set the writing command.
- Set [DBI\\_CTL\\_0](#)[Write Command Dummy Cycles] (bit[30:20]) to configure the number of dummy cycles between commands.
- Set [DBI\\_CTL\\_0](#)[Output Data Sequence] (bit19) to select the MSB or LSB.
- Set [DBI\\_CTL\\_0](#)[Transmit Mode] (bit15) to 1 to select the image path.
- Set [DBI\\_CTL\\_0](#)[Output Data Format] (bit[14:12]) to select RGB111/444/565/666/888.
- Set [DBI\\_CTL\\_0](#)[DBI interface Select] (bit[10:8]) to select the DBI interface type.
- The remaining values of the [DBI\\_CTL\\_0](#) register remain the default value.

**Step 4** Configure [DBI\\_CTL\\_1](#)[DCX\_DATA] (bit22) to 0 to send the image data.

**Step 5** Configure [DBI Video Size](#) (0x0110) to transmit the image size.

**Step 6** Configure [DBI\\_CTL\\_2](#) (0x0108) to set the TE-related parameter.

#### 9.4.4.6 Transmitting Read Command and Read Data Using DBI Mode

**Step 1** Set the [SPI DBI MODE SEL](#) (bit3) of [SPI\\_GCR](#) (0x0004) to 1 to select DBI mode.

**Step 2** Set the [DBI EN MODE SEL](#) (bit[30:29]) of [DBI\\_CTL\\_1](#) (0x0104) to 0.

**Step 3** Configure the [DBI\\_CTL\\_0](#) register (0x0100).

- Set [DBI\\_CTL\\_0](#)[Command Type] (bit31) to 0 to set the reading command.
- Set [DBI\\_CTL\\_0](#)[Output Data Sequence] (bit19) to select the MSB or LSB.
- Set [DBI\\_CTL\\_0](#)[Transmit Mode] (bit15) to 0 to select the command path.
- Set [DBI\\_CTL\\_0](#)[Output Data Format] (bit[14:12]) to 0.
- Set [DBI\\_CTL\\_0](#)[DBI interface Select] (bit[10:8]) to select the DBI interface type.

- The remaining values of the **DBI\_CTL\_0** register remain the default value.

**Step 4** Configure the **DBI\_CTL\_1** register (0x0104).

- Configure **DBI\_CTL\_1**[DCX\_DATA] (bit22) to 0 to send the command.
- Configure **DBI\_CTL\_1**[Read\_MSB\_First] (bit20) to select whether the first bit of the read data is the highest or lowest bit of data.
- Configure **DBI\_CTL\_1**[Read Data Number of Bytes] to set the byte number to be read.
- Configure **DBI\_CTL\_1**[Read Command Dummy Cycles] to set the dummy cycle between the read command and the read data, when the dummy cycle is complete, the data starts to be sampled.

**Step 5** DMA Path: Configure the **SPI\_FCR** register (0x0018).

- Set **SPI\_FCR**[RF\_DRQ\_EN] (bit8) to 1 to enable RXFIFO DMA.
- Set **SPI\_FCR**[RX\_TRIG\_LEVEL] (bit[7:0]) to 32, which indicates the controller requests receiving data from DMA if the data of the RX FIFO is greater than 64.

CPU Path: Read data in RX FIFO from the 0x0300 address.

**Step 6** Set **SPI\_GCR**[DBI\_EN] (bit4) to 1 to start transmitting command.

**Step 7** Wait until **DBI\_INT**[RD\_DONE\_INT] is 1. It indicates that the data is read completely.

### 9.4.5 Register List

Module Name	Base Address
SPI_DBI	0x04026000

Register Name	Offset	Description
SPI_GCR	0x0004	SPI Global Control Register
SPI_TCR	0x0008	SPI Transfer Control Register
SPI_IER	0x0010	SPI Interrupt Control Register
SPI_ISR	0x0014	SPI Interrupt Status Register
SPI_FCR	0x0018	SPI FIFO Control Register
SPI_FSR	0x001C	SPI FIFO Status Register
SPI_WCR	0x0020	SPI Wait Clock Register
SPI_SAMP_DL	0x0028	SPI Sample Delay Control Register
SPI_MBC	0x0030	SPI Master Burst Counter Register
SPI_MTC	0x0034	SPI Master Transmit Counter Register

Register Name	Offset	Description
SPI_BCC	0x0038	SPI Master Burst Control Register
SPI_BATCR	0x0040	SPI Bit-Aligned Transfer Configure Register
SPI_BA_CCR	0x0044	SPI Bit-Aligned Clock Configuration Register
SPI_TBR	0x0048	SPI TX Bit Register
SPI_RBR	0x004C	SPI RX Bit Register
SPI_NDMA_MODE_CTL	0x0088	SPI Normal DMA Mode Control Register
DBI_CTL_0	0x0100	DBI Control Register 0
DBI_CTL_1	0x0104	DBI Control Register 1
DBI_CTL_2	0x0108	DBI Control Register 2
DBI_TIMER	0x010C	DBI Timer Control Register
DBI_VIDEO_SZIE	0x0110	DBI Video Size Configuration Register
DBI_INT	0x0120	DBI Interrupt Register
DBI_DEBUG_0	0x0124	DBI BEBUG 0 Register
DBI_DEBUG_1	0x0128	DBI BEBUG 1 Register
SPI_TXD	0x0200	SPI TX Data register
SPI_RXD	0x0300	SPI RX Data register

### 9.4.6 Register Description

#### 9.4.6.1 0x0004 SPI Global Control Register (Default Value: 0x0000\_0080)

Offset:0x0004			Register Name: SPI_GCR
Bit	Read/Write	Default/Hex	Description
31	R/WAC	0x0	SRST Soft reset Writing '1' to this bit will clear the SPI controller, and auto clear to '0' when reset operation completes. Writing '0' to this bit has no effect.
30:8	/	/	/

Offset:0x0004			Register Name: SPI_GCR
Bit	Read/Write	Default/Hex	Description
7	R/W	0x1	<p>TP_EN Transmit Pause Enable In master mode, it is used to control transmit state machine to stop smart burst sending when RX FIFO is full. 0: Normal operation, ignore RXFIFO status 1: Stop transmit data when RXFIFO full Cannot be written when XCH=1.</p>
6:5	/	/	/
4	R/W	0x0	<p>DBI EN DBI Module Enable Control 0: Disable 1: Enable</p>
3	R/W	0x0	<p>SPI_DBI_MODE_SEL DBI Working Mode Select 0: SPI MODE 1: DBI MODE</p>
2	R/W	0x0	<p>MODE_SELEC Sample Timing Mode Select 0: Old mode of Sample Timing 1: New mode of Sample Timing Cannot be written when XCH=1.</p>
1	R/W	0x0	<p>MODE SPI Function Mode Select 0: Slave mode 1: Master mode Cannot be written when XCH=1.</p>
0	R/W	0x0	<p>EN SPI Module Enable Control 0: Disable 1: Enable After transforming from bit_mode to byte_mode, it must enable the SPI module again.</p>



9.4.6.2 0x0008 SPI Transfer Control Register (Default Value: 0x0000\_0087)

Offset: 0x0008			Register Name: SPI_TCR
Bit	Read/Write	Default/Hex	Description
31	R/WAC	0x0	<p>XCH Exchange Burst In master mode, it is used to start SPI burst</p> <p>0: Idle 1: Initiates exchange.</p> <p>Writing "1" to this bit will start the SPI burst, and will auto-clear after finishing the bursts transfer specified by BC. Writing "1" to SRST will also clear this bit. Writing '0' to this bit has no effect. Cannot be written when XCH=1.</p>
30:16	/	/	/
15	R/W	0x0	<p>SDC1 Master Sample Data Control register1 Set this bit to '1' to make the internal read sample point with a delay of half-cycle of SPI_CLK. It is used in high speed read operation to reduce the error caused by the time delay of SPI_CLK between master and slave.</p> <p>0: Normal operation, do not delay the internal read sample point 1: Delay the internal read sample point Cannot be written when XCH=1.</p>
14	R/W	0x0	<p>SDDM Sending Data Delay Mode 0: Normal sending 1: Delay sending Set the bit to "1" to make the data that should be sent with a delay of half-cycle of SPI_CLK in dual IO mode for SPI mode 0. Cannot be written when XCH=1.</p>
13	R/W	0x0	<p>SDM Master Sample Data Mode 0: Delay sample mode 1: Normal sample mode In normal sample mode, the SPI master samples the data at the correct edge for each SPI mode; In delay sample mode, the SPI master samples data at the edge that is half cycle delayed by the correct edge defined in respective SPI mode. Cannot be written when XCH=1.</p>

Offset: 0x0008			Register Name: SPI_TCR
Bit	Read/Write	Default/Hex	Description
12	R/W	0x0	<p>FBS</p> <p>First Transmit Bit Select</p> <p>0: MSB first. The upper bits are transmitted first.</p> <p>1: LSB first. The lower bits are transmitted first.</p> <p>Cannot be written when XCH=1.</p>
11	R/W	0x0	<p>SDC</p> <p>Master Sample Data Control</p> <p>Set this bit to '1' to make the internal read sample point with a delay of half-cycle of SPI_CLK. It is used in high speed read operation to reduce the error caused by the time delay of SPI_CLK between master and slave.</p> <p>0: Normal operation, do not delay the internal read sample point</p> <p>1: Delay the internal read sample point</p> <p>Cannot be written when XCH=1.</p>
10	R/W	0x0	<p>RPSM</p> <p>Rapids Mode Select</p> <p>Select rapid mode for high speed write.</p> <p>0: Normal write mode</p> <p>1: Rapid write mode</p> <p>Cannot be written when XCH=1.</p>
9	R/W	0x0	<p>DDB</p> <p>Dummy Burst Type</p> <p>0: The bit value of dummy SPI burst is zero</p> <p>1: The bit value of dummy SPI burst is one</p> <p>Cannot be written when XCH=1.</p>
8	R/W	0x0	<p>DHB</p> <p>Discard Hash Burst</p> <p>In master mode, it controls whether discarding unused SPI bursts</p> <p>0: Receiving all SPI bursts in the BC period</p> <p>1: Discard unused SPI bursts, only fetching the SPI bursts during the dummy burst period. The burst number is specified by TC.</p> <p>Cannot be written when XCH=1.</p>

Offset: 0x0008			Register Name: SPI_TCR
Bit	Read/Write	Default/Hex	Description
7	R/W	0x1	<p>SS_LEVEL SPI Chip Select Level</p> <p>When control SS signal manually (SS_OWNER (SPI_TCR[6])=1), set this bit to '1' or '0' to control the level of SS signal.</p> <p>0: Set SS to low 1: Set SS to high</p> <p>Cannot be written when XCH=1.</p>
6	R/W	0x0	<p>SS_OWNER SS Output Owner Select</p> <p>Usually, the controller sends the SS signal automatically with data together. When this bit is set to 1, the software must manually write SS_LEVEL (SPI_TCR[7]) to 1 or 0 to control the level of the SS signal.</p> <p>0: SPI controller 1: Software</p> <p>Cannot be written when XCH=1.</p>
5:4	R/W	0x0	<p>SS_SEL SPI Chip Select</p> <p>Select one of four external SPI Master/Slave Devices</p> <p>00: SPI_SS0 will be asserted 01: SPI_SS1 will be asserted 10: SPI_SS2 will be asserted 11: SPI_SS3 will be asserted</p> <p>Cannot be written when XCH=1.</p>
3	R/W	0x0	<p>SSCTL</p> <p>In master mode, this bit selects the output waveform for the SPI_SSx signal. Only valid when SS_OWNER (SPI_TCR[6])= 0.</p> <p>0: SPI_SSx remains asserted between SPI bursts 1: Negate SPI_SSx between SPI bursts</p> <p>Cannot be written when XCH=1.</p>
2	R/W	0x1	<p>SPOL SPI Chip Select Signal Polarity Control</p> <p>0: Active high polarity (0 = Idle) 1: Active low polarity (1 = Idle)</p> <p>Cannot be written when XCH=1.</p>

Offset: 0x0008			Register Name: SPI_TCR
Bit	Read/Write	Default/Hex	Description
1	R/W	0x1	CPOL SPI Clock Polarity Control 0: Active high polarity (0 = Idle) 1: Active low polarity (1 = Idle) Cannot be written when XCH=1.
0	R/W	0x1	CPHA SPI Clock/Data Phase Control 0: Phase 0 (Leading edge for sample data) 1: Phase 1 (Leading edge for setup data) Cannot be written when XCH=1.

9.4.6.3 0x0010 SPI Interrupt Control Register (Default Value: 0x0000\_0000)

Offset: 0x0010			Register Name: SPI_IER
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13	R/W	0x0	SS_INT_EN SSI Interrupt Enable Chip select signal (SSx) from the valid state to the invalid state 0: Disable 1: Enable
12	R/W	0x0	TC_INT_EN Transfer Completed Interrupt Enable 0: Disable 1: Enable
11	R/W	0x0	TF_UDR_INT_EN TXFIFO Underrun Interrupt Enable 0: Disable 1: Enable
10	R/W	0x0	TF_OVF_INT_EN TX FIFO Overflow Interrupt Enable 0: Disable 1: Enable

Offset: 0x0010			Register Name: SPI_IER
Bit	Read/Write	Default/Hex	Description
9	R/W	0x0	RF_UDR_INT_EN RXFIFO Underrun Interrupt Enable 0: Disable 1: Enable
8	R/W	0x0	RF_OVF_INT_EN RX FIFO Overflow Interrupt Enable 0: Disable 1: Enable
7	/	/	/
6	R/W	0x0	TF_FUL_INT_EN TX FIFO Full Interrupt Enable 0: Disable 1: Enable
5	R/W	0x0	TX_EMP_INT_EN TX FIFO Empty Interrupt Enable 0: Disable 1: Enable
4	R/W	0x0	TX_ERQ_INT_EN TX FIFO Empty Request Interrupt Enable 0: Disable 1: Enable
3	/	/	/
2	R/W	0x0	RF_FUL_INT_EN RX FIFO Full Interrupt Enable 0: Disable 1: Enable
1	R/W	0x0	RX_EMP_INT_EN RX FIFO Empty Interrupt Enable 0: Disable 1: Enable
0	R/W	0x0	RF_RDY_INT_EN RX FIFO Ready Request Interrupt Enable 0: Disable 1: Enable

9.4.6.4 0x0014 SPI Interrupt Status Register (Default Value: 0x0000\_0032)

Offset: 0x0014			Register Name: SPI_ISR
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13	R/W1C	0x0	SSI SS Invalid Interrupt When SSI is 1, it indicates that SS has changed from the valid state to the invalid state. Writing 1 to this bit clears it.
12	R/W1C	0x0	TC Transfer Completed In master mode, it indicates that all bursts specified by BC have been exchanged. In other conditions, when set, this bit indicates that all the data in TXFIFO has been loaded in the Shift register, and the Shift register has shifted out all the bits. Writing 1 to this bit clears it. 0: Busy 1: Transfer completed
11	R/W1C	0x0	TF_UDF TXFIFO Underrun This bit is set when the TXFIFO is underrun. Writing 1 to this bit clears it. 0: TXFIFO is not underrun 1: TXFIFO is underrun
10	R/W1C	0x0	TF_OVF TXFIFO Overflow This bit is set when the TXFIFO is overflowed. Writing 1 to this bit clears it. 0: TXFIFO is not overflowed 1: TXFIFO is overflowed
9	R/W1C	0x0	RX_UDF RXFIFO Underrun When set, this bit indicates that RXFIFO is underrun. Writing 1 to this bit clears it. 0: RXFIFO is not underrun 1: RXFIFO is underrun

Offset: 0x0014			Register Name: SPI_ISR
Bit	Read/Write	Default/Hex	Description
8	R/W1C	0x0	<p>RX_OVF RXFIFO Overflow</p> <p>When set, this bit indicates that RXFIFO is overflowed. Writing 1 to this bit clears it.</p> <p>0: RXFIFO is not overflowed 1: RXFIFO is overflowed</p>
7	/	/	/
6	R/W1C	0x0	<p>TX_FULL TXFIFO Full</p> <p>This bit is set when the TXFIFO is full. Writing 1 to this bit clears it.</p> <p>0: TXFIFO is not Full 1: TXFIFO is Full</p>
5	R/W1C	0x1	<p>TX_EMP TXFIFO Empty</p> <p>This bit is set when the TXFIFO is empty. Writing 1 to this bit clears it.</p> <p>0: TXFIFO contains one or more words. 1: TXFIFO is empty</p>
4	R/W1C	0x1	<p>TX_READY TXFIFO Ready</p> <p>0: TX_WL &gt; TX_TRIG_LEVEL 1: TX_WL &lt;= TX_TRIG_LEVEL</p> <p>This bit will be immediately set to 1 if TX_WL &lt;= TX_TRIG_LEVEL. Writing "1" to this bit clears it. The TX_WL is the water level of TXFIFO.</p>
3	/	/	/
2	R/W1C	0x0	<p>RX_FULL RXFIFO Full</p> <p>This bit is set when the RXFIFO is full. Writing 1 to this bit clears it.</p> <p>0: Not Full 1: Full</p>
1	R/W1C	0x1	<p>RX_EMP RXFIFO Empty</p> <p>This bit is set when the RXFIFO is empty. Writing 1 to this bit clears it.</p> <p>0: Not empty 1: empty</p>

Offset: 0x0014			Register Name: SPI_ISR
Bit	Read/Write	Default/Hex	Description
0	R/W1C	0x0	<p>RX_RDY RXFIFO Ready</p> <p>0: RX_WL &lt; RX_TRIG_LEVEL 1: RX_WL &gt;= RX_TRIG_LEVEL</p> <p>This bit will be immediately set to 1 if RX_WL &gt;= RX_TRIG_LEVEL. Writing "1" to this bit clears it. The RX_WL is the water level of RXFIFO.</p>

9.4.6.5 0x0018 SPI FIFO Control Register (Default Value: 0x0040\_0001)

Offset: 0x0018			Register Name: SPI_FCR
Bit	Read/Write	Default/Hex	Description
31	R/WAC	0x0	<p>TX_FIFO_RST TX FIFO Reset</p> <p>Writing '1' to this bit will reset the control portion of the TXFIFO and auto clear to '0' when completing the reset operation, writing '0' to this bit has no effect.</p>
30	R/W	0x0	<p>TF_TEST_ENB TX Test Mode Enable</p> <p>0: Disable 1: Enable</p> <p>In normal mode, the TXFIFO can only be read by the SPI controller, writing '1' to this bit will switch the read and write function of TXFIFO to AHB bus. This bit is used to test the TXFIFO, do not set in normal operation, and do not set RF_TEST and TF_TEST at the same time.</p>
29:25	/	/	/
24	R/W	0x0	<p>TF_DRQ_EN TX FIFO DMA Request Enable</p> <p>0: Disable 1: Enable</p>
23:16	R/W	0x40	<p>TX_TRIG_LEVEL TX FIFO Empty Request Trigger Level</p>



Offset: 0x0018			Register Name: SPI_FCR
Bit	Read/Write	Default/Hex	Description
15	R/WAC	0x0	RF_RST RXFIFO Reset Writing '1' to this bit will reset the control portion of the receiver FIFO, and auto clear to '0' when completing the reset operation, writing '0' to this bit has no effect.
14	R/W	0x0	RF_TEST RX Test Mode Enable 0: Disable 1: Enable In normal mode, the RXFIFO can only be written by the SPI controller, writing '1' to this bit will switch the read and write function of RXFIFO to AHB bus. This bit is used to test the RXFIFO, do not set in normal operation, and do not set RF_TEST and TF_TEST at the same time.
13:9	/	/	/
8	R/W	0x0	RF_DRQ_EN RXFIFO DMA Request Enable 0: Disable 1: Enable
7:0	R/W	0x1	RX_TRIG_LEVEL RXFIFO Ready Request Trigger Level

9.4.6.6 0x001C SPI FIFO Status Register (Default Value: 0x0000\_0000)

Offset: 0x001C			Register Name: SPI_FSR
Bit	Read/Write	Default/Hex	Description
31	R	0x0	TB_WR TXFIFO Write Buffer Write Enable
30:28	R	0x0	TB_CNT TXFIFO Write Buffer Counter These bits indicate the number of words in TXFIFO Write Buffer
27:24	/	/	/

Offset: 0x001C			Register Name: SPI_FSR
Bit	Read/Write	Default/Hex	Description
23:16	R	0x0	TF_CNT TXFIFO Counter These bits indicate the number of words in TXFIFO 0: 0 byte in TXFIFO 1: 1 byte in TXFIFO ... 64: 64 bytes in TXFIFO other: Reserved
15	R	0x0	RB_WR RXFIFO Read Buffer Write Enable
14:12	R	0x0	RB_CNT RXFIFO Read Buffer Counter These bits indicate the number of words in RXFIFO Read Buffer
11:8	/	/	/
7:0	R	0x0	RF_CNT RXFIFO Counter These bits indicate the number of words in RXFIFO 0: 0 byte in RXFIFO 1: 1 byte in RXFIFO ... 64: 64 bytes in RXFIFO other: Reserved

9.4.6.7 0x0020 SPI Wait Clock Register (Default Value: 0x0000\_0000)

Offset: 0x0020			Register Name: SPI_WCR
Bit	Read/Write	Default/Hex	Description
31:20	/	/	/

Offset: 0x0020			Register Name: SPI_WCR
Bit	Read/Write	Default/Hex	Description
19:16	R/W	0x0	<p>SWC</p> <p>Dual mode direction switch wait clock counter (for master mode only).</p> <p>These bits control the number of wait states to be inserted before starting dual data transfer in dual SPI mode. The SPI module counts SPI_SCLK by SWC for delaying the next word data transfer.</p> <p>0: No wait states inserted n: n SPI_SCLK wait states inserted</p> <p>Cannot be written when XCH=1.</p>
15:0	R/W	0x0	<p>WCC</p> <p>Wait Clock Counter (In master mode)</p> <p>These bits control the number of wait states to be inserted in data transfers. The SPI module counts SPI_SCLK by WCC for delaying the next word data transfer.</p> <p>0: No wait states inserted n: n SPI_SCLK wait states inserted</p> <p>Cannot be written when XCH=1.</p>

9.4.6.8 0x0028 SPI Sample Delay Control Register (Default Value: 0x0000\_2000)

Offset: 0x0028			Register Name: SPI_SAMP_DL
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15	R/W	0x0	<p>SAMP_DL_CAL_START</p> <p>Sample Delay Calibration Start</p> <p>When set, it indicates that start sample delay chain calibration.</p> <p>Cannot be written when XCH=1.</p>
14	R	0x0	<p>SAMP_DL_CAL_DONE</p> <p>Sample Delay Calibration Done</p> <p>When set, it indicates that the sample delay chain calibration is done and the result of calibration is shown in SAMP_DL.</p> <p>Cannot be written when XCH=1.</p>

Offset: 0x0028			Register Name: SPI_SAMP_DL
Bit	Read/Write	Default/Hex	Description
13:8	R	0x20	<p>SAMP_DL</p> <p>Sample Delay</p> <p>It indicates the number of delay cells corresponding to the current card clock. The delay time generated by these delay cells is equal to the cycle of the card clock nearly.</p> <p>Generally, it is necessary to do drive delay calibration when the card clock is changed.</p> <p>This bit is valid only when SAMP_DL_CAL_DONE is set.</p> <p>Cannot be written when XCH=1.</p>
7	R/W	0x0	<p>SAMP_DL_SW_EN</p> <p>Sample Delay Software Enable</p> <p>When set, it indicates that enable sample delay specified at SAMP_DL_SW.</p> <p>Cannot be written when XCH=1.</p>
6	/	/	/
5:0	R/W	0x0	<p>SAMP_DL_SW</p> <p>Sample Delay Software</p> <p>The relative delay between the clock line and command line, data lines.</p> <p>It can be determined according to the value of SAMP_DL, the cycle of the card clock, and the input timing requirement of the device.</p> <p>Cannot be written when XCH=1.</p>

**9.4.6.9 0x0030 SPI Master Burst Counter Register (Default Value: 0x0000\_0000)**

Offset: 0x0030			Register Name: SPI_MBC
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/

Offset: 0x0030			Register Name: SPI_MBC
Bit	Read/Write	Default/Hex	Description
23:0	R/W	0x0	<p>MBC Master Burst Counter</p> <p>In master mode, this field specifies the total burst number which includes the TXD, RXD, and dummy burst.</p> <p>0: 0 burst 1: 1 burst ... N: N bursts</p> <p>Cannot be written when XCH=1.</p>

9.4.6.10 0x0034 SPI Master Transmit Counter Register (Default Value: 0x0000\_0000)

Offset: 0x0034			Register Name: SPI_MTC
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:0	R/W	0x0	<p>MWTC Master Write Transmit Counter</p> <p>In master mode, this field specifies the burst number that should be sent to TXFIFO before automatically sending dummy bursts. For saving bus bandwidth, the dummy bursts (all zero bits or all one bits) are sent by SPI Controller automatically.</p> <p>0: 0 burst 1: 1 burst ... N: N bursts</p> <p>Cannot be written when XCH=1.</p>

9.4.6.11 0x0038 SPI Master Burst Control Counter Register (Default Value: 0x0000\_0000)

Offset: 0x0038			Register Name: SPI_BCC
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/

Offset: 0x0038			Register Name: SPI_BCC
Bit	Read/Write	Default/Hex	Description
29	R/W	0x0	<p>Quad_EN Quad_Mode_EN</p> <p>The Quad mode includes Quad-Input and Quad-Output.</p> <p>0: Quad mode disable 1: Quad mode enable</p> <p>Cannot be written when XCH=1.</p>
28	R/W	0x0	<p>DRM Master Dual Mode RX Enable</p> <p>It is only valid when Quad_Mode_EN=0.</p> <p>0: RX uses the single-bit mode 1: RX uses the dual mode</p> <p>Cannot be written when XCH=1.</p>
27:24	R/W	0x0	<p>DBC Master Dummy Burst Counter</p> <p>In master mode, this field specifies the burst number that should be sent before receiving in dual SPI mode. The data does not care by the device.</p> <p>0: 0 burst 1: 1 burst ... N: N bursts</p> <p>Cannot be written when XCH=1</p>
23:0	R/W	0x0	<p>STC Master Single Mode Transmit Counter</p> <p>In master mode, this field specifies the burst number that should be sent in the single mode before automatically sending dummy bursts. This is the first transmit counter in all bursts.</p> <p>0: 0 burst 1: 1 burst ... N: N bursts</p> <p>Cannot be written when XCH=1</p>

9.4.6.12 0x0040 SPI Bit-Aligned Transfer Configure Register (Default Value: 0x0000\_00A0)

Offset: 0x0040			Register Name: SPI_BATC
Bit	Read/Write	Default/Hex	Description
31	R/WAC	0x0	<p>TCE</p> <p>Transfer Control Enable</p> <p>In master mode, it is used to start to transfer the serial bits frame, it is only valid when Work Mode Select==0x10/0x11.</p> <p>0: Idle</p> <p>1: Initiates transfer</p> <p>Writing “1” to this bit will start to transfer serial bits frame (the value comes from the SPI TX Bit Register or SPI RX Bit Register), and will auto-clear after the bursts transfer completely. Writing ‘0’ to this bit has no effect.</p>
30	R/W	0x0	<p>MSMS</p> <p>Master Sample Standard</p> <p>0: Delay Sample Mode</p> <p>1: Standard Sample Mode</p> <p>In Standard Sample Mode, the SPI master samples the data at the standard rising edge of SCLK for each SPI mode;</p> <p>In Delay Sample Mode, the SPI master samples data at the edge that is half cycle delayed by the standard rising edge of SCLK defined in respective SPI mode.</p>
29:26	/	/	/
25	R/W1C	0x0	<p>TBC</p> <p>Transfer Bits Completed</p> <p>When set, this bit indicates that the last bit of the serial data frame in SPI TX Bit Register (or SPI RX Bit Register) has been transferred completely. Writing 1 to this bit clears it.</p> <p>0: Busy</p> <p>1: Transfer Completed</p> <p>It is only valid when Work Mode Select==0x10/0x11.</p>
24	R/W	0x0	<p>TBC_INT_EN</p> <p>Transfer Bits Completed Interrupt Enable</p> <p>0: Disable</p> <p>1: Enable</p> <p>It is only valid when Work Mode Select==0x10/0x11.</p>
23:22	/	/	/

Offset: 0x0040			Register Name: SPI_BATC
Bit	Read/Write	Default/Hex	Description
21:16	R/W	0x00	<p>RX_FEM_LEN</p> <p>Configure the length of serial data frame (burst) of RX</p> <p>000000: 0 bit</p> <p>000001: 1 bit</p> <p>...</p> <p>100000: 32 bits</p> <p>Other values: reserved</p> <p>It is only valid when Work Mode Select==0x10/0x11, and cannot be written when TCE (SPI_BATC[31])=1.</p>
15:14	/	/	/
13:8	R/W	0x00	<p>TX_FEM_LEN</p> <p>Configure the length of serial data frame (burst) of TX</p> <p>000000: 0 bit</p> <p>000001: 1 bit</p> <p>...</p> <p>100000: 32 bits</p> <p>Other values: reserved</p> <p>It is only valid when Work Mode Select==0x10/0x11, and cannot be written when TCE=1.</p>
7	R/W	0x1	<p>SS_LEVEL</p> <p>When control SS signal manually, set this bit to '1' or '0' to control the level of SS signal.</p> <p>0: Set SS to low</p> <p>1: Set SS to high</p> <p>It is only valid when Work Mode Select==0x10/0x11, and only work in Mode0, cannot be written when TCE=1.</p>
6	R/W	0x0	<p>SS_OWNER</p> <p>SS Output Owner Select</p> <p>Usually, the controller sends the SS signal automatically with data together. When this bit is set to 1, the software must manually write SS_LEVEL (SPI_BATC[7]) to 1 or 0 to control the level of the SS signal.</p> <p>0: SPI controller</p> <p>1: Software</p> <p>It is only valid when Work Mode Select==0x10/0x11, and only work in Mode0, cannot be written when TCE=1.</p>



Offset: 0x0040			Register Name: SPI_BATC
Bit	Read/Write	Default/Hex	Description
5	R/W	0x1	<p>SPOL</p> <p>SPI Chip Select Signal Polarity Control</p> <p>0: Active high polarity (0 = Idle)</p> <p>1: Active low polarity (1 = Idle)</p> <p>It is only valid when Work Mode Select==0x10/0x11, and only work in Mode0, cannot be written when TCE=1.</p>
4	/	/	/
3:2	R/W	0x0	<p>SS_SEL</p> <p>SPI Chip Select</p> <p>Select one of four external SPI Master/Slave Devices</p> <p>00: SPI_SS0 will be asserted</p> <p>01: SPI_SS1 will be asserted</p> <p>10: SPI_SS2 will be asserted</p> <p>11: SPI_SS3 will be asserted</p> <p>It is only valid when Work Mode Select==0x10/0x11, and only work in Mode0, cannot be written when TCE=1.</p>
1:0	R/W	0x0	<p>WMS</p> <p>Work Mode Select</p> <p>00: Data frame is byte aligned in standard SPI, dual-output/dual input SPI, dual IO SPI, and quad-output/quad-input SPI</p> <p>01: Reserved</p> <p>10: Data frame is bit aligned in 3-wire SPI</p> <p>11: Data frame is bit aligned in standard SPI</p>

**9.4.6.13 0x0044 SPI Bit-Aligned Clock Configuration Register (Default Value: 0x0000\_0000)**

Offset: 0x0044			Register Name: SPI_BA_CCR
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0x0	<p>CDR_N</p> <p>Clock Divide Rate (Master Mode Only)</p> <p>The SPI_SCLK is determined according to the following equation:  <math>SPI\_CLK = Source\_CLK / (2 * (CDR\_N + 1))</math>.</p> <p>This register is only valid when Work Mode Select==0x10/0x11.</p>

9.4.6.14 0x0048 SPI TX Bit Register (Default Value: 0x0000\_0000)

Offset: 0x0048			Register Name: SPI_TBR
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	<p>VTB</p> <p>The Value of the Transmit Bits</p> <p>This register is used to store the value of the transmitted serial data frame.</p> <p>In the process of transmission, the LSB is transmitted first.</p> <p>This register is only valid when Work Mode Select==0x10/0x11.</p>

9.4.6.15 0x004C SPI RX Bit Register (Default Value: 0x0000\_0000)

Offset: 0x004C			Register Name: SPI_RBR
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	<p>VRB</p> <p>The Value of the Receive Bits</p> <p>This register is used to store the value of the received serial data frame.</p> <p>In the process of transmission, the LSB is transmitted first.</p> <p>This register is only valid when Work Mode Select==0x10/0x11.</p>

9.4.6.16 0x0088 SPI Normal DMA Mode Control Register (Default Value: 0x0000\_00E5)

Offset: 0x0088			Register Name: SPI_NDMA_MODE_CTL
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:6	R/W	0x11	<p>SPI_ACT_M</p> <p>SPI NDMA Active Mode</p> <p>00: dma_active is low</p> <p>01: dma_active is high</p> <p>10: dma_active is controlled by dma_request (DRQ)</p> <p>11: dma_active is controlled by controller</p>
5	R/W	0x1	<p>SPI_ACK_M</p> <p>SPI NDMA Acknowledge Mode</p> <p>0: active fall do not care ack</p> <p>1: active fall must after detect ack is high</p>

Offset: 0x0088			Register Name: SPI_NDMA_MODE_CTL
Bit	Read/Write	Default/Hex	Description
4:0	R/W	0x05	SPI_DMA_WAIT Delay Cycles The counts of hold cycles from DMA last signal high to dma_active high

9.4.6.17 0x0100 DBI Control Register 0 (Default Value: 0x0010\_0000)

Offset: 0x0100			Register Name: DBI_CTL_0
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	CMDT Command Type 0: Write Command 1: Read Command
30:20	R/W	0x1	WCDC Write Command Dummy Cycles Controls dummy cycles between two write commands Range 1~255 Default Condition: there is a dbi_clk cycle between each command or parameter.
19	R/W	0x0	DAT_SEQ Output Data Sequence 0: MSB First 1: LSB First
18:16	R/W	0x0	RGB_SEQ Output RGB Sequence 000: RGB 001: RBG 010: GRB 011: GBR 100: BRG 101: BGR 110, 111: Reserved

Offset: 0x0100			Register Name: DBI_CTL_0
Bit	Read/Write	Default/Hex	Description
15	R/W	0x0	TRAN_MOD Transmit Mode 0: Command/Parameter 1: Video
14:12	R/W	0x0	DAT_FMT Output Data Format 000: RGB111 001: RGB444 010: RGB565 011: RGB666 100: RGB888 (only for 2 Data Lane Interface) 101~111: Reserved
11	/	/	/
10:8	R/W	0x0	DBI Interface 000: 3 Line Interface I 001: 3 Line Interface II 010: 4 Line Interface I 011: 4 Line Interface II 100: 2 Data Lane Interface

Offset: 0x0100			Register Name: DBI_CTL_0
Bit	Read/Write	Default/Hex	Description
7:4	R/W	0x0	<p>RGB_Source_Format</p> <p>When video_source_type is RGB32 (DBI_CTL_0[bit0] = 0)</p> <p>0000: RGB            0001: RBG            0010: GRB            0011: GBR            0100: BRG            0101: BGR            Others: Reserved</p> <p>When video_source_type is RGB16 (DBI_CTL_0[bit0] = 1)</p> <p>0000: RGB            0001~0100: Reserved            0101: BGR            0110: GRBG_0 {G[5:3]R[4:0]B[4:0]G[2:0]}            0111: GBRG_0 {G[5:3]B[4:0]R[4:0]G[2:0]}            1000: GRBG_1 {G[2:0]R[4:0]B[4:0]G[5:3]}            1001: GBRG_1 {G[2:0]B[4:0]R[4:0]G[5:3]}            Others: Reserved</p>
3	R/W	0x0	<p>DUM_VAL</p> <p>Dummy Cycle Value</p> <p>Output Value During Dummy Cycle</p>
2	R/W	0x0	<p>RGB_BO</p> <p>RGB Bit Order</p> <p>0: Remain the sequence of RGB data            1: Swap the higher bit and the lower bit for each component of DRAM RGB</p>
1	R/W	0x0	<p>ELEMENT_A_POS</p> <p>Element A Position</p> <p>Only for RGB32 Data Format</p> <p>0: A component is in the bit[31:24] of data source            1: A component is in the bit[7:0] of data source</p>
0	R/W	0x0	<p>VI_SRC_TYPE</p> <p>Video Source Type</p> <p>0: RGB32            1: RGB16</p>

9.4.6.18 0x0104 DBI Control Register 1 (Default Value: 0x0000\_0001)

Offset: 0x0104			Register Name: DBI_CTL_1
Bit	Read/Write	Default/Hex	Description
31	R/WAC	0x0	<p>DBI_SOFT_TRG</p> <p>DBI soft trigger</p> <p>It is only available for software trigger mode. Writing '1' to this bit will start DBI TX module and auto clear to '0' when completing start operation, writing '0' to this bit has no effect.</p>
30:29	R/W	0x0	<p>DBI EN MODE SEL</p> <p>DBI Enable Mode Select</p> <p>00: Always on DBI mode</p> <p>01: Software trigger mode</p> <p>10: Timer trigger mode</p> <p>11: TE trigger mode</p>
28	/	/	/
27:26	R/W	0x0	<p>RGB666_FMT</p> <p>2 Data Lane RGB666 Format</p> <p>00: Normal Format</p> <p>01: Special Format for ILITEK</p> <p>10: Special Format for New Vision</p>
25	R/W	0x0	<p>DBI_RXCLK_INV</p> <p>DBI rx clock inverse</p> <p>0: Sample data by using the positive edge of the output clock</p> <p>1: Sample data by using the negative edge of the output clock</p>
24	R/W	0x0	<p>DBI_CLKO_MOD</p> <p>DBI output clock mode</p> <p>0: DBI clock always on (DCX Setup/hold equals one clock cycle)</p> <p>1: DBI clock auto gating (DCX Setup/hold equals to a half clock cycle)</p>

Offset: 0x0104			Register Name: DBI_CTL_1
Bit	Read/Write	Default/Hex	Description
23	R/W	0x0	<p>DBI_CLKO_INV DBI clock output inverse</p> <p><b>When the bit24 (DBI output clock mode) is 0.</b> 0: The falling edge releases the CSX signal, and the falling edge releases data 1: The rising edge releases the CSX signal, and the rising edge releases data</p> <p><b>When the bit24 (DBI output clock mode) is 1.</b> 0: The rising edge releases the CSX signal, and the falling edge releases data 1: The falling edge releases the CSX signal, and the rising edge releases data</p>
22	R/W	0x0	<p>DCX_DATA DCX Data Value</p> <p>0: DCX Value equal to 0 1: DCX Value equal to 1</p>
21	R/W	0x0	<p>RGB 16 Data Source Select RGB 16 Data Source Select</p> <p>0: Pixel1 is stored in the higher bit of address, and Pixel0 is stored in the lower bit of address 1: Pixel0 is stored in the higher bit of address, and Pixel1 is stored in the lower bit of address</p>
20	R/W	0x0	<p>RDAT_LSB Bit Order of Read Data</p> <p>0: A reading data is the higher bit 1: A reading data is the lower bit</p>
19:16	/	/	/
15:8	R/W	0x0	<p>RCDC Read Command Dummy Cycles</p> <p>The dummy cycle between the read command and read data Reading 1-byte (8 bits) data has not dummy cycle.</p>
7:0	R/W	0x1	<p>RDBN Read Data Number of Bytes</p> <p>Sample Bytes data based on configuration.</p>

9.4.6.19 0x0108 DBI Control Register 2 (Default Value: 0x0000\_4000)

Offset: 0x0108			Register Name: DBI_CTL_2
Bit	Read/Write	Default/Hex	Description
31:26	/	/	/
15	R/W	0x0	DBI_FIFO_DRQ_EN DBI FIFO DMA Request Enable 0: Disable 1: Enable
14:8	R/W	0x40	DBI_TRIG_LEVEL DBI FIFO Empty Request Trigger Level
7	/	/	/
6	R/W	0x0	DBI_SDI_OUT_SEL DBI SDI PIN Output Select The signal is used with the DBI SDI PIN Function Sel bit. 0: Output WRX (When DBI DCX PIN Function Sel = 0, the SDI pin outputs data) 1: Output DCX
5	R/W	0x0	DBI_DCX_SEL DBI DCX PIN Function Select 0: DBI DCX Function 1: WRX (2 Data Lane Interface)
4:3	R/W	0x0	DBI_SDI_SEL DBI SDI PIN Function Select 00: DBI_SDI (Interface II) 01: DBI_TE 10: DBI_DCX 11: Reserved
2	R/W	0x0	TE_DBC_SEL TE debounce function select 0: debounce 1: no-debounce
1	R/W	0x0	TE_TRIG_SEL TE edge trigger select 0: TE rising edge 1: TE falling edge
0	R/W	0x0	TE_EN TE enable 0: TE Disable 1: TE Enable



9.4.6.20 0x010C DBI Timer Control Register (Default Value: 0x0000\_0000)

Offset: 0x010C			Register Name: DBI_Timer
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	DBI_TM_EN DBI Timer Enable 0: Enable 1: Disable
30:0	R/W	0x0	DBI Timer Value It sets the time interval between sending data twice, which is frame blanking. It is used to set the time at which the interrupt of the DBI Timer is triggered. When the Timer_EN is 1, the timer starts to count (the clock of the counting is SCLK), and the counter reaches the target value to trigger the Timer_INT of DBI, the data will start to send in series. <b>Note: Do not count when sending the series data.</b>

9.4.6.21 0x0110 DBI Video Size Register (Default Value: 0x01E0\_0140)

Offset: 0x0110			Register Name: DBI_Video_Size
Bit	Read/Write	Default/Hex	Description
31:27	/	/	/
26:16	R/W	0x1E0	V_SIZE It is used to generate the Frame int.
15:11	/	/	/
10:0	R/W	0x140	H_SIZE It is used to generate the Line int.

9.4.6.22 0x0120 DBI Interrupt Register (Default Value: 0x0000\_4000)

Offset: 0x0120			Register Name: DBI_INT
Bit	Read/Write	Default/Hex	Description
31:15	/	/	/

Offset: 0x0120			Register Name: DBI_INT
Bit	Read/Write	Default/Hex	Description
14	R/W1C	0x1	DBI_FIFO_EMPTY_INT DBI FIFO Empty Interrupt Status 0: DBI_FIFO is not empty 1: DBI_FIFO is empty
13	R/W1C	0x0	DBI_FIFO_FULL_INT DBI FIFO Full Interrupt Status 0: DBI_FIFO is not full 1: DBI_FIFO is full
12	R/W1C	0x0	TIMER_INT It indicates that the timer has been count sclk cycles to the value of DBI_Timer Register[30:0]. Writing 1 to this bit clears it. 0: Timer has not been achieved the objective 1: Timer has been achieved the objective
11	R/W1C	0x0	RD_DONE_INT It indicates that the number of byte setting in DBI_Control Register 1[19:8] has been read. Writing 1 to this bit clears it. 0: All data has not been read 1: All data has been read
10	R/W1C	0x0	TE_INT It indicates that the TE signal has been changed. Writing 1 to this bit clears it. 0: TE signal has not been changed 1: TE signal has been changed
9	R/W1C	0x0	FRAM_DONE_INT It indicates that a frame video data has been sent. Writing 1 to this bit clears it. 0: A frame video has not been sent 1: A frame video has been sent
8	R/W1C	0x0	LINE_DONE_INT It indicates that a line of video data has been sent. Writing 1 to this bit clears it. 0: A line of video data has not been sent 1: A line of video data has been sent
7	/	/	/

Offset: 0x0120			Register Name: DBI_INT
Bit	Read/Write	Default/Hex	Description
6	R/W	0x0	DBI_FIFO_EMPTY_INT_EN DBI FIFO Empty Interrupt Enable 0: Disable 1: Enable
5	R/W	0x0	DBI_FIFO_FULL_INT_EN DBI FIFO Full Interrupt Enable 0: Disable 1: Enable
4	R/W	0x0	TIMER_INT_EN Timer Interrupt Enable 0: Disable 1: Enable
3	R/W	0x0	RD_DONE_INT_EN Read Done Interrupt Enable 0: Disable 1: Enable
2	R/W	0x0	TE_INT_EN TE Interrupt Enable 0: Disable 1: Enable
1	R/W	0x0	FRAM_DONE_INT_EN Frame Done Interrupt Enable 0: Disable 1: Enable
0	R/W	0x0	LINE_DONE_INT_EN Line Done Interrupt Enable 0: Disable 1: Enable

**9.4.6.23 0x0124 DBI Debug Register 0 (Default Value: 0x007F\_0000)**

Offset: 0x0124			Register Name: DBI_Debug_0
Bit	Read/Write	Default/Hex	Description
31:23	/	/	/

Offset: 0x0124			Register Name: DBI_Debug_0
Bit	Read/Write	Default/Hex	Description
22:16	R	0x7F	DBI_FIFO_AVAIL DBI_FIFO ROOM VALID 0~127 Words
15:13	/	/	/
12	R	0x0	TE_VAL TE input value 0: TE not Trigger 1: TE Trigger
11:8	R	0x0	DBI_RXCS FSM for DBI Receive RX_BS0 ~ RX_BS6 , Gray - Code
7:4	R	0x0	SH_CS FSM for shifter 0~11 : SH0~SH11
3:2	/	/	DBI_TXCS FSM for DBI Transmit 00: IDLE 01: SHIF 10: DUMY 11: READ
1:0	R	0x0	MEM_CS FSM for DBI Memory 00: IDLE_FRM 01: FRM_POS 10: FRM_RDY

9.4.6.24 0x0128 DBI Debug Register 1 (Default Value: 0x0000\_0000)

Offset: 0x0128			Register Name: DBI_Debug_1
Bit	Read/Write	Default/Hex	Description
31:26	/	/	/
25:16	R	0x0	LCNT Line counter The number of pixel lines that are currently sent
15:12	/	/	/

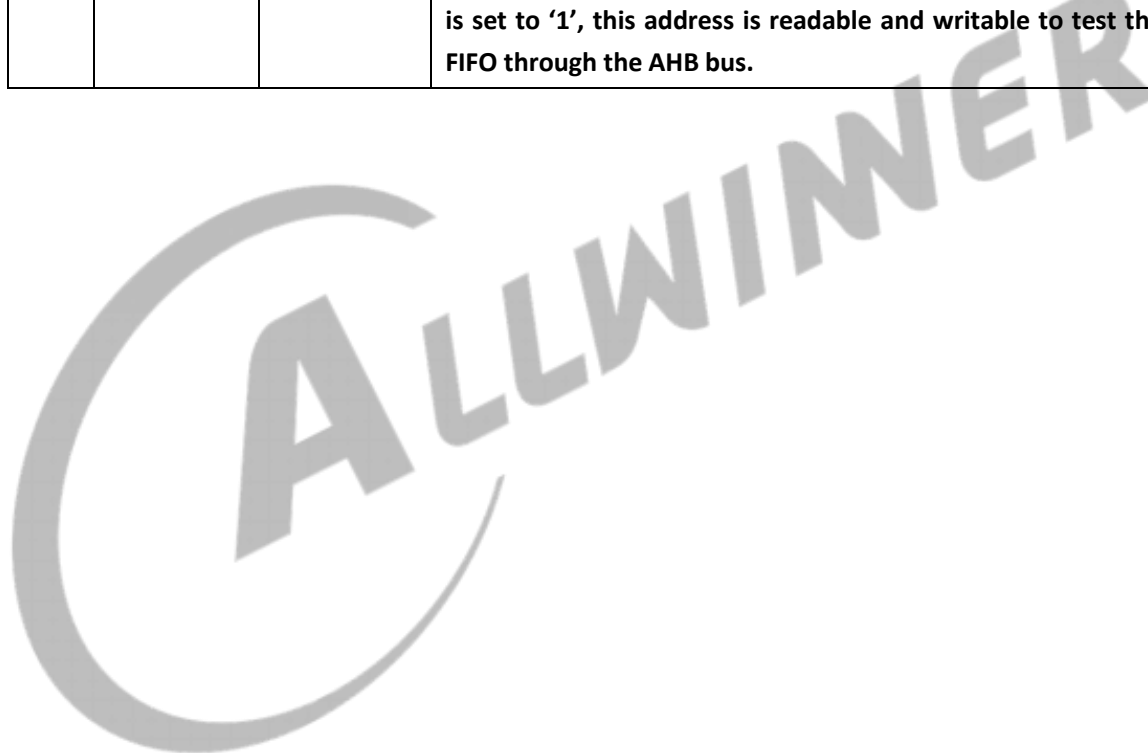
Offset: 0x0128			Register Name: DBI_Debug_1
Bit	Read/Write	Default/Hex	Description
11:0	R	0x0	CCNT Component counter The number of RGB components that are currently sent The field is equal to pixel_cnt *3.

9.4.6.25 0x0200 SPI TX Data Register (Default Value: 0x0000\_0000)

Offset: 0x0200			Register Name: SPI_TXD
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	TDATA Transmit Data This register can be accessed in the byte, half-word, or word unit by AHB. In the byte accessing method, if there are rooms in TXFIFO, one burst data is written to TXFIFO and the depth is increased by 1. In the half-word accessing method, two SPI burst data are written and the TXFIFO depth is increased by 2. In the word accessing method, four SPI burst data are written and the TXFIFO depth is increased by 4.  <b>Note: This address is writable-only if TF_TEST is '0', and if TF_TEST is set to '1', this address is readable and writable to test the TX FIFO through the AHB bus.</b>

9.4.6.26 0x0300 SPI RX Data Register (Default Value: 0x0000\_0000)

Offset: 0x0300			Register Name: SPI_RXD
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	<p>RDATA Receive Data</p> <p>This register can be accessed in the byte, half-word, or word unit by AHB. In the byte accessing method, if there are data in RXFIFO, the top word is returned and the RXFIFO depth is decreased by 1. In the half-word accessing method, two SPI bursts are returned and the RXFIFO depth is decreased by 2. In the word accessing method, the four SPI bursts are returned and the RXFIFO depth is decreased by 4.</p> <p><b>Note:</b> This address is readable-only if RF_TEST is '0', and if RF_TEST is set to '1', this address is readable and writable to test the RX FIFO through the AHB bus.</p>



## 9.5 USB2.0 DRD

### 9.5.1 Overview

The USB2.0 dual-role device (USB2.0 DRD) supports both device and host functions which can also be configured as a Host-only or Device-only controller. It complies with the USB2.0 Specification.

For saving CPU bandwidth, the DMA interface of the DRD module can also support the external DMA controller to do the data transfer between the memory and the DRD FIFO. The DRD core also supports USB power saving functions.

The USB2.0 DRD has the following features:

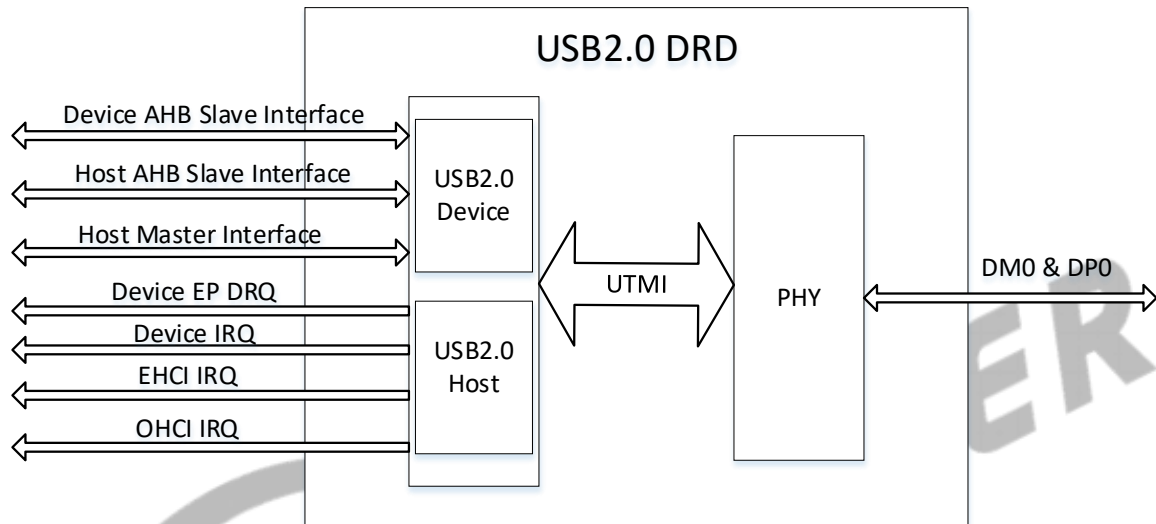
- Complies with USB2.0 Specification
- Supports USB Host function
  - Compatible with Enhanced Host Controller Interface (EHCI) Specification, Version 1.0
  - Compatible with Open Host Controller Interface (OHCI) Specification, Version 1.0a
  - Supports High-Speed (HS, 480 Mbit/s), Full-Speed (FS, 12 Mbit/s), and Low-Speed (LS, 1.5 Mbit/s)
  - Supports only 1 USB Root port shared between EHCI and OHCI
- Supports USB Device function
  - Supports High-Speed (HS, 480 Mbit/s), Full-Speed (FS, 12 Mbit/s)
  - Supports bi-directional endpoint0 (EP0) for Control transfer
  - Up to 10 user-configurable endpoints (EP1+, EP1-, EP2+, EP2-, EP3+, EP3-, EP4+, EP4-, EP5+, EP5-) for Bulk transfer, Isochronous transfer and Interrupt transfer
  - Up to (8 KB + 64 Bytes) FIFO for all EPs (including EP0)
  - Supports interface to an external Normal DMA controller for every EP
- Supports an internal DMA controller for data transfer with memory
- Supports High-Bandwidth Isochronous & Interrupt transfers
- Automated splitting/combining of packets for Bulk transfers
- Supports point-to-point and point-to-multipoint transfer in both Host and Peripheral modes
- Includes automatic ping capabilities
- Soft connect/disconnect function
- Performs all transaction scheduling in hardware
- Power optimization and power management capabilities

- Device and host controller share a 8K SRAM and a physical PHY

### 9.5.2 Block Diagram

The following figure shows the block diagram of USB2.0 DRD Controller.

Figure 9-63 USB2.0 DRD Controller Block Diagram



### 9.5.3 Functional Description

#### 9.5.3.1 External Signals

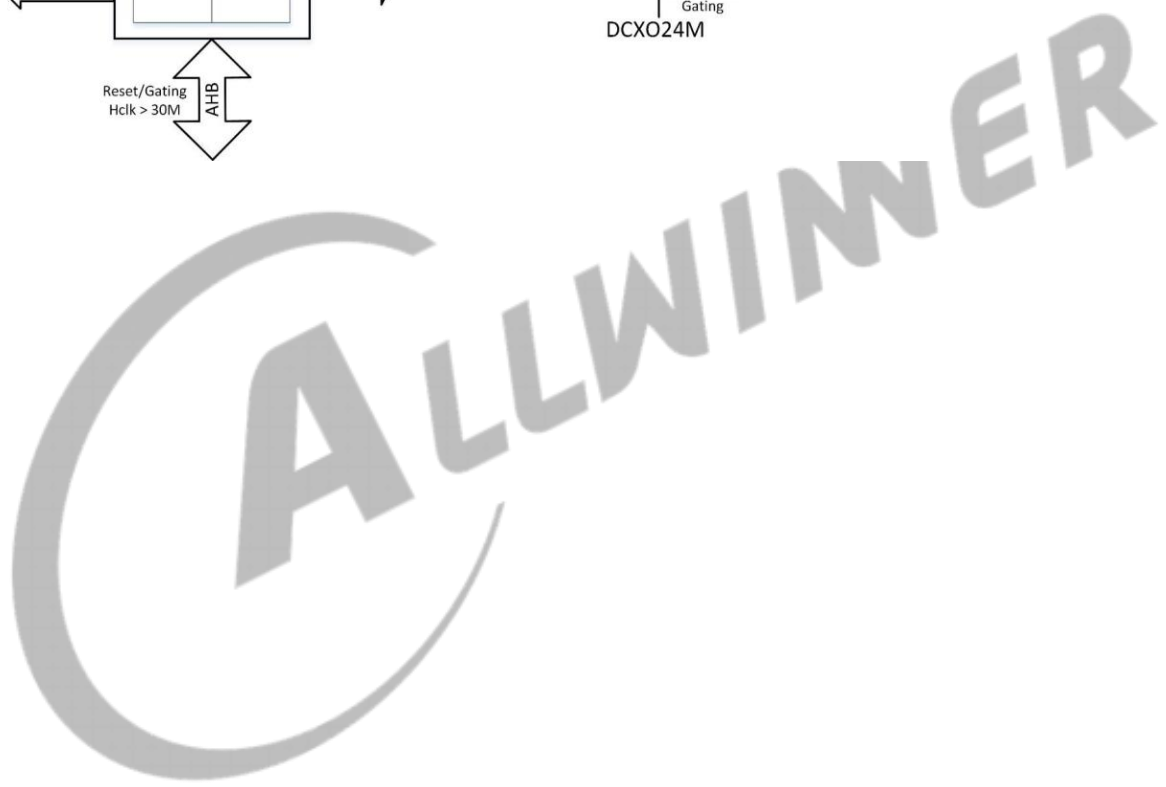
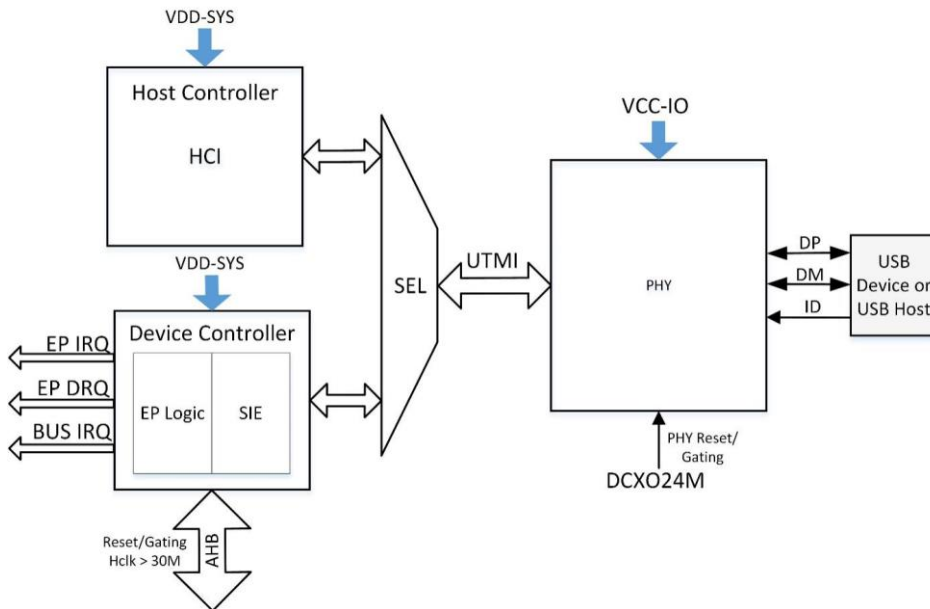
Table 9-18 USB2.0 DRD External Signals

Signal	Description	Type
USB0-DP	USB2.0 DRD differential signal positive	AI/O
USB0-DM	USB2.0 DRD differential signal negative	AI/O



9.5.3.2 Controller and PHY Connection Diagram

Figure 9-64 USB2.0 DRD Controller and PHY Connection Diagram



## 9.6 USB2.0 HOST

### 9.6.1 Overview

The USB Host Controller is fully compliant with USB 2.0 Specification, Enhanced Host Controller Interface (EHCI) Specification Revision 1.0 and Open Host Controller Interface (OHCI) Specification Release 1.0a.

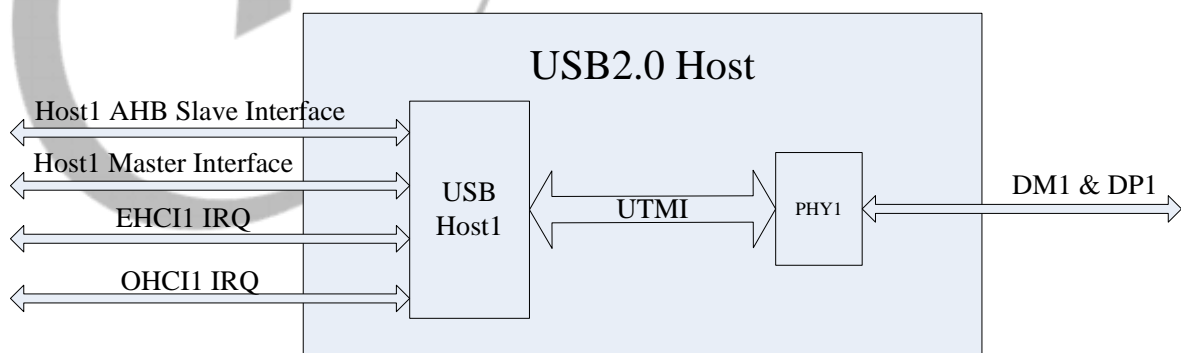
The USB2.0 host controller includes the following features:

- Complies with USB2.0 Specification
- Supports USB2.0 Host function
  - Compatible with Enhanced Host Controller Interface (EHCI) Specification, Version 1.0
  - Compatible with Open Host Controller Interface (OHCI) Specification, Version 1.0a
  - Supports High-Speed (HS, 480 Mbit/s), Full-Speed (FS, 12 Mbit/s) and Low-Speed (LS, 1.5 Mbit/s) Device
  - Supports only 1 USB Root port shared between EHCI and OHCI
- An internal DMA Controller for data transfer with memory

### 9.6.2 Block Diagram

The following figure shows the block diagram of USB2.0 Host Controller.

Figure 9-65 USB2.0 Host Controller Block Diagram



### 9.6.3 Functional Description

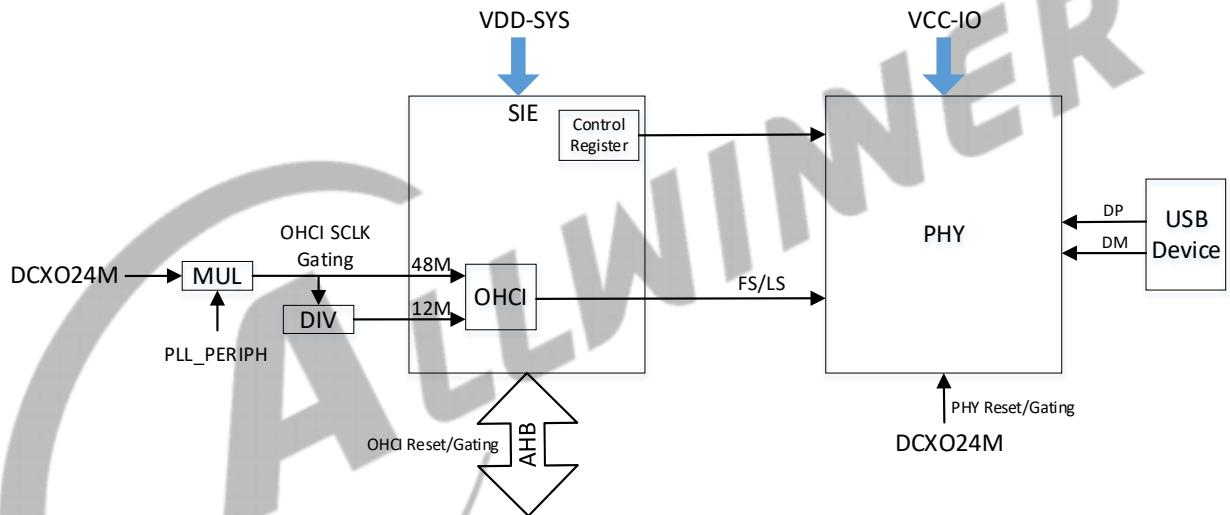
#### 9.6.3.1 External Signals

Table 9-19 USB2.0 Host External Signals

Signal	Description	Type
USB1-DP	USB2.0 Host differential signal positive	AI/O
USB1-DM	USB2.0 Host differential signal negative	AI/O

#### 9.6.3.2 Controller and PHY Connection Diagram

Figure 9-66 USB2.0 Host Controller and PHY Connection Diagram



#### 9.6.4 Register List

Module Name	Base Address
USB1	0x04200000

Register Name	Offset	Description
<b>EHCI Capability Register</b>		
E_CAPLENGTH	0x0000	EHCI Capability Register Length Register
E_HCVERSION	0x0002	EHCI Host Interface Version Number Register
E_HCSPARAMS	0x0004	EHCI Host Control Structural Parameter Register
E_HCCPARAMS	0x0008	EHCI Host Control Capability Parameter Register

Register Name	Offset	Description
E_HCSPORTRROUTE	0x000C	EHCI Companion Port Route Description
<b>EHCI Operational Register</b>		
E_USBCMD	0x0010	EHCI USB Command Register
E_USBSTS	0x0014	EHCI USB Status Register
E_USBINTR	0x0018	EHCI USB Interrupt Enable Register
E_FRINDEX	0x001C	EHCI USB Frame Index Register
E_CTRLDSSEGMENT	0x0020	EHCI 4G Segment Selector Register
E_PERIODICLISTBASE	0x0024	EHCI Frame List Base Address Register
E_ASYNC_LISTADDR	0x0028	EHCI Next Asynchronous List Address Register
E_CONFIGFLAG	0x0050	EHCI Configured Flag Register
E_PORTSC	0x0054	EHCI Port Status/Control Register
<b>OHCI Control and Status Partition Register</b>		
O_HcControl	0x0404	OHCI Control Register
O_HcCommandStatus	0x0408	OHCI Command Status Register
O_HcInterruptStatus	0x040C	OHCI Interrupt Status Register
O_HcInterruptEnable	0x0410	OHCI Interrupt Enable Register
O_HcInterruptDisable	0x0414	OHCI Interrupt Disable Register
<b>OHCI Memory Pointer Partition Register</b>		
O_HcHCCA	0x0418	OHCI HCCA Base
O_HcPeriodCurrentED	0x041C	OHCI Period Current ED Base
O_HcControlHeadED	0x0420	OHCI Control Head ED Base
O_HcControlCurrentED	0x0424	OHCI Control Current ED Base
O_HcBulkHeadED	0x0428	OHCI Bulk Head ED Base
O_HcBulkCurrentED	0x042C	OHCI Bulk Current ED Base
O_HcDoneHead	0x0430	OHCI Done Head Base
<b>OHCI Frame Counter Partition Register</b>		
O_HcFmInterval	0x0434	OHCI Frame Interval Register
O_HcFmRemaining	0x0438	OHCI Frame Remaining Register
O_HcFmNumber	0x043C	OHCI Frame Number Register
O_HcPeriodicStart	0x0440	OHCI Periodic Start Register
O_HcLSThreshold	0x0444	OHCI LS Threshold Register
<b>OHCI Root Hub Partition Register</b>		
O_HcRhDescriptorA	0x0448	OHCI Root Hub Descriptor Register A
O_HcRhDescriptorB	0x044C	OHCI Root Hub Descriptor Register B

Register Name	Offset	Description
O_HcRhStatus	0x0450	OHCI Root Hub Status Register
O_HcRhPortStatus	0x0454	OHCI Root Hub Port Status Register
<b>HCI Controller and PHY Interface Register</b>		
HCI_Interface	0x0800	HCI Interface Register
HCI_CTRL3	0x0808	HCI Control Register
PHY_Control	0x0810	PHY Control Register
PHY_STATUS	0x0824	PHY Status Register
HCI SIE Port Disable Control	0x0828	HCI SIE Port Disable Control Register

## 9.6.5 EHCI Register Description

### 9.6.5.1 0x0000 EHCI Identification Register (Default Value:0x10)

Offset:0x0000			Register Name: CAPLENGTH
Bit	Read/Write	Default/Hex	Description
7:0	R	0x10	CAPLENGTH The value in these bits indicates an offset to add to register base to find the beginning of the Operational Register Space.

### 9.6.5.2 0x0002 EHCI Host Interface Version Number Register (Default Value:0x0100)

Offset: 0x0002			Register Name: HCIVERSION
Bit	Read/Write	Default/Hex	Description
15:0	R	0x0100	HCIVERSION This is a 16-bit register containing a BCD encoding of the EHCI revision number supported by this host controller. The most significant byte of this register represents a major revision and the least significant byte is the minor revision.

### 9.6.5.3 0x0004 EHCI Host Control Structural Parameter Register (Default Value:0x0000\_1101)

Offset: 0x0004			Register Name: HCSPARAMS
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:20	R	0x0	Debug Port Number

Offset: 0x0004			Register Name: HCSPARAMS						
Bit	Read/Write	Default/Hex	Description						
			<p>This register identifies which of the host controller ports is the debug port. The value is the port number (one based) of the debug port.</p> <p>This field will always be '0'.</p>						
19:16	/	/	/						
15:12	R	0x1	<p>Number of Companion Controller (N_CC)</p> <p>This field indicates the number of companion controllers associated with this USB2.0 host controller. A zero in this field indicates there are no companion host controllers. And a value larger than zero in this field indicates there are companion USB1.1 host controller(s).</p> <p>This field will always be '0'.</p>						
11:8	R	0x1	<p>Number of Port per Companion Controller (N_PCC)</p> <p>This field indicates the number of ports supported per companion host controller host controller. It is used to indicate the port routing configuration to system software.</p> <p>This field will always fix with '0'.</p>						
7	R	0x0	<p>Port Routing Rules</p> <p>This field indicates the method used by this implementation for how all ports are mapped to companion controllers. The value of this field has the following interpretation:</p> <table border="1" data-bbox="699 1243 1428 1556"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>The first N_PCC ports are routed to the lowest numbered function companion host controller, the next N_PCC port are routed to the next lowest function companion controller, and so on.</td> </tr> <tr> <td>1</td> <td>The port routing is explicitly enumerated by the first N_PORTS elements of the HCSP-PORTTOUTE array.</td> </tr> </tbody> </table> <p>This field will always be '0'.</p>	Value	Meaning	0	The first N_PCC ports are routed to the lowest numbered function companion host controller, the next N_PCC port are routed to the next lowest function companion controller, and so on.	1	The port routing is explicitly enumerated by the first N_PORTS elements of the HCSP-PORTTOUTE array.
Value	Meaning								
0	The first N_PCC ports are routed to the lowest numbered function companion host controller, the next N_PCC port are routed to the next lowest function companion controller, and so on.								
1	The port routing is explicitly enumerated by the first N_PORTS elements of the HCSP-PORTTOUTE array.								
6:4	/	/	/						
3:0	R	0x1	<p>N_PORTS</p> <p>This field specifies the number of physical downstream ports implemented on this host controller. The value of this field determines how many port registers are addressable in the Operational Register Space. Valid values are in the range of 0x1 to 0x0f.</p> <p>This field is always 1.</p>						

9.6.5.4 0x0008 EHCI Host Control Capability Parameter Register (Default Value:0x0000\_A026)

Offset: 0x0008			Register Name: HCCPARAMS
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:8	R	0xA0	<p><b>EHCI Extended Capabilities Pointer (EECP)</b></p> <p>This optional field indicates the existence of a capabilities list. A value of 00b indicates no extended capabilities are implemented. A non-zero value in this register indicates the offset in PCI configuration space of the first EHCI extended capability. The pointer value must be 40h or greater if implemented to maintain consistency of the PCI header defined for this class of device.</p> <p>The value of this field is always '00b'.</p>
7:4	R	0x2	<p><b>Isochronous Scheduling Threshold</b></p> <p>This field indicates, relative to the current position of the executing host controller, where software can reliably update the isochronous schedule.</p> <p>When bit[7] is zero, the value of the least significant 3 bits indicates the number of micro-frames a host controller can hold a set of isochronous data structures (one or more) before flushing the state. When bit[7] is a one, then host software assumes the host controller may cache an isochronous data structure for an entire frame.</p>
3	/	/	/
2	R	0x1	<p><b>Asynchronous Schedule Park Capability</b></p> <p>If this bit is set to a one, then the host controller supports the park feature for high-speed queue heads in the Asynchronous Schedule. The feature can be disabled or enabled and set to a specific level by using the Asynchronous Schedule Park Mode Enable and Asynchronous Schedule Park Mode Count fields in the USBCMD register.</p>
1	R	0x1	<p><b>Programmable Frame List Flag</b></p> <p>If this bit is set to a zero, then system software must use a frame list length of 1024 elements with this host controller. The USBCMD register Frame List Size field is a read-only register and should be set to zero.</p> <p>If set to 1, then system software can specify and use the frame list in the USBCMD register Frame List Size field to configure the host controller.</p> <p>The frame list must always align on a 4K page boundary. This requirement ensures that the frame list is always physically contiguous.</p>
0	/	/	/

9.6.5.5 0x000C EHCI Companion Port Route Description (Default Value:0x0000\_0000)

Offset: 0x000C			Register Name: HCSP-PORTROUTE
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	<p>HCSP-PORTROUTE</p> <p>This optional field is valid only if Port Routing Rules field in HCSPARAMS register is set to a one.</p> <p>This field is used to allow a host controller implementation to explicitly describe to which companion host controller each implemented port is mapped. This field is a 15-element nibble array (each 4 bit is one array element). Each array location corresponds one-to-one with a physical port provided by the host controller (e.g. PORTROUTE [0] corresponds to the first PORTSC port, PORTROUTE [1] to the second PORTSC port, etc.). The value of each element indicates to which the companion host controllers this port is routed. Only the first N_PORTS elements have valid information. A value of zero indicates that the port is routed to the lowest numbered function companion host controller. A value of one indicates that the port is routed to the next lowest numbered function companion host controller, and so on.</p>

9.6.5.6 0x0010 EHCI USB Command Register (Default Value:0x0008\_0B00)

Offset: 0x0010			Register Name: USBCMD																		
Bit	Read/Write	Default/Hex	Description																		
31:24	/	/	/																		
23:16	R/W	0x08	<p>Interrupt Threshold Control</p> <p>The value in this field is used by system software to select the maximum rate at which the host controller will issue interrupts. The only valid values are defined below:</p> <table border="1" data-bbox="671 1659 1426 2094"> <thead> <tr> <th>Value</th> <th>Minimum Interrupt Interval</th> </tr> </thead> <tbody> <tr> <td>0x00</td> <td>Reserved</td> </tr> <tr> <td>0x01</td> <td>1 micro-frame</td> </tr> <tr> <td>0x02</td> <td>2 micro-frame</td> </tr> <tr> <td>0x04</td> <td>4 micro-frame</td> </tr> <tr> <td>0x08</td> <td>8 micro-frame(default, equates to 1 ms)</td> </tr> <tr> <td>0x10</td> <td>16 micro-frame(2ms)</td> </tr> <tr> <td>0x20</td> <td>32 micro-frame(4ms)</td> </tr> <tr> <td>0x40</td> <td>64 micro-frame(8ms)</td> </tr> </tbody> </table>	Value	Minimum Interrupt Interval	0x00	Reserved	0x01	1 micro-frame	0x02	2 micro-frame	0x04	4 micro-frame	0x08	8 micro-frame(default, equates to 1 ms)	0x10	16 micro-frame(2ms)	0x20	32 micro-frame(4ms)	0x40	64 micro-frame(8ms)
Value	Minimum Interrupt Interval																				
0x00	Reserved																				
0x01	1 micro-frame																				
0x02	2 micro-frame																				
0x04	4 micro-frame																				
0x08	8 micro-frame(default, equates to 1 ms)																				
0x10	16 micro-frame(2ms)																				
0x20	32 micro-frame(4ms)																				
0x40	64 micro-frame(8ms)																				



Offset: 0x0010			Register Name: USBCMD
Bit	Read/Write	Default/Hex	Description
			Any other value in this register yields undefined results. The default value in this field is 0x08 . Software modifications to this bit while HC Halted bit is equal to zero results in undefined behavior.
15:12	/	/	/
11	R	0x1	Asynchronous Schedule Park Mode Enable (OPTIONAL) If the Asynchronous Park Capability bit in the HCCPARAMS register is a one, then this bit defaults to a 1 and is R/W. Otherwise the bit must be a zero and is Read Only. Software uses this bit to enable or disable Park mode. When this bit is one, Park mode is enabled. When this bit is zero, Park mode is disabled.
10	/	/	/
9:8	R	0x3	Asynchronous Schedule Park Mode Count (OPTIONAL) Asynchronous Park Capability bit in the HCCPARAMS register is a one, then this field defaults to 0x3 and is W/R. Otherwise it defaults to zero and is R. It contains a count of the number of successive transactions the host controller is allowed to execute from a high-speed queue head on the Asynchronous schedule before continuing traversal of the Asynchronous schedule. Valid value are 0x1 to 0x3. Software must not write a zero to this bit when Park Mode Enable is a one as it will result in undefined behavior.
7	R/W	0x0	Light Host Controller Reset (OPTIONAL) This control bit is not required. If implemented, it allows the driver to reset the EHCI controller without affecting the state of the ports or relationship to the companion host controllers. For example, the PORSTC registers should not be reset to their default values and the CF bit setting should not go to zero (retaining port ownership relationships). A host software read of this bit as zero indicates the Light Host Controller Reset has completed and it is safe for software to re-initialize the host controller. A host software read of this bit as a one indicates the Light Host
6	R/W	0x0	Interrupt on Async Advance Doorbell This bit is used as a doorbell by software to tell the host controller to issue an interrupt the next time it advances asynchronous schedule. Software must write a 1 to this bit to ring the doorbell. When the host controller has evicted all appropriate cached schedule state, it sets the Interrupt on Async Advance status bit in the USBSTS. If the Interrupt on Async Advance Enable bit in the

Offset: 0x0010			Register Name: USBCMD										
Bit	Read/Write	Default/Hex	Description										
			<p>USBINTR register is a one then the host controller will assert an interrupt at the next interrupt threshold.</p> <p>The host controller sets this bit to a zero after it has set the Interrupt on Async Advance status bit in the USBSTS register to a one.</p> <p>Software should not write a one to this bit when the asynchronous schedule is disabled. Doing so will yield undefined results.</p>										
5	R/W	0x0	<p>Asynchronous Schedule Enable</p> <p>This bit controls whether the host controller skips processing the Asynchronous Schedule. Values mean:</p> <table border="1"> <thead> <tr> <th>Bit Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Do not process the Asynchronous Schedule.</td> </tr> <tr> <td>1</td> <td>Use the ASYNLISTADDR register to access the Asynchronous Schedule.</td> </tr> </tbody> </table> <p>The default value of this field is '0b'.</p>	Bit Value	Meaning	0	Do not process the Asynchronous Schedule.	1	Use the ASYNLISTADDR register to access the Asynchronous Schedule.				
Bit Value	Meaning												
0	Do not process the Asynchronous Schedule.												
1	Use the ASYNLISTADDR register to access the Asynchronous Schedule.												
4	R/W	0x0	<p>Periodic Schedule Enable</p> <p>This bit controls whether the host controller skips processing the Periodic Schedule. Values mean:</p> <table border="1"> <thead> <tr> <th>Bit Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Do not process the Periodic Schedule.</td> </tr> <tr> <td>1</td> <td>Use the PERIODICLISTBASE register to access the Periodic Schedule.</td> </tr> </tbody> </table> <p>The default value of this field is '0b'.</p>	Bit Value	Meaning	0	Do not process the Periodic Schedule.	1	Use the PERIODICLISTBASE register to access the Periodic Schedule.				
Bit Value	Meaning												
0	Do not process the Periodic Schedule.												
1	Use the PERIODICLISTBASE register to access the Periodic Schedule.												
3:2	R/W	0x0	<p>Frame List Size</p> <p>This field is R/W only if Programmable Frame List Flag in the HCCPARAMS registers is set to a one. This field specifies the size of the Frame list. The size the frame list controls which bits in the Frame Index Register should be used for the Frame List Current index. Values mean:</p> <table border="1"> <thead> <tr> <th>Bits</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>00b</td> <td>1024 elements(4096bytes)Default value</td> </tr> <tr> <td>01b</td> <td>512 elements(2048byts)</td> </tr> <tr> <td>10b</td> <td>256 elements(1024bytes)For resource-constrained condition</td> </tr> <tr> <td>11b</td> <td>reserved</td> </tr> </tbody> </table> <p>The default value is '00b'.</p>	Bits	Meaning	00b	1024 elements(4096bytes)Default value	01b	512 elements(2048byts)	10b	256 elements(1024bytes)For resource-constrained condition	11b	reserved
Bits	Meaning												
00b	1024 elements(4096bytes)Default value												
01b	512 elements(2048byts)												
10b	256 elements(1024bytes)For resource-constrained condition												
11b	reserved												

Offset: 0x0010			Register Name: USBCMD
Bit	Read/Write	Default/Hex	Description
1	R/W	0x0	<p>Host Controller Reset</p> <p>This control bit is used by software to reset the host controller. The effects of this on Root Hub registers are similar to a Chip Hardware Reset.</p> <p>When software writes a one to this bit, the Host Controller resets its internal pipelines, timers, counters, state machines, etc. to their initial value. Any transaction currently in progress on USB is immediately terminated. A USB reset is not driven on downstream ports.</p> <p>All operational registers, including port registers and port state machines are set to their initial values. Port ownership reverts to the companion host controller(s). Software must reinitialize the host controller as described in Section 4.1 of the CHEI Specification in order to return the host controller to an operational state.</p> <p>This bit is set to zero by the Host Controller when the reset process is complete. Software cannot terminate the reset process early by writing a zero to this register.</p> <p>Software should not set this bit to a one when the HC Halted bit in the USBSTS register is a zero. Attempting to reset an actively running host controller will result in undefined behavior.</p>
0	R/W	0x0	<p>Run/Stop</p> <p>When set to a 1, the Host Controller proceeds with execution of the schedule. When set to 0, the Host Controller completes the current and any actively pipelined transactions on the USB and then halts. The Host Controller must halt within 16 micro-frames after software clears this bit.</p> <p>The HC Halted bit indicates when the Host Controller has finished its pending pipelined transactions and has entered the stopped state.</p> <p>Software must not write a one to this field unless the Host Controller is in the Halt State.</p> <p>The default value is 0x0.</p>

**9.6.5.7 0x0014 EHCI USB Status Register (Default Value:0x0000\_1000)**

Offset: 0x0014			Register Name: USBSTS
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15	R	0x0	Asynchronous Schedule Status

Offset: 0x0014			Register Name: USBSTS
Bit	Read/Write	Default/Hex	Description
			The bit reports the current real status of Asynchronous Schedule. If this bit is a zero then the status of the Asynchronous Schedule is disabled. If this bit is a one then the status of the Asynchronous Schedule is enabled. The Host Controller is not required to immediately disable or enable the Asynchronous Schedule when software transitions the Asynchronous Schedule Enable bit in the USBCMD register. When this bit and the Asynchronous Schedule Enable bit are the same value, the Asynchronous Schedule is either enabled (1) or disabled (0).
14	R	0x0	<p>Periodic Schedule Status</p> <p>The bit reports the current real status of the Periodic Schedule. If this bit is a zero then the status of the Periodic Schedule is disabled. If this bit is a one then the status of the Periodic Schedule is enabled. The Host Controller is not required to <i>immediately</i> disable or enable the Periodic Schedule when software transitions the <i>Periodic Schedule Enable</i> bit in the USBCMD register. When this bit and the <i>Periodic Schedule Enable</i> bit are the same value, the Periodic Schedule is either enabled (1) or disabled (0).</p>
13	R	0x0	<p>Reclamation</p> <p>This is a read-only status bit, which is used to detect an empty asynchronous schedule.</p>
12	R	0x1	<p>HC Halted</p> <p>This bit is a zero whenever the Run/Stop bit is a one. The Host Controller Sets this bit to one after it has stopped executing as a result of the Run/Stop bit being set to 0, either by software or by the Host Controller Hardware (e.g. internal error). The default value is '1'.</p>
11:6	/	/	/
5	R/WC	0x0	<p>Interrupt on Async Advance</p> <p>System software can force the host controller to issue an interrupt the next time the host controller advances the asynchronous schedule by writing a one to the Interrupt on Async Advance Doorbell bit in the USBCMD register. This status bit indicates the assertion of that interrupt source.</p>
4	R/WC	0x0	<p>Host System Error</p> <p>The Host Controller set this bit to 1 when a serious error occurs during a host system access involving the Host Controller module. When this error occurs, the Host Controller clears the Run/Stop bit in the Command register to prevent further execution of the scheduled TDs.</p>
3	R/WC	0x0	Frame List Rollover

Offset: 0x0014			Register Name: USBSTS
Bit	Read/Write	Default/Hex	Description
			The Host Controller sets this bit to a one when the Frame List Index rolls over from its maximum value to zero. The exact value at which the rollover occurs depends on the frame list size. For example, if the frame list size is 1024, the Frame Index Register rolls over every time FRINDEX [13] toggles. Similarly, if the size is 512, the Host Controller sets this bit to a one every time FRINDEX [12] toggles.
2	R/WC	0x0	<p>Port Change Detect</p> <p>The Host Controller sets this bit to a one when any port for which the Port Owner bit is set to zero has a change bit transition from a zero to a one or a Force Port Resume bit transition from a zero to a one as a result of a J-K transition detected on a suspended port. This bit will also be set as a result of the Connect Status Chang being set to a one after system software has relinquished ownership of a connected port by writing a one to a port's Port Owner bit.</p>
1	R/WC	0x0	<p>USB Error Interrupt(USBERRINT)</p> <p>The Host Controller sets this bit to 1 when completion of USB transaction results in an error condition(e.g. error counter underflow).If the TD on which the error interrupt occurred also had its IOC bit set, both.</p> <p>This bit and USBINT bit are set.</p>
0	R/WC	0x0	<p>USB Interrupt(USBINT)</p> <p>The Host Controller sets this bit to a one on the completion of a USB transaction, which results in the retirement of a Transfer Descriptor that had its IOC bit set.</p> <p>The Host Controller also sets this bit to 1 when a short packet is detected (actual number of bytes received was less than the expected number of bytes)</p>

**9.6.5.8 0x0018 EHCI USB Interrupt Enable Register (Default Value:0x0000\_0000)**

Offset: 0x0018			Register Name: USBINTR
Bit	Read/Write	Default/Hex	Description
31:6	/	/	/
5	R/W	0x0	<p>Interrupt on Async Advance Enable</p> <p>When this bit is 1, and the Interrupt on Async Advance bit in the USBSTS register is 1, the host controller will issue an interrupt at the next interrupt threshold. The interrupt is acknowledged by software clearing the Interrupt on Async Advance bit.</p>

Offset: 0x0018			Register Name: USBINTR
Bit	Read/Write	Default/Hex	Description
4	R/W	0x0	<p>Host System Error Enable</p> <p>When this bit is 1, and the Host System Error Status bit in the USBSTS register is 1, the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Host System Error bit.</p>
3	R/W	0x0	<p>Frame List Rollover Enable</p> <p>When this bit is 1, and the Frame List Rollover bit in the USBSTS register is 1, the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Frame List Rollover bit.</p>
2	R/W	0x0	<p>Port Change Interrupt Enable</p> <p>When this bit is 1, and the Port Chang Detect bit in the USBSTS register is 1, the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Port Chang Detect bit.</p>
1	R/W	0x0	<p>USB Error Interrupt Enable</p> <p>When this bit is 1, and the USBERRINT bit in the USBSTS register is 1, the host controller will issue an interrupt at the next interrupt threshold.</p> <p>The interrupt is acknowledged by software clearing the USBERRINT bit.</p>
0	R/W	0x0	<p>USB Interrupt Enable</p> <p>When this bit is 1, and the USBINT bit in the USBSTS register is 1, the host controller will issue an interrupt at the next interrupt threshold.</p> <p>The interrupt is acknowledged by software clearing the USBINT bit</p>

**9.6.5.9 0x001C EHCI Frame Index Register (Default Value:0x0000\_0000)**

Offset: 0x001C			Register Name: FRINDEX
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13:0	R/W	0	<p>Frame Index</p> <p>The value in this register increment at the end of each time frame (e.g. micro-frame). Bits[N:3] are used for the Frame List current index. It means that each location of the frame list is accessed 8 times (frames or Micro-frames) before moving to the next index. The following illustrates values of N based on the value of the Frame List Size field in the USBCMD register.</p>

Offset: 0x001C			Register Name: FRINDEX		
Bit	Read/Write	Default/Hex	Description		
			USBCMD[Frame List Size]	Number Elements	N
			00b	1024	12
			01b	512	11
			10b	256	10
			11b	Reserved	



**NOTE**

This register must be written as a DWORD. Byte writes produce undefined results.

**9.6.5.10 0x0024 EHCI Periodic Frame List Base Address Register (Default Value:0x0000\_0000)**

Offset: 0x0024			Register Name: PERIODICLISTBASE		
Bit	Read/Write	Default/Hex	Description		
31:12	R/W	0x0	Base Address These bits correspond to memory address signals [31:12], respectively. This register contains the beginning address of the Periodic Frame List in the system memory. System software loads this register prior to starting the schedule execution by the Host Controller. The memory structure referenced by this physical memory pointer is assumed to be 4 Kbyte aligned. The contents of this register are combined with the Frame Index Register (FRINDEX) to enable the Host Controller to step through the Periodic Frame List in sequence.		
11:0	/	/	/		



**NOTE**

Writes must be Dword Writes.

9.6.5.11 0x0028 EHCI Current Asynchronous List Address Register (Default Value:0x0000\_0000)

Offset: 0x0028			Register Name: ASYNCLISTADDR
Bit	Read/Write	Default/Hex	Description
31:5	R/W	0x0	<p>Link Pointer (LP)</p> <p>This field contains the address of the next asynchronous queue head to be executed.</p> <p>These bits correspond to memory address signals [31:5], respectively.</p>
4:0	/	/	/



**NOTE**

Write must be DWord Writes.

9.6.5.12 0x0050 EHCI Configure Flag Register (Default Value:0x0000\_0000)

Offset: 0x0050			Register Name: CONFIGFLAG						
Bit	Read/Write	Default/Hex	Description						
31:1	/	/	/						
0	R/W	0x0	<p>Configure Flag (CF)</p> <p>Host software sets this bit as the last action in its process of configuring the Host Controller. This bit controls the default port-routing control logic as follow:</p> <table border="1" data-bbox="662 1393 1426 1624"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Port routing control logic default-routs each port to an implementation dependent classic host controller.</td> </tr> <tr> <td>1</td> <td>Port routing control logic default-routs all ports to this host controller.</td> </tr> </tbody> </table> <p>The default value of this field is '0'.</p>	Value	Meaning	0	Port routing control logic default-routs each port to an implementation dependent classic host controller.	1	Port routing control logic default-routs all ports to this host controller.
Value	Meaning								
0	Port routing control logic default-routs each port to an implementation dependent classic host controller.								
1	Port routing control logic default-routs all ports to this host controller.								



**NOTE**

This register is not used in the normal implementation.



9.6.5.13 0x0054 EHCI Port Status and Control Register (Default Value:0x0000\_2000)

Offset: 0x0054			Register Name: PORTSC																
Bit	Read/Write	Default/Hex	Description																
31:22	/	/	/																
21	R/W	0x0	<p>Wake on Disconnect Enable (WKDSCNNT_E)</p> <p>Writing this bit to a one enables the port to be sensitive to device disconnects as wake-up events.</p> <p>This field is zero if Port Power is zero.</p> <p>The default value in this field is '0'.</p>																
20	R/W	0x0	<p>Wake on Connect Enable (WKCNNNT_E)</p> <p>Writing this bit to a one enable the port to be sensitive to device connects as wake-up events.</p> <p>This field is zero if Port Power is zero.</p> <p>The default value in this field is '0'.</p>																
19:16	R/W	0x0	<p>Port Test Control</p> <p>The value in this field specifies the test mode of the port. The encoding of the test mode bits are as follows:</p> <table border="1" data-bbox="671 1034 1426 1469"> <thead> <tr> <th>Bits</th> <th>Test Mode</th> </tr> </thead> <tbody> <tr> <td>0000b</td> <td>The port is NOT operating in a test mode.</td> </tr> <tr> <td>0001b</td> <td>Test J_STATE</td> </tr> <tr> <td>0010b</td> <td>Test K_STATE</td> </tr> <tr> <td>0011b</td> <td>Test SEO_NAK</td> </tr> <tr> <td>0100b</td> <td>Test Packet</td> </tr> <tr> <td>0101b</td> <td>Test FORCE_ENABLE</td> </tr> <tr> <td>0110b-1111b</td> <td>Reserved</td> </tr> </tbody> </table> <p>The default value in this field is '0000b'.</p>	Bits	Test Mode	0000b	The port is NOT operating in a test mode.	0001b	Test J_STATE	0010b	Test K_STATE	0011b	Test SEO_NAK	0100b	Test Packet	0101b	Test FORCE_ENABLE	0110b-1111b	Reserved
Bits	Test Mode																		
0000b	The port is NOT operating in a test mode.																		
0001b	Test J_STATE																		
0010b	Test K_STATE																		
0011b	Test SEO_NAK																		
0100b	Test Packet																		
0101b	Test FORCE_ENABLE																		
0110b-1111b	Reserved																		
15:14	/	/	/																
13	R/W	0x1	<p>Port Owner</p> <p>This bit unconditionally goes to a 0b when the Configured bit in the CONFIGFLAG register makes a 0b to 1b transition. This bit unconditionally goes to 1b whenever the Configured bit is zero.</p> <p>System software uses this field to release ownership of the port to selected host controller (in the event that the attached device is not a high-speed device).Software writes a one to this bit when the attached device is not a high-speed device. A one in this bit means that a companion host controller owns and controls the port.</p> <p>Default Value = 1b.</p>																
12	/	/	/																

Offset: 0x0054			Register Name: PORTSC															
Bit	Read/Write	Default/Hex	Description															
11:10	R	0x0	<p>Line Status</p> <p>These bits reflect the current logical levels of the D+ (bit11) and D- (bit10) signal lines. These bits are used for detection of low-speed USB devices prior to port reset and enable sequence. This read only field is valid only when the port enable bit is zero and the current connect status bit is set to a one.</p> <p>The encoding of the bits are:</p> <table border="1"> <thead> <tr> <th>Bit[11:10]</th> <th>USB State</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>00b</td> <td>SE0</td> <td>Not Low-speed device, perform EHCI reset.</td> </tr> <tr> <td>10b</td> <td>J-state</td> <td>Not Low-speed device, perform EHCI reset.</td> </tr> <tr> <td>01b</td> <td>K-state</td> <td>Low-speed device, release ownership of port.</td> </tr> <tr> <td>11b</td> <td>Undefined</td> <td>Not Low-speed device, perform EHCI reset.</td> </tr> </tbody> </table> <p>This value of this field is undefined if Port Power is zero.</p>	Bit[11:10]	USB State	Interpretation	00b	SE0	Not Low-speed device, perform EHCI reset.	10b	J-state	Not Low-speed device, perform EHCI reset.	01b	K-state	Low-speed device, release ownership of port.	11b	Undefined	Not Low-speed device, perform EHCI reset.
Bit[11:10]	USB State	Interpretation																
00b	SE0	Not Low-speed device, perform EHCI reset.																
10b	J-state	Not Low-speed device, perform EHCI reset.																
01b	K-state	Low-speed device, release ownership of port.																
11b	Undefined	Not Low-speed device, perform EHCI reset.																
9	/	/	/															
8	R/W	0x0	<p>Port Reset</p> <p>1=Port is in Reset. 0=Port is not in Reset. Default value = 0.</p> <p>When software writes a one to this bit (from a zero), the bus reset sequence as defined in the USB Specification Revision 2.0 is started. Software writes a zero to this bit to terminate the bus reset sequence. Software must keep this bit at a one long enough to ensure the reset sequence, as specified in the USB Specification Revision 2.0, completes.</p> <p><b>Note: When software writes this bit to a one , it must also write a zero to the Port Enable bit.</b></p> <p>Note that when software writes a zero to this bit there may be a delay before the bit status changes to a zero. The bit status will not read as a zero until after the reset has completed. If the port is in high-speed mode after reset is complete, the host controller will automatically enable this port (e.g. set the Port Enable bit to a one). A host controller must terminate the reset and stabilize the state of the port within 2 milliseconds of software transitioning this bit from a one to a zero. For example: if the port detects that the attached device is high-speed during reset, then the host controller must have the port in the enabled state with 2ms of software writing this bit to a zero.</p>															

Offset: 0x0054			Register Name: PORTSC												
Bit	Read/Write	Default/Hex	Description												
			<p>The HC Halted bit in the USBSTS register should be a zero before software attempts to use this bit. The host controller may hold Port Reset asserted to a one when the HC Halted bit is a one.</p> <p>This field is zero if Port Power is zero.</p>												
7	R/W	0x0	<p>Suspend</p> <p>Port Enabled Bit and Suspend bit of this register define the port states as follows:</p> <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Bits[Port Suspend]</th> <th>Enables,</th> <th>Port State</th> </tr> </thead> <tbody> <tr> <td>0x</td> <td></td> <td>Disable</td> </tr> <tr> <td>10</td> <td></td> <td>Enable</td> </tr> <tr> <td>11</td> <td></td> <td>Suspend</td> </tr> </tbody> </table> <p>When in suspend state, downstream propagation of data is blocked on this port, except for port reset. The blocking occurs at the end of the current transaction, if a transaction was in progress when this bit was written to 1. In the suspend state, the port is sensitive to resume detection. Not that the bit status does not change until the port is suspend and that there may be a delay in suspending a port if there is a transaction currently in progress on the USB.</p> <p>A write of zero to this bit is ignored by the host controller. The host controller will unconditionally set this bit to a zero when:</p> <ol style="list-style-type: none"> <li>① Software sets the Force Port Resume bit to a zero(from a one).</li> <li>② Software sets the Port Reset bit to a one(from a zero).</li> </ol> <p>If host software sets this bit to a one when the port is not enabled(i.e. Port enabled bit is a zero), the results are undefined.</p> <p>This field is zero if Port Power is zero.</p> <p>The default value in this field is '0'.</p>	Bits[Port Suspend]	Enables,	Port State	0x		Disable	10		Enable	11		Suspend
Bits[Port Suspend]	Enables,	Port State													
0x		Disable													
10		Enable													
11		Suspend													
6	R/W	0x0	<p>Force Port Resume</p> <p>1 = Resume detected/driven on port. 0 = No resume (K-state) detected/driven on port. Default value = 0.</p> <p>This functionality defined for manipulating this bit depends on the value of the Suspend bit. For example, if the port is not suspend and software transitions this bit to a one, then the effects on the bus are undefined.</p> <p>Software sets this bit to 1 to drive resume signaling. The Host Controller sets this bit to a 1 if a J-to-K transition is detected while the port is in the Suspend state. When this bit transitions to a one because a J-to-K transition is detected, the Port Change Detect bit in the USBSTS register is also set to a one. If software sets this bit</p>												

Offset: 0x0054			Register Name: PORTSC
Bit	Read/Write	Default/Hex	Description
			<p>to a one, the host controller must not set the Port Change Detect bit.</p> <p>Note that when the EHCI controller owns the port, the resume sequence follows the defined sequence documented in the USB Specification Revision 2.0. The resume signaling (Full-speed 'K') is driven on the port as long as this remains a one. Software must appropriately time the Resume and set this bit to a zero when the appropriate amount of time has elapsed. Writing a zero (from one) causes the port to return high-speed mode (forcing the bus below the port into a high-speed idle). This bit will remain a one until the port has switched to high-speed idle. The host controller must complete this transition within 2 milliseconds of software setting this bit to a zero.</p> <p>This field is zero if Port Power is zero.</p>
5	R/WC	0x0	<p>Over-current Change</p> <p>This bit gets set to a one when there is a change to Over-current Active. Software clears this bit by writing a one to this bit position.</p>
4	R	0x0	<p>Over-current Active</p> <p>0 = This port does not have an over-current condition 1 = This port currently has an over-current condition</p> <p>This bit will automatically transition from a one to a zero when the over current condition is removed.</p> <p>The default value of this bit is '0'.</p>
3	R/WC	0x0	<p>Port Enable/Disable Change</p> <p>1 = Port enabled/disabled status has changed 0 = No change</p> <p>For the root hub, this bit gets set to a one only when a port is disabled due to the appropriate conditions existing at the EOF2 point (See Chapter 11 of the USB Specification for the definition of a Port Error). Software clears this bit by writing a 1 to it.</p> <p>This field is zero if Port Power is zero.</p>
2	R/W	0x0	<p>Port Enabled/Disabled</p> <p>1=Enable 0=Disable</p> <p>Ports can only be enabled by the host controller as a part of the reset and enable. Software cannot enable a port by writing a one to this field. The host controller will only set this bit to a one when the reset sequence determines that the attached device is a high-speed device.</p>

Offset: 0x0054			Register Name: PORTSC
Bit	Read/Write	Default/Hex	Description
			<p>Ports can be disabled by either a fault condition(disconnect event or other fault condition) or by host software. Note that the bit status does not change until the port state actually changes. There may be a delay in disabling or enabling a port due to other host controller and bus events.</p> <p>When the port is disabled, downstream propagation of data is blocked on this port except for reset.</p> <p>The default value of this field is '0'.</p> <p>This field is zero if Port Power is zero.</p>
1	R/WC	0x0	<p>Connect Status Change</p> <p>1=Change in Current Connect Status</p> <p>0=No change</p> <p>Indicates a change has occurred in the current connect status of the port. The host controller sets this bit for all changes to the port device connect status, even if system software has not cleared an existing connect status change. For example, the insertion status changes twice before system software has cleared the changed condition, hub hardware will be “setting” an already-set bit. Software sets this bit to 0 by writing a 1 to it.</p> <p>This field is zero if Port Power is zero.</p>
0	R	0x0	<p>Current Connect Status</p> <p>Device is present on port when the value of this field is a one, and no device is present on port when the value of this field is a zero. This value reflects the current state of the port, and may not correspond directly to the event that caused the Connect Status Change(Bit 1) to be set.</p> <p>This field is zero if Port Power zero.</p>



**NOTE**

This register is only reset by hardware or in response to a host controller reset.

### 9.6.6 OHCI Register Description

#### 9.6.6.1 0x0404 OHCI Control Register (Default Value: 0x0000\_0000)

Offset: 0x0404				Register Name: HcRevision								
Bit	Read/Write		Default/Hex	Description								
	HCD	HC										
31:11	/	/	/	/								
10	R/W	R	0x0	<p><b>RemoteWakeupEnable</b></p> <p>This bit is used by HCD to enable or disable the remote wakeup feature upon the detection of upstream resume signaling. When this bit is set and the <b>ResumeDetected</b> bit in <b>HcInterruptStatus</b> is set, a remote wakeup is signaled to the host system. Setting this bit has no impact on the generation of hardware interrupt.</p>								
9	R/W	R/W	0x0	<p><b>RemoteWakeupConnected</b></p> <p>This bit indicates whether HC supports remote wakeup signaling. If remote wakeup is supported and used by the system, it is the responsibility of system firmware to set this bit during POST. HC clear the bit upon a hardware reset but does not alter it upon a software reset. Remote wakeup signaling of the host system is host-bus-specific and is not described in this specification.</p>								
8	R/W	R	0x0	<p><b>InterruptRouting</b></p> <p>This bit determines the routing of interrupts generated by events registered in HcInterruptStatus. If clear, all interrupt are routed to the normal host bus interrupt mechanism. If set interrupts are routed to the System Management Interrupt. HCD clears this bit upon a hardware reset, but it does not alter this bit upon a software reset. HCD uses this bit as a tag to indicate the ownership of HC.</p>								
7:6	R/W	R/W	0x0	<p>HostControllerFunctionalState for USB</p> <table border="1" data-bbox="673 1473 1426 1653"> <tr> <td>00b</td> <td>USBReset</td> </tr> <tr> <td>01b</td> <td>USBResume</td> </tr> <tr> <td>10b</td> <td>USBOperational</td> </tr> <tr> <td>11b</td> <td>USBSuspend</td> </tr> </table> <p>A transition to USBOperational from another state causes SOF generation to begin 1 ms later. HCD may determine whether HC has begun sending SOFs by reading the StartoFrame field of HcInterruptStatus.</p> <p>This field may be changed by HC only when in the USBSUSPEND state. HC may move from the USBSUSPEND state to the USBRESUME state after detecting the resume signaling from a downstream port.</p> <p>HC enters USBSUSPEND after a software reset, whereas it enters USBRESET after a hardware reset. The latter also resets the Root</p>	00b	USBReset	01b	USBResume	10b	USBOperational	11b	USBSuspend
00b	USBReset											
01b	USBResume											
10b	USBOperational											
11b	USBSuspend											

Offset: 0x0404			Register Name: HcRevision					
Bit	Read/Write		Default/Hex	Description				
	HCD	HC						
				Hub and asserts subsequent reset signaling to downstream ports.				
5	R/W	R	0x0	<p><b>BulkListEnable</b></p> <p>This bit is set to enable the processing of the Bulk list in the next Frame. If cleared by HCD, the processing of the Bulk list does not occur after the next SOF. HC checks this bit whenever it determines to process the list. When disabled, HCD may modify the list. If <i>HcBulkCurrentED</i> is pointing to an ED to be removed, HCD must advance the pointer by updating <i>HcBulkCurrentED</i> before re-enabling processing of the list.</p>				
4	R/W	R	0x0	<p><b>ControlListEnable</b></p> <p>This bit is set to enable the processing of the Control list in the next Frame. If cleared by HCD, the processing of the Control list does not occur after the next SOF. HC must check this bit whenever it determines to process the list. When disabled, HCD may modify the list. If <i>HcControlCurrentED</i> is pointing to an ED to be removed, HCD must advance the pointer by updating <i>HcControlCurrentED</i> before re-enabling processing of the list.</p>				
3	R/W	R	0x0	<p><b>IsochronousEnable</b></p> <p>This bit is used by HCD to enable/disable processing of isochronous EDs. While processing the periodic list in a Frame, HC checks the status of this bit when it finds an Isochronous ED (F=1). If set (enabled), HC continues processing the EDs. If cleared (disabled), HC halts processing of the periodic list (which now contains only isochronous EDs) and begins processing the Bulk/Control lists. Setting this bit is guaranteed to take effect in the next Frame (not the current Frame).</p>				
2	R/W	R	0x0	<p><b>PeriodicListEnable</b></p> <p>This bit is set to enable the processing of periodic list in the next Frame. If cleared by HCD, processing of the periodic list does not occur after the next SOF. HC must check this bit before it starts processing the list.</p>				
1:0	R/W	R	0x0	<p><b>ControlBulkServiceRatio</b></p> <p>This specifies the service ratio between Control and Bulk EDs. Before processing any of the nonperiodic lists, HC must compare the ratio specified with its internal count on how many nonempty Control EDs have been processed, in determining whether to continue serving another Control ED or switching to Bulk EDs. The internal count will be retained when crossing the frame boundary. In case of reset, HCD is responsible for restoring this value.</p> <table border="1" data-bbox="678 1982 1428 2072"> <thead> <tr> <th>CBSR</th> <th>No. of Control EDs Over Bulk EDs Served</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>1:1</td> </tr> </tbody> </table>	CBSR	No. of Control EDs Over Bulk EDs Served	0	1:1
CBSR	No. of Control EDs Over Bulk EDs Served							
0	1:1							

Offset: 0x0404			Register Name: HcRevision							
Bit	Read/Write		Default/Hex	Description						
	HCD	HC								
				<table border="1"> <tr> <td>1</td> <td>2:1</td> </tr> <tr> <td>2</td> <td>3:1</td> </tr> <tr> <td>3</td> <td>4:1</td> </tr> </table>	1	2:1	2	3:1	3	4:1
1	2:1									
2	3:1									
3	4:1									
The default value is 0x0.										

9.6.6.2 0x0408 OHCI Command Status Register (Default Value: 0x0000\_0000)

Offset: 0x0408			Register Name: HcCommandStatus	
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:18	/	/	0x0	<b>Reserved</b>
17:16	R	R/W	0x0	<p><b>SchedulingOverrunCount</b></p> <p>These bits are incremented on each scheduling overrun error. It is initialized to 00b and wraps around at 11b. This will be incremented when a scheduling overrun is detected even if <b>SchedulingOverrun</b> in <i>HcInterruptStatus</i> has already been set. This is used by HCD to monitor any persistent scheduling problem.</p>
15:4	/	/	/	/
3	R/W	R/W	0x0	<p><b>OwershipChangeRequest</b></p> <p>This bit is set by an OS HCD to request a change of control of the HC. When set HC will set the <b>OwnershipChange</b> field in <i>HcInterruptStatus</i>. After the changeover, this bit is cleared and remains so until the next request from OS HCD.</p>
2	R/W	R/W	0x0	<p><b>BulkListFilled</b></p> <p>This bit is used to indicate whether there are any TDs on the Bulk list. It is set by HCD whenever it adds a TD to an ED in the Bulk list. When HC begins to process the head of the Bulk list, it checks BLF. As long as <b>BulkListFilled</b> is 0, HC will not start processing the Bulk list. If <b>BulkListFilled</b> is 1, HC will start processing the Bulk list and will set BF to 0. If HC finds a TD on the list, then HC will set <b>BulkListFilled</b> to 1 causing the Bulk list processing to continue. If no TD is found on the Bulk list, and if HCD does not set <b>BulkListFilled</b>, then <b>BulkListFilled</b> will still be 0 when HC completes processing the Bulk list and Bulk list processing will stop.</p>
1	R/W	R/W	0x0	<p><b>ControlListFilled</b></p> <p>This bit is used to indicate whether there are any TDs on the Control list. It is set by HCD whenever it adds a TD to an ED in the Control list.</p> <p>When HC begins to process the head of the Control list, it checks</p>



Offset: 0x0408			Register Name: HcCommandStatus	
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
				CLF. As long as <b>ControlListFilled</b> is 0, HC will not start processing the Control list. If CF is 1, HC will start processing the Control list and will set <b>ControlListFilled</b> to 0. If HC finds a TD on the list, then HC will set <b>ControlListFilled</b> to 1 causing the Control list processing to continue. If no TD is found on the Control list, and if the HCD does not set <b>ControlListFilled</b> , then <b>ControlListFilled</b> will still be 0 when HC completes processing the Control list and Control list processing will stop.
0	R/W	R/E	0x0	<b>HostControllerReset</b> This bit is by HCD to initiate a software reset of HC. Regardless of the functional state of HC, it moves to the USBsuspend state in which most of the operational registers are reset except those stated otherwise; e.g, the InterruptRouting field of HcControl, and no Host bus accesses are allowed. This bit is cleared by HC upon the completion of the reset operation. The reset operation must be completed within 10 ms. This bit, when set, should not cause a reset to the Root Hub and no subsequent reset signaling should be asserted to its downstream ports.

9.6.6.3 0x040C OHCI Interrupt Status Register (Default Value: 0x0000\_0000)

Offset: 0x040C			Register Name: HcInterruptStatus	
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:7	/	/	/	/
6	R/W	R/W	0x0	<b>RootHubStatusChange</b> This bit is set when the content of <i>HcRhStatus</i> or the content of any of <i>HcRhPortStatus[NumberOfDownstreamPort]</i> has changed.
5	R/W	R/W	0x0	<b>FrameNumberOverflow</b> This bit is set when the MSb of <i>HcFmNumber</i> (bit 15) changes value, from 0 to 1 or from 1 to 0, and after <i>HccaFrameNumber</i> has been updated.
4	R/W	R/W	0x0	<b>UnrecoverableError</b> This bit is set when HC detects a system error not related to USB. HC should not proceed with any processing nor signaling before the system error has been corrected. HCD clears this bit after HC has been reset.
3	R/W	R/W	0x0	<b>ResumeDetected</b> This bit is set when HC detects that a device on the USB is asserting resume signaling. It is the transition from no resume signaling to

Offset: 0x040C				Register Name: HcInterruptStatus
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
				resume signaling causing this bit to be set. This bit is not set when HCD sets the USBResume state.
2	R/W	R/W	0x0	<b>StartofFrame</b> This bit is set by HC at each start of frame and after the update of HccaFrameNumber. HC also generates a SOF token at the same time.
1	R/W	R/W	0x0	<b>WritebackDoneHead</b> This bit is set immediately after HC has written HcDoneHead to HccaDoneHead. Further updates of the HccaDoneHead will not occur until this bit has been cleared. HCD should only clear this bit after it has saved the content of HccaDoneHead.
0	R/W	R/W	0x0	<b>SchedulingOverrun</b> This bit is set when the USB schedule for the current Frame overruns and after the update of HccaFrameNumber. A scheduling overrun will also cause the SchedulingOverrunCount of HcCommandStatus to be incremented.

9.6.6.4 0x0410 OHCI Interrupt Enable Register (Default Value: 0x0000\_0000)

Offset: 0x0410				Register Name: HcInterruptEnable Register	
Bit	Read/Write		Default/Hex	Description	
	HCD	HC			
31	R/W	R	0x0	<b>MasterInterruptEnable</b> A '0' written to this field is ignored by HC. A '1' written to this field enables interrupt generation due to events specified in the other bits of this register. This is used by HCD as Master Interrupt Enable.	
30:7	/	/	/	/	
6	R/W	R	0x0	<b>RootHubStatusChange Interrupt Enable</b>	
				0	Ignore;
				1	Enable interrupt generation due to Root Hub Status Change;
5	R/W	R	0x0	<b>FrameNumberOverflow Interrupt Enable</b>	
				0	Ignore;
				1	Enable interrupt generation due to Frame Number Overflow;
4	R/W	R	0x0	<b>UnrecoverableError Interrupt Enable</b>	
				0	Ignore;
				1	Enable interrupt generation due to Unrecoverable Error;
3	R/W	R	0x0	<b>ResumeDetected Interrupt Enable</b>	

Offset: 0x0410				Register Name: HcInterruptEnable Register	
Bit	Read/Write		Default/Hex	Description	
	HCD	HC			
				0	Ignore;
				1	Enable interrupt generation due to Resume Detected;
2	R/W	R	0x0	<b>StartofFrame Interrupt Enable</b>	
				0	Ignore;
				1	Enable interrupt generation due to Start of Flame;
1	R/W	R	0x0	<b>WritebackDoneHead Interrupt Enable</b>	
				0	Ignore;
				1	Enable interrupt generation due to Write back Done Head;
0	R/W	R	0x0	<b>SchedulingOverrun Interrupt Enable</b>	
				0	Ignore;
				1	Enable interrupt generation due to Scheduling Overrun;

9.6.6.5 0x0414 OHCI Interrupt Disable Register (Default Value: 0x0000\_0000)

Offset: 0x0414				Register Name: HcInterruptDisable Register	
Bit	Read/Write		Default/Hex	Description	
	HCD	HC			
31	R/W	R	0x0	<b>MasterInterruptEnable</b> A written '0' to this field is ignored by HC. A '1' written to this field disables interrupt generation due events specified in the other bits of this register. This field is set after a hardware or software reset.	
30:7	/	/	/	/	
6	R/W	R	0x0	<b>RootHubStatusChange Interrupt Disable</b>	
				0	Ignore;
				1	Disable interrupt generation due to Root Hub Status Change;
5	R/W	R	0x0	<b>FrameNumberOverflow Interrupt Disable</b>	
				0	Ignore;
				1	Disable interrupt generation due to Frame Number Over Flow;
4	R/W	R	0x0	<b>UnrecoverableError Interrupt Disable</b>	
				0	Ignore;
				1	Disable interrupt generation due to Unrecoverable Error;
3	R/W	R	0x0	<b>ResumeDetected Interrupt Disable</b>	
				0	Ignore;
				1	Disable interrupt generation due to Resume Detected;
2	R/W	R	0x0	<b>StartofFrame Interrupt Disable</b>	
				0	Ignore;
				1	Disable interrupt generation due to Start of Flame;

Offset: 0x0414				Register Name: HcInterruptDisable Register	
Bit	Read/Write		Default/Hex	Description	
	HCD	HC			
1	R/W	R	0x0	<b>WritebackDoneHead Interrupt Disable</b>	
				0	Ignore;
				1	Disable interrupt generation due to Write back Done Head;
0	R/w	R	0x0	<b>SchedulingOverrun Interrupt Disable</b>	
				0	Ignore;
				1	Disable interrupt generation due to Scheduling Overrun;

9.6.6.6 0x0418 OHCI HCCA Register (Default Value: 0x0000\_0000)

Offset: 0x0418				Register Name: HcHCCA	
Bit	Read/Write		Default/Hex	Description	
	HCD	HC			
31:8	R/W	R	0x0	<b>HCCA[31:8]</b> This is the base address of the Host Controller Communication Area. This area is used to hold the control structures and the Interrupt table that are accessed by both the Host Controller and the Host Controller Driver.	
7:0	R	R	0x0	<b>HCCA[7:0]</b> The alignment restriction in HcHCCA register is evaluated by examining the number of zeros in the lower order bits. The minimum alignment is 256 bytes, therefore, bits 0 through 7 must always return 0 when read.	

9.6.6.7 0x041C OHCI Period Current ED Register (Default Value: 0x0000\_0000)

Offset: 0x041C				Register Name: HcPeriodCurrentED[PCED]	
Bit	Read/Write		Default/Hex	Description	
	HCD	HC			
31:4	R	R/W	0x0	<b>PCED[31:4]</b> This is used by HC to point to the head of one of the Periodic list which will be processed in the current Frame. The content of this register is updated by HC after a periodic ED has been processed. HCD may read the content in determining which ED is currently being processed at the time of reading.	
3:0	R	R	0x0	<b>PCED[3:0]</b> Because the general TD length is 16 bytes, the memory structure for the TD must be aligned to a 16-byte boundary. So the lower bits in the PCED, through bit 0 to bit 3 must be zero in this field.	

9.6.6.8 0x0420 OHCI Control Head ED Register (Default Value: 0x0000\_0000)

Offset: 0x0420				Register Name: HcControlHeadED[CHED]
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:4	R/W	R	0x0	<b>EHCD[31:4]</b> The HcControlHeadED register contains the physical address of the first Endpoint Descriptor of the Control list. HC traverse the Control list starting with the HcControlHeadED pointer. The content is loaded from HCCA during the initialization of HC.
3:0	R	R	0x0	<b>EHCD[3:0]</b> Because the general TD length is 16 bytes, the memory structure for the TD must be aligned to a 16-byte boundary. So the lower bits in the PCED, through bit 0 to bit 3 must be zero in this field.

9.6.6.9 0x0424 OHCI Control Current ED Register (Default Value: 0x0000\_0000)

Offset: 0x0424				Register Name: HcControlCurrentED[CCED]
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:4	R/W	R/W	0x0	<b>CCED[31:4]</b> The pointer is advanced to the next ED after serving the present one. HC will continue processing the list from where it left off in the last Frame. When it reaches the end of the Control list, HC checks the ControllListFilled of in HcCommandStatus. If set, it copies the content of HcControlHeadED to HcControlCurrentED and clears the bit. If not set, it does nothing. HCD is allowed to modify this register only when the ControllListEnable of HcControl is cleared. When set, HCD only reads the instantaneous value of this register. Initially, this is set to zero to indicate the end of the Control list.
3:0	R	R	0x0	<b>CCED[3:0]</b> Because the general TD length is 16 bytes, the memory structure for the TD must be aligned to a 16-byte boundary. So the lower bits in the PCED, through bit 0 to bit 3 must be zero in this field.

9.6.6.10 0x0428 OHCI Bulk Head ED Register (Default Value: 0x0000\_0000)

Offset: 0x0428			Register Name: HcBulkHeadED[BHED]	
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:4	R/W	R	0x0	<b>BHED[31:4]</b> The HcBulkHeadED register contains the physical address of the first Endpoint Descriptor of the Bulk list. HC traverses the Bulk list starting with the HcBulkHeadED pointer. The content is loaded from HCCA during the initialization of HC.
3:0	R	R	0x0	<b>BHED[3:0]</b> Because the general TD length is 16 bytes, the memory structure for the TD must be aligned to a 16-byte boundary. So the lower bits in the PCED, through bit 0 to bit 3 must be zero in this field.

9.6.6.11 0x042C OHCI Bulk Current ED Register (Default Value: 0x0000\_0000)

Offset: 0x042C			Register Name: HcBulkCurrentED[BCED]	
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:4	R/W	R/W	0x0	<b>BulkCurrentED[31:4]</b> This is advanced to the next ED after the HC has served the present one. HC continues processing the list from where it left off in the last Frame. When it reaches the end of the Bulk list, HC checks the <b>ControlListFilled</b> of HcControl. If set, it copies the content of <i>HcBulkHeadED</i> to <i>HcBulkCurrentED</i> and clears the bit. If it is not set, it does nothing. HCD is only allowed to modify this register when the <b>BulkListEnable</b> of <i>HcControl</i> is cleared. When set, the HCD only reads the instantaneous value of this register. This is initially set to zero to indicate the end of the Bulk list.
3:0	R	R	0x0	<b>BulkCurrentED [3:0]</b> Because the general TD length is 16 bytes, the memory structure for the TD must be aligned to a 16-byte boundary. So the lower bits in the PCED, through bit 0 to bit 3 must be zero in this field.

9.6.6.12 0x0430 OHCI Done Head Register (Default Value: 0x0000\_0000)

Offset: 0x0430			Register Name: HcDoneHead	
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:4	R	R/W	0x0	<b>HcDoneHead[31:4]</b> When a TD is completed, HC writes the content of <i>HcDoneHead</i> to

Offset: 0x0430			Register Name: HcDoneHead	
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
				the NextTD field of the TD. HC then overwrites the content of <i>HcDoneHead</i> with the address of this TD. This is set to zero whenever HC writes the content of this register to HCCA. It also sets the <b>WritebackDoneHead</b> of <i>HcInterruptStatus</i> .
3:0	R	R	0x0	<b>HcDoneHead[3:0]</b> Because the general TD length is 16 bytes, the memory structure for the TD must be aligned to a 16-byte boundary. So the lower bits in the PCED, through bit 0 to bit 3 must be zero in this field.

9.6.6.13 0x0434 OHCI Frame Interval Register (Default Value: 0x0000\_2EDF)

Offset: 0x0434			Register Name: HcFmInterval Register	
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31	R/W	R	0x0	<b>FrameIntervalToggler</b> HCD toggles this bit whenever it loads a new value to <b>FrameInterval</b> .
30:16	R/W	R	0x0	<b>FSLargestDataPacket</b> This field specifies a value which is loaded into the Largest Data Packet Counter at the beginning of each frame. The counter value represents the largest amount of data in bits which can be sent or received by the HC in a single transaction at any given time without causing scheduling overrun. The field value is calculated by the HCD.
15:14	/	/	/	/
13:0	R/W	R	0x2edf	<b>FrameInterval</b> This specifies the interval between two consecutive SOFs in bit times. The nominal value is set to be 11,999. HCD should store the current value of this field before resetting HC. By setting the <b>HostControllerReset</b> field of <i>HcCommandStatus</i> as this will cause the HC to reset this field to its nominal value. HCD may choose to restore the stored value upon the completion of the Reset sequence.

9.6.6.14 0x0438 OHCI Frame Remaining Register (Default Value: 0x0000\_0000)

Offset: 0x0438			Register Name: HcFmRemaining	
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31	R	R/W	0x0	<b>FrameRemaining Toggle</b> This bit is loaded from the <b>FrameIntervalToggle</b> field of <i>HcFmInterval</i> whenever <b>FrameRemaining</b> reaches 0. This bit is used by HCD for the synchronization between <b>FrameInterval</b> and <b>FrameRemaining</b> .
30:14	/	/	/	/
13:0	R	RW	0x0	<b>FramRemaining</b> This counter is decremented at each bit time. When it reaches zero, it is reset by loading the <b>FrameInterval</b> value specified in <i>HcFmInterval</i> at the next bit time boundary. When entering the USBOPERATIONAL state, HC re-loads the content with the <b>FrameInterval</b> of <i>HcFmInterval</i> and uses the updated value from the next SOF.

9.6.6.15 0x043C OHCI Frame Number Register (Default Value: 0x0000\_0000)

Offset: 0x043C			Register Name: HcFmNumber	
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:16	/	/	/	/
15:0	R	R/W	0x0	<b>FrameNumber</b> This is incremented when <i>HcFmRemaining</i> is re-loaded. It will be rolled over to 0x0 after 0x0ffff. When entering the USBOPERATIONAL state, this will be incremented automatically. The content will be written to HCCA after HC has incremented the <b>FrameNumber</b> at each frame boundary and sent a SOF but before HC reads the first ED in that Frame. After writing to HCCA, HC will set the <b>StartofFrame</b> in <i>HcInterruptStatus</i> .

9.6.6.16 0x0440 OHCI Periodic Start Register (Default Value: 0x0000\_0000)

Offset: 0x0440			Register Name: HcPeriodicStatus	
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:14	/	/	/	/
13:0	R/W	R	0x0	<b>PeriodicStart</b> After a hardware reset, this field is cleared. This is then set by HCD



Offset: 0x0440			Register Name: HcPeriodicStatus	
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
				during the HC initialization. The value is calculated roughly as 10% off from <i>HcFmInterval</i> . A typical value will be 0x2A3F (or 0x3e67). When <i>HcFmRemaining</i> reaches the value specified, processing of the periodic lists will have priority over Control/Bulk processing. HC will therefore start processing the Interrupt list after completing the current Control or Bulk transaction that is in progress.

9.6.6.17 0x0444 OHCI LS Threshold Register (Default Value: 0x0000\_0628)

Offset: 0x0444			Register Name: HcLSThreshold	
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:12	/	/	/	/
11:0	R/W	R	0x0628	<p><b>LSThreshold</b></p> <p>This field contains a value which is compared to the <b>FrameRemaining</b> field prior to initiating a Low Speed transaction. The transaction is started only if <b>FrameRemaining</b> this field. The value is calculated by HCD with the consideration of transmission and setup overhead.</p>

9.6.6.18 0x0448 OHCI Root Hub DescriptorA Register (Default Value: 0x0200\_1201)

Offset: 0x0448			Register Name: HcRhDescriptorA			
Bit	Read/Write		Default/Hex	Description		
	HCD	HC				
31:24	R/W	R	0x2	<p><b>PowerOnToPowerGoodTime[POTPGT]</b></p> <p>This byte specifies the duration HCD has to wait before accessing a powered-on port of the Root Hub. It is implementation-specific. The unit of time is 2 ms. The duration is calculated as POTPGT * 2ms.</p>		
23:13	/	/	/	/		
12	R/W	R	0x1	<p><b>NoOverCurrentProtection</b></p> <p>This bit describes how the overcurrent status for the Root Hub ports are reported. When this bit is cleared, the <b>OverCurrentProtectionMode</b> field specifies global or per-port reporting.</p> <table border="1" data-bbox="683 1995 1428 2072"> <tr> <td>0</td> <td>Over-current status is reported collectively for all downstream ports.</td> </tr> </table>	0	Over-current status is reported collectively for all downstream ports.
0	Over-current status is reported collectively for all downstream ports.					

Offset: 0x0448				Register Name: HcRhDescriptorA					
Bit	Read/Write		Default/Hex	Description					
	HCD	HC							
				1	No overcurrent protection supported.				
11	R/W	R	0x0	<p><b>OverCurrentProtectionMode</b></p> <p>This bit describes how the overcurrent status for the Root Hub ports are reported. At reset, these fields should reflect the same mode as <b>PowerSwitchingMode</b>. This field is valid only if the <b>NoOverCurrentProtection</b> field is cleared.</p> <table border="1"> <tr> <td>0</td> <td>Over-current status is reported collectively for all downstream ports.</td> </tr> <tr> <td>1</td> <td>Over-current status is reported on per-port basis.</td> </tr> </table>		0	Over-current status is reported collectively for all downstream ports.	1	Over-current status is reported on per-port basis.
0	Over-current status is reported collectively for all downstream ports.								
1	Over-current status is reported on per-port basis.								
10	R	R	0x0	<p><b>Device Type</b></p> <p>This bit specifies that the Root Hub is not a compound device. The Root Hub is not permitted to be a compound device. This field should always read/write 0.</p>					
9	R/W	R	0x1	<p><b>PowerSwitchingMode</b></p> <p>This bit is used to specify how the power switching of the Root Hub ports is controlled. It is implementation-specific. This field is only valid if the <b>NoPowerSwitching</b> field is cleared.</p> <table border="1"> <tr> <td>0</td> <td>All ports are powered at the same time.</td> </tr> <tr> <td>1</td> <td>Each port is powered individually. This mode allows port power to be controlled by either the global switch or per-port switching. If the <b>PortPowerControlMask</b> bit is set, the port responds only to port power commands (<b>Set/ClearPortPower</b>). If the port mask is cleared, then the port is controlled only by the global power switch (<b>Set/ClearGlobalPower</b>).</td> </tr> </table>		0	All ports are powered at the same time.	1	Each port is powered individually. This mode allows port power to be controlled by either the global switch or per-port switching. If the <b>PortPowerControlMask</b> bit is set, the port responds only to port power commands ( <b>Set/ClearPortPower</b> ). If the port mask is cleared, then the port is controlled only by the global power switch ( <b>Set/ClearGlobalPower</b> ).
0	All ports are powered at the same time.								
1	Each port is powered individually. This mode allows port power to be controlled by either the global switch or per-port switching. If the <b>PortPowerControlMask</b> bit is set, the port responds only to port power commands ( <b>Set/ClearPortPower</b> ). If the port mask is cleared, then the port is controlled only by the global power switch ( <b>Set/ClearGlobalPower</b> ).								
8	R/W	R	0x0	<p><b>NoPowerSwitcing</b></p> <p>These bits are used to specify whether power switching is supported or ports are always powered. It is implementation-specific. When this bit is cleared, the <b>PowerSwitchingMode</b> specifies global or per-port switching.</p> <table border="1"> <tr> <td>0</td> <td>Ports are power switched.</td> </tr> <tr> <td>1</td> <td>Ports are always powered on when the HC is powered on.</td> </tr> </table>		0	Ports are power switched.	1	Ports are always powered on when the HC is powered on.
0	Ports are power switched.								
1	Ports are always powered on when the HC is powered on.								
7:0	R	R	0x01	<p><b>NumberDownstreamPorts</b></p> <p>These bits specify the number of downstream ports supported by the Root Hub. It is implementation-specific. The minimum number of ports is 1. The maximum number of ports supported.</p>					

9.6.6.19 0x044C HcRhDescriptorB Register (Default Value: 0x0000\_0000)

Offset: 0x044C				Register Name: HcRhDescriptorB Register										
Bit	Read/Write		Default/Hex	Description										
	HCD	HC												
31:16	R/W	R	0x0	<p><b>PortPowerControlMask</b></p> <p>Each bit indicates if a port is affected by a global power control command when <b>PowerSwitchingMode</b> is set. When set, the port's power state is only affected by per-port power control (<b>Set/ClearPortPower</b>). When cleared, the port is controlled by the global power switch (<b>Set/ClearGlobalPower</b>). If the device is configured to global switching mode (<b>PowerSwitchingMode = 0</b>), this field is not valid.</p> <table border="1"> <tr><td>Bit0</td><td>Reserved</td></tr> <tr><td>Bit1</td><td>Ganged-power mask on Port #1.</td></tr> <tr><td>Bit2</td><td>Ganged-power mask on Port #2.</td></tr> <tr><td>...</td><td></td></tr> <tr><td>Bit15</td><td>Ganged-power mask on Port #15.</td></tr> </table>	Bit0	Reserved	Bit1	Ganged-power mask on Port #1.	Bit2	Ganged-power mask on Port #2.	...		Bit15	Ganged-power mask on Port #15.
Bit0	Reserved													
Bit1	Ganged-power mask on Port #1.													
Bit2	Ganged-power mask on Port #2.													
...														
Bit15	Ganged-power mask on Port #15.													
15:0	R/W	R	0x0	<p><b>DeviceRemovable</b></p> <p>Each bit is dedicated to a port of the Root Hub. When cleared, the attached device is removable. When set, the attached device is not removable.</p> <table border="1"> <tr><td>Bit0</td><td>Reserved</td></tr> <tr><td>Bit1</td><td>Device attached to Port #1.</td></tr> <tr><td>Bit2</td><td>Device attached to Port #2.</td></tr> <tr><td>...</td><td></td></tr> <tr><td>Bit15</td><td>Device attached to Port #15.</td></tr> </table>	Bit0	Reserved	Bit1	Device attached to Port #1.	Bit2	Device attached to Port #2.	...		Bit15	Device attached to Port #15.
Bit0	Reserved													
Bit1	Device attached to Port #1.													
Bit2	Device attached to Port #2.													
...														
Bit15	Device attached to Port #15.													

9.6.6.20 0x0450 HcRhStatus Register (Default Value: 0x0000\_0000)

Offset: 0x0450				Register Name: HcRhStatus Register
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31	W	R	0x0	<p><b>(write)ClearRemoteWakeupEnable</b></p> <p>Write a '1' clears <b>DeviceRemoteWakeupEnable</b>. Writing a '0' has no effect.</p>
30:18	/	/	/	/
17	R/W	R	0x0	<p><b>OverCurrentIndicatorChang</b></p> <p>This bit is set by hardware when a change has occurred to the <b>OverCurrentIndicator</b> field of this register. The HCD clears this bit by writing a '1'. Writing a '0' has no effect.</p>
16	R/W	R	0x0	<p><b>(read)LocalPowerStartusChange</b></p>

Offset: 0x0450				Register Name: HcRhStatus Register				
Bit	Read/Write		Default/Hex	Description				
	HCD	HC						
				<p>The Root Hub does not support the local power status features, thus, this bit is always read as '0'.</p> <p><b>(write)SetGlobalPower</b> In global power mode (<b>PowerSwitchingMode=0</b>), This bit is written to '1' to turn on power to all ports (clear <b>PortPowerStatus</b>). In per-port power mode, it sets <b>PortPowerStatus</b> only on ports whose <b>PortPowerControlMask</b> bit is not set. Writing a '0' has no effect.</p>				
15	R/W	R	0x0	<p><b>(read)DeviceRemoteWakeupEnable</b> This bit enables a <b>ConnectStatusChange</b> bit as a resume event, causing a USBsuspend to USBResume state transition and setting the <b>ResumeDetected</b> interrupt.</p> <table border="1"> <tr> <td>0</td> <td><b>ConnectStatusChange</b> is not a remote wakeup event.</td> </tr> <tr> <td>1</td> <td><b>ConnectStatusChange</b> is a remote wakeup event.</td> </tr> </table> <p><b>(write)SetRemoteWakeupEnable</b> Writing a '1' sets DeviceRemoveWakeupEnable. Writing a '0' has no effect.</p>	0	<b>ConnectStatusChange</b> is not a remote wakeup event.	1	<b>ConnectStatusChange</b> is a remote wakeup event.
0	<b>ConnectStatusChange</b> is not a remote wakeup event.							
1	<b>ConnectStatusChange</b> is a remote wakeup event.							
14:2	/	/	/	/				
1	R	R/W	0x0	<p><b>OverCurrentIndicator</b> This bit reports overcurrent conditions when the global reporting is implemented. When set, an overcurrent condition exists. When cleared, all power operations are normal. If per-port overcurrent protection is implemented this bit is always '0'</p>				
0	R/W	R	0x0	<p><b>(Read)LocalPowerStatus</b> When read, this bit returns the LocalPowerStatus of the Root Hub. The Root Hub does not support the local power status feature; thus, this bit is always read as '0'.</p> <p><b>(Write)ClearGlobalPower</b> When write, this bit is operated as the ClearGlobalPower. In global power mode (<b>PowerSwitchingMode=0</b>), This bit is written to '1' to turn off power to all ports (clear <b>PortPowerStatus</b>). In per-port power mode, it clears <b>PortPowerStatus</b> only on ports whose <b>PortPowerControlMask</b> bit is not set. Writing a '0' has no effect.</p>				

9.6.6.21 0x0454 HcRhPortStatus Register (Default Value: 0x0000\_0100)

Offset: 0x0454				Register Name: HcRhPortStatus
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:21	/	/	/	/
20	R/W	R/W	0x0	<b>PortResetStatusChange</b> This bit is set at the end of the 10-ms port reset signal. The HCD writes a '1' to clear this bit. Writing a '0' has no effect.
				<table border="1"> <tr> <td>0</td> <td>port reset is not complete</td> </tr> <tr> <td>1</td> <td>port reset is complete</td> </tr> </table>
0	port reset is not complete			
1	port reset is complete			
19	R/W	R/W	0x0	<b>PortOverCurrentIndicatorChange</b> This bit is valid only if overcurrent conditions are reported on a per-port basis. This bit is set when Root Hub changes the <b>PortOverCurrentIndicator</b> bit. The HCD writes a '1' to clear this bit. Writing a '0' has no effect.
				<table border="1"> <tr> <td>0</td> <td>no change in <b>PortOverCurrentIndicator</b></td> </tr> <tr> <td>1</td> <td><b>PortOverCurrentIndicator</b> has changed</td> </tr> </table>
0	no change in <b>PortOverCurrentIndicator</b>			
1	<b>PortOverCurrentIndicator</b> has changed			
18	R/W	R/W	0x0	<b>PortSuspendStatusChange</b> This bit is set when the full resume sequence has been completed. This sequence includes the 20-s resume pulse, LS EOP, and 3-ms resynchronization delay. The HCD writes a '1' to clear this bit. Writing a '0' has no effect. This bit is also cleared when <b>ResetStatusChange</b> is set.
				<table border="1"> <tr> <td>0</td> <td>resume is not completed</td> </tr> <tr> <td>1</td> <td>resume completed</td> </tr> </table>
0	resume is not completed			
1	resume completed			
17	R/W	R/W	0x0	<b>PortEnableStatusChange</b> This bit is set when hardware events cause the <b>PortEnableStatus</b> bit to be cleared. Changes from HCD writes do not set this bit. The HCD writes a '1' to clear this bit. Writing a '0' has no effect.
				<table border="1"> <tr> <td>0</td> <td>no change in <b>PortEnableStatus</b></td> </tr> <tr> <td>1</td> <td>change in <b>PortEnableStatus</b></td> </tr> </table>
0	no change in <b>PortEnableStatus</b>			
1	change in <b>PortEnableStatus</b>			
16	R/W	R/W	0x0	<b>ConnectStatusChange</b> This bit is set whenever a connect or disconnect event occurs. The HCD writes a '1' to clear this bit. Writing a '0' has no effect. If <b>CurrentConnectStatus</b> is cleared when a <b>SetPortReset</b> , <b>SetPortEnable</b> , or <b>SetPortSuspend</b> write occurs, this bit is set to force the driver to re-evaluate the connection status since these writes should not occur if the port is disconnected.
				<table border="1"> <tr> <td>0</td> <td>no change in <b>PortEnableStatus</b></td> </tr> <tr> <td>1</td> <td>change in <b>PortEnableStatus</b></td> </tr> </table>
0	no change in <b>PortEnableStatus</b>			
1	change in <b>PortEnableStatus</b>			
<b>Note: If the DeviceRemovable[NDP] bit is set, this bit is set only</b>				

Offset: 0x0454				Register Name: HcRhPortStatus				
Bit	Read/Write		Default/Hex	Description				
	HCD	HC						
				<b>after a Root Hub reset to inform the system that the device is attached.</b>				
15:10	/	/	/	/				
9	R/W	R/W	0x0	<p>(read)<b>LowSpeedDeviceAttached</b> This bit indicates the speed of the device attached to this port. When set, a Low Speed device is attached to this port. When clear, a Full Speed device is attached to this port. This field is valid only when the <b>CurrentConnectStatus</b> is set.</p> <table border="1"> <tr> <td>0</td> <td>full speed device attached</td> </tr> <tr> <td>1</td> <td>low speed device attached</td> </tr> </table> <p>(write)<b>ClearPortPower</b> The HCD clears the <b>PortPowerStatus</b> bit by writing a '1' to this bit. Writing a '0' has no effect.</p>	0	full speed device attached	1	low speed device attached
0	full speed device attached							
1	low speed device attached							
8	R/W	R/W	0x1	<p>(read)<b>PortPowerStatus</b> This bit reflects the port's power status, regardless of the type of power switching implemented. This bit is cleared if an overcurrent condition is detected. HCD sets this bit by writing <b>SetPortPower</b> or <b>SetGlobalPower</b>. HCD clears this bit by writing <b>ClearPortPower</b> or <b>ClearGlobalPower</b>. Which power control switches are enabled is determined by <b>PowerSwitchingMode</b> and <b>PortPortControlMask[NumberDownstreamPort]</b>. In global switching mode(<b>PowerSwitchingMode=0</b>), only <b>Set/ClearGlobalPower</b> controls this bit. In per-port power switching (<b>PowerSwitchingMode=1</b>), if the <b>PortPowerControlMask[NDP]</b> bit for the port is set, only <b>Set/ClearPortPower</b> commands are enabled. If the mask is not set, only <b>Set/ClearGlobalPower</b> commands are enabled. When port power is disabled, <b>CurrentConnectStatus</b>, <b>PortEnableStatus</b>, <b>PortSuspendStatus</b>, and <b>PortResetStatus</b> should be reset.</p> <table border="1"> <tr> <td>0</td> <td>port power is off</td> </tr> <tr> <td>1</td> <td>port power is on</td> </tr> </table> <p>(write)<b>SetPortPower</b> The HCD writes a '1' to set the <b>PortPowerStatus</b> bit. Writing a '0' has no effect.</p> <p><b>Note: This bit is always reads '1b' if power switching is not supported.</b></p>	0	port power is off	1	port power is on
0	port power is off							
1	port power is on							
7:5	/	/	/	/				
4	R/W	R/W	0x0	(read) <b>PortResetStatus</b>				

Offset: 0x0454				Register Name: HcRhPortStatus				
Bit	Read/Write		Default/Hex	Description				
	HCD	HC						
				<p>When this bit is set by a write to <b>SetPortReset</b>, port reset signaling is asserted. When reset is completed, this bit is cleared when <b>PortResetStatusChange</b> is set. This bit cannot be set if <b>CurrentConnectStatus</b> is cleared.</p> <table border="1"> <tr> <td>0</td> <td>port reset signal is not active</td> </tr> <tr> <td>1</td> <td>port reset signal is active</td> </tr> </table> <p>(write)<b>SetPortReset</b> The HCD sets the port reset signaling by writing a '1' to this bit. Writing a '0' has no effect. If <b>CurrentConnectStatus</b> is cleared, this write does not set <b>PortResetStatus</b>, but instead sets <b>ConnectStatusChange</b>. This informs the driver that it attempted to reset a disconnected port.</p>	0	port reset signal is not active	1	port reset signal is active
0	port reset signal is not active							
1	port reset signal is active							
3	R/W	R/W	0x0	<p>(read)<b>PortOverCurrentIndicator</b> This bit is only valid when the Root Hub is configured in such a way that overcurrent conditions are reported on a per-port basis. If per-port overcurrent reporting is not supported, this bit is set to 0. If cleared, all power operations are normal for this port. If set, an overcurrent condition exists on this port. This bit always reflects the overcurrent input signal.</p> <table border="1"> <tr> <td>0</td> <td>no overcurrent condition.</td> </tr> <tr> <td>1</td> <td>overcurrent condition detected.</td> </tr> </table> <p>(write)<b>ClearSuspendStatus</b> The HCD writes a '1' to initiate a resume. Writing a '0' has no effect. A resume is initiated only if <b>PortSuspendStatus</b> is set.</p>	0	no overcurrent condition.	1	overcurrent condition detected.
0	no overcurrent condition.							
1	overcurrent condition detected.							
2	R/W	R/W	0x0	<p>(read)<b>PortSuspendStatus</b> This bit indicates the port is suspended or in the resume sequence. It is set by a <b>SetSuspendState</b> write and cleared when <b>PortSuspendStatusChange</b> is set at the end of the resume interval. This bit cannot be set if <b>CurrentConnectStatus</b> is cleared. This bit is also cleared when <b>PortResetStatusChange</b> is set at the end of the port reset or when the HC is placed in the USBRESUME state. If an upstream resume is in progress, it should propagate to the HC.</p> <table border="1"> <tr> <td>0</td> <td>port is not suspended</td> </tr> <tr> <td>1</td> <td>port is suspended</td> </tr> </table> <p>(write)<b>SetPortSuspend</b> The HCD sets the <b>PortSuspendStatus</b> bit by writing a '1' to this bit. Writing a '0' has no effect. If <b>CurrentConnectStatus</b> is cleared, this</p>	0	port is not suspended	1	port is suspended
0	port is not suspended							
1	port is suspended							



Offset: 0x0454			Register Name: HcRhPortStatus					
Bit	Read/Write		Default/Hex	Description				
	HCD	HC						
				write does not set <b>PortSuspendStatus</b> ; instead it sets <b>ConnectStatusChange</b> . This informs the driver that it attempted to suspend a disconnected port.				
1	R/W	R/W	0x0	<p>(read)<b>PortEnableStatus</b>            This bit indicates whether the port is enabled or disabled. The Root Hub may clear this bit when an overcurrent condition, disconnect event, switched-off power, or operational bus error such as babble is detected. This change also causes <b>PortEnabledStatusChange</b> to be set. HCD sets this bit by writing <b>SetPortEnable</b> and clears it by writing <b>ClearPortEnable</b>. This bit cannot be set when <b>CurrentConnectStatus</b> is cleared. This bit is also set, if not already, at the completion of a port reset when <b>ResetStatusChange</b> is set or port suspend when <b>SuspendStatusChange</b> is set.</p> <table border="1"> <tr> <td>0</td> <td>port is disabled</td> </tr> <tr> <td>1</td> <td>port is enabled</td> </tr> </table> <p>(write)<b>SetPortEnable</b>            The HCD sets <b>PortEnableStatus</b> by writing a '1'. Writing a '0' has no effect. If <b>CurrentConnectStatus</b> is cleared, this write does not set <b>PortEnableStatus</b>, but instead sets <b>ConnectStatusChange</b>. This informs the driver that it attempted to enable a disconnected Port.</p>	0	port is disabled	1	port is enabled
0	port is disabled							
1	port is enabled							
0	R/W	R/W	0x0	<p>(read)<b>CurrentConnectStatus</b>            This bit reflects the current state of the downstream port.</p> <table border="1"> <tr> <td>0</td> <td>No device connected</td> </tr> <tr> <td>1</td> <td>Device connected</td> </tr> </table> <p>(write)<b>ClearPortEnable</b>            The HCD writes a '1' to clear the PortEnableStatus bit. Writing '0' to this bit has no effect. The CurrentConnectStatus is not affected by any write.  <b>Note: This bit is always read '1' when the attached device is nonremovable (DviceRemoveable[NumberDownstreamPort]).</b></p>	0	No device connected	1	Device connected
0	No device connected							
1	Device connected							

9.6.6.22 0x0800 HCI Interface Register (Default Value: 0x1000\_0000)

Offset: 0x0800			Register Name: USB_CTRL
Bit	Read/Write	Default/Hex	Description
31:29	/	/	Reserved



Offset: 0x0800			Register Name: USB_CTRL
Bit	Read/Write	Default/Hex	Description
28	R	1	DMA Transfer Status Enable 0: Disable 1: Enable
27:26	/	/	/
25	R/W	0	OHCI count select 1: Simulation mode. The counters will be much shorter then real time 0: Normal mode. The counters will count full time
24:19	/	/	/
18	R/W	0	1: Within 2 us of the resume-K to SEO transition 0: Random time value of the resume-K to SEO transition
17:13	/	/	/
12	R/W	0	PP2VBUS 1: ULPI wrapper interface will automatically set or clear DrvVbus register in ULPI PHY according to the port power status form the root hub 0: ULPI wrapper will ignore the difference between power status of root hub and ULPI PHY
11	R/W	0	AHB Master interface INCR16 enable 1: Use INCR16 when appropriate 0: Do not use INCR16, use other enabled INCRX or unspecified length burst INCR
10	R/W	0	AHB Master interface INCR8 enable 1: Use INCR8 when appropriate 0: Do not use INCR8, use other enabled INCRX or unspecified length burst INCR
9	R/W	0	AHB Master interface burst type INCR4 enable 1: Use INCR4 when appropriate 0: Do not use INCR4, use other enabled INCRX or unspecified length burst INCR
8	R/W	0	AHB Master interface INCRX align enable 1: Start INCRx burst only on burst x-align address 0: Start burst on any double word boundary Note: This bit must enable if any bit of bit[11:9] is enabled
7:1	/	/	/
0	R/W	0	ULPI bypass enable 1: Enable UTMI interface, disable ULPI interface 0: Enable ULPI interface, disable UTMI interface

9.6.6.23 0x0808 HCI Control 3 Register (Default Value: 0x0001\_0000)

Offset: 0x0808			Register Name: HCI_CTRL3
Bit	Read/Write	Default/Hex	Description
31:17	/	/	Reserved
16	R/W1C	1	Linestate Change Detect 0: Linestate change not detected 1: Linestate change detected Write '1' to clear.
15:4	/	/	Reserved
3	R/W	0	Remote Wakeup Enable 1: Enable 0: Disable
2	/	/	Reserved
1	R/W	0	Linestate Change Interrupt Enable 1: Enable 0: Disable
0	R/W	0	Linestate Change Detect Enable 1: Enable 0: Disable

9.6.6.24 0x0810 PHY Control Register (Default Value: 0x0000\_0008)

Offset: 0x0810			Register Name: PHY_CTRL
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0	bist_en_a
15:8	R/W	0	vc_addr
7	R/W	0	vc_di
6:4	/	/	/
3	R/W	0x1	SIDDQ 1: Write 1 to disable phy 0: Write 0 to enable phy
2:1	/	/	/
0	R/W	0x0	vc_clk

9.6.6.25 0x0824 PHY Status Register (Default Value: 0x0000\_0000)

Offset: 0x0824			Register Name: PHY_STATUS
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17	R	0	Bist_error
16	R	0	bist_done
15:1	/	/	/
0	R	0	vc_do

9.6.6.26 0x0828 HCI SIE Port Disable Control Register (Default Value: 0x0000\_0000)

Offset: 0x0828			Register Name: USB_SPDCR
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0	SE0 Status This bit is set when no-se0 is detected before SOF when bit[1:0] is 10b or 11b
15:5	/	/	/
4	R/W	0	resume_sel When set k-se0 transition 2 us, setting this bit to 1, which is cooperated with ss_utmi_backward_enb_i.
3:2	/	/	/
1:0	R/W	0	Port Disable Control 00: Port Disable when no-se0 detect before SOF 01: Port Disable when no-se0 detect before SOF 10: No Port Disable when no-se0 detect before SOF 11: Port Disable when no-se0 3 time detect before SOF during 8 frames

## 9.7 GPIO

### 9.7.1 Overview

The general purpose input/output (GPIO) is one of the blocks controlling the chip multiplexing pins. The D1s supports 6 groups of GPIO pins. Each pin can be configured as input or output and these pins are used to generate input signals or output signals for special purposes.

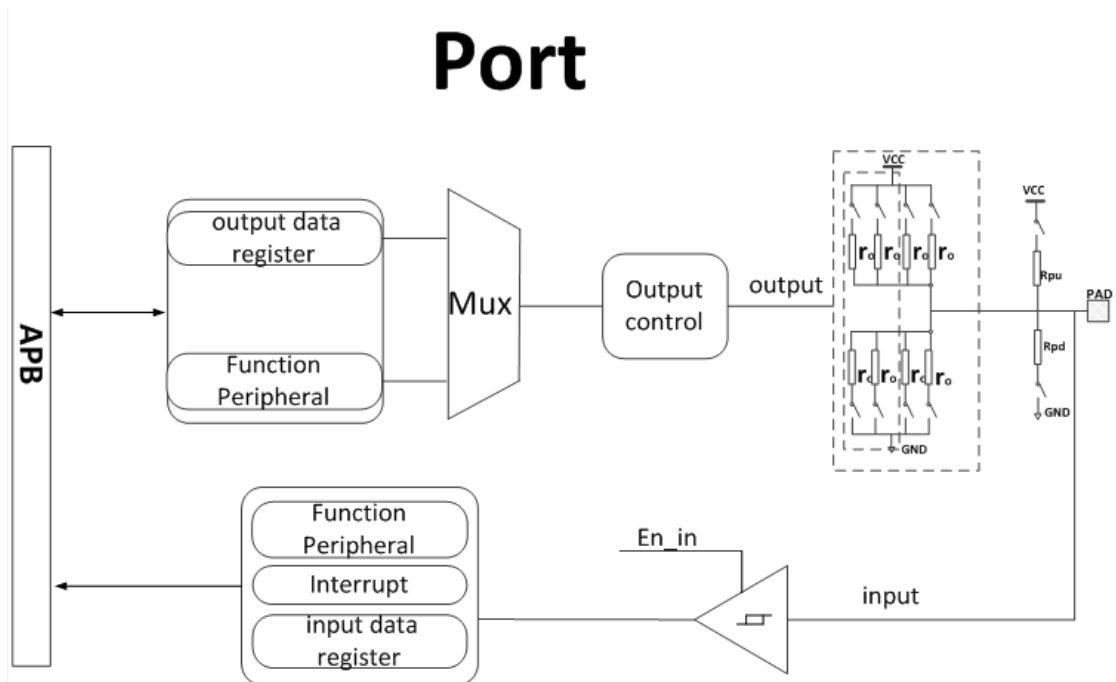
The Port Controller has the following features:

- 6 groups of ports (PB, PC, PD, PE, PF, PG)
- Software control for each signal pin
- Data input (capture)/output (drive)
- Each GPIO peripheral can produce an interrupt
- Pull-up/Pull-down/no-Pull register control
- Control the direction of every signal
- 4 drive strengths in each operating mode
- Up to 72 interrupts
- Configurable interrupt edges

### 9.7.2 Block Diagram

The following figure shows the block diagram of the GPIO.

Figure 9-67 GPIO Block Diagram



The GPIO consists of the digital part (GPIO, external interface) and IO analog part (output buffer, dual pull down, pad). The digital part can select the output interface by the MUX switch; the analog part can configure pull up/down and buffer strength.

When executing GPIO read state, the GPIO reads the current level of the pin into the internal register bus. When not executing GPIO read state, the external pin and the internal register bus are off-status, which is high-impedance.

### 9.7.3 Functional Description

#### 9.7.3.1 Multi-function Port

The D1s includes 72 multi-functional input/output port pins. There are 6 ports as listed below.

Table 9-20 Multi-function Port

Port Name	Number of Pins	Input Driver	Output Driver	Multiplex Pins	Power
PB	6	Schmitt	CMOS	LCD/I2S/TWI/PWM/IR/UART/PB-EINT	3.3 V
PC	6	Schmitt	CMOS	SPI/SMHC/UART/BOOT/TWI/TCON/PC-EINT	3.3 V
PD	23	Schmitt	CMOS	LCD/LVDS/OWA/TWI/IR/DSI/SPI-DBI/DMIC/UART/PWM/IR/PD-EINT	3.3 V/ 1.8 V

Port Name	Number of Pins	Input Driver	Output Driver	Multiplex Pins	Power
PE	14	Schmitt	CMOS	NCSI/TWI/UART/PWM/LCD/OWA/LEDC/IR/JTAG/EMAC/PE-EINT	3.3 V/ 2.8 V/ 1.8 V
PF	7	Schmitt	CMOS	SMHC/JTAG/UART/OWA/TWI/IR/I2S/LEDC/PWM/PF-EINT	3.3 V
PG	16	Schmitt	CMOS	SMHC/UART/PWM/I2S/TWI/EMAC/OWA/IR/TCON/LEDC/SPI/PG-EINT	3.3 V/ 1.8 V



9.7.3.2 GPIO Multiplex Function

Table 9-21 to Table 9-26 show the multiplex function pins of the D1s.

NOTE

For each GPIO, Function0 is input function; Function1 is output function; Function9 to Function13 are reserved.

Table 9-21 PB Multiplex Function

GPIO Port	Function 2	Function 3	Function 4	Function 5	Function 6	Function 7	Function 8	Function 14
PB2	LCD0-D0	I2S2-DOUT2	TWI0-SDA	I2S2-DIN2	LCD0-D18	UART4-TX		PB-EINT2
PB3	LCD0-D1	I2S2-DOUT1	TWI0-SCK	I2S2-DIN0	LCD0-D19	UART4-RX		PB-EINT3
PB4	LCD0-D8	I2S2-DOUT0	TWI1-SCK	I2S2-DIN1	LCD0-D20	UART5-TX		PB-EINT4
PB5	LCD0-D9	I2S2-BCLK	TWI1-SDA	PWM0	LCD0-D21	UART5-RX		PB-EINT5
PB6	LCD0-D16	I2S2-LRCK	TWI3-SCK	PWM1	LCD0-D22	UART3-TX	CPUBIST0	PB-EINT6
PB7	LCD0-D17	I2S2-MCLK	TWI3-SDA	IR-RX	LCD0-D23	UART3-RX	CPUBIST1	PB-EINT7

Table 9-22 PC Multiplex Function

GPIO Port	Function 2	Function 3	Function 4	Function 5	Function 6	Function 7	Function 8	Function 14
PC2	SPI0-CLK	SDC2-CLK						PC-EINT2
PC3	SPI0-CS0	SDC2-CMD						PC-EINT3
PC4	SPI0-MOSI	SDC2-D2	BOOT-SEL0					PC-EINT4
PC5	SPI0-MISO	SDC2-D1	BOOT-SEL1					PC-EINT5
PC6	SPI0-WP	SDC2-D0	UART3-TX	TWI3-SCK	DBG-CLK			PC-EINT6
PC7	SPI0-HOLD	SDC2-D3	UART3-RX	TWI3-SDA	TCON-TRIG			PC-EINT7

Table 9-23 PD Multiplex Function

GPIO Port	Function 2	Function 3	Function 4	Function 5	Function 6	Function 7	Function 8	Function 14
PD0	LCD0-D2	LVDS0-V0P	DSI-D0P	TWI0-SCK				PD-EINT0
PD1	LCD0-D3	LVDS0-V0N	DSI-D0N	UART2-TX				PD-EINT1
PD2	LCD0-D4	LVDS0-V1P	DSI-D1P	UART2-RX				PD-EINT2
PD3	LCD0-D5	LVDS0-V1N	DSI-D1N	UART2-RTS				PD-EINT3
PD4	LCD0-D6	LVDS0-V2P	DSI-CKP	UART2-CTS				PD-EINT4
PD5	LCD0-D7	LVDS0-V2N	DSI-CKN	UART5-TX				PD-EINT5
PD6	LCD0-D10	LVDS0-CKP	DSI-D2P	UART5-RX				PD-EINT6
PD7	LCD0-D11	LVDS0-CKN	DSI-D2N	UART4-TX				PD-EINT7
PD8	LCD0-D12	LVDS0-V3P	DSI-D3P	UART4-RX				PD-EINT8
PD9	LCD0-D13	LVDS0-V3N	DSI-D3N	PWM6				PD-EINT9
PD10	LCD0-D14	LVDS1-V0P	SPI1-CS/DBI-CSX	UART3-TX				PD-EINT10
PD11	LCD0-D15	LVDS1-V0N	SPI1-CLK/ DBI-SCLK	UART3-RX				PD-EINT11
PD12	LCD0-D18	LVDS1-V1P	SPI1-MOSI/ DBI-SDO	TWI0-SDA				PD-EINT12

GPIO Port	Function 2	Function 3	Function 4	Function 5	Function 6	Function 7	Function 8	Function 14
PD13	LCD0-D19	LVDS1-V1N	SPI1-MISO/ DBI-SDI/DBI-TE/ DBI-DCX	UART3-RTS				PD-EINT13
PD14	LCD0-D20	LVDS1-V2P	SPI1-HOLD/ DBI-DCX/ DBI-WRX	UART3-CTS				PD-EINT14
PD15	LCD0-D21	LVDS1-V2N	SPI1-WP/DBI-TE	IR-RX				PD-EINT15
PD16	LCD0-D22	LVDS1-CKP	DMIC-DATA3	PWM0				PD-EINT16
PD17	LCD0-D23	LVDS1-CKN	DMIC-DATA2	PWM1				PD-EINT17
PD18	LCD0-CLK	LVDS1-V3P	DMIC-DATA1	PWM2				PD-EINT18
PD19	LCD0-DE	LVDS1-V3N	DMIC-DATA0	PWM3				PD-EINT19
PD20	LCD0-HSYNC	TWI2-SCK	DMIC-CLK	PWM4				PD-EINT20
PD21	LCD0-VSYNC	TWI2-SDA	UART1-TX	PWM5				PD-EINT21
PD22	OWA-OUT	IR-RX	UART1-RX	PWM7				PD-EINT22

Table 9-24 PE Multiplex Function

GPIO Port	Function 2	Function 3	Function 4	Function 5	Function 6	Function 7	Function 8	Function 14
PE0	NCSI0-HSYNC	UART2-RTS	TWI1-SCK	LCD0-HSYNC			RGMII-RXCTRL/ RMII-CRS-DV	PE-EINT0
PE1	NCSI0-VSYNC	UART2-CTS	TWI1-SDA	LCD0-VSYNC			RGMII-RXD0/ RMII-RXD0	PE-EINT1
PE2	NCSI0-PCLK	UART2-TX	TWI0-SCK	CLK-FANOUT0	UART0-TX		RGMII-RXD1/ RMII-RXD1	PE-EINT2
PE3	NCSI0-MCLK	UART2-RX	TWI0-SDA	CLK-FANOUT1	UART0-RX		RGMII-TXCK/ RMII-TXCK	PE-EINT3
PE4	NCSI0-D0	UART4-TX	TWI2-SCK	CLK-FANOUT2		R-JTAG-MS	RGMII-TXD0/ RMII-TXD0	PE-EINT4
PE5	NCSI0-D1	UART4-RX	TWI2-SDA	LEDC-DO		R-JTAG-DI	RGMII-TXD1/ RMII-TXD1	PE-EINT5
PE6	NCSI0-D2	UART5-TX	TWI3-SCK	OWA-IN		R-JTAG-DO	RGMII-TXCTRL/ RMII-TXEN	PE-EINT6
PE7	NCSI0-D3	UART5-RX	TWI3-SDA	OWA-OUT		R-JTAG-CK	RGMII-CLKIN/ RMII-RXER	PE-EINT7
PE8	NCSI0-D4	UART1-RTS	PWM2	UART3-TX			MDC	PE-EINT8
PE9	NCSI0-D5	UART1-CTS	PWM3	UART3-RX			MDIO	PE-EINT9
PE10	NCSI0-D6	UART1-TX	PWM4	IR-RX			EPHY-25M	PE-EINT10
PE11	NCSI0-D7	UART1-RX					RGMII-TXD2	PE-EINT11
PE12	TWI2-SCK	NCSI0-FIELD					RGMII-TXD3	PE-EINT12
PE13	TWI2-SDA	PWM5			DMIC-DATA3		RGMII-RXD2	PE-EINT13

Table 9-25 PF Multiplex Function

GPIO Port	Function 2	Function 3	Function 4	Function 5	Function 6	Function 7	Function 8	Function 14
PF0	SDC0-D1		R-JTAG-MS	I2S2-DOUT1	I2S2-DIN0			PF-EINT0
PF1	SDC0-D0		R-JTAG-DI	I2S2-DOUT0	I2S2-DIN1			PF-EINT1
PF2	SDC0-CLK	UART0-TX	TWI0-SCK	LEDC-DO	OWA-IN			PF-EINT2
PF3	SDC0-CMD		R-JTAG-DO	I2S2-BCLK				PF-EINT3
PF4	SDC0-D3	UART0-RX	TWI0-SDA	PWM6	IR-TX			PF-EINT4



GPIO Port	Function 2	Function 3	Function 4	Function 5	Function 6	Function 7	Function 8	Function 14
PF5	SDC0-D2		R-JTAG-CK	I2S2-LRCK				PF-EINT5
PF6		OWA-OUT	IR-RX	I2S2-MCLK	PWM5			PF-EINT6

Table 9-26 PG Multiplex Function

GPIO Port	Function 2	Function 3	Function 4	Function 5	Function 6	Function 7	Function 8	Function 14
PG0	SDC1-CLK	UART3-TX	RGMII-RXCTRL/ RMII-CRS-DV	PWM7				PG-EINT0
PG1	SDC1-CMD	UART3-RX	RGMII-RXD0/ RMII-RXD0	PWM6				PG-EINT1
PG2	SDC1-D0	UART3-RTS	RGMII-RXD1/ RMII-RXD1	UART4-TX				PG-EINT2
PG3	SDC1-D1	UART3-CTS	RGMII-TXCK/ RMII-TXCK	UART4-RX				PG-EINT3
PG4	SDC1-D2	UART5-TX	RGMII-TXD0/ RMII-TXD0	PWM5				PG-EINT4
PG5	SDC1-D3	UART5-RX	RGMII-TXD1/ RMII-TXD1	PWM4				PG-EINT5
PG6	UART1-TX	TWI2-SCK	RGMII-TXD2	PWM1				PG-EINT6
PG7	UART1-RX	TWI2-SDA	RGMII-TXD3	OWA-IN				PG-EINT7
PG8	UART1-RTS	TWI1-SCK	RGMII-RXD2	UART3-TX				PG-EINT8
PG9	UART1-CTS	TWI1-SDA	RGMII-RXD3	UART3-RX				PG-EINT9
PG10	PWM3	TWI3-SCK	RGMII-RXCK	CLK-FANOUT0	IR-RX			PG-EINT10
PG11	I2S1-MCLK	TWI3-SDA	EPHY-25M	CLK-FANOUT1	TCON-TRIG			PG-EINT11
PG12	I2S1-LRCK	TWI0-SCK	RGMII-TXCTRL/ RMII-TXEN	CLK-FANOUT2	PWM0	UART1-TX		PG-EINT12
PG13	I2S1-BCLK	TWI0-SDA	RGMII-CLKIN/ RMII-RXER	PWM2	LEDC-DO	UART1-RX		PG-EINT13
PG14	I2S1-DIN0	TWI2-SCK	MDC	I2S1-DOUT1	SPI0-WP	UART1-RTS		PG-EINT14
PG15	I2S1-DOUT0	TWI2-SDA	MDIO	I2S1-DIN1	SPI0-HOLD	UART1-CTS		PG-EINT15

### 9.7.3.3 Port Function

The Port Controller supports 6 GPIOs, every GPIO can configure as Input, Output, Function Peripheral, IO disable or Interrupt function. The configuration instruction of every function is as follows.

**Table 9-27 Port Function**

	Function	Buffer Strength	Pull Up	Pull Down
Input	GPIO/Multiplexing Input	/	X	X
Output	GPIO/Multiplexing Output	Y	X	X
Disable	Pull Up	/	Y	N
	Pull Down	/	N	Y
Interrupt	Trigger	/	X	X

/: non-configure, configuration is invalid

Y: configure

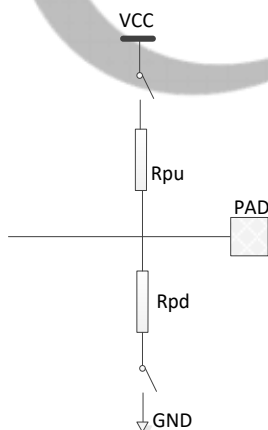
X: Select configuration according to the actual situation

N: Forbid to configure

### 9.7.3.4 Pull Up/Down and High-Impedance Logic

Each IO pin can configure the internal pull-up/down function or high-impedance.

**Figure 9-68 Pull up/down Logic**



**High-impedance**, the output is float state, all buffer is off, the level is decided by external high/low level. When high-impedance, the software configures the switch on Rpu and Rpd as off, and the multiplexing function of IO is set as IO disable or input by software.

**Pull-up**, an uncertain signal is pulled high by resistance, the resistance has a current-limiting function. When pulling up, the switch on Rpu is conducted by software configuration, the IO is pulled up to VCC by Rpu.

**Pull-down**, an uncertain signal is pulled low by a resistance. When pulling down, the switch on Rpd is conducted by software configuration, the IO is pulled down to GND by Rpd.

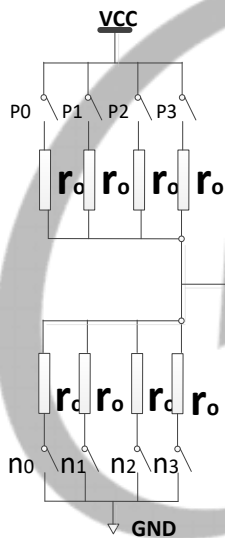
The pull-up/down of each IO is weak pull-up/down.

The setting of pull-down, pull-up, high-impedance is decided by the external circuit.

### 9.7.3.5 Buffer Strength

Each IO can be set as different buffer strength. The IO buffer diagram is as follows.

Figure 9-69 IO Buffer Strength Diagram



When output high level, the n0, n1, n2, n3 of NMOS is off, the p0, p1, p2, p3 of PMOS is on. When the buffer strength is set to 0 (buffer strength is weakest), only the p0 is on, the output impedance is maximum, the impedance value is  $r_0$ . When the buffer strength is set to 1, only the p0 and p1 is on, the output impedance is equivalent to two  $r_0$  in parallel, the impedance value is  $r_0/2$ . When the buffer strength is 2, only the p0, p1, and p2 is on, the output impedance is equivalent to three  $r_0$  in parallel, the impedance value is  $r_0/3$ . When buffer strength is 3, the p0, p1, p2, and p3 is on, the output impedance is equivalent to four  $r_0$  in parallel, the impedance value is  $r_0/4$ .

When output low level, the p0, p1, p2, p3 of PMOS is off, the n0, n1, n2, n3 of NMOS is on. When the buffer strength is set to 0 (buffer strength is weakest), only the n0 is on, the output impedance is maximum, the

impedance value is  $r_0$ . When the buffer strength is set to 1, only the  $n_0$  and  $n_1$  is on, the output impedance is equivalent to two  $r_0$  in parallel, the impedance value is  $r_0/2$ . When the buffer strength is 2, only the  $n_0$ ,  $n_1$ , and  $n_2$  is on, the output impedance is equivalent to three  $r_0$  in parallel, the impedance value is  $r_0/3$ . When the buffer strength is 3, the  $n_0$ ,  $n_1$ ,  $n_2$ , and  $n_3$  is on, the output impedance is equivalent to four  $r_0$  in parallel, the impedance value is  $r_0/4$ .

When GPIO is set to input or interrupt function, between the output driver circuit and the port is unconnected, the driver configuration is invalid.



**NOTE**

The typical value of  $r_0$  is  $180\Omega$ .

### 9.7.3.6 Interrupt

Each group IO has an independent interrupt number. The IO within-group uses one interrupt number when one IO generates interrupt, the GPIO pins sent interrupt request to interrupt module. External Interrupt Status Register is used to query which IO generates interrupt.

The interrupt trigger of GPIO supports the following trigger types.

- Positive Edge: When a low level changes to a high level, the interrupt will generate. No matter how long a high level keeps, the interrupt generates only once.
- Negative Edge: When a high level changes to a low level, the interrupt will generate. No matter how long a low level keeps, the interrupt generates only once.
- High Level: Just keep a high level and the interrupt will always generate.
- Low Level: Just keep a low level and the interrupt will always generate.
- Double Edge: Positive and negative edge.

External Interrupt Configure Register is used to configure the trigger type.

The GPIO interrupt supports hardware debounce function by setting External Interrupt Debounce Register. Sample trigger signal using a lower sample clock, to reach the debounce effect because the dither frequency of the signal is higher than the sample frequency.

Set the sample clock source by `PIO_INT_CLK_SELECT` and the prescale factor by `DEB_CLK_PRE_SCALE`.

### 9.7.4 Register List

Module Name	Base Address
GPIO	0x02000000

Register Name	Offset	Description
PB_CFG0	0x0030	PB Configure Register 0
PB_CFG1	0x0034	PB Configure Register 1
PB_DAT	0x0040	PB Data Register
PB_DRV0	0x0044	PB Multi_Driving Register 0
PB_DRV1	0x0048	PB Multi_Driving Register 1
PB_PULL0	0x0054	PB Pull Register 0
PC_CFG0	0x0060	PC Configure Register 0
PC_DAT	0x0070	PC Data Register
PC_DRV0	0x0074	PC Multi_Driving Register 0
PC_PULL0	0x0084	PC Pull Register 0
PD_CFG0	0x0090	PD Configure Register 0
PD_CFG1	0x0094	PD Configure Register 1
PD_CFG2	0x0098	PD Configure Register 2
PD_DAT	0x00A0	PD Data Register
PD_DRV0	0x00A4	PD Multi_Driving Register 0
PD_DRV1	0x00A8	PD Multi_Driving Register 1
PD_DRV2	0x00AC	PD Multi_Driving Register 2
PD_PULL0	0x00B4	PD Pull Register 0
PD_PULL1	0x00B8	PD Pull Register 1
PE_CFG0	0x00C0	PE Configure Register 0
PE_CFG1	0x00C4	PE Configure Register 1
PE_DAT	0x00D0	PE Data Register
PE_DRV0	0x00D4	PE Multi_Driving Register 0
PE_DRV1	0x00D8	PE Multi_Driving Register 1
PE_PULL0	0x00E4	PE Pull Register 0
PF_CFG0	0x00F0	PF Configure Register 0
PF_DAT	0x0100	PF Data Register
PF_DRV0	0x0104	PF Multi_Driving Register 0
PF_PULL0	0x0114	PF Pull Register 0
PG_CFG0	0x0120	PG Configure Register 0
PG_CFG1	0x0124	PG Configure Register 1
PG_DAT	0x0130	PG Data Register

Register Name	Offset	Description
PG_DRV0	0x0134	PG Multi_Driving Register 0
PG_DRV1	0x0138	PG Multi_Driving Register 1
PG_DRV3	0x0140	PG Multi_Driving Register 3
PG_PULL0	0x0144	PG Pull Register 0
PB_EINT_CFG0	0x0220	PB External Interrupt Configure Register 0
PB_EINT_CTL	0x0230	PB External Interrupt Control Register
PB_EINT_STATUS	0x0234	PB External Interrupt Status Register
PB_EINT_DEB	0x0238	PB External Interrupt Debounce Register
PC_EINT_CFG0	0x0240	PC External Interrupt Configure Register 0
PC_EINT_CTL	0x0250	PC External Interrupt Control Register
PC_EINT_STATUS	0x0254	PC External Interrupt Status Register
PC_EINT_DEB	0x0258	PC External Interrupt Debounce Register
PD_EINT_CFG0	0x0260	PD External Interrupt Configure Register 0
PD_EINT_CFG1	0x0264	PD External Interrupt Configure Register 1
PD_EINT_CFG2	0x0268	PD External Interrupt Configure Register 2
PD_EINT_CTL	0x0270	PD External Interrupt Control Register
PD_EINT_STATUS	0x0274	PD External Interrupt Status Register
PD_EINT_DEB	0x0278	PD External Interrupt Debounce Register
PE_EINT_CFG0	0x0280	PE External Interrupt Configure Register 0
PE_EINT_CFG1	0x0284	PE External Interrupt Configure Register 1
PE_EINT_CTL	0x0290	PE External Interrupt Control Register
PE_EINT_STATUS	0x0294	PE External Interrupt Status Register
PE_EINT_DEB	0x0298	PE External Interrupt Debounce Register
PF_EINT_CFG0	0x02A0	PF External Interrupt Configure Register 0
PF_EINT_CTL	0x02B0	PF External Interrupt Control Register
PF_EINT_STATUS	0x02B4	PF External Interrupt Status Register
PF_EINT_DEB	0x02B8	PF External Interrupt Debounce Register
PG_EINT_CFG0	0x02C0	PG External Interrupt Configure Register 0
PG_EINT_CFG1	0x02C4	PG External Interrupt Configure Register 1
PG_EINT_CTL	0x02D0	PG External Interrupt Control Register
PG_EINT_STATUS	0x02D4	PG External Interrupt Status Register
PG_EINT_DEB	0x02D8	PG External Interrupt Debounce Register
PIO_POW_MOD_SEL	0x0340	PIO Group Withstand Voltage Mode Select Register
PIO_POW_MS_CTL	0x0344	PIO Group Withstand Voltage Mode Select Control Register
PIO_POW_VAL	0x0348	PIO Group Power Value Register

Register Name	Offset	Description
PIO_POW_VOL_SEL_CTL	0x0350	PIO Group Power Voltage Select Control Register

### 9.7.5 Register Description

#### 9.7.5.1 0x0030 PB Configure Register 0 (Default Value: 0xFFFF\_FFFF)

Offset: 0x0030			Register Name: PB_CFG0	
Bit	Read/Write	Default/Hex	Description	
31:28	R/W	0xF	PB7_SELECT PB7 Select 0000:Input 0010:LCD0-D17 0100:TWI3-SDA 0110:LCD0-D23 1000:CPUBIST1 1110:PB-EINT7	0001:Output 0011:I2S2-MCLK 0101:IR-RX 0111:UART3-RX 1001:Reserved 1111:IO Disable
27:24	R/W	0xF	PB6_SELECT PB6 Select 0000:Input 0010:LCD0-D16 0100:TWI3-SCK 0110:LCD0-D22 1000:CPUBIST0 1110:PB-EINT6	0001:Output 0011:I2S2-LRCK 0101:PWM1 0111:UART3-TX 1001:Reserved 1111:IO Disable
23:20	R/W	0xF	PB5_SELECT PB5 Select 0000:Input 0010:LCD0-D9 0100:TWI1-SDA 0110:LCD0-D21 1000:Reserved 1110:PB-EINT5	0001:Output 0011:I2S2-BCLK 0101:PWM0 0111:UART5-RX 1001:Reserved 1111:IO Disable

Offset: 0x0030			Register Name: PB_CFG0
Bit	Read/Write	Default/Hex	Description
19:16	R/W	0xF	PB4_SELECT PB4 Select 0000:Input                      0001:Output 0010:LCD0-D8                    0011:I2S2-DOUT0 0100:TWI1-SCK                   0101:I2S2-DIN1 0110:LCD0-D20                   0111:UART5-TX 1000:Reserved                    1001:Reserved 1110:PB-EINT4                   1111:IO Disable
15:12	R/W	0xF	PB3_SELECT PB3 Select 0000:Input                      0001:Output 0010:LCD0-D1                    0011:I2S2-DOUT1 0100:TWI0-SCK                   0101:I2S2-DIN0 0110:LCD0-D19                   0111:UART4-RX 1000:Reserved                    1001:Reserved 1110:PB-EINT3                   1111:IO Disable
11:8	R/W	0xF	PB2_SELECT PB2 Select 0000:Input                      0001:Output 0010:LCD0-D0                    0011:I2S2-DOUT2 0100:TWI0-SDA                   0101:I2S2-DIN2 0110:LCD0-D18                   0111:UART4-TX 1000:Reserved                    1001:Reserved 1110:PB-EINT2                   1111:IO Disable
7:4	R/W	0xF	Reserved
3:0	R/W	0xF	Reserved

9.7.5.2 0x0034 PB Configure Register 1 (Default Value: 0x000F\_FFFF)

Offset: 0x0034			Register Name: PB_CFG1
Bit	Read/Write	Default/Hex	Description
31:20	/	/	/
19:16	R/W	0xF	Reserved
15:12	R/W	0xF	Reserved
11:8	R/W	0xF	Reserved



Offset: 0x0034			Register Name: PB_CFG1
Bit	Read/Write	Default/Hex	Description
7:4	R/W	0xF	Reserved
3:0	R/W	0xF	Reserved

**9.7.5.3 0x0040 PB Data Register (Default Value: 0x0000\_0000)**

Offset: 0x0040			Register Name: PB_DAT
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0x0	PB_DAT If the port is configured as the input function, the corresponding bit is the pin state. If the port is configured as the output function, the pin state is the same as the corresponding bit. The read bit value is the value set up by software. If the port is configured as a functional pin, the undefined value will be read.

**9.7.5.4 0x0044 PB Multi\_Driving Register 0 (Default Value: 0x1111\_1111)**

Offset: 0x0044			Register Name: PB_DRV0
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:28	R/W	0x1	PB7_DRV PB7 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                     11: Level 3
27:26	/	/	/
25:24	R/W	0x1	PB6_DRV PB6 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                     11: Level 3
23:22	/	/	/
21:20	R/W	0x1	PB5_DRV PB5 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                     11: Level 3

Offset: 0x0044			Register Name: PB_DRV0
Bit	Read/Write	Default/Hex	Description
19:18	/	/	/
17:16	R/W	0x1	PB4_DRV PB4 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
15:14	/	/	/
13:12	R/W	0x1	PB3_DRV PB3 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
11:10	/	/	/
9:8	R/W	0x1	PB2_DRV PB2 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
7:6	/	/	/
5:4	R/W	0x1	Reserved
3:2	/	/	/
1:0	R/W	0x1	Reserved

**9.7.5.5 0x0048 PB Multi\_Driving Register 1 (Default Value: 0x0001\_1111)**

Offset: 0x0048			Register Name: PB_DRV1
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17:16	R/W	0x1	Reserved
15:14	/	/	/
13:12	R/W	0x1	Reserved
11:10	/	/	/
9:8	R/W	0x1	Reserved
7:6	/	/	/
5:4	R/W	0x1	Reserved
3:2	/	/	/
1:0	R/W	0x1	Reserved

9.7.5.6 0x0054 PB Pull Register 0 (Default Value: 0x0000\_0000)

Offset: 0x0054			Register Name: PB_PULL0
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:14	R/W	0x0	PB7_PULL PB7 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
13:12	R/W	0x0	PB6_PULL PB6 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
11:10	R/W	0x0	PB5_PULL PB5 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
9:8	R/W	0x0	PB4_PULL PB4 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
7:6	R/W	0x0	PB3_PULL PB3 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
5:4	R/W	0x0	PB2_PULL PB2 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
3:2	R/W	0x0	Reserved
1:0	R/W	0x0	Reserved

9.7.5.7 0x0060 PC Configure Register 0 (Default Value: 0xFFFF\_FFFF)

Offset: 0x0060			Register Name: PC_CFG0
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0xF	PC7_SELECT PC7 Select 0000:Input                      0001:Output 0010:SPIO-HOLD                0011:SDC2-D3 0100:UART3-RX                0101:TWI3-SDA 0110:TCON-TRIG                0111:Reserved 1000:Reserved                 1001:Reserved 1110:PC-EINT7                 1111:IO Disable
27:24	R/W	0xF	PC6_SELECT PC6 Select 0000:Input                      0001:Output 0010:SPIO-WP                  0011:SDC2-D0 0100:UART3-TX                0101:TWI3-SCK 0110:DBG-CLK                 0111:Reserved 1000:Reserved                 1001:Reserved 1110:PC-EINT6                 1111:IO Disable
23:20	R/W	0xF	PC5_SELECT PC5 Select 0000:Input                      0001:Output 0010:SPIO-MISO                0011:SDC2-D1 0100:BOOT-SEL1                0101:Reserved 0110:Reserved                 0111:Reserved 1000:Reserved                 1001:Reserved 1110:PC-EINT5                 1111:IO Disable
19:16	R/W	0xF	PC4_SELECT PC4 Select. 0000:Input                      0001:Output 0010:SPIO-MOSI                0011:SDC2-D2 0100:BOOT-SELO                0101:Reserved 0110:Reserved                 0111:Reserved 1000:Reserved                 1001:Reserved 1110:PC-EINT4                 1111:IO Disable

Offset: 0x0060			Register Name: PC_CFG0
Bit	Read/Write	Default/Hex	Description
15:12	R/W	0xF	PC3_SELECT PC3 Select 0000:Input                      0001:Output 0010:SPIO-CS0                0011:SDC2-CMD 0100:Reserved                0101:Reserved 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PC-EINT3                1111:IO Disable
11:8	R/W	0xF	PC2_SELECT PC2 Select 0000:Input                      0001:Output 0010:SPIO-CLK                0011:SDC2-CLK 0100:Reserved                0101:Reserved 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PC-EINT2                1111:IO Disable
7:4	R/W	0xF	Reserved
3:0	R/W	0xF	Reserved

**9.7.5.8 0x0070 PC Data Register (Default Value: 0x0000\_0000)**

Offset: 0x0070			Register Name: PC_DAT
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0x0	PC_DAT If the port is configured as the input function, the corresponding bit is the pin state. If the port is configured as the output function, the pin state is the same as the corresponding bit. The read bit value is the value set up by software. If the port is configured as a functional pin, the undefined value will be read.

9.7.5.9 0x0074 PC Multi\_Driving Register 0 (Default Value: 0x1111\_1111)

Offset: 0x0074			Register Name: PC_DRV0
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:28	R/W	0x1	PC7_DRV PC7 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
27:26	/	/	/
25:24	R/W	0x1	PC6_DRV PC6 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
23:22	/	/	/
21:20	R/W	0x1	PC5_DRV PC5 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
19:18	/	/	/
17:16	R/W	0x1	PC4_DRV PC4 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
15:14	/	/	/
13:12	R/W	0x1	PC3_DRV PC3 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
11:10	/	/	/
9:8	R/W	0x1	PC2_DRV PC2 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
7:6	/	/	/
5:4	R/W	0x1	Reserved
3:2	/	/	/
1:0	R/W	0x1	Reserved

9.7.5.10 0x0084 PC Pull Register 0 (Default Value: 0x0000\_0540)

Offset: 0x0084			Register Name: PC_PULL0
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:14	R/W	0x0	PC7_PULL PC7 Pull_up/down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
13:12	R/W	0x0	PC6_PULL PC6 Pull_up/down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
11:10	R/W	0x1	PC5_PULL PC5 Pull_up/down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
9:8	R/W	0x1	PC4_PULL PC4 Pull_up/down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
7:6	R/W	0x1	PC3_PULL PC3 Pull_up/down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
5:4	R/W	0x0	PC2_PULL PC2 Pull_up/down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
3:2	R/W	0x0	Reserved
1:0	R/W	0x0	Reserved

9.7.5.11 0x0090 PD Configure Register 0 (Default Value: 0xFFFF\_FFFF)

Offset: 0x0090			Register Name: PD_CFG0
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0xF	PD7_SELECT PD7 Select 0000:Input                    0001:Output 0010:LCD0-D11                0011:LVDS0-CKN 0100:DSI-D2N                 0101:UART4-TX 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PD-EINT7                1111:IO Disable
27:24	R/W	0xF	PD6_SELECT PD6 Select 0000:Input                    0001:Output 0010:LCD0-D10                0011:LVDS0-CKP 0100:DSI-D2P                 0101:UART5-RX 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PD-EINT6                1111:IO Disable
23:20	R/W	0xF	PD5_SELECT PD5 Select 0000:Input                    0001:Output 0010:LCD0-D7                 0011:LVDS0-V2N 0100:DSI-CKN                 0101:UART5-TX 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PD-EINT5                1111:IO Disable
19:16	R/W	0xF	PD4_SELECT PD4 Select 0000:Input                    0001:Output 0010:LCD0-D6                 0011:LVDS0-V2P 0100:DSI-CKP                 0101:UART2-CTS 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PD-EINT4                1111:IO Disable



Offset: 0x0090			Register Name: PD_CFG0
Bit	Read/Write	Default/Hex	Description
15:12	R/W	0xF	PD3_SELECT PD3 Select 0000:Input                      0001:Output 0010:LCD0-D5                    0011:LVDS0-V1N 0100:DSI-D1N                    0101:UART2-RTS 0110:Reserved                    0111:Reserved 1000:Reserved                    1001:Reserved 1110:PD-EINT3                    1111:IO Disable
11:8	R/W	0xF	PD2_SELECT PD2 Select 0000:Input                      0001:Output 0010:LCD0-D4                    0011:LVDS0-V1P 0100:DSI-D1P                    0101:UART2-RX 0110:Reserved                    0111:Reserved 1000:Reserved                    1001:Reserved 1110:PD-EINT2                    1111:IO Disable
7:4	R/W	0xF	PD1_SELECT PD1 Select 0000:Input                      0001:Output 0010:LCD0-D3                    0011:LVDS0-V0N 0100:DSI-D0N                    0101:UART2-TX 0110:Reserved                    0111:Reserved 1000:Reserved                    1001:Reserved 1110:PD-EINT1                    1111:IO Disable
3:0	R/W	0xF	PD0_SELECT PD0 Select 0000:Input                      0001:Output 0010:LCD0-D2                    0011:LVDS0-V0P 0100:DSI-D0P                    0101:TWIO-SCK 0110:Reserved                    0111:Reserved 1000:Reserved                    1001:Reserved 1110:PD-EINT0                    1111:IO Disable

9.7.5.12 0x0094 PD Configure Register 1 (Default Value: 0xFFFF\_FFFF)

Offset: 0x0094			Register Name: PD_CFG1
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0xF	PD15_SELECT PD15 Select 0000:Input                    0001:Output 0010:LCD0-D21                0011:LVDS1-V2N 0100:SPI1-WP/DBI-TE        0101:IR-RX 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PD-EINT15               1111:IO Disable
27:24	R/W	0xF	PD14_SELECT PD14 Select 0000:Input                    0001:Output 0010:LCD0-D20                0011:LVDS1-V2P 0100:SPI1-HOLD/DBI-DCX/DBI-WRX 0101:UART3-CTS 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PD-EINT14               1111:IO Disable
23:20	R/W	0xF	PD13_SELECT PD13 Select 0000:Input                    0001:Output 0010:LCD0-D19                0011:LVDS1-V1N 0100:SPI1-MISO/DBI-SDI/DBI-TE/DBI-DCX 0101:UART3-RTS 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PD-EINT13               1111:IO Disable
19:16	R/W	0xF	PD12_SELECT PD12 Select 0000:Input                    0001:Output 0010:LCD0-D18                0011:LVDS1-V1P 0100:SPI1-MOSI/DBI-SDO      0101:TWI0-SDA 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PD-EINT12               1111:IO Disable

Offset: 0x0094			Register Name: PD_CFG1
Bit	Read/Write	Default/Hex	Description
15:12	R/W	0xF	PD11_SELECT PD11 Select 0000:Input                      0001:Output 0010:LCD0-D15                0011:LVDS1-V0N 0100:SPI1-CLK/DBI-SCLK    0101:UART3-RX 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PD-EINT11               1111:IO Disable
11:8	R/W	0xF	PD10_SELECT PD10 Select 0000:Input                      0001:Output 0010:LCD0-D14                0011:LVDS1-V0P 0100:SPI1-CS/DBI-CSX       0101:UART3-TX 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PD-EINT10               1111:IO Disable
7:4	R/W	0xF	PD9_SELECT PD9 Select 0000:Input                      0001:Output 0010:LCD0-D13                0011:LVDS0-V3N 0100:DSI-D3N                 0101:PWM6 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PD-EINT9                1111:IO Disable
3:0	R/W	0xF	PD8_SELECT PD8 Select 0000:Input                      0001:Output 0010:LCD0-D12                0011:LVDS0-V3P 0100:DSI-D3P                 0101:UART4-RX 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PD-EINT8                1111:IO Disable

9.7.5.13 0x0098 PD Configure Register 2 (Default Value: 0x0FFF\_FFFF)

Offset: 0x0098			Register Name: PD_CFG2
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:24	R/W	0xF	PD22_SELECT PD22 Select 0000:Input                      0001:Output 0010:OWA-OUT                  0011:IR-RX 0100:UART1-RX                0101:PWM7 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PD-EINT22               1111:IO Disable
23:20	R/W	0xF	PD21_SELECT PD21 Select 0000:Input                      0001:Output 0010:LCD0-VSYNC              0011:TWI2-SDA 0100:UART1-TX                0101:PWM5 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PD-EINT21               1111:IO Disable
19:16	R/W	0xF	PD20_SELECT PD20 Select 0000:Input                      0001:Output 0010:LCD0-HSYNC              0011:TWI2-SCK 0100:DMIC-CLK                0101:PWM4 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PD-EINT20               1111:IO Disable
15:12	R/W	0xF	PD19_SELECT PD19 Select 0000:Input                      0001:Output 0010:LCD0-DE                  0011:LVDS1-V3N 0100:DMIC-DATA0              0101:PWM3 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PD-EINT19               1111:IO Disable

Offset: 0x0098			Register Name: PD_CFG2
Bit	Read/Write	Default/Hex	Description
11:8	R/W	0xF	PD18_SELECT PD18 Select 0000:Input                      0001:Output 0010:LCD0-CLK                0011:LVDS1-V3P 0100:DMIC-DATA1            0101:PWM2 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PD-EINT18              1111:IO Disable
7:4	R/W	0xF	PD17_SELECT PD17 Select. 0000:Input                      0001:Output 0010:LCD0-D23                0011:LVDS1-CKN 0100:DMIC-DATA2            0101:PWM1 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PD-EINT17              1111:IO Disable
3:0	R/W	0xF	PD16_SELECT PD16 Select 0000:Input                      0001:Output 0010:LCD0-D22                0011:LVDS1-CKP 0100:DMIC-DATA3            0101:PWM0 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PD-EINT16              1111:IO Disable

9.7.5.14 0x00A0 PD Data Register (Default Value: 0x0000\_0000)

Offset: 0x00A0			Register Name: PD_DAT
Bit	Read/Write	Default/Hex	Description
31:23	/	/	/

Offset: 0x00A0			Register Name: PD_DAT
Bit	Read/Write	Default/Hex	Description
22:0	R/W	0x0	PD_DAT PD Data If the port is configured as the input function, the corresponding bit is the pin state. If the port is configured as the output function, the pin state is the same as the corresponding bit. The read bit value is the value set up by software. If the port is configured as a functional pin, the undefined value will be read.

9.7.5.15 0x00A4 PD Multi\_Driving Register 0 (Default Value: 0x1111\_1111)

Offset: 0x00A4			Register Name: PD_DRV0
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:28	R/W	0x1	PD7_DRV PD7 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                     11: Level 3
27:26	/	/	/
25:24	R/W	0x1	PD6_DRV PD6 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                     11: Level 3
23:22	/	/	/
21:20	R/W	0x1	PD5_DRV PD5 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                     11: Level 3
19:18	/	/	/
17:16	R/W	0x1	PD4_DRV PD4 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                     11: Level 3
15:14	/	/	/

Offset: 0x00A4			Register Name: PD_DRV0
Bit	Read/Write	Default/Hex	Description
13:12	R/W	0x1	PD3_DRV PD3 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
11:10	/	/	/
9:8	R/W	0x1	PD2_DRV PD2 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
7:6	/	/	/
5:4	R/W	0x1	PD1_DRV PD1 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
3:2	/	/	/
1:0	R/W	0x1	PD0_DRV PD0 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3

9.7.5.16 0x00A8 PD Multi\_Driving Register 1 (Default Value: 0x1111\_1111)

Offset: 0x00A8			Register Name: PD_DRV1
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:28	R/W	0x1	PD15_DRV PD15 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
27:26	/	/	/
25:24	R/W	0x1	PD14_DRV PD14 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
23:22	/	/	/

Offset: 0x00A8			Register Name: PD_DRV1
Bit	Read/Write	Default/Hex	Description
21:20	R/W	0x1	PD13_DRV PD13 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
19:18	/	/	/
17:16	R/W	0x1	PD12_DRV PD12 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
15:14	/	/	/
13:12	R/W	0x1	PD11_DRV PD11 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
11:10	/	/	/
9:8	R/W	0x1	PD10_DRV PD10 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
7:6	/	/	/
5:4	R/W	0x1	PD9_DRV PD9 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
3:2	/	/	/
1:0	R/W	0x1	PD8_DRV PD8 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3

**9.7.5.17 0x00AC PD Multi\_Driving Register 2 (Default Value: 0x0111\_1111)**

Offset: 0x00AC			Register Name: PD_DRV2
Bit	Read/Write	Default/Hex	Description
31:26	/	/	/



Offset: 0x00AC			Register Name: PD_DRV2
Bit	Read/Write	Default/Hex	Description
25:24	R/W	0x1	PD22_DRV PD22 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
23:22	/	/	/
21:20	R/W	0x1	PD21_DRV PD21 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
19:18	/	/	/
17:16	R/W	0x1	PD20_DRV PD20 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
15:14	/	/	/
13:12	R/W	0x1	PD19_DRV PD19 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
11:10	/	/	/
9:8	R/W	0x1	PD18_DRV PD18 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
7:6	/	/	/
5:4	R/W	0x1	PD17_DRV PD17 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
3:2	/	/	/
1:0	R/W	0x1	PD16_DRV PD16 Multi_Driving Select. 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3

9.7.5.18 0x00B4 PD Pull Register 0 (Default Value: 0x0000\_0000)

Offset: 0x00B4			Register Name: PD_PULL0
Bit	Read/Write	Default/Hex	Description
31:30	R/W	0x0	PD15_PULL PD15 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
29:28	R/W	0x0	PD14_PULL PD14 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
27:26	R/W	0x0	PD13_PULL PD13 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
25:24	R/W	0x0	PD12_PULL PD12 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
23:22	R/W	0x0	PD11_PULL PD11 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
21:20	R/W	0x0	PD10_PULL PD10 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
19:18	R/W	0x0	PD9_PULL PD9 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
17:16	R/W	0x0	PD8_PULL PD8 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved

Offset: 0x00B4			Register Name: PD_PULL0
Bit	Read/Write	Default/Hex	Description
15:14	R/W	0x0	PD7_PULL PD7 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
13:12	R/W	0x0	PD6_PULL PD6 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
11:10	R/W	0x0	PD5_PULL PD5 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
9:8	R/W	0x0	PD4_PULL PD4 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
7:6	R/W	0x0	PD3_PULL PD3 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
5:4	R/W	0x0	PD2_PULL PD2 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
3:2	R/W	0x0	PD1_PULL PD1 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
1:0	R/W	0x0	PD0_PULL PD0 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved

9.7.5.19 0x00B8 PD Pull Register 1 (Default Value: 0x0000\_0000)

Offset: 0x00B8			Register Name: PD_PULL1
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13:12	R/W	0x0	PD22_PULL PD22 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
11:10	R/W	0x0	PD21_PULL PD21 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
9:8	R/W	0x0	PD20_PULL PD20 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
7:6	R/W	0x0	PD19_PULL PD19 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
5:4	R/W	0x0	PD18_PULL PD18 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
3:2	R/W	0x0	PD17_PULL PD17 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
1:0	R/W	0x0	PD16_PULL PD16 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved

9.7.5.20 0x00C0 PE Configure Register 0 (Default Value: 0xFFFF\_FFFF)

Offset: 0x00C0			Register Name: PE_CFG0
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0xF	PE7_SELECT PE7 Select 0000:Input                      0001:Output 0010:NCSI0-D3                      0011:UART5-RX 0100:TWI3-SDA                      0101:OWA-OUT 0110:Reserved                      0111:R-JTAG-CK 1000:RGMII-CLKIN/RMII-RXER    1001:Reserved 1110:PE-EINT7                      1111:IO Disable
27:24	R/W	0xF	PE6_SELECT PE6 Select 0000:Input                      0001:Output 0010:NCSI0-D2                      0011:UART5-TX 0100:TWI3-SCK                      0101:OWA-IN 0110:Reserved                      0111:R-JTAG-DO 1000:RMII-TXCTRL/RMII-TXEN    1001:Reserved 1110:PE-EINT6                      1111:IO Disable
23:20	R/W	0xF	PE5_SELECT PE5 Select 0000:Input                      0001:Output 0010:NCSI0-D1                      0011:UART4-RX 0100:TWI2-SDA                      0101:LEDC-DO 0110:Reserved                      0111:R-JTAG-DI 1000:RGMII-TXD1/RMII-TXD1    1001:Reserved 1110:PE-EINT5                      1111:IO Disable
19:16	R/W	0xF	PE4_SELECT PE4 Select 0000:Input                      0001:Output 0010:NCSI0-D0                      0011:UART4-TX 0100:TWI2-SCK                      0101:CLK-FANOUT2 0110:Reserved                      0111:R-JTAG-MS 1000:RGMII-TXD0/RMII-TXD0    1001:Reserved 1110:PE-EINT4                      1111:IO Disable

Offset: 0x00C0			Register Name: PE_CFG0
Bit	Read/Write	Default/Hex	Description
15:12	R/W	0xF	PE3_SELECT PE3 Select 0000:Input                      0001:Output 0010:NCSI0-MCLK              0011:UART2-RX 0100:TWI0-SDA                0101:CLK-FANOUT1 0110:UART0-RX                0111:Reserved 1000:RGMII-TXCK/RMII-TXCK 1001:Reserved 1110:PE-EINT3                1111:IO Disable
11:8	R/W	0xF	PE2_SELECT PE2 Select 0000:Input                      0001:Output 0010:NCSI0-PCLK              0011:UART2-TX 0100:TWI0-SCK                0101:CLK-FANOUT0 0110:UART0-TX                0111:Reserved 1000:RGMII-RXD1/RMII-RXD1 1001:Reserved 1110:PE-EINT2                1111:IO Disable
7:4	R/W	0xF	PE1_SELECT PE1 Select 0000:Input                      0001:Output 0010:NCSI0-VSYNC            0011:UART2-CTS 0100:TWI1-SDA                0101:LCD0-VSYNC 0110:Reserved                 0111:Reserved 1000:RGMII-RXD0/RMII-RXD0 1001:Reserved 1110:PE-EINT1                1111:IO Disable
3:0	R/W	0xF	PE0_SELECT PE0 Select 0000:Input                      0001:Output 0010:NCSI0-HSYNC            0011:UART2-RTS 0100:TWI1-SCK                0101:LCD0-HSYNC 0110:Reserved                 0111:Reserved 1000:RGMII-RXCTRL/RMII-CRS-DV 1001:Reserved 1110:PE-EINT0                1111:IO Disable

9.7.5.21 0x00C4 PE Configure Register 1 (Default Value: 0xFFFF\_FFFF)

Offset: 0x00C4			Register Name: PE_CFG1
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0xF	Reserved
27:24	R/W	0xF	Reserved
23:20	R/W	0xF	PE13_SELECT PE13 Select 0000:Input                      0001:Output 0010:TWI2-SDA                0011:PWM5 0100:Reserved                0101:Reserved 0110:DMIC-DATA3            0111:Reserved 1000:RGMII-RXD2            1001:Reserved 1110:PE-EINT13              1111:IO Disable
19:16	R/W	0xF	PE12_SELECT PE12 Select 0000:Input                      0001:Output 0010:TWI2-SCK               0011:NCSI0-FIELD 0100:Reserved                0101:Reserved 0110:Reserved                0111:Reserved 1000:RGMII-TXD3            1001:Reserved 1110:PE-EINT12              1111:IO Disable
15:12	R/W	0xF	PE11_SELECT PE11 Select 0000:Input                      0001:Output 0010:NCSI0-D7                0011:UART1-RX 0100:Reserved                0101:Reserved 0110:Reserved                0111:Reserved 1000:RGMII-TXD2            1001:Reserved 1110:PE-EINT11              1111:IO Disable
11:8	R/W	0xF	PE10_SELECT PE10 Select 0000:Input                      0001:Output 0010:NCSI0-D6                0011:UART1-TX 0100:PWM4                      0101:IR-RX 0110:Reserved                0111:Reserved 1000:EPHY-25M               1001:Reserved 1110:PE-EINT10              1111:IO Disable

Offset: 0x00C4			Register Name: PE_CFG1
Bit	Read/Write	Default/Hex	Description
7:4	R/W	0xF	PE9_SELECT PE9 Select 0000:Input                      0001:Output 0010:NCSI0-D5                  0011:UART1-CTS 0100:PWM3                      0101:UART3-RX 0110:Reserved                  0111:Reserved 1000:MDIO                      1001:Reserved 1110:PE-EINT9                  1111:IO Disable
3:0	R/W	0xF	PE8_SELECT PE8 Select 0000:Input                      0001:Output 0010:NCSI0-D4                  0011:UART1-RTS 0100:PWM_2                      0101:UART3-TX 0110:Reserved                  0111:Reserved 1000:MDC                      1001:Reserved 1110:PE_EINT8                  1111:IO Disable

9.7.5.22 0x00D0 PE Data Register (Default Value: 0x0000\_0000)

Offset: 0x00D0			Register Name: PE_DAT
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13:0	R/W	0x0	PE_DAT PE Data If the port is configured as input, the corresponding bit is the pin state. If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.



9.7.5.23 0x00D4 PE Multi\_Driving Register 0 (Default Value: 0x1111\_1111)

Offset: 0x00D4			Register Name: PE_DRV0
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:28	R/W	0x1	PE7_DRV PE7 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
27:26	/	/	/
25:24	R/W	0x1	PE6_DRV PE6 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
23:22	/	/	/
21:20	R/W	0x1	PE5_DRV PE5 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
19:18	/	/	/
17:16	R/W	0x1	PE4_DRV PE4 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
15:14	/	/	/
13:12	R/W	0x1	PE3_DRV PE3 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
11:10	/	/	/
9:8	R/W	0x1	PE2_DRV PE2 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
7:6	/	/	/

Offset: 0x00D4			Register Name: PE_DRV0
Bit	Read/Write	Default/Hex	Description
5:4	R/W	0x1	PE1_DRV PE1 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
3:2	/	/	/
1:0	R/W	0x1	PE0_DRV PE0 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3

9.7.5.24 0x00D8 PE Multi\_Driving Register 1 (Default Value: 0x1111\_1111)

Offset: 0x00D8			Register Name: PE_DRV1
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:28	R/W	0x1	Reserved
27:26	/	/	/
25:24	R/W	0x1	Reserved
23:22	/	/	/
21:20	R/W	0x1	PE13_DRV PE13 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
19:18	/	/	/
17:16	R/W	0x1	PE12_DRV PE12 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
15:14	/	/	/
13:12	R/W	0x1	PE11_DRV PE11 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
11:10	/	/	/

Offset: 0x00D8			Register Name: PE_DRV1
Bit	Read/Write	Default/Hex	Description
9:8	R/W	0x1	PE10_DRV PE10 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
7:6	/	/	/
5:4	R/W	0x1	PE9_DRV PE9 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
3:2	/	/	/
1:0	R/W	0x1	PE8_DRV PE8 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3

9.7.5.25 0x00E4 PE Pull Register 0 (Default Value: 0x0000\_0000)

Offset: 0x00E4			Register Name: PE_PULL0
Bit	Read/Write	Default/Hex	Description
31:30	R/W	0x0	Reserved
29:28	R/W	0x0	Reserved
27:26	R/W	0x0	PE13_PULL PE13 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                      11: Reserved
25:24	R/W	0x0	PE12_PULL PE12 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                      11: Reserved
23:22	R/W	0x0	PE11_PULL PE11 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                      11: Reserved

Offset: 0x00E4			Register Name: PE_PULL0
Bit	Read/Write	Default/Hex	Description
21:20	R/W	0x0	PE10_PULL PE10 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
19:18	R/W	0x0	PE9_PULL PE9 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
17:16	R/W	0x0	PE8_PULL PE8 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
15:14	R/W	0x0	PE7_PULL PE7 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
13:12	R/W	0x0	PE6_PULL PE6 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
11:10	R/W	0x0	PE5_PULL PE5 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
9:8	R/W	0x0	PE4_PULL PE4 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
7:6	R/W	0x0	PE3_PULL PE3 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
5:4	R/W	0x0	PE2_PULL PE2 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved

Offset: 0x00E4			Register Name: PE_PULL0
Bit	Read/Write	Default/Hex	Description
3:2	R/W	0x0	PE1_PULL PE1 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
1:0	R/W	0x0	PE0_PULL PE0 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved

9.7.5.26 0x00F0 PF Configure Register 0 (Default Value: 0x0FFF\_FFFF)

Offset: 0x00F0			Register Name: PF_CFG0
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:24	R/W	0xF	PF6_SELECT PF6 Select 0000:Input                    0001:Output 0010:Reserved                0011:OWA-OUT 0100:IR-RX                    0101:I2S2-MCLK 0110:PWM5                    0111:Reserved 1000:Reserved                1001:Reserved 1110:PF-EINT6                1111:IO Disable
23:20	R/W	0xF	PF5_SELECT PF5 Select 0000:Input                    0001:Output 0010:SDC0-D2                0011:Reserved 0100:R-JTAG-CK               0101:I2S2-LRCK 0110:Reserved                0111:Reserved 1000:Reserved                1001:Reserved 1110:PF-EINT5                1111:IO Disable

Offset: 0x00F0			Register Name: PF_CFG0
Bit	Read/Write	Default/Hex	Description
19:16	R/W	0xF	PF4_SELECT PF4 Select 0000:Input                      0001:Output 0010:SDC0-D3                      0011:UART0-RX 0100:TWI0-SDA                      0101:PWM6 0110:IR-TX                      0111:Reserved 1000:Reserved                      1001:Reserved 1110:PF-EINT4                      1111:IO Disable
15:12	R/W	0xF	PF3_SELECT PF3 Select 0000:Input                      0001:Output 0010:SDC0-CMD                      0011:Reserved 0100:R-JTAG-DO                      0101:I2S2-BCLK 0110:Reserved                      0111:Reserved 1000:Reserved                      1001:Reserved 1110:PF-EINT3                      1111:IO Disable
11:8	R/W	0xF	PF2_SELECT PF2 Select 0000:Input                      0001:Output 0010:SDC0-CLK                      0011:UART0-TX 0100:TWI0-SCK                      0101:LEDC-DO 0110:OWA-IN                      0111:Reserved 1000:Reserved                      1001:Reserved 1110:PF-EINT2                      1111:IO Disable
7:4	R/W	0xF	PF1_SELECT PF1 Select 0000:Input                      0001:Output 0010:SDC0-D0                      0011:Reserved 0100:R-JTAG-DI                      0101:I2S2-DOUT0 0110:I2S2-DIN1                      0111:Reserved 1000:Reserved                      1001:Reserved 1110:PF-EINT1                      1111:IO Disable

Offset: 0x00F0			Register Name: PF_CFG0
Bit	Read/Write	Default/Hex	Description
3:0	R/W	0xF	PF0_SELECT PF0 Select 0000:Input                      0001:Output 0010:SDC0-D1                  0011:Reserved 0100:R-JTAG-MS                0101:I2S2-DOUT1 0110:I2S2-DINO                0111:Reserved 1000:Reserved                  1001:Reserved 1110:PF-EINT0                 1111:IO Disable

9.7.5.27 0x0100 PF Data Register (Default Value: 0x0000\_0000)

Offset: 0x0100			Register Name: PF_DAT
Bit	Read/Write	Default/Hex	Description
31:7	/	/	/
6:0	R/W	0	PF_DAT PF Data If the port is configured as input, the corresponding bit is the pin state. If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.

9.7.5.28 0x0104 PF Multi\_Driving Register 0 (Default Value: 0x0111\_1111)

Offset: 0x0104			Register Name: PF_DRV0
Bit	Read/Write	Default/Hex	Description
31:26	/	/	/
25:24	R/W	0x1	PF6_DRV PF6 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
23:22	/	/	/

Offset: 0x0104			Register Name: PF_DRV0
Bit	Read/Write	Default/Hex	Description
21:20	R/W	0x1	PF5_DRV PF5 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
19:18	/	/	/
17:16	R/W	0x1	PF4_DRV PF4 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
15:14	/	/	/
13:12	R/W	0x1	PF3_DRV PF3 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
11:10	/	/	/
9:8	R/W	0x1	PF2_DRV PF2 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
7:6	/	/	/
5:4	R/W	0x1	PF1_DRV PF1 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
3:2	/	/	/
1:0	R/W	0x1	PF0_DRV PF0 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3

9.7.5.29 0x0114 PF Pull Register 0 (Default Value: 0x0000\_0000)

Offset: 0x0114			Register Name: PF_PULL0
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/



Offset: 0x0114			Register Name: PF_PULL0
Bit	Read/Write	Default/Hex	Description
13:12	R/W	0x0	PF6_PULL PF6 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
11:10	R/W	0x0	PF5_PULL PF5 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
9:8	R/W	0x0	PF4_PULL PF4 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
7:6	R/W	0x0	PF3_PULL PF3 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
5:4	R/W	0x0	PF2_PULL PF2 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
3:2	R/W	0x0	PF1_PULL PF1 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
1:0	R/W	0x0	PF0_PULL PF0 Pull_up or down Select 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved

9.7.5.30 0x0120 PG Configure Register 0 (Default Value: 0xFFFF\_FFFF)

Offset: 0x0120			Register Name: PG_CFG0
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0xF	PG7_SELECT PG7 Select 0000:Input                    0001:Output 0010:UART1-RX            0011:TWI2-SDA 0100:RGMII-TXD3        0101:OWA-IN 0110:Reserved            0111:Reserved 1000:Reserved            1001:Reserved 1110:PG-EINT7            1111:IO Disable
27:24	R/W	0xF	PG6_SELECT PG6 Select 0000:Input                    0001:Output 0010:UART1-TX            0011:TWI2-SCK 0100:RGMII-TXD2        0101:PWM1 0110:Reserved            0111:Reserved 1000:Reserved            1001:Reserved 1110:PG-EINT6            1111:IO Disable
23:20	R/W	0xF	PG5_SELECT PG5 Select 0000:Input                    0001:Output 0010:SDC1-D3            0011:UART5-RX 0100:RGMII-TXD1/RMII-TXD1 0101:PWM4 0110:Reserved            0111:Reserved 1000:Reserved            1001:Reserved 1110:PG-EINT5            1111:IO Disable
19:16	R/W	0xF	PG4_SELECT PG4 Select 0000:Input                    0001:Output 0010:SDC1-D2            0011:UART5-TX 0100:RGMII-TXD0/RMII-TXD0 0101:PWM5 0110:Reserved            0111:Reserved 1000:Reserved            1001:Reserved 1110:PG-EINT4            1111:IO Disable

Offset: 0x0120			Register Name: PG_CFG0
Bit	Read/Write	Default/Hex	Description
15:12	R/W	0xF	PG3_SELECT PG3 Select 0000:Input                      0001:Output 0010:SDC1-D1                    0011:UART3-CTS 0100:RGMII-TXCK/RMII-TXCK    0101:UART4-RX 0110:Reserved                    0111:Reserved 1000:Reserved                    1001:Reserved 1110:PG-EINT3                    1111:IO Disable
11:8	R/W	0xF	PG2_SELECT PG2 Select 0000:Input                      0001:Output 0010:SDC1-D0                    0011:UART3-RTS 0100:RGMII-RXD1/RMII-RXD1    0101:UART4-TX 0110:Reserved                    0111:Reserved 1000:Reserved                    1001:Reserved 1110:PG-EINT2                    1111:IO Disable
7:4	R/W	0xF	PG1_SELECT PG1 Select 0000:Input                      0001:Output 0010:SDC1-CMD                    0011:UART3-RX 0100:RGMII-RXD0/RMII-RXD0    0101:PWM6 0110:Reserved                    0111:Reserved 1000:Reserved                    1001:Reserved 1110:PG-EINT1                    1111:IO Disable
3:0	R/W	0xF	PG0_SELECT PG0 Select 0000:Input                      0001:Output 0010:SDC1-CLK                    0011:UART3-TX 0100:RGMII-RXCTRL/RMII-CRS-DV 0101:PWM7 0110:Reserved                    0111:Reserved 1000:Reserved                    1001:Reserved 1110:PG-EINT0                    1111:IO Disable

9.7.5.31 0x0124 PG Configure Register 1 (Default Value: 0xFFFF\_FFFF)

Offset: 0x0124			Register Name: PG_CFG1
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0xF	PG15_SELECT PG15 Select 0000:Input                    0001:Output 0010:I2S1-DOUT0            0011:TWI2-SDA 0100:MDIO                    0101:I2S1-DIN1 0110:SPIO-HOLD            0111:UART1-CTS 1000:Reserved                1001:Reserved 1110:PG-EINT15              1111:IO Disable
27:24	R/W	0xF	PG14_SELECT PG14 Select 0000:Input                    0001:Output 0010:I2S1-DIN0            0011:TWI2-SCK 0100:MDC                      0101:I2S1-DOUT1 0110:SPIO-WP                0111:UART1-RTS 1000:Reserved                1001:Reserved 1110:PG-EINT14              1111:IO Disable
23:20	R/W	0xF	PG13_SELECT PG13 Select. 0000:Input                    0001:Output 0010:I2S1-BCLK              0011:TWI0-SDA 0100:RGMII-CLKIN/RMII-RXER 0101:PWM2 0110:LEDC-DO                0111:UART1_RX 1000:Reserved                1001:Reserved 1110:PG-EINT13              1111:IO Disable
19:16	R/W	0xF	PG12_SELECT PG12 Select 0000:Input                    0001:Output 0010:I2S1-LRCK              0011:TWI0-SCK 0100:RGMII-TXCTRL/RMII-TXEN 0101:CLK-FANOUT2 0110:PWM0                    0111:UART1-TX 1000:Reserved                1001:Reserved 1110:PG-EINT12              1111:IO Disable

Offset: 0x0124			Register Name: PG_CFG1
Bit	Read/Write	Default/Hex	Description
15:12	R/W	0xF	PG11_SELECT PG11 Select 0000:Input                      0001:Output 0010:I2S1-MCLK                0011:TWI3-SDA 0100:EPHY-25M                0101:CLK-FANOUT1 0110:TCON-TRIG               0111:Reserved 1000:Reserved                 1001:Reserved 1110:PG-EINT11                1111:IO Disable
11:8	R/W	0xF	PG10_SELECT PG10 Select 0000:Input                      0001:Output 0010:PWM3                      0011:TWI3-SCK 0100:RGMII-RXCK               0101:CLK-FANOUT0 0110:IR-RX                      0111:Reserved 1000:Reserved                 1001:Reserved 1110:PG-EINT10                1111:IO Disable
7:4	R/W	0xF	PG9_SELECT PG9 Select. 0000:Input                      0001:Output 0010:UART1-CTS                0011:TWI1-SDA 0100:RGMII-RXD3               0101:UART3-RX 0110:Reserved                 0111:Reserved 1000:Reserved                 1001:Reserved 1110:PG-EINT9                 1111:IO Disable
3:0	R/W	0xF	PG8_SELECT PG8 Select 0000:Input                      0001:Output 0010:UART1-RTS                0011:TWI1-SCK 0100:RGMII-RXD2               0101:UART3-TX 0110:Reserved                 0111:Reserved 1000:Reserved                 1001:Reserved 1110:PG-EINT8                 1111:IO Disable

9.7.5.32 0x0130 PG Data Register (Default Value: 0x0000\_0000)

Offset: 0x0130			Register Name: PG_DAT
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x0	PG_DAT If the port is configured as the input function, the corresponding bit is the pin state. If the port is configured as the output function, the pin state is the same as the corresponding bit. The read bit value is the value set up by software. If the port is configured as a functional pin, the undefined value will be read.

9.7.5.33 0x0134 PG Multi\_Driving Register 0 (Default Value: 0x1111\_1111)

Offset: 0x0134			Register Name: PG_DRV0
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:28	R/W	0x1	PG7_DRV PG7 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
27:26	/	/	/
25:24	R/W	0x1	PG6_DRV PG6 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
23:22	/	/	/
21:20	R/W	0x1	PG5_DRV PG5 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
19:18	/	/	/
17:16	R/W	0x1	PG4_DRV PG4 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
15:14	/	/	/

Offset: 0x0134			Register Name: PG_DRV0
Bit	Read/Write	Default/Hex	Description
13:12	R/W	0x1	PG3_DRV PG3 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
11:10	/	/	/
9:8	R/W	0x1	PG2_DRV PG2 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
7:6	/	/	/
5:4	R/W	0x1	PG1_DRV PG1 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
3:2	/	/	/
1:0	R/W	0x1	PG0_DRV PG0 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3

9.7.5.34 0x0138 PG Multi\_Driving Register 1 (Default Value: 0x1111\_1111)

Offset: 0x0138			Register Name: PG_DRV1
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:28	R/W	0x1	PG15_DRV PG15 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
27:26	/	/	/
25:24	R/W	0x1	PG14_DRV PG14 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
23:22	/	/	/

Offset: 0x0138			Register Name: PG_DRV1
Bit	Read/Write	Default/Hex	Description
21:20	R/W	0x1	PG13_DRV PG13 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
19:18	/	/	/
17:16	R/W	0x1	PG12_DRV PG12 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
15:14	/	/	/
13:12	R/W	0x1	PG11_DRV PG11 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
11:10	/	/	/
9:8	R/W	0x1	PG10_DRV PG10 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
7:6	/	/	/
5:4	R/W	0x1	PG9_DRV PG9 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3
3:2	/	/	/
1:0	R/W	0x1	PG8_DRV PG8 Multi_Driving Select 00: Level 0                      01: Level 1 10: Level 2                      11: Level 3



9.7.5.35 0x0144 PG Pull Register 0 (Default Value: 0x0000\_0000)

Offset: 0x0144			Register Name: PG_PULL0
Bit	Read/Write	Default/Hex	Description
31:30	R/W	0x0	PG15_PULL PG15 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
29:28	R/W	0x0	PG14_PULL PG14 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
27:26	R/W	0x0	PG13_PULL PG13 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
25:24	R/W	0x0	PG12_PULL PG12 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
23:22	R/W	0x0	PG11_PULL PG11 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
21:20	R/W	0x0	PG10_PULL PG10 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
19:18	R/W	0x0	PG9_PULL PG9 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
17:16	R/W	0x0	PG8_PULL PG8 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved

Offset: 0x0144			Register Name: PG_PULL0
Bit	Read/Write	Default/Hex	Description
15:14	R/W	0x0	PG7_PULL PG7 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
13:12	R/W	0x0	PG6_PULL PG6 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
11:10	R/W	0x0	PG5_PULL PG5 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
9:8	R/W	0x0	PG4_PULL PG4 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
7:6	R/W	0x0	PG3_PULL PG3 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
5:4	R/W	0x0	PG2_PULL PG2 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
3:2	R/W	0x0	PG1_PULL PG1 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved
1:0	R/W	0x0	PG0_PULL PG0 Pull_up or down Select. 00: Pull_up/down disable    01: Pull_up 10: Pull_down                    11: Reserved

9.7.5.36 0x0220 PB External Interrupt Configure Register 0 (Default Value: 0x0000\_0000)

Offset: 0x0220			Register Name:PB_EINT_CFG0
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0x0	EINT7_CFG External INT7 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
27:24	R/W	0x0	EINT6_CFG External INT6 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
23:20	R/W	0x0	EINT5_CFG External INT5 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
19:16	R/W	0x0	EINT4_CFG External INT4 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Offset: 0x0220			Register Name: PB_EINT_CFG0
Bit	Read/Write	Default/Hex	Description
15:12	R/W	0x0	EINT3_CFG External INT3 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
11:8	R/W	0x0	EINT2_CFG External INT2 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
7:4	R/W	0x0	Reserved
3:0	R/W	0x0	Reserved

9.7.5.37 0x0230 PB External Interrupt Control Register (Default Value: 0x0000\_0000)

Offset: 0x0230			Register Name: PB_EINT_CTL
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	EINT7_CTL External INT7 Enable 0: Disable 1: Enable
6	R/W	0x0	EINT6_CTL External INT6 Enable 0: Disable 1: Enable

Offset: 0x0230			Register Name: PB_EINT_CTL
Bit	Read/Write	Default/Hex	Description
5	R/W	0x0	EINT5_CTL External INT5 Enable 0: Disable 1: Enable
4	R/W	0x0	EINT4_CTL External INT4 Enable 0: Disable 1: Enable
3	R/W	0x0	EINT3_CTL External INT3 Enable 0: Disable 1: Enable
2	R/W	0x0	EINT2_CTL External INT2 Enable 0: Disable 1: Enable
1	R/W	0x0	Reserved
0	R/W	0x0	Reserved

9.7.5.38 0x0234 PB External Interrupt Status Register (Default Value: 0x0000\_0000)

Offset: 0x0234			Register Name: PB_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R/W1C	0x0	EINT7_STATUS External INT7 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
6	R/W1C	0x0	EINT6_STATUS External INT6 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Offset: 0x0234			Register Name: PB_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
5	R/W1C	0x0	EINT5_STATUS External INT5 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
4	R/W1C	0x0	EINT4_STATUS External INT4 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
3	R/W1C	0x0	EINT3_STATUS External INT3 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
2	R/W1C	0x0	EINT2_STATUS External INT2 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
1	R/W	0x0	Reserved
0	R/W	0x0	Reserved

9.7.5.39 0x0238 PB External Interrupt Debounce Register (Default Value: 0x0000\_0000)

Offset: 0x0238			Register Name: PB_EINT_DEB
Bit	Read/Write	Default/Hex	Description
31:7	/	/	/
6:4	R/W	0x0	DEB_CLK_PRE_SCALE Debounce Clock Pre_scale n The selected clock source is prescaled by 2^n.
3:1	/	/	/

Offset: 0x0238			Register Name: PB_EINT_DEB
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	PIO_INT_CLK_SELECT PIO Interrupt Clock Select 0: LOSC 32KHz 1: HOSC 24MHz

9.7.5.40 0x0240 PC External Interrupt Configure Register 0 (Default Value: 0x0000\_0000)

Offset: 0x0240			Register Name: PC_EINT_CFG0
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0x0	EINT7_CFG External INT7 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
27:24	R/W	0x0	EINT6_CFG External INT6 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
23:20	R/W	0x0	EINT5_CFG External INT5 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Offset: 0x0240			Register Name: PC_EINT_CFG0
Bit	Read/Write	Default/Hex	Description
19:16	R/W	0x0	EINT4_CFG External INT4 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
15:12	R/W	0x0	EINT3_CFG External INT3 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
11:8	R/W	0x0	EINT2_CFG External INT2 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
7:4	R/W	0x0	Reserved
3:0	R/W	0x0	Reserved

**9.7.5.41 0x0250 PC External Interrupt Control Register (Default Value: 0x0000\_0000)**

Offset: 0x0250			Register Name: PC_EINT_CTL
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/



Offset: 0x0250			Register Name: PC_EINT_CTL
Bit	Read/Write	Default/Hex	Description
7	R/W	0x0	EINT7_CTL External INT7 Enable 0: Disable 1: Enable
6	R/W	0x0	EINT6_CTL External INT6 Enable 0: Disable 1: Enable
5	R/W	0x0	EINT5_CTL External INT5 Enable 0: Disable 1: Enable
4	R/W	0x0	EINT4_CTL External INT4 Enable 0: Disable 1: Enable
3	R/W	0x0	EINT3_CTL External INT3 Enable 0: Disable 1: Enable
2	R/W	0x0	EINT2_CTL External INT2 Enable 0: Disable 1: Enable
1	R/W	0x0	Reserved
0	R/W	0x0	Reserved

**9.7.5.42 0x0254 PC External Interrupt Status Register (Default Value: 0x0000\_0000)**

Offset: 0x0254			Register Name: PC_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/

Offset: 0x0254			Register Name: PC_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
7	R/W1C	0x0	EINT7_STATUS External INT7 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
6	R/W1C	0x0	EINT6_STATUS External INT6 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
5	R/W1C	0x0	EINT5_STATUS External INT5 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
4	R/W1C	0x0	EINT4_STATUS External INT4 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
3	R/W1C	0x0	EINT3_STATUS External INT3 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
2	R/W1C	0x0	EINT2_STATUS External INT2 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
1	R/W	0x0	Reserved
0	R/W	0x0	Reserved

9.7.5.43 0x0258 PC External Interrupt Debounce Register (Default Value: 0x0000\_0000)

Offset: 0x0258			Register Name: PC_EINT_DEB
Bit	Read/Write	Default/Hex	Description
31:7	/	/	/
6:4	R/W	0x0	DEB_CLK_PRE_SCALE Debounce Clock Pre_scale n The selected clock source is prescaled by 2^n.
3:1	/	/	/
0	R/W	0x0	PIO_INT_CLK_SELECT PIO Interrupt Clock Select 0: LOSC 32KHz 1: HOSC 24MHz

9.7.5.44 0x0260 PD External Interrupt Configure Register 0 (Default Value: 0x0000\_0000)

Offset: 0x0260			Register Name: PD_EINT_CFG0
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0x0	EINT7_CFG External INT7 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
27:24	R/W	0x0	EINT6_CFG External INT6 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Offset: 0x0260			Register Name:PD_EINT_CFG0
Bit	Read/Write	Default/Hex	Description
23:20	R/W	0x0	EINT5_CFG External INT5 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
19:16	R/W	0x0	EINT4_CFG External INT4 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
15:12	R/W	0x0	EINT3_CFG External INT3 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
11:8	R/W	0x0	EINT2_CFG External INT2 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Offset: 0x0260			Register Name:PD_EINT_CFG0
Bit	Read/Write	Default/Hex	Description
7:4	R/W	0x0	EINT1_CFG External INT1 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
3:0	R/W	0x0	EINT0_CFG External INTO Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

9.7.5.45 0x0264 PD External Interrupt Configure Register 1 (Default Value: 0x0000\_0000)

Offset: 0x0264			Register Name:PD_EINT_CFG1
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0x0	EINT15_CFG External INT15 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Offset: 0x0264			Register Name:PD_EINT_CFG1
Bit	Read/Write	Default/Hex	Description
27:24	R/W	0x0	EINT14_CFG External INT14 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
23:20	R/W	0x0	EINT13_CFG External INT13 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
19:16	R/W	0x0	EINT12_CFG External INT12 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
15:12	R/W	0x0	EINT11_CFG External INT11 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Offset: 0x0264			Register Name:PD_EINT_CFG1
Bit	Read/Write	Default/Hex	Description
11:8	R/W	0x0	EINT10_CFG External INT10 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
7:4	R/W	0x0	EINT9_CFG External INT9 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
3:0	R/W	0x0	EINT8_CFG External INT8 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

9.7.5.46 0x0268 PD External Interrupt Configure Register 2 (Default Value: 0x0000\_0000)

Offset: 0x0268			Register Name:PD_EINT_CFG2
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/

Offset: 0x0268			Register Name:PD_EINT_CFG2
Bit	Read/Write	Default/Hex	Description
27:24	R/W	0x0	EINT22_CFG External INT22 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
23:20	R/W	0x0	EINT21_CFG External INT21 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
19:16	R/W	0x0	EINT20_CFG External INT20 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
15:12	R/W	0x0	EINT19_CFG External INT19 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved



Offset: 0x0268			Register Name: PD_EINT_CFG2
Bit	Read/Write	Default/Hex	Description
11:8	R/W	0x0	EINT18_CFG External INT18 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
7:4	R/W	0x0	EINT17_CFG External INT17 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
3:0	R/W	0x0	EINT16_CFG External INT16 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

**9.7.5.47 0x0270 PD External Interrupt Control Register (Default Value: 0x0000\_0000)**

Offset: 0x0270			Register Name: PD_EINT_CTL
Bit	Read/Write	Default/Hex	Description
31:23	/	/	/
22	R/W	0x0	EINT22_CTL External INT22 Enable 0: Disable 1: Enable

Offset: 0x0270			Register Name: PD_EINT_CTL
Bit	Read/Write	Default/Hex	Description
21	R/W	0x0	EINT21_CTL External INT21 Enable 0: Disable 1: Enable
20	R/W	0x0	EINT20_CTL External INT20 Enable 0: Disable 1: Enable
19	R/W	0x0	EINT19_CTL External INT19 Enable 0: Disable 1: Enable
18	R/W	0x0	EINT18_CTL External INT18 Enable 0: Disable 1: Enable
17	R/W	0x0	EINT17_CTL External INT17 Enable 0: Disable 1: Enable
16	R/W	0x0	EINT16_CTL External INT16 Enable 0: Disable 1: Enable
15	R/W	0x0	EINT15_CTL External INT15 Enable 0: Disable 1: Enable
14	R/W	0x0	EINT14_CTL External INT14 Enable 0: Disable 1: Enable
13	R/W	0x0	EINT13_CTL External INT13 Enable 0: Disable 1: Enable

Offset: 0x0270			Register Name: PD_EINT_CTL
Bit	Read/Write	Default/Hex	Description
12	R/W	0x0	EINT12_CTL External INT12 Enable 0: Disable 1: Enable
11	R/W	0x0	EINT11_CTL External INT11 Enable 0: Disable 1: Enable
10	R/W	0x0	EINT10_CTL External INT10 Enable 0: Disable 1: Enable
9	R/W	0x0	EINT9_CTL External INT9 Enable 0: Disable 1: Enable
8	R/W	0x0	EINT8_CTL External INT8 Enable 0: Disable 1: Enable
7	R/W	0x0	EINT7_CTL External INT7 Enable 0: Disable 1: Enable
6	R/W	0x0	EINT6_CTL External INT6 Enable 0: Disable 1: Enable
5	R/W	0x0	EINT5_CTL External INT5 Enable 0: Disable 1: Enable
4	R/W	0x0	EINT4_CTL External INT4 Enable 0: Disable 1: Enable

Offset: 0x0270			Register Name: PD_EINT_CTL
Bit	Read/Write	Default/Hex	Description
3	R/W	0x0	EINT3_CTL External INT3 Enable 0: Disable 1: Enable
2	R/W	0x0	EINT2_CTL External INT2 Enable 0: Disable 1: Enable
1	R/W	0x0	EINT1_CTL External INT1 Enable 0: Disable 1: Enable
0	R/W	0x0	EINT0_CTL External INT0 Enable 0: Disable 1: Enable

9.7.5.48 0x0274 PD External Interrupt Status Register (Default Value: 0x0000\_0000)

Offset: 0x0274			Register Name: PD_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
31:23	/	/	/
22	R/W1C	0x0	EINT22_STATUS External INT22 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
21	R/W1C	0x0	EINT21_STATUS External INT21 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Offset: 0x0274			Register Name: PD_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
20	R/W1C	0x0	EINT20_STATUS External INT20 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
19	R/W1C	0x0	EINT19_STATUS External INT19 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
18	R/W1C	0x0	EINT18_STATUS External INT18 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
17	R/W1C	0x0	EINT17_STATUS External INT17 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
16	R/W1C	0x0	EINT16_STATUS External INT16 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
15	R/W1C	0x0	EINT15_STATUS External INT15 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
14	R/W1C	0x0	EINT14_STATUS External INT14 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Offset: 0x0274			Register Name: PD_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
13	R/W1C	0x0	EINT13_STATUS External INT13 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
12	R/W1C	0x0	EINT12_STATUS External INT12 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
11	R/W1C	0x0	EINT11_STATUS External INT11 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
10	R/W1C	0x0	EINT10_STATUS External INT10 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
9	R/W1C	0x0	EINT9_STATUS External INT9 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
8	R/W1C	0x0	EINT8_STATUS External INT8 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
7	R/W1C	0x0	EINT7_STATUS External INT7 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Offset: 0x0274			Register Name: PD_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
6	R/W1C	0x0	EINT6_STATUS External INT6 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
5	R/W1C	0x0	EINT5_STATUS External INT5 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
4	R/W1C	0x0	EINT4_STATUS External INT4 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
3	R/W1C	0x0	EINT3_STATUS External INT3 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
2	R/W1C	0x0	EINT2_STATUS External INT2 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
1	R/W1C	0x0	EINT1_STATUS External INT1 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
0	R/W1C	0x0	EINT0_STATUS External INT0 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

9.7.5.49 0x0278 PD External Interrupt Debounce Register (Default Value: 0x0000\_0000)

Offset: 0x0278			Register Name: PD_EINT_DEB
Bit	Read/Write	Default/Hex	Description
31:7	/	/	/
6:4	R/W	0x0	DEB_CLK_PRE_SCALE Debounce Clock Pre_scale n The selected clock source is prescaled by 2^n.
3:1	/	/	/
0	R/W	0x0	PIO_INT_CLK_SELECT PIO Interrupt Clock Select 0: LOSC 32KHz 1: HOSC 24MHz

9.7.5.50 0x0280 PE External Interrupt Configure Register 0 (Default Value: 0x0000\_0000)

Offset: 0x0280			Register Name: PE_EINT_CFG0
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0x0	EINT7_CFG External INT7 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
27:24	R/W	0x0	EINT6_CFG External INT6 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved



Offset: 0x0280			Register Name: PE_EINT_CFG0
Bit	Read/Write	Default/Hex	Description
23:20	R/W	0x0	EINT5_CFG External INT5 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
19:16	R/W	0x0	EINT4_CFG External INT4 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
15:12	R/W	0x0	EINT3_CFG External INT3 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
11:8	R/W	0x0	EINT2_CFG External INT2 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Offset: 0x0280			Register Name: PE_EINT_CFG0
Bit	Read/Write	Default/Hex	Description
7:4	R/W	0x0	EINT1_CFG External INT1 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
3:0	R/W	0x0	EINT0_CFG External INT0 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

9.7.5.51 0x0284 PE External Interrupt Configure Register 1 (Default Value: 0x0000\_0000)

Offset: 0x0284			Register Name: PE_EINT_CFG1
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0x0	Reserved
27:24	R/W	0x0	Reserved
23:20	R/W	0x0	EINT13_CFG External INT13 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Offset: 0x0284			Register Name: PE_EINT_CFG1
Bit	Read/Write	Default/Hex	Description
19:16	R/W	0x0	EINT12_CFG External INT12 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
15:12	R/W	0x0	EINT11_CFG External INT11 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
11:8	R/W	0x0	EINT10_CFG External INT10 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
7:4	R/W	0x0	EINT9_CFG External INT9 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Offset: 0x0284			Register Name: PE_EINT_CFG1
Bit	Read/Write	Default/Hex	Description
3:0	R/W	0x0	EINT8_CFG External INT8 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

9.7.5.52 0x0290 PE External Interrupt Control Register (Default Value: 0x0000\_0000)

Offset: 0x0290			Register Name: PE_EINT_CTL
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13	R/W	0x0	EINT13_CTL External INT13 Enable 0: Disable 1: Enable
12	R/W	0x0	EINT12_CTL External INT12 Enable 0: Disable 1: Enable
11	R/W	0x0	EINT11_CTL External INT11 Enable 0: Disable 1: Enable
10	R/W	0x0	EINT10_CTL External INT10 Enable 0: Disable 1: Enable
9	R/W	0x0	EINT9_CTL External INT9 Enable 0: Disable 1: Enable

Offset: 0x0290			Register Name: PE_EINT_CTL
Bit	Read/Write	Default/Hex	Description
8	R/W	0x0	EINT8_CTL External INT8 Enable 0: Disable 1: Enable
7	R/W	0x0	EINT7_CTL External INT7 Enable 0: Disable 1: Enable
6	R/W	0x0	EINT6_CTL External INT6 Enable 0: Disable 1: Enable
5	R/W	0x0	EINT5_CTL External INT5 Enable 0: Disable 1: Enable
4	R/W	0x0	EINT4_CTL External INT4 Enable 0: Disable 1: Enable
3	R/W	0x0	EINT3_CTL External INT3 Enable 0: Disable 1: Enable
2	R/W	0x0	EINT2_CTL External INT2 Enable 0: Disable 1: Enable
1	R/W	0x0	EINT1_CTL External INT1 Enable 0: Disable 1: Enable
0	R/W	0x0	EINT0_CTL External INT0 Enable 0: Disable 1: Enable

9.7.5.53 0x0294 PE External Interrupt Status Register (Default Value: 0x0000\_0000)

Offset: 0x0294			Register Name: PE_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13	R/W1C	0x0	EINT13_STATUS External INT13 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
12	R/W1C	0x0	EINT12_STATUS External INT12 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
11	R/W1C	0x0	EINT11_STATUS External INT11 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
10	R/W1C	0x0	EINT10_STATUS External INT10 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
9	R/W1C	0x0	EINT9_STATUS External INT9 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
8	R/W1C	0x0	EINT8_STATUS External INT8 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Offset: 0x0294			Register Name: PE_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
7	R/W1C	0x0	EINT7_STATUS External INT7 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
6	R/W1C	0x0	EINT6_STATUS External INT6 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
5	R/W1C	0x0	EINT5_STATUS External INT5 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
4	R/W1C	0x0	EINT4_STATUS External INT4 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
3	R/W1C	0x0	EINT3_STATUS External INT3 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
2	R/W1C	0x0	EINT2_STATUS External INT2 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
1	R/W1C	0x0	EINT1_STATUS External INT1 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Offset: 0x0294			Register Name: PE_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
0	R/W1C	0x0	EINT0_STATUS External INT0 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

9.7.5.54 0x0298 PE External Interrupt Debounce Register (Default Value: 0x0000\_0000)

Offset: 0x0298			Register Name: PE_EINT_DEB
Bit	Read/Write	Default/Hex	Description
31:7	/	/	/
6:4	R/W	0x0	DEB_CLK_PRE_SCALE Debounce Clock Pre_scale n The selected clock source is prescaled by 2^n.
3:1	/	/	/
0	R/W	0x0	PIO_INT_CLK_SELECT PIO Interrupt Clock Select 0: LOSC 32KHz 1: HOSC 24MHz

9.7.5.55 0x02A0 PF External Interrupt Configure Register 0 (Default Value: 0x0000\_0000)

Offset: 0x02A0			Register Name: PF_EINT_CFG0
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:24	R/W	0x0	EINT6_CFG External INT6 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved



Offset: 0x02A0			Register Name: PF_EINT_CFG0
Bit	Read/Write	Default/Hex	Description
23:20	R/W	0x0	EINT5_CFG External INT5 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
19:16	R/W	0x0	EINT4_CFG External INT4 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
15:12	R/W	0x0	EINT3_CFG External INT3 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
11:8	R/W	0x0	EINT2_CFG External INT2 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Offset: 0x02A0			Register Name: PF_EINT_CFG0
Bit	Read/Write	Default/Hex	Description
7:4	R/W	0x0	EINT1_CFG External INT1 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
3:0	R/W	0x0	EINT0_CFG External INT0 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

9.7.5.56 0x02B0 PF External Interrupt Control Register (Default Value: 0x0000\_0000)

Offset: 0x02B0			Register Name: PF_EINT_CTL
Bit	Read/Write	Default/Hex	Description
31:7	/	/	/
6	R/W	0x0	EINT6_CTL External INT6 Enable 0: Disable 1: Enable
5	R/W	0x0	EINT5_CTL External INT5 Enable 0: Disable 1: Enable
4	R/W	0x0	EINT4_CTL External INT4 Enable 0: Disable 1: Enable

Offset: 0x02B0			Register Name: PF_EINT_CTL
Bit	Read/Write	Default/Hex	Description
3	R/W	0x0	EINT3_CTL External INT3 Enable 0: Disable 1: Enable
2	R/W	0x0	EINT2_CTL External INT2 Enable 0: Disable 1: Enable
1	R/W	0x0	EINT1_CTL External INT1 Enable 0: Disable 1: Enable
0	R/W	0x0	EINT0_CTL External INTO Enable 0: Disable 1: Enable

9.7.5.57 0x02B4 PF External Interrupt Status Register (Default Value: 0x0000\_0000)

Offset: 0x02B4			Register Name: PF_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
31:7	/	/	/
6	R/W1C	0x0	EINT6_STATUS External INT6 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
5	R/W1C	0x0	EINT5_STATUS External INT5 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Offset: 0x02B4			Register Name: PF_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
4	R/W1C	0x0	EINT4_STATUS External INT4 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
3	R/W1C	0x0	EINT3_STATUS External INT3 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
2	R/W1C	0x0	EINT2_STATUS External INT2 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
1	R/W1C	0x0	EINT1_STATUS External INT1 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
0	R/W1C	0x0	EINT0_STATUS External INTO Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

9.7.5.58 0x02B8 PF External Interrupt Debounce Register (Default Value: 0x0000\_0000)

Offset: 0x02B8			Register Name: PF_EINT_DEB
Bit	Read/Write	Default/Hex	Description
31:7	/	/	/
6:4	R/W	0x0	DEB_CLK_PRE_SCALE Debounce Clock Pre_scale n The selected clock source is prescaled by 2^n.
3:1	/	/	/

Offset: 0x02B8			Register Name: PF_EINT_DEB
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	PIO_INT_CLK_SELECT PIO Interrupt Clock Select 0: LOSC 32KHz 1: HOSC 24MHz

9.7.5.59 0x02C0 PG External Interrupt Configure Register 0 (Default Value: 0x0000\_0000)

Offset: 0x02C0			Register Name:PG_EINT_CFG0
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0x0	EINT7_CFG External INT7 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
27:24	R/W	0x0	EINT6_CFG External INT6 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
23:20	R/W	0x0	EINT5_CFG External INT5 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Offset: 0x02C0			Register Name:PG_EINT_CFG0
Bit	Read/Write	Default/Hex	Description
19:16	R/W	0x0	EINT4_CFG External INT4 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
15:12	R/W	0x0	EINT3_CFG External INT3 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
11:8	R/W	0x0	EINT2_CFG External INT2 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
7:4	R/W	0x0	EINT1_CFG External INT1 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Offset: 0x02C0			Register Name: PG_EINT_CFG0
Bit	Read/Write	Default/Hex	Description
3:0	R/W	0x0	EINT0_CFG External INT0 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

9.7.5.60 0x02C4 PG External Interrupt Configure Register 1 (Default Value: 0x0000\_0000)

Offset: 0x02C4			Register Name: PG_EINT_CFG1
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0x0	EINT15_CFG External INT15 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
27:24	R/W	0x0	EINT14_CFG External INT14 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Offset: 0x02C4			Register Name: PG_EINT_CFG1
Bit	Read/Write	Default/Hex	Description
23:20	R/W	0x0	EINT13_CFG External INT13 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
19:16	R/W	0x0	EINT12_CFG External INT12 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
15:12	R/W	0x0	EINT11_CFG External INT11 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
11:8	R/W	0x0	EINT10_CFG External INT10 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved



Offset: 0x02C4			Register Name: PG_EINT_CFG1
Bit	Read/Write	Default/Hex	Description
7:4	R/W	0x0	EINT9_CFG External INT9 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
3:0	R/W	0x0	EINT8_CFG External INT8 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

9.7.5.61 0x02D0 PG External Interrupt Control Register (Default Value: 0x0000\_0000)

Offset: 0x02D0			Register Name: PG_EINT_CTL
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15	R/W	0x0	EINT15_CTL External INT15 Enable 0: Disable 1: Enable
14	R/W	0x0	EINT14_CTL External INT14 Enable 0: Disable 1: Enable
13	R/W	0x0	EINT13_CTL External INT13 Enable 0: Disable 1: Enable

Offset: 0x02D0			Register Name: PG_EINT_CTL
Bit	Read/Write	Default/Hex	Description
12	R/W	0x0	EINT12_CTL External INT12 Enable 0: Disable 1: Enable
11	R/W	0x0	EINT11_CTL External INT11 Enable 0: Disable 1: Enable
10	R/W	0x0	EINT10_CTL External INT10 Enable 0: Disable 1: Enable
9	R/W	0x0	EINT9_CTL External INT9 Enable 0: Disable 1: Enable
8	R/W	0x0	EINT8_CTL External INT8 Enable 0: Disable 1: Enable
7	R/W	0x0	EINT7_CTL External INT7 Enable 0: Disable 1: Enable
6	R/W	0x0	EINT6_CTL External INT6 Enable 0: Disable 1: Enable
5	R/W	0x0	EINT5_CTL External INT5 Enable 0: Disable 1: Enable
4	R/W	0x0	EINT4_CTL External INT4 Enable 0: Disable 1: Enable

Offset: 0x02D0			Register Name: PG_EINT_CTL
Bit	Read/Write	Default/Hex	Description
3	R/W	0x0	EINT3_CTL External INT3 Enable 0: Disable 1: Enable
2	R/W	0x0	EINT2_CTL External INT2 Enable 0: Disable 1: Enable
1	R/W	0x0	EINT1_CTL External INT1 Enable 0: Disable 1: Enable
0	R/W	0x0	EINT0_CTL External INTO Enable 0: Disable 1: Enable

9.7.5.62 0x02D4 PG External Interrupt Status Register (Default Value: 0x0000\_0000)

Offset: 0x02D4			Register Name: PG_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15	R/W1C	0x0	EINT15_STATUS External INT15 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
14	R/W1C	0x0	EINT14_STATUS External INT14 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Offset: 0x02D4			Register Name: PG_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
13	R/W1C	0x0	EINT13_STATUS External INT13 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
12	R/W1C	0x0	EINT12_STATUS External INT12 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
11	R/W1C	0x0	EINT11_STATUS External INT11 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
10	R/W1C	0x0	EINT10_STATUS External INT10 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
9	R/W1C	0x0	EINT9_STATUS External INT9 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
8	R/W1C	0x0	EINT8_STATUS External INT8 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
7	R/W1C	0x0	EINT7_STATUS External INT7 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Offset: 0x02D4			Register Name: PG_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
6	R/W1C	0x0	EINT6_STATUS External INT6 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
5	R/W1C	0x0	EINT5_STATUS External INT5 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
4	R/W1C	0x0	EINT4_STATUS External INT4 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
3	R/W1C	0x0	EINT3_STATUS External INT3 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
2	R/W1C	0x0	EINT2_STATUS External INT2 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
1	R/W1C	0x0	EINT1_STATUS External INT1 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
0	R/W1C	0x0	EINT0_STATUS External INT0 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

**9.7.5.63 0x02D8 PG External Interrupt Debounce Register (Default Value: 0x0000\_0000)**

Offset: 0x02D8			Register Name: PG_EINT_DEB
Bit	Read/Write	Default/Hex	Description
31:7	/	/	/
6:4	R/W	0x0	DEB_CLK_PRE_SCALE Debounce Clock Pre_scale n The selected clock source is prescaled by 2 <sup>n</sup> .
3:1	/	/	/
0	R/W	0x0	PIO_INT_CLK_SELECT PIO Interrupt Clock Select 0: LOSC 32KHz 1: HOSC 24MHz

**9.7.5.64 0x0340 PIO Group Withstand Voltage Mode Select Register (Default Value: 0x0000\_0000)**

When the power domain of GPIO is larger than 1.8 V, the withstand voltage is set to 3.3 V mode, the corresponding value in the 0x0340 register is set to 0.

When the power domain of GPIO is 1.8 V, the withstand voltage is set to 1.8 V mode, the corresponding value in the 0x0340 register is set to 1.

Offset: 0x0340			Register Name: PIO_POW_MOD_SEL
Bit	Read/Write	Default/Hex	Description
31:13	/	/	/
12	R/W	0x0	VCCIO_PWR_MOD_SEL VCC_IO POWER MODE Select 0: 3.3 V 1: 1.8 V
11:7	/	/	/
6	R/W	0x0	PG_PWR_MOD_SEL PG_POWER MODE Select 0: 3.3 V 1: 1.8 V If PG_Port Power Source selects VCC_IO, this bit is invalid.

Offset: 0x0340			Register Name: PIO_POW_MOD_SEL
Bit	Read/Write	Default/Hex	Description
5	R/W	0x0	PF_PWR_MOD_SEL PF_POWER MODE Select 0: 3.3 V 1: 1.8 V If PF_Port Power Source selects VCC_IO, this bit is invalid.
4	R/W	0x0	PE_PWR_MOD_SEL PE_POWER MODE Select 0: 3.3 V 1: 1.8 V If PE_Port Power Source selects VCC_IO, this bit is invalid.
3	R/W	0x0	PD_PWR_MOD_SEL PD_POWER MODE Select 0: 3.3 V 1: 1.8 V If PD_Port Power Source selects VCC_IO, this bit is invalid.
2	R/W	0x0	PC_PWR_MOD_SEL PC_POWER MODE Select 0: 3.3 V 1: 1.8 V If PC_Port Power Source selects VCC_IO, this bit is invalid.
1:0	/	/	/

**9.7.5.65 0x0344 PIO Group Withstand Voltage Mode Select Control Register (Default Value: 0x0000\_0000)**

For 1.8 V and 3.3 V power, the withstand function is enabled by default, the corresponding bit in the 0x0344 register is set to 0.

For 2.5 V power, the withstand function is disabled, the corresponding bit in the 0x0344 register is set to 1, and the corresponding withstand voltage in the 0x0340 register needs to be set to 3.3 V.

Offset: 0x0344			Register Name: PIO_POW_MS_CTL
Bit	Read/Write	Default/Hex	Description
31:13	/	/	/
12	R/W	0x0	VCCIO_WS_VOL_MOD_SEL VCC_IO Withstand Voltage Mode Select Control 0: Enable 1: Disable

Offset: 0x0344			Register Name: PIO_POW_MS_CTL
Bit	Read/Write	Default/Hex	Description
11:7	/	/	/
6	R/W	0x0	VCC_PG_WS_VOL_MOD_SEL VCC_PG Withstand Voltage Mode Select Control 0: Enable 1: Disable
5	R/W	0x0	VCC_PF_WS_VOL_MOD_SEL VCC_PF Withstand Voltage Mode Select Control 0: Enable 1: Disable
4	R/W	0x0	VCC_PE_WS_VOL_MOD_SEL VCC_PE Withstand Voltage Mode Select Control 0: Enable 1: Disable
3	R/W	0x0	VCC_PD_WS_VOL_MOD_SEL VCC_PD Withstand Voltage Mode Select Control 0: Enable 1: Disable
2	R/W	0x0	VCC_PC_WS_VOL_MOD_SEL VCC_PC Withstand Voltage Mode Select Control 0: Enable 1: Disable
1:0	/	/	/

#### 9.7.5.66 0x0348 PIO Group Power Value Register (Default Value: 0x0000\_0000)

When the reading value of the 0x0348 register is 0, it indicates that the IO power voltage is greater than 2.5 V.  
When the reading value of the 0x0348 register is 1, it indicates that the IO power voltage is less than 2.0 V.

Offset: 0x0348			Register Name: PIO_POW_VAL
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R	0x0	VCCIO_PWR_VAL VCC_IO Power Value
15:7	/	/	/



Offset: 0x0348			Register Name: PIO_POW_VAL
Bit	Read/Write	Default/Hex	Description
6	R	0x0	PG_PWR_VAL PG_Port Power Value If PG_Port power source selects VCC_IO, this bit is invalid.
5	R	0x0	PF_PWR_VAL PF_Port Power Value If PF_Port power source selects VCC_IO, this bit is invalid.
4	R	0x0	PE_PWR_VAL PE_Port Power Value If PE_Port power source selects VCC_IO, this bit is invalid.
3	R	0x0	PD_PWR_VAL PD_Port Power Value If PD_Port power source selects VCC_IO, this bit is invalid.
2	R	0x0	PC_PWR_VAL PC_Port Power Value If PC_Port power source selects VCC_IO, this bit is invalid.
1:0	/	/	/

9.7.5.67 0x0350 PIO Group Power Voltage Select Control Register (Default Value: 0x0000\_0001)

Offset: 0x0350			Register Name: PIO_POW_VOL_SEL_CTL
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x1	VCC-PF Power Voltage Select Control 0: 1.8 V 1: 3.3 V

## 9.8 GPADC

### 9.8.1 Overview

The General Purpose ADC (GPADC) can convert the external signal into a certain proportion of digital value, to realize the measurement of analog signal, which can be applied to power detection and key detection. This ADC is a type of successive approximation register (SAR) A/D converter.

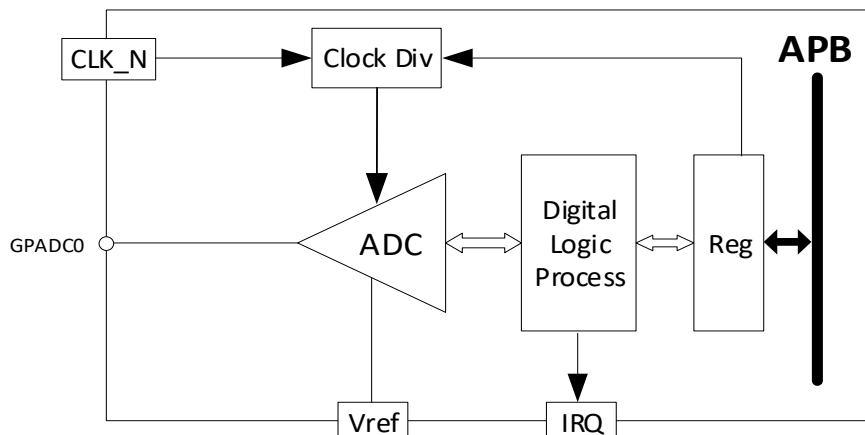
The GPADC has the following features:

- One independent channel
- 12-bit sampling resolution and 8-bit precision
- 64 FIFO depth of data register
- Power reference voltage: AVCC, and analog input voltage range: 0 to AVCC
- Maximum sampling frequency: 1 MHz
- Supports data compare and interrupt
- Supports three operation modes
  - Single conversion mode
  - Continuous conversion mode
  - Burst conversion mode

### 9.8.2 Block Diagram

Figure 9-70 shows the block diagram of the GPADC.

Figure 9-70 GPADC Block Diagram



### 9.8.3 Functional Description

#### 9.8.3.1 External Signals

The following table describes the external signals of the GPADC.

**Table 9-28 GPADC External Signals**

Signal	Description	Type
GPADC0	ADC Input Channel0	AI

#### 9.8.3.2 Clock Sources

The GPADC has one clock source. The following table describes the clock source for GPADC. Users can see section 3.3 “[CCU](#)” for clock setting, configuration, and gating information.

**Table 9-29 GPADC Clock Sources**

Clock Sources	Description
HOSC	24 MHz

#### 9.8.3.3 GPADC Work Mode

- **Single conversion mode**

The GPADC completes one conversion in a specified channel, the converted data is updated at the data register of the corresponding channel.

- **Continuous conversion mode**

The GPADC has continuous conversion in a specified channel until the software stops, the converted data is updated at the data register of the corresponding channel.

- **Burst conversion mode**

The GPADC samples and converts in a specified channel, and sequentially stores the results in FIFO.

#### 9.8.3.4 Clock and Timing Requirements

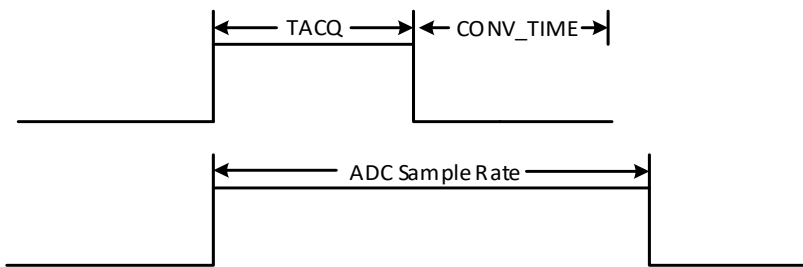
CLK\_IN = 24 MHz

$CONV\_TIME(\text{Conversion Time}) = 1/(24\text{MHz}/14\text{Cycles}) = 0.583 \text{ (us)}$

$TACQ > 10RC$  (R is output impedance of ADC sample circuit, C = 6.4 pF)

ADC Sample Frequency > TACQ+CONV\_TIME

**Figure 9-71 GPADC Clock and Timing Requirement**



### 9.8.3.5 GPADC Calculate Formula

GPADC calculate formula:  $GPADC\_DATA = V_{in}/V_{REF} * 4095$

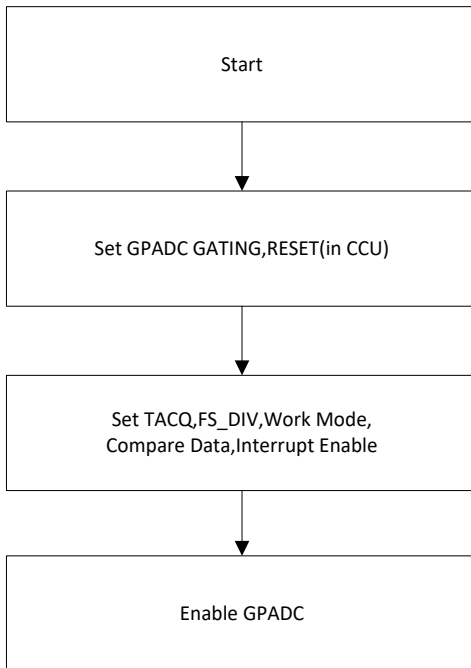
Where:

$V_{REF} = 1.8 \text{ V}$

### 9.8.4 Programming Guidelines

The GPADC initial process is as follows.

Figure 9-72 GPADC Initial Process



(1).Query Mode

- Step 1** Write 0x1 to the bit[16] of [GPADC BGR REG](#) to dessert reset.
- Step 2** Write 0x1 to the bit[0] of [GPADC BGR REG](#) to enable the GPADC clock.
- Step 3** Write 0x2F to the bit[15:0] of [GP\\_SR\\_CON](#) to set the *acquiring time of ADC*.
- Step 4** Write 0x1DF to the bit[31:16] of [GP\\_SR\\_CON](#) to set the *ADC sample frequency divider*.
- Step 5** Write 0x2 to the bit[19:18] of [GP\\_CTRL](#) to set the *continuous conversion mode*.
- Step 6** Write 0x1 to the bit[0] of [GP\\_CS\\_EN](#) to enable the *analog input channel*.
- Step 7** Write 0x1 to the bit[16] of [GP\\_CTRL](#) to enable the ADC function.
- Step 8** Read the bit[0] of [GP\\_DATA\\_INTS](#), if the bit is 1, then data conversion is complete.
- Step 9** Read the bit[11:0] of [GP\\_CHO\\_DATA](#), and calculate voltage value based on GPADC formula.

(2).Interrupt Mode

- Step 1** Write 0x1 to the bit[16] of [GPADC BGR REG](#) to dessert reset.
- Step 2** Write 0x1 to the bit[0] of [GPADC BGR REG](#) to enable the GPADC clock.
- Step 3** Write 0x2F to the bit[15:0] of [GP\\_SR\\_CON](#) to set the *acquiring time of ADC*.
- Step 4** Write 0x1DF to the bit[31:16] of [GP\\_SR\\_CON](#) to set the *ADC sample frequency divider*.
- Step 5** Write 0x2 to the bit[19:18] of [GP\\_CTRL](#) to set the *continuous conversion mode*.

- Step 6** Write 0x1 to the bit[0] of [GP\\_CS\\_EN](#) to enable the *analog input channel*.
- Step 7** Write 0x1 to the bit[0] of [GP\\_DATA\\_INTC](#) to enable the GPADC data interrupt.
- Step 8** Set interrupt based on PLIC module.
- Step 9** Put interrupt handler address into interrupt vector table.
- Step 10** Write 0x1 to the bit16 of [GP\\_CTRL](#) to enable the ADC function.
- Step 11** Read the bit[11:0] of [GP\\_CHO\\_DATA](#) from the interrupt handler, calculate voltage value based on GPADC formula.

### 9.8.5 Register List

Module Name	Base Address
GPADC	0x02009000

Register Name	Offset	Description
GP_SR_CON	0x0000	GPADC Sample Rate Configure Register
GP_CTRL	0x0004	GPADC Control Register
GP_CS_EN	0x0008	GPADC Compare and Select Enable Register
GP_FIFO_INTC	0x000C	GPADC FIFO Interrupt Control Register
GP_FIFO_INTS	0x0010	GPADC FIFO Interrupt Status Register
GP_FIFO_DATA	0x0014	GPADC FIFO Data Register
GP_CDATA	0x0018	GPADC Calibration Data Register
GP_DATAH_INTC	0x0020	GPADC Data Low Interrupt Configure Register
GP_DATAH_INTC	0x0024	GPADC Data High Interrupt Configure Register
GP_DATA_INTC	0x0028	GPADC Data Interrupt Configure Register
GP_DATAH_INTC	0x0030	GPADC Data Low Interrupt Status Register
GP_DATAH_INTC	0x0034	GPADC Data High Interrupt Status Register
GP_DATA_INTC	0x0038	GPADC Data Interrupt Status Register
GP_CHO_CMP_DATA	0x0040	GPADC CH0 Compare Data Register
GP_CHO_DATA	0x0080	GPADC CH0 Data Register

## 9.8.6 Register Description

### 9.8.6.1 0x0000 GPADC Sample Rate Configure Register (Default Value: 0x01DF\_002F)

Offset: 0x0000			Register Name: GP_SR_CON
Bit	Read/Write	Default/Hex	Description
31:16	R/W	0x1DF	FS_DIV ADC sample frequency divider $CLK\_IN/(n+1)$ Default value: 50K
15:0	R/W	0x2F	TACQ ADC acquire time $(n+1)/CLK\_IN$ Default value: 2 us

### 9.8.6.2 0x0004 GPADC Control Register (Default Value: 0x0080\_0000)

Offset: 0x0004			Register Name: GP_CTRL
Bit	Read/Write	Default/Hex	Description
31:24	R/ W	0x0	ADC_FIRST_DLY ADC First Convert Delay Setting ADC conversion of each channel is delayed by N samples.
23	R/ W	0x1	ADC_AUTOCALI_EN ADC Auto Calibration
22	/	/	/
21:20	R/W	0x0	ADC_OP_BIAS ADC OP Bias Adjust the bandwidth of the ADC amplifier
19:18	R/W	0x0	GPADC Work Mode 00: Single conversion mode 01: Reserved 10: Continuous conversion mode 11: Burst conversion mode
17	R/W	0x0	ADC_CALI_EN ADC Calibration 1: Start Calibration, it is cleared to 0 after calibration

Offset: 0x0004			Register Name: GP_CTRL
Bit	Read/Write	Default/Hex	Description
16	R/W	0x0	ADC_EN ADC Function Enable Before the bit is enabled, configure ADC parameters including the work mode and channel number, etc. 0: Disable 1: Enable <b>Note: When selecting a single conversion mode, the bit can be cleared automatically after the switch is completed.</b>
15:0	/	/	/

9.8.6.3 0x0008 GPADC Compare and Select Enable Register (Default Value: 0x0000\_0000)

Offset: 0x0008			Register Name: GP_CS_EN
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	ADC_CHO_CMP_EN Channel 0 Compare Enable 0: Disable 1: Enable
15:1	/	/	/
0	R/W	0x0	ADC_CHO_SELECT Analog Input Channel 0 Select 0: Disable 1: Enable

9.8.6.4 0x000C GPADC FIFO Interrupt Control Register (Default Value: 0x0000\_1F00)

Offset: 0x000C			Register Name: GP_FIFO_INTC
Bit	Read/Write	Default/Hex	Description
31:19	/	/	/
18	R/W	0x0	FIFO_DATA_DRQ_EN ADC FIFO Data DRQ Enable 0: Disable 1: Enable



Offset: 0x000C			Register Name: GP_FIFO_INTC
Bit	Read/Write	Default/Hex	Description
17	R/W	0x0	FIFO_OVERRUN_IRQ_EN ADC FIFO Overrun IRQ Enable 0: Disable 1: Enable
16	R/W	0x0	FIFO_DATA_IRQ_EN ADC FIFO Data Available IRQ Enable 0: Disable 1: Enable
15:14	/	/	/
13:8	R/W	0x1F	FIFO_TRIG_LEVEL Interrupt trigger level for ADC Trigger Level = TXTL + 1
7:5	/	/	/
4	R/WAC	0x0	FIFO_FLUSH ADC FIFO Flush Write '1' to flush TX FIFO, clear automatically to '0'.
3:0	/	/	/

9.8.6.5 0x0010 GPADC FIFO Interrupt Status Register (Default Value: 0x0000\_0000)

Offset: 0x0010			Register Name: GP_FIFO_INTS
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17	R/W1C	0x0	FIFO_OVERRUN_PENDING ADC FIFO Overrun IRQ Pending 0: No Pending IRQ 1: FIFO Overrun Pending IRQ Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.
16	R/W1C	0x0	FIFO_DATA_PENDING ADC FIFO Data Available Pending Bit 0: NO Pending IRQ 1: FIFO Available Pending IRQ Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.

Offset: 0x0010			Register Name: GP_FIFO_INTS
Bit	Read/Write	Default/Hex	Description
15:14	/	/	/
13:8	R	0x0	RXA_CNT ADC FIFO available sample word counter
7:0	/	/	/

9.8.6.6 0x0014 GPADC FIFO Data Register (Default Value: 0x0000\_0000)

Offset: 0x0014			Register Name: GP_FIFO_DATA
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R	0x0	GP_FIFO_DATA GPADC Data in FIFO

9.8.6.7 0x0018 GPADC Calibration Data Register (Default Value: 0x0000\_0000)

Offset: 0x0018			Register Name: GP_CDATA
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R/W	0x0	GP_CDATA GPADC Calibration Data

9.8.6.8 0x0020 GPADC Low Interrupt Configure Register (Default Value: 0x0000\_0000)

Offset: 0x0020			Register Name: GP_DATA1_INTC
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	CHO_LOW_IRQ_EN Channel 0 Voltage Low Available Interrupt Enable 0: Disable 1: Enable

9.8.6.9 0x0024 GPADC High Interrupt Configure Register (Default Value: 0x0000\_0000)

Offset: 0x0024			Register Name: GP_DATAH_INTC
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	CHO_HIG_IRQ_EN Channel 0 Voltage High Available Interrupt Enable 0: Disable 1: Enable

9.8.6.10 0x0028 GPADC DATA Interrupt Configure Register (Default Value: 0x0000\_0000)

Offset: 0x0028			Register Name: GP_DATA_INTC
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	CHO_DATA_IRQ_EN 0: Disable 1: Enable

9.8.6.11 0x0030 GPADC Low Interrupt Status Register (Default Value: 0x0000\_0000)

Offset: 0x0030			Register Name: GP_DATA_L_INTS
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W1C	0x0	CHO_LOW_PENGDING Channel 0 Voltage Low Available Interrupt Status 0: NO Pending IRQ 1: Channel 0 Voltage Low Available Pending IRQ Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.

9.8.6.12 0x0034 GPADC High Interrupt Status Register (Default Value: 0x0000\_0000)

Offset: 0x0034			Register Name: GP_DATAH_INTS
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W1C	0x0	CHO_HIG_PENGDING 0: No Pending IRQ 1: Channel 0 Voltage High Available Pending IRQ Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.

9.8.6.13 0x0038 GPADC Data Interrupt Status Register (Default Value: 0x0000\_0000)

Offset: 0x0038			Register Name: GP_DATA_INTS
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W1C	0x0	CHO_DATA_PENGDING Channel 0 Data Available Interrupt Status 0: No Pending IRQ 1: Channel 0 Data Available Pending IRQ Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.

9.8.6.14 0x0040 GPADC CH0 Compare Data Register (Default Value: 0x0BFF\_0400)

Offset: 0x0040			Register Name: GP_CH0_CMP_DATA
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0xBFF	CHO_CMP_HIG_DATA Channel 0 Voltage High Value
15:12	/	/	/
11:0	R/W	0x400	CHO_CMP_LOW_DATA Channel 0 Voltage Low Value

9.8.6.15 0x0080 GPADC CH0 Data Register (Default Value: 0x0000\_0000)

Offset: 0x0080			Register Name: GP_CH0_DATA
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R	0x000	GP_CH0_DATA Channel 0 Data





## 9.9 TPADC

### 9.9.1 Overview

The Touch Panel ADC (TPADC) is a 4-wire resistive touch screen controller, including a 12-bit SAR type A/D converter.

The TPADC has the following features:

- 12 bit SAR type A/D converter
- Configurable sample frequency up to 1 MHz
- One 32x12 FIFO for storing A/D conversion result
- Supports DMA slave interface
- Supports 4-wire resistive touch panel input detection
  - Supports pen down detection with programmable sensitivity
  - Supports single touch coordinate measurement
  - Supports dual touch detection
  - Supports touch pressure measurement with programmable threshold
  - Supports median and averaging filter for noise reduction
  - Supports X and Y coordinate exchange function
- Supports Aux ADC with up to 4 channels

### 9.9.2 Functional Description

#### 9.9.2.1 External Signals

The following table describes the external signals of the TPADC.

**Table 9-30 TPADC External Signals**

Signal	Description	Type
TP-X1	Touch Panel X1 Input	AI
TP-X2	Touch Panel X2 Input	AI
TP-Y1	Touch Panel Y1 Input	AI
TP-Y2	Touch Panel Y2 Input	AI

### 9.9.2.2 Single-ended Mode and Differential Mode

The controller is a typical type of successive approximation ADC (SAR ADC) which contains a sample/hold, analog-to-digital conversion, serial data output functions.

The analog inputs (X+, X-, Y+, Y-) enter the ADC through the control register, the ADC can work in single-ended or differential mode. Selecting Aux ADC should work in single-ended mode; for a touch screen application, it works in a differential mode, which can effectively eliminate the impact on conversion accuracy caused by the parasitic resistance of the driver switch and external interference.

Figure 9-73 shows TPADC Single-Ended Mode for the measurement of Aux, using the 1.8 V reference source as the ADC reference voltage.

Figure 9-73 TPADC Single-Ended Mode for AUX ADC

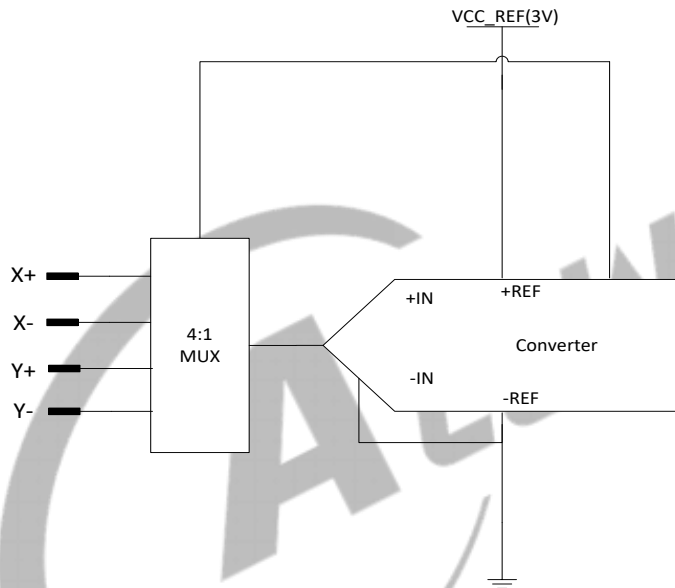
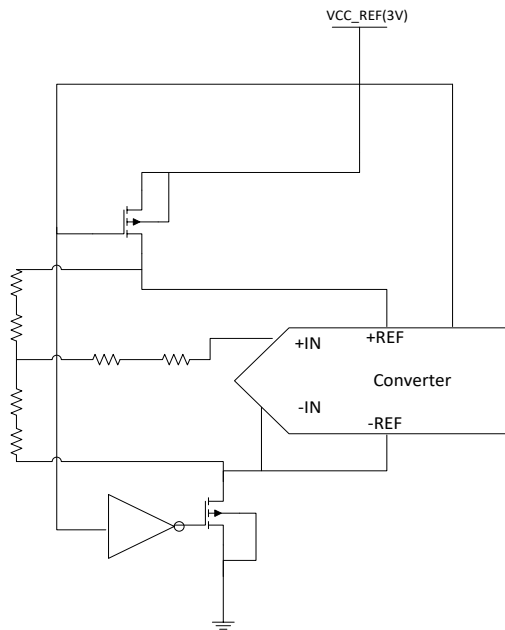


Figure 9-74 shows TPADC differential mode for the measurement of X/Y/Z coordinate of Touch Panel. The advantage of differential mode: +REF and -REF can directly input to the Y+ and Y-(or X+ and X-), which can eliminate the measurement error of X+/X-(or Y+/Y-) because of the switch on resistance. The disadvantage is that: both the sample or conversion process, the driver will need to be enabled. Compared with single-ended mode, the power consumption increases.



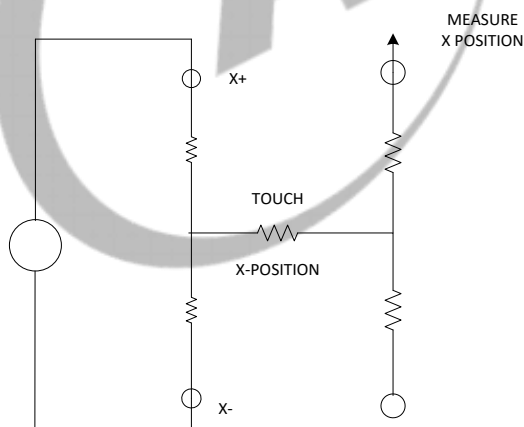
Figure 9-74 TPADC Differential Mode for Touch Panel



### 9.9.2.3 Single Touch Measurement

The following figure shows the operation principle of the single touch X-Coordinate measurement.

Figure 9-75 Single Touch X-Coordinate Measurement for Touch Panel



For an X coordinate measurement, the X+ pin is internally switched to VCC\_REF and X- to GND. The X plate becomes a potential divider, and the voltage at the point of contact is proportional to its X co-ordinate. This voltage is measured on the Y+, which carries no current (hence there is no voltage drop in RY+ or RY-). Due to the ratio metric measurement method, the supply voltage does not affect measurement accuracy. The voltage

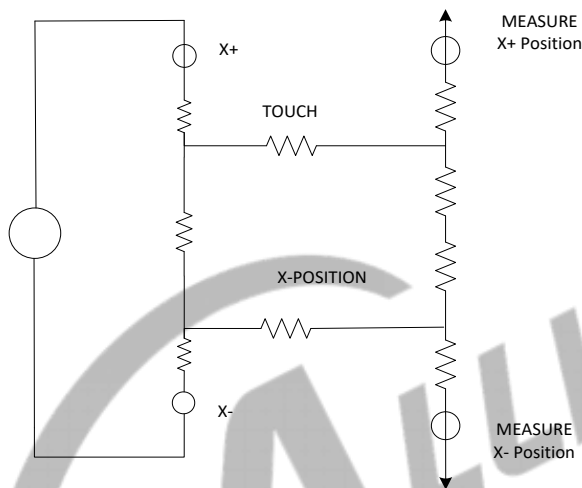
references VREF+ and VREF- are taken from after the matrix switches, so that any voltage drop in these switches has no effect on the ADC measurement.

Y coordinate measurements are similar to X coordinate measurements, with the X and Y plates interchanged.

#### 9.9.2.4 Dual Touch Measurement

In single touch mode, it only needs to test X+ and Y+ signals. But in dual touch mode, it needs to test X+, X-, Y+, and Y- signals. The following figure shows the operation principle of dual touch detection for touch panel.

Figure 9-76 Dual Touch Detection for Touch Panel

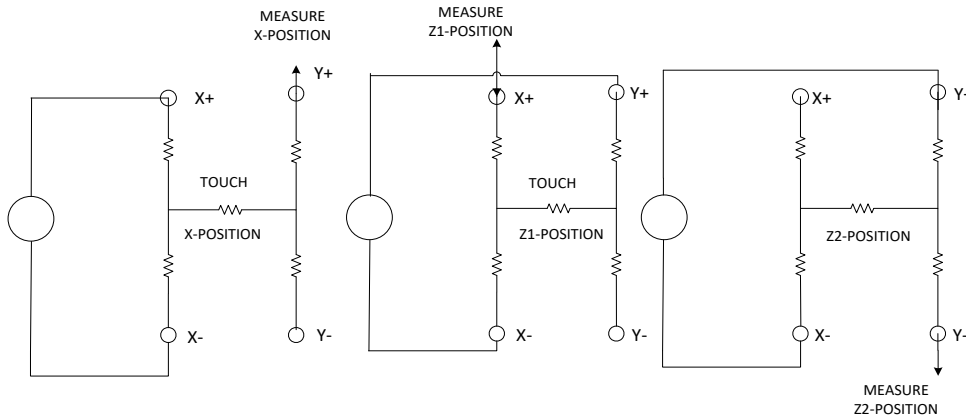


For X coordinates measurement, the X+ pin is internally switched to 3 V and X- to GND. The controller needs to test Y+ and Y-, Y coordinates measurement is similar. And record  $\Delta X = |X+ - X-|$ ,  $\Delta Y = |Y+ - Y-|$ . In practice, we can set a threshold. If  $\Delta X$  or  $\Delta Y$  is greater than the threshold, we consider it as a dual touch, otherwise as a single touch.

#### 9.9.2.5 Touch Pressure Measurement

The pressure applied to the touch screen by a pen or finger to filter unavailable can also be measured by the controller using some simple calculations. The contact resistance between the X and Y plates is measured, which provides a good indication of the size of the depressed area and the applied pressure. The area of the touch spot  $t$  is proportional to the size of the object touching it. And the value of this resistance ( $R_{touch}$ ) can be calculated using two different methods.

Figure 9-77 Touch Pressure Measurement for Touch Panel



(1) First Method

The first method requires the user to know the total resistance of the X plate tablet ( $R_{XPLATE}$ ). Three touch screen conversions are required: measurement of the X position,  $X_{POSITION}$  (Y+ input); measurement of the X+ input with the excitation voltage applied to Y+ and X- (Z1 measurement); and measurement of the Y- input with the excitation voltage applied to Y+ and X- (Z2 measurement). These three measurements are illustrated in following Figure. The controller have two special ADC channel settings to configure the X and Y switches for the Z1 and Z2 measurements and store the results in the Z1 and Z2 result registers. The touch resistance ( $R_{TOUCH}$ ) can then be calculated using the following equation.

$$R_{TOUCH} = (R_{XPLATE}) \times (X_{POSITION}/4096) \times [(Z2/Z1) - 1]$$

(2) Second Method

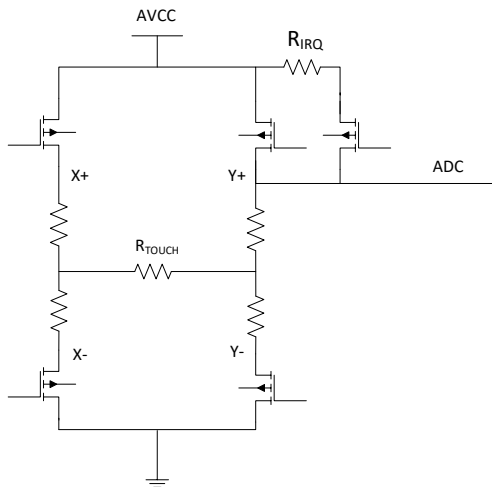
The second method requires the user to know the resistance of the X-plate and Y-plate tablets. Three touch screen conversions are required: a measurement of the X position ( $X_{POSITION}$ ), the Y position ( $Y_{POSITION}$ ), and the Z1 position. The following equation also calculates the touch resistance ( $R_{TOUCH}$ ).

$$R_{TOUCH} = R_{XPLATE} \times (X_{POSITION}/4096) \times [(4096/Z1) - 1] - R_{YPLATE} \times [1 - (Y_{POSITION}/4096)]$$

9.9.2.6 Pen Down Detection

Pen down detection is used as an interrupt to the host. RIRQ is an internal pull-up resistor with a programmable value from 6 kΩ to 96 kΩ (default 48 kΩ).

Figure 9-78 Pen Down Detection for Touch Panel



The pen down IRQ output is pulled high by an internal pull-up. In the pen down detection, the Y- driver is enabled and connected to GND, and the pen down IRQ output is connected to the X+ input. When the panel is touched, the X+ input is pulled to ground through the touch screen, and the pen down IRQ output goes low because of the current path through the panel to GND, initiating an interrupt to the processor.

During the measurement cycle for X-, Y-, and Z-position, the X+ input is disconnected from the pen down IRQ pull-down transistor to eliminate any pull-up resistor leakage current from flowing through the touch screen, thus causing no errors.

### 9.9.2.7 Median and Averaging Filter

Touch screens are composed of two resistive layers, normally placed over an LCD screen. Because these layers are in close proximity to the LCD screen, noise can be coupled from the screen onto these resistive layers, causing errors in the touch screen positional measurements.

The controller contain a filtering block to process the data and discard the spurious noise before sending the information to the host. The purpose of this block is not only the suppression of noise; the on-chip filtering also greatly reduces the host processing loading.

The processing function consists of two filters that are applied to the converted results: the median filter and the averaging filter. The median filter suppresses the isolated out-of-range noise and sets the number of measurements to be taken. These measurements are arranged in a temporary array, where the first value is the smallest measurement and the last value is the largest measurement. Then the averaging filter size determines the number of values to average. There are four choices which is configured by TP\_CTRL3 register (bit 1 and bit 0) to filtrate the ADC sampling data.

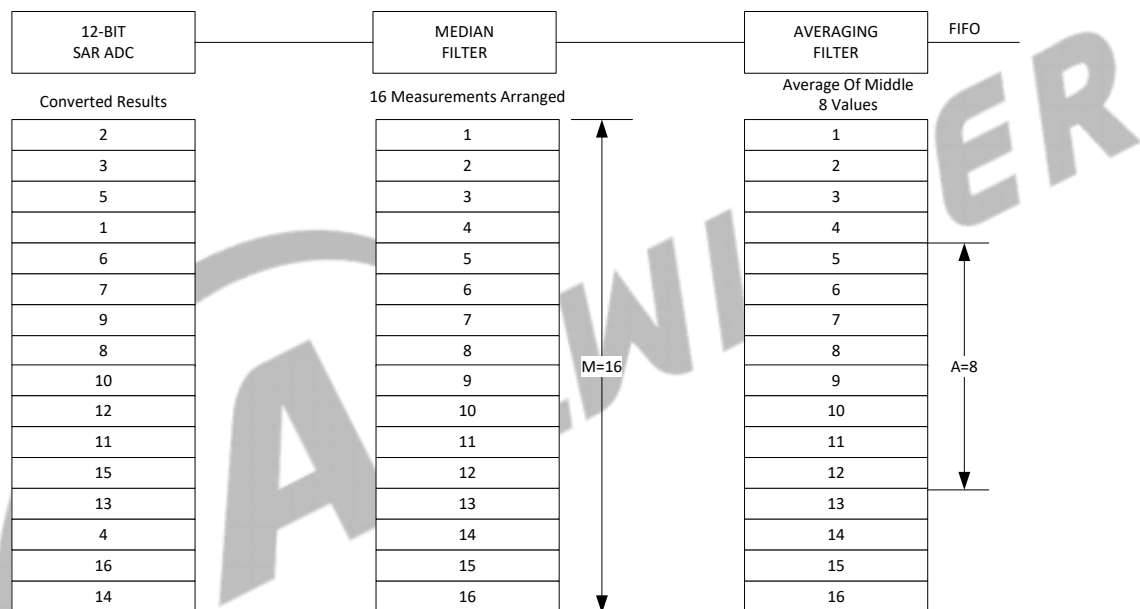
Figure 9-79 Median and Averaging Filter Size

Bit1	Bit0	Averaging Filter Size	Median Filter Size
0	0	2	4

Bit1	Bit0	Averaging Filter Size	Median Filter Size
0	1	3	5
1	0	4	8
1	1	8	16

Example: In this example, the bit[1:0] of TP\_CTRL\_REG3 is configured as 2'b11. So the median filter has a window size of 16. This means that 16 measurements are taken and arranged in descending order in a temporary array. The averaging window size in this example is 8. The output is the average of the middle eight values of the 16 measurements taken with the median filter.

Figure 9-80 Median and Averaging Filter Example



### 9.9.3 Register List

Module Name	Base Address
TPADC	0x02009C00

Register Name	Offset	Description
TP_CTRL_REG0	0x0000	TP Control Register 0
TP_CTRL_REG1	0x0004	TP Control Register 1
TP_CTRL_REG2	0x0008	TP Control Register 2
TP_CTRL_REG3	0x000C	TP Control Register 3
TP_INT_FIFO_CTRL_REG	0x0010	TP Interrupt FIFO Control Register
TP_INT_FIFO_STAT_REG	0x0014	TP Interrupt FIFO Status Register

Register Name	Offset	Description
TP_CALI_DATA_REG	0x001C	TP Calibration Data Register
TP_DATA_REG	0x0024	TP Data Register

### 9.9.4 Register Description

#### 9.9.4.1 0x0000 TP Control Register 0 (Default Value:0x0F80\_0000)

Offset: 0x0000			Register Name: TP_CTRL0
Bit	Read/Write	Default/Hex	Description
31:24	R/W	0xF	ADC_FIRST_DLY ADC First Convert Delay Time (T_FCDT) Setting Based on ADC first convert delay mode select (Bit 23) $T\_FCDT = \text{ADC\_FIRST\_DLY} * \text{ADC\_FIRST\_DLY\_MODE}$
23	R/W	0x1	ADC_FIRST_DLY_MODE ADC First Convert Delay Mode Select 0: CLK_IN/16 1: CLK_IN/16*256
22	/	/	/
21:20	R/W	0x0	ADC_CLK_DIVIDER ADC Clock Divider (CLK_IN) 00: CLK/2 01: CLK/3 10: CLK/6 11: CLK/1
19:16	R/W	0x0	FS_DIV ADC Sample Frequency Divider 0000: CLK_IN/2 <sup>(20-0)</sup> 0001: CLK_IN/2 <sup>(20-1)</sup> 0010: CLK_IN/2 <sup>(20-2)</sup> .... 1111: CLK_IN/2 <sup>(20-15)</sup>
15:0	R/W	0x0	TACQ Touch panel ADC acquire time CLK_IN/(16*(N+1))

9.9.4.2 0x0004 TP Control Register 1 (Default Value:0x0000\_0101)

Offset: 0x0004			Register Name: TP_CTRL_REG1
Bit	Read/Write	Default/Hex	Description
31:20	/	/	/
19:12	R/W	0x0	STYLUS_UP_DEBOUNCE Stylus Up De-bounce Time Setting 0x00: 0 .... 0xFF: 2N*(CLK_IN/16*256)
11:10	/	/	/
9	R/W	0x0	STYLUS_UP_DEBOUNCE_EN Stylus Up Debounce Function Select 0: Disable 1: Enable
8	R/W	0x1	CHOPPER_EN T-sensor Chopping Enable 0: Disable 1: Enable This field is not used when there is no T-sensor in TPADC
7	R/W	0x0	TOUCH_PAN_CALI_EN Touch Panel Calibration 1: Start calibration, it is cleared to 0 after calibration
6	R/W	0x0	TP_DUAL_EN Touch Panel Double Point Enable 0: Disable 1: Enable
5	R/W	0x0	TP_EN. TP Function Enable 0: Disable 1: Enable
4	R/W	0x0	TP_MODE_SELECT. Touch Panel Mode and Auxiliary ADC Mode Select 0: TP 1: Auxiliary ADC

Offset: 0x0004			Register Name: TP_CTRL_REG1
Bit	Read/Write	Default/Hex	Description
3	R/W	0x0	ADC_CHAN3_SELECT Analog Input Channel 3 Select 0: Disable 1: Enable
2	R/W	0x0	ADC_CHAN2_SELECT Analog Input Channel 2 Select 0: Disable 1: Enable
1	R/W	0x0	ADC_CHAN1_SELECT Analog Input Channel 1 Select 0: Disable 1: Enable
0	R/W	0x1	ADC_CHAN0_SELECT Analog Input Channel 0 Select 0: Disable 1: Enable

CHAN0–3 can be selected at the same time. If N channel is selected, each channel has 1/N full speed of the ADC. If only one channel is selected, it has the full conversion rate. CHAN0–3 correspond to TP\_YN, TP\_YP, TP\_XN, TP\_XP.

**9.9.4.3 0x0008 TP Control Register 2 (Default Value:0x8000\_0FFF)**

Offset: 0x0008			Register Name: TP_CTRL_REG2
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0x8	TP_SENSITIVE_ADJUST Internal Pull-up Resistor Control 0000: least sensitive ..... 1111: most sensitive This field is used to adjust sensitivity of pen down detection.



Offset: 0x0008			Register Name: TP_CTRL_REG2
Bit	Read/Write	Default/Hex	Description
27:26	R/W	0x0	<p>TP_FIFO_MODE_SELECT TP FIFO Access Data Mode Select</p> <p>00: FIFO store X1,Y1 data for single touch no pressure mode 01: FIFO store X1,Y1, ΔX, ΔY data for dual touch no pressure mode 10: FIFO store X1,Y1, X2,Y2 data for dual touch no pressure mode 11: FIFO store X1,Y1, X2,Y2,Z1,Z2 data for dual touch and pressure mode</p> <p>The ADC output data in single touch mode can store in FIFO with TP_FIFO_MODE_SELECT configured as 01,10,11. But the data ΔX, ΔY is theoretically equal to X1,Y1, and X2,Y2 is equal to 0.</p> <p>When PRE_MEA_EN is set and TP_FIFO_MODE_SELECT is not configured as 2'b11, X and Y data will not be stored unless <math>x1*(z2-z1)/z1 &lt; PRE\_MEA\_THRE\_CNT</math>.</p> <p>Z data will always be zero when TP_FIFO_MODE_SELECT is configured as 2'b11 but PRE_MEA_EN is not set.</p>
25	/	/	/
24	R/W	0x0	<p>PRE_MEA_EN TP Pressure Measurement Enable Control</p> <p>0: Disable 1: Enable</p>
23:0	R/W	0xFFFF	<p>PRE_MEA_THRE_CNT TP Pressure Measurement Threshold Control</p> <p>0x000000:least sensitive 0xFFFFFFF: most sensitive</p> <p>This field is used to adjust sensitivity of touch.</p>

9.9.4.4 0x000C TP Control Register 3 (Default Value:0x0000\_0001)

Offset: 0x000C			Register Name: TP_CTRL_REG3
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/

Offset: 0x000C			Register Name: TP_CTRL_REG3
Bit	Read/Write	Default/Hex	Description
2	R/W	0x0	FILTER_EN Filter Enable 0: Disable 1: Enable
1:0	R/W	0x1	FILTER_TYPE Filter Type 00: 4/2 01: 5/3 10: 8/4 11: 16/8

9.9.4.5 0x0010 TP Interrupt& FIFO Control Register (Default Value:0x0000\_0F00)

Offset: 0x0010			Register Name: TP_INT_FIFO_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17	R/W	0x0	TP_OVERRUN_IRQ_EN TP FIFO Overrun IRQ Enable 0: Disable 1: Enable
16	R/W	0x0	TP_DATA_IRQ_EN TP FIFO Data Available IRQ Enable 0: Disable 1: Enable
15:14	/	/	/
13	R/W	0x0	TP_DATA_XY_CHANGE TP FIFO X,Y Data Interchange Function Select 0: Disable 1: Enable
12:8	R/W	0xF	TP_FIFO_TRIG_LEVEL TP FIFO Data Available Trigger Level Interrupt and DMA request trigger level for TP or Auxiliary ADC Trigger Level = TXTL + 1

Offset: 0x0010			Register Name: TP_INT_FIFO_CTRL_REG
Bit	Read/Write	Default/Hex	Description
7	R/W	0x0	TP_DATA_DRQ_EN TP FIFO Data Available DRQ Enable 0: Disable 1: Enable
6:5	/	/	/
4	R/WAC	0x0	TP_FIFO_FLUSH TP FIFO Flush Write '1' to flush TX FIFO, self clear to '0'
3:2	/	/	/
1	R/W	0x0	TP_UP_IRQ_EN Touch Panel Last Touch (Stylus Up) IRQ Enable 0: Disable 1: Enable
0	R/W	0x0	TP_DOWN_IRQ_EN Touch Panel First Touch (Stylus Down) IRQ Enable 0: Disable 1: Enable

9.9.4.6 0x0014 TP Interrupt& FIFO Status Register (Default Value:0x0000\_0000)

Offset: 0x0014			Register Name: TP_INT_FIFO_STAT_REG
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17	R/W1C	0x0	FIFO_OVERRUN_PENDING TP FIFO Overrun IRQ pending 0: No Pending IRQ 1: FIFO Overrun Pending IRQ Write '1' to clear this interrupt or automatically clear if interrupt condition fails.
16	R/W1C	0x0	FIFO_DATA_PENDING TP FIFO Data Available Pending Bit 0: NO Pending IRQ 1: FIFO Available Pending IRQ Write '1' to clear this interrupt or automatically clear if FIFO flushed.

Offset: 0x0014			Register Name: TP_INT_FIFO_STAT_REG
Bit	Read/Write	Default/Hex	Description
15:14	/	/	/
13:8	R	0x0	RXA_CNT TP FIFO Available Sample Word Counter
7:3	/	/	/
2	R	0x0	TP_IDLE_FLG Touch Panel Idle Flag 0: idle 1: not idle
1	R/W1C	0x0	TP_UP_PENDING Touch Panel Last Touch (Stylus Up) IRQ Pending bit 0: No IRQ 1: IRQ Writing 1 to the bit clears it and its corresponding interrupt if the interrupt is enabled.
0	R/W1C	0x0	TP_DOWN_PENDING Touch Panel First Touch (Stylus Down) IRQ Pending bit 0: No IRQ 1: IRQ Writing 1 to the bit clears it and its corresponding interrupt if the interrupt is enabled.

**9.9.4.7 0x001C TP Calibration Data Register (Default Value:0x0000\_0800)**

Offset: 0x001C			Register Name: TP_CALI_DATA_REG
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R/W	0x800	TP_CDAT TP Common Data It is used to adjust the tolerance of the internal ADC.

**9.9.4.8 0x0024 TP Data Register (Default Value:0x0000\_0000)**

In touch panel mode, the data stored in this register bases on TP\_FIFO\_MODE\_SELECT. In Auxiliary ADC mode, the data stored in this register bases on ADC\_CHAN\_SELECT. If four channels are enabled, FIFO will access the input data in successive turn (ADC\_CHAN0 -> ADC\_CHAN1 -> ADC\_CHAN2 -> ADC\_CHAN3). If only two or three

channels are selected, such as ADC\_CHAN0 and ADC\_CHAN3, firstly ADC\_CHAN0 input data is accessed, then ADC\_CHAN3 input data.

Offset: 0x0024			Register Name: TP_DATA_REG
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R	0x0	TP_DATA Touch Panel X ,Y data or Auxiliary analogy input data converted by the internal ADC.



## 9.10 PWM

### 9.10.1 Overview

The Pulse Width Modulation (PWM) module can output the configurable PWM waveforms and measure the external input waveforms.

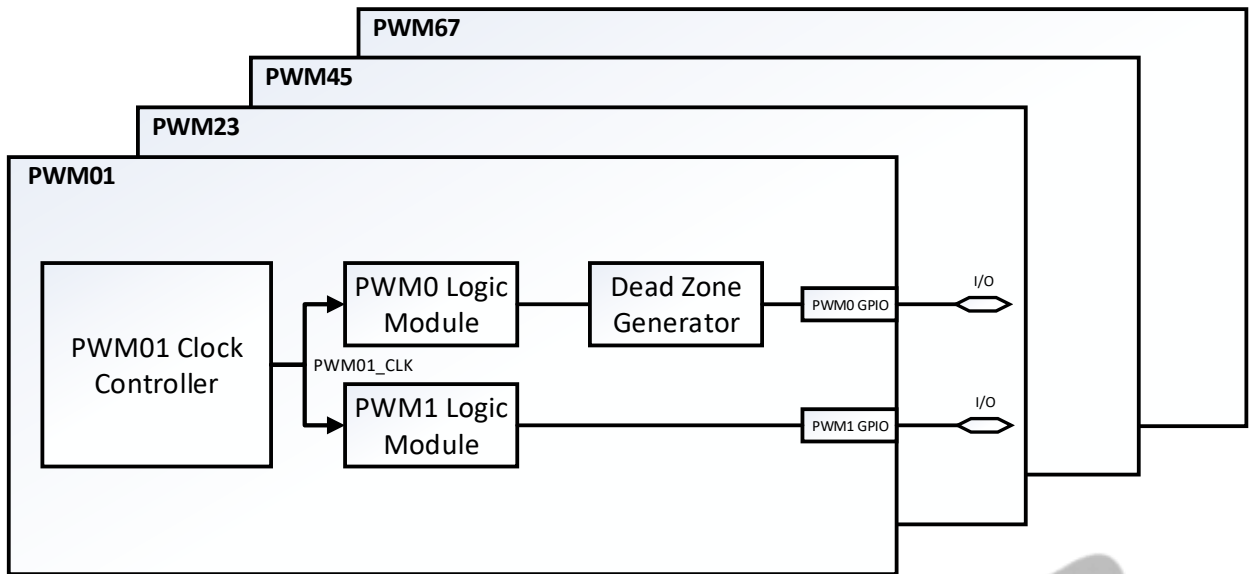
The PWM has the following features:

- Supports 8 independent PWM channels (PWM0 to PWM7)
  - Supports PWM continuous mode output
  - Supports PWM pulse mode output, and the pulse number is configurable
  - Output frequency range: 0 to 24 MHz or 100 MHz
  - Various duty-cycle: 0% to 100%
  - Minimum resolution: 1/65536
- Supports 4 complementary pairs output
  - PWM01 pair (PWM0 + PWM1), PWM23 pair (PWM2 + PWM3), PWM45 pair (PWM4 + PWM5), PWM67 pair (PWM6 + PWM7)
  - Supports dead-zone generator, and the dead-zone time is configurable
- Supports 4 group of PWM channel output for controlling stepping motors
  - Supports any plural channels to form a group, and output the same duty-cycle pulse
  - In group mode, the relative phase of the output waveform for each channel is configurable
- Supports 8 channels capture input
  - Supports rising edge detection and falling edge detection for input waveform pulse
  - Supports pulse-width measurement for input waveform pulse

### 9.10.2 Block Diagram

The PWM includes multi PWM channels. Each channel can generate different PWM waveform by the independent counter and duty-ratio configuration register. Each PWM pair shares one group of clock and dead-zone generator to generate PWM waveform.

Figure 9-81 PWM Block Diagram



Each PWM pair consists of 1 clock module, 2 timer logic module, and 1 programmable dead-zone generator.

### 9.10.3 Functional Description

#### 9.10.3.1 External Signals

The following table describes the external signals of the PWM.

Table 9-31 PWM External Signals

Signal	Description	Type
PWM0	Pulse Width Module Channel0	I/O
PWM1	Pulse Width Module Channel1	I/O
PWM2	Pulse Width Module Channel2	I/O
PWM3	Pulse Width Module Channel3	I/O
PWM4	Pulse Width Module Channel4	I/O
PWM5	Pulse Width Module Channel5	I/O
PWM6	Pulse Width Module Channel6	I/O
PWM7	Pulse Width Module Channel7	I/O

#### 9.10.3.2 Typical Application

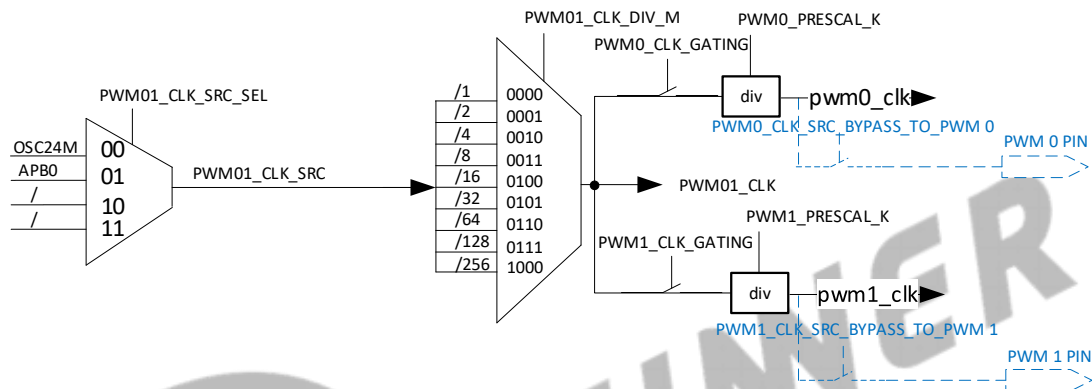
- Suitable for display device, such as LCD

- Suitable for electric motor control

### 9.10.3.3 Clock Controller

Using PWM01 as an example. The other PWM pairs are the same as PWM01.

Figure 9-82 PWM01 Clock Controller Diagram



The clock controller of each PWM pair includes clock source select ([PWM01\\_CLK\\_SRC](#)), 1~256 scaler ([PWM01\\_CLK\\_DIV\\_M](#)). Each PWM channel has the secondary frequency division ([PWM\\_PRESCAL\\_K](#)), clock source bypass ([PWMx\\_CLK\\_BYPASS](#)) and clock switch ([PWMx\\_CLK\\_GATING](#)).

The clock sources have HOSC and APB0. The HOSC comes from the external high-frequency oscillator; the APB0 is APB0 bus clock.

The bypass function of the clock source is that the clock source directly accesses PWM output, the PWM output waveform is the waveform of the clock controller output. The BYPASS gridlines in the above figure indicate the bypass function of the clock source, see Figure 9-83 for the details about implement. At last, the output clock of the clock controller is sent to the PWM logic module.

### 9.10.3.4 PWM Output

Taking PWM01 as an example, Figure 9-83 indicates the PWM01 output logic diagram. The logic diagrams of other PWM pairs are the same as PWM01.

The timer logic module of PWM consists of one 16-bit up-counter ([PCNTR](#)) and three 16-bit parameters ([PWM\\_ENTIRE\\_CYCLE](#), [PWM\\_ACT\\_CYCLE](#), [PWM\\_COUNTER\\_START](#)). The [PWM\\_ENTIRE\\_CYCLE](#) is used to



control the PWM cycle, the [PWM\\_ACT\\_CYCLE](#) is used to control the duty-cycle, the [PWM\\_COUNTER\\_START](#) is used to control the output phase (multi-channel synchronization work requirements).

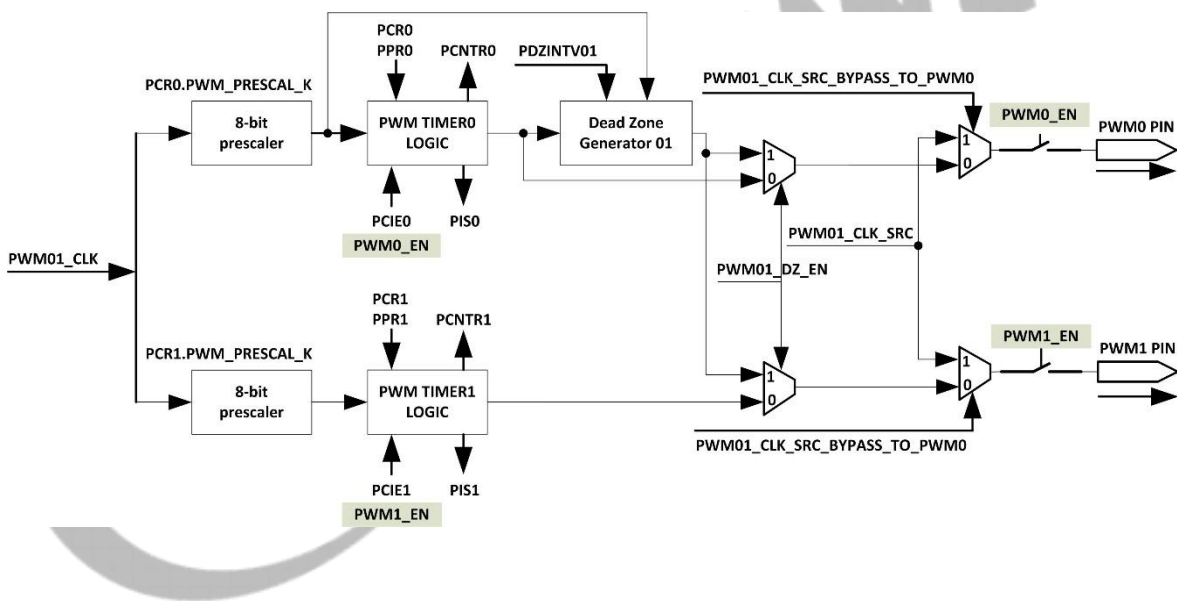
The [PWM\\_ENTIRE\\_CYCLE](#) and the [PWM\\_ACT\\_CYCLE](#) support the cache load, after PWM output is enabled, the register values of the [PWM\\_ENTIRE\\_CYCLE](#) and the [PWM\\_ACT\\_CYCLE](#) can be changed anytime, the changed value caches into the cache register. When the PCNTR counter outputs a period of PWM waveform, the value of the cache register can be updated for the PCNTR control. The purpose of the cache load is to avoid the unstable PWM output waveform with the burred feature when updating the values of the [PWM\\_ENTIRE\\_CYCLE](#) and [PWM\\_ACT\\_CYCLE](#).

The PWM supports cycle and pulse waveform output.

**Cycle mode:** The PWM outputs the setting PWM waveform continually, that is, the output waveform is a continuous PWM square wave.

**Pulse mode:** After setting the [PWM\\_PUL\\_NUM](#) parameter, the PWM outputs (PWM\_PULNUM+1) periods of PWM waveform, that is, the waveform with several pulses are output.

Figure 9-83 PWM01 Output Logic Module Diagram



### 9.10.3.5 Up-Counter and Comparator

The period, duty-cycle, and phase of PWM output waveform are decided by the [PCNTR](#), [PWM\\_ENTIRE\\_CYCLE](#), [PWM\\_ACT\\_CYCLE](#), and [PWM\\_COUNTER\\_START](#). The rules are as follows.

- $PCNTR = (PCNTR == PWM\_ENTIRE\_CYCLE) ? 0 : PCNTR + 1$
- PCNTR starts to count by [PWM\\_COUNTER\\_START](#), the counter of a PWM period is (PWM\_ENTIRE\_CYCLE+1).
- $PCNTR > (PWM\_ENTIRE\_CYCLE - PWM\_ACT\_CYCLE)$ , output “active state”

- $PCNTR \leq (PWM\_ENTIRE\_CYCLE - PWM\_ACT\_CYCLE)$ , output “~ (active state)”

**Active state of PWM0 channel is high level (PCR0. PWM\_ACT\_STA = 1)**

When  $PCNTR0 > (PPR0.PWM\_ENTIRE\_CYCLE - PPR0.PWM\_ACT\_CYCLE)$ , then PWM0 outputs 1 (high level).

When  $PCNTR0 \leq (PPR0.PWM\_ENTIRE\_CYCLE - PPR0.PWM\_ACT\_CYCLE)$ , then PWM0 outputs 0 (low level).

The formula of the output period and the duty-cycle for PWM are as follows.

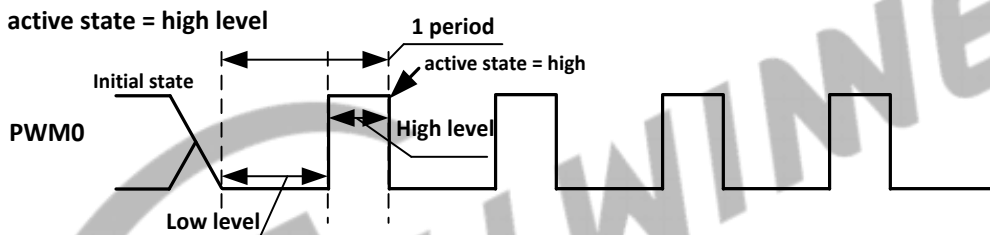
$$T_{period} = (PWM01\_CLK / PWM0\_PRESCALE\_K)^{-1} * (PPR0.PWM\_ENTIRE\_CYCLE + 1)$$

$$T_{high-level} = (PWM01\_CLK / PWM0\_PRESCALE\_K)^{-1} * PPR0.PWM\_ACT\_CYCLE$$

$$T_{low-level} = (PWM01\_CLK / PWM0\_PRESCALE\_K)^{-1} * (PPR0.PWM\_ENTIRE\_CYCLE + 1 - PPR0.PWM\_ACT\_CYCLE)$$

$$Duty-cycle = (high\ level\ time) / (1\ period\ time) = T_{high-level} / T_{period}$$

**Figure 9-84 PWM0 High Level Active State**



**Active state of PWM0 channel is low level (PCR0. PWM\_ACT\_STA = 0)**

When  $PCNTR0 > (PPR0.PWM\_ENTIRE\_CYCLE - PPR0.PWM\_ACT\_CYCLE)$ , then PWM0 outputs 0.

When  $PCNTR0 \leq (PPR0.PWM\_ENTIRE\_CYCLE - PPR0.PWM\_ACT\_CYCLE)$ , then PWM0 outputs 1.

The formula of the output period and the duty-cycle for PWM are as follows.

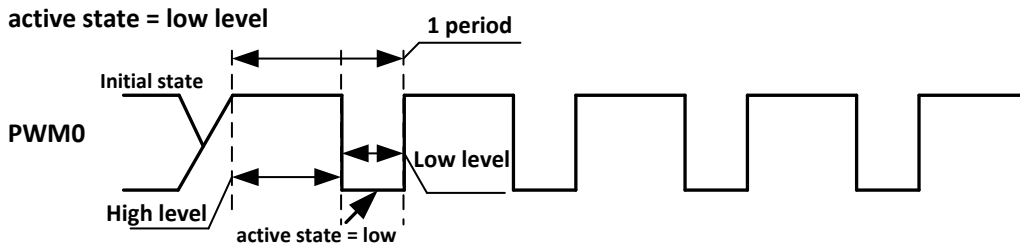
$$T_{period} = (PWM01\_CLK / PWM0\_PRESCALE\_K)^{-1} * (PPR0.PWM\_ENTIRE\_CYCLE + 1)$$

$$T_{high-level} = (PWM01\_CLK / PWM0\_PRESCALE\_K)^{-1} * (PPR0.PWM\_ENTIRE\_CYCLE + 1 - PPR0.PWM\_ACT\_CYCLE)$$

$$T_{low-level} = (PWM01\_CLK / PWM0\_PRESCALE\_K)^{-1} * PPR0.PWM\_ACT\_CYCLE$$

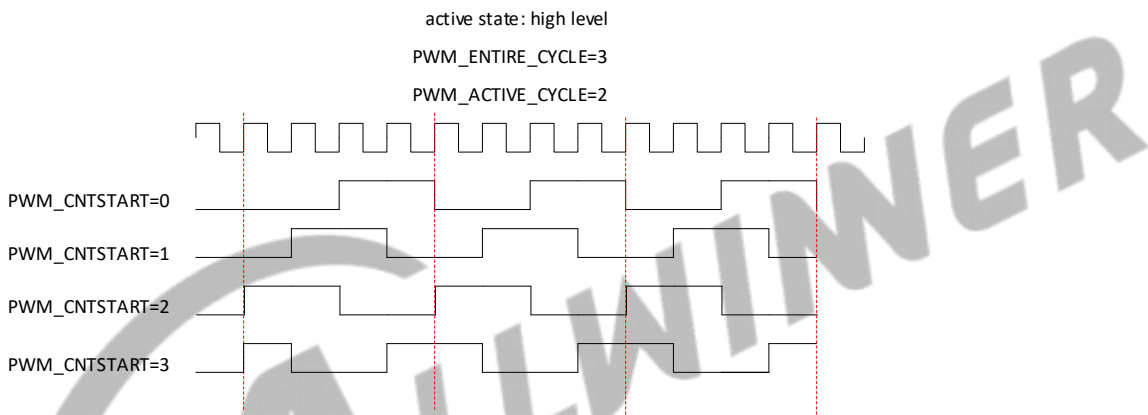
$$Duty-cycle = (low\ level\ time) / (1\ period\ time) = T_{low-level} / T_{period}$$

Figure 9-85 PWM0 Low Level Active State



The counter of PCNTR starts from 0 by default, it can output the pulse control of the waveform by setting [PWM\\_COUNTER\\_START](#). The figure is as follows.

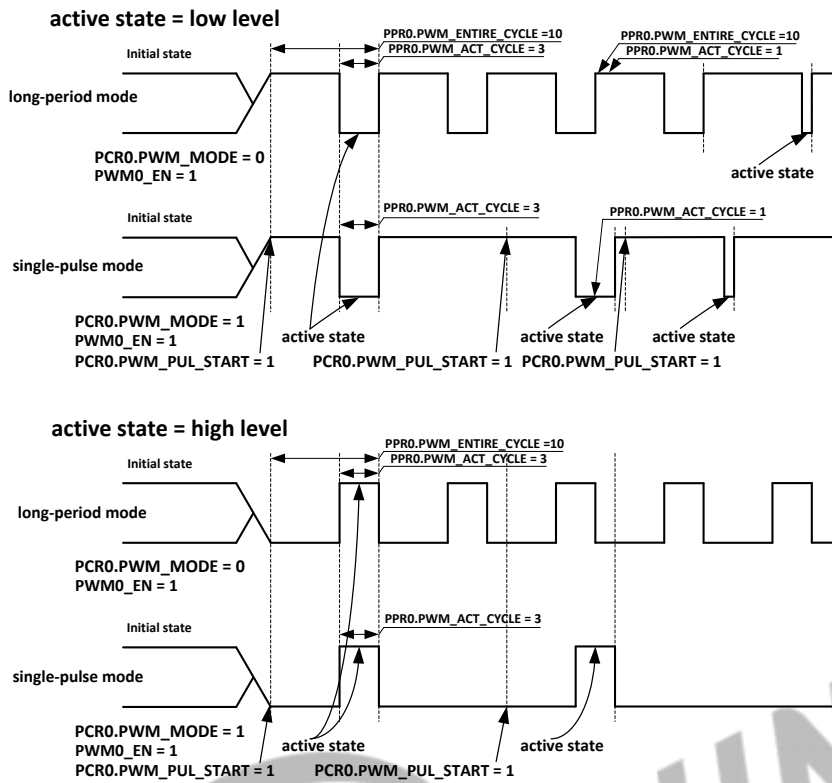
Figure 9-86 Phase of PWM0 High Level Active State



### 9.10.3.6 Pulse Mode and Cycle Mode

The PWM output supports pulse mode and cycle mode. PWM in pulse mode outputs one pulse waveform, but PWM in cycle mode outputs continuous waveform. Figure 9-87 shows the PWM output waveform in pulse mode and cycle mode.

Figure 9-87 PWM0 Output Waveform in Pulse Mode and Cycle Mode



Each channel of the PWM module supports the PWM output of pulse mode and cycle mode, the active state of the PWM output waveform can be programmed to control.

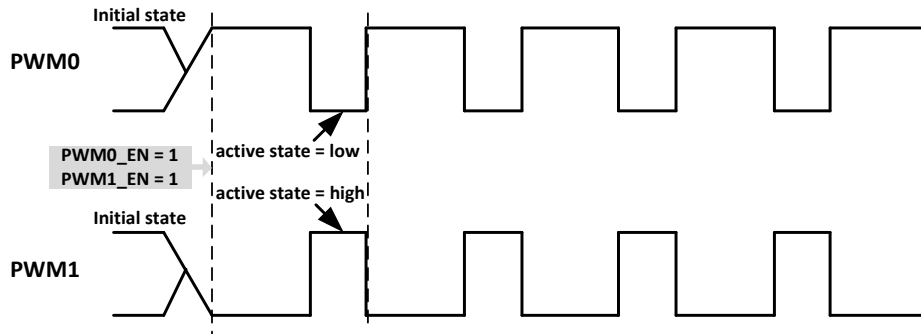
When [PCR0\[PWM\\_MODE\]](#) is 0, the PWM0 outputs in cycle mode. When [PCR0\[PWM\\_MODE\]](#) is 1, the PWM0 outputs in pulse mode.

Specifically, in pulse mode, after the PWM0 channel enabled, [PCR0\[PWM\\_PUL\\_START\]](#) needs to be set to 1 when the PWM0 needs to output pulse waveform, after completed the output, [PCR0\[PWM\\_PUL\\_START\]](#) can be cleared to 0 by hardware. The next setting 1 can be operated after [PCR0\[PWM\\_PUL\\_START\]](#) is cleared.

### 9.10.3.7 Complementary Pair Output

Every PWM pair supports complementary pair output and PWM pair with dead-time. Figure 9-88 shows the complementary pair output of PWM01.

Figure 9-88 PWM01 Complementary Pair Output



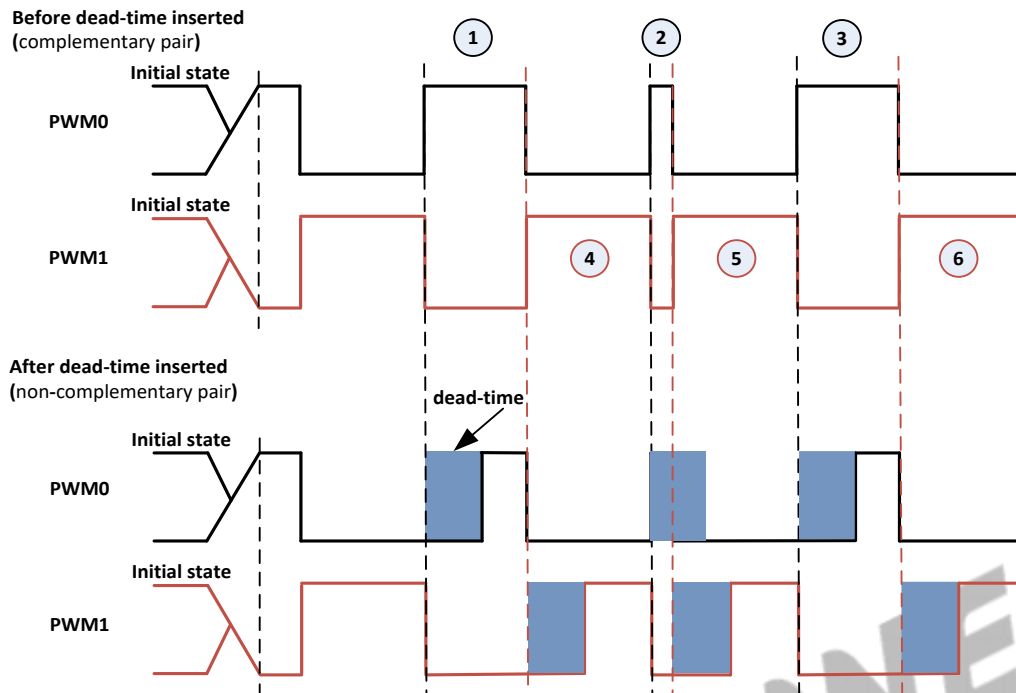
The complementary pair output needs to satisfy the following conditions:

- PWM0 and PWM1 have the same clock divider, frequency, duty-cycle, and phase
- PWM0 and PWM1 have an opposite active state
- Enable the clock gating of PWM0 and PWM1 at the same time
- Enable the waveform output of PWM0 and PWM1 at the same time

#### 9.10.3.8 Dead-time Generator

Every PWM pair has a programmable dead-time generator. When the dead-time function of the PWM pair enabled, the PWM01 output waveform is decided by PWM timer logic and DeadZone Generator. Figure 9-89 shows the output waveform.

Figure 9-89 Dead-time Output Waveform



The PWM waveform before the insertion of dead-time indicates a complementary waveform pair of non-inserted dead-time in Dead Zone Generator 01.

The PWM waveform after the insertion of dead-time indicates a non-complementary PWM waveform pair inserted dead-time in a complementary waveform pair of Dead Zone Generator 01. The PWM waveform pair at last outputs to PWM0 pin and PWM1 pin.

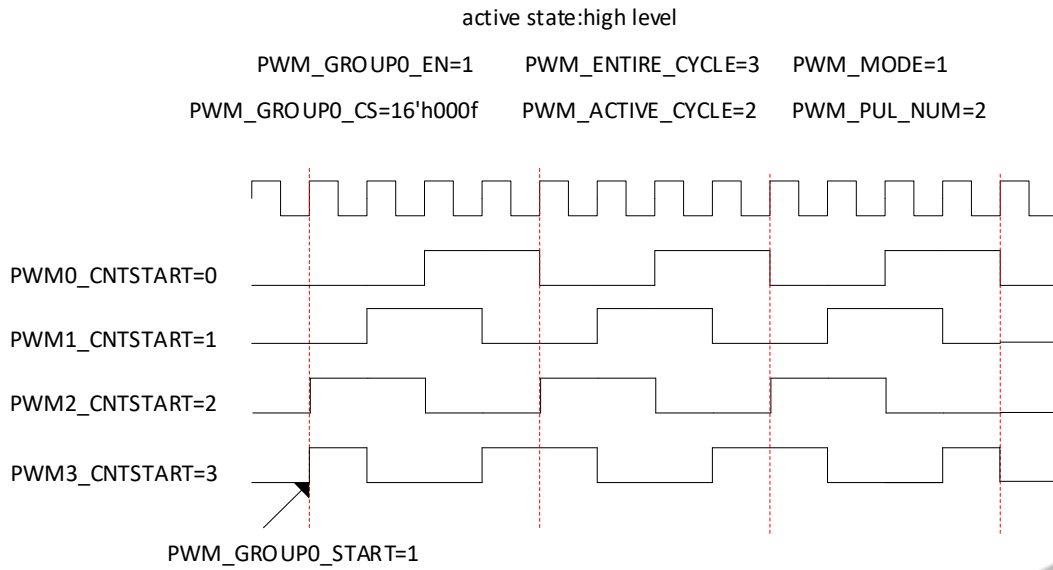
For the complementary pair of Dead Zone Generator 01, the principle of inserting dead-time is that to insert dead-time as soon as the rising edge came. If the high level time for mark ② in the above figure is less than dead-time, then dead-time will override the high level. The setting of dead-time needs to consider the period and the duty-cycle of the output waveform. The dead-time formula is defined as follows:

$$\text{Dead-time} = (\text{PWM01\_CLK} / \text{PWM0\_PRESCALE\_K})^{-1} * \text{PDZINTV01}$$

### 9.10.3.9 PWM Group Mode

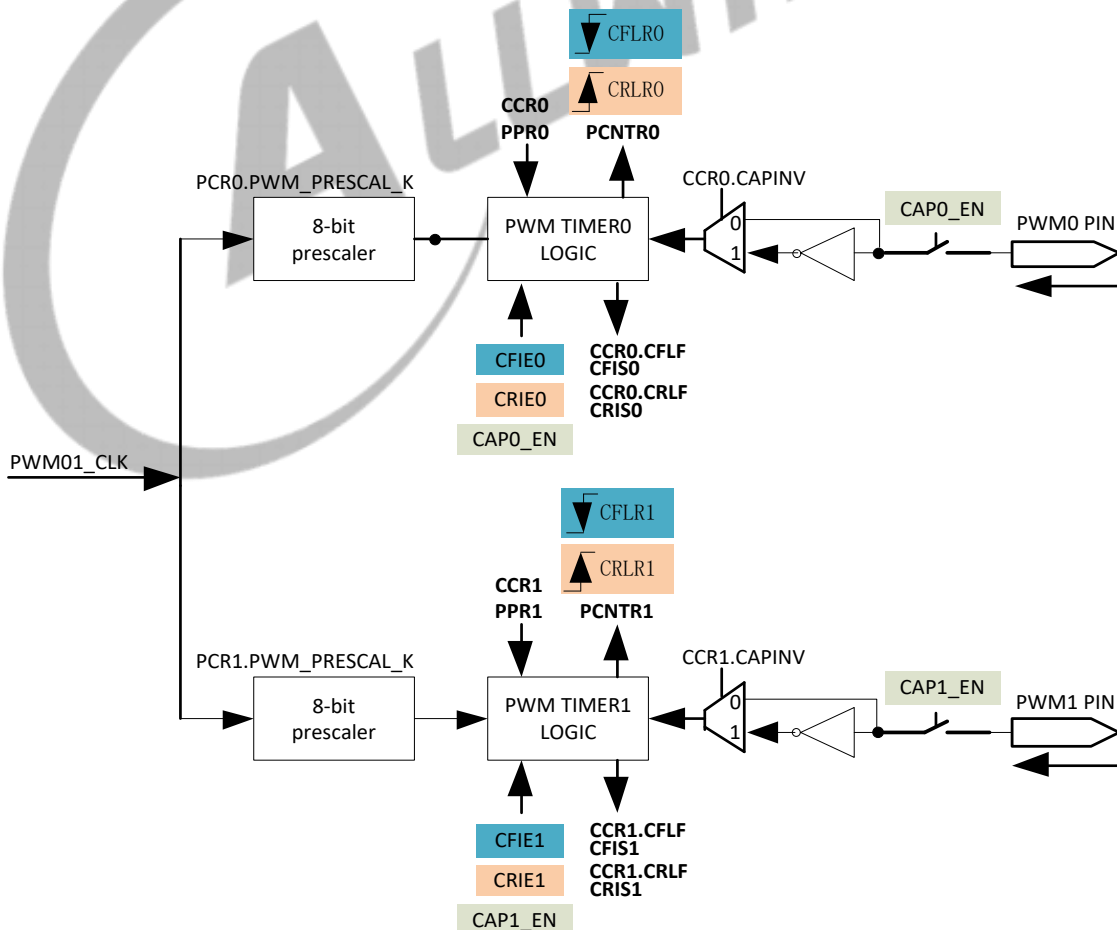
Taking PWM Group0 as an example. The same group of PWM channel is selected to work by PGR0.CS; the same [PWM\\_ENTIRE\\_CYCLE](#), [PWM\\_ACT\\_CYCLE](#) are set by the same clock configuration; the different [PWM\\_COUNTER\\_START](#) can output PWM group signals with the same duty-cycle and the different phase.

Figure 9-90 Group 0–3 PWM Signal Output



9.10.3.10 Capture Input

Figure 9-91 PWM01 Capture Logic Module Diagram



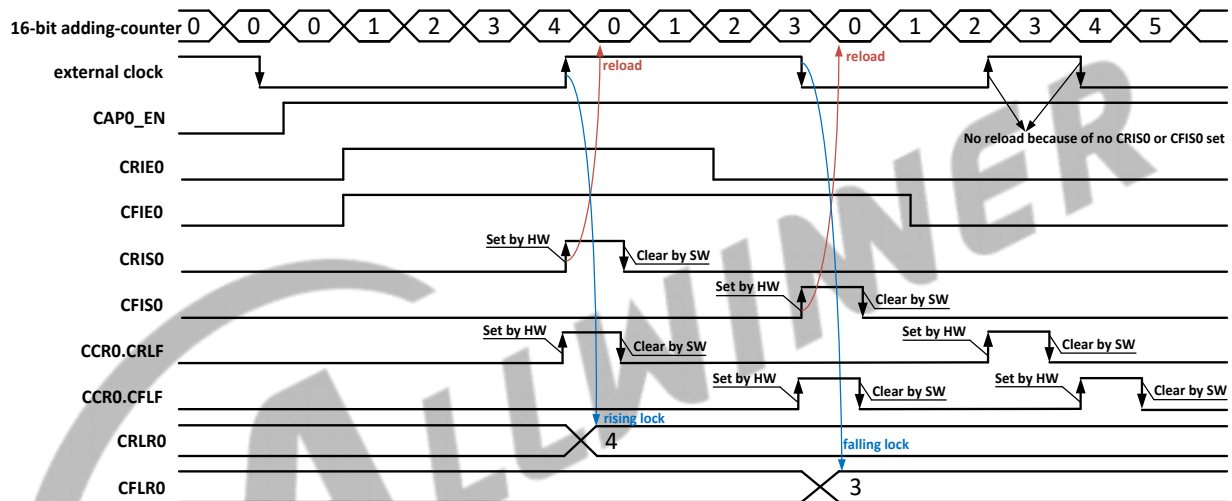
Besides the timer logic module of every PWM channel generates PWM output, it can be used to capture the rising edge and the falling edge of the external clock. Using the PWM0 channel as an example, the PWM0 channel has one **CFLR0** and one **CRLR0** for capturing up-counter value on the falling edge and rising edge, respectively. You can calculate the period of the external clock by **CFLR0** and **CRLR0**.

$$T_{\text{high-level}} = (\text{PWM01\_CLK} / \text{PWM0\_PRESCALE\_K})^{-1} * \text{CRLR0}$$

$$T_{\text{low-level}} = (\text{PWM01\_CLK} / \text{PWM0\_PRESCALE\_K})^{-1} * \text{CFLR0}$$

$$T_{\text{period}} = T_{\text{high-level}} + T_{\text{low-level}}$$

Figure 9-92 PWM0 Channel Capture Timing



When the capture input function of the PWM0 channel is enabled, the PCNTR of the PWM0 channel starts to work.

When the timer logic module of PWM0 captures a rising edge, the current value of the up-counter is locked to **CRLR0** and **CCR0[CRLF]** is set to 1. If **CRIE0** is 1, then **CRISO** is set to 1, the PWM0 channel sends interrupt requests, and the up-counter is loaded to 0 and continues to count. If **CRIE0** is 0, the timer logic module of PWM0 captures a rising edge, **CRISO** cannot be set to 1, the up-counter is not loaded to 0.

When the timer logic module of PWM0 captures one falling edge, the current value of PCNTR is locked to **CFLR0** and **CCR0[CFLF]** is set to 1. If **CFIE0** is 1, then **CFISO** is set to 1, the PWM0 channel sends interrupt requests, and the up-counter is loaded to 0 and continues to count. If **CFIE0** is 0, the timer logic module of PWM0 captures a falling edge, **CFISO** cannot be set to 1, the up-counter is not loaded to 0.



### 9.10.3.11 Interrupt

The PWM supports an interrupt generation when configuring the PWM channel to PWM output or capturing input.

For PWM output function, when the controller outputs one period of PWM waveform in cycle mode, the PIS of the corresponding PWM channel is set to 1; when the controller outputs (PWM\_PULNUM+1) periods of PWM waveform in pulse mode, the PIS of the corresponding PWM channel is set to 1.

---

 **NOTE**

The PIS bit is set to 1 automatically by hardware and cleared by software.

---

For capturing input function, when the timer logic module of the capture channel0 captures rising edge, and [CRIEO](#) is 1, then [CRISO](#) is set to 1; when the timer logic module of the capture channel0 captures falling edge, and [CFIEO](#) is 1, then [CFISO](#) is set to 1.

## 9.10.4 Programming Guidelines

The following working mode takes PWM01 as an example, other PWM pairs and PWM01 are consistent.

### 9.10.4.1 Configuring Clock

- Step 1** PWM gating: When using PWM, write 1 to [PCGR](#)[PWMx\_CLK\_GATING].
- Step 2** PWM clock source select: Set [PCCR01](#)[PWM01\_CLK\_SRC] to select HOSC or APB0 clock.
- Step 3** PWM clock divider: Set [PCCR01](#)[PWM01\_CLK\_DIV\_M] to select different frequency division coefficient (1/2/4/8/16/32/64/128/256).
- Step 4** PWM clock bypass: Set [PCGR](#)[PWM\_CLK\_SRC\_BYPASS\_TO\_PWM] to 1, output the PWM clock after the secondary frequency division to the corresponding PWM output pin.
- Step 5** PWM internal clock configuration: Set [PCR](#)[PWM\_PRESCAL\_K] to select any frequency division coefficient from 1 to 256.

---

 **NOTE**

For the channel of complementary output and group mode, firstly, set the same clock configurations (clock source selects APB0, clock division configures the same division factor); secondly, open clock gating at the same time; thirdly, configure PWM parameters; finally, enable PWM output at the same time to ensure each channel sync.

We suggest that the two channels of the same PWM pair cannot subject to two groups because of they have the same first level clock division and gating. If must allocate based on this way, the first level of clock division of the channel used by all groups needs to set to the same coefficient and open gating at the same time. And the total module needs to be reset when the group mode regroup.

#### 9.10.4.2 Configuring PWM

- Step 1** PWM mode: Set [PCR](#)[PWM\_MODE] to select cycle mode or pulse mode, if pulse mode, [PCR](#)[PWM\_PUL\_NUM] needs to be configured.
- Step 2** PWM active level: Set [PCR](#)[PWM\_ACT\_STA] to select a low level or high level.
- Step 3** PWM duty-cycle: Configure [PPR](#)[PWM\_ENTIRE\_CYCLE] and [PPR](#)[PWM\_ACT\_CYCLE] after clock gating is opened.
- Step 4** PWM starting/stopping phase: Configure [PCNTR](#)[PWM\_COUNTER\_START] after the clock gating is enabled and before the PWM is enabled. You can verify whether the configuration was successful by reading back [PCNTR](#)[PWM\_COUNTER\_STATUS].
- Step 5** Enable PWM: Configure PER to select the corresponding PWM enable bit; when selecting pulse mode, [PCR](#)[PWM\_PUL\_START] needs to be enabled.

#### 9.10.4.3 Configuring Deadzone

- Step 1** Set initial value: set [PDZINTV01].
- Step 2** Enable Deadzone: set [PWM01\_DZ\_CN].

#### 9.10.4.4 Configuring Capture Input

- Step 1** Enable capture: Configure [CER](#) to enable the corresponding channel.
- Step 2** Capture mode: Configure [CCR](#)[CRLF] and [CCR](#)[CFLF] to select rising edge capture or falling edge capture, configure [CCR](#)[CAPINV] to select whether the input signal does reverse processing.

#### 9.10.5 Register List

Module Name	Base Address
PWM	0x02000C00

Register Name	Offset	Description
PIER	0x0000	PWM IRQ Enable Register
PISR	0x0004	PWM IRQ Status Register
CIER	0x0010	Capture IRQ Enable Register
CISR	0x0014	Capture IRQ Status Register
PCCR01	0x0020	PWM01 Clock Configuration Register
PCCR23	0x0024	PWM23 Clock Configuration Register
PCCR45	0x0028	PWM45 Clock Configuration Register
PCCR67	0x002C	PWM67 Clock Configuration Register
PCGR	0x0040	PWM Clock Gating Register
PDZCR01	0x0060	PWM01 Dead Zone Control Register
PDZCR23	0x0064	PWM23 Dead Zone Control Register
PDZCR45	0x0068	PWM45 Dead Zone Control Register
PDZCR67	0x006C	PWM67 Dead Zone Control Register
PER	0x0080	PWM Enable Register
PGR0	0x0090	PWM Group0 Register
PGR1	0x0094	PWM Group1 Register
PGR2	0x0098	PWM Group2 Register
PGR3	0x009C	PWM Group3 Register
CER	0x00C0	Capture Enable Register
PCR	0x0100+0x0000+N*0x0020 (N= 0-7)	PWM Control Register
PPR	0x0100+0x0004+N*0x0020 (N= 0-7)	PWM Period Register
PCNTR	0x0100+0x0008+N*0x0020 (N= 0-7)	PWM Count Register
PPCNTR	0x0100+0x000C+N*0x0020 (N= 0-7)	PWM Pulse Count Register
CCR	0x0100+0x0010+N*0x0020 (N= 0-7)	Capture Control Register
CRLR	0x0100+0x0014+N*0x0020 (N= 0-7)	Capture Rise Lock Register
CFLR	0x0100+0x0018+N*0x0020 (N= 0-7)	Capture Fall Lock Register

### 9.10.6 Register Description

#### 9.10.6.1 0x0000 PWM IRQ Enable Register (Default Value: 0x0000\_0000)

Offset:0x0000			Register Name: PIER
Bit	Read/Write	Default/Hex	Description
31:20	/	/	/
19	R/W	0x0	PGIE3 PWM Group 3 Interrupt Enable 0: Disable 1: Enable
18	R/W	0x0	PGIE2 PWM Group 2 Interrupt Enable 0: Disable 1: Enable
17	R/W	0x0	PGIE1 PWM Group 1 Interrupt Enable 0: Disable 1: Enable
16	R/W	0x0	PGIE0 PWM Group 0 Interrupt Enable 0: Disable 1: Enable
15:8	/	/	/
7	R/W	0x0	PCIE7 PWM Channel 7 Interrupt Enable 0: PWM Channel 7 Interrupt Disable 1: PWM Channel 7 Interrupt Enable
6	R/W	0x0	PCIE6 PWM Channel 6 Interrupt Enable 0: PWM Channel 6 Interrupt Disable 1: PWM Channel 6 Interrupt Enable
5	R/W	0x0	PCIE5 PWM Channel 5 Interrupt Enable 0: PWM Channel 5 Interrupt Disable 1: PWM Channel 5 Interrupt Enable

Offset:0x0000			Register Name: PIER
Bit	Read/Write	Default/Hex	Description
4	R/W	0x0	PCIE4 PWM Channel 4 Interrupt Enable 0: PWM Channel 4 Interrupt Disable 1: PWM Channel 4 Interrupt Enable
3	R/W	0x0	PCIE3 PWM Channel 3 Interrupt Enable 0: PWM Channel 3 Interrupt Disable 1: PWM Channel 3 Interrupt Enable
2	R/W	0x0	PCIE2 PWM Channel 2 Interrupt Enable 0: PWM Channel 2 Interrupt Disable 1: PWM Channel 2 Interrupt Enable
1	R/W	0x0	PCIE1 PWM Channel 1 Interrupt Enable 0: PWM Channel 1 Interrupt Disable 1: PWM Channel 1 Interrupt Enable
0	R/W	0x0	PCIE0 PWM Channel 0 Interrupt Enable 0: PWM Channel 0 Interrupt Disable 1: PWM Channel 0 Interrupt Enable

**9.10.6.2 0x0004 PWM IRQ Status Register (Default Value: 0x0000\_0000)**

Offset:0x0004			Register Name: PISR
Bit	Read/Write	Default/Hex	Description
31:20	/	/	/
19	R/W1C	0x0	PGIS3 PWM Group 3 Interrupt Status
18	R/W1C	0x0	PGIS2 PWM Group 2 Interrupt Status
17	R/W1C	0x0	PGIS1 PWM Group 1 Interrupt Status
16	R/W1C	0x0	PGIS0 PWM Group 0 Interrupt Status

Offset:0x0004			Register Name: PISR
Bit	Read/Write	Default/Hex	Description
15:8	/	/	/
7	R/W1C	0x0	<p>PIS7</p> <p>PWM Channel 7 Interrupt Status</p> <p>When the PWM channel 7 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: PWM channel 7 interrupt is not pending.</p> <p>Reads 1: PWM channel 7 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear PWM channel 7 interrupt status.</p>
6	R/W1C	0x0	<p>PIS6</p> <p>PWM Channel 6 Interrupt Status</p> <p>When the PWM channel 6 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: PWM channel 6 interrupt is not pending.</p> <p>Reads 1: PWM channel 6 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear PWM channel 6 interrupt status.</p>
5	R/W1C	0x0	<p>PIS5</p> <p>PWM Channel 5 Interrupt Status</p> <p>When the PWM channel 5 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: PWM channel 5 interrupt is not pending.</p> <p>Reads 1: PWM channel 5 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear PWM channel 5 interrupt status.</p>
4	R/W1C	0x0	<p>PIS4</p> <p>PWM Channel 4 Interrupt Status</p> <p>When the PWM channel 4 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: PWM channel 4 interrupt is not pending.</p> <p>Reads 1: PWM channel 4 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear PWM channel 4 interrupt status.</p>

Offset:0x0004			Register Name: PISR
Bit	Read/Write	Default/Hex	Description
3	R/W1C	0x0	<p>PIS3</p> <p>PWM Channel 3 Interrupt Status</p> <p>When the PWM channel 3 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: PWM channel 3 interrupt is not pending.</p> <p>Reads 1: PWM channel 3 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear PWM channel 3 interrupt status.</p>
2	R/W1C	0x0	<p>PIS2</p> <p>PWM Channel 2 Interrupt Status</p> <p>When the PWM channel 2 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: PWM channel 2 interrupt is not pending.</p> <p>Reads 1: PWM channel 2 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear PWM channel 2 interrupt status.</p>
1	R/W1C	0x0	<p>PIS1</p> <p>PWM Channel 1 Interrupt Status</p> <p>When the PWM channel 1 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: PWM channel 1 interrupt is not pending.</p> <p>Reads 1: PWM channel 1 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear PWM channel 1 interrupt status.</p>
0	R/W1C	0x0	<p>PIS0</p> <p>PWM Channel 0 Interrupt Status</p> <p>When the PWM channel 0 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: PWM channel 0 interrupt is not pending.</p> <p>Reads 1: PWM channel 0 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear PWM channel 0 interrupt status.</p>

9.10.6.3 0x0010 PWM Capture IRQ Enable Register (Default Value: 0x0000\_0000)

Offset:0x0010			Register Name: CIER
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15	R/W	0x0	<p>CFIE7</p> <p>If the enable bit is set to 1, when the capture channel 7 captures falling edge, it generates a capture channel 7 pending.</p> <p>0: Capture channel 7 fall lock interrupt disable 1: Capture channel 7 fall lock interrupt enable</p>
14	R/W	0x0	<p>CRIE7</p> <p>If the enable bit is set to 1, when the capture channel 7 captures rising edge, it generates a capture channel 7 pending.</p> <p>0: Capture channel 7 rise lock interrupt disable 1: Capture channel 7 rise lock interrupt enable</p>
13	R/W	0x0	<p>CFIE6</p> <p>If the enable bit is set to 1, when the capture channel 6 captures falling edge, it generates a capture channel 6 pending.</p> <p>0: Capture channel 6 fall lock interrupt disable 1: Capture channel 6 fall lock interrupt enable</p>
12	R/W	0x0	<p>CRIE6</p> <p>If the enable bit is set to 1, when the capture channel 6 captures rising edge, it generates a capture channel 6 pending.</p> <p>0: Capture channel 6 rise lock interrupt disable 1: Capture channel 6 rise lock interrupt enable</p>
11	R/W	0x0	<p>CFIE5</p> <p>If the enable bit is set to 1, when the capture channel 5 captures falling edge, it generates a capture channel 5 pending.</p> <p>0: Capture channel 5 fall lock interrupt disable 1: Capture channel 5 fall lock interrupt enable</p>
10	R/W	0x0	<p>CRIE5</p> <p>If the enable bit is set to 1, when the capture channel 5 captures rising edge, it generates a capture channel 5 pending.</p> <p>0: Capture channel 5 rise lock interrupt disable 1: Capture channel 5 rise lock interrupt enable</p>



Offset:0x0010			Register Name: CIER
Bit	Read/Write	Default/Hex	Description
9	R/W	0x0	<p>CFIE4</p> <p>If the enable bit is set to 1, when the capture channel 4 captures falling edge, it generates a capture channel 4 pending.</p> <p>0: Capture channel 4 fall lock interrupt disable 1: Capture channel 4 fall lock interrupt enable</p>
8	R/W	0x0	<p>CRIE4</p> <p>If the enable bit is set to 1, when the capture channel 4 captures rising edge, it generates a capture channel 4 pending.</p> <p>0: Capture channel 4 rise lock interrupt disable 1: Capture channel 4 rise lock interrupt enable</p>
7	R/W	0x0	<p>CFIE3</p> <p>If the enable bit is set to 1, when the capture channel 3 captures falling edge, it generates a capture channel 3 pending.</p> <p>0: Capture channel 3 fall lock interrupt disable 1: Capture channel 3 fall lock interrupt enable</p>
6	R/W	0x0	<p>CRIE3</p> <p>If the enable bit is set to 1, when the capture channel 3 captures rising edge, it generates a capture channel 3 pending.</p> <p>0: Capture channel 3 rise lock interrupt disable 1: Capture channel 3 rise lock interrupt enable</p>
5	R/W	0x0	<p>CFIE2</p> <p>If the enable bit is set to 1, when the capture channel 2 captures falling edge, it generates a capture channel 2 pending.</p> <p>0: Capture channel 2 fall lock interrupt disable 1: Capture channel 2 fall lock interrupt enable</p>
4	R/W	0x0	<p>CRIE2</p> <p>If the enable bit is set to 1, when the capture channel 2 captures rising edge, it generates a capture channel 2 pending.</p> <p>0: Capture channel 2 rise lock interrupt disable 1: Capture channel 2 rise lock interrupt enable</p>
3	R/W	0x0	<p>CFIE1</p> <p>If the enable bit is set to 1, when the capture channel 1 captures falling edge, it generates a capture channel 1 pending.</p> <p>0: Capture channel 1 fall lock interrupt disable 1: Capture channel 1 fall lock interrupt enable</p>

Offset:0x0010			Register Name: CIER
Bit	Read/Write	Default/Hex	Description
2	R/W	0x0	<p>CRIE1</p> <p>If the enable bit is set to 1, when the capture channel 1 captures rising edge, it generates a capture channel 1 pending.</p> <p>0: Capture channel 1 rise lock interrupt disable</p> <p>1: Capture channel 1 rise lock interrupt enable</p>
1	R/W	0x0	<p>CFIE0</p> <p>If the enable bit is set to 1, when the capture channel 0 captures falling edge, it generates a capture channel 0 pending.</p> <p>0: Capture channel 0 fall lock interrupt disable</p> <p>1: Capture channel 0 fall lock interrupt enable</p>
0	R/W	0x0	<p>CRIE0</p> <p>If the enable bit is set to 1, when the capture channel 0 captures rising edge, it generates a capture channel 0 pending.</p> <p>0: Capture channel 0 rise lock interrupt disable</p> <p>1: Capture channel 0 rise lock interrupt enable</p>

9.10.6.4 0x0014 PWM Capture IRQ Status Register (Default Value: 0x0000\_0000)

Offset:0x0014			Register Name: CISR
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
15	R/W1C	0x0	<p>CFIS7</p> <p>Status of the capture channel 7 falling lock interrupt</p> <p>When the capture channel 7 captures falling edge, if the fall lock interrupt (<a href="#">CFIE7</a>) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: The capture channel 7 interrupt is not pending.</p> <p>Reads 1: The capture channel 7 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear the status of the capture channel 7 interrupt.</p>

Offset:0x0014			Register Name: CISR
Bit	Read/Write	Default/Hex	Description
14	R/W1C	0x0	<p>CRIS7</p> <p>Status of the capture channel 7 rising lock interrupt</p> <p>When the capture channel 7 captures rising edge, if the rise lock interrupt (<b>CRIE7</b>) is enabled, this bit is set to 1 by hardware. Write 1 to clear this bit.</p> <p>Reads 0: The capture channel 7 interrupt is not pending.</p> <p>Reads 1: The capture channel 7 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear the status of the capture channel 7 interrupt.</p>
13	R/W1C	0x0	<p>CFIS6</p> <p>Status of the capture channel 6 falling lock interrupt</p> <p>When the capture channel 6 captures falling edge, if the fall lock interrupt (<b>CFIE6</b>) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: The capture channel 6 interrupt is not pending.</p> <p>Reads 1: The capture channel 6 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear the status of the capture channel 6 interrupt.</p>
12	R/W1C	0x0	<p>CRIS6</p> <p>Status of the capture channel 6 rising lock interrupt.</p> <p>When the capture channel 6 captures rising edge, if the rise lock interrupt (<b>CRIE6</b>) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: The capture channel 6 interrupt is not pending.</p> <p>Reads 1: The capture channel 6 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear the status of the capture channel 6 interrupt.</p>
11	R/W1C	0x0	<p>CFIS5</p> <p>Status of the capture channel 5 falling lock interrupt</p> <p>When the capturing channel 5 captures falling edge, if the fall lock interrupt (<b>CFIE5</b>) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: The capture channel 5 interrupt is not pending.</p> <p>Reads 1: The capture channel 5 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear the status of the capture channel 5 interrupt.</p>

Offset:0x0014			Register Name: CISR
Bit	Read/Write	Default/Hex	Description
10	R/W1C	0x0	<p>CRIS5</p> <p>Status of the capture channel 5 rising lock interrupt</p> <p>When the capture channel 5 captures rising edge, if the rise lock interrupt (<b>CRIE5</b>) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: The capture channel 5 interrupt is not pending.</p> <p>Reads 1: The capture channel 5 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear the status of the capture channel 5 interrupt.</p>
9	R/W1C	0x0	<p>CFIS4</p> <p>Status of the capture channel 4 falling lock interrupt</p> <p>When the capture channel 4 captures falling edge, if the fall lock interrupt (<b>CFIE4</b>) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: The capture channel 4 interrupt is not pending.</p> <p>Reads 1: The capture channel 4 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear the status of the capture channel 4 interrupt.</p>
8	R/W1C	0x0	<p>CRIS4</p> <p>Status of the capture channel 4 rising lock interrupt.</p> <p>When the capture channel 4 captures rising edge, if the rise lock interrupt (<b>CRIE4</b>) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: The capture channel 4 interrupt is not pending.</p> <p>Reads 1: The capture channel 4 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear the status of the capture channel 4 interrupt status.</p>
7	R/W1C	0x0	<p>CFIS3</p> <p>Status of the capture channel 3 falling lock interrupt.</p> <p>When the capture channel 3 captures falling edge, if the fall lock interrupt (<b>CFIE3</b>) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: The capture channel 3 interrupt is not pending.</p> <p>Reads 1: The capture channel 3 interrupt is pending.</p> <p>Writes 0: no effect.</p> <p>Writes 1: Clear the status of the capture channel 3 interrupt.</p>

Offset:0x0014			Register Name: CISR
Bit	Read/Write	Default/Hex	Description
6	R/W1C	0x0	<p>CRIS3</p> <p>Status of the capture channel 3 rising lock interrupt</p> <p>When the capture channel 3 captures rising edge, if the rise lock interrupt (<b>CRIE3</b>) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: The capture channel 3 interrupt is not pending.</p> <p>Reads 1: The capture channel 3 interrupt is pending.</p> <p>Writes 0: no effect.</p> <p>Writes 1: Clear the status of the capture channel 3 interrupt.</p>
5	R/W1C	0x0	<p>CFIS2</p> <p>Status of the capture channel 2 falling lock interrupt</p> <p>When the capture channel 2 captures falling edge, if the fall lock interrupt (<b>CFIE2</b>) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: The capture channel 2 interrupt is not pending.</p> <p>Reads 1: The capture channel 2 interrupt is pending.</p> <p>Writes 0: no effect.</p> <p>Writes 1: Clear the status of the capture channel 2 interrupt.</p>
4	R/W1C	0x0	<p>CRIS2</p> <p>Status of the capture channel 2 rising lock interrupt.</p> <p>When the capture channel 2 captures rising edge, if the rise lock interrupt (<b>CRIE2</b>) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: The capture channel 2 interrupt is not pending.</p> <p>Reads 1: The capture channel 2 interrupt is pending.</p> <p>Writes 0: no effect.</p> <p>Writes 1: Clear the status of the capture channel 2 interrupt.</p>
3	R/W1C	0x0	<p>CFIS1</p> <p>Status of the capture channel 1 falling lock interrupt</p> <p>When the capture channel 1 captures falling edge, if the fall lock interrupt (<b>CFIE1</b>) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: The capture channel 1 interrupt is not pending.</p> <p>Reads 1: The capture channel 1 interrupt is pending.</p> <p>Writes 0: no effect.</p> <p>Writes 1: Clear the status of the capture channel 1 interrupt.</p>

Offset:0x0014			Register Name: CISR
Bit	Read/Write	Default/Hex	Description
2	R/W1C	0x0	<p>CRIS1</p> <p>Status of the capture channel 1 rising lock interrupt.</p> <p>When the capture channel 1 captures rising edge, if the rise lock interrupt (<b>CRIE1</b>) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: The capture channel 1 interrupt is not pending.</p> <p>Reads 1: The capture channel 1 interrupt is pending.</p> <p>Writes 0: no effect.</p> <p>Writes 1: Clear the status of the capture channel 1 interrupt.</p>
1	R/W1C	0x0	<p>CFIS0</p> <p>Status of the capture channel 0 falling lock interrupt</p> <p>When the capture channel 0 captures falling edge, if the fall lock interrupt (<b>CFIE0</b>) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: The capture channel 0 interrupt is not pending.</p> <p>Reads 1: The capture channel 0 interrupt is pending.</p> <p>Writes 0: no effect.</p> <p>Writes 1: Clear the status of the capture channel 0 interrupt.</p>
0	R/W1C	0x0	<p>CRIS0</p> <p>Status of the capture channel 0 rising lock interrupt</p> <p>When the capture channel 0 captures rising edge, if the rise lock interrupt (<b>CRIE0</b>) is enabled, this bit is set 1 by hardware. Writing 1 to clear this bit.</p> <p>Reads 0: The capture channel 0 interrupt is not pending.</p> <p>Reads 1: The capture channel 0 interrupt is pending.</p> <p>Writes 0: no effect.</p> <p>Writes 1: Clear the status of the capture channel 0 interrupt.</p>

9.10.6.5 0x0020 PWM01 Clock Configuration Register (Default Value: 0x0000\_0000)

Offset:0x0020			Register Name: PCCR01
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/

Offset:0x0020			Register Name: PCCR01
Bit	Read/Write	Default/Hex	Description
8:7	R/W	0x0	PWM01_CLK_SRC Select PWM01 Clock Source 00: HOSC 01: APB0 Others: Reserved
6:4	/	/	/
3:0	R/W	0x0	PWM01_CLK_DIV_M PWM01 Clock Divide M 0000: /1 0001: /2 0010: /4 0011: /8 0100: /16 0101: /32 0110: /64 0111: /128 1000: /256 Others: Reserved

9.10.6.6 0x0024 PWM23 Clock Configuration Register (Default Value: 0x0000\_0000)

Offset:0x0024			Register Name: PCCR23
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8:7	R/W	0x0	PWM23_CLK_SRC_SEL Select PWM23 Clock Source 00: HOSC 01: APB0 Others: Reserved
6:4	/	/	/

Offset:0x0024			Register Name: PCCR23
Bit	Read/Write	Default/Hex	Description
3:0	R/W	0x0	PWM23_CLK_DIV_M PWM23 Clock Divide M 0000: /1 0001: /2 0010: /4 0011: /8 0100: /16 0101: /32 0110: /64 0111: /128 1000: /256 Others: Reserved

9.10.6.7 0x0028 PWM45 Clock Configuration Register (Default Value: 0x0000\_0000)

Offset:0x0028			Register Name: PCCR45
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8:7	R/W	0x0	PWM45_CLK_SRC_SEL Select PWM45 Clock Source 00: HOSC 01: APB0 Others: Reserved
6:4	/	/	/



Offset:0x0028			Register Name: PCCR45
Bit	Read/Write	Default/Hex	Description
3:0	R/W	0x0	PWM45_CLK_DIV_M PWM45 Clock Divide M 0000: /1 0001: /2 0010: /4 0011: /8 0100: /16 0101: /32 0110: /64 0111: /128 1000: /256 Others: Reserved

9.10.6.8 0x002C PWM67 Clock Configuration Register (Default Value: 0x0000\_0000)

Offset:0x002C			Register Name: PCCR67
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8:7	R/W	0x0	PWM67_CLK_SRC_SEL Select PWM67 Clock Source 00: HOSC 01: APB0 Others: Reserved
6:4	/	/	/

Offset:0x002C			Register Name: PCCR67
Bit	Read/Write	Default/Hex	Description
3:0	R/W	0x0	PWM67_CLK_DIV_M PWM67 Clock Divide M 0000: /1 0001: /2 0010: /4 0011: /8 0100: /16 0101: /32 0110: /64 0111: /128 1000: /256 Others: Reserved

9.10.6.9 0x0040 PWM Clock Gating Register (Default Value: 0x0000\_0000)

Offset: 0x0040			Register Name: PCGR
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23	R/W	0x0	PWM7_CLK_BYPASS Bypass clock source (after pre-scale) to PWM7 output 0: not bypass 1: bypass
22	R/W	0x0	PWM6_CLK_BYPASS Bypass clock source (after pre-scale) to PWM6 output 0: not bypass 1: bypass
21	R/W	0x0	PWM5_CLK_BYPASS Bypass clock source (after pre-scale) to PWM5 output 0: not bypass 1: bypass
20	R/W	0x0	PWM4_CLK_BYPASS Bypass clock source (after pre-scale) to PWM4 output 0: not bypass 1: bypass

Offset: 0x0040			Register Name: PCGR
Bit	Read/Write	Default/Hex	Description
19	R/W	0x0	PWM3_CLK_BYPASS Bypass clock source (after pre-scale) to PWM3 output 0: not bypass 1: bypass
18	R/W	0x0	PWM2_CLK_BYPASS Bypass clock source (after pre-scale) to PWM2 output 0: not bypass 1: bypass
17	R/W	0x0	PWM1_CLK_BYPASS Bypass clock source (after pre-scale) to PWM1 output 0: not bypass 1: bypass
16	R/W	0x0	PWM0_CLK_BYPASS Bypass clock source (after pre-scale) to PWM0 output 0: not bypass 1: bypass
15:8	/	/	/
7	R/W	0x0	PWM7_CLK_GATING Gating clock for PWM7 0: Mask 1: Pass
6	R/W	0x0	PWM6_CLK_GATING Gating clock for PWM6 0: Mask 1: Pass
5	R/W	0x0	PWM5_CLK_GATING Gating clock for PWM5 0: Mask 1: Pass
4	R/W	0x0	PWM4_CLK_GATING Gating clock for PWM4 0: Mask 1: Pass

Offset: 0x0040			Register Name: PCGR
Bit	Read/Write	Default/Hex	Description
3	R/W	0x0	PWM3_CLK_GATING Gating clock for PWM3 0: Mask 1: Pass
2	R/W	0x0	PWM2_CLK_GATING Gating clock for PWM2 0: Mask 1: Pass
1	R/W	0x0	PWM1_CLK_GATING Gating clock for PWM1 0: Mask 1: Pass
0	R/W	0x0	PWM0_CLK_GATING Gating clock for PWM0 0: Mask 1: Pass

9.10.6.10 0x0060 PWM01 Dead Zone Control Register (Default Value: 0x0000\_0000)

Offset:0x0060			Register Name: PDZCR01
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:8	R/W	0x0	PWM01_DZ_INTV PWM01 Dead Zone Interval Value
7:1	/	/	/
0	R/W	0x0	PWM01_DZ_EN PWM01 Dead Zone Enable 0: Dead Zone disable 1: Dead Zone enable

9.10.6.11 0x0064 PWM23 Dead Zone Control Register (Default Value: 0x0000\_0000)

Offset:0x0064			Register Name: PDZCR23
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:8	R/W	0x0	PWM23_DZ_INTV PWM23 Dead Zone Interval Value
7:1	/	/	/
0	R/W	0x0	PWM23_DZ_EN PWM23 Dead Zone Enable 0: Dead Zone disable 1: Dead Zone enable

9.10.6.12 0x0068 PWM45 Dead Zone Control Register (Default Value: 0x0000\_0000)

Offset:0x0068			Register Name: PDZCR45
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:8	R/W	0x0	PWM45_DZ_INTV PWM45 Dead Zone Interval Value
7:1	/	/	/
0	R/W	0x0	PWM45_DZ_EN PWM45 Dead Zone Enable 0: Dead Zone disable 1: Dead Zone enable

9.10.6.13 0x006C PWM67 Dead Zone Control Register (Default Value: 0x0000\_0000)

Offset:0x006C			Register Name: PDZCR67
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:8	R/W	0x0	PWM67_DZ_INTV PWM67 Dead Zone Interval Value
7:1	/	/	/

Offset:0x006C			Register Name: PDZCR67
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	PWM67_DZ_EN PWM67 Dead Zone Enable 0: Dead Zone disable 1: Dead Zone enable

9.10.6.14 0x0080 PWM Enable Register (Default Value: 0x0000\_0000)

Offset:0x0080			Register Name: PER
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	PWM7_EN When PWM is enabled, the 16-bit up-counter starts working and PWM channel7 is permitted to output PWM waveform. 0: PWM disable 1: PWM enable
6	R/W	0x0	PWM6_EN When PWM is enabled, the 16-bit up-counter starts working and PWM channel6 is permitted to output PWM waveform. 0: PWM disable 1: PWM enable
5	R/W	0x0	PWM5_EN When PWM is enabled, the 16-bit up-counter starts working and PWM channel5 is permitted to output PWM waveform. 0: PWM disable 1: PWM enable
4	R/W	0x0	PWM4_EN When PWM is enabled, the 16-bit up-counter starts working and PWM channel4 is permitted to output PWM waveform. 0: PWM disable 1: PWM enable
3	R/W	0x0	PWM3_EN When PWM is enabled, the 16-bit up-counter starts working and PWM channel3 is permitted to output PWM waveform. 0: PWM disable 1: PWM enable

Offset:0x0080			Register Name: PER
Bit	Read/Write	Default/Hex	Description
2	R/W	0x0	<p>PWM2_EN</p> <p>When PWM is enabled, the 16-bit up-counter starts working and PWM channel2 is permitted to output PWM waveform.</p> <p>0: PWM disable</p> <p>1: PWM enable</p>
1	R/W	0x0	<p>PWM1_EN</p> <p>When PWM is enabled, the 16-bit up-counter starts working and PWM channel1 is permitted to output PWM waveform.</p> <p>0: PWM disable</p> <p>1: PWM enable</p>
0	R/W	0x0	<p>PWM0_EN</p> <p>When PWM is enabled, the 16-bit up-counter starts working and PWM channel0 is permitted to output PWM waveform.</p> <p>0: PWM disable</p> <p>1: PWM enable</p>

9.10.6.15 0x0090 PWM Group0 Register (Default Value: 0x0000\_0000)

Offset: 0x0090			Register Name: PGR0
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17	R/WAC	0x0	<p>PWMGO_START</p> <p>The PWM channels selected in <b>PWMGO_CS</b> start to output PWM waveform at the same time.</p>
16	R/W	0x0	<p>PWMGO_EN</p> <p>PWM Group0 Enable.</p>
15:0	R/W	0x0	<p>PWMGO_CS</p> <p>If bit[i] is set, the PWM i is selected as one channel of PWM Group0.</p>

9.10.6.16 0x0094 PWM Group1 Register (Default Value: 0x0000\_0000)

Offset: 0x0094			Register Name: PGR1
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/

Offset: 0x0094			Register Name: PGR1
Bit	Read/Write	Default/Hex	Description
17	R/WAC	0x0	PWMG1_START The PWM channels selected in <b>PWMG1_CS</b> start to output PWM waveform at the same time.
16	R/W	0x0	PWMG1_EN PWM Group1 Enable.
15:0	R/W	0x0	PWMG1_CS If bit[i] is set, the PWM i is selected as one channel of PWM Group1.

9.10.6.17 0x0098 PWM Group2 Register (Default Value: 0x0000\_0000)

Offset: 0x0098			Register Name: PGR2
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17	R/WAC	0x0	PWMG2_START The PWM channels selected in <b>PWMG2_CS</b> start to output PWM waveform at the same time.
16	R/W	0x0	PWMG2_EN PWM Group2 Enable.
15:0	R/W	0x0	PWMG2_CS If bit[i] is set, the PWM i is selected as one channel of PWM Group2.

9.10.6.18 0x009C PWM Group3 Register (Default Value: 0x0000\_0000)

Offset: 0x009C			Register Name: PGR3
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17	R/WAC	0x0	PWMG3_START The PWM channels selected in <b>PWMG3_CS</b> start to output PWM waveform at the same time.
16	R/W	0x0	PWMG3_EN PWM Group3 Enable.
15:0	R/W	0x0	PWMG3_CS If bit[i] is set, the PWM i is selected as one channel of PWM Group3.



9.10.6.19 0x00C0 Capture Enable Register (Default Value: 0x0000\_0000)

Offset: 0x00C0			Register Name: CER
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	<p>CAP7_EN</p> <p>When enabling the capture function, the 16-bit up-counter starts working, and the capture channel7 is permitted to capture external falling edge or rising edge.</p> <p>0: Capture disable 1: Capture enable</p>
6	R/W	0x0	<p>CAP6_EN</p> <p>When enabling the capture function, the 16-bit up-counter starts working, and the capture channel6 is permitted to capture external falling edge or rising edge.</p> <p>0: Capture disable 1: Capture enable</p>
5	R/W	0x0	<p>CAP5_EN</p> <p>When enabling the capture function, the 16-bit up-counter starts working, and the capture channel5 is permitted to capture external falling edge or rising edge.</p> <p>0: Capture disable 1: Capture enable</p>
4	R/W	0x0	<p>CAP4_EN</p> <p>When enabling the capture function, the 16-bit up-counter starts working, and the capture channel4 is permitted to capture external falling edge or rising edge.</p> <p>0: Capture disable 1: Capture enable</p>
3	R/W	0x0	<p>CAP3_EN</p> <p>When enabling the capture function, the 16-bit up-counter starts working, and the capture channel3 is permitted to capture external falling edge or rising edge.</p> <p>0: Capture disable 1: Capture enable</p>

Offset: 0x00C0			Register Name: CER
Bit	Read/Write	Default/Hex	Description
2	R/W	0x0	<p>CAP2_EN</p> <p>When enabling the capture function, the 16-bit up-counter starts working, and the capture channel2 is permitted to capture external falling edge or rising edge.</p> <p>0: Capture disable 1: Capture enable</p>
1	R/W	0x0	<p>CAP1_EN</p> <p>When enabling the capture function, the 16-bit up-counter starts working, and the capture channel1 is permitted to capture external falling edge or rising edge.</p> <p>0: Capture disable 1: Capture enable</p>
0	R/W	0x0	<p>CAPO_EN</p> <p>When enabling the capture function, the 16-bit up-counter starts working, and the capture channel is permitted to capture external falling edge or rising edge.</p> <p>0: Capture disable 1: Capture enable</p>

9.10.6.20 0x0100 + N\*0x20 PWM Control Register (Default Value: 0x0000\_0000)

Offset:0x0100+0x0+N*0x20 (N=0~7)			Register Name: PCR
Bit	Read/Write	Default/Hex	Description
31:16	R/W	0x0	<p>PWM_PUL_NUM</p> <p>In pulse mode, the PWM outputs pulse for PWM_CYCLE_NUM+1 times and then stops.</p>
15:12	/	/	/
11	R	0x0	<p>PWM_PERIOD_RDY</p> <p>PWM Period Register Ready</p> <p>0: PWM period register is ready to write 1: PWM period register is busy</p>

Offset:0x0100+0x0+N*0x20 (N=0~7)			Register Name: PCR
Bit	Read/Write	Default/Hex	Description
10	R/WAC	0x0	PWM_PUL_START PWM Pulse Output Start 0: No effect 1: Output pulse for PWM_CYCLE_NUM+1. After finishing configuration for the output pulse, set this bit once , then PWM would output waveform. After the waveform is finished, the bit will be cleared automatically.
9	R/W	0x0	PWM_MODE PWM Output Mode Select 0: Cycle mode 1: Pulse mode
8	R/W	0x0	PWM_ACT_STA PWM Active State 0: Low Level 1: High Level
7:0	R/W	0x0	PWM_PRESCAL_K PWM pre-scale K, actual pre-scale is (K+1). K = 0, actual pre-scale: 1 K = 1, actual pre-scale: 2 K = 2, actual pre-scale: 3 K = 3, actual pre-scale: 4 ... K = 255, actual pre-scale: 256

9.10.6.21 0x0104 + N\*0x20 PWM Period Register (Default Value: 0x0000\_0000)

Offset:0x0100+0x04+N*0x20 (N=0~7)			Register Name: PPR
Bit	Read/Write	Default/Hex	Description
31:16	R/W	0x0	PWM_ENTIRE_CYCLE Number of the entire cycles in the PWM clock. 0: 1 cycle 1: 2 cycles ... N: N+1 cycles If the register needs to be modified dynamically, the PCLK should be faster than the PWM CLK.

Offset:0x0100+0x04+N*0x20 (N=0~7)			Register Name: PPR
Bit	Read/Write	Default/Hex	Description
15:0	R/W	0x0	PWM_ACT_CYCLE Number of the active cycles in the PWM clock. 0: 0 cycle 1: 1 cycle ... N: N cycles

9.10.6.22 0x0108 + N\*0x20 PWM Counter Register (Default Value: 0x0000\_0000)

Offset:0x0100+0x08+N*0x20 (N=0~7)			Register Name: PCNTR
Bit	Read/Write	Default/Hex	Description
31:16	R/W	0x0	PWM_COUNTER_START PWM counter value is set for phase control.
15:0	R	0x0	PWM_COUNTER_STATUS On PWM output or capture input, reading this register could get the current value of the PWM 16-bit up-counter.

9.10.6.23 0x010C + N\*0x20 PWM Pulse Counter Register (Default Value: 0x0000\_0000)

Offset: 0x0100+0x0C+N*0x20 (N=0~7)			Register Name: PPCNTR
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R	0x0	PWM_PUL_COUNTER_STATUS On PWM output, reading this register could get the current value of the PWM pulse counter.

9.10.6.24 0x0110 + N\*0x20 PWM Capture Control Register (Default Value: 0x0000\_0000)

Offset:0x0100+0x10+N*0x20 (N=0~7)			Register Name: CCR
Bit	Read/Write	Default/Hex	Description
31:5	/	/	/

Offset:0x0100+0x10+N*0x20 (N=0~7)			Register Name: CCR
Bit	Read/Write	Default/Hex	Description
4	R/W1C	0x0	CRLF When the capture channel captures a rising edge, the current value of the 16-bit up-counter is latched to CRLR, and then this bit is set 1 by hardware. Write 1 to clear this bit.
3	R/W1C	0x0	CFLF When the capture channel captures a falling edge, the current value of the 16-bit up-counter is latched to CFLR, and then this bit is set 1 by hardware. Write 1 to clear this bit.
2	R/W	0x0	CRTE Rising edge capture trigger enable
1	R/W	0x0	CFTE Falling edge capture trigger enable
0	R/W	0x0	CAPINV Inverse the signal input from capture channel before 16-bit counter of capture channel. 0: not inverse 1: inverse

9.10.6.25 0x0114 + N\*0x20 PWM Capture Rise Lock Register (Default Value: 0x0000\_0000)

Offset:0x0100+0x14+N*0x20 (N=0~7)			Register Name: CRLR
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R	0x0	CRLR When the capture channel captures a rising edge, the current value of the 16-bit up-counter is latched to the register.

9.10.6.26 0x0118 + N\*0x20 PWM Capture Fall Lock Register (Default Value: 0x0000\_0000)

Offset:0x0100+0x18+N*0x20 (N=0~7)			Register Name: CFLR
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/

Offset:0x0100+0x18+N*0x20 (N=0~7)			Register Name: CFLR
Bit	Read/Write	Default/Hex	Description
15:0	R	0x0	CFLR When the capture channel captures a falling edge, the current value of the 16-bit up-counter is latched to the register.



## 9.11 LEDC

### 9.11.1 Overview

The LEDC is used to control the external LED lamp.

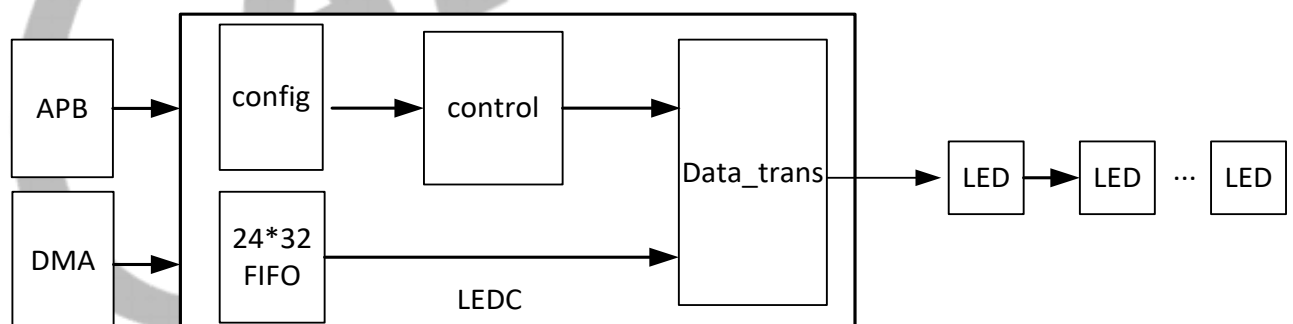
The LEDC has the following features:

- Configurable LED output high-/low-level width
- Configurable LED reset time
- LEDC data supports DMA configuration mode and CPU configuration mode
- Maximum 1024 LEDs serial connect
- LED data transfer rate up to 800 kbit/s
- Configurable RGB display mode
- The default level of non-data output is configurable

### 9.11.2 Block Diagram

The following figure shows a block diagram of the LEDC.

Figure 9-93 LEDC Block Diagram



LEDC contains the following sub-blocks:

Table 9-32 LEDC Sub-blocks

Sub-block	Description
config	register configuration
control	LEDC timing control and status control
FIFO	24-bit width x 32 depth
Data_trans	Convert input data to the 0 and 1 characters of LED

### 9.11.3 Functional Description

#### 9.11.3.1 External Signals

The following table describes the external signals of the LEDC.

**Table 9-33 LEDC External Signals**

Signal	Description	Type
LEDC-DO	Intelligent Control LED Signal Output	O

#### 9.11.3.2 Clock Sources

The following table describes the clock sources of the LEDC.

**Table 9-34 LEDC Clock Sources**

Clock Sources	Description
HOSC	24 MHz
PLL_PERI(1X)	Peripheral Clock. The default value is 600 MHz

#### 9.11.3.3 LEDC Timing

**Figure 9-94 LEDC Package Output Timing Diagram**

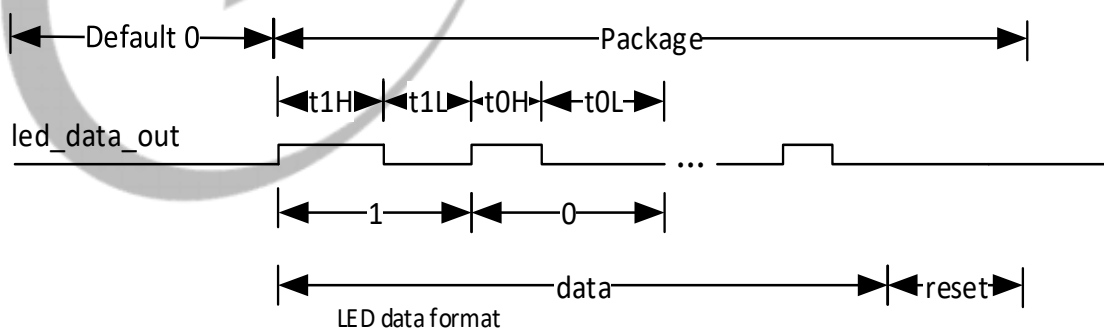
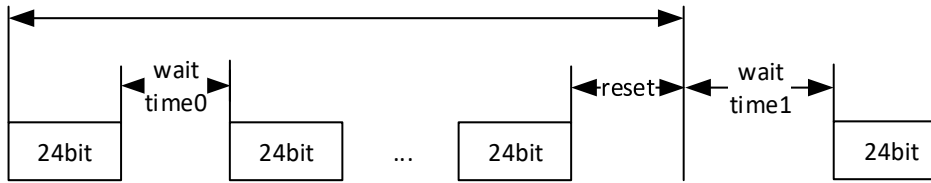




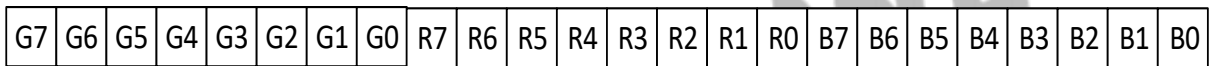
Figure 9-95 LEDC 1-frame Output Timing Diagram



### 9.11.3.4 LEDC Input Data Structure

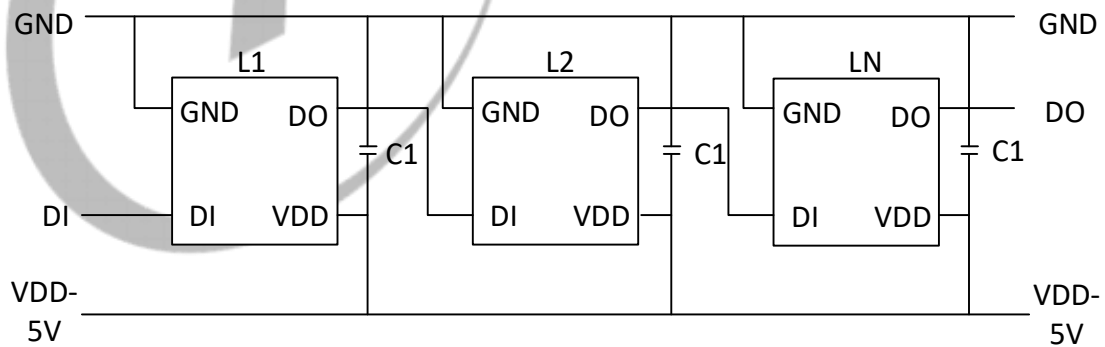
The RGB mode of LEDC data is configurable. By default, the data is sent in GRB order, and the higher bit is transmitted first.

Figure 9-96 LEDC Input Data Structure



### 9.11.3.5 LEDC Typical Circuit

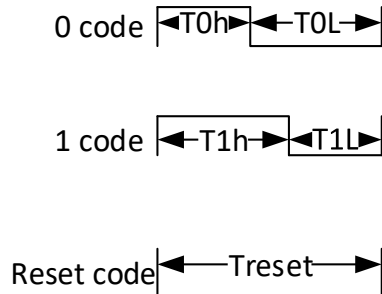
Figure 9-97 LEDC Typical Circuit



C1 is the bypass capacitor of LED light, and its value is usually 100 nF.

### 9.11.3.6 LEDC Data Input Code

Figure 9-98 LEDC Data Input Code



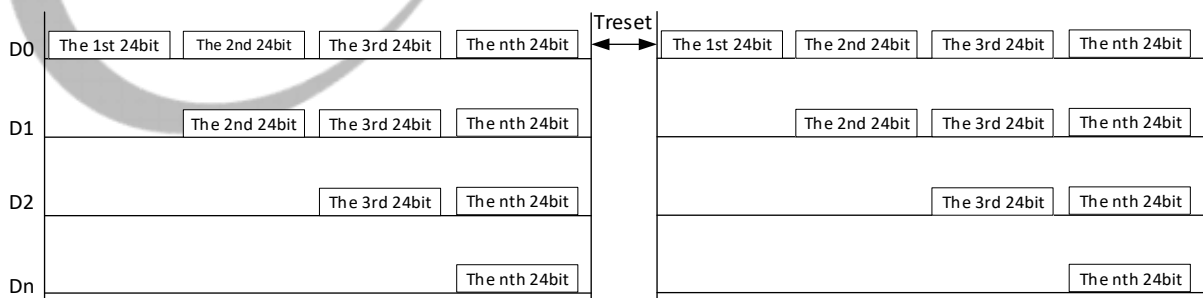
### 9.11.3.7 LEDC Data Transfer Time

The time parameter of the typical LED specification shows as follows.

<b>T0H</b>	0 code, high-level time	220 ns to 380 ns
<b>T0L</b>	0 code, low-level time	580 ns to 1.6 us
<b>T1H</b>	1 code, high-level time	580 ns to 1.6 us
<b>T1L</b>	1 code, low-level time	220 ns to 420 ns
<b>RESET</b>	Frame unit, low-level time	> 280 us

### 9.11.3.8 LEDC Data Transfer Mode

Figure 9-99 LEDC Data Transfer Mode



### 9.11.3.9 LEDC Parameter

1. PAD rate > 800 kbit/s
2. LED number supported:

T<sub>0-code</sub>: 800 ns to 1980 ns, T<sub>1-code</sub>: 800 ns to 2020 ns

When the LED refresh rate is 30 frame/s, LED number supported is  $(1 \text{ s}/30-280 \text{ us})/((800 \text{ ns to } 2020 \text{ ns}) * 24)$   
 =1024 →to 681.

When the LED refresh rate is 60 frame/s, LED number supported is  $(1 \text{ s}/60-280 \text{ us})/((800 \text{ ns to } 2020 \text{ ns}) * 24)$   
 =853 →to 338.

### 9.11.3.10 LEDC Data Transfer

The LEDC supports DMA data transfer mode or CPU data transfer mode. The DMA data transfer mode is set by LEDC\_DMA\_EN.

#### Data transfer in DMA mode

When the valid space of internal FIFO is greater than the setting FIFO free space threshold, the LEDC sends DMA\_REQ to require DMA to transfer data from DRAM to LEDC. The maximum data transfer size in DMA mode is 16 words. (The internal FIFO level is 32.)

#### Data transfer in CPU mode

When the valid space of internal FIFO is greater than the setting FIFO free space threshold, the LEDC sends LEDC\_CPUREQ\_INT to require CPU to transfer data to LEDC. The transfer data size in CPU mode is controlled by software. The internal FIFO destination address is 0x06700014. The data width is 32-bit. (The lower 24-bit is valid.)

### 9.11.3.11 LEDC Interrupt

Module Name	Description
FIFO_OVERFLOW_INT	<p>FIFO overflow interrupt.</p> <p>The data written by external is more than the maximum storage space of LED FIFO, the LEDC will be in data loss state. At this time, software needs to deal with the abnormal situation. The processing mode is as follows.</p> <p>The software can query LED_FIFO_DATA_REG to determine which data has been stored in the internal FIFO of LEDC. The LEDC performs soft_reset operation to refresh all data.</p>

Module Name	Description
WAITDATA_TIMEOUT_INT	<p>Wait for data timeout interrupt</p> <p>When internal FIFO of LEDC cannot get data because of some abnormal situation, the timeout interrupt is set after led_wait_data_time, now the LEDC is in WAIT_DATA state, and the LEDC outputs a level state configured by LED_POLARITY; in the course of wait_data, if the new data arrives, the LEDC will continue to send data, at this time the software needs to notice whether the waiting time of the LEDC exceeds the operation time of reset. If the waiting time of the LEDC exceeds the operation time of reset (this is equivalent to reset operation sent by LEDC), the LED may enter in refresh state, the data has not been sent.</p>
FIFO_CPUREQ_INT	<p>FIFO request CPU data interrupt</p> <p>When FIFO data is less than a threshold, the interrupt will be reported to the CPU.</p>
LEDC_TRANS_FINISH_INT	<p>Data transfer complete interrupt</p> <p>The value indicates that the data configured as total_data_length has been transferred completely.</p>

LEDC interrupt usage scenario:

**CPU mode**

The software can enable GLOBAL\_INT\_EN, FIFO\_CPUREQ\_INT\_EN, WAITDATA\_TIMEOUT\_INT\_EN, FIFO\_OVERFLOW\_INT\_EN, LEDC\_TRANS\_FINISH\_INT\_EN, and cooperate with LEDC\_FIFO\_TRIG\_LEVEL to use. When FIFO\_CPUREQ\_INT is set to 1, the software can configure data of LEDC\_FIFO\_TRIG\_LEVEL to LEDC.

**DMA mode**

The software can enable GLOBAL\_INT\_EN, WAITDATA\_TIMEOUT\_INT\_EN, FIFO\_OVERFLOW\_INT\_EN, LEDC\_TRANS\_FINISH\_INT\_EN, and cooperate with LEDC\_FIFO\_TRIG\_LEVEL to use. When DMA receives LEDC DMA\_REQ, DMA can transfer data of LEDC\_FIFO\_TRIG\_LEVEL to LEDC.

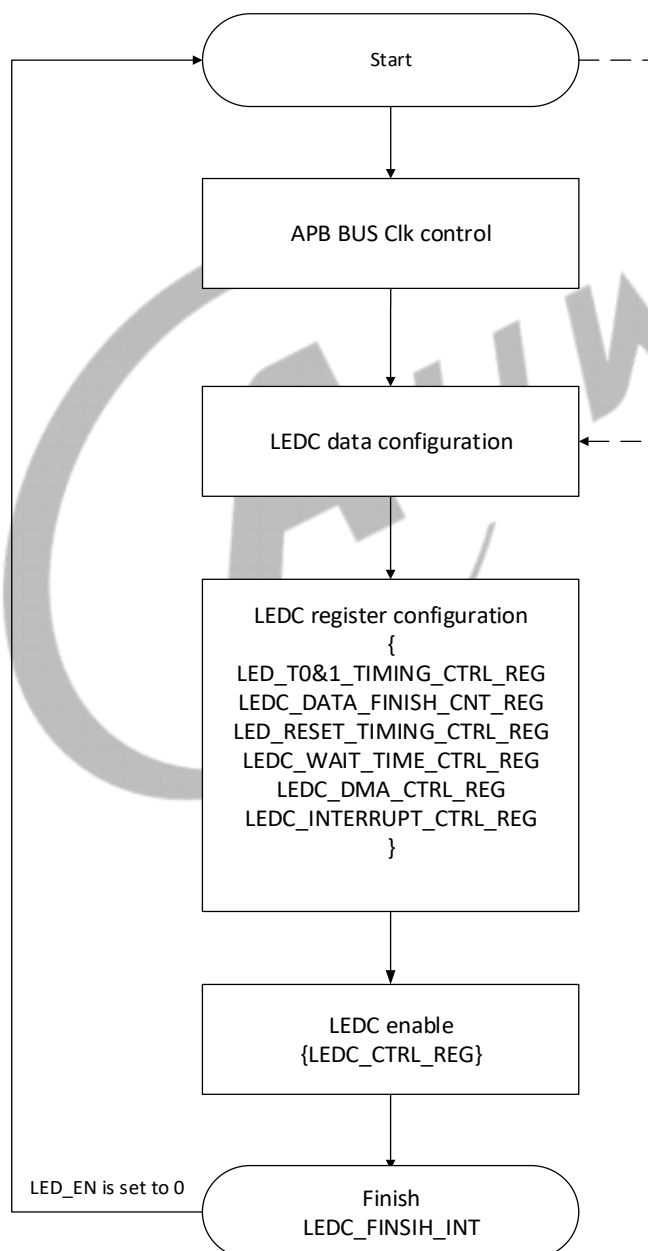
**9.11.4 Programming Guidelines**

**9.11.4.1 LEDC Normal Configuration Process**

- Step 1** Configure LEDC\_CLK and bus pclk.
- Step 2** Configure the written LEDC data.

- Step 3** Configure [LED\\_T0&1\\_TIMING\\_CTRL\\_REG](#), [LEDC\\_DATA\\_FINISH\\_CNT\\_REG](#), [LED\\_RESET\\_TIMING\\_CTRL\\_REG](#), [LEDC\\_WAIT\\_TIME0\\_CTRL\\_REG](#), [LEDC\\_DMA\\_CTRL\\_REG](#), [LEDC\\_INTERRUPT\\_CTRL\\_REG](#). Configure 0-code, 1-code, reset time, LEDC waiting time, and the number of external connected LEDC and the threshold of DMA transfer data.
- Step 4** Configure [LEDC\\_CTRL\\_REG](#) to enable LEDC\_EN, the LEDC will start to output data.
- Step 5** When the LEDC interrupt is pulled up, it indicates the configured data has transferred complete, at this time LED\_EN will be set to 0, and the read/write point of LEDC FIFO is cleared to 0.
- Step 6** Repeat step1, 2, 3, 4 to re-execute a new round of configuration, enable LEDC\_EN, the LEDC will start new data transfer.

Figure 9-100 LEDC Normal Configuration Process

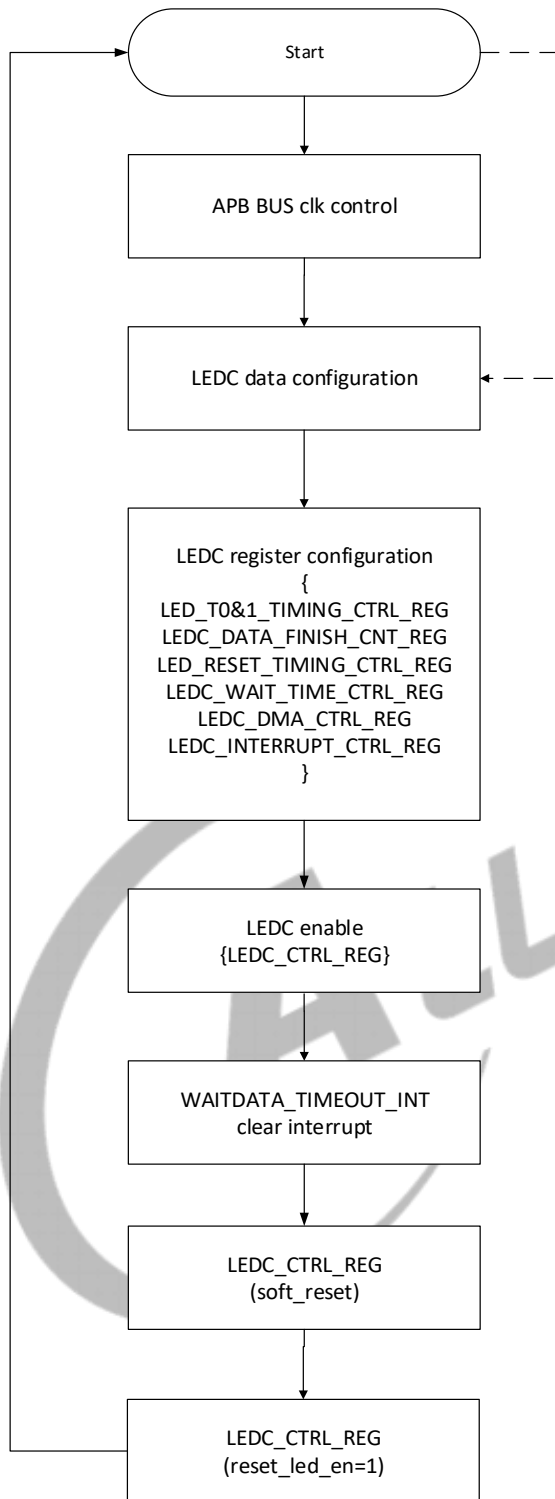


#### 9.11.4.2 LEDC Abnormal Scene Processing Flow

##### WAITDATA\_TIMEOUT Abnormal Status

- Step 1** When WAITDATA\_TIMEOUT\_INT appears, it indicates the internal FIFO data request of LEDC cannot obtain a response, at this time if the default output level is low, then the external LED may think there was a reset operation and cause LED data to be flushed incorrectly.
- Step 2** The LEDC needs to be performed soft\_reset operation (LEDC\_SOFT\_RESET=1); after soft\_reset, the LEDC\_EN will be pulled-down automatically, all internal status register and control state machine will return to the idle state, the LEDC FIFO read & write point is cleared to 0, the LEDC interrupt is cleared.
- Step 3** Setting reset\_led\_en to 1 indicates LEDC can actively send a reset operation to ensure the external LED lamp in the right state.
- Step 4** The software reads the status of reset\_led\_en, when the status value is 1, it indicates LEDC does not perform the transmission of LED reset operation; when the status value is 0, the LEDC completes the transmission of LED reset operation.
- Step 5** When LEDC reset operation finishes, the LEDC data and register configuration need to be re-operated to start re-transmission data operation.

Figure 9-101 LEDC Timeout Abnormal Processing Flow



**FIFO Overflow Abnormal Status**

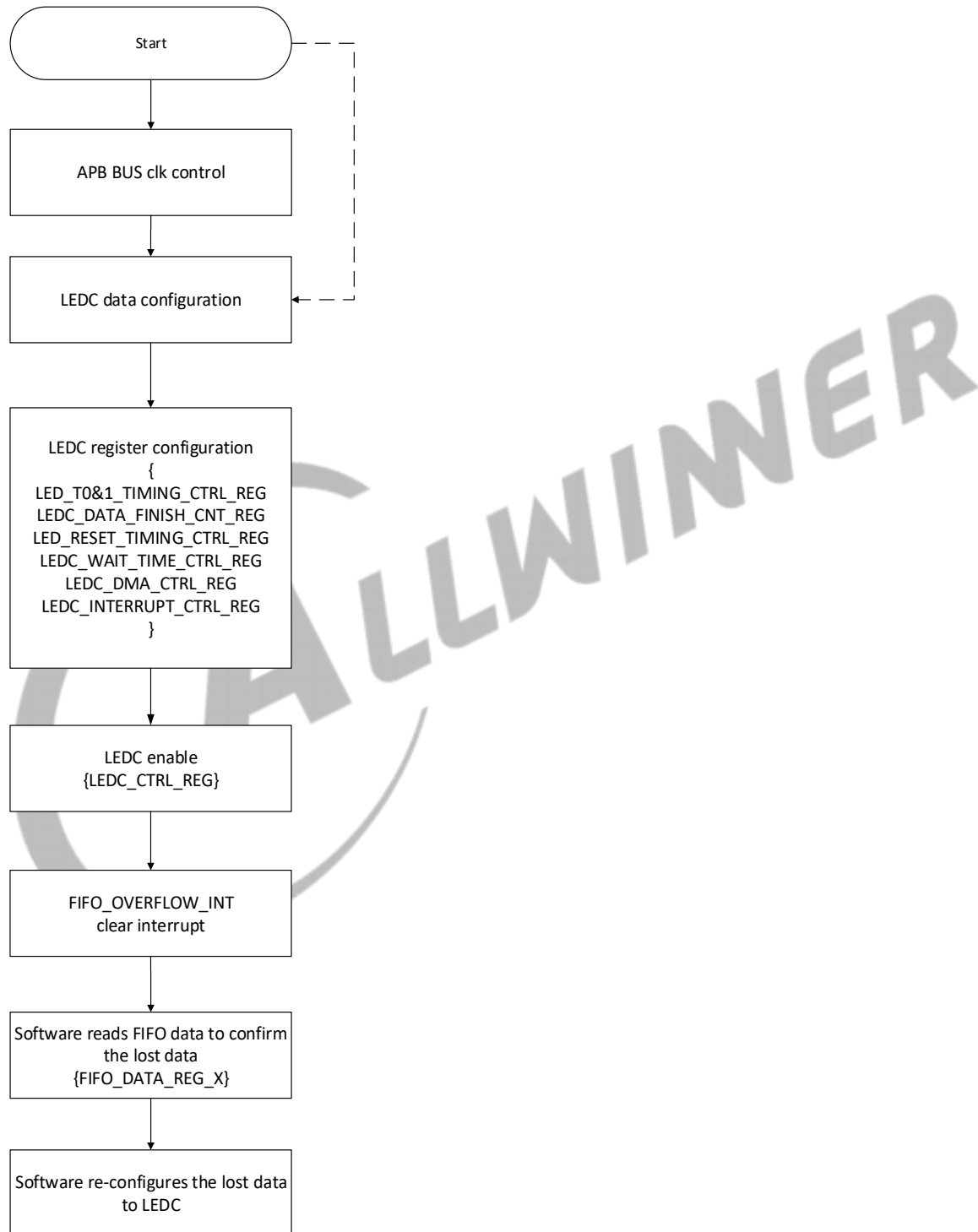
**Step 1** When FIFO\_OVERFLOW\_INT appears, it indicates the data configured by software exceeds the LEDC FIFO space, at this time the redundant data will be lost.

**Step 2** The software needs to read data in [LEDC\\_FIFO\\_DATA\\_X](#) to confirm the lost data.

**Step 3** The software re-configures the lost data to the LEDC.

**Step 4** If the software uses the soft\_reset operation, the operation is the same with the timeout abnormal processing flow.

**Figure 9-102 FIFO Overflow Abnormal Processing Flow**





### 9.11.5 Register List

Module Name	Base Address
LEDC	0x02008000

Register Name	Offset	Description
LEDC_CTRL_REG	0x0000	LEDC Control Register
LED_TO1_TIMING_CTRL_REG	0x0004	LEDC T0 & T1 Timing Control Register
LEDC_DATA_FINISH_CNT_REG	0x0008	LEDC Data Finish Counter Register
LED_RESET_TIMING_CTRL_REG	0x000C	LEDC Reset Timing Control Register
LEDC_WAIT_TIME0_CTRL_REG	0x0010	LEDC Wait Time0 Control Register
LEDC_DATA_REG	0x0014	LEDC Data Register
LEDC_DMA_CTRL_REG	0x0018	LEDC DMA Control Register
LEDC_INT_CTRL_REG	0x001C	LEDC Interrupt Control Register
LEDC_INT_STS_REG	0x0020	LEDC Interrupt Status Register
LEDC_WAIT_TIME1_CTRL_REG	0x0028	LEDC Wait Time1 Control Register
LEDC_FIFO_DATA_REG	0x0030+0x04*N	LEDC FIFO Data Register

### 9.11.6 Register Description

#### 9.11.6.1 0x0000 LEDC Control Register (Default Value: 0x0000\_003C)

Offset: 0x0000			Register Name: LEDC_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	0x0	<p>TOTAL_DATA_LENGTH Total length of transfer data (range: 0 to 8K, unit: 32-bit, only low 24-bit is valid)</p> <p>The field is recommended to be set to an integer multiple of (LED_NUM+1). If TOTAL_DATA_LENGTH is greater than (LED_NUM+1), but non-integer multiple, the last frame of data will transfer data less than (LED_NUM+1).</p>
15:11	/	/	/
10	R/W	0x0	<p>RESET_LED_EN Write operation:</p>

Offset: 0x0000			Register Name: LEDC_CTRL_REG																									
Bit	Read/Write	Default/Hex	Description																									
			<p>The software writes 1 to the bit, the CPU triggers LEDC to transfer a reset to LED.</p> <p><b>Only when LEDC is in IDLE status, the reset can be performed. After the reset finished, the control state machine returns to the IDLE status. To return LEDC to the IDLE status, it also needs to be used with SOFT_RESET.</b></p> <p><b>When the software sets the bit, the software can read the bit to check if the reset is complete.</b></p> <p>Read operation:            0: LEDC completes the transmission of the LED reset operation            1: LEDC does not complete the transmission of the LED reset operation</p>																									
9	/	/	/																									
8:6	R/W	0x0	<p>LED_RGB_MODE</p> <p>000 GRB (bypass)            001 GBR            010 RGB            011 RBG            100 BGR            101 BRG</p> <p>By default, the software configures data to LEDC according to GRB (MSB) mode, the LEDC internal combines data to output to the external LED.</p> <p>Other modes configure as follows.</p> <table border="1"> <thead> <tr> <th>Software Input Mode</th> <th>Configuration</th> <th>LEDC Output Mode</th> </tr> </thead> <tbody> <tr> <td rowspan="6">GRB</td> <td>000</td> <td>GRB</td> </tr> <tr> <td>001</td> <td>GBR</td> </tr> <tr> <td>010</td> <td>RGB</td> </tr> <tr> <td>011</td> <td>RBG</td> </tr> <tr> <td>100</td> <td>BGR</td> </tr> <tr> <td>101</td> <td>BRG</td> </tr> <tr> <td rowspan="4">GBR</td> <td>000</td> <td>GBR</td> </tr> <tr> <td>001</td> <td>GRB</td> </tr> <tr> <td>010</td> <td>BGR</td> </tr> <tr> <td>011</td> <td>BRG</td> </tr> </tbody> </table>	Software Input Mode	Configuration	LEDC Output Mode	GRB	000	GRB	001	GBR	010	RGB	011	RBG	100	BGR	101	BRG	GBR	000	GBR	001	GRB	010	BGR	011	BRG
Software Input Mode	Configuration	LEDC Output Mode																										
GRB	000	GRB																										
	001	GBR																										
	010	RGB																										
	011	RBG																										
	100	BGR																										
	101	BRG																										
GBR	000	GBR																										
	001	GRB																										
	010	BGR																										
	011	BRG																										

Offset: 0x0000			Register Name: LEDC_CTRL_REG		
Bit	Read/Write	Default/Hex	Description		
				100	RGB
				101	RBG
			RGB	000	RGB
				001	RBG
				010	GRB
				011	GBR
				100	BRG
				101	BGR
			RBG	000	RBG
				001	RGB
				010	BRG
				011	BGR
				100	GRB
				101	GBR
			BGR	000	BGR
				001	BRG
				010	GBR
				011	GRB
				100	RBG
				101	RGB
			BRG	000	BRG
				001	BGR
				010	RBG
				011	RGB
100	GBR				
101	GRB				
5	R/W	0x1	LED_MSB_TOP Adjust sequence of the combined GRB data 0: LSB 1: MSB		
4	R/W	0x1	LED_MSB_G MSB control for Green data 0: LSB		

Offset: 0x0000			Register Name: LEDC_CTRL_REG
Bit	Read/Write	Default/Hex	Description
			1: MSB
3	R/W	0x1	LED_MSB_R MSB control for Red data 0: LSB 1: MSB
2	R/W	0x1	LED_MSB_B MSB control for Blue data 0: LSB 1: MSB
1	R/W1C	0x0	LEDC_SOFT_RESET LEDC soft reset Write 1 to clear it automatically. The ranges of LEDC soft reset include the following points: all internal status registers, the control state machine returns to idle status, the LEDC FIFO read & write point is cleared to 0, the LEDC interrupt is cleared; and the affected registers are followed. 1.LEDC_CTRL_REG (LEDC_EN is cleared to 0); 2. PLL_TO&1_TIMING_CTRL_REG remains unchanged; 3. LEDC_DATA_FINISH_CNT_REG (LEDC_DATA_FINISH_CNT is cleared to 0); 4.LED_RESET_TIMING_CTRL_REG remains unchanged; 5. LEDC_WAIT_TIME_CTRL_REG remains unchanged; 6. LEDC_DMA_CTRL_REG remains unchanged; 7. LEDC_INTERRUPT_CTRL_REG remains unchanged; 8.LEDC_INT_STS_REG is cleared to 0; 9. LEDC_CLK_GATING_REG remains unchanged; 10.LEDC_FIFO_DATA_REG remains unchanged;
0	R/W	0x0	LEDC_EN LEDC Enable 0: Disable 1: Enable That the bit is enabled indicates LEDC can be started when LEDC data finished transmission or LEDC_EN is cleared to 0 by hardware in LEDC_SOFT_RESET situation.

9.11.6.2 0x0004 LEDC T0 & T1 Timing Control Register (Default Value: 0x0286\_01D3)

Offset: 0x0004			Register Name: LED_T01_TIMING_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:27	/	/	/
26:21	R/W	0x14	T1H_TIME LED T1H time Unit: cycle (24 MHz), T1H_TIME =42 ns*(N+1) The default value is 882 ns, the range is 80 ns–2560 ns. N: 1–3F. When is 0, T1H_TIME = 3F
20:16	R/W	0x6	T1L_TIME LED T1L time Unit: cycle (24 MHz), T1L_TIME =42 ns*(N+1) The default value is 294 ns, the range is 80 ns–1280 ns. N: 1–1F. When is 0, T1L_TIME = 1F
15:11	/	/	/
10:6	R/W	0x7	T0H_TIME LED T0h time Unit: cycle (24 MHz), T0H_TIME =42 ns*(N+1) The default value is 336 ns, the range is 80 ns–1280 ns. N: 1–1F. When is 0, T0H_TIME = 1F
5:0	R/W	0x13	T0L_TIME LED T0l time Unit: cycle (24 MHz), T0L_TIME =42 ns*(N+1) The default value is 840 ns, the range is 80 ns–2560 ns. N: 1–3F. When is 0, T0L_TIME = 3F

9.11.6.3 0x0008 LEDC Data Finish Counter Register (Default Value: 0x1D4C\_0000)

Offset: 0x0008			Register Name: LEDC_DATA_FINISH_CNT_REG
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/

Offset: 0x0008			Register Name: LEDC_DATA_FINISH_CNT_REG
Bit	Read/Write	Default/Hex	Description
29:16	R/W	0x1D4C	<p>LED_WAIT_DATA_TIME</p> <p>The value is the time that internal FIFO in LEDC is waiting for data. When the time is exceeded, the LEDC will send the wait_data_timeout_int interrupt. (This is an abnormal situation, software needs to reset LEDC.)</p> <p>The value is about 300 us by default.</p> <p>The adjust range is 80 ns–655 us.</p> <p>led_wait_data_time=42ns*(N+1).</p> <p>N: 1–1FFF. When the field is 0, LEDC_WAIT_DATA_TIME=1FFF</p>
15:13	/	/	/
12:0	R	0x0	<p>LED_DATA_FINISH_CNT</p> <p>The value is the total LED data that have been sent. (Range: 0–8k)</p>

9.11.6.4 0x000C LEDC Reset Timing Control Register (Default Value: 0x1D4C\_0000)

Offset: 0x000C			Register Name: LED_RESET_TIMING_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	0x1D4C	<p>TR_TIME</p> <p>Reset time control of LED lamp</p> <p>Unit: cycle (24 MHz), tr_time=42 ns*(N+1)</p> <p>The default value is 300 us.</p> <p>The adjust range is 80 ns–327 us.</p> <p>N: 1–1FFF</p>
15:10	/	/	/
9:0	R/W	0x0	<p>LED_NUM</p> <p>The value is the number of external LED lamp. Maximum up to 1024.</p> <p>The default value 0 indicates that 1 LED lamp is external connected. The range is from 0 to 1023.</p>

9.11.6.5 0x0010 LEDC Wait Time 0 Control Register (Default Value: 0x0000\_00FF)

Offset: 0x0010			Register Name: LEDC_WAIT_TIME0_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8	R/W	0x0	<p>WAIT_TIMO_EN WAIT_TIME0 enable</p> <p>When it is 1, the controller automatically inserts waiting time between LED package data.</p> <p>0: Disable 1: Enable</p>
7:0	R/W	0xFF	<p>TOTAL_WAIT_TIME0 Waiting time between 2 LED data. The LEDC output is low level. The adjust range is 80 ns–10 us.</p> <p>wait_time0=42 ns*(N+1) Unit: cycle(24 MHz) N: 1–FF</p>

9.11.6.6 0x0014 LEDC Data Register (Default Value: 0x0000\_0000)

Offset: 0x0014			Register Name: LEDC_DATA_REG
Bit	Read/Write	Default/Hex	Description
31:0	W	0x0	<p>LEDC DATA LED display data (the lower 24-bit is valid)</p>

9.11.6.7 0x0018 LEDC DMA Control Register (Default Value: 0x0000\_002F)

Offset: 0x0018			Register Name: LEDC_DMA_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:6	/	/	/
5	R/W	0x1	<p>LEDC_DMA_EN LEDC DMA request enable</p> <p>0: Disable request of DMA transfer data 1: Enable request of DMA transfer data</p>

Offset: 0x0018			Register Name: LEDC_DMA_CTRL_REG
Bit	Read/Write	Default/Hex	Description
4:0	R/W	0x0F	<p>LEDC_FIFO_TRIG_LEVEL</p> <p>The remaining space of internal FIFO in LEDC</p> <p>The internal FIFO in LEDC is 24*32.</p> <p>When the remaining space of internal FIFO in LEDC is more than or equal to LEDFIFO_TRIG_LEVEL, the DMA or the CPU request will generate. The default value is 15.</p> <p>The adjusted value is from 1 to 31. The recommended configuration is 7 or 15. When the configuration value is 0, LEDFIFO_TRIG_LEVEL=F.</p>

**9.11.6.8 0x001C LEDC Interrupt Control Register (Default Value: 0x0000\_0000)**

Offset: 0x001C			Register Name: LEDC_INTERRUPT_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:6	/	/	/
5	R/W	0x0	<p>GLOBAL_INT_EN</p> <p>Global interrupt enable</p> <p>0: Disable</p> <p>1: Enable</p>
4	R/W	0x0	<p>FIFO_OVERFLOW_INT_EN</p> <p>FIFO overflow interrupt enable</p> <p>When the data written by the software is more than the internal FIFO level of LEDC, the LEDC is in the data loss state.</p> <p>0: Disable</p> <p>1: Enable</p>
3	R/W	0x0	<p>WAITDATA_TIMEOUT_INT_EN</p> <p>The internal FIFO in LEDC cannot get data because of some abnormal situation, after the time of led_wait_data_time, the interrupt will be enabled.</p> <p>0: Disable</p> <p>1: Enable</p>
2	/	/	/
1	R/W	0x0	<p>FIFO_CPUREQ_INT_EN</p> <p>FIFO request CPU data interrupt enable</p> <p>0: Disable</p> <p>1: Enable</p>



Offset: 0x001C			Register Name: LEDC_INTERRUPT_CTRL_REG
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	LED_TRANS_FINISH_INT_EN Data transmission complete interrupt enable 0: Disable 1: Enable

9.11.6.9 0x0020 LEDC Interrupt Status Register (Default Value: 0x0002\_0000)

Offset: 0x0020			Register Name: LEDC_INT_STS_REG
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17	R	0x1	FIFO_EMPTY FIFO empty status flag
16	R	0x0	FIFO_FULL FIFO full status flag
15:10	R	0x0	FIFO_WLW FIFO internal valid data depth It indicates the space FIFO has been occupied.
9:5	/	/	/
4	R/W1C	0x0	FIFO_OVERFLOW_INT FIFO overflow interrupt The data written by external is more than the maximum storage space of LED FIFO, the LEDC will be in the data loss state. At this time, the software needs to deal with the abnormal situation. The processing mode is as follows. (1) The software can query LED_FIFO_DATA_REG to determine which data has been stored in the internal FIFO of LEDC. (2) The LEDC performs soft_reset operation to refresh all data. 0: FIFO not overflow 1: FIFO overflow

Offset: 0x0020			Register Name: LEDC_INT_STS_REG
Bit	Read/Write	Default/Hex	Description
3	R/W1C	0x0	<p>WAITDATA_TIMEOUT_INT</p> <p>When internal FIFO of LEDC cannot get data because of some abnormal situation, after led_wait_data_time, the timeout interrupt is set, the LEDC is in WAIT_DATA state, the LEDC outputs a level state configured by LED_POLARITY; in the course of wait_data, if the new data arrives, the LEDC will continue to send data, at this time software needs to notice whether the waiting time of LEDC exceeds the operation time of reset. If the waiting time of LEDC exceeds the operation time of reset (this is equivalent to reset operation sent by LEDC), the LED may enter in refresh state, the data has not been sent.</p> <p>0: LEDC not timeout 1: LEDC timeout</p>
2	/	/	/
1	R/W1C	0x0	<p>FIFO_CPUREQ_INT</p> <p>FIFO request CPU data interrupt</p> <p>When FIFO data is less than the threshold, the interrupt will be reported to the CPU.</p> <p>0: FIFO does not request that CPU transfers data 1: FIFO requests that CPU transfers data</p>
0	R/W1C	0x0	<p>LED_TRANS_FINISH_INT</p> <p>Data transfer complete interrupt</p> <p>The value indicates that the data configured as total_data_length is transferred completely.</p> <p>0: Data is not transferred completely 1: Data is transferred completely</p>

9.11.6.10 0x0028 LEDC Wait Time 1 Control Register (Default Value: 0x01FF\_FFFF)

Offset: 0x0028			Register Name: LEDC_WAIT_TIME1_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	<p>WAIT_TIM1_EN</p> <p>0: Disable 1: Enable</p> <p>WAIT_TIME1 enable</p> <p>When the bit is 1, the controller automatically inserts the waiting time between the LED frame data.</p>

Offset: 0x0028			Register Name: LEDC_WAIT_TIME1_CTRL_REG
Bit	Read/Write	Default/Hex	Description
30:0	R/W	0x01FFFFFF	<p>TOTAL_WAIT_TIME1</p> <p>Waiting time between 2 frame data.</p> <p>The LEDC output is low level.</p> <p>The adjust range is 80 ns– 85 s. wait_time1=42 ns*(N+1)</p> <p>Unit: cycle (24 MHz)</p> <p>N: 0x80–0x7FFFFFFF</p> <p>If the value is 0, TOTAL_WAIT_TIME1=0x7FFFFFFF</p>

9.11.6.11 0x0030+N\*0x04 LEDC FIFO Data Register X (Default Value: 0x0000\_0000)

Offset: 0x0030+N*0x04 (N=0–31)			Register Name: LEDC_FIFO_DATA_X
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	<p>LEDC_FIFO_DATA_X</p> <p>Internal FIFO data of LEDC</p> <p>The lower 24-bit is valid.</p>

## 9.12 EMAC

### 9.12.1 Overview

The Ethernet Medium Access Controller (EMAC) enables a host to transmit and receive data over Ethernet in compliance with the IEEE 802.3-2002 standard. It supports 10/100/1000 Mbit/s external PHY with RMII/RGMII interface in full-duplex and half-duplex modes. The internal DMA is designed for packet-oriented data transfers based on a linked list of descriptors.

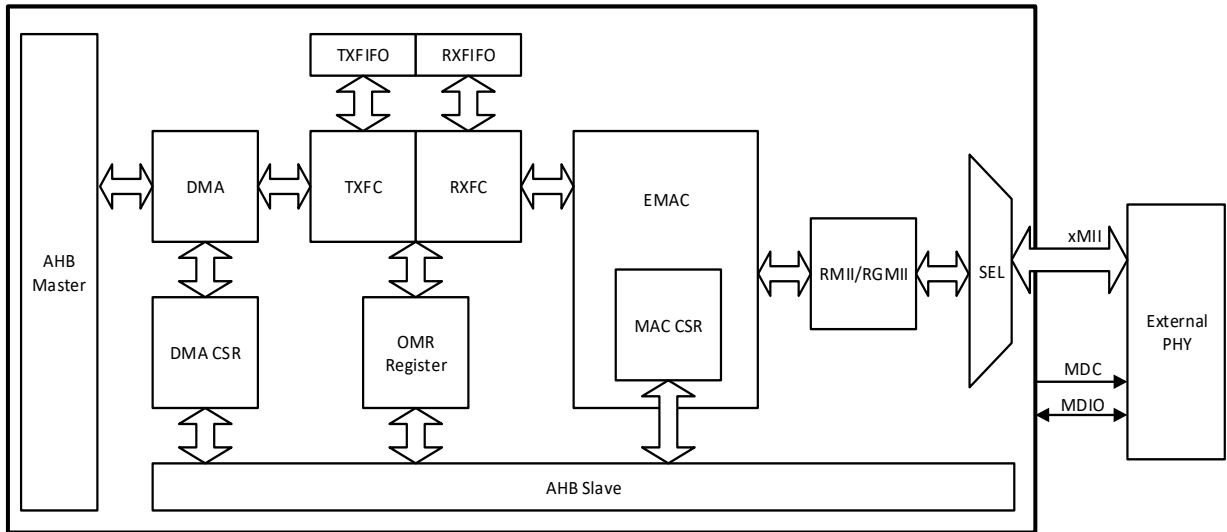
The EMAC has the following features:

- One 10/100/1000 Mbit/s Ethernet port with reduced gigabit media independent interface (RGMII) and reduced media independent interface (RMII) interfaces, for connecting the external EPHY
- Compliant with IEEE 802.3-2002 standard
- Provides the management data input/output (MDIO) interface for PHY device configuration and management with configurable clock frequencies
- Supports both full-duplex and half-duplex operations
- Programmable frame length to support Standard or Jumbo Ethernet frames with sizes up to 16 KB
- Supports a variety of flexible address filtering modes
- Separate 32-bit status returned for transmission and reception packets
- Optimization for packet-oriented DMA transfers with frame delimiters
  - Supports linked-list descriptor list structure
  - Descriptor architecture, allowing large blocks of data transfer with minimum CPU intervention; each descriptor can transfer up to 4 KB of data
  - Comprehensive status reporting for normal operation and transfers with errors
- 4 KB TXFIFO for transmission packets and 16 KB RXFIFO for reception packets
- Programmable interrupt options for different operational conditions

### 9.12.2 Block Diagram

The following figure shows the block diagram of EMAC.

Figure 9-103 EMAC Block Diagram



### 9.12.3 Functional Description

#### 9.12.3.1 External Signals

The following table describes the pin mapping of EMAC.

Table 9-35 EMAC Pin Mapping

Pin Name	RGMII	RMII
RGMII-RXD3	RXD3	/
RGMII-RXD2	RXD2	/
RGMII-RXD1/RMII-RXD1	RXD1	RXD1
RGMII-RXD0/RMII-RXD0	RXD0	RXD0
RGMII-RXCK	RXCK	/
RGMII-RXCTRL/RMII-CRS-DV	RXCTL	CRS-DV
RGMII-TXD3	TXD3	/
RGMII-TXD2	TXD2	/
RGMII-TXD1/RMII-TXD1	TXD1	TXD1
RGMII-TXD0/RMII-TXD0	TXD0	TXD0
RGMII-TXCK/RMII-TXCK	TXCK	TXCK
RGMII-TXCTRL/RMII-TXEN	TXCTL	TXEN
RGMII-CLKIN/RMII-RXER	CLKIN	RXER
MDC	MDC	MDC
MDIO	MDIO	MDIO

Pin Name	RGMII	RMII
EPHY-25M	EPHY-25M	EPHY-25M

The following table describes the pin list of RGMII.

**Table 9-36 EMAC RGMII Pin List**

Pin Name	Description	Type
RGMII-TXD[3:0]	EMAC RGMII transmit data	O
RGMII-TXCTRL	EMAC RGMII transmit control	O
RGMII-TXCK	EMAC RGMII transmit clock	O
RGMII-RXD[3:0]	EMAC RGMII receive data	I
RGMII-RXCTRL	EMAC RGMII receive control	I
RGMII-RXCK	EMAC RGMII receive clock	I
RGMII-CKIN	EMAC RGMII 125M reference clock input	I
MDC	EMAC management data clock	O
MDIO	EMAC management data input output	I/O
EPHY-25M	25 MHz output for EMAC PHY	O

The following table describes the pin list of RMII.

**Table 9-37 EMAC RMII Pin List**

Pin Name	Description	Type
RMII-TXD[1:0]	EMAC RMII transmit data	O
RMII-TXEN	EMAC RMII transmit enable	O
RMII-TXCK	EMAC RMII transmit clock	I
RMII-RXD[1:0]	EMAC RMII receive data	I
RMII-CRS-DV	EMAC RMII receive data valid	I
RMII-RXER	EMAC RMII receive error	I
MDC	EMAC management data clock	O
MDIO	EMAC management data input output	I/O
EPHY-25M	25 MHz output for EMAC PHY	O

### 9.12.3.2 Clock Characteristics

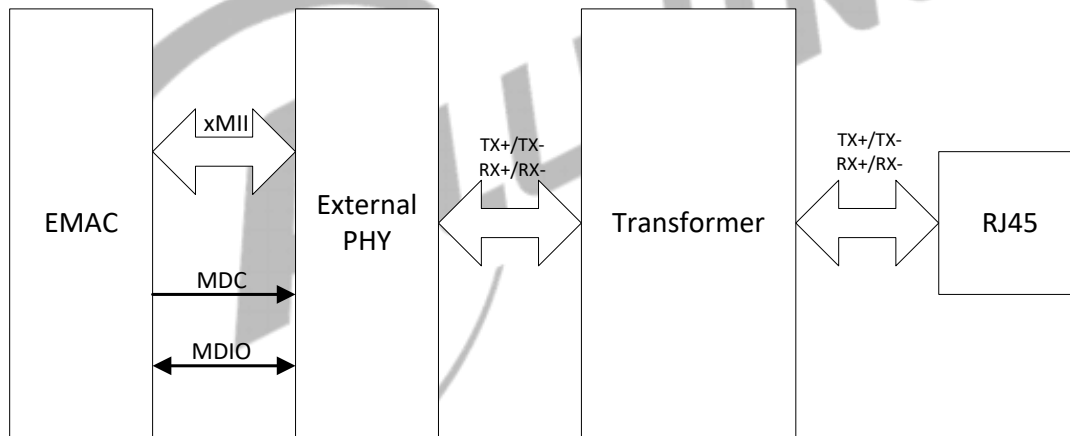
The following table describes the clock of EMAC.

**Table 9-38 EMAC Clock Characteristics**

Clock Name	Description	Type
RGMIIO-TXCK/RMIIIO-TXCK	In RGMII mode, output 2.5 MHz/25 MHz/125 MHz. In RMII mode, input 5 MHz/50 MHz.	O/I
RGMIIO-RXCK	In RGMII mode, input 2.5 MHz/25 MHz/125 MHz. In RMII mode, no input.	I
RGMIIO-CLKIN	In RGMII mode, input 125M Reference Clock. In RMII mode, no clock.	I

### 9.12.3.3 Typical Application

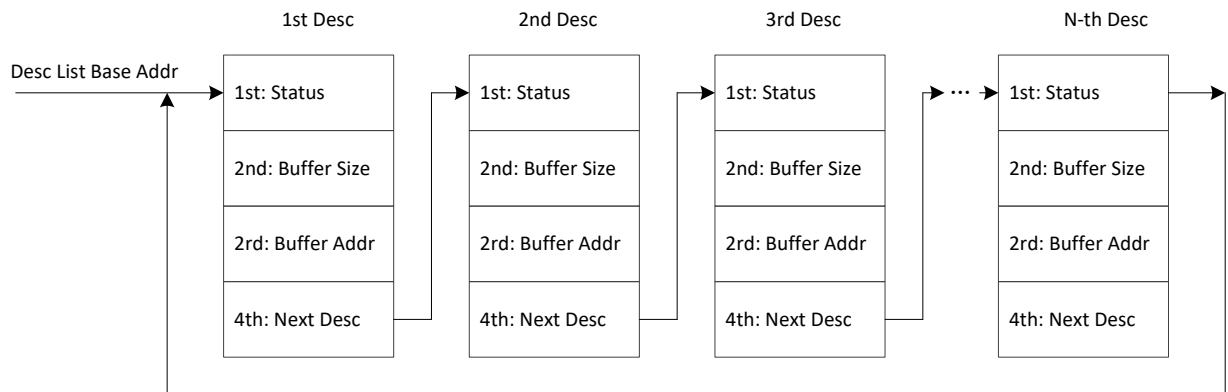
**Figure 9-104 EMAC Typical Application**



### 9.12.3.4 EMAC RX/TX Descriptor

The internal DMA of EMAC transfers data between host memory and internal RX/TX FIFO by a linked list of descriptors. Each descriptor consists of four words and contains some necessary information to transfer TX and RX frames. The following figure shows the descriptor list structure. The address of each descriptor must be 32-bit aligned.

Figure 9-105 EMAC RX/TX Descriptor List



### 9.12.3.5 TX Descriptor

#### 1st Word of TX Descriptor

Bits	Description
31	TX_DESC_CTL When set, the current descriptor can be used by DMA. This bit is cleared by DMA when the whole frame is transmitted or all data in the buffer of the current descriptor are transmitted.
30:17	Reserved
16	TX_HEADER_ERR When set, the checksum of the header for the transmitted frame is wrong.
15	Reserved
14	TX LENGHT_ERR When set, the length of the transmitted frame is wrong.
13	Reserved
12	TX_PAYLOAD_ERR When set, the checksum of the payload for the transmitted frame is wrong.
11	Reserved
10	TX_CR_S_ERR When set, the carrier is lost during transmission.
9	TX_COL_ERR_0 When set, the frame is aborted because of a collision after the contention period.
8	TX_COL_ERR_1 When set, the frame is aborted because of too many collisions.
7	Reserved



Bits	Description
6:3	TX_COL_CNT The number of collisions before transmission.
2	TX_DEFER_ERR When set, the frame is aborted because of too much deferral.
1	TX_UNDERFLOW_ERR When set, the frame is aborted because of the TX FIFO underflow error.
0	TX_DEFER When set in Half-Duplex mode, the EMAC defers the frame transmission.

### 2nd Word of TX Descriptor

Bits	Description
31	TX_INT_CTL When it is set and the current frame has been transmitted, the TX_INT in Interrupt Status Register will be set.
30	LAST_DESC When it is set, the current descriptor is the last one of the current frame.
29	FIR_DESC When it is set, the current descriptor is the first one of the current frame.
28:27	CHECKSUM_CTL These bits control to insert checksum in the transmit frame.
26	CRC_CTL When it is set, the CRC field is not transmitted.
25:11	Reserved
10:0	BUF_SIZE The size of the buffer specified by the current descriptor.

### 3rd Word of TX Descriptor

Bits	Description
31:0	BUF_ADDR The address of the buffer specified by the current descriptor.

#### 4th Word of TX Descriptor

Bits	Description
31:0	NEXT_DESC_ADDR The address of the next descriptor. It must be 32-bit aligned.

#### 9.12.3.6 RX Descriptor

##### 1st Word of RX Descriptor

Bits	Description
31	RX_DESC_CTL When it is set, the current descriptor can be used by DMA. This bit is cleared by DMA when the complete frame is received or the buffer of the current descriptor is full.
30	RX_DAF_FAIL When it is set, the current frame does not pass the DA filter.
29:16	RX_FRM_LEN When LAST_DESC is not set and no error bit is set, this field is the length of received data for the current frame. When LAST_DESC is set, RX_OVERFLOW_ERR and RX_NO_ENOUGH_BUF_ERR are not set, this field is the length of the received frame.
15	Reserved
14	RX_NO_ENOUGH_BUF_ERR When it is set, the current frame is clipped because of no enough buffer.
13	RX_SAF_FAIL When it is set, the current frame does not pass the SA filter.
12	Reserved
11	RX_OVERFLOW_ERR When it is set, a buffer overflow error occurred and the current frame is wrong.
10	Reserved
9	FIR_DESC When it is set, the current descriptor is the first descriptor of the current frame.
8	LAST_DESC When it is set, the current descriptor is the last descriptor of the current frame.
7	RX_HEADER_ERR When it is set, the checksum of the frame header is wrong.

Bits	Description
6	RX_COL_ERR When it is set, there is a late collision during the reception in half-duplex mode.
5	Reserved
4	RX_LENGTH_ERR When it is set, the length of the current frame is wrong.
3	RX_PHY_ERR When it is set, the receive error signal from PHY is asserted during the reception.
2	Reserved
1	RX_CRC_ERR When it is set, the CRC field of the received frame is wrong.
0	RX_PAYLOAD_ERR When it is set, the checksum or length of the payload for the received frame is wrong.

#### 2nd Word of RX Descriptor

Bits	Description
31	RX_INT_CTL When it is set and a frame has been received, the RX_INT will not be set.
30:11	Reserved
10:0	BUF_SIZE The size of the buffer is specified by the current descriptor.

#### 3rd Word of RX Descriptor

Bits	Description
31:0	BUF_ADDR The address of the buffer specified by the current descriptor.

#### 4th Word of RX Descriptor

Bits	Description
31:0	NEXT_DESC_ADDR The address of the next descriptor. This field must be 32-bit aligned.

## 9.12.4 Programming Guidelines

### 9.12.4.1 EMAC System Configuration

Perform the following steps:

- Step 1** Write 0 to [EMAC\\_BGR\\_REG](#)[bit16] to assert the module reset.
- Step 2** Write 1 to [EMAC\\_BGR\\_REG](#)[bit16] to deassert the module reset.
- Step 3** Write 1 to [EMAC\\_BGR\\_REG](#)[bit0] to enable the bus clock of the module.
- Step 4** Configure the pin interfaces of EMAC by setting GPIO module.
- Step 5** Configure [EMAC\\_EPHY\\_CLK\\_REG0](#) to set the transmission clock source of RGMII/RMII.

For RGMII RXCLK/CLK125M:

In RGMII mode, in addition to the configuration of the transmission clock source, it is generally necessary to adjust the timing by configuring the transmission clock delay, reception clock delay, transmission clock reverse, reception clock reverse.

- Write 0 to the bit[13] and write 1 to the bit[2] to select the RGMII interface.
- If selecting RXCLK as the clock source of RGMII, write 2 to the bit[1:0]; if selecting CLK125M as the clock source of RGMII, write 1 to the bit[1:0].
- Write 0 to the bit[3], write 0 to the bit[4], write 31 to the bit[9:5], and write 7 to the bit[12:10] to transmit the reception sequence adjustment.

For RMII TXCLK:

- Write 1 to the bit[13] and write 0 to the bit[2] to select the RMII interface.
- Write 0 to the bit[0] to select TXCLK as the clock source of RMII.

The configuration value of [EMAC\\_EPHY\\_CLK\\_REG0](#) can refer to the following table.

**Table 9-39 EMAC\_EPHY\_CLK\_REG0 Configuration Value**

EMAC_EPHY_CLK_REG0	PHY_SEL	RMII_EN	ETXDC	ERXDC	ERXIE	ETXIE	RMII/RGMII	ETCS
	Bit15	Bit13	Bit[12:10]	Bit[9:5]	Bit4	Bit3	Bit2	Bit[1:0]
RGMII	0	0	7	31	0	0	1	1/2
RMII	0	1	0	0	0	0	0	0

### 9.12.4.2 EMAC Initialization

- Step 1** Write 1 to [EMAC\\_BASIC\\_CTL1](#)[bit0] to perform the software reset.

- Step 2** Write 1 to [EMAC\\_BASIC\\_CTL1](#)[bit1] to set the DMA priority of TX/RX.
- Step 3** Configure [EMAC\\_TX\\_CTL1](#) and [EMAC\\_RX\\_CTL1](#) to set the configuration of DMA TX and DMA RX.
- Step 4** Configure [EMAC\\_INT\\_EN](#) to set the corresponding interrupts and shield the needless interrupts.
- Step 5** Configure [EMAC\\_TX\\_DMA\\_LIST](#) and [EMAC\\_RX\\_DMA\\_LIST](#) to set the first address of the TX descriptor and the RX descriptor, respectively.
- Step 6** Configure [EMAC\\_TX\\_CTLO](#) and [EMAC\\_RX\\_CTLO](#) to set the TX and RX parameters. Configure [EMAC\\_BASIC\\_CTLO](#) to set the speed, duplex mode, loopback configuration. (If enabled the auto-negotiation, the configuration is performed as a result of the negotiation)
- Step 7** Configure [EMAC\\_RX\\_FRM\\_FLT](#) to set the RX frame filter.
- Step 8** Configure [EMAC\\_TX\\_FLOW\\_CTL](#) and [EMAC\\_RX\\_CTLO](#) to set the control mechanism of TX and RX.
- Step 9** Clear all interrupt flags.
- Step 10** Write 1 to [EMAC\\_TX\\_CTLO](#)[bit31] and write 1 to [EMAC\\_RX\\_CTLO](#)[bit31] to enable the TX and RX functions.

### 9.12.5 Register List

Module Name	Base Address
EMAC	0x04500000

Register Name	Offset	Description
EMAC_BASIC_CTLO	0x0000	EMAC Basic Control Register0
EMAC_BASIC_CTL1	0x0004	EMAC Basic Control Register1
EMAC_INT_STA	0x0008	EMAC Interrupt Status Register
EMAC_INT_EN	0x000C	EMAC Interrupt Enable Register
EMAC_TX_CTLO	0x0010	EMAC Transmit Control Register0
EMAC_TX_CTL1	0x0014	EMAC Transmit Control Register1
EMAC_TX_FLOW_CTL	0x001C	EMAC Transmit Flow Control Register
EMAC_TX_DMA_DESC_LIST	0x0020	EMAC Transmit Descriptor List Address Register
EMAC_RX_CTLO	0x0024	EMAC Receive Control Register0
EMAC_RX_CTL1	0x0028	EMAC Receive Control Register1
EMAC_RX_DMA_DESC_LIST	0x0034	EMAC Receive Descriptor List Address Register
EMAC_RX_FRM_FLT	0x0038	EMAC Receive Frame Filter Register
EMAC_RX_HASH0	0x0040	EMAC Hash Table Register0

Register Name	Offset	Description
EMAC_RX_HASH1	0x0044	EMAC Hash Table Register1
EMAC_MII_CMD	0x0048	EMAC Management Interface Command Register
EMAC_MII_DATA	0x004C	EMAC Management Interface Data Register
EMAC_ADDR_HIGH0	0x0050	EMAC MAC Address High Register0
EMAC_ADDR_LOW0	0x0054	EMAC MAC Address Low Register0
EMAC_ADDR_HIGHN	0x0050+0x08*N (N=1-7)	EMAC MAC Address High Register N (N=1-7)
EMAC_ADDR_LOWN	0x0054+0x08*N (N=1-7)	EMAC MAC Address Low Register N (N=1-7)
EMAC_TX_DMA_STA	0x00B0	EMAC Transmit DMA Status Register
EMAC_TX_CUR_DESC	0x00B4	EMAC Current Transmit Descriptor Register
EMAC_TX_CUR_BUF	0x00B8	EMAC Current Transmit Buffer Address Register
EMAC_RX_DMA_STA	0x00C0	EMAC Receive DMA Status Register
EMAC_RX_CUR_DESC	0x00C4	EMAC Current Receive Descriptor Register
EMAC_RX_CUR_BUF	0x00C8	EMAC Current Receive Buffer Address Register
EMAC_RGMII_STA	0x00D0	EMAC RGMII Status Register

### 9.12.6 Register Description

#### 9.12.6.1 0x0000 EMAC Basic Control Register0 (Default Value: 0x0000\_0000)

Offset: 0x0000			Register Name: EMAC_BASIC_CTL0
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
3:2	R/W	0x0	SPEED EMAC Working Speed 00: 1000 Mbit/s 01: Reserved 10: 10 Mbit/s 11: 100 Mbit/s
1	R/W	0x0	LOOPBACK EMAC Loopback Mode For Test 0: Disable 1: Enable

Offset: 0x0000			Register Name: EMAC_BASIC_CTL0
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	DUPLEX EMAC Transfer Mode 0: Half-duplex 1: Full-duplex

9.12.6.2 0x0004 EMAC Basic Control Register1 (Default Value: 0x0800\_0000)

Offset: 0x0004			Register Name: EMAC_BASIC_CTL1
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:24	R/W	0x8	BURST_LEN The burst length of RX and TX DMA transfer.
23:2	/	/	/
1	R/W	0x0	RX_TX_PRI RX TX DMA Priority 0: Same priority 1: RX priority is over TX
0	R/W	0x0	SOFT_RST Soft Reset all Registers and Logic 0: No valid 1: Reset All clock inputs must be valid before soft reset. This bit is cleared internally when the reset operation is completed fully. Before writing any register, this bit should read a 0.

9.12.6.3 0x0008 EMAC Interrupt Status Register (Default Value: 0x4000\_0000)

Offset: 0x0008			Register Name: EMAC_INT_STA
Bit	Read/Write	Default/Hex	Description
31:17	R	0x2000	Reserved
16	R/W1C	0x0	RGMII_LINK_STA_P RMII Link Status Changed Interrupt Pending 0: No Pending 1: Pending Write '1' to clear it.

Offset: 0x0008			Register Name: EMAC_INT_STA
Bit	Read/Write	Default/Hex	Description
15:14	/	/	/
13	R/W1C	0x0	<p>RX_EARLY_P RX DMA Filled First Data Buffer of the Receive Frame Interrupt Pending</p> <p>0: No Pending 1: Pending</p> <p>Write '1' to clear it.</p>
12	R/W1C	0x0	<p>RX_OVERFLOW_P RX FIFO Overflow Error Interrupt Pending</p> <p>0: No Pending 1: Pending</p> <p>Write '1' to clear it.</p>
11	R/W1C	0x0	<p>RX_TIMEOUT_P RX Timeout Interrupt Pending</p> <p>0: No Pending 1: Pending</p> <p>Write '1' to clear it. When this bit is asserted, the length of the received frame is greater than 2048 bytes (10240 when JUMBO_FRM_EN is set)</p>
10	R/W1C	0x0	<p>RX_DMA_STOPPED_P When this bit asserted, the RX DMA FSM is stopped.</p>
9	R/W1C	0x0	<p>RX_BUF_UA_P RX Buffer UA Interrupt Pending</p> <p>0: No Pending 1: Pending</p> <p>Write '1' to clear it. When this bit is asserted, the RX DMA cannot acquire the next RX descriptor and RX DMA FSM is suspended. The ownership of the next RX descriptor should be changed to RX DMA. The RX DMA FSM will resume when the RX_DMA_START is written or the next receive frame is coming.</p>
8	R/W1C	0x0	<p>RX_P Frame RX Completed Interrupt Pending</p> <p>0: No Pending 1: Pending</p> <p>Write '1' to clear it. When this bit is asserted, a frame reception is completed. The RX DMA FSM remains running.</p>
7:6	/	/	/



Offset: 0x0008			Register Name: EMAC_INT_STA
Bit	Read/Write	Default/Hex	Description
5	R/W1C	0x0	<p>TX_EARLY_P</p> <p>Total interrupt pending which the frame is transmitted to FIFO</p> <p>0: No Pending</p> <p>1: Pending</p> <p>Write '1' to clear it.</p>
4	R/W1C	0x0	<p>TX_UNDERFLOW_P</p> <p>TX FIFO Underflow Interrupt Pending</p> <p>0: No Pending</p> <p>1: Pending</p> <p>Write '1' to clear it.</p>
3	R/W1C	0x0	<p>TX_TIMEOUT_P</p> <p>Transmitter Timeout Interrupt Pending</p> <p>0: No Pending</p> <p>1: Pending</p> <p>Write '1' to clear it.</p>
2	R/W1C	0x0	<p>TX_BUF_UA_P</p> <p>TX Buffer UA Interrupt Pending</p> <p>0: No Pending</p> <p>1: Pending</p> <p>When this asserted, the TX DMA can not acquire the next TX descriptor and the TX DMA FSM is suspended. The ownership of the next TX descriptor should be changed to TX DMA. The TX DMA FSM will resume when writing to TX_DMA_START bit.</p>
1	R/W1C	0x0	<p>TX_DMA_STOPPED_P</p> <p>Transmission DMA Stopped Interrupt Pending</p> <p>0: No Pending</p> <p>1: Pending</p> <p>Write '1' to clear it.</p>
0	R/W1C	0x0	<p>TX_P</p> <p>Frame Transmission Interrupt Pending</p> <p>0: No Pending</p> <p>1: Pending</p> <p>Write '1' to clear it.</p>

9.12.6.4 0x000C EMAC Interrupt Enable Register (Default Value: 0x0000\_0000)

Offset: 0x000C			Register Name: EMAC_INT_EN
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13	R/W	0x0	RX_EARLY_INT_EN Early Receive Interrupt 0: Disable 1: Enable
12	R/W	0x0	RX_OVERFLOW_INT_EN Receive Overflow Interrupt 0: Disable 1: Enable
11	R/W	0x0	RX_TIMEOUT_INT_EN Receive Timeout Interrupt 0: Disable 1: Enable
10	R/W	0x0	RX_DMA_STOPPED_INT_EN Receive DMA FSM Stopped Interrupt 0: Disable 1: Enable
9	R/W	0x0	RX_BUF_UA_INT_EN Receive Buffer Unavailable Interrupt 0: Disable 1: Enable
8	R/W	0x0	RX_INT_EN Receive Interrupt 0: Disable 1: Enable
7:6	/	/	/
5	R/W	0x0	TX_EARLY_INT_EN Early Transmit Interrupt 0: Disable 1: Enable
4	R/W	0x0	TX_UNDERFLOW_INT_EN Transmit Underflow Interrupt 0: Disable 1: Enable

Offset: 0x000C			Register Name: EMAC_INT_EN
Bit	Read/Write	Default/Hex	Description
3	R/W	0x0	TX_TIMEOUT_INT_EN Transmit Timeout Interrupt 0: Disable 1: Enable
2	R/W	0x0	TX_BUF_UA_INT_EN Transmit Buffer Available Interrupt 0: Disable 1: Enable
1	R/W	0x0	TX_DMA_STOPPED_INT_EN Transmit DMA FSM Stopped Interrupt 0: Disable 1: Enable
0	R/W	0x0	TX_INT_EN Transmit Interrupt 0: Disable 1: Enable

9.12.6.5 0x0010 EMAC Transmit Control Register0 (Default Value: 0x0000\_0000)

Offset: 0x0010			Register Name: EMAC_TX_CTL0
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	TX_EN Enable Transmitter 0: Disable 1: Enable When disabled, the transmission will continue until the current transmission finishes.
30	R/W	0x0	TX_FRM_LEN_CTL Frame Transmit Length Control 0: Up to 2,048 bytes (JUMBO_FRM_EN==0) Up to 10,240 bytes (JUMBO_FRM_EN==1) 1: Up to 16,384 bytes Any bytes after that is cut off.
29:0	/	/	/

9.12.6.6 0x0014 EMAC Transmit Control Register1 (Default Value: 0x0000\_0000)

Offset: 0x0014			Register Name: EMAC_TX_CTL1
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	TX_DMA_START Transmit DMA FSM Start 0: No valid 1: Start It is cleared internally and always read a 0.
30	R/W	0x0	TX_DMA_EN 0: Stop TX DMA after the completion of current frame transmission 1: Start and run TX DMA
29:11	/	/	/
10:8	R/W	0x0	TX_TH Threshold value of TX DMA FIFO When TX_MD is 0, the transmission starts when the frame size in TX DMA FIFO is greater than the threshold. In addition, the full frames with a length less than the threshold are transferred automatically. 000: 64 001: 128 010: 192 011: 256 Others: Reserved
7:2	/	/	/
1	R/W	0x0	TX_MD Transmission Mode 0: TX starts after the TX DMA FIFO bytes is greater than the TX_TH 1: TX starts after the TX DMA FIFO is located a full frame
0	R/WAC	0x0	FLUSH_TX_FIFO Flush the data in the TX FIFO 0: Enable 1: Disable

9.12.6.7 0x001C EMAC Transmit Flow Control Register (Default Value: 0x0000\_0000)

Offset: 0x001C			Register Name: EMAC_TX_FLOW_CTL
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	TX_FLOW_CTL_STA This bit indicates a pause frame transmission is in progress. When the configuration of flow control is ready, set this bit to transmit a pause frame in full-duplex mode or activate the backpressure function. After the transmission is completed, this bit will be cleared automatically. Before writing TX_FLOW_CTRL register, this bit must be read as 0.
30:22	/	/	/
21:20	R/W	0x0	TX_PAUSE_FRM_SLOT The threshold of the pause timer at which the input flow control signal is checked for automatic re-transmission of the pause frame. The threshold values should be always less than PAUSE_TIME.
19:4	R/W	0x0	PAUSE_TIME The pause time field in the transmitted control frame.
3:2	/	/	/
1	R/W	0x0	ZQP_FRM_EN 0: Disable 1: Enable When set, enable the functionality to generate the Zero-Quanta Pause control frame.
0	R/W	0x0	TX_FLOW_CTL_EN TX Flow Control Enable 0: Disable 1: Enable When set, enable flow control operation to transmit pause frames in full-duplex mode, or enable the back-pressure operation in half-duplex mode.

9.12.6.8 0x0020 EMAC Transmit DMA Descriptor List Address Register (Default Value: 0x0000\_0000)

Offset: 0x0020			Register Name: EMAC_TX_DMA_LIST
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	TX_DESC_LIST The base address of the transmission descriptor list It must be 32-bit aligned.

9.12.6.9 0x0024 EMAC Receive Control Register0 (Default Value: 0x0000\_0000)

Offset: 0x0024			Register Name: EMAC_RX_CTL0
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	RX_EN Enable Receiver 0: Disable receiver after current reception 1: Enable
30	R/W	0x0	RX_FRM_LEN_CTL Frame Receive Length Control 0: Up to 2048 bytes (JUMBO_FRM_EN==0) Up to 10240 bytes (JUMBO_FRM_EN==1) 1: Up to 16384 bytes Any bytes after that is cut off.
29	R/W	0x0	JUMBO_FRM_EN Jumbo Frame Enable 0: Disable 1: Enable Jumbo frames of 9018 bytes without reporting a giant
28	R/W	0x0	STRIP_FCS When set, strip the Pad/FCS field on received frames only when the length of field value is less than or equal to 1500 bytes.
27	R/W	0x0	CHECK_CRC Check CRC Enable 0: Disable 1: Calculate CRC and check the IPv4 Header Checksum
26:18	/	/	/
17	R/W	0x0	RX_PAUSE_FRM_MD 0: Only detect multicast pause frame specified in the 802.3x standard. 1: In addition to detect multicast pause frame specified in the 802.3x standard, also detect unicast pause frame with the address specified in MAC Address 0 High Register and MAC address 0 Low Register.
16	R/W	0x0	RX_FLOW_CTL_EN When set, enable the functionality that decodes the received pause frame and disable its transmitter for a specified time by pause frame.
15:0	/	/	/

9.12.6.10 0x0028 EMAC Receive Control Register1 (Default Value: 0x0000\_0000)

Offset: 0x0028			Register Name: EMAC_RX_CTL1
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	RX_DMA_START When set, the RX DMA will work. It is cleared internally and always read a 0.
30	R/W	0x0	RX_DMA_EN Receive DMA Enable 0: Stop RX DMA after finishing the received current frame 1: Start and run RX DMA
29:25	/	/	/
24	R/W	0x0	RX_FIFO_FLOW_CTL Receive FIFO Flow Control Enable 0: Disable 1: Enable, base on RX_FLOW_CTL_TH_DEACT and RX_FLOW_CTL_TH_ACT
23:22	R/W	0x0	RX_FLOW_CTL_TH_DEACT Threshold for Deactivating Flow Control 00: Full minus 1 KB 01: Full minus 2 KB 10: Full minus 3 KB 11: Full minus 4 KB Valid in both half-duplex mode and full-duplex mode.
21:20	R/W	0x0	RX_FLOW_CTL_TH_ACT Threshold for Activating Flow Control 00: Full minus 1 KB 01: Full minus 2 KB 10: Full minus 3 KB 11: Full minus 4 KB Valid in both half-duplex mode and full-duplex mode.
19:6	/	/	/

Offset: 0x0028			Register Name: EMAC_RX_CTL1
Bit	Read/Write	Default/Hex	Description
5:4	R/W	0x0	<p>RX_TH Threshold for RX DMA FIFO Start</p> <p>00: 64 01: 32 10: 96 11: 128</p> <p>Only valid when RX_MD == 0, the full frames with a length less than the threshold are transferred automatically.</p>
3	R/W	0x0	<p>RX_ERR_FRM 0: RX DMA drops frames with error 1: RX DMA forwards frames with error</p>
2	R/W	0x0	<p>RX_RUNT_FRM When the bit is set to 1, it indicates forward undersized frames with no error and length less than 64 bytes.</p>
1	R/W	0x0	<p>RX_MD Receive Mode</p> <p>0: RX starts to read after the RX DMA FIFO byte is greater than RX_TH 1: RX starts to read after the RX DMA FIFO is located a full frame</p>
0	R/W	0x0	<p>FLUSH_RX_FRM Flush Receive Frames</p> <p>0: Enable when the receive descriptors/buffers are unavailable 1: Disable</p>

**9.12.6.11 0x0034 EMAC Receive DMA Descriptor List Address Register (Default Value: 0x0000\_0000)**

Offset: 0x0034			Register Name: EMAC_RX_DMA_LIST
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	<p>RX_DESC_LIST The base address of the received descriptor list It must be 32-bit aligned.</p>



9.12.6.12 0x0038 EMAC Receive Frame Filter Register (Default Value: 0x0000\_0000)

Offset: 0x0038			Register Name: EMAC_RX_FRM_FLT
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	DIS_ADDR_FILTER Disable Address Filter 0: Enable 1: Disable
30:18	/	/	/
17	R/W	0x0	DIS_BROADCAST Disable Receive Broadcast Frames 0: Receive 1: Drop
16	R/W	0x0	RX_ALL_MULTICAST Receive All Multicast Frames Filter 0: Filter according to HASH_MULTICAST 1: Receive all
15:14	/	/	/
13:12	R/W	0x0	CTL_FRM_FILTER Receive Control Frames Filter 00: Drop all control frames 01: Drop all control frames 10: Receive all control frames 11: Receive all control frames when passing the address filter
11:10	/	/	/
9	R/W	0x0	HASH_MULTICAST Filter Multicast Frames Set 0: By comparing the DA field in DA MAC address registers 1: According to the hash table
8	R/W	0x0	HASH_UNICAST Filter Unicast Frames Set 0: By comparing the DA field in DA MAC address registers 1: According to the hash table
7	/	/	/

Offset: 0x0038			Register Name: EMAC_RX_FRM_FLT
Bit	Read/Write	Default/Hex	Description
6	R/W	0x0	SA_FILTER_EN Receive SA Filter Enable 0: Receive frames and update the result of SA filter 1: Update the result of the SA filter. In addition, if the SA field of the received frame does not match the values in SA MAC address registers, drop this frame.
5	R/W	0x0	SA_INV_FILTER Receive SA Invert Filter Set 0: Pass frames whose SA field matches SA MAC address registers 1: Pass frames whose SA field does not match SA MAC address registers
4	R/W	0x0	DA_INV_FILTER 0: Normal filtering of frames is performed 1: Filter both unicast and multicast frames by comparing DA field in inverse filtering mode
3:2	/	/	/
1	R/W	0x0	FLT_MD 0: If the HASH_MULTICAST or HASH_UNICAST is set, the frame is passed only when it matches the Hash filter 1: Receive the frame when it passes the address register filter or the hash filter (set by HASH_MULTICAST or HASH_UNICAST)
0	R/W	0x0	RX_ALL Receive All Frame 0: Receive the frames that pass the SA/DA address filter 1: Receive all frames and update the result of address filter (pass or fail) in the receive status word

9.12.6.13 0x0040 EMAC Receive Hash Table Register0 (Default Value: 0x0000\_0000)

Offset: 0x0040			Register Name: EMAC_RX_HASH0
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	HASH_TAB0 The upper 32 bits of Hash table for the received frame filter.

9.12.6.14 0x0044 EMAC Receive Hash Table Register1 (Default Value: 0x0000\_0000)

Offset: 0x0044			Register Name: EMAC_RX_HASH1
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	HASH_TAB1 The lower 32 bits of Hash table for the received frame filter.

9.12.6.15 0x0048 EMAC MII Command Register (Default Value: 0x0000\_0000)

Offset: 0x0048			Register Name: EMAC_MII_CMD
Bit	Read/Write	Default/Hex	Description
31:23	/	/	/
22:20	R/W	0x0	MDC_DIV_RATIO_M MDC Clock Divider Ratio The MDC Clock is divided from the AHB clock. 000: 16 001: 32 010: 64 011: 128 Others: Reserved
19:17	/	/	/
16:12	R/W	0x0	PHY_ADDR PHY Address
11:9	/	/	/
8:4	R/W	0x0	PHY_REG_ADDR PHY Register Address
3:2	/	/	/
1	R/W	0x0	MII_WR MII Write and Read 0: Read 1: Write
0	R/WAC	0x0	MII_BUSY MII Status 0: Writing 0 is no valid, and reading 0 indicates the read/write operation is finished 1: Writing 1 starts the read/write operation, and reading 1 indicates busy.

9.12.6.16 0x004C EMAC MII Data Register (Default Value: 0x0000\_0000)

Offset: 0x004C			Register Name: EMAC_MII_DATA
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x0	MII_DATA Write to or read from the register in the selected PHY.

9.12.6.17 0x0050 EMAC MAC Address High Register0 (Default Value: 0x0000\_FFFF)

Offset: 0x0050			Register Name: EMAC_ADDR_HIGH0
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0xFFFF	MAC_ADDR_HIGH0 The upper 16 bits of the 1st MAC address.

9.12.6.18 0x0054 EMAC MAC Address Low Register0 (Default Value: 0xFFFF\_FFFF)

Offset: 0x0054			Register Name: EMAC_ADDR_LOW0
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0xFFFFFFFF	MAC_ADDR_LOW0 The lower 32 bits of 1st MAC address.

9.12.6.19 0x0050+0x08\*N EMAC MAC Address High Register N (Default Value: 0x0000\_FFFF)

Offset: 0x0050+0x08*N (N=1~7)			Register Name: EMAC_ADDR_HIGHN
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	MAC_ADDR_CTL MAC Address Valid 0: Not valid 1: Valid
30	R/W	0x0	MAC_ADDR_TYPE MAC Address Type 0: Used to compare with the destination address of the received frame 1: Used to compare with the source address of the received frame

Offset: 0x0050+0x08*N (N=1~7)			Register Name: EMAC_ADDR_HIGHN
Bit	Read/Write	Default/Hex	Description
29:24	R/W	0x0	MAC_ADDR_BYTE_CTL MAC Address Byte Control Mask The lower bit of mask controls the lower byte of the MAC address. When the bit of mask is 1, do not compare the corresponding byte.
23:16	/	/	/
15:0	R/W	0xFFFF	MAC_ADDR_HIGH The upper 16 bits of the MAC address.

**9.12.6.20 0x0054+0x08\*N EMAC MAC Address Low Register N (Default Value: 0x0000\_0000)**

Offset: 0x0054+0x08*N (N=1~7)			Register Name: EMAC_ADDR_LOWN
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	MAC_ADDR_LOWN The lower 32 bits of MAC address N (N: 1-7).

**9.12.6.21 0x00B0 EMAC Transmit DMA Status Register (Default Value: 0x0000\_0000)**

Offset: 0x00B0			Register Name: EMAC_TX_DMA_STA
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/
2:0	R	0x0	TX_DMA_STA The State of Transmit DMA FSM 000: STOP, when reset or disable TX DMA 001: RUN_FETCH_DESC, fetching TX DMA descriptor 010: RUN_WAIT_STA, waiting for the status of TX frame 011: RUN_TRANS_DATA, passing the frame from host memory to TX DMA FIFO 100: Reserved 101: Reserved 111: RUN_CLOSE_DESC, closing TX descriptor 110: SUSPEND, TX descriptor is unavailable or TX DMA FIFO underflow

9.12.6.22 0x00B4 EMAC Transmit DMA Current Descriptor Register (Default Value: 0x0000\_0000)

Offset: 0x00B4			Register Name: EMAC_TX_DMA_CUR_DESC
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	TX_DMA_CUR_DESC The address of current transmit descriptor.

9.12.6.23 0x00B8 EMAC Transmit DMA Current Buffer Address Register (Default Value: 0x0000\_0000)

Offset: 0x00B8			Register Name: EMAC_TX_DMA_CUR_BUF
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	TX_DMA_CUR_BUF The address of current transmit DMA buffer.

9.12.6.24 0x00C0 EMAC Receive DMA Status Register (Default Value: 0x0000\_0000)

Offset: 0x00C0			Register Name: EMAC_RX_DMA_STA
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/
2:0	R	0x0	RX_DMA_STA The State of RX DMA FSM 000: STOP, when reset or disable RX DMA 001: RUN_FETCH_DESC, fetching RX DMA descriptor 010: Reserved 011: RUN_WAIT_FRM, waiting for the frame 100: SUSPEND, RX descriptor is unavailable 101: RUN_CLOSE_DESC, closing RX descriptor 110: Reserved 111: RUN_TRANS_DATA, passing the frame from host memory to RX DMA FIFO

9.12.6.25 0x00C4 EMAC Receive DMA Current Descriptor Register (Default Value: 0x0000\_0000)

Offset: 0x00C4			Register Name: EMAC_RX_DMA_CUR_DESC
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	RX_DMA_CUR_DESC The address of current receive descriptor

9.12.6.26 0x00C8 EMAC Receive DMA Current Buffer Address Register (Default Value: 0x0000\_0000)

Offset: 0x00C8			Register Name: EMAC_RX_DMA_CUR_BUF
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	RX_DMA_CUR_BUF The address of current receive DMA buffer

9.12.6.27 0x00D0 EMAC RGMII Status Register (Default Value: 0x0000\_0000)

Offset: 0x00D0			Register Name: EMAC_RGMII_STA
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
3	R	0x0	RGMII_LINK The link status of the RGMII interface 0: Down 1: Up
2:1	R	0x0	RGMII_LINK_SPD The link speed of the RGMII interface 00: 2.5 MHz 01: 25 MHz 10: 125 MHz 11: Reserved
0	R	0x0	RGMII_LINK_MD The link mode of the RGMII interface 0: Half-Duplex 1: Full-Duplex

## 9.13 CIR Receiver

### 9.13.1 Overview

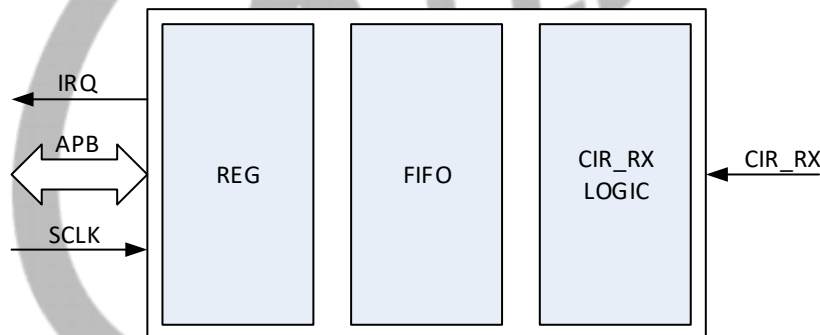
The Consumer Infrared receiver (CIR\_RX) captures pulse from the IR Receiver module and uses the Run-Length Code (RLC) to encode the pulse.

The CIR receiver has the following features:

- Supports CIR remote control receiver
- Supports NEC IR protocol
- 64x8 bits RX FIFO for data buffer
- Programmable RX FIFO thresholds
- Supports interrupt
- Sample clock up to 1 MHz

### 9.13.2 Block Diagram

Figure 9-106 CIR\_RX Block Diagram



The CIR\_RX samples the input signal on the programmable frequency and records these samples into RX FIFO when one CIR signal is found on the air. The CIR\_RX uses Run-Length Code (RLC) to encode pulse width. The encoded data is buffered in 64 levels and 8-bit width RX FIFO; the MSB bit is used to record the polarity of the receiving CIR signal, the rest 7 bits are used for the length of RLC. The maximum length of the RLC is 128. If the duration of one level (high or low level) is more than 128, another byte is used.



### 9.13.3 Functional Description

#### 9.13.3.1 External Signals

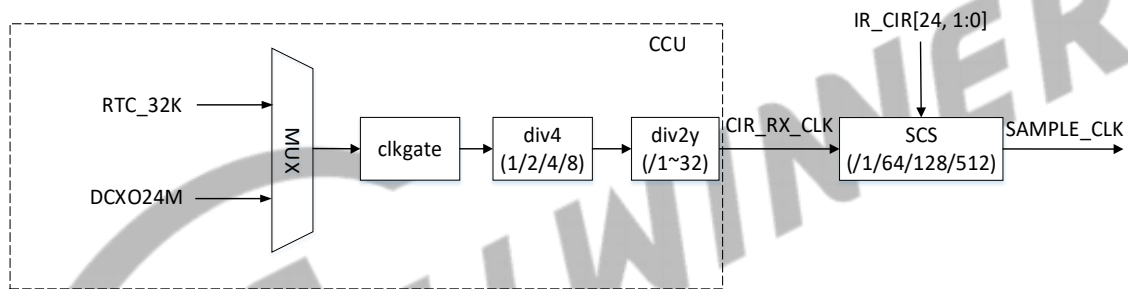
The following table describes the external signals of CIR\_RX.

**Table 9-40 CIR\_RX External Signals**

Signal	Description	Type
IR-RX	Consumer Infrared Receiver	I

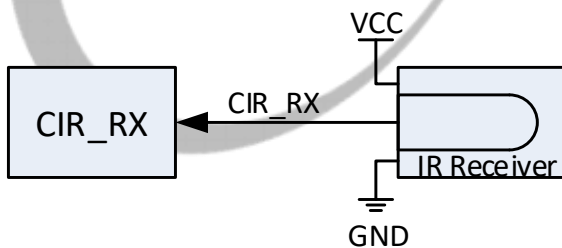
#### 9.13.3.2 Clock Sources

**Figure 9-107 CIR\_RX Clock System**



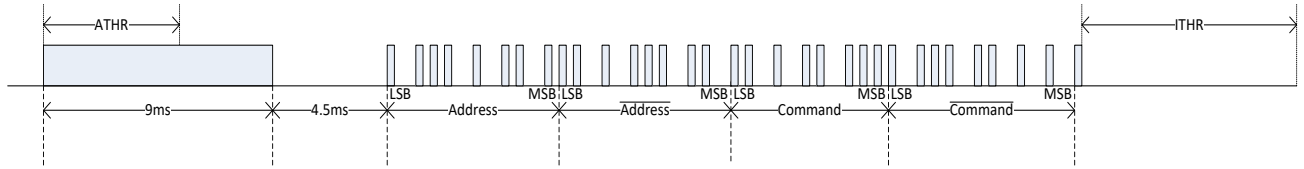
#### 9.13.3.3 Typical Application

**Figure 9-108 CIR\_RX Application Diagram**



### 9.13.3.4 NEC Protocol Format

Figure 9-109 NEC Protocol



The CIR\_RX module is a timer with a capture function.

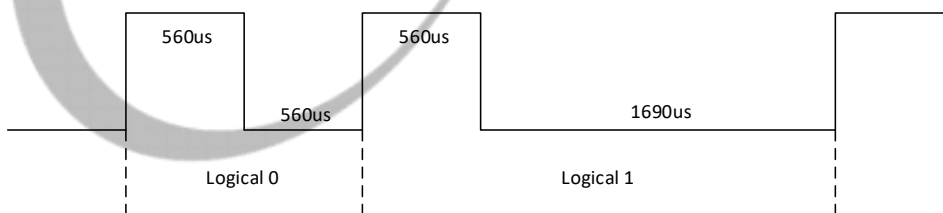
When CIR\_RX signals satisfy the Active Threshold (ATHR), the CIR receiver can start to capture. In the process, the signal is ignored if the pulse width of the signal is less than NTHR. When CIR\_RX signals satisfy ITHR (Idle Threshold), the capture process is stopped and the Receiver Packet End interrupt is generated, then the Receiver Packet End Flag is asserted.

In a capture process, every effective pulse is buffered to FIFO in bytes according to the form of the Run-Length Code. The MSB bit of a byte is the polarity of pulse, and the rest 7 bits is pulse width by taking Sample Clock as a basic unit. This is the code form of the RLC-Byte. When the level changes or the pulse width counter overflows, the RLC-Byte is buffered to FIFO. The CIR\_RX module receives the infrared signals transmitted by the infrared remote control, the software decodes the signals.

### 9.13.3.5 Operating Mode

#### Sample Clock

Figure 9-110 Logical '0' and Logical '1' of NEC Protocol



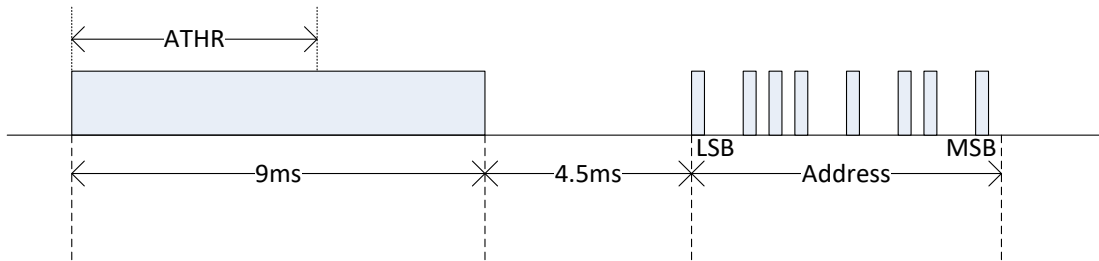
For NEC protocol, a logical "1" takes 2.25 ms (560 us+1680 us) to transmit, while a logical "0" is only half of that, being 1.12 ms (560 us+560 us).

For example, if the sample clock is 31.25 kHz, a sample cycle is 32 us, then 18 sample cycles are 560 us. So the RLC of 560 us low level is 0x12 (b'00010010), the RLC of 560 us high level is 0x92 (b'10010010). Then a logical "1" takes code 0x12 (b'00010010) and code 0xb5 (b'10110101) to transmit, a logical "0" takes code 0x12 and code 0x92 to transmit.

**Active Threshold (ATHR)**

When the CIR\_RX is in Idle state, if the electrical level of the CIR\_RX signal changes (positive jump or negative jump), and the duration reaches this threshold, then the CIR\_RX takes the starting of the signal as a lead code, and the CIR\_RX turns into an active state and starts to capture CIR\_RX signals.

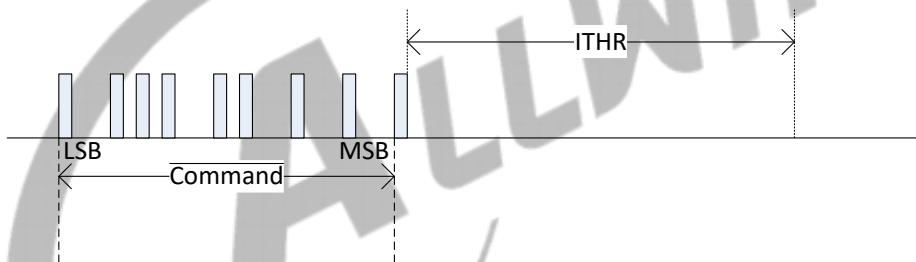
**Figure 9-111 ATHR Definition**



**Idle Threshold (ITHR)**

If the electrical level of CIR\_RX signals has no change, and the duration reaches this threshold, then the CIR\_RX enters into Idle state and ends this capture.

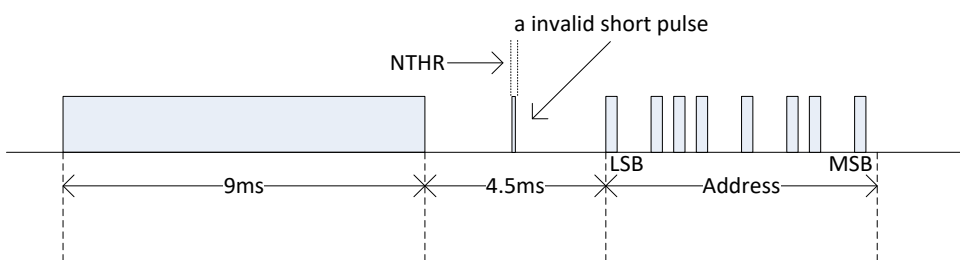
**Figure 9-112 ITHR Definition**



**Noise Threshold (NTHR)**

In the capture process, the pulse is ignored if the pulse width is less than the Noise Threshold.

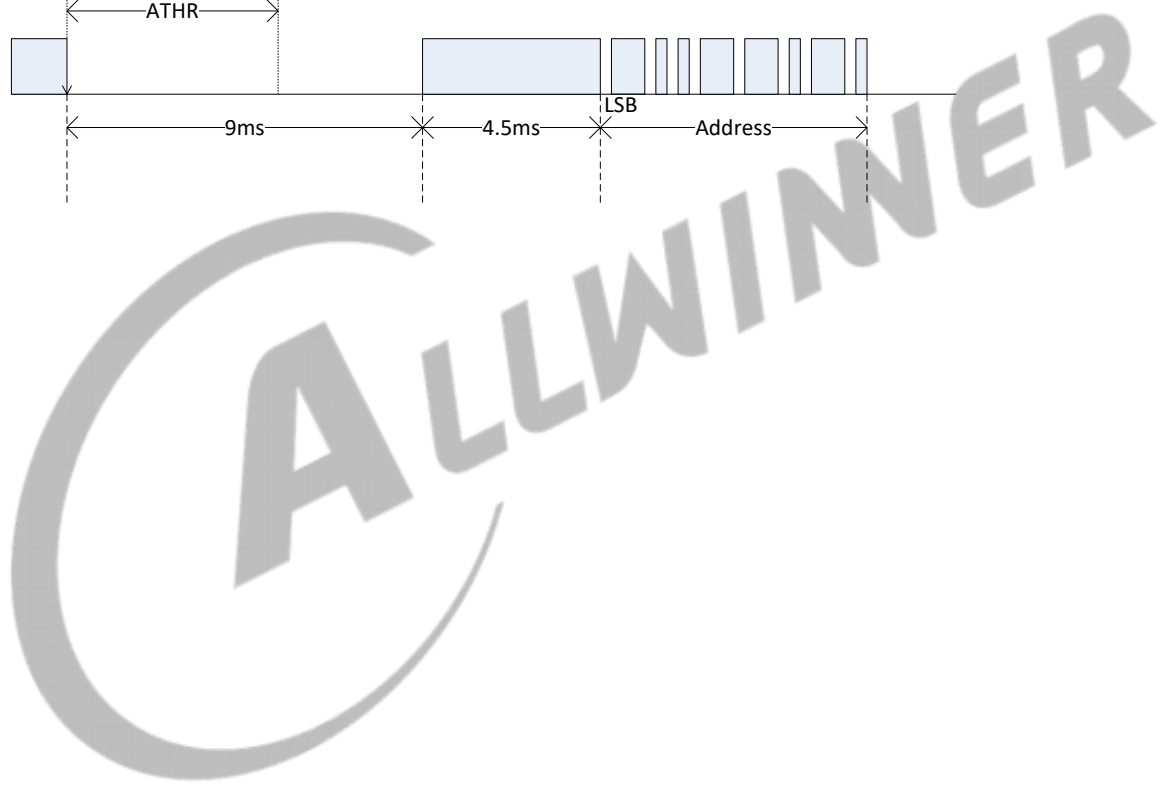
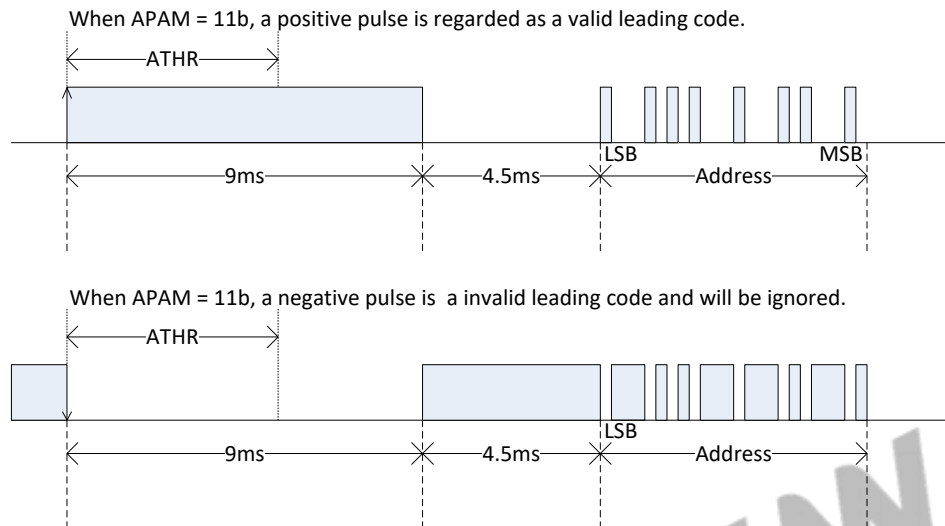
**Figure 9-113 NTHR Definition**



### Active Pulse Accept Mode (APAM)

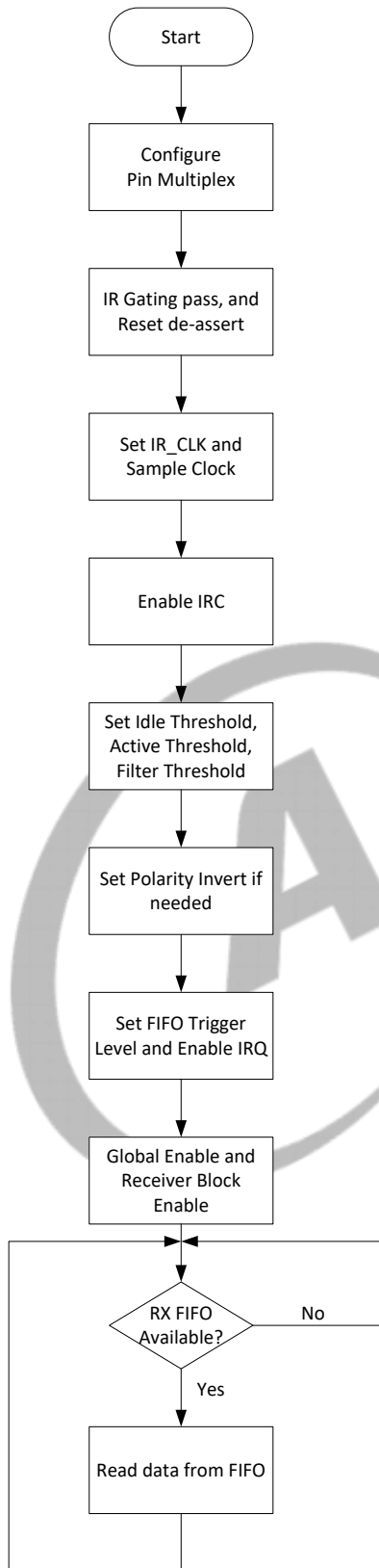
The APAM is used to fit the type of lead code. If a pulse does not fit the type of lead code, it is not regarded as a lead code even if the pulse width reaches ATHR.

Figure 9-114 APAM Definition



### 9.13.4 Programming Guidelines

Figure 9-115 CIR Receiver Process



### 9.13.5 Register List

Module Name	Base Address
CIR_RX	0x07040000

Register Name	Offset	Description
CIR_CTL	0x0000	CIR Control Register
CIR_RXPCFG	0x0010	CIR Receiver Pulse Configure Register
CIR_RXFIFO	0x0020	CIR Receiver FIFO Register
CIR_RXINT	0x002C	CIR Receiver Interrupt Control Register
CIR_RXSTA	0x0030	CIR Receiver Status Register
CIR_RXCFG	0x0034	CIR Receiver Configure Register

### 9.13.6 Register Description

#### 9.13.6.1 0x0000 CIR Receiver Control Register (Default Value: 0x0000\_0000)

Offset: 0x0000			Register Name: CIR_CTL
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:6	R/W	0x0	APAM Active Pulse Accept Mode 00, 01: Both positive and negative pulses are valid as a leading code 10: Only negative pulse is valid as a leading code 11: Only positive pulse is valid as a leading code
5:4	R/W	0x0	CIR ENABLE 00–10: Reserved 11: CIR mode enable
3:2	/	/	/
1	R/W	0x0	RXEN Receiver Block Enable 0: Disable 1: Enable

Offset: 0x0000			Register Name: CIR_CTL
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	GEN Global Enable A disable on this bit overrides any other block or channel enables and flushes all FIFOs. 0: Disable 1: Enable

**9.13.6.2 0x0010 CIR Receiver Pulse Configure Register (Default Value: 0x0000\_0004)**

Offset: 0x0010			Register Name: CIR_RXPCFG
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/
2	R/W	0x1	RPPI Receiver Pulse Polarity Invert 0: Do not invert receiver signal 1: Invert receiver signal
1:0	/	/	/

**9.13.6.3 0x0020 CIR Receiver FIFO Register (Default Value: 0x0000\_0000)**

Offset: 0x0020			Register Name: CIR_RXFIFO
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R	0x0	RBF Receiver Byte FIFO

**9.13.6.4 0x002C CIR Receiver Interrupt Control Register (Default Value: 0x0000\_0000)**

Offset: 0x002C			Register Name: CIR_RXINT
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/

Offset: 0x002C			Register Name: CIR_RXINT
Bit	Read/Write	Default/Hex	Description
13:8	R/W	0x0	RAL RX FIFO available received byte level for interrupt and DMA request TRIGGER_LEVEL = RAL + 1
5	R/W	0x0	DRQ_EN RX FIFO DMA Enable 0: Disable 1: Enable When it is set to '1', the Receiver FIFO DRQ is asserted if reaching RAL. The DRQ is de-asserted when the condition fails.
4	R/W	0x0	RAI_EN RX FIFO Available Interrupt Enable 0: Disable 1: Enable When it is set to '1', the Receiver FIFO IRQ is asserted if reaching RAL. The IRQ is de-asserted when the condition fails.
3:2	/	/	/
1	R/W	0x0	RPEI_EN Receiver Packet End Interrupt Enable 0: Disable 1: Enable
0	R/W	0x0	ROI_EN Receiver FIFO Overrun Interrupt Enable 0: Disable 1: Enable

9.13.6.5 0x0030 CIR Receiver Status Register (Default Value: 0x0000\_0000)

Offset: 0x0030			Register Name: CIR_RXSTA
Bit	Read/Write	Default/Hex	Description
31:15	/	/	/



Offset: 0x0030			Register Name: CIR_RXSTA
Bit	Read/Write	Default/Hex	Description
14:8	R	0x0	<p>RAC</p> <p>RX FIFO Available Counter</p> <p>0: No available data in RX FIFO</p> <p>1: 1-byte available data in RX FIFO</p> <p>2: 2-bytes available data in RX FIFO</p> <p>...</p> <p>64: 64-bytes available data in RX FIFO</p>
7	R	0x0	<p>STAT</p> <p>Status of CIR</p> <p>0: Idle</p> <p>1: Busy</p>
6:5	/	/	/
4	R/W1C	0x0	<p>RA</p> <p>RX FIFO Available</p> <p>0: RX FIFO not available according to its level</p> <p>1: RX FIFO available according to its level</p> <p>Writing 1 clears this bit.</p>
3:2	/	/	/
1	R/W1C	0x0	<p>RPE</p> <p>Receiver Packet End Flag</p> <p>0: STO was not detected. In CIR mode, one CIR symbol is receiving or not detected.</p> <p>1: STO field or packet abort symbol (7'b0000,000 and 8'b0000,0000 for MIR and FIR) is detected. In CIR mode, one CIR symbol is received.</p> <p>Writing 1 clears this bit.</p>
0	R/W1C	0x0	<p>ROI</p> <p>Receiver FIFO Overrun</p> <p>0: Receiver FIFO not overrun</p> <p>1: Receiver FIFO overrun</p> <p>Writing 1 clears this bit.</p>

9.13.6.6 0x0034 CIR Receiver Configure Register (Default Value: 0x0000\_1828)

Offset: 0x0034			Register Name: CIR_RXCFG
Bit	Read/Write	Default/Hex	Description
31:25	/	/	/
24	R/W	0x0	SCS2 Bit2 of Sample Clock Select for CIR This bit is defined by SCS bits below.
23	R/W	0x0	ATHC Active Threshold Control for CIR 0: ATHR in a unit of (Sample Clock) 1: ATHR in a unit of (128*Sample Clocks)
22:16	R/W	0x0	ATHR Active Threshold for CIR These bits control the duration of CIR from the idle to the active state. The duration can be calculated by ((ATHR + 1)*(ATHC? Sample Clock: 128*Sample Clock)).
15:8	R/W	0x18	ITHR Idle Threshold for CIR The Receiver uses it to decide whether the CIR command is received. If there is no CIR signal on the air, the receiver is staying in IDLE status. One active pulse will bring the receiver from IDLE status to Receiving status. After the CIR receiver ends, the inputting signal will keep the specified level (high or low level) for a long time. The receiver can use this idle signal duration to decide that it has received the CIR command. The corresponding flag is asserted. If the corresponding interrupt is enabled, the interrupt line is asserted to the CPU.  When the duration of the signal keeps one status (high or low level) for the specified duration ((ITHR + 1)*128 sample_clk), this means that the previous CIR command is finished.

Offset: 0x0034			Register Name: CIR_RXCFG																																				
Bit	Read/Write	Default/Hex	Description																																				
7:2	R/W	0xA	<p>NTHR</p> <p>Noise Threshold for CIR</p> <p>When the duration of the signal pulse (high or low level) is less than NTHR, the pulse is taken as noise and should be discarded by hardware.</p> <p>0: All samples are recorded into RX FIFO</p> <p>1: If the signal is only one sample duration, it is taken as noise and discarded.</p> <p>2: If the signal is less than (&lt;=) two sample duration, it is taken as noise and discarded.</p> <p>...</p> <p>61: If the signal is less than (&lt;=) sixty-one sample duration, it is taken as noise and discarded.</p>																																				
1:0	R/W	0x0	<p>SCS</p> <p>Sample Clock Select for CIR</p> <table border="1"> <thead> <tr> <th>SCS2</th> <th>SCS[1]</th> <th>SCS[0]</th> <th>Sample Clock</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>CIR_CLK/64</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>CIR_CLK/128</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>CIR_CLK/256</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>CIR_CLK/512</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>CIR_CLK</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>Reserved</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>Reserved</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>Reserved</td> </tr> </tbody> </table>	SCS2	SCS[1]	SCS[0]	Sample Clock	0	0	0	CIR_CLK/64	0	0	1	CIR_CLK/128	0	1	0	CIR_CLK/256	0	1	1	CIR_CLK/512	1	0	0	CIR_CLK	1	0	1	Reserved	1	1	0	Reserved	1	1	1	Reserved
SCS2	SCS[1]	SCS[0]	Sample Clock																																				
0	0	0	CIR_CLK/64																																				
0	0	1	CIR_CLK/128																																				
0	1	0	CIR_CLK/256																																				
0	1	1	CIR_CLK/512																																				
1	0	0	CIR_CLK																																				
1	0	1	Reserved																																				
1	1	0	Reserved																																				
1	1	1	Reserved																																				

## 9.14 CIR Transmitter

### 9.14.1 Overview

The CIR transmitter (CIR\_TX) can transfer arbitrary waves which can be modulated with configurable carrier waves such as 38 kHz.

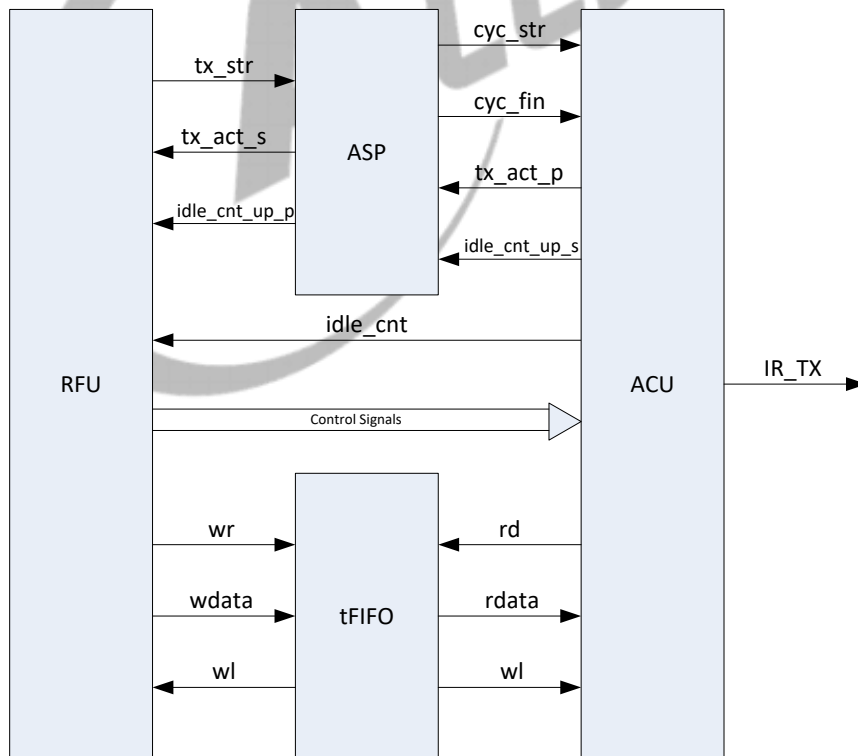
The CIR\_TX has the following features:

- Supports CIR remote control transmitter
- 128 bytes FIFO for data buffer
- Configurable carrier frequency
- Supports Interrupt and DMA
- Supports handshake mode and waiting mode of DMA

### 9.14.2 Block Diagram

The following figure shows a block diagram of the CIR\_TX.

Figure 9-116 CIR\_TX Block Diagram



### 9.14.3 Functional Description

#### 9.14.3.1 External Signals

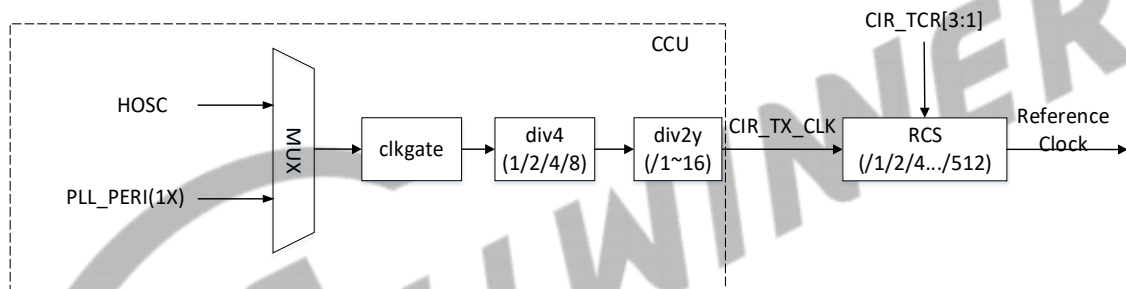
The following table describes the external signals of CIR\_TX.

**Table 9-41 CIR\_TX External Signals**

Signal	Description	Type
CIR-TX	Consumer infrared transmitter	O

#### 9.14.3.2 Clock Sources

**Figure 9-117 CIR\_TX Clock System**



#### 9.14.3.3 Function Implementation

The CIR\_TX is used to generate a waveform of arbitrary length, arbitrary shape, and no high-speed requirement, and it can change the data into the high-/low-level sequence of a certain length. Every transmitting data is in bytes, the Bit[7] of a byte means whether the level of a transmitting wave is high or low, the Bit[6:0] is the length of this wave. If the current transmitting frequency-division is 1, 0x88 is a high level of 8 cycles, 0x08 is a low level of 8 cycles. If the current transmitting frequency-division is 4, 0x88 is a high level of 32 cycles, 0x08 is a low level of 32 cycles.

The CIR\_TX has two transmission modes: non-cycle transmission, and cycle transmission.

The non-cycle transmission is to transmit all the data in TX\_FIFO until the FIFO is empty.

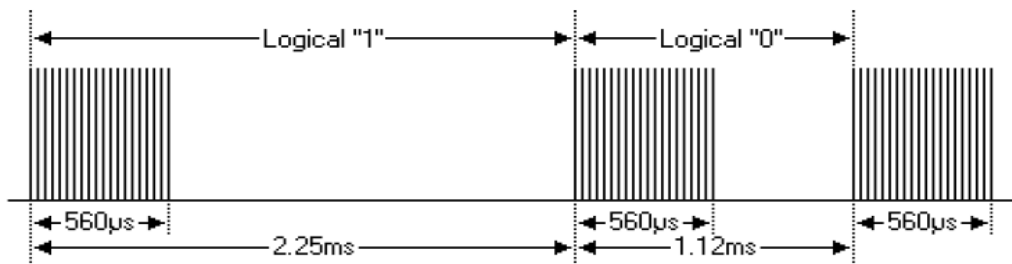
The cycle transmission is to transmit all the data in TX\_FIFO, after the transmission completion, wait for a certain time to recover the data in TX\_FIFO and then send it until a stop signal is detected. The data recovery in FIFO is implemented by clearing the read pointer.

### 9.14.3.4 Timing Diagram

The CIR remote control contains many protocols designed by different manufacturers. Here to NEC protocol as an example, the CIR-TX module uses a variable pulse-width modulation technique to encompass the various formats of infrared encoding for remote-control applications. A message is started by a 9 ms AGC burst, which is used to set the gain of the earlier CIR receivers. This AGC burst is then followed by a 4.5 ms space, which is then followed by the address and command.

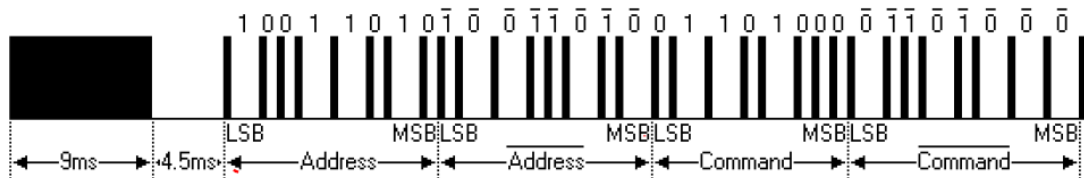
Bit definition: the logical "1" takes 2.25 ms to transmit, while a logical "0" is only 1.12 ms.

Figure 9-118 Definitions of Logical "1" and Logical "0"



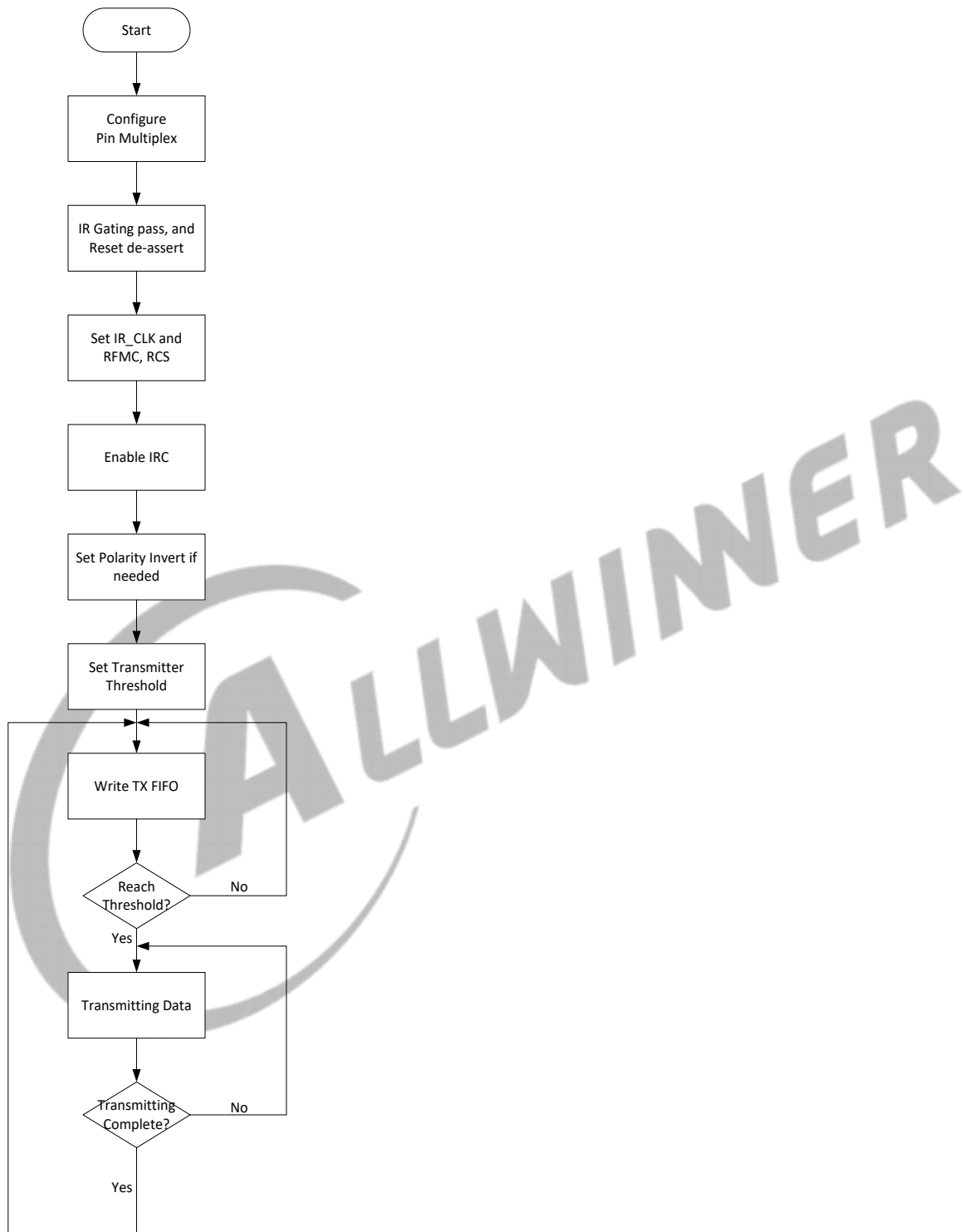
Timing for a message:

Figure 9-119 CIR Message Timing Diagram



### 9.14.4 Programming Guidelines

Figure 9-120 CIR Transmitter Process



### 9.14.5 Register List

Module Name	Base Address
CIR_TX	0x02003000

Register Name	Offset	Description
CIR_TGLR	0x0000	CIR Transmit Global Register
CIR_TMCR	0x0004	CIR Transmit Modulation Control Register
CIR_TCR	0x0008	CIR Transmit Control Register
CIR_IDC_H	0x000C	CIR Transmit Idle Duration Threshold High Bit Register
CIR_IDC_L	0x0010	CIR Transmit Idle Duration Threshold Low Bit Register
CIR_TICR_H	0x0014	CIR Transmit Idle Counter High Bit Register
CIR_TICR_L	0x0018	CIR Transmit Idle Counter Low Bit Register
CIR_TEL	0x0020	CIR TX FIFO Empty Level Register
CIR_TXINT	0x0024	CIR Transmit Interrupt Control Register
CIR_TAC	0x0028	CIR Transmit FIFO Available Counter Register
CIR_TXSTA	0x002C	CIR Transmit Status Register
CIR_TXT	0x0030	CIR Transmit Threshold Register
CIR_DMA	0x0034	CIR DMA Control Register
CIR_TXFIFO	0x0080	CIR Transmit FIFO Data Register

### 9.14.6 Register Description

#### 9.14.6.1 0x0000 CIR Transmitter Global Register (Default Value: 0x0000\_0000)

Offset: 0x0000			Register Name: CIR_TGLR
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	IMS Internal Modulation Select 0: The transmitting signal is not modulated 1: The transmitting signal is modulated internally



Offset: 0x0000			Register Name: CIR_TGLR
Bit	Read/Write	Default/Hex	Description
6:5	R/W	0x0	DRMC Duty ratio of modulated carrier is high level/low level. 00: Low level is equal to high level 01: Low level is the double of high level 10: Low level is the triple of high level 11: Reserved
4:3	/	/	/
2	R/W	0x0	TPPI Transmit Pulse Polarity Invert 0: Not invert transmit pulse 1: Invert transmit pulse
1	R/W	0x0	TR Transmit Reset When this bit is set, the transmitting is reset. The FIFO will be flushed, the TIC filed and the CSS field will be cleared during Transmit Reset. This field will automatically be cleared when the Transmit Reset is finished, and the CIR transmitter will state Idle.
0	R/W	0x0	TXEN Transmit Block Enable 0: Disable the CIR Transmitter 1: Enable the CIR Transmitter

**9.14.6.2 0x0004 CIR Transmitter Modulation Control Register (Default Value: 0x0000\_009E)**

Offset: 0x0004			Register Name: CIR_TMCR
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0x9E	RFMC Reference Frequency of modulated carrier. Reference Frequency of modulated carrier based on a division of a fixed functional clock (FCLK). The range of the modulated carrier is usually 30 kHz to 60 kHz. Most consumer electronics is 38 kHz.  The default modulated carrier is 38 kHz when FCLK is 12 MHz. RFMC= FCLK/((N+1)*(DRMC+2)).

9.14.6.3 0x0008 CIR Transmitter Control Register (Default Value: 0x0000\_0000)

Offset: 0x0008			Register Name: CIR_TCR
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	<p>CSS Cyclical Pulse Start/Stop Control</p> <p>0: Stop when cleared to '0'. From start to stop, all data in FIFO must be transmitted.</p> <p>1: Start. Start to transmit when it is set to '1'.</p>
6:4	/	/	/
3:1	R/W	0x0	<p>RCS Reference Clock Select for CIR Transmit</p> <p>The data in TX_FIFO is used to describe the pulse in Run-Length Code. The basic unit of pulse width is Reference Clock.</p> <p>000: CIR Transmit reference clock is ir_clk 001: CIR Transmit reference clock is ir_clk/2 010: CIR Transmit reference clock is ir_clk/4 011: CIR Transmit reference clock is ir_clk/8 100: CIR Transmit reference clock is ir_clk/64 101: CIR Transmit reference clock is ir_clk/128 110: CIR Transmit reference clock is ir_clk/256 111: CIR Transmit reference clock is ir_clk/512</p>
0	R/W	0x0	<p>TTS Type of the transmission signal</p> <p>0: The transmitting wave is a single non-cyclical pulse. 1: The transmitting wave is a cyclical short-pulse.</p>

9.14.6.4 0x000C CIR Transmitter Idle Duration Counter High Bit Register (Default Value: 0x0000\_0000)

Offset: 0x000C			Register Name: CIR_IDC_H
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/

Offset: 0x000C			Register Name: CIR_IDC_H
Bit	Read/Write	Default/Hex	Description
3:0	R/W	0x0	<p>IDC_H</p> <p>Idle Duration Counter Threshold (High 4 bits)</p> <p>Idle Duration = 128*IDC*Ts (IDC = 0~4095)</p> <p>It is used in cyclical transmission mode. When all the data in FIFO is transmitted, the signals can be transmitted after a specific time.</p>

**9.14.6.5 0x0010 CIR Transmitter Idle Duration Counter Low Bit Register (Default Value: 0x0000\_0000)**

Offset: 0x0010			Register Name: CIR_IDC_L
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
3:0	R/W	0x0	<p>IDC_L</p> <p>Idle Duration Counter Threshold (Low 8 bits)</p> <p>Idle Duration = 128*IDC*Ts (IDC = 0~4095)</p> <p>It is used in cyclical transmission mode. When all the data in FIFO is transmitted, the signals can be transmitted after a specific time.</p>

**9.14.6.6 0x0014 CIR Transmitter Idle Counter High Bit Register (Default Value: 0x0000\_0000)**

Offset: 0x0014			Register Name: CIR_TICR_H
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R	0x0	<p>TIC_H</p> <p>Transmit Idle Counter_H (High 8 bits)</p> <p>It is used to count the idle duration of CIR transmitter by software.</p> <p>Count in 128*Ts (Sample Duration, 1/Fs) when the transmitter is idle, and it should be reset when the transmitter is active.</p> <p>When this counter reaches the maximum value (0xFFFF), it will stop automatically, and should not be cleared to zero.</p>

9.14.6.7 0x0018 CIR Transmitter Idle Counter Low Bit Register (Default Value: 0x0000\_0000)

Offset: 0x0018			Register Name: CIR_TICR_L
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R	0x0	<p>TIC_L Transmit Idle Counter_L (Low 8 bits)</p> <p>It is used to count the idle duration of CIR transmitter by software.</p> <p>Count in 128*Ts (Sample Duration, 1/Fs) when the transmitter is idle, and it should be reset when the transmitter is active.</p> <p>When this counter reaches the maximum value (0xFFFF), it will stop automatically, and should not be cleared to zero.</p>

9.14.6.8 0x0020 CIR Transmitter FIFO Empty Register (Default Value: 0x0000\_0000)

Offset: 0x0020			Register Name: CIR_TEL
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0x0	<p>TEL TX FIFO empty Level for DRQ and IRQ. TRIGGER_LEVEL = TEL + 1</p>

9.14.6.9 0x0024 CIR Transmitter Interrupt Control Register (Default Value: 0x0000\_0000)

Offset: 0x0024			Register Name: CIR_TXINT
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/
2	R/W	0x0	<p>DRQ_EN TX FIFO DMA Enable</p> <p>0: Disable 1: Enable</p> <p>When it is set to '1', the TX FIFO DRQ is asserted if the number of the transmitting data in the FIFO is less than the RAL. The DRQ is de-asserted when the condition fails.</p>

Offset: 0x0024			Register Name: CIR_TXINT
Bit	Read/Write	Default/Hex	Description
1	R/W	0x0	TAI_EN TX FIFO Available Interrupt Enable 0: Disable 1: Enable
0	R/W	0x0	TPEI_EN Transmit Packet End Interrupt Enable for Cyclical Pulse 0: Disable 1: Enable  TUI_EN Transmitter FIFO Underrun Interrupt Enable for Non-cyclical Pulse 0: Disable 1: Enable

9.14.6.10 0x0028 CIR Transmitter FIFO Available Counter Register (Default Value: 0x0000\_0080)

Offset: 0x0028			Register Name: CIR_TAC
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R	0x80	TAC TX FIFO Available Space Counter 0x00: No available space in TX FIFO 0x01: 1 byte available space in TX FIFO 0x02: 2 bytes available space in TX FIFO ... 0x80: 128 bytes available space in TX FIFO

9.14.6.11 0x002C CIR Transmitter Status Register (Default Value: 0x0000\_0002)

Offset: 0x002C			Register Name: CIR_TXSTA
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/

Offset: 0x002C			Register Name: CIR_TXSTA
Bit	Read/Write	Default/Hex	Description
3	R	0x0	<p>STCT Status of CIR Transmitter</p> <p>0: Idle 1: Active</p> <p>This bit will automatically set when the controller begins to transmit the data in the FIFO. The “1” will last when the data in the FIFO. It will automatically be cleared to “0” when all data in the FIFO is transmitted.</p> <p>The bit is for debugging.</p> <p>The output Level of Idle state is determined by the level of the last data output.</p>
2	R	0x0	<p>DRQ DMA Request Flag</p> <p>When set to ‘1’, the TX FIFO DRQ is asserted if the number of the transmission data in the FIFO is less than the RAL. The DRQ is de-asserted when the condition fails.</p> <p>This bit is for debugging.</p>
1	R/W	0x1	<p>TAI TX FIFO Available Interrupt Flag</p> <p>0: TX FIFO not available by its level 1: TX FIFO available by its level</p> <p>Writing 1 clears this bit.</p>
0	R/W	0x0	<p>TPE Transmitter Packet End Flag for Cyclical Pulse</p> <p>0: Transmissions of address, control and data fields not completed 1: Transmissions of address, control and data fields completed</p> <p>TUR Transmitter FIFO Underrun Flag for Non-cyclical Pulse</p> <p>0: No transmitter FIFO underrun 1: Transmitter FIFO underrun</p> <p>Writing 1 clears this bit.</p>

9.14.6.12 0x0030 CIR Transmitter Threshold Register (Default Value: 0x0000\_0000)

Offset: 0x0030			Register Name: CIR_TXT
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0x0	<p>NCTT</p> <p>Non-cyclical Pulse Transmit Threshold</p> <p>The controller will trigger transmitting the data in the FIFO when the data byte number has reached the Transmit Threshold set in this field.</p>

9.14.6.13 0x0034 CIR Transmitter DMA Control Register (Default Value: 0x0000\_00A5)

Offset: 0x0034			Register Name: CIR_DMA_CTL
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0xA5	<p>DMA Handshake Configuration</p> <p>0xA5: DMA waiting cycle mode</p> <p>0xEA: DMA handshake mode</p>

9.14.6.14 0x0080 CIR Transmitter FIFO Data Register (Default Value: 0x0000\_0000)

Offset: 0x0080			Register Name: CIR_TXFIFO
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	W	0x0	<p>TBF</p> <p>Transmit Byte FIFO</p> <p>When the transmission is triggered, the data in the FIFO will be transmitted until the data number is transmitted completely.</p>

# Contents

10	Security System.....	1280
10.1	Crypto Engine.....	1280
10.1.1	Overview.....	1280
10.1.2	Block Diagram.....	1281
10.1.3	Functional Descriptions.....	1281
10.1.4	Programming Guidelines.....	1301
10.1.5	Register List.....	1304
10.1.6	Register Description.....	1305
10.2	Security ID.....	1309





# Figures

Figure 10-1 CE Block Diagram .....	1281
Figure 10-2 DES Encryption and Decryption .....	1282
Figure 10-3 3DES Encryption and Decryption of a 3-key Operation and a 2-key Operation .....	1283
Figure 10-4 ECB Mode Encryption and Decryption.....	1284
Figure 10-5 CBC Mode Encryption and Decryption .....	1285
Figure 10-6 CTR Mode Encryption and Decryption.....	1286
Figure 10-7 CFB Mode Encryption and Decryption.....	1287
Figure 10-8 OFB Mode Encryption and Decryption .....	1288
Figure 10-9 CTS Mode Encryption and Decryption.....	1289
Figure 10-10 Structure of Task Descriptor Chaining .....	1290
Figure 10-11 Word Address of Message .....	1297
Figure 10-12 Byte Order.....	1297
Figure 10-13 Bit Order .....	1297
Figure 10-14 The Storage Method of 32-bit IV .....	1298
Figure 10-15 The Storage Method of 64-bit IV .....	1298
Figure 10-16 Task Request Process.....	1301

## 10 Security System

### 10.1 Crypto Engine

#### 10.1.1 Overview

The Crypto Engine (CE) module is one encryption/decryption algorithm accelerator. It supports kinds of symmetric, asymmetric, Hash, and RNG algorithms.

The symmetric algorithm supports data encryption and decryption by following the data encryption standard (DES), 3DES, or advanced encryption standard (AES) algorithms. It can encrypt or decrypt a large amount of data effectively.

The Rivest-Shamir-Adleman (RSA) asymmetric algorithm is used for data encryption/decryption and digital signature verification. It is a public key encryption/decryption algorithm implemented through the modular exponentiation operation.

The Hash algorithm supports data integrity authentication and digital signature. The hash supports MD5, SHA1, SHA224, SHA256, SHA384, SHA512, HMAC-SHA1, and HMAC-SHA256 algorithms.

The RNG algorithm can generate true-random numbers and pseudo-random numbers.

The software interface of the CE is simple, only setting interrupt control, task description address, and load tag. The algorithm control information is written in memory by task descriptor, then the CE automatically reads it when executing a request. It supports parallel execution of 4 channels and has an internal DMA controller to transfer data between CE and memory.

The CE has the following features:

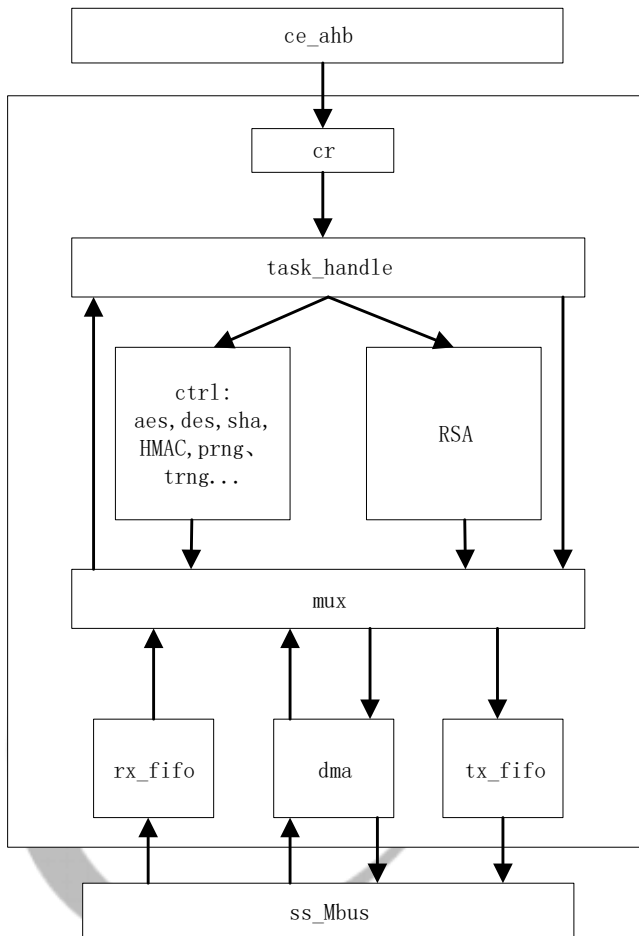
- Symmetrical algorithm: AES, DES, 3DES
- Hash algorithm: MD5, SHA1, SHA224, SHA256, SHA384, SHA512, HMAC-SHA1, HMAC-SHA256
- Asymmetrical algorithm: RSA512/1024/2048-bit
- 160-bit hardware PRNG with 175-bit seed
- 256-bit hardware TRNG
- Electronic codebook (ECB), cipher block chaining (CBC), counter (CTR), cipher text stealing (CTS), 128-output feedback (OFB), 1-/8-/64-/128-cipher feedback (CFB) modes for AES algorithm
- ECB, CBC, CTR modes for DES/3DES algorithm
- 128-, 192-, 256-bit key size for AES algorithms
- 16-, 32-, 64-, 128-bit wide size for AES CTR mode
- 16-, 32-, 64-bit wide size for DES/3DES CTR mode

- One or more blocks mode for MD5/SHA1/SHA224/SHA256/SHA384/SHA512
- Internal DMA controller for data transfer with memory

### 10.1.2 Block Diagram

The following figure shows the block diagram of Crypto Engine.

Figure 10-1 CE Block Diagram

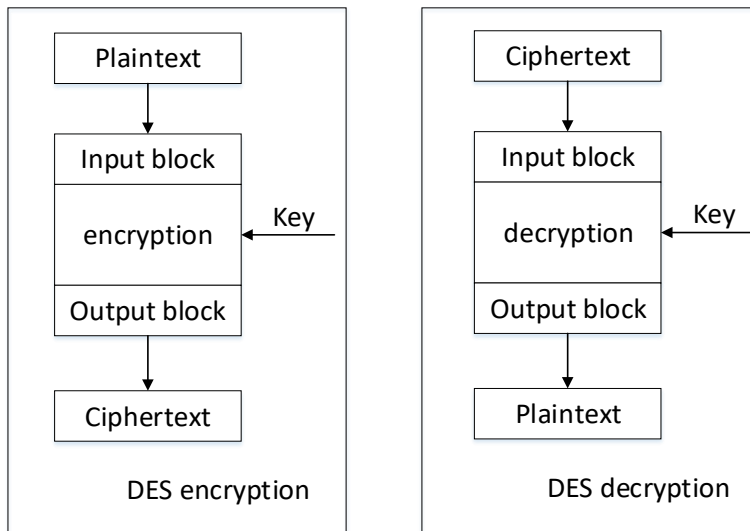


### 10.1.3 Functional Description

#### 10.1.3.1 DES Algorithm

Figure 10-2 shows the DES encryption and decryption operation.

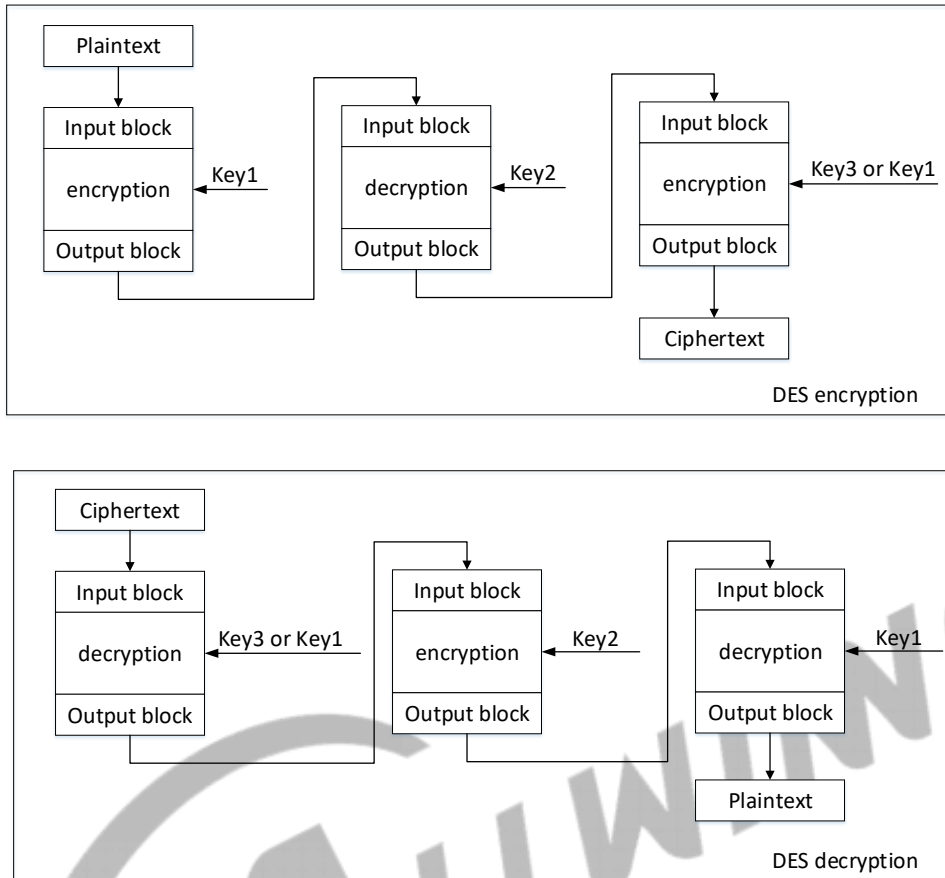
Figure 10-2 DES Encryption and Decryption



### 10.1.3.2 3DES Algorithm

The 3DES algorithm supports both 3-key and 2-key operations. A 2-key operation can be regarded as a simplified 3-key operation. To be specific, key 3 is represented by key 1 in a 2-key operation. Figure 10-3 shows the 3DES encryption and decryption operation of a 3-key operation and a 2-key operation.

Figure 10-3 3DES Encryption and Decryption of a 3-key Operation and a 2-key Operation

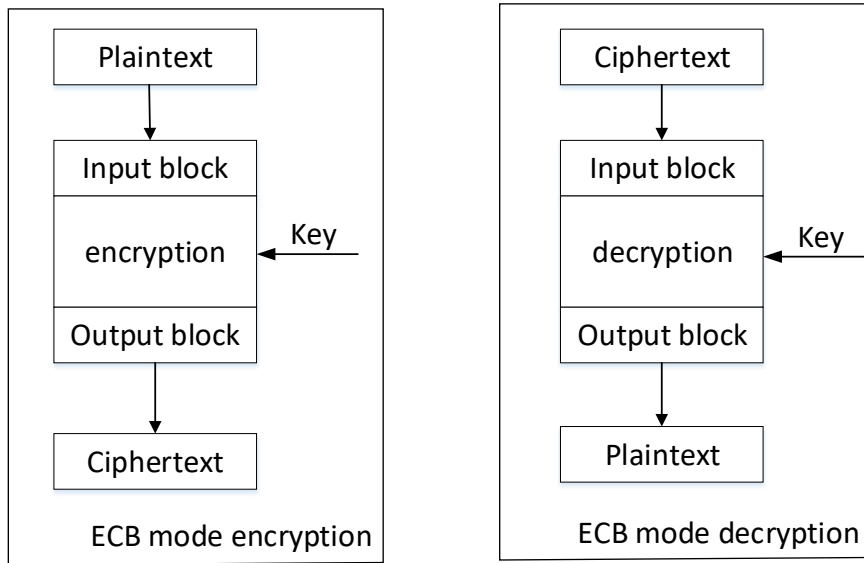


### 10.1.3.3 ECB Mode

The ECB mode is a confidentiality mode that features, for a given key, the assignment of a fixed ciphertext block to each plaintext block, analogous to the assignment of code words in a codebook.

In ECB mode, encryption and decryption algorithms are directly applied to the block data. The operation of each block is independent, so the plaintext encryption and ciphertext decryption can be performed concurrently.

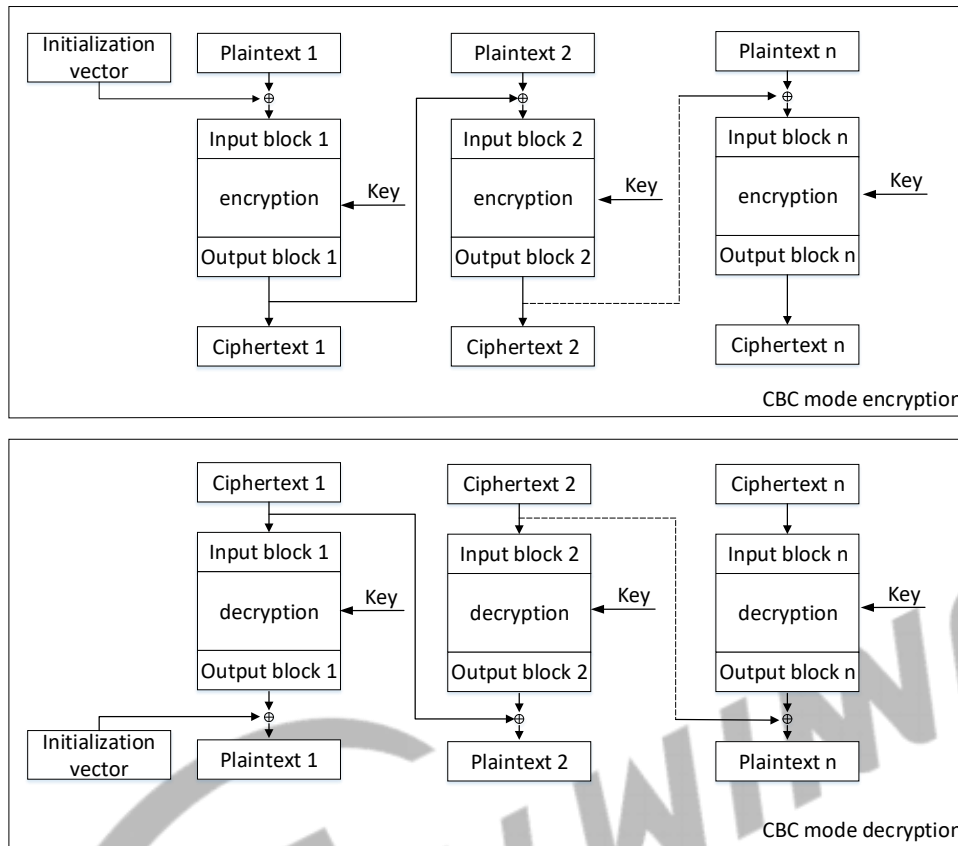
Figure 10-4 ECB Mode Encryption and Decryption



#### 10.1.3.4 CBC Mode

The CBC mode is a confidentiality mode whose encryption process features the combining of the plaintext blocks with the previous ciphertext blocks. The CBC mode requires an initialization vector (IV) to combine with the first plaintext block. The encryption process of each plaintext block is related to the block processing result of the previous ciphertext blocks, so encryption operations cannot be concurrently performed in CBC mode. The decryption operation is independent of output plain text of the previous block, so decryption operations can be performed concurrently.

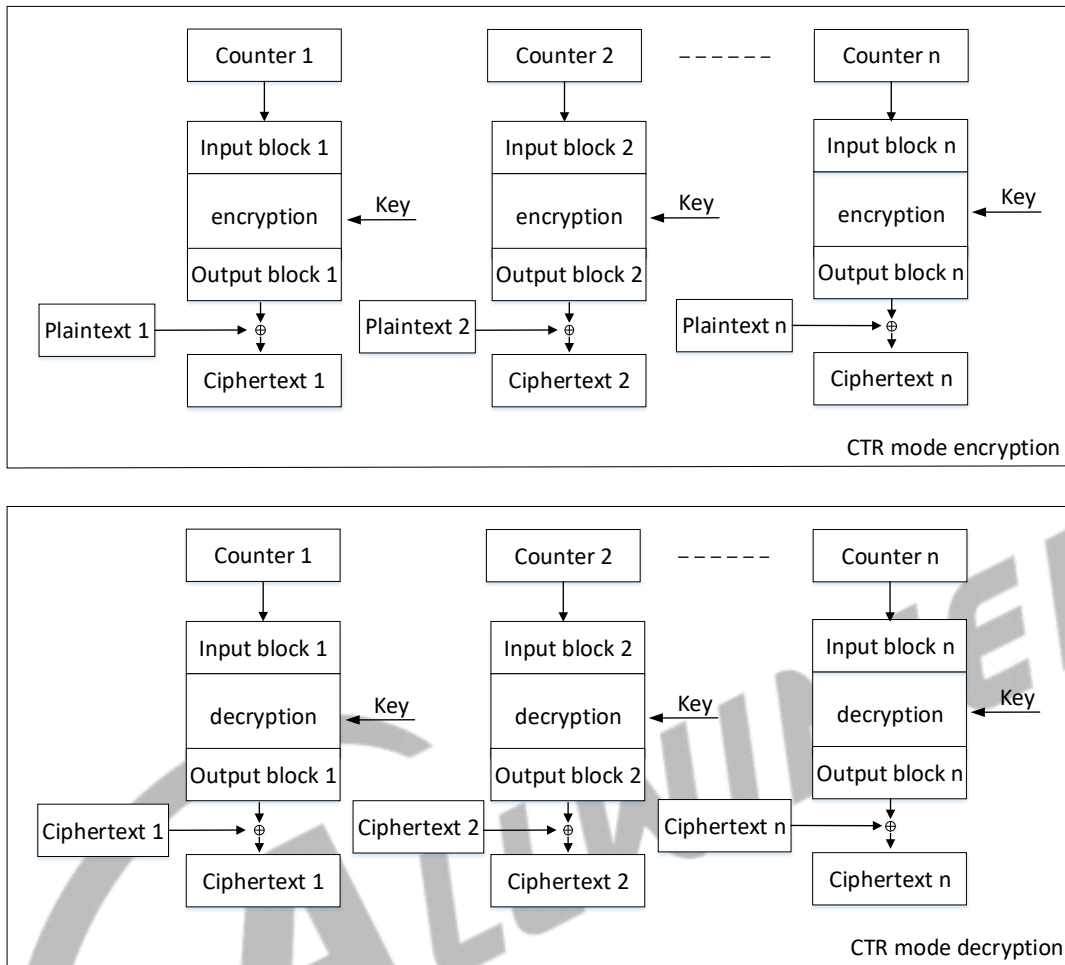
Figure 10-5 CBC Mode Encryption and Decryption



### 10.1.3.5 CTR Mode

The CTR mode is a confidentiality mode that features the application of the forward cipher to a set of input blocks, called counters, to produce a sequence of output blocks that are exclusive-ORed with the plaintext to produce the ciphertext, and vice versa. All of the counters must be distinct.

Figure 10-6 CTR Mode Encryption and Decryption



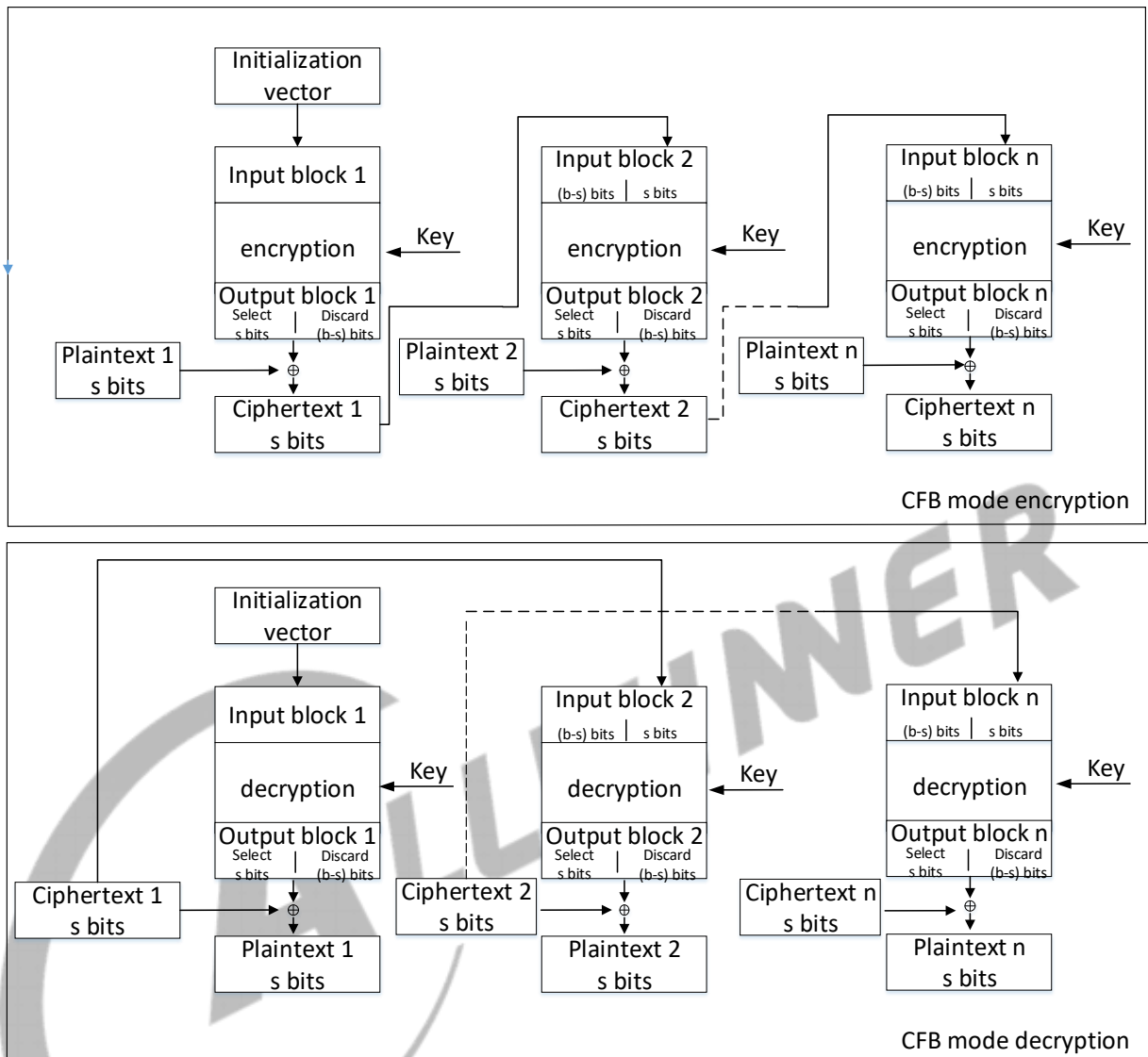
10.1.3.6 CFB Mode

The CFB mode is a confidentiality mode that features the feedback of successive ciphertext segments into the input blocks of the forward cipher to generate output blocks that are exclusive-ORed with the plaintext to produce the ciphertext, and vice versa. The CFB mode requires an IV as the initial input block, and the forward cipher operation is applied to the IV to produce the first output block. The first ciphertext segment is produced by exclusive-ORing the first plaintext segment with the *s* most significant bits of the first output block. The value of *s* is 1 bit, 8 bits, 64 bits, or 128 bits.

The following figure shows the *s*-bit CFB mode of the AES algorithms.



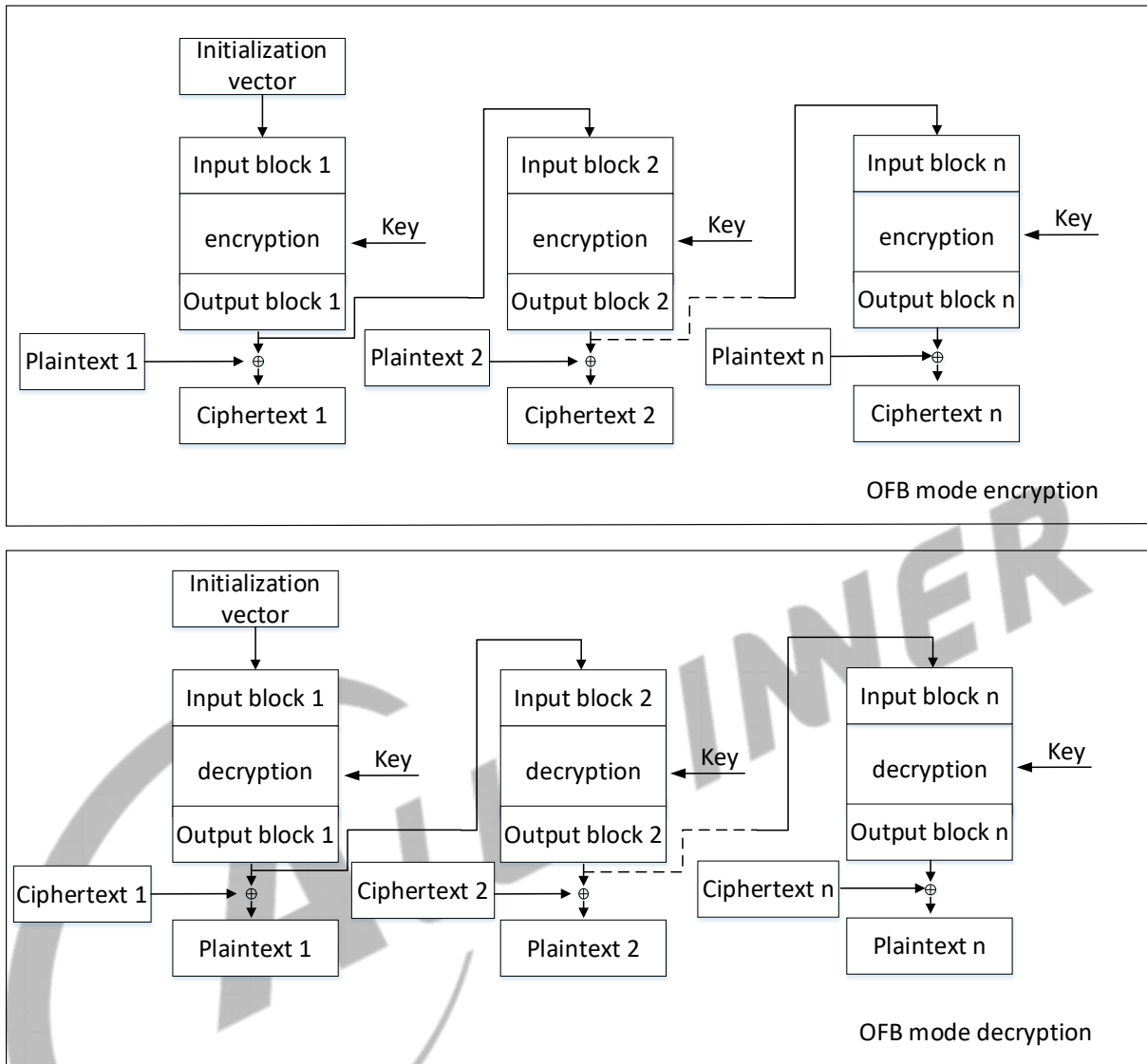
Figure 10-7 CFB Mode Encryption and Decryption



10.1.3.7 OFB Mode

The OFB mode is a confidentiality mode that features the iteration of the forward cipher on an IV to generate a sequence of output blocks that are exclusive-ORed with the plaintext to produce the ciphertext, and vice versa. If a same key is used, different IVs must be used to ensure operation security.

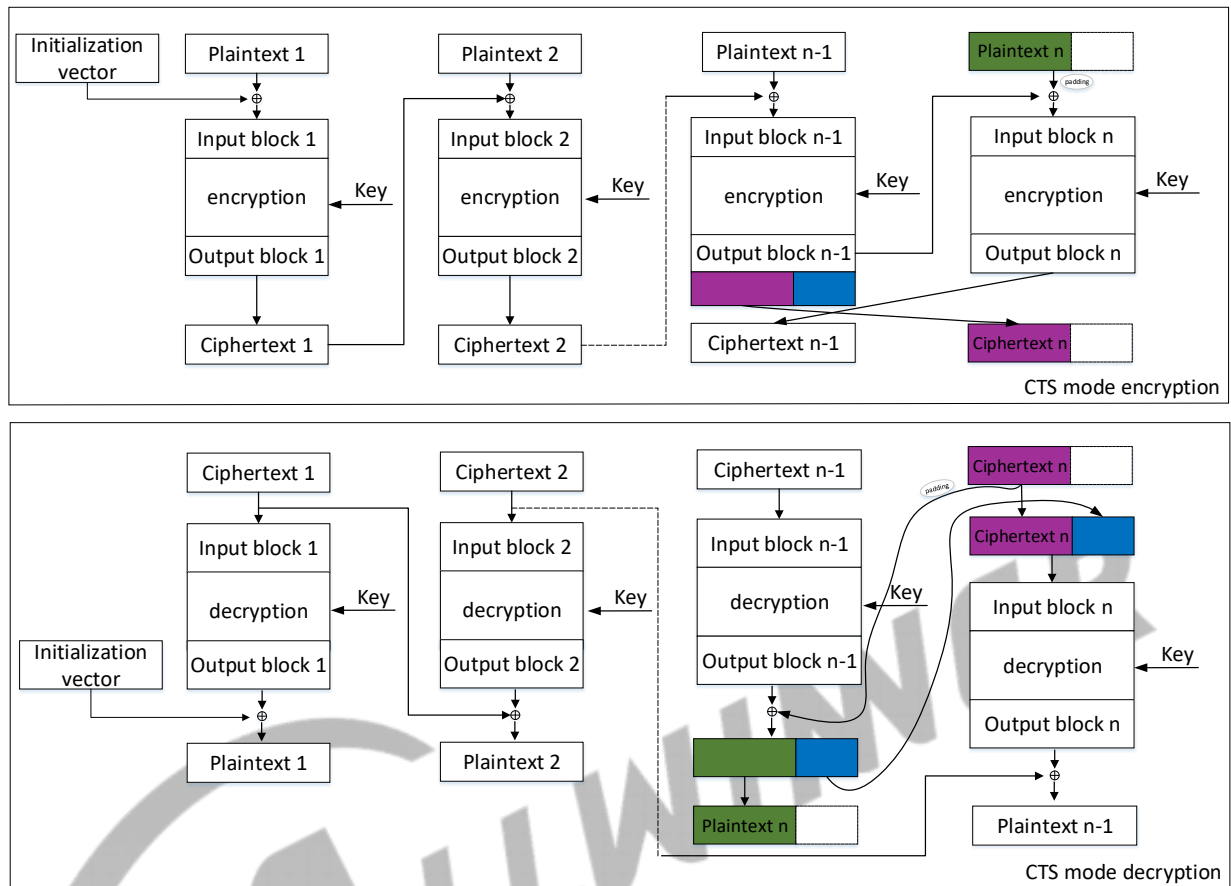
Figure 10-8 OFB Mode Encryption and Decryption



### 10.1.3.8 CTS Mode

The CTS mode is a confidentiality mode that accepts any plaintext input whose bit length is greater than or equal to the block size but not necessarily a multiple of the block size. Below are the diagrams for CTS encryption and decryption.

Figure 10-9 CTS Mode Encryption and Decryption



### 10.1.3.9 HASH Algorithm

The hash algorithms support MD5, SHA1, SHA224, SHA256, SHA384, SHA512, HMAC-SHA1, and HMAC-SHA256. All algorithms are iterative, one-way hash functions that can process a message to produce a condensed representation called a *message digest*. When a message is received, the *message digest* can be used to verify whether the data has changed, that is, to verify its integrity.

The hash algorithm of the CE supports block-aligned total length of the input data (padded by software), that is, a multiple of 64 bytes. The message length after padding by software is used as the configured data length for the hash algorithm.

### 10.1.3.10 RSA Algorithm

The RSA is a public key encryption/decryption algorithm implemented through the modular exponentiation operation.

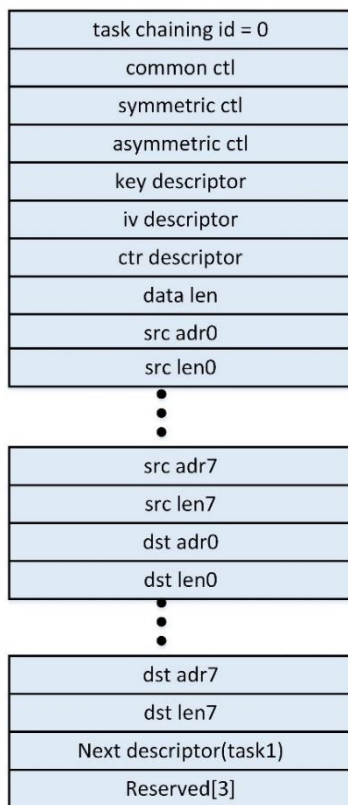
The ciphertext is obtained as follows:  $C = M^E \text{ mod } N$ . The plaintext is obtained as follows:  $M = C^D \text{ mod } N$ .

M indicates the plaintext, C indicates the ciphertext, (N, E) indicates the public key, and (N, D) indicates the private key.

### 10.1.3.11 Task Descriptor

The software makes request through task descriptor, including algorithm type, algorithm mode, key address, source/destination address and size, and so on. The structure of the task descriptor is as follows.

**Figure 10-10 Structure of Task Descriptor Chaining**



The bit definitions of the task descriptor are as follows.

#### Task ID

Bit	Read/Write	Default/Hex	Description
31:4	/	/	/

Bit	Read/Write	Default/Hex	Description
3:0	R/W	0x0	<p>CHN</p> <p>Task channel ID</p> <p>Indicates which channel the task is running on.</p> <p>It supports 0 to 3.</p>

**Common Control**

Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	<p>Interrupt enable (IE) for the current task</p> <p>0: disable interrupt</p> <p>1: enable interrupt</p> <p>Represents whether an interrupt signal is generated when the task chain ends at the end of this task.</p> <p>When the last task in a task chain ends, the operation of the task chain will end normally; if a task fails in the middle, the task chain will be aborted abnormally. And it is determined whether to generate an interrupt signal according to the IE configuration of the current task when the current task ends or aborts. Therefore, if you want to use interrupts, it is recommended that not only the IE of the last task of each task chain is configured to 1 to generate the end interrupt of the task chain, but also the IEs of other tasks in this task chain are also configured to 1. The purpose is to generate an interrupt signal once an abnormal error occurs in these tasks and the interrupt is aborted.</p>
30:17	/	/	/
16	R/W	0x0	<p>IV mode</p> <p>IV mode for SHA1/SHA224/SHA256/SHA384/SHA512/MD5 or constants</p> <p>0: use initial constants defined in FIPS-180</p> <p>1: use input iv</p>
15	R/W	0x0	<p>Last HMAC plaintext</p> <p>0: not the last HMAC plaintext package</p> <p>1: the last HMAC plaintext package</p>
14:9	/	/	/

Bit	Read/Write	Default/Hex	Description
8	R/W	0x0	OP DIR Algorithm Operation Direction 0: Encryption 1: Decryption Configure according to the requirements of encryption or decryption.
7	/	/	/
6:0	R/W	0x0	Algorithm Type 0x0: AES 0x1: DES 0x2: Triple DES (3DES) 0x10: MD5 0x11: SHA-1 0x12: SHA-224 0x13: SHA-256 0x14: SHA-384 0x15: SHA-512 0x16: HMAC-SHA1 0x17: HMAC-SHA256 0x20: RSA 0x30: TRNG 0x31: PRNG Others: reserved

**Symmetric Control**

Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:20	R/W	0x0	KEY_SELECT key select for AES 0000: Select input CE_KEYx (Normal Mode) 0001: Select {SSK} 0010: Select {HUK} 0011: Select {RSSK} 0100-0111: Reserved 1000-1111: Select internal Key n (n from 0 to 7)

Bit	Read/Write	Default/Hex	Description
19:18	R/W	0x0	CFB_WIDTH AES-CFB width 00: CFB1 01: CFB8 10: CFB64 11: CFB128
17	R/W	0x0	PRNG_LD Load new 15 bits key into linear feedback shift register (LFSR) for PRNG. When the algorithm type is PRNG, it is necessary to post-process the random number output by PRNG through the linear shift operation to generate the operand. When the PRNG_LD is configured to 1, use iv_addr[14:0] as the input number for linear shift operation, and do XOR operation between the data and the random number output by PRNG to generate the post-processing result of further operation.
16	R/W	0x0	CTS_LPKG AES CTS last package flag When set to '1', it means this is the last package for AES-CTS mode (the size of the last package is larger than 128 bits).
15:12	/	/	/
11:8	R/W	0x0	ALGORITHM_MODE CE algorithm mode 0000: Electronic Code Book (ECB) mode 0001: Cipher Block Chaining (CBC) mode 0010: Counter (CTR) mode 0011: Cipher Text Stealing (CTS) mode 0100: Output Feedback (OFB) mode 0101: Cipher Feedback (CFB) mode Other: reserved
7:4	/	/	/
3:2	R/W	0x0	CTR WIDTH Counter width for CTR mode 00: 16-bit Counter 01: 32-bit Counter 10: 64-bit Counter 11: 128-bit Counter

Bit	Read/Write	Default/Hex	Description
1:0	R/W	0x0	AES KEY SIZE 00: 128-bit 01: 192-bit 10: 256-bit 11: Reserved

**Asymmetric Control**

Bit	Read/Write	Default/Hex	Description
31	/	/	/
30:28	R/W	0x0	RSA Public Modulus Width 000: 512-bit 001: 1024-bit 010: 2048-bit Other: reserved
27:19	/	/	/
18:16	R/W	0x0	RSA MODE RSA algorithm mode. For modular computation: 000: modular exponent(RSA) 001: modular div 010: modular mul 011: modular inv others: reserved
15:0	/	/	/

**Key Descriptor**

Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	Key Address The address of KEY that needs to be stored.



### IV Descriptor

Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	IV Address The address of IV that needs to be stored.

### Counter Descriptor

Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	CTR Data Output Address The address of CTR data output that needs to be stored.

### Data Length

Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	Data Length Configure the data length of the corresponding segment. The data length size needs to be consistent with dst_data_length (destination data length 0 +... + destination data length 7).  The data length field in the task descriptor has different meanings for different algorithms.  For AES-CTS, the data length field indicates byte numbers of source data, for others indicate word numbers of source data.  For PRNG, the data length should be 5 words aligned.  For TRNG, it should be 8 words aligned.

### Source Address 0~7

Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	Source Data Address The address of the source data that needs to be stored.

**Source Data Length 0~7**

Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	Source Data Length The length of the source data. Unit: byte

**Destination Address 0~7**

Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	Destination Data Address The address of the destination data that needs to be stored.

**Destination Data Length 0~7**

Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	Destination Data Length The length of the destination data. Unit: byte

**Next Descriptor Address**

Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	Next Task Address The address where the descriptor of the next task in a task-chain is saved. If there is the only task or the last task of a task-chain, the next task address must be 32'h0.

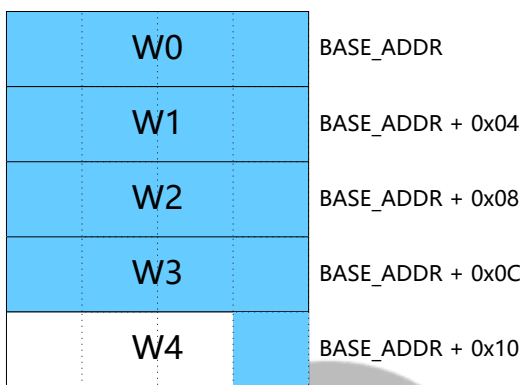
**10.1.3.12 Storing Message**

In the application, a message may not be stored contiguously in the memory, but divided into multiple segments. Or a piece of continuously stored messages can be artificially split into multiple pieces as needs.

Then each segment corresponds to a set of the source address and source length in the descriptor. Multiple segments correspond to groups 0-7 source address/source length in sequence.

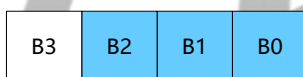
Each task supports up to 8 message segments, and the data volume of each message segment supports up to 4 GWord (AES-CTS is 1 GByte). The total amount of all segments in a task (that is a package) supports up to 4 GWord (AES-CTS is 1 GByte). If a message is divided into multiple packages, all others are required to be whole words; when the last package of AES-CTS is less than one word, 0 needs to be padded, and those less than one word are counted as one word. The following figure shows the address order structure.

**Figure 10-11 Word Address of Message**



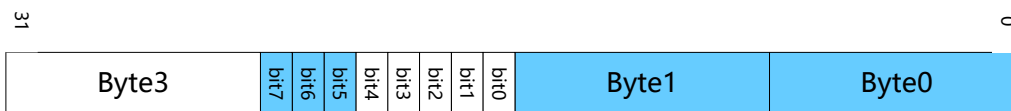
Byte order: low byte first, high byte last. When the data is less than one word, the low byte is filled first. The following figure shows the byte order structure (blue means it is filled by the message).

**Figure 10-12 Byte Order**



Bit order: high bit first, low bit last. When the data is less than one Byte, the high bit is filled first. The following figure shows the bit order structure.

**Figure 10-13 Bit Order**



**10.1.3.13 Storing Key**

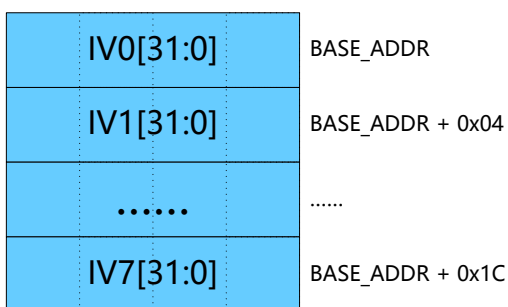
The length of KEY must be an integer multiple of word, refer to section 10.1.3.15 “[Algorithm Length Properties](#)”.

### 10.1.3.14 Storing IV

For different algorithms, the length of IV is different. But they are integer multiples of word. To keep the byte order of IV and HASH digest output consistent, the byte order of IV is different from that of the message. For the multi-packet operation, the first address of the digest output result of the previous HASH can be directly configured to the first address of the next IV, and the software does not need to do any processing on the digest.

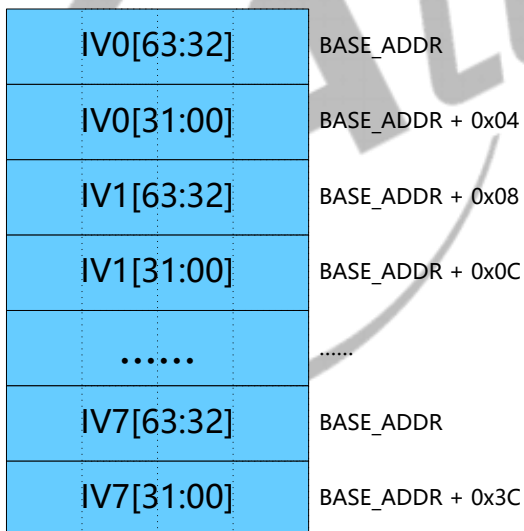
The following figure shows the storage method of 32-bit IV value.

**Figure 10-14 The Storage Method of 32-bit IV**



The following figure shows the storage method of 64-bit IV value.

**Figure 10-15 The Storage Method of 64-bit IV**



### 10.1.3.15 Algorithm Length Properties

The algorithm length has different requirements for different algorithms.

**Table 10-1 Symmetric Algorithm Configuration Properties**

Algorithm	Length Setting				Alignment	Software Padding
	Source Size	Destination Size	KEY	IV		
AES (except CTS)	< 4 GWord	< 4 GWord	AES-128: 4 Word AES-192: 6 Word AES 256: 8 Word	4 Word	Word-aligned	need
AES-CTS	< 1 GByte	< 1 GByte	AES-128: 4 word AES-192: 6 word AES 256: 8 word	4 Word	Word-aligned	need
DES	< 4 GWord	< 4 GWord	2 Word	2 Word	Word-aligned	need
TDES	< 4 GWord	< 4 GWord	6 Word	2 Word	Word-aligned	need

**Table 10-2 Hash Algorithm Configuration Properties**

Algorithm	Length Setting				Alignment	Software Padding
	Source Size	Destination Size	KEY	IV		
MD5	< 4 GWord	4 Word	Fixed to 0	4 Word	Word-aligned	need
SHA-1	< 4 GWord	5 Word	Fixed to 0	5 Word	Word-aligned	need
SHA-224	< 4 GWord	8 Word	Fixed to 0	8 Word	Word-aligned	need
SHA-256	< 4 GWord	8 Word	Fixed to 0	8 Word	Word-aligned	need
SHA-384	< 4 GWord	16 Word	Fixed to 0	16 Word	Word-aligned	need
SHA-512	< 4 GWord	16 Word	Fixed to 0	16 Word	Word-aligned	need
HMAC-SHA1	< 4 GWord	5 Word	16 Word	5 Word	Word-aligned	need
HMAC-SHA256	< 4 GWord	8 Word	16 Word	8 Word	Word-aligned	need

**Table 10-3 RNG Configuration Properties**

Algorithm	Length Setting				Alignment	Software Padding
	Source Size	Destination Size	KEY	IV		
TRNG	< 4 GWord	< 4 GWord	Fixed to 0	4 Word	Word-aligned	need
PRNG	< 4 GWord	< 4 GWord	6 Word	4 Word	Word-aligned	need

**Table 10-4 Asymmetric Algorithm Configuration Properties**

Algorithm	Length Setting				Alignment	Software Padding
	Source Size	Destination Size	KEY	IV		
RSA512	16 Word	16 Word	16 Word	Not use IV	Word-aligned	need
RSA1024	32 Word	32 Word	32 Word	Not use IV	Word-aligned	need
RSA2048	64 Word	64 Word	64 Word	Not use IV	Word-aligned	need

### 10.1.3.16 Error Detection

The CE module includes error detection for task configuration, data computing error, and authentication invalid. When the algorithm type in task descriptor is read into the CE module, the CE will check whether this type is supported through checking algorithm type field in common control. If the type value is out of scope, the CE will issue interrupt signal and set error state. Each type has certain input and output data size. After getting a task descriptor, the input size and output size configuration will be checked to avoid size error. If the size configuration is wrong, the CE will issue interrupt signal and set error state.

### 10.1.3.17 Clock Requirement

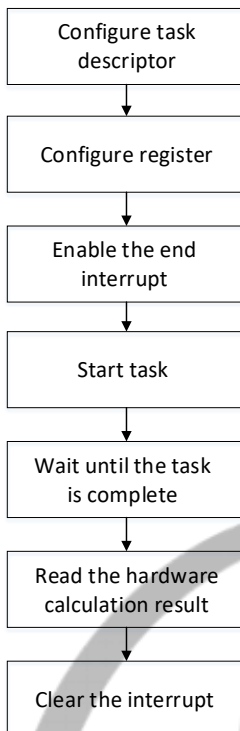
Clock Name	Description	Requirement
hclk	AHB bus clock	24 MHz – 200 MHz
mclk	MBUS clock	24 MHz – 400 MHz
ce_clk	CE work clock	24 MHz – 400 MHz

## 10.1.4 Programming Guidelines

### 10.1.4.1 Symmetrical/Asymmetrical/Hash/RNG Algorithm Operation Process

The following figure shows the process of an algorithm operation.

**Figure 10-16 Task Request Process**



- Step 1** The software should configure a task descriptor in memory, including the related fields in the descriptor. The channel id corresponds to one channel in CE. According to algorithm type, the software should set the fields in common control, symmetric control, asymmetric control, then provide key/iv/ctr address and the data length of this task. The source and destination address and size are set based on the upper application. If there is another task concatenating after this task, then set its descriptor address at the next descriptor field. For more details for task descriptor, see section 10.1.4.2, section 10.1.4.3 and section 10.1.4.4.
- Step 2** The software should set registers. Configure the first address of the task descriptor structure to [CE Task Descriptor Address Register](#). Configure the source/destination address to [CE Current Source Address Register/CE Current Destination Address Register](#).
- Step 3** Enable the end interrupt of the corresponding task channel by setting [CE Interrupt Control Register](#).
- Step 4** The software reads [CE Task Load Register](#) to ensure that the bit0 is 0. If the bit0 is not read out to be 0, wait until it is 0, then configure the bit0 to be 1 to start task.
- Step 5** Wait for interrupt status by reading [CE Interrupt Status Register](#).
- Step 6** Read the result from the destination address.

**Step 7** Clear the interrupt.

#### 10.1.4.2 Configuring Task Descriptor of AES

- **Common control:** Configure [Common Control](#)[6:0] to 0x0 to select AES algorithm type.
- **Symmetric control:** According to the corresponding algorithm requirements, configure [Symmetric Control](#) to select the key size, CTR width, CTS last package flag, CFB width, and AES algorithm mode, and so on.
- **Asymmetric control:** The symmetric algorithm does not need to be configured for this field.
- **Key descriptor:** Because the storage of the key requires word alignment, ensure that this descriptor is the first address of the KEY (word address).
- **IV descriptor:** In the task that requires the IV value, configure the first address of the storage space where the IV is stored here. Because the storage of the IV requires word alignment, ensure that this descriptor is the first address of the IV (word address).
- **Data length:** Configure the data length of the corresponding segment. The data length size needs to be consistent with `dst_data_length` (destination data length 0 +... + destination data length 7). When the algorithm is CTS mode, the higher 30-bit of the data length is the word numbers of data volume; when the `data_length[1:0]` is 0, the data length is the higher 30-bit, otherwise it is increased by 1. For AES CTS, the data length indicates the byte numbers of the source data; for other algorithms, it indicates the word numbers.
- **Source address:** The first address of source data segments. Because the storage of the source data requires word alignment, ensure that this descriptor is the first address (word address).
- **Source data length:** The data volume of source data segments. The unit is word, and those less than one word are counted as one word. Note that only the last word of the entire message is allowed to be non-integer words, and the others must be integer words.
- **Destination address:** The first address of destination data segments. Because the storage of the destination data requires word alignment, ensure that this descriptor is the first address (word address).
- **Destination data length:** The data volume of destination data segments. The unit is word, and those less than one word are counted as one word. Note that only the last word of the entire message is allowed to be non-integer words, and the others must be integer words.
- **Next descriptor:** The first address of the next task descriptor. Because the storage of the descriptor requires word alignment, ensure that this descriptor is the first address (word address).
- **Reserved:** Configure to 0x0.



### 10.1.4.3 Configuring Task Descriptor of HASH

- **Common control**
  - **Algorithm type:** Configure [Common Control](#)[6:0] to select SHA or HMAC algorithm type.
  - **Last HMAC plaintext:** If the algorithm type is HMAC, and the task is the last package of the message or if the message has only one package, then [Common Control](#)[15] needs to set to 1.
  - **IV mode:** The [Common Control](#)[16] (IV MODE) bit is only set to 1 in the following two scenarios, except that the bit must be configured to 0. (1). When the message is split into multiple packages, the [Common Control](#)[16] bit of other packages needs to be set to 1, except that the bit of the first package needs to be cleared to 0. (2). In special applications, if you need to customize the IV value to form the initial value of a certain HASH algorithm, you need to set the [Common Control](#)[16] bit of the first (or only one) package to 1, and the first address of the storage space where the customized IV value is stored in IV address.
- **Key descriptor:** Because the storage of the key requires word alignment, ensure that this descriptor is the first address of the KEY (word address).
- **IV descriptor:** In the task that requires the IV value, configure the first address of the storage space where the IV is stored here. Because the storage of the IV requires word alignment, ensure that this descriptor is the first address of the IV (word address).
- **Data length:** Configure the data length of the corresponding segment. The data length size needs to be consistent with `dst_data_length` (destination data length 0 +... + destination data length 7).
- **Source address:** The first address of source data segments. Because the storage of the source data requires word alignment, ensure that this descriptor is the first address (word address).
- **Source data length:** The data volume of source data segments. The unit is word, and those less than one word are counted as one word. Note that only the last word of the entire message is allowed to be non-integer words, and the others must be integer words.
- **Destination address:** The first address of destination data segments. Because the storage of the destination data requires word alignment, ensure that this descriptor is the first address (word address).
- **Destination data length:** The data volume of destination data segments. The unit is word, and those less than one word are counted as one word. Note that only the last word of the entire message is allowed to be non-integer words, and the others must be integer words.
- **Next descriptor:** The first address of the next task descriptor. Because the storage of the descriptor requires word alignment, ensure that this descriptor is the first address (word address).
- **Reserved:** Configure to 0x0.

#### 10.1.4.4 Configuring Task Descriptor of RSA

- **Common control:** Configure [Common Control](#)[6:0] to 0x20 to select RSA algorithm type.
- **Asymmetric control:** Configure [Asymmetric Control](#)[30:28] to select the RSA width.
- **Key descriptor:** Because the storage of the key requires word alignment, ensure that this descriptor is the first address of the KEY (word address).
- **Data length:** Configure the data length of the corresponding segment. The data length size needs to be consistent with `dst_data_length` (destination data length 0 +... + destination data length 7).
- **Source address:** The first address of source data segments. Because the storage of the source data requires word alignment, ensure that this descriptor is the first address (word address).
- **Source data length:** The data volume of source data segments. The unit is word, and those less than one word are counted as one word. Note that only the last word of the entire message is allowed to be non-integer words, and the others must be integer words.
- **Destination address:** The first address of destination data segments. Because the storage of the destination data requires word alignment, ensure that this descriptor is the first address (word address).
- **Destination data length:** The data volume of destination data segments. The unit is word, and those less than one word are counted as one word. Note that only the last word of the entire message is allowed to be non-integer words, and the others must be integer words.
- **Next descriptor:** The first address of the next task descriptor. Because the storage of the descriptor requires word alignment, ensure that this descriptor is the first address (word address).
- **Reserved:** Configure to 0x0.

#### 10.1.5 Register List

Module Name	Base Address
CE_NS	0x03040000

Register Name	Offset	Description
CE_TDA	0x0000	Task Descriptor Address
CE_ICR	0x0008	Interrupt Control Register
CE_ISR	0x000C	Interrupt Status Register
CE_TLR	0x0010	Task Load Register
CE_TSR	0x0014	Task Status Register
CE_ESR	0x0018	Error Status Register

Register Name	Offset	Description
CE_CSA	0x0024	DMA Current Source Address
CE_CDA	0x0028	DMA Current Destination Address
CE_TPR	0x002C	Throughput Register

### 10.1.6 Register Description

#### 10.1.6.1 0x0000 CE Task Descriptor Address Register (Default Value: 0x0000\_0000)

Offset: 0x0000			Register Name: CE_TDA
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	Task Descriptor Address Configure as the first address of the descriptor structure.

#### 10.1.6.2 0x0008 CE Interrupt Control Register (Default Value: 0x0000\_0000)

Offset: 0x0008			Register Name: CE_ICR
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
3:0	R/W	0x0	Task Channel3~0 Interrupt Enable 0: Disable 1: Enable

#### 10.1.6.3 0x000C CE Interrupt Status Register (Default Value: 0x0000\_0000)

Offset: 0x000C			Register Name: CE_ISR
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/

Offset: 0x000C			Register Name: CE_ISR
Bit	Read/Write	Default/Hex	Description
3:0	R/W1C	0x0	<p>Task Channel3~0 End Pending</p> <p>0: Not finished</p> <p>1: Finished</p> <p>It indicates whether task is completed.</p> <p>Write the corresponding channel bit of the register to clear the end flag.</p> <p>When the last task in the task chain ends, the operation of the task chain will end normally. If the task fails in the middle, the task chain will be aborted. The CE_ISR register will be automatically updated when it ends normally or aborts abnormally. And it is determined whether to generate an interrupt signal according to the IE configuration (bit31) of <a href="#">Common Control</a> when the current task ends or aborts.</p> <p>If using interrupt, after receiving the interrupt, read the corresponding channel bit of CE_ISR to judge whether it ends successfully or stops failure.</p> <p>If not using interrupt, the CE_ISR status register can be continuously queried for the channel bit until the successful end flag is set or the failure stop flag is set. Write the corresponding channel bit of the register to clear the end flag.</p> <p>If it fails to stop, you can read the error code on the channel corresponding to the <a href="#">CE ESR</a> register.</p>

**10.1.6.4 0x0010 CE Task Load Register (Default Value: 0x0000\_0000)**

Offset: 0x0010			Register Name: CE_TLR
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	<p>Task Load</p> <p>When set, the CE can load the descriptor of task if the task FIFO is not full.</p>

10.1.6.5 0x0014 CE Task Status Register (Default Value: 0x0000\_0000)

Offset: 0x0014			Register Name: CE_TSR
Bit	Read/Write	Default/Hex	Description
31:2	/	/	/
1:0	R	0x0	Running Channel Number 00: Task channel0 01: Task channel1 10: Task channel2 11: Task channel3

10.1.6.6 0x0018 CE Error Status Register (Default Value: 0x0000\_0000)

Offset: 0x0018			Register Name: CE_ESR
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:12	R/W1C	0x0	Task Channel3 Error Type xxx1: Algorithm not support xx1x: Data length error x1xx: keysram access error for AES 1xxx: Reserved
11:8	R/W1C	0x0	Task Channel2 Error Type xxx1: Algorithm not support xx1x: Data length error x1xx: keysram access error for AES 1xxx: Reserved
7:4	R/W1C	0x0	Task Channel1 Error Type xxx1: Algorithm not support xx1x: Data length error x1xx: keysram access error for AES 1xxx: Reserved
3:0	R/W1C	0x0	Task Channel0 Error Type xxx1: Algorithm not support xx1x: Data length error x1xx: keysram access error for AES 1xxx: Reserved

**10.1.6.7 0x0024 CE Current Source Address Register (Default Value: 0x0000\_0000)**

Offset: 0x0024			Register Name: CE_CSA
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	CUR_SRC_ADDR Current source address

**10.1.6.8 0x0028 CE Current Destination Address Register (Default Value: 0x0000\_0000)**

Offset: 0x0028			Register Name: CE_CDA
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	CUR_DST_ADDR Current destination address

**10.1.6.9 0x002C CE Throughput Register (Default Value: 0x0000\_0000)**

Offset: 0x002C			Register Name: CE_TPR
Bit	Read/Write	Default/Hex	Description
31:0	R/WC	0x0	TP_NUM It indicates the throughput writing to this register at last time. Writing to this register will clear it to 0.

## 10.2 Security ID

The Security ID (SID) is used to program and read keys which include chip ID, thermal sensor, HASH code, and so on.

The SID module has the following features:

- 2 Kbits electrical fuse (eFuse)
- Backup eFuse information by using SID\_SRAM
- A fuse only can program one time
- Burning the key to the SID
- Reading the key use status in the SID
- Loading the key to the CE



Before performing the burning operation, ensure that the power supply of the eFuse power pin is stable. After the burning operation is completed, cancel the power supply of the eFuse power pin.

---

## Appendix: Glossary

The following table contains acronyms and abbreviations used in this document.

Term	Meaning
<b>A</b>	
ADC	Analog-to-Digital Converter
AE	Automatic Exposure
AEC	Audio Echo Cancellation
AES	Advanced Encryption Standard
AF	Automatic Focus
AGC	Automatic Gain Control
AHB	AMBA High-Speed Bus
ALC	Automatic Level Control
ANR	Active Noise Reduction
APB	Advanced Peripheral Bus
ARM	Advanced RISC Machine
AVS	Audio Video Synchronization
AWB	Automatic White Balance
<b>B</b>	
BROM	Boot ROM
<b>C</b>	
CIR	Consumer Infrared
CMOS	Complementary Metal-Oxide Semiconductor
CP15	Coprocessor 15
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
CSI	Camera Serial Interface
CVBS	Composite Video Broadcast Signal
<b>D</b>	
DDR	Double Data Rate
DES	Data Encryption Standard
DLL	Delay-Locked Loop
DMA	Direct Memory Access
DRC	Dynamic Range Compression
DVFS	Dynamic Voltage and Frequency Scaling
<b>E</b>	
ECC	Error Correction Code
eFuse	Electrical Fuse, A one-time programmable memory
EHCI	Enhanced Host Controller Interface
eMMC	Embedded Multi-Media Card
ESD	Electrostatic Discharge
<b>F</b>	
FBGA	Fine Pitch Ball Grid Array



Term	Meaning
FEL	Fireware Exchange Launch
FIFO	First In First Out
<b>G</b>	
GPIO	General Purpose Input Output
<b>I</b>	
I2C	Inter Integrated Circuit
I2S	Inter IC Sound
ISP	Image Signal Processor
<b>J</b>	
JEDEC	Joint Electron Device Engineering Council
JPEG	Joint Photographic Experts Group
JTAG	Joint Test Action Group
<b>L</b>	
LCD	Liquid-Crystal Display
LFBGA	Low Profile Fine Pitch Ball Grid Array
LSB	Least Significant Bit
LVDS	Low Voltage Differential Signaling
<b>M</b>	
MAC	Media Access Control
MIC	Microphone
MIPI	Mobile Industry Processor Interface
MLC	Multi-Level Cell
MMC	Multimedia Card
MPEG	Motion Pictures Expert Group
MSB	Most Significant Bit
<b>N</b>	
N/A	Not Application
NMI	Non Maskable Interrupt
NTSC	National Television Standards Committee
NVM	Non Volatile Storage Medium
<b>O</b>	
OHCI	Open Host Controller Interface
OTP	One Time Programmable
OWA	One Wire Audio
<b>P</b>	
PAL	Phase Alternating Line
PCM	Pulse Code Modulation
PHY	Physical Layer Controller
PID	Packet Identifier
PLIC	Platform-Level Interrupt Controller
PLL	Phase-Locked Loop
POR	Power-On Reset
PRCM	Power Reset Clock Management

Term	Meaning
PWM	Pulse Width Modulation
<b>R</b>	
R	Read only/non-Write
RGB	Read Green Blue
RGMII	Reduced Gigabit Media Independent Interface
RMII	Reduced Media Independent Interface
ROM	Read Only Memory
RSA	Rivest-Shamir-Adleman
RTC	Real Time Clock
<b>S</b>	
SAR	Successive Approximation Register
SD	Secure Digital
SDIO	Secure Digital Input Output
SDK	Software Development Kit
SDRAM	Synchronous Dynamic Random Access Memory
SDXC	Secure Digital Extended Capacity
SLC	Single-Level Cell
SoC	System on Chip
SPI	Serial Peripheral Interface
SRAM	Static Random Access Memory
<b>T</b>	
TDES	Triple Data Encryption Standard
TWI	Two Wire Interface
<b>U</b>	
UART	Universal Asynchronous Receiver Transmitter
UDF	Undefined
USB DRD	Universal Serial Bus Dual Role Device
UTMI	USB2.0 Transceiver Macrocell Interface

**Copyright©2022 Allwinner Technology Co.,Ltd. All Rights Reserved.**

This documentation is the original work and copyrighted property of Allwinner Technology Co.,Ltd (“Allwinner”). No part of this document may be reproduced, modify, publish or transmitted in any form or by any means without prior written consent of Allwinner.

#### **Trademarks and Permissions**

Allwinner and the Allwinner logo (incomplete enumeration) are trademarks of Allwinner Technology Co.,Ltd. All other trademarks, trade names, product or service names mentioned in this document are the property of their respective owners.

#### **Important Notice and Disclaimer**

The purchased products, services and features are stipulated by the contract made between Allwinner Technology Co.,Ltd (“Allwinner”) and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Please read the terms and conditions of the contract and relevant instructions carefully before using, and follow the instructions in this documentation strictly. Allwinner assumes no responsibility for the consequences of improper use (including but not limited to overvoltage, overclock, or excessive temperature).

The information in this document is provided just as a reference or typical applications, and is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents. Allwinner is not responsible for any damage (including but not limited to indirect, incidental or special loss) or any infringement of third party rights arising from the use of this document. All statements, information, and recommendations in this document do not constitute a warranty or commitment of any kind, express or implied.

No license is granted by Allwinner herein express or implied or otherwise to any patent or intellectual property of Allwinner. Third party licences may be required to implement the solution/product. Customers shall be solely responsible to obtain all appropriately required third party licences. Allwinner shall not be liable for any licence fee or royalty due in respect of any required third party licence. Allwinner shall have no warranty, indemnity or other obligations with respect to third party licences.