

| Offset: 0x0170 | | | Register Name: LCD_CPU_TRI4_REG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 28 | R/W | 0x0 | PLUG_MODE_EN Enable the plug mode used in dsi command mode. 0: Disable 1: Enable |
| 27:25 | / | / | / |
| 24 | R/W | 0x0 | A1_First_Valid Valid in first Block. |
| 23:0 | R/W | 0x0 | D23_TO_D0_First_Valid Valid in first Block. |

7.1.6.30 0x0174 LCD CPU Panel Trigger Register5 (Default Value: 0x0000_0000)

| Offset: 0x0174 | | | Register Name: LCD_CPU_TRI5_REG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:25 | / | / | / |
| 24 | R/W | 0x0 | A1_NON_First_Valid Valid in Block except first. |
| 23:0 | R/W | 0x0 | D23_TO_D0_NON_First_Valid Valid in Block except first. |

7.1.6.31 0x0180 LCD Color Map Control Register (Default Value: 0x0000_0000)

| Offset: 0x0180 | | | Register Name: LCD_CMAP_CTL_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31 | R/W | 0x0 | COLOR_MAP_EN Enable the color map function. This module only works when X is divided by 4. 0: Bypass 1: Enable |
| 30:1 | / | / | / |
| 0 | R/W | 0x0 | OUT_FORMAT Set the pixel output format in color map function. 0: 4 pixel output mode: Out0 -> Out1 -> Out2 -> Out3 1: 2 pixel output mode: Out0 -> Out1 |

7.1.6.32 0x0190 LCD Color Map Odd Line Register0 (Default Value: 0x0000_0000)

| Offset: 0x0190 | | | Register Name: LCD_CMAP_ODD0_REG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | R/W | 0x0 | <p>OUT_ODD1 Indicates the output order of components.</p> <p>bit15-12: Reserved</p> <p>bit11-08: Out_Odd0[23:16]</p> <p>bit07-04: Out_Odd0[15:8]</p> <p>bit03-00: Out_Odd0[7:0]</p> <p>0000: in_b0</p> <p>0001: in_g0</p> <p>0010: in_r0</p> <p>0011: Reserved</p> <p>0100: in_b1</p> <p>0101: in_g1</p> <p>0110: in_r1</p> <p>0111: Reserved</p> <p>1000: in_b2</p> <p>1001: in_g2</p> <p>1010: in_r2</p> <p>1011: Reserved</p> <p>1100: in_b3</p> <p>1101: in_g3</p> <p>1110: in_r3</p> <p>1111: Reserved</p> |

| Offset: 0x0190 | | | Register Name: LCD_CMAP_ODD0_REG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 15:0 | R/W | 0x0 | <p>OUT_ODD0 Indicates the output order of components.</p> <p>bit15-12: Reserved</p> <p>bit11-08: Out_Odd0[23:16]</p> <p>bit07-04: Out_Odd0[15:8]</p> <p>bit03-00: Out_Odd0[7:0]</p> <p>0000: in_b0</p> <p>0001: in_g0</p> <p>0010: in_r0</p> <p>0011: Reserved</p> <p>0100: in_b1</p> <p>0101: in_g1</p> <p>0110: in_r1</p> <p>0111: Reserved</p> <p>1000: in_b2</p> <p>1001: in_g2</p> <p>1010: in_r2</p> <p>1011: Reserved</p> <p>1100: in_b3</p> <p>1101: in_g3</p> <p>1110: in_r3</p> <p>1111: Reserved</p> |

7.1.6.33 0x0194 LCD Color Map Odd Line Register1 (Default Value: 0x0000_0000)

| Offset: 0x0194 | | | Register Name: LCD_CMAP_ODD1_REG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | R/W | 0x0 | <p>OUT_ODD3 Indicates the output order of components.</p> <p>bit15-12: Reserved</p> <p>bit11-08: Out_Odd0[23:16]</p> <p>bit07-04: Out_Odd0[15:8]</p> <p>bit03-00: Out_Odd0[7:0]</p> <p>0000: in_b0</p> <p>0001: in_g0</p> <p>0010: in_r0</p> <p>0011: Reserved</p> <p>0100: in_b1</p> <p>0101: in_g1</p> <p>0110: in_r1</p> <p>0111: Reserved</p> <p>1000: in_b2</p> <p>1001: in_g2</p> <p>1010: in_r2</p> <p>1011: Reserved</p> <p>1100: in_b3</p> <p>1101: in_g3</p> <p>1110: in_r3</p> <p>1111: Reserved</p> |

| Offset: 0x0194 | | | Register Name: LCD_CMAP_ODD1_REG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 15:0 | R/W | 0x0 | <p>OUT_ODD2 Indicates the output order of components.</p> <p>bit15-12: Reserved</p> <p>bit11-08: Out_Odd0[23:16]</p> <p>bit07-04: Out_Odd0[15:8]</p> <p>bit03-00: Out_Odd0[7:0]</p> <p>0000: in_b0</p> <p>0001: in_g0</p> <p>0010: in_r0</p> <p>0011: Reserved</p> <p>0100: in_b1</p> <p>0101: in_g1</p> <p>0110: in_r1</p> <p>0111: Reserved</p> <p>1000: in_b2</p> <p>1001: in_g2</p> <p>1010: in_r2</p> <p>1011: Reserved</p> <p>1100: in_b3</p> <p>1101: in_g3</p> <p>1110: in_r3</p> <p>1111: Reserved</p> |

7.1.6.34 0x0198 LCD Color Map Even Line Register0 (Default Value: 0x0000_0000)

| Offset: 0x0198 | | | Register Name: LCD_CMAP_EVEN0_REG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | R/W | 0x0 | <p>OUT_EVEN1 Indicates the output order of components. bit15-12: Reserved bit11-08: Out_Odd0[23:16] bit07-04: Out_Odd0[15:8] bit03-00: Out_Odd0[7:0]</p> <p>0000: in_b0 0001: in_g0 0010: in_r0 0011: Reserved 0100: in_b1 0101: in_g1 0110: in_r1 0111: Reserved 1000: in_b2 1001: in_g2 1010: in_r2 1011: Reserved 1100: in_b3 1101: in_g3 1110: in_r3 1111: Reserved</p> |

| Offset: 0x0198 | | | Register Name: LCD_CMAP_EVENO_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 15:0 | R/W | 0x0 | <p>OUT_EVENO Indicates the output order of components.</p> <p>bit15-12: Reserved</p> <p>bit11-08: Out_Odd0[23:16]</p> <p>bit07-04: Out_Odd0[15:8]</p> <p>bit03-00: Out_Odd0[7:0]</p> <p>0000: in_b0</p> <p>0001: in_g0</p> <p>0010: in_r0</p> <p>0011: Reserved</p> <p>0100: in_b1</p> <p>0101: in_g1</p> <p>0110: in_r1</p> <p>0111: Reserved</p> <p>1000: in_b2</p> <p>1001: in_g2</p> <p>1010: in_r2</p> <p>1011: Reserved</p> <p>1100: in_b3</p> <p>1101: in_g3</p> <p>1110: in_r3</p> <p>1111: Reserved</p> |

7.1.6.35 0x019C LCD Color Map Even Line Register1 (Default Value: 0x0000_0000)

| Offset: 0x019C | | | Register Name: LCD_CMAP_EVEN1_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | R/W | 0x0 | <p>OUT_EVEN3 Indicates the output order of components.</p> <p>bit15-12: Reserved</p> <p>bit11-08: Out_Odd0[23:16]</p> <p>bit07-04: Out_Odd0[15:8]</p> <p>bit03-00: Out_Odd0[7:0]</p> <p>0000: in_b0</p> <p>0001: in_g0</p> <p>0010: in_r0</p> <p>0011: Reserved</p> <p>0100: in_b1</p> <p>0101: in_g1</p> <p>0110: in_r1</p> <p>0111: Reserved</p> <p>1000: in_b2</p> <p>1001: in_g2</p> <p>1010: in_r2</p> <p>1011: Reserved</p> <p>1100: in_b3</p> <p>1101: in_g3</p> <p>1110: in_r3</p> <p>1111: Reserved</p> |

| Offset: 0x019C | | | Register Name: LCD_CMAP_EVEN1_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 15:0 | R/W | 0x0 | <p>OUT_EVEN2 Indicates the output order of components.</p> <p>bit15-12: Reserved</p> <p>bit11-08: Out_Odd0[23:16]</p> <p>bit07-04: Out_Odd0[15:8]</p> <p>bit03-00: Out_Odd0[7:0]</p> <p>0000: in_b0</p> <p>0001: in_g0</p> <p>0010: in_r0</p> <p>0011: Reserved</p> <p>0100: in_b1</p> <p>0101: in_g1</p> <p>0110: in_r1</p> <p>0111: Reserved</p> <p>1000: in_b2</p> <p>1001: in_g2</p> <p>1010: in_r2</p> <p>1011: Reserved</p> <p>1100: in_b3</p> <p>1101: in_g3</p> <p>1110: in_r3</p> <p>1111: Reserved</p> |

7.1.6.36 0x01F0 LCD Safe Period Register (Default Value: 0x0000_0000)

| Offset: 0x01F0 | | | Register Name: LCD_SAFE_PERIOD_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:29 | / | / | / |
| 28:16 | R/W | 0x0 | <p>SAFE_PERIOD_FIFO_NUM</p> <p>When the data length in line buffer is more than SAFE_PERIOD_FIFO_NUM, the LCD controller will allow dram controller to stop working to change frequency.</p> |
| 15:4 | R/W | 0x0 | <p>SAFE_PERIOD_LINE</p> <p>Set a fixed line and during the line time, the LCD controller allow dram controller to change frequency. The fixed line should be set in the blanking area.</p> |
| 3 | / | / | / |

| Offset: 0x01F0 | | | Register Name: LCD_SAFE_PERIOD_REG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 2:0 | R/W | 0x0 | <p>SAFE_PERIOD_MODE Select the save mode</p> <p>000: unsafe 001: safe 010: safe at FIFO_CURR_NUM > SAFE_PERIOD_FIFO_NUM (this mode is unavailable in V853/V853S) 011: safe at 2 and safe at sync active 100: safe at line</p> |

7.1.6.37 0x0228 LCD FSYNC Generate Control Register (Default Value: 0x0000_0000)

| Offset: 0x0228 | | | Register Name: FSYNC_GEN_CTRL_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:19 | / | / | / |
| 18:8 | R/W | 0x0 | <p>SENSOR_DIS_TIME Delay 0–2047 Hsync Period When HSYNC_POL_SEL is 0, the actual delay is SENSOR_DIS_TIME-1. When HSYNC_POL_SEL is 1, the actual delay is SENSOR_DIS_TIME.</p> |
| 7 | / | / | / |
| 6 | R/W | 0x0 | <p>SENSOR_ACT1_VALUE Sensor Active1 Value 0: Fsync active_1 period output 0 1: Fsync active_1 period output 1</p> |
| 5 | R/W | 0x0 | <p>SENSOR_ACT0_VALUE Sensor Active0 Value 0: Fsync active_0 period output 0 1: Fsync active_0 period output 1</p> |
| 4 | R/W | 0x0 | <p>SENSOR_DIS_VALUE Sensor Disable Value 0: Fsync disable period output 0 1: Fsync disable period output 1</p> |
| 3 | / | / | / |
| 2 | R/W | 0x0 | <p>HSYNC_POL_SEL Hsync Polarity Select 0: Normal 1: Opposite hsync to hysnc counter</p> |
| 1 | R/W | 0x0 | <p>SEL_VSYNC_EN Select Vsync Enable 0: Select vsync falling edge to start state machine 1: Select vsync rising edge to start state machine</p> |

| Offset: 0x0228 | | | Register Name: FSYNC_GEN_CTRL_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 0 | R/W | 0x0 | FSYNC_GEN_EN Fsync Generate Enable 0: Disable 1: Enable |

7.1.6.38 0x022C LCD FSYNC Generate Delay Register (Default Value: 0x0000_0000)

| Offset: 0x022C | | | Register Name: FSYNC_GEN_DLY_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | / | / | / |
| 27:16 | R/W | 0x0 | SENSOR_ACT0_TIME Delay 0~4095 Pixel CLK Period The actual delay is sensor_act0_time+1. |
| 15:12 | / | / | / |
| 11:0 | R/W | 0x0 | SENSOR_ACT1_TIME Delay 0~4095 Pixel CLK Period The actual delay is sensor_act1_time+1. |

7.1.6.39 0x0230 LCD Sync Control Register (Default Value: 0x0000_0000)

| Offset: 0x0230 | | | Register Name: LCD_SYNC_CTL_REG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:9 | / | / | / |
| 8 | R/W | 0x0 | LCD_CTRL_WORK_MODE LCD Controller Work mode 0: Single DSI mode 1: Dual DSI mode |
| 7:5 | / | / | / |
| 4 | R/W | 0x0 | LCD_CYRL_SYNC_MASTER_SLAVE LCD Controller Sync Master Slave 0: Master 1: Slave Note: Only use in Single DSI mode. |
| 3:1 | / | / | / |
| 0 | R/W | 0x0 | LCD_CTRL_SYNC_MODE LCD Controller Sync Mode 0: Sync in the first time 1: Sync every frame Note: Only use in Single DSI mode. |

7.1.6.40 0x0234 LCD Sync Position Register (Default Value: 0x0000_0000)

| Offset: 0x0234 | | | Register Name: LCD_SYNC_POS_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | / | / | / |
| 27:16 | R/W | 0x0 | <p>LCD_Sync_Pixel_Num Set the pixel number of master LCD controller which is used to trigger the slave LCD controller to start working. This value is the number of pixels between the trigger point and the end of the line. Tri pos = $Tline * LCD_Sync_Line_Num + Tpixel * (HT - LCD_Sync_Pixel_Num)$</p> <p>Note: Only use in Single DSI mode.</p> |
| 15:12 | / | / | / |
| 11:0 | R/W | 0x0 | <p>LCD_Sync_Line_Num Set the line number of master LCD controller which is used to trigger the slave LCD controller to start working. Note: It is only set in master LCD controller. It is not necessarily to set in slave LCD controller. Tri pos = $Tline * LCD_Sync_Line_Num + Tpixel * (HT - LCD_Sync_Pixel_Num)$</p> <p>Note: Only use in Single DSI mode.</p> |

7.1.6.41 0x0238 LCD Slave Stop Position Register (Default Value: 0x0000_0000)

| Offset: 0x0238 | | | Register Name: LCD_SLAVE_STOP_POS_REG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:0 | R/W | 0x0 | <p>STOP_VAL Set the stop position of the slave LCD. This value is the number of pixels between the stop position and the end of the HFP. Stop_pos = HFP - Stop_val. $0 < Stop_pos < HFP - 2$</p> <p>Note: Only use in Single DSI mode.</p> |

7.2 MIPI DSI

The Display Serial Interface is a high-speed interface between a host processor and peripheral devices that adhere to MIPI Alliance specifications for mobile device interfaces. This DSI module is composed of a DSI controller which is compliance with MIPI DSI specification V1.02 and a D-PHY module which is compliance with MIPI DPHY specification V1.2.

The MIPI DSI includes the following features:

- Compliance with MIPI DSI v1.02
- Up to 4 lanes
- Supports 1280 x 720@60fps and 1920 x 1200@60fps
- Supports non-burst mode with sync pulse/sync event and burst mode
- Supports pixel format: RGB888, RGB666, RGB666 loosely packed and RGB565
- Supports continuous lane clock mode and non-continuous lane clock mode
- Compliance with MIPI DCS v1.01, bidirectional communication in LP through data lane 0
- Supports bidirectional communication of all generic commands in LP through data lane 0
- Supports low power data transmission
- Supports ULPS and Escape modes
- Hardware checksum capabilities

Contents

| | | |
|-------|------------------------------|-----|
| 8 | Audio | 598 |
| 8.1 | I2S/PCM..... | 598 |
| 8.1.1 | Overview | 598 |
| 8.1.2 | Block Diagram | 599 |
| 8.1.3 | Functional Descriptions..... | 600 |
| 8.1.4 | Operation Modes | 603 |
| 8.1.5 | Register List..... | 604 |
| 8.1.6 | Register Description | 605 |
| 8.2 | DMIC..... | 641 |
| 8.2.1 | Overview | 641 |
| 8.2.2 | Block Diagram | 641 |
| 8.2.3 | Functional Description | 641 |
| 8.2.4 | Operation Modes | 642 |
| 8.2.5 | Register List..... | 643 |
| 8.2.6 | Register Description | 644 |
| 8.3 | Audio Codec | 656 |
| 8.3.1 | Overview | 656 |
| 8.3.2 | Block Diagram | 657 |
| 8.3.3 | Functional Description | 657 |
| 8.3.4 | Programming Guidelines..... | 664 |
| 8.3.5 | Register List..... | 665 |
| 8.3.6 | Register Description | 669 |

Figures

| | |
|---|-----|
| Figure 8-1 I2S/PCM Interface System Block Diagram | 599 |
| Figure 8-2 Typical Application of I2S/PCM Interface | 599 |
| Figure 8-3 I2S Standard Mode Timing..... | 601 |
| Figure 8-4 Left-justified Mode Timing..... | 601 |
| Figure 8-5 Right-justified Mode Timing | 601 |
| Figure 8-6 PCM Long Frame Mode Timing..... | 602 |
| Figure 8-7 PCM Short Frame Mode Timing..... | 602 |
| Figure 8-8 I2S/PCM Operation Flow | 603 |
| Figure 8-9 DMIC Block Diagram | 641 |
| Figure 8-10 DMIC Operation Mode..... | 642 |
| Figure 8-11 Audio Codec Block Diagram..... | 657 |
| Figure 8-12 Audio Codec Clock Diagram..... | 658 |
| Figure 8-13 Digital Part Reset System..... | 659 |
| Figure 8-14 Analog Part Reset System | 659 |
| Figure 8-15 Audio Codec Data Path Diagram..... | 660 |
| Figure 8-16 Audio Codec Interrupt System..... | 661 |
| Figure 8-17 DAP Data Flow | 661 |
| Figure 8-18 HPF Logic Structure..... | 662 |
| Figure 8-19 DRC static Curve Parameters | 662 |
| Figure 8-20 DRC Block Diagram | 662 |
| Figure 8-21 Energy Filter Structure..... | 663 |
| Figure 8-22 Gain Smooth Filter | 664 |

Tables

| | |
|--|-----|
| Table 8-1 I2S/PCM External Signals | 600 |
| Table 8-2 I2S/PCM Clock Sources..... | 600 |
| Table 8-3 DMIC External Signals | 641 |
| Table 8-4 DMIC Clock Sources..... | 642 |
| Table 8-5 Audio Codec analog I/O | 657 |
| Table 8-6 Audio Codec reference I/O..... | 657 |



8 Audio

8.1 I2S/PCM

8.1.1 Overview

The I2S/PCM Controller is designed to transfer the streaming audio-data between the system memory and the codec chip. The controller supports the standard I2S format, Left-justified mode format, Right-justified mode format, PCM mode format, and TDM mode format.

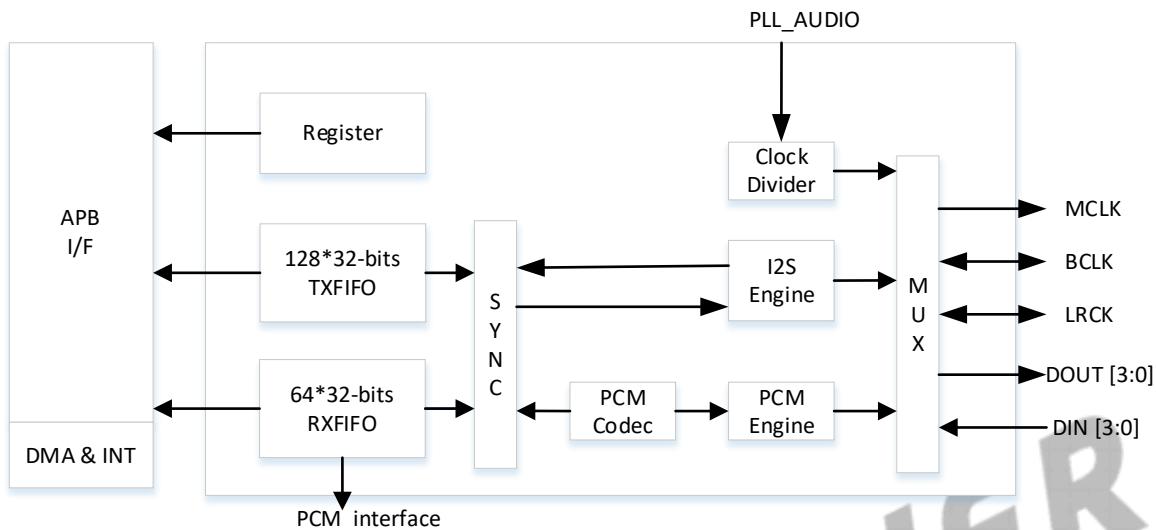
The I2S/PCM controller includes the following features:

- Two I2S/PCM external interfaces (I2S0, I2S1) for connecting external power amplifier and MIC ADC
- Compliant with standard Philips Inter-IC sound (I2S) bus specification
 - Left-justified, Right-justified, PCM mode, and Time Division Multiplexing (TDM) format
 - Programmable PCM frame width: 1 BCLK width (short frame) and 2 BCLKs width (long frame)
- Transmit and Receive data FIFOs
 - Programmable FIFO thresholds
 - 128 depth x 32-bit width TXFIFO and 64 depth x 32-bit width RXFIFO
- Supports multiple function clock
 - Clock up to 24.576 MHz Data Output of I2S/PCM in Master mode (Only if the IO PAD and Peripheral I2S/PCM satisfy Timing Parameters)
 - Clock up to 12.288 MHz Data Input of I2S/PCM in Master mode
- Supports TX/RX DMA Slave interface
- Supports multiple application scenarios
 - Up to 16 channels ($f_s = 48 \text{ kHz}$) which has adjustable width from 8-bit to 32-bit
 - Sample rate from 8 kHz to 384 kHz (CHAN = 2)
 - 8-bit u-law and 8-bit A-law companded sample
- Supports master/slave mode

8.1.2 Block Diagram

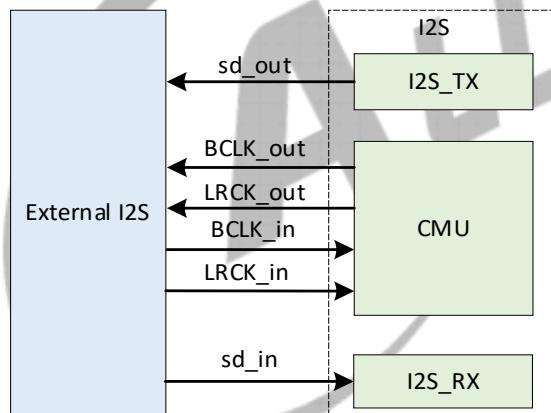
The following figure shows the functional block diagram of the I2S/PCM interface.

Figure 8-1 I2S/PCM Interface System Block Diagram



The following figure shows the typical application of the I2S/PCM interface.

Figure 8-2 Typical Application of I2S/PCM Interface



The I2S/PCM interface system integrates one I2S_TX and one I2S_RX.

- The I2S_TX is for playing music in I2S or PCM format.
- The I2S_RX is for receiving data in I2S or PCM format.
- When the I2S works in the slave mode, the external I2S module provides BCLK_in and LRCK_in for the clock management unit (CMU), and the I2S_TX and I2S_RX work with the two external clocks.
- When the I2S works in the master mode, the CMU provides clocks BCLK_out and LRCK_out for the external I2S module, and the I2S_TX and I2S_RX work with the internal clocks.

8.1.3 Functional Descriptions

8.1.3.1 External Signals

The following table describes the external signals of the I2S/PCM interface.

LRCK and BCLK are bidirectional I/O. When the I2S/PCM interface works in the Master mode, LRCK and BCLK are output pins. When the I2S/PCM interface works in the Slave mode, LRCK and BCLK are input pins.

MCLK is an output pin for external devices. DOUT is a serial data output pin and DIN is an serial data input pin. For details about General Purpose I/O port, refer to section 10.5 GPIO.

Table 8-1 I2S/PCM External Signals

| Signal | Description | Type |
|----------------|--|------|
| I2S0 | | |
| I2S0-MCLK | I2S0 Master Clock | O |
| I2S0-LRCK | I2S0/PCM0 Sample Rate Clock/Sync | I/O |
| I2S0-BCLK | I2S0/PCM0 Sample Rate Clock | I/O |
| I2S0-DOUT0 | I2S0/PCM0 Serial Data Output Channel 0 | O |
| I2S0-DIN0 | I2S0/PCM0 Serial Data Input Channel 0 | I |
| I2S1 | | |
| I2S1-MCLK | I2S1 Master Clock | O |
| I2S1-LRCK | I2S1/PCM1 Sample Rate Clock/Sync | I/O |
| I2S1-BCLK | I2S1/PCM1 Sample Rate Clock | I/O |
| I2S1-DOUT[3:0] | I2S1/PCM1 Serial Data Output Channel [3:0] | O |
| I2S1-DIN[3:0] | I2S1/PCM1 Serial Data Input Channel [3:0] | I |

8.1.3.2 Clock Sources

The following table describes the clock sources for I2S/PCM. For clock setting, configurations, and gating information, refer to section 3.4 Clock Controller Unit (CCU).

Table 8-2 I2S/PCM Clock Sources

| Clock Sources | Description |
|---------------|--|
| PLL_AUDIO | 24.576 MHz or 22.5792 MHz generated by PLL_AUDIO to produce 48 KHz or 44.1 KHz sample frequency. |

8.1.3.3 Timing Diagram

The I2S/PCM support standard I2S mode, Left-justified I2S mode, Right-justified I2S mode, PCM mode, and TDM mode. Software can select one of them in which the I2S/PCM works by setting the [I2S/PCM CTL \(Offset: 0x0000\)](#). The following figures describe the waveforms for LRCK, BCLK, DOUT, and DIN in different modes.

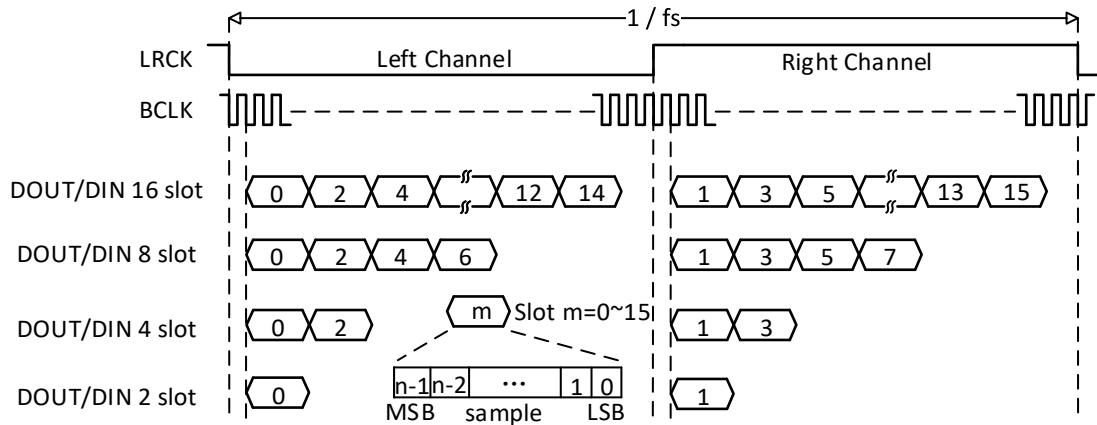
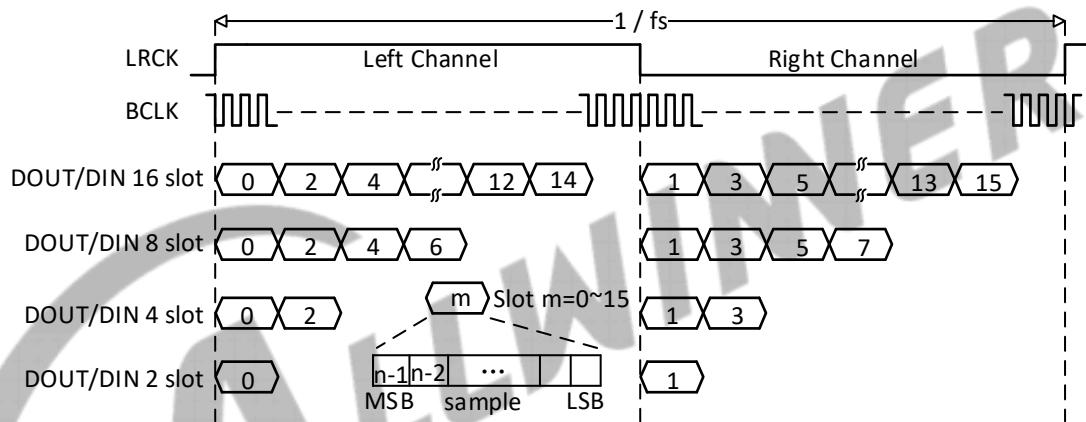
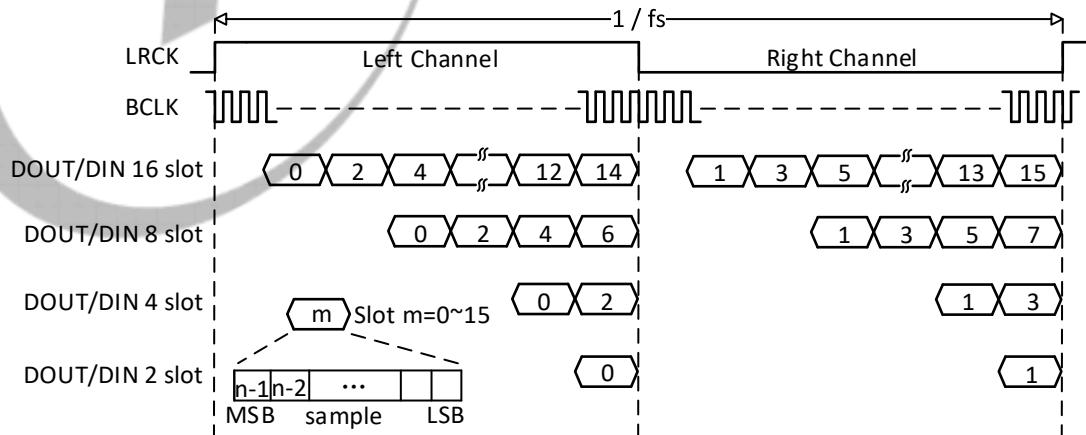
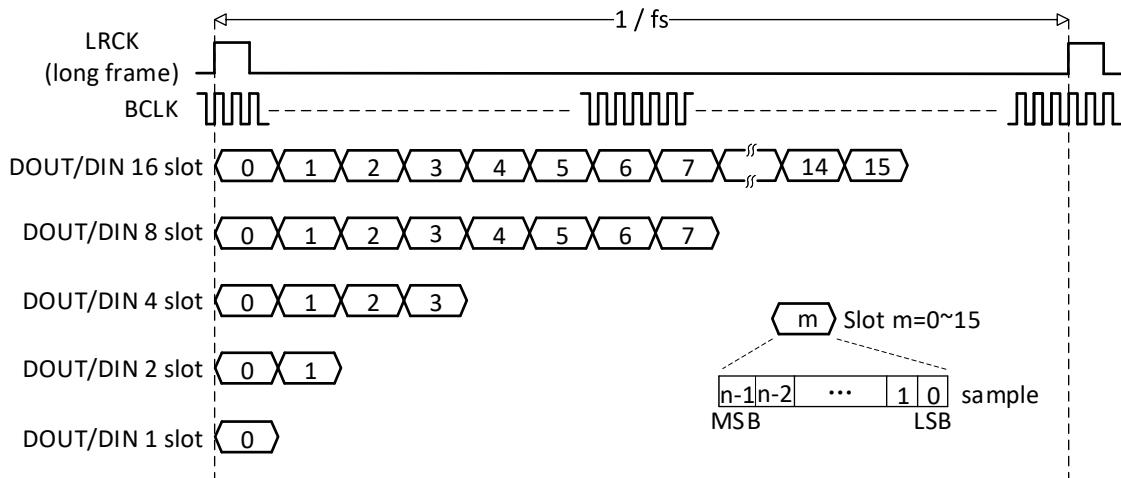
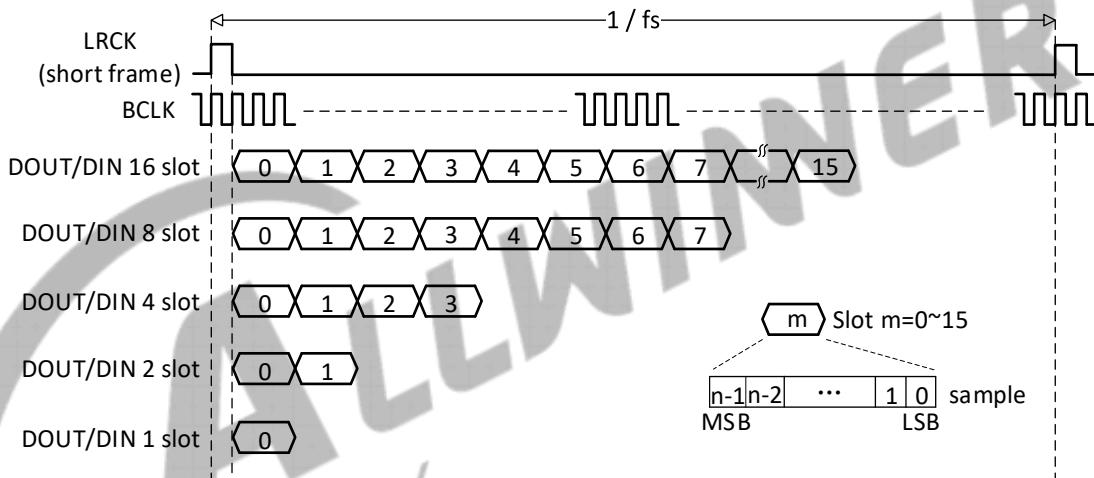
Figure 8-3 I2S Standard Mode Timing**Figure 8-4 Left-justified Mode Timing****Figure 8-5 Right-justified Mode Timing**

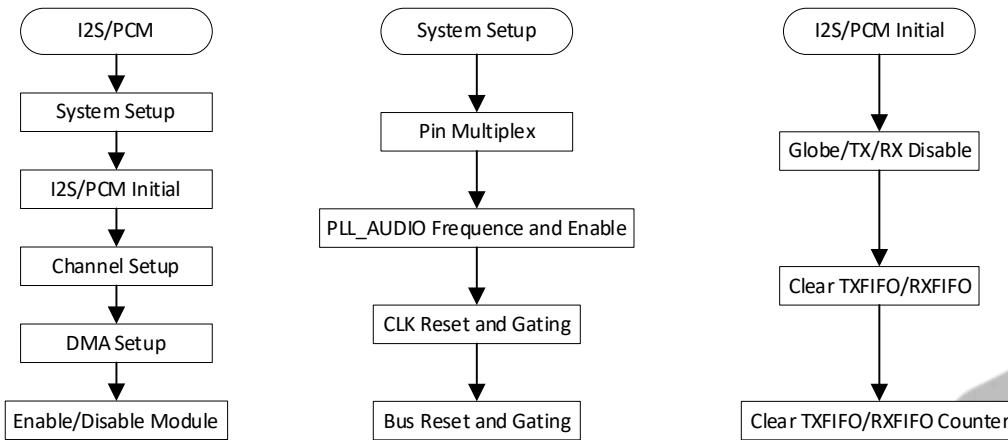
Figure 8-6 PCM Long Frame Mode Timing**Figure 8-7 PCM Short Frame Mode Timing**

8.1.4 Operation Modes

The software operation of the I2S/PCM is divided into five steps: system setup, I2S/PCM initialization, the channel setup, DMA setup, and Enable/Disable module.

The following figure shows the whole operation flow of I2S/PCM.

Figure 8-8 I2S/PCM Operation Flow



8.1.4.1 System Setup and I2S/PCM Initialization

The first step in the system setup is properly programming the GPIO. Because the I2S/PCM port is a multiplex pin. You can find the function in the pin multiplex specification.

The clock source for the I2S/PCM should be followed.

1. Disable the PLL_AUDIO though the [PLL_ENABLE bit of PLL_AUDIO_CTRL_REG \(Offset: 0x0078\)](#) in the CCU.
2. Set up the frequency of the PLL_AUDIO in the [PLL_AUDIO_CTRL_REG \(Offset: 0x0078\)](#).
3. Enable PLL_AUDIO.
4. Enable the I2S/PCM gating though the [I2SPCMx_CLK_REG \(Offset: 0x0A10\)](#) when you checkout that the LOCK bit of [PLL_AUDIO_CTRL_REG \(Offset: 0x0078\)](#) becomes to 1.
5. Reset and enable the I2S/PCM bus gating in the [I2SPCM_BGR_REG \(Offset: 0x0A20\)](#).

After the system setup, set up the register of I2S/PCM.

1. Initialize the I2S/PCM. Close the globe enable bit ([I2S/PCM_CTL \(Offset: 0x0000\) \[0\]](#)), TX enable bit ([I2S/PCM_CTL \(Offset: 0x0000\) \[2\]](#)), and RX enable bit ([I2S/PCM_CTL \(Offset: 0x0000\) \[1\]](#)) by writing 0 to it.
2. Clear the TX/RX FIFO by writing 0 to the bit[25:24] of [I2S/PCM_FCTL \(Offset: 0x0014\)](#).
3. Clear the TX/RX FIFO counter by writing 0 to [I2S/PCM_TXCNT \(Offset: 0x0028\)](#) and [I2S/PCM_RXCNT \(Offset: 0x002C\)](#).

8.1.4.2 Channel Setup and DMA Setup

1. Set up the I2S/PCM of master and slave. The configuration can be referred to the protocol of I2S/PCM.
2. Set up the translation mode, the sample resolution, the width of slot, the channel slot number, the trigger level, etc. The setup of register can be found in the specification.

The I2S/PCM supports two methods to transfer the data. The most common way is DMA, the setup of DMA can be found in the 3.8 Direct Memory Access Controller (DMAC). In this module, just enable the DRQ.

8.1.4.3 Enabling and Disabling the I2S/PCM

Refer to the following steps to enable this function.

1. Enable TX/RX by writing the [I2S/PCM_CTL \(Offset: 0x0000\)](#) [2:1].
2. Enable I2S/PCM by writing 1 to the Globe Enable bit in the [I2S/PCM_CTL \(Offset: 0x0000\)](#).

If you want to disable this function, write 0 to the Globe Enable bit to disable I2S/PCM.

8.1.5 Register List

| Module Name | Base Address | Comments |
|-------------|--------------|---|
| I2S PCM0 | 0x02032000 | Use for Speech Input. |
| I2S PCM1 | 0x02033000 | I2S PCM1 register is the same with I2S PCM0 .Use for Bluetooth. |

| Register Name | Offset | Description |
|-------------------|--------|--|
| I2S PCM_CTL | 0x0000 | I2S PCM Control Register |
| I2S PCM_FMT0 | 0x0004 | I2S PCM Format Register 0 |
| I2S PCM_FMT1 | 0x0008 | I2S PCM Format Register 1 |
| I2S PCMISTA | 0x000C | I2S PCM Interrupt Status Register |
| I2S PCM_RXFIFO | 0x0010 | I2S PCM RXFIFO Register |
| I2S PCM_FCTL | 0x0014 | I2S PCM FIFO Control Register |
| I2S PCM_FSTA | 0x0018 | I2S PCM FIFO Status Register |
| I2S PCM_INT | 0x001C | I2S PCM DMA And Interrupt Control Register |
| I2S PCM_TXFIFO | 0x0020 | I2S PCM TXFIFO Register |
| I2S PCM_CLKD | 0x0024 | I2S PCM Clock Divide Register |
| I2S PCM_TXCNT | 0x0028 | I2S PCM TX Sample Counter Register |
| I2S PCM_RXCNT | 0x002C | I2S PCM RX Sample Counter Register |
| I2S PCM_CHCFG | 0x0030 | I2S PCM Channel Configuration Register |
| I2S PCM_TX0CHSEL | 0x0034 | I2S PCM TX0 Channel Select Register |
| I2S PCM_TX1CHSEL | 0x0038 | I2S PCM TX1 Channel Select Register |
| I2S PCM_TX2CHSEL | 0x003C | I2S PCM TX2 Channel Select Register |
| I2S PCM_TX3CHSEL | 0x0040 | I2S PCM TX3 Channel Select Register |
| I2S PCM_TX0CHMAPO | 0x0044 | I2S PCM TX0 Channel Mapping Register0 |

| Register Name | Offset | Description |
|-------------------|--------|---------------------------------------|
| I2S PCM_TX0CHMAP1 | 0x0048 | I2S PCM TX0 Channel Mapping Register1 |
| I2S PCM_TX1CHMAP0 | 0x004C | I2S PCM TX1 Channel Mapping Register0 |
| I2S PCM_TX1CHMAP1 | 0x0050 | I2S PCM TX1 Channel Mapping Register1 |
| I2S PCM_TX2CHMAP0 | 0x0054 | I2S PCM TX2 Channel Mapping Register0 |
| I2S PCM_TX2CHMAP1 | 0x0058 | I2S PCM TX2 Channel Mapping Register1 |
| I2S PCM_TX3CHMAP0 | 0x005C | I2S PCM TX3 Channel Mapping Register0 |
| I2S PCM_TX3CHMAP1 | 0x0060 | I2S PCM TX3 Channel Mapping Register1 |
| I2S PCM_RXCHSEL | 0x0064 | I2S PCM RX Channel Select Register |
| I2S PCM_RXCHMAP0 | 0x0068 | I2S PCM RX Channel Mapping Register0 |
| I2S PCM_RXCHMAP1 | 0x006C | I2S PCM RX Channel Mapping Register1 |
| I2S PCM_RXCHMAP2 | 0x0070 | I2S PCM RX Channel Mapping Register2 |
| I2S PCM_RXCHMAP3 | 0x0074 | I2S PCM RX Channel Mapping Register3 |

8.1.6 Register Description

8.1.6.1 0x0000 I2S/PCM Control Register (Default Value: 0x0006_0000)

| Offset: 0x0000 | | | Register Name: I2S/PCM_CTL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:22 | / | / | / |
| 21 | R/W | 0x0 | RX_SYNC_EN_START The bit takes effect only when RX_SYNC_EN is set to 1. I2S0/I2S1 Synchronize Enable Start. 0: Disabled 1: Enabled |
| 20 | R/W | 0x0 | RX_SYNC_EN I2S RX Synchronize Enable 0: Disabled 1: Enabled |
| 19 | / | / | / |
| 18 | R/W | 0x1 | BCLK_OUT Bit Clock Direction Select 0: Input 1: Output |
| 17 | R/W | 0x1 | LRCK_OUT LRCK Direction Select 0: Input 1: Output |
| 16:12 | / | / | / |
| 11 | R/W | 0x0 | DOUT3_EN Data3 Output Enable 0: Disabled, Hi-Z State 1: Enabled |

| Offset: 0x0000 | | | Register Name: I2S/PCM_CTL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 10 | R/W | 0x0 | DOUT2_EN Data2 Output Enable 0: Disabled, Hi-Z State 1: Enabled |
| 9 | R/W | 0x0 | DOUT1_EN Data1 Output Enable 0: Disabled, Hi-Z State 1: Enabled |
| 8 | R/W | 0x0 | DOUT0_EN Data0 Output Enable 0: Disabled, Hi-Z State 1: Enabled |
| 7 | / | / | / |
| 6 | R/W | 0x0 | OUT_Mute Data Output Mute Enable 0: Normal Transfer 1: Force DOUT to output 0 |
| 5:4 | R/W | 0x0 | MODE_SEL Mode Selection 00: PCM Mode (offset 0: Long Frame; offset 1: Short Frame) 01: Left Mode (offset 0: LJ Mode; offset 1: I2S Mode) 10: Right-Justified Mode 11: Reserved |
| 3 | R/W | 0x0 | LOOP Loopback Test 0: Normal Mode 1: Loopback Test When set to '1', the bit indicates that the DOUT is connected to the DIN. |
| 2 | R/W | 0x0 | TXEN Transmitter Block Enable 0: Disabled 1: Enabled |
| 1 | R/W | 0x0 | RXEN Receiver Block Enable 0: Disabled 1: Enabled |
| 0 | R/W | 0x0 | GEN Globe Enable 0: Disabled 1: Enabled |

8.1.6.2 0x0004 I2S/PCM Format Register 0 (Default Value: 0x0000_0033)

| Offset: 0x0004 | | | Register Name: I2S/PCM_FMT0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31 | / | / | / |
| 30 | R/W | 0x0 | LRCK_WIDTH LRCK Width (only applies to the PCM mode) 0: LRCK = 1 BCLK Width (Short Frame) 1: LRCK = 2 BCLK Width (Long Frame) |
| 29:20 | / | / | / |
| 19 | R/W | 0x0 | LRCK_POLARITY In I2S/Left-Justified/Right-Justified mode: 0: Left Channel when LRCK is low. 1: Left channel when LRCK is high. In PCM mode: 0: PCM LRCK asserted at the negative edge. 1: PCM LRCK asserted at the positive edge. |
| 18 | / | / | / |
| 17:8 | R/W | 0x0 | LRCK_PERIOD It is used to program the number of BCLKs per channel of the sample frame. This value is interpreted as follows. PCM mode: Number of BCLKs within (Left + Right) channel width. I2S/Left-Justified/Right-Justified mode: Number of BCLKs within each channel width (Left or Right). For example: N = 7: 8 BCLKs width ... N = 1023: 1024 BCLKs width |
| 7 | R/W | 0x0 | BCLK_POLARITY 0: Normal mode, DOUT drives data at negative edge 1: Invert mode, DOUT drives data at positive edge |
| 6:4 | R/W | 0x3 | SR Sample Resolution 000: Reserved 001: 8-bit 010: 12-bit 011: 16-bit 100: 20-bit 101: 24-bit 110: 28-bit 111: 32-bit |

| Offset: 0x0004 | | | Register Name: I2S/PCM_FMT0 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 3 | R/W | 0x0 | <p>EDGE_TRANSFER Edge Transfer 0: DOUT drives data and DIN sample data at the different BCLK edge 1: DOUT drives data and DIN sample data at the same BCLK edge</p> <p>BCLK_POLARITY = 0, EDGE_TRANSFER = 0, DIN sample data at positive edge; BCLK_POLARITY = 0, EDGE_TRANSFER = 1, DIN sample data at negative edge; BCLK_POLARITY = 1, EDGE_TRANSFER = 0, DIN sample data at negative edge; BCLK_POLARITY = 1, EDGE_TRANSFER = 1, DIN sample data at positive edge.</p> |
| 2:0 | R/W | 0x3 | <p>SW Slot Width Select 000: Reserved 001: 8-bit 010: 12-bit 011: 16-bit 100: 20-bit 101: 24-bit 110: 28-bit 111: 32-bit</p> |

8.1.6.3 0x0008 I2S/PCM Format Register 1 (Default Value: 0x0000_0030)

| Offset: 0x0008 | | | Register Name: I2S/PCM_FMT1 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7 | R/W | 0x0 | <p>RX MLS MSB/LSB First Select 0: MSB First 1: LSB First</p> |
| 6 | R/W | 0x0 | <p>TX MLS MSB/LSB First Select 0: MSB First 1: LSB First</p> |

| Offset: 0x0008 | | | Register Name: I2S/PCM_FMT1 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 5:4 | R/W | 0x3 | <p>SEXT</p> <p>Sign Extend in Slot [Sample Resolution < Slot Width]</p> <p>00: Zeros or audio gain padding at LSB position</p> <p>01: Sign extension at MSB position</p> <p>10: Reserved</p> <p>11: Transfer 0 after each sample in each Slot</p> |
| 3:2 | R/W | 0x0 | <p>RX_PDM</p> <p>PCM Data Mode</p> <p>00: Linear PCM</p> <p>01: Reserved</p> <p>10: 8-bit u-law</p> <p>11: 8-bit A-law</p> |
| 1:0 | R/W | 0x0 | <p>TX_PDM</p> <p>PCM Data Mode</p> <p>00: Linear PCM</p> <p>01: Reserved</p> <p>10: 8-bit u-law</p> <p>11: 8-bit A-law</p> |

8.1.6.4 0x000C I2S/PCM Interrupt Status Register (Default Value: 0x0000_0010)

| Offset: 0x000C | | | Register Name: I2S/PCMISTA |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:7 | / | / | / |
| 6 | R/W1C | 0x0 | <p>TXU_INT</p> <p>TXFIFO Underrun Pending Interrupt</p> <p>0: No pending interrupt</p> <p>1: TXFIFO underrun pending interrupt</p> <p>Write '1' to clear this interrupt.</p> |
| 5 | R/W1C | 0x0 | <p>TXO_INT</p> <p>TXFIFO Overrun Pending Interrupt</p> <p>0: No pending interrupt</p> <p>1: TXFIFO overrun pending interrupt</p> <p>Write '1' to clear this interrupt.</p> |
| 4 | R | 0x1 | <p>TXE_INT</p> <p>TXFIFO Empty Pending Interrupt</p> <p>0: No pending IRQ</p> <p>1: TXFIFO empty pending interrupt when data in TXFIFO are less than TX trigger level</p> |
| 3 | / | / | / |

| Offset: 0x000C | | | Register Name: I2S/PCMISTA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 2 | R/W1C | 0x0 | RXU_INT RXFIFO Underrun Pending Interrupt 0: No pending interrupt 1: RXFIFO underrun pending interrupt Write '1' to clear this interrupt. |
| 1 | R/W1C | 0x0 | RXO_INT RXFIFO Overrun Pending Interrupt 0: No pending IRQ 1: RXFIFO overrun pending IRQ Write '1' to clear this interrupt. |
| 0 | R/W | 0x0 | RXA_INT RXFIFO Data Available Pending Interrupt 0: No pending IRQ 1: Data available pending IRQ when data in RXFIFO are more than RX trigger level |

8.1.6.5 0x0010 I2S/PCM RXFIFO Register (Default Value: 0x0000_0000)

| Offset: 0x0010 | | | Register Name: I2S/PCM_RXFIFO |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R | 0x0 | RX_DATA RX Sample The host can get one sample by reading this register. The left channel sample data is first and then the right channel sample. |

8.1.6.6 0x0014 I2S/PCM FIFO Control Register (Default Value: 0x0004_00F0)

| Offset: 0x0014 | | | Register Name: I2S/PCM_FCTL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31 | R/W | 0x0 | HUB_EN Audio Hub Enable The bit takes effect only when TXEN is set to 1. I2S0/I2S1 Hub Enable. 0: Disabled 1: Enabled |
| 30:26 | / | / | / |
| 25 | R/WAC | 0x0 | FTX Write '1' to flush TXFIFO, self clear to '0'. |
| 24 | R/WAC | 0x0 | FRX Write '1' to flush RXFIFO, self clear to '0'. |
| 23:19 | / | / | / |

| Offset: 0x0014 | | | Register Name: I2S/PCM_FCTL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 18:12 | R/W | 0x40 | <p>TXTL TXFIFO Empty Trigger Level Interrupt and DMA request trigger level for TXFIFO normal condition. Trigger Level = TXTL</p> |
| 11:10 | / | / | / |
| 9:4 | R/W | 0xF | <p>RXTL RXFIFO Trigger Level Interrupt and DMA request trigger level for RXFIFO normal condition. Trigger Level = RXTL + 1</p> |
| 3 | / | / | / |
| 2 | R/W | 0x0 | <p>TXIM TXFIFO Input Mode (Mode 0, 1) 0: Valid data at the MSB of TXFIFO register 1: Valid data at the LSB of TXFIFO register Example for 20-bit transmitted audio sample: Mode 0: TXFIFO[31:0] = {APB_WDATA[31:12], 12'h0} Mode 1: TXFIFO[31:0] = {APB_WDATA[19:0], 12'h0}</p> |
| 1:0 | R/W | 0x0 | <p>RXOM RXFIFO Output Mode (Mode 0, 1, 2, 3) 00: Expanding '0' at LSB of RXFIFO register 01: Expanding received sample sign bit at MSB of RXFIFO register 10: Truncating received samples at high half-word of RXFIFO register and low half-word of RXFIFO register is filled by '0' 11: Truncating received samples at low half-word of RXFIFO register and high half-word of RXFIFO register is expanded by its sign bit Example for 20-bit received audio sample: Mode 0: APB_RDATA[31:0] = {RXFIFO[31:12], 12'h0} Mode 1: APB_RDATA[31:0] = {12{RXFIFO[31]}, RXFIFO[31:12]} Mode 2: APB_RDATA [31:0] = {RXFIFO[31:16], 16'h0} Mode 3: APB_RDATA[31:0] = {16{RXFIFO[31]}, RXFIFO[31:16]}</p> |

8.1.6.7 0x0018 I2S/PCM FIFO Status Register (Default Value: 0x1080_0080)

| Offset: 0x0018 | | | Register Name: I2S/PCM_FSTA |
|----------------|------------|-------------|-----------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:29 | / | / | / |

| Offset: 0x0018 | | | Register Name: I2S/PCM_FSTA |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 28 | R | 0x1 | TXE TXFIFO Empty 0: No room for new sample in TXFIFO 1: More than one room for new sample in TXFIFO (>= 1 Word) |
| 27:24 | / | / | / |
| 23:16 | R | 0x80 | TXE_CNT TXFIFO Empty Space Word Counter |
| 15:9 | / | / | / |
| 8 | R | 0x0 | RXA RXFIFO Available 0: No available data in RXFIFO 1: More than one sample in RXFIFO (>= 1 Word) |
| 7 | R | 0x1 | PLACE HOLDER NO Meaning. |
| 6:0 | R | 0x0 | RXA_CNT RXFIFO available sample word counter |

8.1.6.8 0x001C I2S/PCM DMA & Interrupt Control Register (Default Value: 0x0000_0000)

| Offset: 0x001C | | | Register Name: I2S/PCM_INT |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7 | R/W | 0x0 | TX_DRQ TXFIFO Empty DRQ Enable 0: Disabled 1: Enabled |
| 6 | R/W | 0x0 | TXUI_EN TXFIFO Underrun Interrupt Enable 0: Disabled 1: Enabled |
| 5 | R/W | 0x0 | TXOI_EN TXFIFO Overrun Interrupt Enable 0: Disabled 1: Enabled When set to '1', an interrupt happens when writing new audio data if TXFIFO is full. |
| 4 | R/W | 0x0 | TXEI_EN TXFIFO Empty Interrupt Enable 0: Disabled 1: Enabled |

| Offset: 0x001C | | | Register Name: I2S/PCM_INT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 3 | R/W | 0x0 | <p>RX_DRQ RXFIFO Data Available DRQ Enable 0: Disabled 1: Enabled When set to '1', RXFIFO DMA request line is asserted if data is available in RXFIFO.</p> |
| 2 | R/W | 0x0 | <p>RXUI_EN RXFIFO Underrun Interrupt Enable 0: Disabled 1: Enabled</p> |
| 1 | R/W | 0x0 | <p>RXOI_EN RXFIFO Overrun Interrupt Enable 0: Disabled 1: Enabled</p> |
| 0 | R/W | 0x0 | <p>RXAI_EN RXFIFO Data Available Interrupt Enable 0: Disabled 1: Enabled</p> |

8.1.6.9 0x0020 I2S/PCM TXFIFO Register (Default Value: 0x0000_0000)

| Offset: 0x0020 | | | Register Name: I2S/PCM_TXFIFO |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | W | 0x0 | <p>TX_DATA TX Sample Transmitting left, right channel sample data should be written to this register one by one. The left channel sample data is first and then the right channel sample.</p> |

8.1.6.10 0x0024 I2S/PCM Clock Divide Register (Default Value: 0x0000_0000)

| Offset: 0x0024 | | | Register Name: I2S/PCM_CLKD |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:9 | / | / | / |
| 8 | R/W | 0x0 | <p>MCLKO_EN 0: Disable MCLK Output 1: Enable MCLK Output Note: Whether in slave or master mode, when this bit is set to '1', MCLK should be output.</p> |

| Offset: 0x0024 | | | Register Name: I2S/PCM_CLKD |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 7:4 | R/W | 0x0 | BCLKDIV BCLK Divide Ratio from PLL_AUDIO 0000: Reserved 0001: Divide by 1 0010: Divide by 2 0011: Divide by 4 0100: Divide by 6 0101: Divide by 8 0110: Divide by 12 0111: Divide by 16 1000: Divide by 24 1001: Divide by 32 1010: Divide by 48 1011: Divide by 64 1100: Divide by 96 1101: Divide by 128 1110: Divide by 176 1111: Divide by 192 |
| 3:0 | R/W | 0x0 | MCLKDIV MCLK Divide Ratio from PLL_AUDIO 0000: Reserved 0001: Divide by 1 0010: Divide by 2 0011: Divide by 4 0100: Divide by 6 0101: Divide by 8 0110: Divide by 12 0111: Divide by 16 1000: Divide by 24 1001: Divide by 32 1010: Divide by 48 1011: Divide by 64 1100: Divide by 96 1101: Divide by 128 1110: Divide by 176 1111: Divide by 192 |

8.1.6.11 0x0028 I2S/PCM TX Sample Counter Register (Default Value: 0x0000_0000)

| Offset: 0x0028 | | | Register Name: I2S/PCM_TXCNT |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R/W | 0x0 | <p>TX_CNT TX Sample Counter</p> <p>The audio sample number of sending into TXFIFO. When one sample is put into TXFIFO by DMA or by host IO, the TX sample counter register increases by one. The TX sample counter register can be set to any initial value at any time. After being updated by the initial value, the counter register should count on the base of this initial value.</p> |

8.1.6.12 0x002C I2S/PCM RX Sample Counter Register (Default Value: 0x0000_0000)

| Offset: 0x002C | | | Register Name: I2S/PCM_RXCNT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R/W | 0x0 | <p>RX_CNT RX Sample Counter</p> <p>The audio sample number of writing into RXFIFO. When one sample is written by Digital Audio Engine, the RX sample counter register increases by one. The RX sample counter register can be set to any initial value at any time. After being updated by the initial value, the counter register should count on the base of this initial value.</p> |

8.1.6.13 0x0030 I2S/PCM Channel Configuration Register (Default Value: 0x0000_0000)

| Offset: 0x0030 | | | Register Name: I2S/PCM_CHCFG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:10 | / | / | / |
| 9 | R/W | 0x0 | <p>TX_SLOT_HIZ</p> <p>0: Normal mode for the last half-cycle of BCLK in the slot</p> <p>1: Turn to Hi-Z state for the last half-cycle of BCLK in the slot</p> |
| 8 | R/W | 0x0 | <p>TX_STATE</p> <p>0: Transfer level 0 in non-transferring slot</p> <p>1: Turn to Hi-Z State (TDM) in non-transferring slot</p> |

| Offset: 0x0030 | | | Register Name: I2S/PCM_CHCFG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 7:4 | R/W | 0x0 | <p>RX_SLOT_NUM RX Channel/Slot number between CPU/DMA and RXFIFO 0000: 1 channel or slot ... 0111: 8 channels or slots 1000: 9 channels or slots ... 1111: 16 channels or slots</p> |
| 3:0 | R/W | 0x0 | <p>TX_SLOT_NUM TX Channel/Slot number between CPU/DMA and TXFIFO 0000: 1 channel or slot ... 0111: 8 channels or slots 1000: 9 channels or slots ... 1111: 16 channels or slots</p> |

8.1.6.14 0x0034 I2S/PCM TX0 Channel Select Register (Default Value: 0x0000_0000)

| Offset: 0x0034 | | | Register Name: I2S/PCM_TX0CHSEL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:22 | / | / | / |
| 21:20 | R/W | 0x0 | <p>TX0_OFFSET TX0 Offset Tune (TX0 Data offset to LRCK) 0: No offset n: Data is offset by n BCLKs to LRCK</p> |
| 19:16 | R/W | 0x0 | <p>TX0_CHSEL TX0 Channel (Slot) Number Select for Each Output 0000: 1 channel or slot ... 0111: 8 channels or slots 1000: 9 channels or slots ... 1111: 16 channels or slots</p> |
| 15:0 | R/W | 0x0 | <p>TX0_CHEN TX0 Channel (Slot) Enable The bit[15:0] refer to Slot [15:0]. When one or more slots are disabled, the affected slots are set to the disable state. 0: Disabled 1: Enabled</p> |

8.1.6.15 0x0038 I2S/PCM TX1 Channel Select Register (Default Value: 0x0000_0000)

| Offset: 0x0038 | | | Register Name: I2S/PCM_TX1CHSEL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:22 | / | / | / |
| 21:20 | R/W | 0x0 | TX1_OFFSET TX1 Offset Tune (TX1 Data offset to LRCK) 0: No offset n: Data is offset by n BCLKs to LRCK |
| 19:16 | R/W | 0x0 | TX1_CHSEL TX1 Channel (Slot) Number Select for Each Output 0000: 1 channel or slot ... 0111: 8 channels or slots 1000: 9 channels or slots ... 1111: 16 channels or slots |
| 15:0 | R/W | 0x0 | TX1_CHEN TX1 Channel (Slot) Enable The bit[15:0] refers to Slot [15:0]. When one or more slots are disabled, the affected slots are set to the disable state. 0: Disabled 1: Enabled |

8.1.6.16 0x003C I2S/PCM TX2 Channel Select Register (Default Value: 0x0000_0000)

| Offset: 0x003C | | | Register Name: I2S/PCM_TX2CHSEL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:22 | / | / | / |
| 21:20 | R/W | 0x0 | TX2_OFFSET TX2 Offset Tune (TX2 Data offset to LRCK) 0: No offset n: Data is offset by n BCLKs to LRCK |
| 19:16 | R/W | 0x0 | TX2_CHSEL TX2 Channel (Slot) Number Select for Each Output 0000: 1 channel or slot ... 0111: 8 channels or slots 1000: 9 channels or slots ... 1111: 16 channels or slots |

| Offset: 0x003C | | | Register Name: I2S/PCM_TX2CHSEL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 15:0 | R/W | 0x0 | <p>TX2_CHEN</p> <p>TX2 Channel (Slot) Enable</p> <p>The bit[15:0] refers to Slot [15:0]. When one or more slots are disabled, the affected slots are set to the disable state.</p> <p>0: Disabled</p> <p>1: Enabled</p> |

8.1.6.17 0x0040 I2S/PCM TX3 Channel Select Register (Default Value: 0x0000_0000)

| Offset: 0x0040 | | | Register Name: I2S/PCM_TX3CHSEL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:22 | / | / | / |
| 21:20 | R/W | 0x0 | <p>TX3_OFFSET</p> <p>TX3 Offset Tune (TX3 Data offset to LRCK)</p> <p>0: No offset</p> <p>n: Data is offset by n BCLKs to LRCK</p> |
| 19:16 | R/W | 0x0 | <p>TX3_CHSEL</p> <p>TX3 Channel (Slot) Number Select for Each Output</p> <p>0000: 1 channel or slot</p> <p>...</p> <p>0111: 8 channels or slots</p> <p>1000: 9 channels or slots</p> <p>...</p> <p>1111: 16 channels or slots</p> |
| 15:0 | R/W | 0x0 | <p>TX3_CHEN</p> <p>TX3 Channel (Slot) Enable</p> <p>The bit[15:0] refers to Slot [15:0]. When one or more slots are disabled, the affected slots are set to the disable state.</p> <p>0: Disabled</p> <p>1: Enabled</p> |

8.1.6.18 0x0044 I2S/PCM TX0 Channel Mapping0 Register 0 (Default Value: 0x0000_0000)

| Offset: 0x0044 | | | Register Name: I2S/PCM_TX0CHMAP0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | R/W | 0x0 | TX0_CH15_MAP TX0 Channel 15 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 27:24 | R/W | 0x0 | TX0_CH14_MAP TX0 Channel 14 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 23:20 | R/W | 0x0 | TX0_CH13_MAP TX0 Channel 13 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 19:16 | R/W | 0x0 | TX0_CH12_MAP TX0 Channel 12 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 15:12 | R/W | 0x0 | TX0_CH11_MAP TX0 Channel 11 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

| Offset: 0x0044 | | | Register Name: I2S/PCM_TX0CHMAP0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 11:8 | R/W | 0x0 | TX0_CH10_MAP TX0 Channel 10 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 7:4 | R/W | 0x0 | TX0_CH9_MAP TX0 Channel 9 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 3:0 | R/W | 0x0 | TX0_CH8_MAP TX0 Channel 8 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

8.1.6.19 0x0048 I2S/PCM TX0 Channel Mapping1 Register 1 (Default Value: 0x0000_0000)

| Offset: 0x0048 | | | Register Name: I2S/PCM_TX0CHMAP1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | R/W | 0x0 | TX0_CH7_MAP TX0 Channel 7 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

| Offset: 0x0048 | | | Register Name: I2S/PCM_TX0CHMAP1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 27:24 | R/W | 0x0 | TX0_CH6_MAP TX0 Channel 6 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 23:20 | R/W | 0x0 | TX0_CH5_MAP TX0 Channel 5 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 19:16 | R/W | 0x0 | TX0_CH4_MAP TX0 Channel 4 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 15:12 | R/W | 0x0 | TX0_CH3_MAP TX0 Channel 3 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 11:8 | R/W | 0x0 | TX0_CH2_MAP TX0 Channel 2 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

| Offset: 0x0048 | | | Register Name: I2S/PCM_TX0CHMAP1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 7:4 | R/W | 0x0 | TX0_CH1_MAP TX0 Channel 1 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 3:0 | R/W | 0x0 | TX0_CHO_MAP TX0 Channel 0 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

8.1.6.20 0x004C I2S/PCM TX1 Channel Mapping0 Register 0 (Default Value: 0x0000_0000)

| Offset: 0x004C | | | Register Name: I2S/PCM_TX1CHMAP0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | R/W | 0x0 | TX1_CH15_MAP TX1 Channel 15 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 27:24 | R/W | 0x0 | TX1_CH14_MAP TX1 Channel 14 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

| Offset: 0x004C | | | Register Name: I2S/PCM_TX1CHMAP0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 23:20 | R/W | 0x0 | TX1_CH13_MAP TX1 Channel 13 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 19:16 | R/W | 0x0 | TX1_CH12_MAP TX1 Channel 12 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 15:12 | R/W | 0x0 | TX1_CH11_MAP TX1 Channel 11 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 11:8 | R/W | 0x0 | TX1_CH10_MAP TX1 Channel 10 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 7:4 | R/W | 0x0 | TX1_CH9_MAP TX1 Channel 9 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

| Offset: 0x004C | | | Register Name: I2S/PCM_TX1CHMAP0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 3:0 | R/W | 0x0 | TX1_CH8_MAP TX1 Channel 8 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

8.1.6.21 0x0050 I2S/PCM TX1 Channel Mapping1 Register 1 (Default Value: 0x0000_0000)

| Offset: 0x0050 | | | Register Name: I2S/PCM_TX1CHMAP1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | R/W | 0x0 | TX1_CH7_MAP TX1 Channel 7 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 27:24 | R/W | 0x0 | TX1_CH6_MAP TX1 Channel 6 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 23:20 | R/W | 0x0 | TX1_CH5_MAP TX1 Channel 5 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

| Offset: 0x0050 | | | Register Name: I2S/PCM_TX1CHMAP1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 19:16 | R/W | 0x0 | TX1_CH4_MAP TX1 Channel 4 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 15:12 | R/W | 0x0 | TX1_CH3_MAP TX1 Channel 3 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 11:8 | R/W | 0x0 | TX1_CH2_MAP TX1 Channel 2 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 7:4 | R/W | 0x0 | TX1_CH1_MAP TX1 Channel 1 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 3:0 | R/W | 0x0 | TX1_CHO_MAP TX1 Channel 0 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

8.1.6.22 0x0054 I2S/PCM TX2 Channel Mapping0 Register 0 (Default Value: 0x0000_0000)

| Offset: 0x0054 | | | Register Name: I2S/PCM_TX2CHMAP0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | R/W | 0x0 | TX2_CH15_MAP TX2 Channel 15 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 27:24 | R/W | 0x0 | TX2_CH14_MAP TX2 Channel 14 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 23:20 | R/W | 0x0 | TX2_CH13_MAP TX2 Channel 13 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 19:16 | R/W | 0x0 | TX2_CH12_MAP TX2 Channel 12 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 15:12 | R/W | 0x0 | TX2_CH11_MAP TX2 Channel 11 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

| Offset: 0x0054 | | | Register Name: I2S/PCM_TX2CHMAP0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 11:8 | R/W | 0x0 | TX2_CH10_MAP TX2 Channel 10 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 7:4 | R/W | 0x0 | TX2_CH9_MAP TX2 Channel 9 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 3:0 | R/W | 0x0 | TX2_CH8_MAP TX2 Channel 8 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

8.1.6.23 0x0058 I2S/PCM TX2 Channel Mapping1 Register 1 (Default Value: 0x0000_0000)

| Offset: 0x0058 | | | Register Name: I2S/PCM_TX2CHMAP1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | R/W | 0x0 | TX2_CH7_MAP TX2 Channel 7 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

| Offset: 0x0058 | | | Register Name: I2S/PCM_TX2CHMAP1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 27:24 | R/W | 0x0 | TX2_CH6_MAP TX2 Channel 6 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 23:20 | R/W | 0x0 | TX2_CH5_MAP TX2 Channel 5 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 19:16 | R/W | 0x0 | TX2_CH4_MAP TX2 Channel 4 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 15:12 | R/W | 0x0 | TX2_CH3_MAP TX2 Channel 3 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 11:8 | R/W | 0x0 | TX2_CH2_MAP TX2 Channel 2 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

| Offset: 0x0058 | | | Register Name: I2S/PCM_TX2CHMAP1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 7:4 | R/W | 0x0 | TX2_CH1_MAP TX2 Channel 1 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 3:0 | R/W | 0x0 | TX2_CHO_MAP TX2 Channel 0 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

8.1.6.24 0x005C I2S/PCM TX3 Channel Mapping0 Register 0 (Default Value: 0x0000_0000)

| Offset: 0x005C | | | Register Name: I2S/PCM_TX3CHMAP0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | R/W | 0x0 | TX3_CH15_MAP TX3 Channel 15 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 27:24 | R/W | 0x0 | TX3_CH14_MAP TX3 Channel 14 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

| Offset: 0x005C | | | Register Name: I2S/PCM_TX3CHMAP0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 23:20 | R/W | 0x0 | TX3_CH13_MAP TX3 Channel 13 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 19:16 | R/W | 0x0 | TX3_CH12_MAP TX3 Channel 12 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 15:12 | R/W | 0x0 | TX3_CH11_MAP TX3 Channel 11 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 11:8 | R/W | 0x0 | TX3_CH10_MAP TX3 Channel 10 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 7:4 | R/W | 0x0 | TX3_CH9_MAP TX3 Channel 9 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

| Offset: 0x005C | | | Register Name: I2S/PCM_TX3CHMAP0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 3:0 | R/W | 0x0 | TX3_CH8_MAP TX3 Channel 8 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

8.1.6.25 0x0060 I2S/PCM TX3 Channel Mapping1 Register 1 (Default Value: 0x0000_0000)

| Offset: 0x0060 | | | Register Name: I2S/PCM_TX3CHMAP1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | R/W | 0x0 | TX3_CH7_MAP TX3 Channel 7 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 27:24 | R/W | 0x0 | TX3_CH6_MAP TX3 Channel 6 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 23:20 | R/W | 0x0 | TX3_CH5_MAP TX3 Channel 5 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

| Offset: 0x0060 | | | Register Name: I2S/PCM_TX3CHMAP1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 19:16 | R/W | 0x0 | TX3_CH4_MAP TX3 Channel 4 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 15:12 | R/W | 0x0 | TX3_CH3_MAP TX3 Channel 3 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 11:8 | R/W | 0x0 | TX3_CH2_MAP TX3 Channel 2 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 7:4 | R/W | 0x0 | TX3_CH1_MAP TX3 Channel 1 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 3:0 | R/W | 0x0 | TX3_CHO_MAP TX3 Channel 0 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

8.1.6.26 0x0064 I2S/PCM RX Channel Select Register (Default Value: 0x0000_0000)

| Offset: 0x0064 | | | Register Name: I2S/PCM_RXCHSEL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:22 | / | / | / |
| 21:20 | R/W | 0x0 | RX_OFFSET RX Offset Tune (RX Data offset to LRCK) 0: No offset n: Data is offset by n BCLKs to LRCK |
| 19:16 | R/W | 0x0 | RX_CHSEL RX Channel (Slot) Number Select for Input 0000: 1 channel or slot ... 0111: 8 channels or slots 1000: 9 channels or slots ... 1111: 16 channels or slots |
| 15:0 | / | / | / |

8.1.6.27 0x0068 I2S/PCM RX Channel Mapping Register0 (Default Value: 0x0000_0000)

| Offset: 0x0068 | | | Register Name: I2S/PCM_RXCHMAP0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:30 | / | / | / |
| 29:28 | R/W | 0x0 | RX_CH15_SELECT RX Channel 15 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3 |
| 27:24 | R/W | 0x0 | RX_CH15_MAP RX Channel 15 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 23:22 | / | / | / |

| Offset: 0x0068 | | | Register Name: I2S/PCM_RXCHMAP0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 21:20 | R/W | 0x0 | RX_CH14_SELECT RX Channel 14 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3 |
| 19:16 | R/W | 0x0 | RX_CH14_MAP RX Channel 14 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 15:14 | / | / | / |
| 13:12 | R/W | 0x0 | RX_CH13_SELECT RX Channel 13 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3 |
| 11:8 | R/W | 0x0 | RX_CH13_MAP RX Channel 13 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 7:6 | / | / | / |
| 5:4 | R/W | 0x0 | RX_CH12_SELECT RX Channel 12 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3 |

| Offset: 0x0068 | | | Register Name: I2S/PCM_RXCHMAP0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 3:0 | R/W | 0x0 | RX_CH12_MAP RX Channel 12 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

8.1.6.28 0x006C I2S/PCM RX Channel Mapping Register1 (Default Value: 0x0000_0000)

| Offset: 0x006C | | | Register Name: I2S/PCM_RXCHMAP1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:30 | / | / | / |
| 29:28 | R/W | 0x0 | RX_CH11_SELECT RX Channel 11 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3 |
| 27:24 | R/W | 0x0 | RX_CH11_MAP RX Channel 11 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 23:22 | / | / | / |
| 21:20 | R/W | 0x0 | RX_CH10_SELECT RX Channel 10 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3 |

| Offset: 0x006C | | | Register Name: I2S/PCM_RXCHMAP1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 19:16 | R/W | 0x0 | RX_CH10_MAP RX Channel 10 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 15:14 | / | / | / |
| 13:12 | R/W | 0x0 | RX_CH9_SELECT RX Channel 9 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3 |
| 11:8 | R/W | 0x0 | RX_CH9_MAP RX Channel 9 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 7:6 | / | / | / |
| 5:4 | R/W | 0x0 | RX_CH8_SELECT RX Channel 8 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3 |
| 3:0 | R/W | 0x0 | RX_CH8_MAP RX Channel 8 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

8.1.6.29 0x0070 I2S/PCM RX Channel Mapping Register2 (Default Value: 0x0000_0000)

| Offset: 0x0070 | | | Register Name: I2S/PCM_RXCHMAP2 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:30 | / | / | / |
| 29:28 | R/W | 0x0 | RX_CH7_SELECT RX Channel 7 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3 |
| 27:24 | R/W | 0x0 | RX_CH7_MAP RX Channel 7 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 23:22 | / | / | / |
| 21:20 | R/W | 0x0 | RX_CH6_SELECT RX Channel 6 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3 |
| 19:16 | R/W | 0x0 | RX_CH6_MAP RX Channel 6 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 15:14 | / | / | / |
| 13:12 | R/W | 0x0 | RX_CH5_SELECT RX Channel 5 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3 |

| Offset: 0x0070 | | | Register Name: I2S/PCM_RXCHMAP2 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 11: 8 | R/W | 0x0 | RX_CH5_MAP RX Channel 5 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |
| 7:6 | / | / | / |
| 5:4 | R/W | 0x0 | RX_CH4_SELECT RX Channel 4 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3 |
| 3:0 | R/W | 0x0 | RX_CH4_MAP RX Channel 4 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample |

8.1.6.30 0x0074 I2S/PCM RX Channel Mapping Register3 (Default Value: 0x0000_0000)

| Offset: 0x0074 | | | Register Name: I2S/PCM_RXCHMAP3 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:30 | / | / | / |
| 29:28 | R/W | 0x0 | RX_CH3_SELECT RX Channel 3 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3 |

| Offset: 0x0074 | | | Register Name: I2S/PCM_RXCHMAP3 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 27:24 | R/W | 0x0 | <p>RX_CH3_MAP RX Channel 3 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample</p> |
| 23:22 | / | / | / |
| 21:20 | R/W | 0x0 | <p>RX_CH2_SELECT RX Channel 2 Select 00: SDIO 01: SDI1 10: SDI2 11: SDI3</p> |
| 19:16 | R/W | 0x0 | <p>RX_CH2_MAP RX Channel 2 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample</p> |
| 15:14 | / | / | / |
| 13:12 | R/W | 0x0 | <p>RX_CH1_SELECT RX Channel 1 Select 00: SDIO 01: SDI1 10: SDI2 11: SDI3</p> |
| 11:8 | R/W | 0x0 | <p>RX_CH1_MAP RX Channel 1 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample</p> |
| 7:6 | / | / | / |

| Offset: 0x0074 | | | Register Name: I2S/PCM_RXCHMAP3 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 5:4 | R/W | 0x0 | <p>RX_CH0_SELECT RX Channel 0 Select 00: SDIO 01: SDI1 10: SDI2 11: SDI3</p> |
| 3:0 | R/W | 0x0 | <p>RX_CH0_MAP RX Channel 0 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample</p> |

8.2 DMIC

8.2.1 Overview

The DMIC controller supports one 8-channel digital microphone interface and can output 128 fs or 64 fs (fs = ADC sample rate).

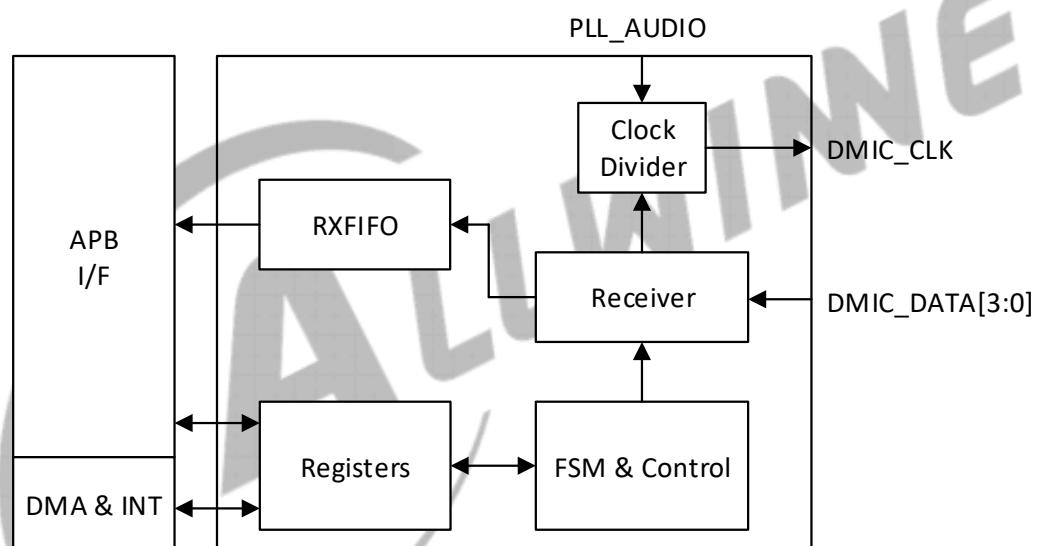
The DMIC controller includes the following features:

- Supports up to 8 channels
- Sample rate from 8 kHz to 48 kHz

8.2.2 Block Diagram

The following figure shows a block diagram of the DMIC.

Figure 8-9 DMIC Block Diagram



8.2.3 Functional Description

8.2.3.1 External Signals

The following table describes the external signals of DMIC.

Table 8-3 DMIC External Signals

| Signal | Description | Type |
|------------|---------------------------------|------|
| DMIC_CLK | Digital Microphone Clock Output | O |
| DMIC_DATA0 | Digital Microphone Data Input | I |
| DMIC_DATA1 | Digital Microphone Data Input | I |
| DMIC_DATA2 | Digital Microphone Data Input | I |
| DMIC_DATA3 | Digital Microphone Data Input | I |

8.2.3.2 Clock Sources

The following table describes the clock source for DMIC. For clock setting, configurations, and gating information, refer to section 3.4 Clock Controller Unit (CCU).

Table 8-4 DMIC Clock Sources

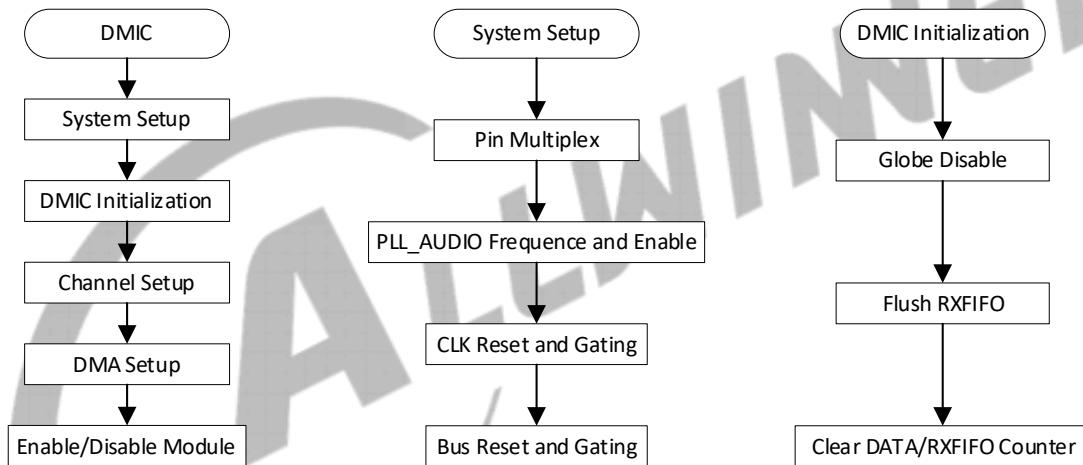
| Clock Sources | Description |
|---------------|--|
| PLL_AUDIO | 24.576 MHz or 22.5792 MHz generated by PLL_AUDIO to produce 48 KHz or 44.1 KHz sample frequency. |

8.2.4 Operation Modes

The software operation of the DMIC is divided into five steps: system setup, DMIC initialization, channel setup, DMA setup, and Enable/Disable module.

The following figure shows the flow chart of the whole operation, the system setup, and the DMIC initialization.

Figure 8-10 DMIC Operation Mode



8.2.4.1 System Setup and DMIC Initialization

The first step in the DMIC initialization is properly programming the GPIO. Because the DMIC port is a multiplex pin. You can find the function in the pin multiplex specification.

The clock source for the DMIC should be followed.

1. disable the PLL_AUDIO through PLL_AUDIOx Control Register[PLL_ENABLE].
2. set up the frequency of the PLL_AUDIO in PLL_AUDIOx Control Register.
3. Enable PLL_AUDIO
4. Enable the DMIC gating through DMIC_CLK_REG when you checkout that the LOCK bit of PLL_AUDIOx Control Register becomes 1.
5. reset and enable the DMIC bus gating by DMIC_BGR_REG.

After the system setup, set up the register of DMIC.

1. Initialize the DMIC. Close the globe enable bit (DMIC_EN[8]), data channel enable bit (DMIC_EN[7:0]) by writing 0 to it.
2. Flush the RXFIFO by writing 1 to DMIC_RXFIFO_CTR[31].
3. Clear the Data/RXFIFO counter by writing 1 to DMIC_RXFIFO_STA, DMIC_CNT.

8.2.4.2 Channel Setup and DMA Setup

You can set up the sample rate, the sample resolution, the over-sample rate, the channel number, the RXFIFO output mode, the RXFIFO trigger level, and so on. The setup of the register can be found in the specification.

The DMIC supports two methods to transfer the data. The most common way is DMA, the setup of DMA can be found in section 3.9 "DMAC". In this module, you just enable the DRQ.

8.2.4.3 Enable and Disable DMIC

Refer to the following steps to enable this function.

1. Enable the data channel enable bit (DMIC_EN[7:0]) by writing 1 to it.
2. enable DMIC by writing 1 to the Globe Enable bit (DMIC_EN[8]).

If you want to disable this function, write 0 to DMIC_EN[8] to disable DMIC.

8.2.5 Register List

| Module Name | Base Address | Comments |
|-------------|--------------|-------------------------------|
| DMIC | 0x02031000 | Digital microphone controller |

| Register Name | Offset | Description |
|---------------------|--------|--|
| DMIC_EN | 0x0000 | DMIC Enable Control Register |
| DMIC_SR | 0x0004 | DMIC Sample Rate Register |
| DMIC_CTR | 0x0008 | DMIC Control Register |
| DMIC_DATA | 0x0010 | DMIC DATA Register |
| DMIC_INTC | 0x0014 | DMIC Interrupt Control Register |
| DMIC_INTS | 0x0018 | DMIC Interrupt Status Register |
| DMIC_RXFIFO_CTR | 0x001C | DMIC RXFIFO Control Register |
| DMIC_RXFIFO_STA | 0x0020 | DMIC RXFIFO Status Register |
| DMIC_CH_NUM | 0x0024 | DMIC Channel Numbers Register |
| DMIC_CH_MAP | 0x0028 | DMIC Channel Mapping Register |
| DMIC_CNT | 0x002C | DMIC Counter Register |
| DATA0_DATA1_VOL_CTR | 0x0030 | DATA0 And DATA1 Volume Control Register |
| DATA2_DATA3_VOL_CTR | 0x0034 | DATA2 And DATA3 Volume Control Register |
| HPF_EN_CTR | 0x0038 | High Pass Filter Enable Control Register |

| Register Name | Offset | Description |
|---------------|--------|--------------------------------|
| HPF_COEF_REG | 0x003C | High Pass Filter Coef Register |
| HPF_GAIN_REG | 0x0040 | High Pass Filter Gain Register |

8.2.6 Register Description

8.2.6.1 0x0000 DMIC Enable Control Register (Default Value: 0x0000_0000)

| Offset: 0x0000 | | | Register Name: DMIC_EN |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:30 | / | / | / |
| 29 | R/W | 0x0 | <p>RX_SYNC_EN_START Audio Subsys RX Synchronize Enable Start Includes Audio codec/I2S0/I2S1/DMIC. The bit takes effect only when RX_SYNC_EN is set to 1. 0: Disabled 1: Enabled</p> |
| 28 | R/W | 0x0 | <p>RX_SYNC_EN DMIC RX Synchronize Enable 0: Disabled 1: Enabled</p> |
| 27:9 | / | / | / |
| 8 | R/W | 0x0 | <p>GLOBE_EN DMIC Globe Enable 0: Disabled 1: Enabled</p> |
| 7 | R/W | 0x0 | <p>DATA3_CHR_EN DATA3 Right Channel Enable 0: Disabled 1: Enabled</p> |
| 6 | R/W | 0x0 | <p>DATA3_CHL_EN DATA3 Left Channel Enable 0: Disabled 1: Enabled</p> |
| 5 | R/W | 0x0 | <p>DATA2_CHR_EN DATA2 Right Channel Enable 0: Disabled 1: Enabled</p> |
| 4 | R/W | 0x0 | <p>DATA2_CHL_EN DATA2 Left Channel Enable 0: Disabled 1: Enabled</p> |

| Offset: 0x0000 | | | Register Name: DMIC_EN |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 3 | R/W | 0x0 | DATA1_CHR_EN DATA1 Right Channel Enable 0: Disabled 1: Enabled |
| 2 | R/W | 0x0 | DATA1_CHL_EN DATA1 Left Channel Enable 0: Disabled 1: Enabled |
| 1 | R/W | 0x0 | DATA0_CHR_EN DATA0 Right Channel Enable 0: Disabled 1: Enabled |
| 0 | R/W | 0x0 | DATA0_CHL_EN DATA0 Left Channel Enable 0: Disabled 1: Enabled |

8.2.6.2 0x0004 DMIC Sample Rate Register (Default Value: 0x0000_0000)

| Offset: 0x0004 | | | Register Name: DMIC_SR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:3 | / | / | / |
| 2:0 | R/W | 0x0 | DMIC_SR Sample Rate of DMIC 000: 48 kHz 010: 24 kHz 100: 12 kHz 110: Reserved 001: 32 kHz 011: 16 kHz 101: 8 kHz 111: Reserved 44.1 kHz/22.05 kHz/11.025 kHz can be supported by Audio PLL Configure Bit. |

8.2.6.3 0x0008 DMIC Control Register (Default Value: 0x0000_0000)

| Offset: 0x0008 | | | Register Name: DMIC_CTR |
|----------------|------------|-------------|-------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:11 | / | / | / |

| Offset: 0x0008 | | | Register Name: DMIC_CTR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 10:9 | R/W | 0x0 | DMICFDT DMIC RXFIFO Delay Time for Writing Data after GLOBE_EN 00: 5 ms 01: 10 ms 10: 20 ms 11: 30 ms |
| 8 | R/W | 0x0 | DMICDFEN DMIC RXFIFO Delay Function for Writing Data after GLOBE_EN 0: Disabled 1: Enabled |
| 7 | R/W | 0x0 | DATA3 Left Data and Right Data Swap Enable 0: Disabled 1: Enabled |
| 6 | R/W | 0x0 | DATA2 Left Data and Right Data Swap Enable 0: Disabled 1: Enabled |
| 5 | R/W | 0x0 | DATA1 Left Data and Right Data Swap Enable 0: Disabled 1: Enabled |
| 4 | R/W | 0x0 | DATA0 Left Data and Right Data Swap Enable 0: Disabled 1: Enabled |
| 3:1 | / | / | / |
| 0 | R/W | 0x0 | DMIC Oversample Rate 0: 128 (Supports 8 kHz to 24 kHz) 1: 64 (Supports 16 kHz to 48 kHz) |

8.2.6.4 0x0010 DMIC DATA Register (Default Value: 0x0000_0000)

| Offset: 0x0010 | | | Register Name: DMIC_DATA |
|----------------|------------|-------------|--------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R | 0x0 | DMIC_DATA |

8.2.6.5 0x0014 DMIC Interrupt Control Register (Default Value: 0x0000_0000)

| Offset: 0x0014 | | | Register Name: DMIC_INTC |
|----------------|------------|-------------|--------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:3 | / | / | / |

| Offset: 0x0014 | | | Register Name: DMIC_INTC |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 2 | R/W | 0x0 | RXFIFO_DRQ_EN DMIC RXFIFO Data Available DRQ Enable 0: Disabled 1: Enabled |
| 1 | R/W | 0x0 | RXFIFO_OVERRUN_IRQ_EN DMIC RXFIFO Overrun IRQ Enable 0: Disabled 1: Enabled |
| 0 | R/W | 0x0 | DATA_IRQ_EN DMIC RXFIFO Data Available IRQ Enable 0: Disabled 1: Enabled |

8.2.6.6 0x0018 DMIC Interrupt Status Register (Default Value: 0x0000_0000)

| Offset: 0x0018 | | | Register Name: DMIC_INTS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:2 | / | / | / |
| 1 | R/W1C | 0x0 | RXFIFO_OVERRUN_IRQ_PENDING DMIC RXFIFO Overrun Pending Interrupt 0: No pending IRQ 1: RXFIFO overrun pending IRQ Writing '1' to clear this interrupt or automatically clear if the interrupt condition fails. |
| 0 | R/W1C | 0x0 | RXFIFO_DATA_IRQ_PENDING DMIC RXFIFO Data Available Pending Interrupt 0: No pending IRQ 1: Data available pending IRQ Writing '1' to clear this interrupt or automatically clear if the interrupt condition fails. |

8.2.6.7 0x001C DMIC RXFIFO Control Register (Default Value: 0x0000_0040)

| Offset: 0x001C | | | Register Name: DMIC_RXFIFO_CTR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31 | R/W1C | 0x0 | DMIC_RXFIFO_FLUSH DMIC RXFIFO Flush Writing '1' to flush RXFIFO, self clear to '0' |
| 30:10 | / | / | / |

| Offset: 0x001C | | | Register Name: DMIC_RXFIFO_CTR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 9 | R/W | 0x0 | <p>RXFIFO_MODE RXFIFO Output Mode (Mode 0, 1) 0: Expanding '0' at LSB of RXFIFO register 1: Expanding received sample sign bit at MSB of RXFIFO register For 24-bit received audio sample: Mode 0: RXDATA[31:0] = {RXFIFO_O[20:0], 11'h0} Mode 1: RXDATA[31:0] = {8{RXFIFO_O[20]}, RXFIFO_O[20:0], 3'h0} For 16-bit received audio sample: Mode 0: RXDATA[31:0] = {RXFIFO_O[20:5], 16'h0} Mode 1: RXDATA[31:0] = {16{RXFIFO_O[20]}, RXFIFO_O[20:5]}</p> |
| 8 | R/W | 0x0 | <p>Sample_Resolution 0: 16-bit 1: 24- bit</p> |
| 7:0 | R/W | 0x40 | <p>RXFIFO_TRG_LEVEL RXFIFO Trigger Level (TRLV[7:0]) Interrupt and DMA request trigger level for DMIC RXFIFO normal condition IRQ/DRQ Generated when WLEVEL > TRLV[7:0] WLEVEL represents the number of valid samples in the DMIC RXFIFO</p> |

8.2.6.8 0x0020 DMIC RXFIFO Status Register (Default Value: 0x0000_0000)

| Offset: 0x0020 | | | Register Name: DMIC_RXFIFO_STA |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:9 | / | / | / |
| 8 | R | 0x1 | Reserved |
| 7:0 | R/W | 0x0 | DMIC_DATA_CNT DMIC RXFIFO Available Sample Word Counter |

8.2.6.9 0x0024 DMIC Channel Numbers Register (Default Value: 0x0000_0001)

| Offset: 0x0024 | | | Register Name: DMIC_CH_NUM |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:3 | / | / | / |
| 2:0 | R/W | 0x1 | DMIC_CH_NUM DMIC enable channel numbers are (N + 1). |

8.2.6.10 0x0028 DMIC Channel Mapping Register (Default Value: 0x7654_3210)

| Offset: 0x0028 | | | Register Name: DMIC_CH_MAP |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | R/W | 0x7 | DMIC_CH7_MAP DMIC Channel 7 Mapping 0000: DATA0 Left Channel 0001: DATA0 Right Channel 0010: DATA1 Left Channel 0011: DATA1 Right Channel 0100: DATA2 Left Channel 0101: DATA2 Right Channel 0110: DATA3 Left Channel 0111: DATA3 Right Channel |
| 27:24 | R/W | 0x6 | DMIC_CH6_MAP DMIC Channel 6 Mapping 0000: DATA0 Left Channel 0001: DATA0 Right Channel 0010: DATA1 Left Channel 0011: DATA1 Right Channel 0100: DATA2 Left Channel 0101: DATA2 Right Channel 0110: DATA3 Left Channel 0111: DATA3 Right Channel |
| 23:20 | R/W | 0x5 | DMIC_CH5_MAP DMIC Channel 5 Mapping 0000: DATA0 Left Channel 0001: DATA0 Right Channel 0010: DATA1 Left Channel 0011: DATA1 Right Channel 0100: DATA2 Left Channel 0101: DATA2 Right Channel 0110: DATA3 Left Channel 0111: DATA3 Right Channel |
| 19:16 | R/W | 0x4 | DMIC_CH4_MAP DMIC Channel 4 Mapping 0000: DATA0 Left Channel 0001: DATA0 Right Channel 0010: DATA1 Left Channel 0011: DATA1 Right Channel 0100: DATA2 Left Channel 0101: DATA2 Right Channel 0110: DATA3 Left Channel 0111: DATA3 Right Channel |

| Offset: 0x0028 | | | Register Name: DMIC_CH_MAP |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 15:12 | R/W | 0x3 | DMIC_CH3_MAP DMIC Channel 3 Mapping 0000: DATA0 Left Channel 0001: DATA0 Right Channel 0010: DATA1 Left Channel 0011: DATA1 Right Channel 0100: DATA2 Left Channel 0101: DATA2 Right Channel 0110: DATA3 Left Channel 0111: DATA3 Right Channel |
| 11:8 | R/W | 0x2 | DMIC_CH2_MAP DMIC Channel 2 Mapping 0000: DATA0 Left Channel 0001: DATA0 Right Channel 0010: DATA1 Left Channel 0011: DATA1 Right Channel 0100: DATA2 Left Channel 0101: DATA2 Right Channel 0110: DATA3 Left Channel 0111: DATA3 Right Channel |
| 7:4 | R/W | 0x1 | DMIC_CH1_MAP DMIC Channel 1 Mapping 0000: DATA0 Left Channel 0001: DATA0 Right Channel 0010: DATA1 Left Channel 0011: DATA1 Right Channel 0100: DATA2 Left Channel 0101: DATA2 Right Channel 0110: DATA3 Left Channel 0111: DATA3 Right Channel |
| 3:0 | R/W | 0x0 | DMIC_CH0_MAP DMIC Channel0 Mapping 0000: DATA0 Left Channel 0001: DATA0 Right Channel 0010: DATA1 Left Channel 0011: DATA1 Right Channel 0100: DATA2 Left Channel 0101: DATA2 Right Channel 0110: DATA3 Left Channel 0111: DATA3 Right Channel |

8.2.6.11 0x002C DMIC Counter Register (Default Value: 0x0000_0000)

| Offset: 0x002C | | | Register Name: DMIC_CNT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R/W | 0x0 | <p>DMIC_CNT RX Sample Counter The audio sample number of writing into RXFIFO. When one sample is read by Digital Audio Engine, the RX sample counter register increases by one. The RX sample counter register can be set to any initial value at any time. After being updated by the initial value, the counter register should count on the base of this initial value.</p> <p>Note: It is used for Audio/Video Synchronization.</p> |

8.2.6.12 0x0030 DATA0 and DATA1 Volume Control Register (Default Value: 0xA0A0_A0A0)

| Offset: 0x0030 | | | Register Name: DATA0_DATA1_VOL_CTR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:24 | R/W | 0xA0 | <p>DATA1L_VOL Data1 Left Channel Volume Control (-119.25 dB to 71.25 dB, 0.75 dB/Step) 0x00: Mute 0x01: -119.25 dB ... 0x9F: -0.75 dB 0xA0: 0 dB 0xA1: 0.75 dB ... 0xFF: 71.25 dB</p> |
| 23:16 | R/W | 0xA0 | <p>DATA1R_VOL Data1 Right Channel Volume Control (-119.25 dB to 71.25 dB, 0.75 dB/Step) 0x00: Mute 0x01: -119.25 dB ... 0x9F: -0.75 dB 0xA0: 0 dB 0xA1: 0.75 dB ... 0xFF: 71.25 dB</p> |

| Offset: 0x0030 | | | Register Name: DATA0_DATA1_VOL_CTR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 15:8 | R/W | 0xA0 | <p>DATA0L_VOL Data0 Left Channel Volume Control (-119.25 dB to 71.25 dB, 0.75 dB/Step)</p> <p>0x00: Mute 0x01: -119.25 dB ... 0x9F: -0.75 dB 0xA0: 0 dB 0xA1: 0.75 dB ... 0xFF: 71.25 dB</p> |
| 7:0 | R/W | 0xA0 | <p>DATA0R_VOL Data0 Right Channel Volume Control (-119.25 dB to 71.25 dB, 0.75 dB/Step)</p> <p>0x00: Mute 0x01: -119.25 dB ... 0x9F: -0.75 dB 0xA0: 0 dB 0xA1: 0.75 dB ... 0xFF: 71.25 dB</p> |

8.2.6.13 0x0034 DATA2 and DATA3 Volume Control Register (Default Value: 0xA0A0_A0A0)

| Offset: 0x0034 | | | Register Name: DATA2_DATA3_VOL_CTR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:24 | R/W | 0xA0 | <p>DATA3L_VOL Data3 Light Channel Volume Control (-119.25 dB to 71.25 dB, 0.75 dB/Step)</p> <p>0x00: Mute 0x01: -119.25 dB ... 0x9F: -0.75 dB 0xA0: 0 dB 0xA1: 0.75 dB ... 0xFF: 71.25 dB</p> |

| Offset: 0x0034 | | | Register Name: DATA2_DATA3_VOL_CTR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 23:16 | R/W | 0xA0 | <p>DATA3R_VOL Data3 Right Channel Volume Control (-119.25 dB to 71.25 dB, 0.75 dB/Step)</p> <p>0x00: Mute 0x01: -119.25 dB ... 0x9F: -0.75 dB 0xA0: 0 dB 0xA1: 0.75 dB ... 0xFF: 71.25 dB</p> |
| 15:8 | R/W | 0xA0 | <p>DATA2L_VOL Data2 Light Channel Volume Control (-119.25 dB to 71.25 dB, 0.75 dB/Step)</p> <p>0x00: Mute 0x01: -119.25 dB ... 0x9F: -0.75 dB 0xA0: 0 dB 0xA1: 0.75 dB ... 0xFF: 71.25 dB</p> |
| 7:0 | R/W | 0xA0 | <p>DATA2R_VOL Data2 Right Channel Volume Control (-119.25 dB to 71.25 dB, 0.75 dB/Step)</p> <p>0x00: Mute 0x01: -119.25 dB ... 0x9F: -0.75 dB 0xA0: 0 dB 0xA1: 0.75 dB ... 0xFF: 71.25 dB</p> |

8.2.6.14 0x0038 High Pass Filter Enable Control Register (Default Value: 0x0000_0000)

| Offset: 0x0038 | | | Register Name: HPF_EN_CTR |
|----------------|------------|-------------|---------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |

| Offset: 0x0038 | | | Register Name: HPF_EN_CTR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 7 | R/W | 0x0 | HPF_DATA3_CHR_EN High Pass Filter DATA3 Right Channel Enable 0: Disabled 1: Enabled |
| 6 | R/W | 0x0 | HPF_DATA3_CHL_EN High Pass Filter DATA3 Left Channel Enable 0: Disabled 1: Enabled |
| 5 | R/W | 0x0 | HPF_DATA2_CHR_EN High Pass Filter DATA2 Right Channel Enable 0: Disabled 1: Enabled |
| 4 | R/W | 0x0 | HPF_DATA2_CHL_EN High Pass Filter DATA2 Left Channel Enable 0: Disabled 1: Enabled |
| 3 | R/W | 0x0 | HPF_DATA1_CHR_EN High Pass Filter DATA1 Right Channel Enable 0: Disabled 1: Enabled |
| 2 | R/W | 0x0 | HPF_DATA1_CHL_EN High Pass Filter DATA1 Left Channel Enable 0: Disabled 1: Enabled |
| 1 | R/W | 0x0 | HPF_DATA0_CHR_EN High Pass Filter DATA0 Right Channel Enable 0: Disabled 1: Enabled |
| 0 | R/W | 0x0 | HPF_DATA0_CHL_EN High Pass Filter DATA0 Left Channel Enable 0: Disabled 1: Enabled |

8.2.6.15 0x003C High Pass Filter Coefficient Register (Default Value: 0x00FF_AA45)

| Offset: 0x003C | | | Register Name: HPF_COEF_REG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R/W | 0x00FFAA45 | HPF_COE High Pass Filter Coefficient |

8.2.6.16 0x0040 High Pass Filter Gain Register (Default Value: 0x00FF_D522)

| Offset: 0x0040 | | | Register Name: HPF_GAIN_REG |
|----------------|------------|-------------|-----------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R/W | 0x00FFD522 | HPF_GAIN High Pass Filter Gain |



8.3 Audio Codec

8.3.1 Overview

The Audio Codec is high-performance audio encoder and decoder module which supports DAC/ADC, dynamic range controller (DRC) and dynamic voltage controller (DVC) functions.

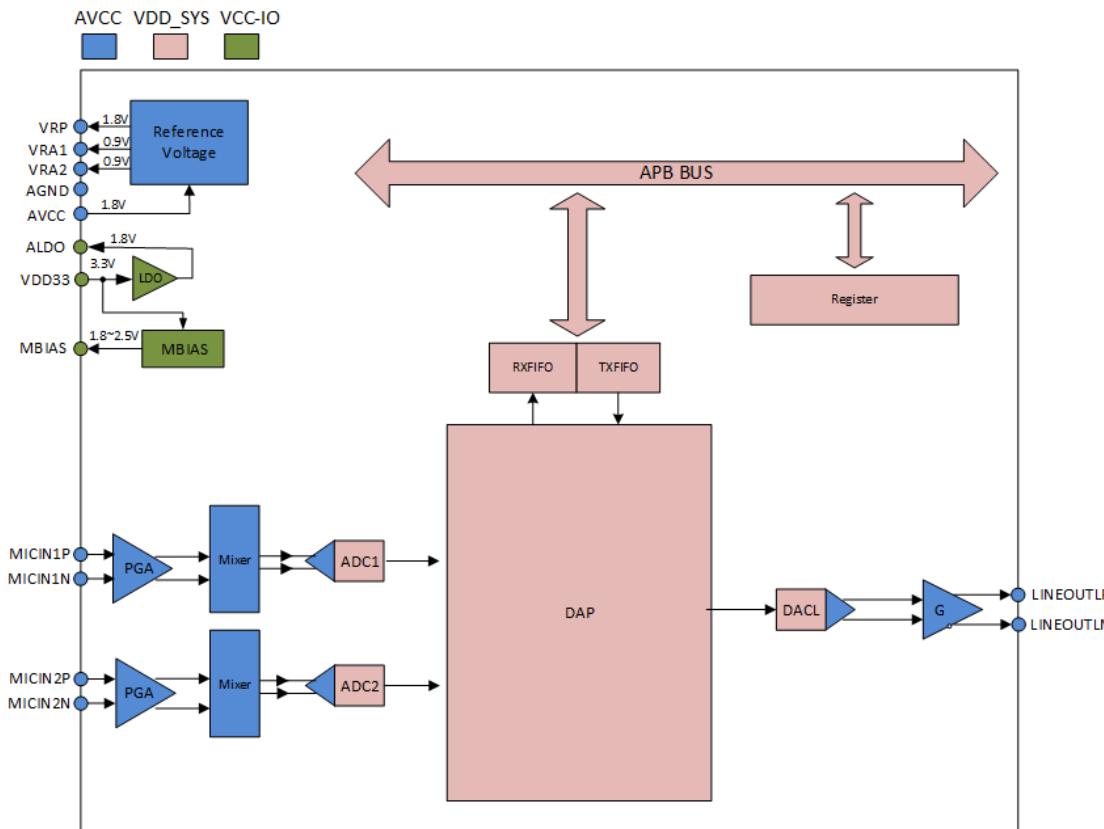
The Audio Codec has the following features:

- One audio digital-to-analog converter (DAC) channels
 - Supports the DAC sample rate from 8 kHz to 192 kHz
 - 95 ± 2 dB SNR@A-weight, -85 ± 3 dB THD + N
 - Supports 16-bit and 20-bit audio sample resolution
- Two audio analog-to-digital converter (ADC) channels
 - Supports the ADC sample rate from 8 kHz to 48 kHz
 - 95 ± 3 dB SNR@A-weight, -80 ± 3 dB THD + N
 - Supports 16-bit and 20-bit audio sample resolution
- Two audio inputs:
 - Two differential microphone inputs: MIC1P/N, MIC2P/N
- One analog audio output:
 - One differential lineout output (LINEOUTP/N)
- Supports Dynamic Range Controller (DRC) adjusting the ADC recording and DAC playback
- One 128x20-bit FIFO for DAC data transmit, one 128x20-bit FIFO for ADC data receive
- Programmable FIFO thresholds
- Supports interrupts and DMA

8.3.2 Block Diagram

The following figure shows the block diagram of Audio Codec.

Figure 8-11 Audio Codec Block Diagram



8.3.3 Functional Description

8.3.3.1 External Signals

Table 8-5 Audio Codec analog I/O

| Name | Type | Description |
|----------|------|--|
| MICIN1P | I | Positive differential input for MIC1 |
| MICIN1N | I | Negative differential input for MIC1 |
| MICIN2P | I | Positive differential input for MIC2 |
| MICIN2N | I | Negative differential input for MIC2 |
| LINEOUTP | O | Positive differential output for lineout |
| LINEOUTN | O | Negative differential output for lineout |

Table 8-6 Audio Codec reference I/O

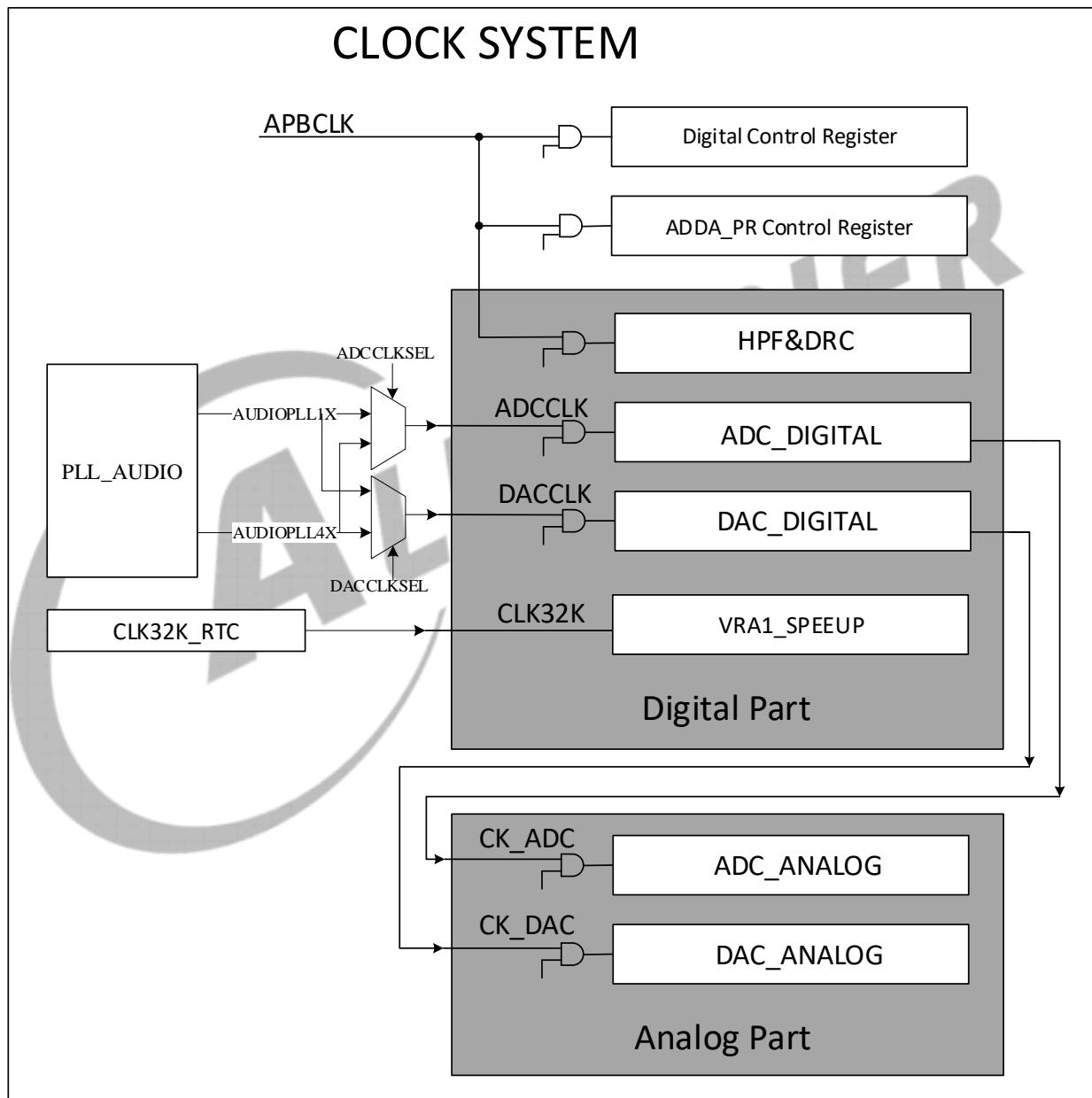
| Name | Type | Description |
|-------|------|---|
| MBIAS | O | First bias voltage output for main microphone |
| VRA1 | O | Internal reference voltage |

| Name | Type | Description |
|------|------|----------------------------|
| VRA2 | O | Internal reference voltage |
| VRP | O | Internal reference voltage |

8.3.3.2 Clock Sources

The following figure describes the clock source of Audio Codec. For clock setting, configuration, and gating information, refer to section 3.4 Clock Controller Unit (CCU).

Figure 8-12 Audio Codec Clock Diagram



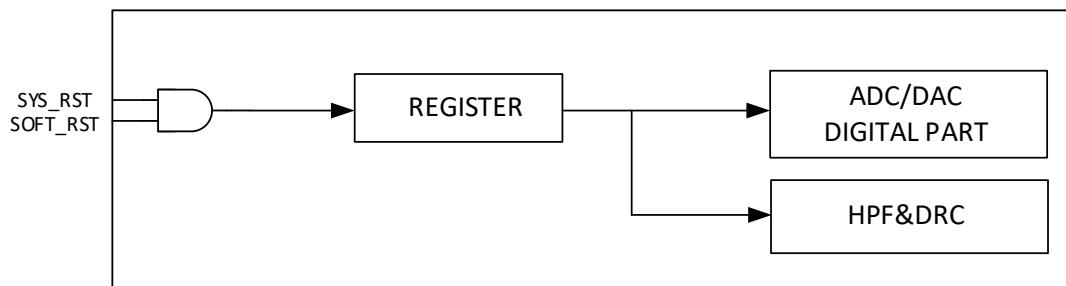
The clock source for the digital part is **PLL_AUDIO**. For the ADC clock, configure [AUDIO_CODEC ADC CLK REG\[24\]](#) to select the clock source. For the DAC clock, configure [AUDIO_CODEC DAC CLK REG\[24\]](#) to select the clock source. The PK-PK jitter of **PLL_AUDIO** should be less than 200 ps.

8.3.3.3 Reset System

Digital Part Reset System

The SYS_RST comes from the VDD_SYS domain and produced by the RTC domain which controlled by CCMU when it is powered on. Each domain has the de-bounce to confirm the reset system is strong. The codec register part will be reset by the SYS_RST when being powered on or the system soft is writing the reset control logic. The other parts will be reset by the soft configuration through writing the register. The following figure shows the reset system of the audio codec digital part.

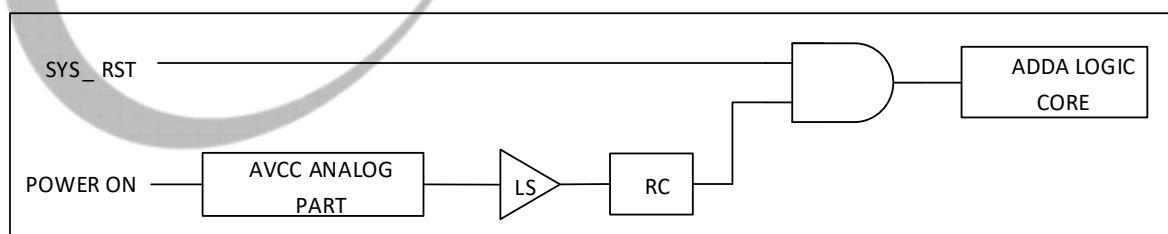
Figure 8-13 Digital Part Reset System



Analog Part Reset System

When AVCC is powered on, it sends the AVCC_POR signal. The AVCC_POR signal passes the level shift and RC filter part to the ADDA logic core. The following figure shows the reset system of the audio codec analog part.

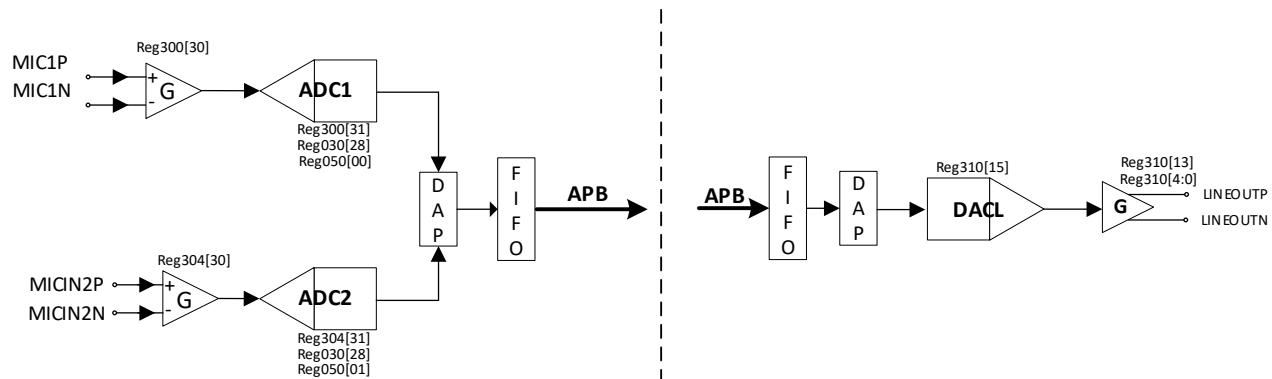
Figure 8-14 Analog Part Reset System



8.3.3.4 Data Path Diagram

The following figure shows a data path of the Audio Codec.

Figure 8-15 Audio Codec Data Path Diagram



8.3.3.5 Two ADCs

The two ADCs are used for recording stereo sound and a reference signal. The sample rates of the two ADCs are independent of the DAC sample rate. The digital ADC part can be enabled or disabled by the bit[28] of the [AC_ADC_FIFOC \(Offset: 0x0030\)](#).

8.3.3.6 Mono DAC

The mono DAC sample rate can be configured by setting the register. To save power, the analog DAC can be enabled or disabled by setting the bit[15] of the [DAC_REG \(Offset: 0x0310\)](#). The digital DAC part can be enabled or disabled by the bit[31] of the [AC_DAC_DPC \(Offset: 0x0000\)](#).

8.3.3.7 Analog Audio Input Path

The Audio Codec supports two analogue audio input paths:

- MICIN1P/N
- MICIN2P/N

MICIN1P/N provide differential input that can be mixed into the ADC1 record mixer. MICIN2P/N provide differential input that can be mixed into the ADC2 record mixer. MICIN is a high impedance, low capacitance input suitable for connection to various differential microphones of different dynamics and sensitivity. The gain for each pre-amplifier can be set independently. MBIAS provides reference voltage for electret condenser type(ECM) microphones.

8.3.3.8 Analog Audio Output Port

The Codec has One type analog output ports:

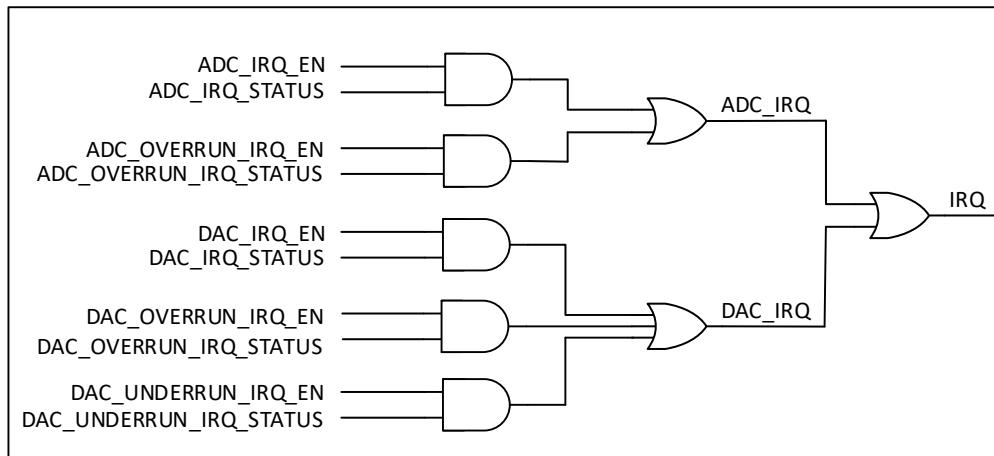
- LINEOUT

LINEOUTL provides one differential output to drive line signals to external audio equipment. The LINEOUTLP/N output source from DACL. The volume control is logarithmic with an 43.5 dB range in 1.5 dB step from -43.5 dB to 0 dB. The LINEOUTL/R output buffer power up or down by bit[13] or bit[11] of DAC_REG (Offset: 0x0310).

8.3.3.9 Interrupts

The Audio Codec has two interrupts. The following figure describes the Audio Codec interrupt system.

Figure 8-16 Audio Codec Interrupt System

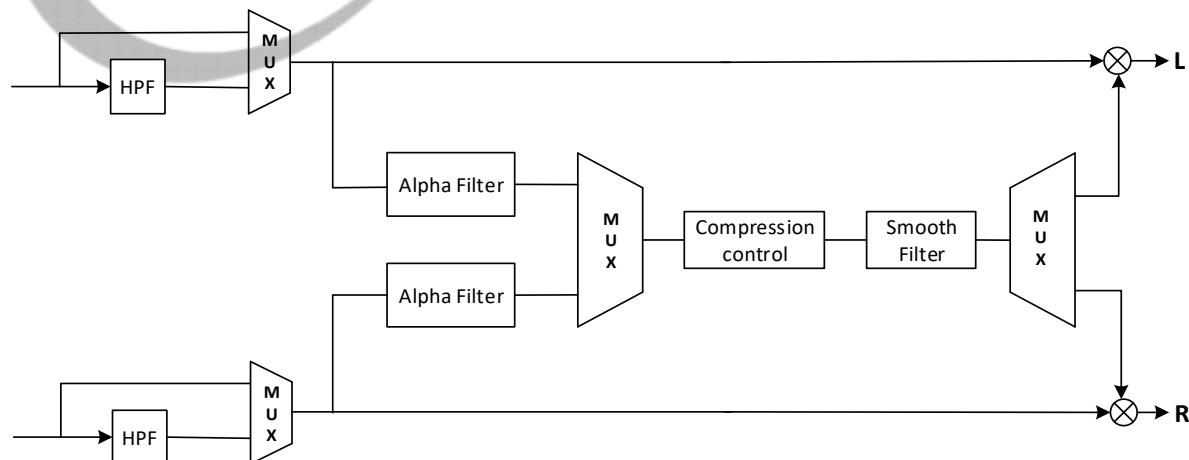


8.3.3.10 Digital Audio Processor (DAP)

The DAP module is used to remove the DC offset and automatically adjusts the volume to a flatten volume level. It mainly consists of two HPF and one DRC.

The following figure shows the DAP data flow.

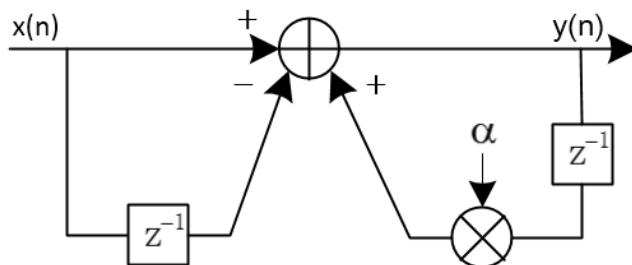
Figure 8-17 DAP Data Flow



HPF Function

The DAP has individual channel high pass filter (HPF, -3dB cutoff < 1Hz) that can be enabled and disabled. The filter cutoff frequency is less than 1 Hz that can be removed DC offset from ADC recording. The HPF can also be bypassed.

Figure 8-18 HPF Logic Structure



DRC Function

The DRC scheme has three thresholds, three offsets, and four slopes (all programmable). There is one ganged DRC for the left and right channels. The following figure shows the diagram of DRC input/output.

Figure 8-19 DRC static Curve Parameters

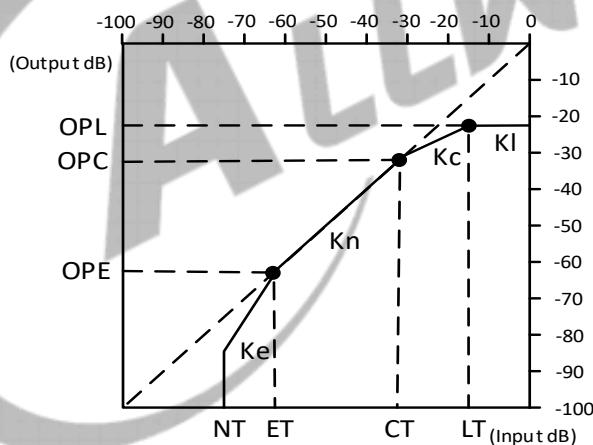


Figure 8-20 DRC Block Diagram



Professional-quality dynamic range compression automatically adjusts volume to flatten volume level.

One DRC for left/right and one DRC for subwoofer.

Each DRC has the adjustable threshold, offset, compression levels, programmable energy, attack, and decay time constants.

Transparent compression: Compressors can attack fast enough to avoid apparent clipping before engaging, and decay times can be set slow enough to avoid pumping.

Configure the DRC parameters according to the following guidelines:

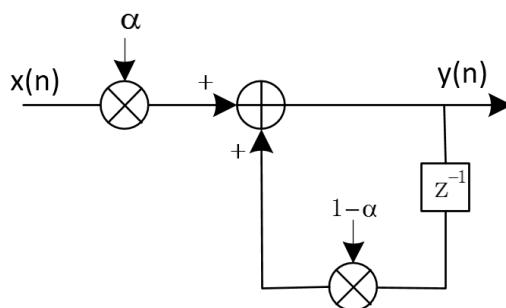
- Number format

The Number format is N.M which means there are N bits to the left of the decimal point including the sign bit and M bits to the right of the decimal point. For example, Numbers formatted 9.23 means that there are 9 bits at the left of the decimal point and 23 bits at the right decimal point.

- Energy Filter

The following figure shows the structure of the energy filter.

Figure 8-21 Energy Filter Structure



The Energy Filter is to estimate of the RMS value of the audio data stream into DRC, and has two parameters, which determine the time window over which RMS to be made. The parameter is computed by

$$\alpha = 1 - e^{-2.2Ts/\tau_a}$$

Compression Control

This element has six parameters (ET, CT, LT, Ke, Kn, Kc, Kl, OPL, OPC, OPE), which are all programmable, and the computation will be explained as follows.

- Threshold Parameter Computation (T parameter)

The threshold is the value that determines the signal to be compressed or not. When the signal's RMS is larger than the threshold, the signal will be compressed. The value of threshold input to the coefficient register is

$$\text{computed by } Tin = -\frac{T_{dB}}{6.0206}$$

Where, T_{dB} must less than zero, the positive value is illegal.

For example, it is desired to set CT=-40dB, then the Tin require to set CT to -40dB is $CTin = -(-40dB)/6.0206 = 6.644$, CTin is entered as a 32-bit number in 8.24 format.

Therefore, $CTin = 6.644 = 0000\ 0110.\underline{1010}\ 0100\ 1101\ 0011\ 1100\ 0000 = 0x06A4\ D3C0$ in 8.24 format.

- Slope Parameter Computation (K parameter)

The K is the slope within compression region. For example, a n:1 compression means that an output increase 1dB is for n dB RMS input. The k input to the coefficient ram is computed by $K = \frac{1}{n}$

Where, n is from 1 to 50, and must be integer.

For example, it is desired to set 2:1, then the Kc require to set to 2:1 is $K_c = 1/2 = 0.5$, Kc is entered as a 32-bit number in 8.24 format.

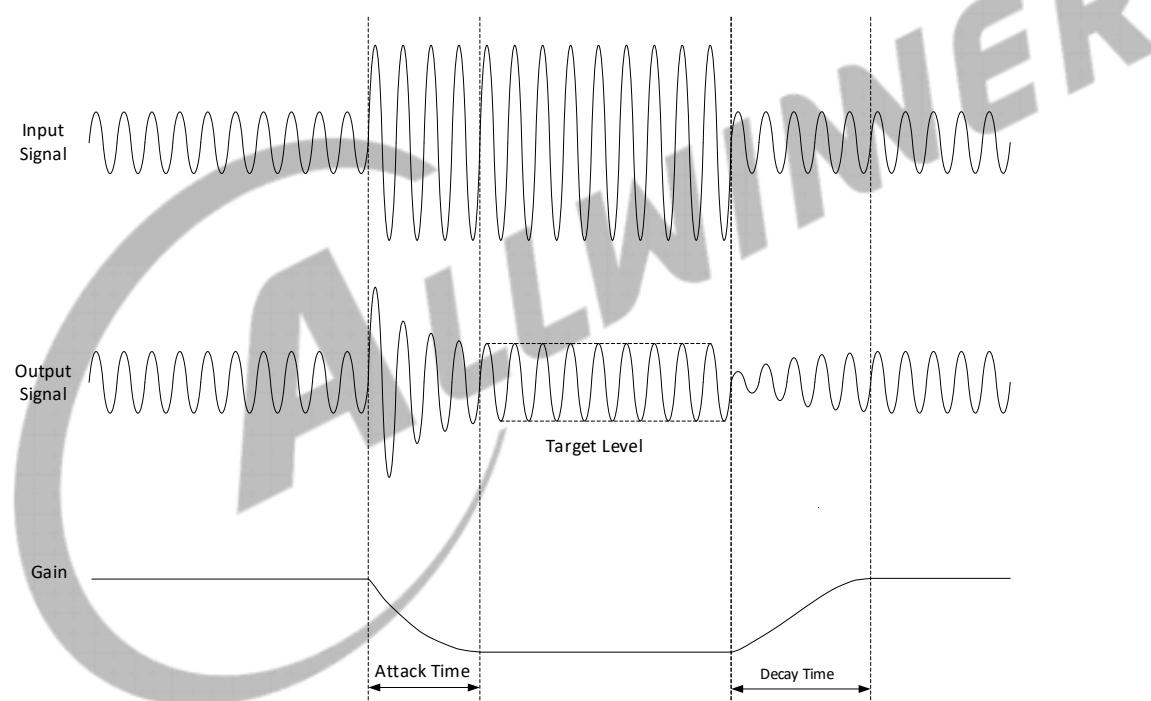
Therefore, $K_c = 0.5 = 0000\ 0000.1000\ 0000\ 0000\ 0000\ 0000 = 0x0080\ 0000$ in 8.24 format.

- Gain Smooth Filter

The Gain Smooth Filter is to smooth the gain and control the ratio of gain increase and decrease. The decay time and attack is shown in Figure 1-17. The structure of the Gain Smooth filter is also the Alpha filter, so the

rise time computation is the same as the Energy filter which is $\alpha = 1 - e^{-2.2T_s/ta}$.

Figure 8-22 Gain Smooth Filter



8.3.4 Programming Guidelines

8.3.4.1 Record Process

In recording mode, the analog audio signals are recorded from the microphones at the specified sample rate, processed by the ADC, and then transferred to the DRAM via the DMA.

1. Codec initialization: configure [AUDIO_CODEC_BGR_REG](#) to open the audio codec bus clock gating and de-assert bus reset; configure [AUDIO_CODEC_ADC_CLK_REG](#) and [PLL_AUDIO_CTRL_REG](#) to configure PLL_Audio frequency and enable PLL_Audio. For details, refer to section
2. Configure the sample rate and data transfer format, then open the ADC.

3. Configure the DMA and DMA request.
4. Enable the ADC DRQ and DMA.

8.3.4.2 Playback Process

In playback mode, the audio data are transferred from the DRAM via DMA, processed by the DAC, and finally output via the analog interface.

1. Codec initialization: configure [AUDIO_CODEC_BGR_REG](#) to open the audio codec bus clock gating and de-assert bus reset; configure [AUDIO_CODEC_DAC_CLK_REG](#) and [PLL_AUDIO_CTRL_REG](#) to configure PLL_Audio frequency and enable PLL_Audio. For details, refer to section 3.4 Clock Controller Unit (CCU).
2. Configure the sample rate and data transfer format, then open the DAC.
3. Configure the DMA and DMA request.
4. Enable the DAC DRQ and DMA.

8.3.5 Register List

| Module Name | Base Address | Comments |
|-------------|--------------|---------------|
| Audio Codec | 0x02030000 | ADC/DAC Codec |

| Register Name | Offset | Description |
|-----------------------|--------|-----------------------------------|
| Audio Codec | | |
| AC_DAC_DPC | 0x0000 | DAC Digital Part Control Register |
| DAC_VOL_CTRL | 0x0004 | DAC Volume Control Register |
| AC_DAC_FIFOC | 0x0010 | DAC FIFO Control Register |
| AC_DAC_FIFOS | 0x0014 | DAC FIFO Status Register |
| AC_DAC_TXDATA | 0x0020 | DAC TX DATA Register |
| AC_DAC_CNT | 0x0024 | DAC TX FIFO Counter Register |
| AC_DAC_DG | 0x0028 | DAC Debug Register |
| AC_ADC_FIFOC | 0x0030 | ADC FIFO Control Register |
| ADC_VOL_CTRL1 | 0x0034 | ADC Volume Control1 Register |
| AC_ADC_FIFOS | 0x0038 | ADC FIFO Status Register |
| AC_ADC_RXDATA | 0x0040 | ADC RX Data Register |
| AC_ADC_CNT | 0x0044 | ADC RX Counter Register |
| AC_ADC_DG | 0x004C | ADC Debug Register |
| ADC_DIG_CTRL | 0x0050 | ADC Digital Control Register |
| VRA1SPEEDUP_DOWN_CTRL | 0x0054 | VRA1Speedup Down Control Register |
| AC_DAC_DAP_CTRL | 0x00F0 | DAC DAP Control Register |
| AC_ADC_DAP_CTR | 0x00F8 | ADC DAP Control Register |
| AC_DAC_DRC_HHPFC | 0x0100 | DAC DRC High HPF Coef Register |
| AC_DAC_DRC_LHPFC | 0x0104 | DAC DRC Low HPF Coef Register |
| AC_DAC_DRC_CTRL | 0x0108 | DAC DRC Control Register |

| Register Name | Offset | Description |
|--------------------|--------|---|
| AC_DAC_DRC_LPFHAT | 0x010C | DAC DRC Left Peak Filter High Attack Time Coef Register |
| AC_DAC_DRC_LPFLAT | 0x0110 | DAC DRC Left Peak Filter Low Attack Time Coef Register |
| AC_DAC_DRC_RPFHAT | 0x0114 | DAC DRC Right Peak Filter High Attack Time Coef Register |
| AC_DAC_DRC_RPFLAT | 0x0118 | DAC DRC Left Peak Filter Low Attack Time Coef Register |
| AC_DAC_DRC_LPFHRT | 0x011C | DAC DRC Left Peak Filter High Release Time Coef Register |
| AC_DAC_DRC_LPFLRT | 0x0120 | DAC DRC Left Peak Filter Low Release Time Coef Register |
| AC_DAC_DRC_RPFHRT | 0x0124 | DAC DRC Right Peak filter High Release Time Coef Register |
| AC_DAC_DRC_RPFLRT | 0x0128 | DAC DRC Right Peak filter Low Release Time Coef Register |
| AC_DAC_DRC_LRMSHAT | 0x012C | DAC DRC Left RMS Filter High Coef Register |
| AC_DAC_DRC_LRMSLAT | 0x0130 | DAC DRC Left RMS Filter Low Coef Register |
| AC_DAC_DRC_RRMSHAT | 0x0134 | DAC DRC Right RMS Filter High Coef Register |
| AC_DAC_DRC_RRMSLAT | 0x0138 | DAC DRC Right RMS Filter Low Coef Register |
| AC_DAC_DRC_HCT | 0x013C | DAC DRC Compressor Threshold High Setting Register |
| AC_DAC_DRC_LCT | 0x0140 | DAC DRC Compressor Slope High Setting Register |
| AC_DAC_DRC_HKC | 0x0144 | DAC DRC Compressor Slope High Setting Register |
| AC_DAC_DRC_LKC | 0x0148 | DAC DRC Compressor Slope Low Setting Register |
| AC_DAC_DRC_HOPC | 0x014C | DAC DRC Compressor High Output at Compressor Threshold Register |
| AC_DAC_DRC_LOPC | 0x0150 | DAC DRC Compressor Low Output at Compressor Threshold Register |
| AC_DAC_DRC_HLT | 0x0154 | DAC DRC Limiter Threshold High Setting Register |
| AC_DAC_DRC_LLT | 0x0158 | DAC DRC Limiter Threshold Low Setting Register |
| AC_DAC_DRC_HKI | 0x015C | DAC DRC Limiter Slope High Setting Register |
| AC_DAC_DRC_LKI | 0x0160 | DAC DRC Limiter Slope Low Setting Register |
| AC_DAC_DRC_HOPL | 0x0164 | DAC DRC Limiter High Output at Limiter Threshold |
| AC_DAC_DRC_LOPL | 0x0168 | DAC DRC Limiter Low Output at Limiter Threshold |
| AC_DAC_DRC_HET | 0x016C | DAC DRC Expander Threshold High Setting Register |
| AC_DAC_DRC_LET | 0x0170 | DAC DRC Expander Threshold Low Setting Register |
| AC_DAC_DRC_HKE | 0x0174 | DAC DRC Expander Slope High Setting Register |
| AC_DAC_DRC_LKE | 0x0178 | DAC DRC Expander Slope Low Setting Register |
| AC_DAC_DRC_HOPE | 0x017C | DAC DRC Expander High Output at Expander Threshold |
| AC_DAC_DRC_LOPE | 0x0180 | DAC DRC Expander Low Output at Expander Threshold |
| AC_DAC_DRC_HKN | 0x0184 | DAC DRC Linear Slope High Setting Register |
| AC_DAC_DRC_LKN | 0x0188 | DAC DRC Linear Slope Low Setting Register |
| AC_DAC_DRC_SFHAT | 0x018C | DAC DRC Smooth filter Gain High Attack Time Coef Register |
| AC_DAC_DRC_SFLAT | 0x0190 | DAC DRC Smooth filter Gain Low Attack Time Coef Register |

| Register Name | Offset | Description |
|---------------------|--------|---|
| AC_DAC_DRC_SFVRT | 0x0194 | DAC DRC Smooth filter Gain High Release Time Coef Register |
| AC_DAC_DRC_SFLRT | 0x0198 | DAC DRC Smooth filter Gain Low Release Time Coef Register |
| AC_DAC_DRC_MXGHS | 0x019C | DAC DRC MAX Gain High Setting Register |
| AC_DAC_DRC_MXGLS | 0x01A0 | DAC DRC MAX Gain Low Setting Register |
| AC_DAC_DRC_MNGHS | 0x01A4 | DAC DRC MIN Gain High Setting Register |
| AC_DAC_DRC_MNGLS | 0x01A8 | DAC DRC MIN Gain Low Setting Register |
| AC_DAC_DRC_EPSHC | 0x01AC | DAC DRC Expander Smooth Time High Coef Register |
| AC_DAC_DRC_EPSLC | 0x01B0 | DAC DRC Expander Smooth Time Low Coef Register |
| AC_DAC_DRC_HPFHGAIN | 0x01B8 | DAC DRC HPF Gain High Coef Register |
| AC_DAC_DRC_HPFLGAIN | 0x01BC | DAC DRC HPF Gain Low Coef Register |
| AC_ADC_DRC_HHPFC | 0x0200 | ADC DRC High HPF Coef Register |
| AC_ADC_DRC_LHPFC | 0x0204 | ADC DRC Low HPF Coef Register |
| AC_ADC_DRC_CTRL | 0x0208 | ADC DRC Control Register |
| AC_ADC_DRC_LPFBAT | 0x020C | ADC DRC Left Peak Filter High Attack Time Coef Register |
| AC_ADC_DRC_LPFLAT | 0x0210 | ADC DRC Left Peak Filter Low Attack Time Coef Register |
| AC_ADC_DRC_RPFHAT | 0x0214 | ADC DRC Right Peak Filter High Attack Time Coef Register |
| AC_ADC_DRC_RPFLAT | 0x0218 | ADC DRC Right Peak Filter Low Attack Time Coef Register |
| AC_ADC_DRC_LPFHRT | 0x021C | ADC DRC Left Peak Filter High Release Time Coef Register |
| AC_ADC_DRC_LPFLRT | 0x0220 | ADC DRC Left Peak Filter Low Release Time Coef Register |
| AC_ADC_DRC_RPFHRT | 0x0224 | ADC DRC Right Peak filter High Release Time Coef Register |
| AC_ADC_DRC_RPFLRT | 0x0228 | ADC DRC Right Peak filter Low Release Time Coef Register |
| AC_ADC_DRC_LRMSBAT | 0x022C | ADC DRC Left RMS Filter High Coef Register |
| AC_ADC_DRC_LRMSLAT | 0x0230 | ADC DRC Left RMS Filter Low Coef Register |
| AC_ADC_DRC_RRMSBAT | 0x0234 | ADC DRC Right RMS Filter High Coef Register |
| AC_ADC_DRC_RRMSLAT | 0x0238 | ADC DRC Right RMS Filter Low Coef Register |
| AC_ADC_DRC_HCT | 0x023C | ADC DRC Compressor Threshold High Setting Register |
| AC_ADC_DRC_LCT | 0x0240 | ADC DRC Compressor Slope High Setting Register |
| AC_ADC_DRC_HKC | 0x0244 | ADC DRC Compressor Slope High Setting Register |
| AC_ADC_DRC_LKC | 0x0248 | ADC DRC Compressor Slope Low Setting Register |
| AC_ADC_DRC_HOPC | 0x024C | ADC DRC Compressor High Output at Compressor Threshold Register |
| AC_ADC_DRC_LOPC | 0x0250 | ADC DRC Compressor Low Output at Compressor Threshold Register |
| AC_ADC_DRC_HLT | 0x0254 | ADC DRC Limiter Threshold High Setting Register |
| AC_ADC_DRC_LLTT | 0x0258 | ADC DRC Limiter Threshold Low Setting Register |

| Register Name | Offset | Description |
|---------------------|--------|--|
| AC_ADC_DRC_HKI | 0x025C | ADC DRC Limiter Slope High Setting Register |
| AC_ADC_DRC_LKI | 0x0260 | ADC DRC Limiter Slope Low Setting Register |
| AC_ADC_DRC_HOPL | 0x0264 | ADC DRC Limiter High Output at Limiter Threshold |
| AC_ADC_DRC_LOPL | 0x0268 | ADC DRC Limiter Low Output at Limiter Threshold |
| AC_ADC_DRC_HET | 0x026C | ADC DRC Expander Threshold High Setting Register |
| AC_ADC_DRC LET | 0x0270 | ADC DRC Expander Threshold Low Setting Register |
| AC_ADC_DRC_HKE | 0x0274 | ADC DRC Expander Slope High Setting Register |
| AC_ADC_DRC_LKE | 0x0278 | ADC DRC Expander Slope Low Setting Register |
| AC_ADC_DRC_HOPE | 0x027C | ADC DRC Expander High Output at Expander Threshold |
| AC_ADC_DRC_LOPE | 0x0280 | ADC DRC Expander Low Output at Expander Threshold |
| AC_ADC_DRC_HKN | 0x0284 | ADC DRC Linear Slope High Setting Register |
| AC_ADC_DRC_LKN | 0x0288 | ADC DRC Linear Slope Low Setting Register |
| AC_ADC_DRC_SFHAT | 0x028C | ADC DRC Smooth filter Gain High Attack Time Coef Register |
| AC_ADC_DRC_SFLAT | 0x0290 | ADC DRC Smooth filter Gain Low Attack Time Coef Register |
| AC_ADC_DRC_SFVRT | 0x0294 | ADC DRC Smooth filter Gain High Release Time Coef Register |
| AC_ADC_DRC_SFLRT | 0x0298 | ADC DRC Smooth filter Gain Low Release Time Coef Register |
| AC_ADC_DRC_MXGHS | 0x029C | ADC DRC MAX Gain High Setting Register |
| AC_ADC_DRC_MXGLS | 0x02A0 | ADC DRC MAX Gain Low Setting Register |
| AC_ADC_DRC_MNGHS | 0x02A4 | ADC DRC MIN Gain High Setting Register |
| AC_ADC_DRC_MNGLS | 0x02A8 | ADC DRC MIN Gain Low Setting Register |
| AC_ADC_DRC_EPSHC | 0x02AC | ADC DRC Expander Smooth Time High Coef Register |
| AC_ADC_DRC_EPSLC | 0x02B0 | ADC DRC Expander Smooth Time Low Coef Register |
| AC_ADC_DRC_HPFHGAIN | 0x02B8 | ADC DRC HPF Gain High Coef Register |
| AC_ADC_DRC_HPFLGAIN | 0x02BC | ADC DRC HPF Gain Low Coef Register |
| ADC1_REG | 0x0300 | ADC1 Analog Control Register |
| ADC2_REG | 0x0304 | ADC2 Analog Control Register |
| DAC_REG | 0x0310 | DAC Analog Control Register |
| MICBIAS_REG | 0x0318 | MICBIAS Analog Control Register |
| BIAS_REG | 0x0320 | BIAS Analog Control Register |
| POWER_REG | 0x0348 | POWER Analog Control Register |
| ADC_CUR_REG | 0x034C | ADC Current Analog Control Register |

8.3.6 Register Description

8.3.6.1 0x0000 DAC Digital Part Control Register (Default Value: 0x0000_0000)

| Offset: 0x0000 | | | Register Name: AC_DAC_DPC |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31 | R/W | 0x0 | EN_DA DAC Digital Part Enable 0: Disabled 1: Enabled |
| 30:29 | / | / | / |
| 28:25 | R/W | 0x0 | MODQU Internal DAC Quantization Levels Levels = [7*(21 + MODQU[3:0])]/128 Default levels = 7*21/128 = 1.15 |
| 24 | R/W | 0x0 | DWA DWA Function Disable 0: Enabled 1: Disabled |
| 23:19 | / | / | / |
| 18 | R/W | 0x0 | HPF_EN High Pass Filter Enable 0: Disabled 1: Enabled |
| 17:12 | R/W | 0x0 | DVOL Digital volume control: DVC, ATT = DVC[5:0]*(-1.16 dB) 64 steps, -1.16 dB/step |
| 11:1 | / | / | / |
| 0 | R/W | 0x0 | HUB_EN Audio Hub Enable The bit takes effect only when the EN_DA is set to 1. System Domain: Audio Codec/I2S0/I2S1 TXFIFO Hub Enable. 0: Disabled 1: Enabled |

8.3.6.2 0x0004 DAC Volume Control Register (Default Value: 0x0000_A0A0)

| Offset: 0x0004 | | | Register Name: DAC_VOL_CTRL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:17 | / | / | / |
| 16 | R/W | 0x0 | DAC_VOL_SEL DAC Volume Control Selection Enable 0: Disabled 1: Enabled |

| Offset: 0x0004 | | | Register Name: DAC_VOL_CTRL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 15:8 | R/W | 0xA0 | <p>DAC_VOL_L DAC left channel volume (-119.25 dB to 71.25 dB, 0.75 dB/Step)</p> <p>0x00: Mute 0x01: -119.25 dB ... 0x9F = -0.75 dB 0xA0 = 0 dB 0xA1 = 0.75 dB ... 0xFF = 71.25 dB</p> |
| 7:0 | R/W | 0xA0 | <p>DAC_VOL_R DAC right channel volume (-119.25 dB To 71.25 dB, 0.75 dB/Step)</p> <p>0x00: Mute 0x01: -119.25 dB ... 0x9F = -0.75 dB 0xA0 = 0 dB 0xA1 = 0.75 dB ... 0xFF = 71.25 dB</p> |

8.3.6.3 0x0010 DAC FIFO Control Register (Default Value: 0x0000_4000)

| Offset: 0x0010 | | | Register Name: AC_DAC_FIFOC |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:29 | R/W | 0x0 | <p>DAC_FS Sample Rate of DAC 000: 48 kHz 010: 24 kHz 100: 12 kHz 110: 192 kHz 001: 32 kHz 011: 16 kHz 101: 8 kHz 111: 96 kHz 44.1 kHz/22.05 kHz/11.025 kHz can be supported by Audio PLL Configure Bit</p> |

| Offset: 0x0010 | | | Register Name: AC_DAC_FIFOC |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 28 | R/W | 0x0 | FIR_VER FIR Version 0: 64-Tap FIR 1: 32-Tap FIR |
| 27 | / | / | / |
| 26 | R/W | 0x0 | SEND_LASAT Audio sample select when TX FIFO underrun 0: Sending zero 1: Sending the last audio sample |
| 25:24 | R/W | 0x0 | FIFO_MODE For 20-bit transmitted audio sample: 00/10: FIFO_I[19:0] = {TXDATA[31:12]} 01/11: FIFO_I[19:0] = {TXDATA[19:0]} For 16-bit transmitted audio sample: 00/10: FIFO_I[19:0] = {TXDATA[31:16], 4'b0} 01/11: FIFO_I[19:0] = {TXDATA[15:0], 4'b0} |
| 23 | / | / | / |
| 22:21 | R/W | 0x0 | DAC_DRQ_CLR_CNT When TX FIFO available room is less than or equal N, the DRQ request will be de-asserted. N is defined here: 00: IRQ/DRQ de-asserted when WLEVEL > TXTL 01: 4 10: 8 11: 16 |
| 20:15 | / | / | / |
| 14:8 | R/W | 0x40 | TX_TRIG_LEVEL TX FIFO Empty Trigger Level (TXTL[12:0]) Interrupt and DMA request trigger level for TX FIFO normal condition. IRQ/DRQ generated when WLEVEL ≤ TXTL Note: WLEVEL represents the number of valid samples in the TX FIFO. Only TXTL[6:0] valid when TXMODE = 0. |
| 7 | / | / | / |
| 6 | R/W | 0x0 | DAC_MONO_EN DAC Mono Enable 0: Stereo, 64 levels FIFO 1: Mono, 128 levels FIFO When enabled, L & R channel send the same data. |
| 5 | R/W | 0x0 | TX_SAMPLE_BITS Transmitting Audio Sample Resolution 0: 16 bits 1: 20 bits |

| Offset: 0x0010 | | | Register Name: AC_DAC_FIFOC |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 4 | R/W | 0x0 | DAC_DRQ_EN DAC FIFO Empty DRQ Enable 0: Disabled 1: Enabled |
| 3 | R/W | 0x0 | DAC_IRQ_EN DAC FIFO Empty IRQ Enable 0: Disabled 1: Enabled |
| 2 | R/W | 0x0 | FIFO_UNDERRUN_IRQ_EN DAC FIFO Underrun IRQ Enable 0: Disabled 1: Enabled |
| 1 | R/W | 0x0 | FIFO_OVERRUN_IRQ_EN DAC FIFO Overrun IRQ Enable 0: Disabled 1: Enabled |
| 0 | R/WC | 0x0 | FIFO_FLUSH DAC FIFO Flush Write '1' to flush TX FIFO, self clear to '0' |

8.3.6.4 0x0014 DAC FIFO Status Register (Default Value: 0x0080_8008)

| Offset: 0x0014 | | | Register Name: AC_DAC_FIFOS |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:24 | / | / | / |
| 23 | R | 0x1 | TX_EMPTY TX FIFO Empty 0: No room for new sample in TX FIFO 1: More than one room for new sample in TX FIFO (>= 1 word) |
| 22:8 | R | 0x80 | TXE_CNT TX FIFO Empty Space Word Counter |
| 7:4 | / | / | / |
| 3 | R/W1C | 0x1 | TXE_INT TX FIFO Empty Pending Interrupt 0: No Pending IRQ 1: FIFO Empty Pending Interrupt Write '1' to clear this interrupt or automatically clear if the interrupt condition fails. |

| Offset: 0x0014 | | | Register Name: AC_DAC_FIFOS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 2 | R/W1C | 0x0 | TXU_INT TX FIFO Underrun Pending Interrupt 0: No Pending Interrupt 1: FIFO Underrun Pending Interrupt Write '1' to clear this interrupt |
| 1 | R/W1C | 0x0 | TXO_INT TX FIFO Overrun Pending Interrupt 0: No Pending Interrupt 1: FIFO Overrun Pending Interrupt Write '1' to clear this interrupt |
| 0 | / | / | / |

8.3.6.5 0x0020 DAC TX DATA Register (Default Value: 0x0000_0000)

| Offset: 0x0020 | | | Register Name: AC_DAC_TXDATA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | W | 0x0 | TX_DATA Write the transmitting left and right channel sample data to this register one by one. Write the left channel sample data first and then the right channel sample. |

8.3.6.6 0x0024 DAC TX Counter Register (Default Value: 0x0000_0000)

| Offset: 0x0024 | | | Register Name: AC_DAC_CNT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R/W | 0x0 | TX_CNT TX Sample Counter The audio sample number of sending into TXFIFO. When one sample is put into TXFIFO by DMA or by host IO, the TX sample counter register increases by one. The TX sample counter register can be set to any initial value at any time. After being updated by the initial value, the counter register should count from this initial value. Note: It is used for Audio/Video Synchronization. |

8.3.6.7 0x0028 DAC Debug Register (Default Value: 0x0000_0000)

| Offset: 0x0028 | | | Register Name: AC_DAC_DG |
|----------------|------------|-------------|--------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:12 | / | / | / |

| Offset: 0x0028 | | | Register Name: AC_DAC_DG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 11 | R/W | 0x0 | DAC_MODU_SELECT DAC Modulator Debug 0: DAC Modulator Normal Mode 1: DAC Modulator Debug Mode |
| 10:9 | R/W | 0x0 | DAC_PATTERN_SELECT DAC Pattern Select 00: Normal (Audio sample from TX FIFO) 01: -6 dB Sin wave 10: -60 dB Sin wave 11: Silent wave |
| 8 | R/W | 0x0 | CODEC_CLK_SELECT CODEC Clock Source Select 0: CODEC clock from PLL 1: CODEC clock from OSC (for Debug) |
| 7 | / | / | / |
| 6 | R/W | 0x0 | DA_SWP DAC Output Channel Swap Enable 0: Disabled 1: Enabled |
| 5:3 | / | / | / |
| 2:0 | R/W | 0x0 | ADDA_LOOP_MODE ADDA Loop Mode Select 000: Disabled 001: ADDA LOOP MODE DACL/DACR is connected to ADC1/ADC2 010: ADDA LOOP MODE DACL/DACR is connected to ADC3 Others: Reserved |

8.3.6.8 0x0030 ADC FIFO Control Register (Default Value: 0x0000_0400)

| Offset: 0x0030 | | | Register Name: AC_ADC_FIFOC |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:29 | R/W | 0x0 | <p>ADFS Sample Rate of ADC 000: 48 kHz 010: 24 kHz 100: 12 kHz 110: Reserved 001: 32 kHz 011: 16 kHz 101: 8 kHz 111: Reserved 44.1 kHz/22.05 kHz/11.025 kHz can be supported by Audio PLL Configure Bit.</p> |
| 28 | R/W | 0x0 | <p>EN_AD ADC Digital Part Enable 0: Disabled 1: Enabled</p> |
| 27:26 | R/W | 0x0 | <p>ADCFDT ADC FIFO delay time for writing data after EN_AD 00: 5 ms 01: 10 ms 10: 20 ms 11: 30 ms</p> |
| 25 | R/W | 0x0 | <p>ADCDFEN ADC FIFO delay function for writing data after EN_AD 0: Disabled 1: Enabled</p> |
| 24 | R/W | 0x0 | <p>RX_FIFO_MODE RX FIFO Output Mode (Mode 0, 1) 0: Expanding '0' at LSB of TX FIFO register 1: Expanding received sample sign bit at MSB of TX FIFO register For 20-bit received audio sample: Mode 0: RXDATA[31:0] = {FIFO_O[19:0], 12'h0} Mode 1: RXDATA[31:0] = {12{FIFO_O[19]}, FIFO_O[19:0]} For 16-bit received audio sample: Mode 0: RXDATA[31:0] = {FIFO_O[19:4], 16'h0} Mode 1: RXDATA[31:0] = {16{FIFO_O[19]}, FIFO_O[19:4]}</p> |
| 23:22 | / | / | / |

| Offset: 0x0030 | | | Register Name: AC_ADC_FIFOC |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 21 | R/W | 0x0 | <p>RX_SYNC_EN_START The bit takes effect only when RX_SYNC_EN is set to 1. System Domain: Audio codec/I2S0/I2S1/DMIC Synchronize Enable Start. 0: Disabled 1: Enabled</p> |
| 20 | R/W | 0x0 | <p>RX_SYNC_EN Audio codec RX Synchronize Enable 0: Disabled 1: Enabled</p> |
| 19:17 | / | / | / |
| 16 | R/W | 0x0 | <p>RX_SAMPLE_BITS Receiving Audio Sample Resolution 0: 16 bits 1: 20 bits</p> |
| 15:12 | / | / | / |
| 11:4 | R/W | 0x40 | <p>RX_FIFO_TRG_LEVEL RX FIFO Trigger Level (RXTL[5:0]) Interrupt and DMA request trigger level for RX FIFO normal condition IRQ/DRQ generated when WLEVEL > RXTL[5:0]</p> <p>Note: WLEVEL represents the number of valid samples in the RX FIFO.</p> |
| 3 | R/W | 0x0 | <p>ADC_DRQ_EN ADC FIFO Data Available DRQ Enable 0: Disabled 1: Enabled</p> |
| 2 | R/W | 0x0 | <p>ADC_IRQ_EN ADC FIFO Data Available IRQ Enable 0: Disabled 1: Enabled</p> |
| 1 | R/W | 0x0 | <p>ADC_OVERRUN_IRQ_EN ADC FIFO Overrun IRQ Enable 0: Disabled 1: Enabled</p> |
| 0 | R/WC | 0x0 | <p>ADC_FIFO_FLUSH ADC FIFO Flush Write '1' to flush TX FIFO, self clear to '0'.</p> |

8.3.6.9 0x0034 ADC Volume Control1 Register (Default Value: 0xA0A0_A0A0)

| Offset: 0x0034 | | | Register Name: ADC_VOL_CTRL1 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:24 | R/W | 0xA0 | Reserved |
| 23:16 | R/W | 0xA0 | ADC3_VOL ADC3 channel volume (-119.25 dB To 71.25 dB, 0.75 dB/Step) 0x00: Mute 0x01: -119.25 dB ... 0x9F: -0.75 dB 0xA0: 0 dB 0xA1: 0.75 dB ... 0xFF: 71.25 dB |
| 15:8 | R/W | 0xA0 | ADC2_VOL ADC2 channel volume (-119.25 dB To 71.25 dB, 0.75 dB/Step) 0x00: Mute 0x01: -119.25 dB ... 0x9F: -0.75 dB 0xA0: 0 dB 0xA1: 0.75 dB ... 0xFF: 71.25 dB |
| 7:0 | R/W | 0xA0 | ADC1_VOL ADC1 channel volume (-119.25 dB To 71.25 dB, 0.75 dB/Step) 0x00: Mute 0x01: -119.25 dB ... 0x9F: -0.75 dB 0xA0: 0 dB 0xA1: 0.75 dB ... 0xFF: 71.25 dB |

8.3.6.10 0x0038 ADC FIFO Status Register (Default Value: 0x0000_0001)

| Offset: 0x0038 | | | Register Name: AC_ADC_FIFOS |
|----------------|------------|-------------|-----------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:24 | / | / | / |

| Offset: 0x0038 | | | Register Name: AC_ADC_FIFOS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 23 | R | 0x0 | RXA RX FIFO Available 0: No available data in RX FIFO 1: More than one sample in RX FIFO (>= 1 word) |
| 22:17 | / | / | / |
| 16:8 | R | 0x0 | RXA_CNT RX FIFO Available Sample Word Counter |
| 7:4 | / | / | / |
| 3 | R/W1C | 0x0 | RXA_INT RX FIFO Data Available Pending Interrupt 0: No Pending IRQ 1: Data Available Pending IRQ Write '1' to clear this interrupt or automatic clear if the interrupt condition fails. |
| 2 | / | / | / |
| 1 | R/W1C | 0x0 | RXO_INT RX FIFO Overrun Pending Interrupt 0: No Pending IRQ 1: FIFO Overrun Pending IRQ Write '1' to clear this interrupt. |
| 0 | R | 0x1 | Reserved |

8.3.6.11 0x0040 ADC RX DATA Register (Default Value: 0x0000_0000)

| Offset: 0x0040 | | | Register Name: AC_ADC_RXDATA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R | 0x0 | RX_DATA RX Sample The host can get one sample by reading this register. The left channel sample data comes first and then the right channel sample. |

8.3.6.12 0x0044 ADC RX Counter Register (Default Value: 0x0000_0000)

| Offset: 0x0044 | | | Register Name: AC_ADC_CNT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R/W | 0x0 | <p>RX_CNT RX Sample Counter</p> <p>The audio sample number of writing into RXFIFO. When one sample is written by Digital Audio Engine, the RX sample counter register increases by one. The RX sample counter register can be set to any initial value at any time. After being updated by the initial value, the counter register should count from this initial value.</p> <p>Note: It is used for Audio/Video Synchronization.</p> |

8.3.6.13 0x004C ADC Debug Register (Default Value: 0x0000_0000)

| Offset: 0x004C | | | Register Name: AC_ADC_DG_REG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:25 | / | / | / |
| 24 | R/W | 0x0 | <p>AD_SWP1 ADC output channel swap enable (for digital filter)</p> <p>0: Disabled 1: Enabled</p> <p>Note: ADC1 and ADC2 swap data.</p> |
| 23:0 | / | / | / |

8.3.6.14 0x0050 ADC Digital Control Register (Default Value: 0x0000_0000)

| Offset: 0x0050 | | | Register Name: ADC_DIG_CTRL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:17 | / | / | / |
| 16 | R/W | 0x0 | <p>ADC1_2_VOL_EN ADC1/2 Volume Control Enable</p> <p>0: Disabled 1: Enabled</p> |
| 15:2 | / | / | / |
| 1:0 | R/W | 0x0 | <p>ADC_CHANNEL_EN Bit 1: ADC2 enabled Bit 0: ADC1 enabled</p> |

8.3.6.15 0x0054 VRA1 Speedup Down Control Register (Default Value: 0x0000_0000)

| Offset: 0x0054 | | | Register Name: VRA1SPEEDUP_DOWN_CTRL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:5 | / | / | / |
| 4 | R | 0x0 | VRA1SPEEDUP_DOWN_STATE Only if VAR1SPEEDUP_DOWN_Further_CTRL (0x310[22]) is set 0, VAR1Speedup Down State is valid. 0: VAR1Speedup_Down does not work. 1: VAR1Speedup_Down works. |
| 3:2 | / | / | / |
| 1 | R/W | 0x0 | VRA1SPEEDUP_DOWN_CTRL VAR1Speedup Down Manual Control Enable 0: Disabled. VAR1Speedup Down converts to 1 after the bus rst releases 32 ms. 1: Enabled. VAR1Speedup Down converts to 1 immediately. |
| 0 | R/W | 0x0 | VRA1SPEEDUP_DOWN_RST_CTRL VAR1Speedup Down RST Manual Control Enable 0: Disabled. VAR1Speedup Down converts to 1 after the bus rst releases 32 ms. 1: Enabled. VAR1Speedup Down reset 0 immediately. |

8.3.6.16 0x00F0 DAC DAP Control Register (Default Value: 0x0000_0000)

| Offset: 0x00F0 | | | Register Name: AC_DAC_DAP_CTR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31 | R/W | 0x0 | DDAP_EN DAP for DRC enable 0: Bypassed 1: Enabled |
| 30 | / | / | / |
| 29 | R/W | 0x0 | DDAP_DRC_EN DRC enable control 0: Disabled 1: Enabled |
| 28 | R/W | 0x0 | DDAP_HPF_EN HPF enable control 0: Disabled 1: Enabled |
| 27:0 | / | / | / |

8.3.6.17 0x00F8 ADC DAP Control Register (Default Value: 0x0000_0000)

| Offset: 0x00F8 | | | Register Name: AC_ADC_DAP_CTR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31 | R/W | 0x0 | ADC_DAPO_EN (control the DAP of ADC1/2) DAP for ADC enable 0: Bypassed 1: Enabled |
| 30 | / | / | / |
| 29 | R/W | 0x0 | ADC_DRCO_EN ADC DRCO enable control 0: Disabled 1: Enabled |
| 28 | R/W | 0x0 | ADC_HPF0_EN ADC HPF0 enable control 0: Disabled 1: Enabled |
| 27:0 | / | / | / |

8.3.6.18 0x0100 DAC DRC High HPF Coef Register (Default Value: 0x0000_0OFF)

| Offset: 0x0100 | | | Register Name: AC_DAC_DRC_HHPFC |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:11 | / | / | / |
| 10:0 | R/W | 0xFF | HPF coefficient setting and the data is 3.24 format. |

8.3.6.19 0x0104 DAC DRC Low HPF Coef Register (Default Value: 0x0000_FAC1)

| Offset: 0x0104 | | | Register Name: AC_DAC_DRC_LHPFC |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xFAC1 | HPF coefficient setting and the data is 3.24 format. |

8.3.6.20 0x0108 DAC DRC Control Register (Default Value: 0x0000_0080)

| Offset: 0x0108 | | | Register Name: AC_DAC_DRC_CTRL |
|----------------|------------|-------------|--------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |

| Offset: 0x0108 | | | Register Name: AC_DAC_DRC_CTRL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 15 | R | 0x0 | DRC delay buffer data output state when The DRC delay function is enabled and the DRC function is disabled. After disabling the DRC function and this bit goes to 0, write the DRC delay function bit to 0. 0: Not completed 1: Completed |
| 14:10 | / | / | / |
| 13:8 | R/W | 0x0 | Signal delay time setting 6'h00: (8x1) fs 6'h01: (8x2) fs 6'h02: (8x3) fs ----- 6'h2e: (8*47) fs 6'h2f: (8*48) fs 6'h30 -- 6'h3f: (8*48) fs Delay time = 8*(n + 1) fs, n<6'h30; When the delay function is disabled, the signal delay time is unused. |
| 7 | R/W | 0x1 | DAC_DRC_DELAY_BUF_EN The delay buffer use or not when the DRC is disabled and the DRC buffer data output completely. 0: Do not use the buffer. 1: Use the buffer. |
| 6 | R/W | 0x0 | DAC_DRC_GAIN_MAX_LIMIT_EN DRC gain max limit enable 0: Disabled 1: Enabled |
| 5 | R/W | 0x0 | DAC_DRC_GAIN_MIN_LIMIT_EN DRC gain min limit enable When this function is enabled, it will overwrite the noise detect function. 0: Disabled 1: Enabled |
| 4 | R/W | 0x0 | DAC_DRC_DETECT_NOISE_EN Control the DRC to detect noise when ET is enabled. 0: Disabled 1: Enabled |

| Offset: 0x0108 | | | Register Name: AC_DAC_DRC_CTRL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 3 | R/W | 0x0 | <p>DAC_DRC_SIGNAL_FUNC_SEL Signal function select 0: RMS filter 1: Peak filter</p> <p>When the signal function selects the Peak filter, the RMS parameter is unused. (AC_DRC_LRMSHAT, AC_DRC_LRMSLAT, AC_DRC_LRMSSHAT, AC_DRC_LRMSLAT)</p> <p>When the signal function selects the RMS filter, the Peak filter parameter is unused. (AC_DRC_LPFHAT, AC_DRC_LPFLAT, AC_DRC_RPFHAT, AC_DRC_RPFLAT, AC_DRC_LPFHRT, AC_DRC_LPFLRT, AC_DRC_RPFHRT, and AC_DRC_RPFLRT)</p> |
| 2 | R/W | 0x0 | <p>DAC_DRC_DELAY_FUNC_EN Delay function enable 0: Disabled 1: Enabled</p> <p>When the bit is disabled, the signal delay time is unused.</p> |
| 1 | R/W | 0x0 | <p>DAC_DRC_LT_EN DRC LT enable 0: Disabled 1: Enabled</p> <p>When the bit is disabled, KI and OPL parameter is unused.</p> |
| 0 | R/W | 0x0 | <p>DAC_DRC_ET_EN DRC ET enable 0: Disabled 1: Enabled</p> <p>When the bit is disabled, Ke and OPE parameter is unused.</p> |

8.3.6.21 0x010C DAC DRC Left Peak Filter High Attack Time Coef Register (Default Value: 0x0000_000B)

| Offset: 0x010C | | | Register Name: AC_DAC_DRC_LPFHAT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:11 | / | / | / |
| 10:0 | R/W | 0x000B | <p>DAC_DRC_LPFHAT</p> <p>The left peak filter attack time parameter setting, which is determined by the equation that $AT = 1 - \exp(-2.2Ts/ta)$. The format is 3.24. (The default value is 1 ms)</p> |

8.3.6.22 0x0110 DAC DRC Left Peak Filter Low Attack Time Coef Register (Default Value: 0x0000_77BF)

| Offset: 0x0110 | | | Register Name: AC_DAC_DRC_LPFLAT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x77BF | DAC_DAC_LPFLAT The left peak filter attack time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/ta). The format is 3.24. (The default value is 1 ms) |

8.3.6.23 0x0114 DAC DRC Right Peak Filter High Attack Time Coef Register (Default Value: 0x0000_000B)

| Offset: 0x0114 | | | Register Name: AC_DAC_DRC_RPFHAT |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xB | DAC_DAC_RPFHAT The right peak filter attack time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/ta). The format is 3.24. (The default value is 1 ms) |

8.3.6.24 0x0118 DAC DRC Left Peak Filter Low Attack Time Coef Register (Default Value: 0x0000_77BF)

| Offset: 0x0118 | | | Register Name: AC_DAC_DRC_RPFLAT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x77BF | DAC_DAC_RPFLAT The left peak filter attack time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/ta). The format is 3.24. (The default value is 1 ms) |

8.3.6.25 0x011C DAC DRC Left Peak Filter High Release Time Coef Register (Default Value: 0x0000_00FF)

| Offset: 0x011C | | | Register Name: AC_DAC_DRC_LPFHRT |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:11 | / | / | / |
| 10:0 | R/W | 0x00FF | DAC_DAC_LPFHRT The left peak filter release time parameter setting, which is determined by the equation that RT = exp (-2.2Ts/tr). The format is 3.24. (The default value is 100 ms) |

8.3.6.26 0x0120 DAC DRC Left Peak Filter Low Release Time Coef Register (Default Value: 0x0000_E1F8)

| Offset: 0x0120 | | | Register Name: AC_DAC_DRC_LPFLRT |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xE1F8 | DAC_DAC_LPFLRT The left peak filter release time parameter setting, which is determined by the equation that RT = exp (-2.2Ts/tr). The format is 3.24. (The default value is 100 ms) |

8.3.6.27 0x0124 DAC DRC Right Peak Filter High Release Time Coef Register (Default Value: 0x0000_00FF)

| Offset: 0x0124 | | | Register Name: AC_DAC_DRC_RPFHRT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xFF | DAC_DAC_RPFHRT The right peak filter release time parameter setting, which is determined by the equation that RT = exp (-2.2Ts/tr). The format is 3.24. (The default value is 100 ms) |

8.3.6.28 0x0128 DAC DRC Right Peak filter Low Release Time Coef Register (Default Value: 0x0000_E1F8)

| Offset: 0x0128 | | | Register Name: AC_DAC_DRC_RPFLRT |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xE1F8 | DAC_DAC_RPFLRT The right peak filter release time parameter setting, which is determined by the equation that AT = exp(-2.2Ts/tr). The format is 3.24. (The default value is 100 ms) |

8.3.6.29 0x012C DAC DRC Left RMS Filter High Coef Register (Default Value: 0x0000_0001)

| Offset: 0x012C | | | Register Name: AC_DAC_DRC_LRMSHAT |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:11 | / | / | / |
| 10:0 | R/W | 0x0001 | DAC_DAC_LRMSHAT The left RMS filter average time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/tav). The format is 3.24. (The default value is 10 ms) |

8.3.6.30 0x0130 DAC DRC Left RMS Filter Low Coef Register (Default Value: 0x0000_2BAF)

| Offset: 0x0130 | | | Register Name: AC_DAC_DRC_LRMSLAT |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x2BAF | DAC_DAC_LRMSLAT The left RMS filter average time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/tav). The format is 3.24. (The default value is 10 ms) |

8.3.6.31 0x0134 DAC DRC Right RMS Filter High Coef Register (Default Value: 0x0000_0001)

| Offset: 0x0134 | | | Register Name: AC_DAC_DRC_RRMSHAT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x2BAF | DAC_DAC_RRMSHAT The right RMS filter average time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/tav). The format is 3.24. (The default value is 10 ms) |

8.3.6.32 0x0138 DAC DRC Right RMS Filter Low Coef Register (Default Value: 0x0000_2BAF)

| Offset: 0x0138 | | | Register Name: AC_DAC_DRC_RRMSLAT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x2BAF | DAC_DAC_RRMSLAT The right RMS filter average time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/tav). The format is 3.24. (The default value is 10 ms) |

8.3.6.33 0x013C DAC DRC Compressor Threshold High Setting Register (Default Value: 0x0000_06A4)

| Offset: 0x013C | | | Register Name: AC_DAC_DRC_HCT |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x06A4 | DAC_DAC_HCT The compressor threshold setting, which is set by the equation that CTin = -CT/6.0206. The format is 8.24. (The default value is -40 dB) |

8.3.6.34 0x0140 DAC DRC Compressor Slope High Setting Register (Default Value: 0x0000_D3C0)

| Offset: 0x0140 | | | Register Name: AC_DAC_DRC_LCT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xD3C0 | DAC_DRC_LCT The compressor threshold setting, which is set by the equation that $CTin = -CT/6.0206$. The format is 8.24. (The default value is -40 dB) |

8.3.6.35 0x0144 DAC DRC Compressor Slope High Setting Register (Default Value: 0x0000_0080)

| Offset: 0x0144 | | | Register Name: AC_DAC_DRC_HKC |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x0080 | DAC_DRC_HKC The slope of the compressor, which is determined by the equation that $Kc = 1/R$. R is the ratio of the compressor, which is always an integer. The format is 8.24. (The default value is 2:1) |

8.3.6.36 0x0148 DAC DRC Compressor Slope Low Setting Register (Default Value: 0x0000_0000)

| Offset: 0x0148 | | | Register Name: AC_DAC_DRC_LKC |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x0000 | DAC_DRC_LKC The slope of the compressor, which is determined by the equation that $Kc = 1/R$. R is the ratio of the compressor, which is always an integer. The format is 8.24. (The default value is 2:1) |

8.3.6.37 0x014C DAC DRC Compressor High Output at Compressor Threshold Register (Default Value: 0x0000_F95B)

| Offset: 0x014C | | | Register Name: AC_DAC_DRC_HOPC |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xF95B | DAC_DRC_HOPC The output of the compressor, which is determined by the equation $-OPC/6.0206$. The format is 8.24 (The default value is -40 dB) |

8.3.6.38 0x0150 DAC DRC Compressor Low Output at Compressor Threshold Register (Default Value: 0x0000_2C3F)

| Offset: 0x0150 | | | Register Name: AC_DAC_DRC_LOPC |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x2C3F | DAC_DRC_LOPC The output of the compressor, which is determined by the equation OPC/6.0206. The format is 8.24. (The default value is -40 dB) |

8.3.6.39 0x0154 DAC DRC Limiter Threshold High Setting Register (Default Value: 0x0000_01A9)

| Offset: 0x0154 | | | Register Name: AC_DAC_DRC_HLT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x01A9 | DAC_DRC_HLT The limiter threshold setting, which is set by the equation that LTin = -LT/6.0206, The format is 8.24. (The default value is -10 dB) |

8.3.6.40 0x0158 DAC DRC Limiter Threshold Low Setting Register (Default Value: 0x0000_34F0)

| Offset: 0x0158 | | | Register Name: AC_DAC_DRC_LLT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x34F0 | DAC_DRC_LLT The limiter threshold setting, which is set by the equation that LTin = -LT/6.0206. The format is 8.24. (The default value is -10 dB) |

8.3.6.41 0x015C DAC DRC Limiter Slope High Setting Register (Default Value: 0x0000_0005)

| Offset: 0x015C | | | Register Name: AC_DAC_DRC_HKI |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x0005 | DAC_DRC_HKI The slope of the limiter which is determined by the equation that KI = 1/R. R is the ratio of the limiter, which is always an integer. The format is 8.24. (The default value is <50:1>) |

8.3.6.42 0x0160 DAC DRC Limiter Slope Low Setting Register (Default Value: 0x0000_1EB8)

| Offset: 0x0160 | | | Register Name: AC_DAC_DRC_LKI |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x1EB8 | DAC_DRC_LKI The slope of the limiter, which is determined by the equation that KI = 1/R. R is the ratio of the limiter, which is always an integer. The format is 8.24. (The default value is <50:1>) |

8.3.6.43 0x0164 DAC DRC Limiter High Output at Limiter Threshold Register (Default Value: 0x0000_FBD8)

| Offset: 0x0164 | | | Register Name: AC_DAC_DRC_HOPL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xFBD8 | DAC_DRC_HOPL The output of the limiter, which is determined by equation OPT/6.0206. The format is 8.24. (The default value is -25 dB) |

8.3.6.44 0x0168 DAC DRC Limiter Low Output at Limiter Threshold Register (Default Value: 0x0000_FBA7)

| Offset: 0x0168 | | | Register Name: AC_DAC_DRC_LOPL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xFBA7 | DAC_DRC_LOPL The output of the limiter, which is determined by equation OPT/6.0206. The format is 8.24. (The default value is -25 dB) |

8.3.6.45 0x016C DAC DRC Expander Threshold High Setting Register (Default Value: 0x0000_0BA0)

| Offset: 0x016C | | | Register Name: AC_DAC_DRC_HET |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x0BA0 | DAC_DRC_HET The expander threshold setting, which is set by the equation that ETin = -ET/6.0206. The format is 8.24. (The default value is -70 dB) |

8.3.6.46 0x0170 DAC DRC Expander Threshold Low Setting Register (Default Value: 0x0000_7291)

| Offset: 0x0170 | | | Register Name: AC_DAC_DRC LET |
|----------------|------------|-------------|-------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |

| Offset: 0x0170 | | | Register Name: AC_DAC_DRC_LET |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 15:0 | R/W | 0x7291 | DAC_DRC_LET The expander threshold setting, which is set by the equation that ETin = -ET/6.0206. The format is 8.24. (The default value is -70 dB) |

8.3.6.47 0x0174 DAC DRC Expander Slope High Setting Register (Default Value: 0x0000_0500)

| Offset: 0x0174 | | | Register Name: AC_DAC_DRC_HKE |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:14 | / | / | / |
| 13:0 | R/W | 0x0500 | DAC_DRC_HKE The slope of the expander, which is determined by the equation that Ke = 1/R. R is the ratio of the expander, which is always an integer and the ke must larger than 50. The format is 8.24. (The default value is <1:5>) |

8.3.6.48 0x0178 DAC DRC Expander Slope Low Setting Register (Default Value: 0x0000_0000)

| Offset: 0x0178 | | | Register Name: AC_DAC_DRC_LKE |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x0000 | DAC_DRC_LKE The slope of the expander, which is determined by the equation that Ke = 1/R. R is the ratio of the expander, which is always an integer and the ke must larger than 50. The format is 8.24. (The default value is <1:5>) |

8.3.6.49 0x017C DAC DRC Expander High Output at Expander Threshold Register (Default Value: 0x0000_F45F)

| Offset: 0x017C | | | Register Name: AC_DAC_DRC_HOPE |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xF45F | DAC_DRC_HOPE The output of the expander, which is determined by equation OPE/6.0206. The format is 8.24. (The default value is -70 dB) |

8.3.6.50 0x0180 DAC DRC Expander Low Output at Expander Threshold Register (Default Value: 0x0000_8D6E)

| Offset: 0x0180 | | | Register Name: AC_DAC_DRC_LOPE |
|----------------|------------|-------------|--------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |

| Offset: 0x0180 | | | Register Name: AC_DAC_DRC_LOPE |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 15:0 | R/W | 0x8D6E | DAC_DRC_LOPE The output of the expander which is determined by equation OPE/6.0206. The format is 8.24. (The default value is -70 dB) |

8.3.6.51 0x0184 DAC DRC Linear Slope High Setting Register (Default Value: 0x0000_0100)

| Offset: 0x0184 | | | Register Name: AC_DAC_DRC_HKN |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x0100 | DAC_DRC_HKN The slope of the linear, which is determined by the equation that Kn = 1/R. R is the ratio of the linear, which is always an integer. The format is 8.24. (The default value is <1:1>) |

8.3.6.52 0x0188 DAC DRC Linear Slope Low Setting Register (Default Value: 0x0000_0000)

| Offset: 0x0188 | | | Register Name: AC_DAC_DRC_LKN |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x0000 | DAC_DRC_LKN The slope of the linear, which is determined by the equation that Kn = 1/R. R is the ratio of the linear, which is always an integer. The format is 8.24. (The default value is <1:1>) |

8.3.6.53 0x018C DAC DRC Smooth Filter Gain High Attack Time Coef Register (Default Value: 0x0000_0002)

| Offset: 0x018C | | | Register Name: AC_DAC_DRC_SFHAT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:11 | / | / | / |
| 10:0 | R/W | 0x0002 | DAC_DRC_SFHAT The smooth filter attack time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/tr). The format is 3.24. (The default value is 5 ms) |

8.3.6.54 0x0190 DAC DRC Smooth Filter Gain Low Attack Time Coef Register (Default Value: 0x0000_5600)

| Offset: 0x0190 | | | Register Name: AC_DAC_DRC_SFLAT |
|----------------|------------|-------------|---------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |

| Offset: 0x0190 | | | Register Name: AC_DAC_DRC_SFLAT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 15:0 | R/W | 0x5600 | DAC_DRC_SFLAT The smooth filter attack time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/tr). The format is 3.24. (The default value is 5 ms) |

8.3.6.55 0x0194 DAC DRC Smooth Filter Gain High Release Time Coef Register (Default Value: 0x0000_0000)

| Offset: 0x0194 | | | Register Name: AC_DAC_DRC_SFHRT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:11 | / | / | / |
| 10:0 | R/W | 0x0000 | DAC_DRC_SFHRT The gain smooth filter release time parameter setting, which is determined by the equation that RT = 1-exp (-2.2Ts/tr). The format is 3.24. (The default value is 200 ms) |

8.3.6.56 0x0198 DAC DRC Smooth Filter Gain Low Release Time Coef Register (Default Value: 0x0000_0F04)

| Offset: 0x0198 | | | Register Name: AC_DAC_DRC_SFLRT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x0F04 | DAC_DRC_SFLRT The gain smooth filter release time parameter setting, which is determined by the equation that RT = 1-exp (-2.2Ts/tr). The format is 3.24. (The default value is 200 ms) |

8.3.6.57 0x019C DAC DRC MAX Gain High Setting Register (Default Value: 0x0000_FE56)

| Offset: 0x019C | | | Register Name: AC_DAC_DRC_MXGHS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xFE56 | DAC_DRC_MXGHS The max gain setting, which is determined by equation MXGin=MXG/6.0206. The format is 8.24 and must -20 dB < MXG < 30 dB (The default value is -10 dB) |

8.3.6.58 0x01A0 DAC DRC MAX Gain Low Setting Register (Default Value: 0x0000_CB0F)

| Offset: 0x01A0 | | | Register Name: AC_DAC_DRC_MXGLS |
|----------------|------------|-------------|---------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |

| Offset: 0x01A0 | | | Register Name: AC_DAC_DRC_MXGLS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 15:0 | R/W | 0xCBOF | DAC_DRC_MXGLS The max gain setting, which is determined by equation MXGin=MXG/6.0206. The format is 8.24 and must -20 dB < MXG < 30 dB (The default value is -10 dB) |

8.3.6.59 0x01A4 DAC DRC MIN Gain High Setting Register (Default Value: 0x0000_F95B)

| Offset: 0x01A4 | | | Register Name: AC_DAC_DRC_MNGHS |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xF95B | DAC_DRC_MNGHS The min gain setting, which is determined by equation MXGin=MXG/6.0206. The format is 8.24 and must -60 dB ≤ MNG ≤ -40 dB (The default value is -40 dB) |

8.3.6.60 0x01A8 DAC DRC MIN Gain Low Setting Register (Default Value: 0x0000_2C3F)

| Offset: 0x01A8 | | | Register Name: AC_DAC_DRC_MNGLS |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x2C3F | DAC_DRC_MNGLS The min gain setting, which is determined by equation MXGin=MNG/6.0206. The format is 8.24 and must -60 dB ≤ MNG ≤ -40 dB (The default value is -40 dB) |

8.3.6.61 0x01AC DAC DRC Expander Smooth Time High Coef Register (Default Value: 0x0000_0000)

| Offset: 0x01AC | | | Register Name: AC_DAC_DRC_EPSHC |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:11 | / | / | / |
| 10:0 | R/W | 0x0000 | DAC_DRC_EPSHC The gain smooth filter release and attack time parameter setting in expander region, which are determined by the equation that RT = 1-exp (-2.2Ts/tr). The format is 3.24. (The default value is 30 ms) |

8.3.6.62 0x01B0 DAC DRC Expander Smooth Time Low Coef Register (Default Value: 0x0000_640C)

| Offset: 0x01B0 | | | Register Name: AC_DAC_DRC_EPSLC |
|----------------|------------|-------------|---------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |

| Offset: 0x01B0 | | | Register Name: AC_DAC_DRC_EPSLC |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 15:0 | R/W | 0x640C | DAC_DRC_EPSLC The gain smooth filter release and attack time parameter setting in expander region, which are determined by the equation that $RT = 1 - \exp(-2.2Ts/tr)$. The format is 3.24. (The default value is 30 ms) |

8.3.6.63 0x01B8 DAC DRC HPF Gain High Coef Register (Default Value: 0x0000_0100)

| Offset: 0x01B8 | | | Register Name: AC_DAC_DRC_HPFHGAIN |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:11 | / | / | / |
| 10:0 | R/W | 0x100 | DAC_DRC_HPFHGAIN The gain of HPF coefficient. The format is 3.24. (gain = 1) |

8.3.6.64 0x01BC DAC DRC HPF Gain Low Coef Register (Default Value: 0x0000_0000)

| Offset: 0x01BC | | | Register Name: AC_DAC_DRC_HPFLGAIN |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x0000 | DAC_DRC_HPFLGAIN The gain of HPF coefficient. The format is 3.24. (gain = 1) |

8.3.6.65 0x0200 ADC DRC High HPF Coef Register (Default Value: 0x0000_00FF)

| Offset: 0x0200 | | | Register Name: AC_ADC_DRC_HHPFC |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:11 | / | / | / |
| 10:0 | R/W | 0xFF | ADC_DRC_HHPFC HPF coefficient setting and the data is 3.24 format. |

8.3.6.66 0x0204 ADC DRC Low HPF Coef Register (Default Value: 0x0000_FAC1)

| Offset: 0x0204 | | | Register Name: AC_ADC_DRC_LHPFC |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xFAC1 | HPF coefficient setting and the data is 3.24 format. |

8.3.6.67 0x0208 ADC DRC Control Register (Default Value: 0x0000_0080)

| Offset: 0x0208 | | | Register Name: AC_ADC_DRC_CTRL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15 | R | 0x0 | <p>ADC_DRC_DELAY_BUF_OUTPUT_STATE DRC delay buffer data output state when The DRC delay function is enabled and the DRC function is disabled. After disabled DRC function and this bit goes to 0, the user should write the DRC delay function bit to 0.</p> <p>0: Not completed 1: Completed</p> |
| 14:10 | / | / | / |
| 13:8 | R/W | 0x0 | <p>ADC_DRC_SIGNAL_DELAY_TIME_SET Signal delay time setting 6'h00: (8x1) fs 6'h01: (8x2) fs 6'h02: (8x3) fs</p> <p>-----</p> <p>6'h2e: (8*47) fs 6'h2f: (8*48) fs 6'h30 -- 6'h3f: (8*48) fs Delay time = 8*(n + 1) fs, n < 6'h30; When the delay function is disabled, the signal delay time is unused.</p> |
| 7 | R/W | 0x1 | <p>ADC_DRC_DELAY_BUF_EN The delay buffer use or not when the DRC is disabled and the DRC buffer data output completely.</p> <p>0: Do not use the buffer 1: Use the buffer</p> |
| 6 | R/W | 0x0 | <p>ADC_DRC_GAIN_MAX_LIMIT_EN DRC gain max limit enable</p> <p>0: Disabled 1: Enabled</p> |
| 5 | R/W | 0x0 | <p>ADC_DRC_GAIN_MIN_LIMIT_EN DRC gain min limit enable</p> <p>When this fuction is enabled, it will overwrite the noise detect function.</p> <p>0: Disabled 1: Enabled</p> |
| 4 | R/W | 0x0 | <p>ADC_DRC_DETECT_NOISE_EN Control the DRC to detect noise when ET is enabled</p> <p>0: Disabled 1: Enabled</p> |

| Offset: 0x0208 | | | Register Name: AC_ADC_DRC_CTRL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 3 | R/W | 0x0 | <p>ADC_DRC_SIGNAL_FUNC_SEL Signal function select 0: RMS filter 1: Peak filter</p> <p>When the signal function selects the Peak filter, the RMS parameter is unused. (AC_DRC_LRMSHAT, AC_DRC_LRMSLAT, AC_DRC_LRMSHAT, and AC_DRC_LRMSLAT)</p> <p>When the signal function selects the RMS filter, the Peak filter parameter is unused. (AC_DRC_LPFHAT, AC_DRC_LPFLAT, AC_DRC_RPFHAT, AC_DRC_RPFLAT, AC_DRC_LPFHRT, AC_DRC_LPFLRT, AC_DRC_RPFHRT, and AC_DRC_RPFLRT)</p> |
| 2 | R/W | 0x0 | <p>ADC_DRC_DELAY_FUNC_EN Delay function enable 0: Disabled 1: Enabled</p> <p>When the bit is disabled, the signal delay time is unused.</p> |
| 1 | R/W | 0x0 | <p>ADC_DRC_LT_EN DRC LT enable 0: Disabled 1: Enabled</p> <p>When the bit is disabled, KI and OPL parameter is unused.</p> |
| 0 | R/W | 0x0 | <p>ADC_DRC_ET_EN DRC ET enable 0: Disabled 1: Enabled</p> <p>When the bit is disabled, Ke and OPE parameter is unused.</p> |

8.3.6.68 0x020C ADC DRC Left Peak Filter High Attack Time Coef Register (Default Value: 0x0000_000B)

| Offset: 0x020C | | | Register Name: AC_ADC_DRC_LPFHAT |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:11 | / | / | / |
| 10:0 | R/W | 0x000B | <p>ADC_DRC_LPFHAT</p> <p>The left peak filter attack time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/ta). The format is 3.24. (The default value is 1 ms)</p> |

8.3.6.69 0x0210 ADC DRC Left Peak Filter Low Attack Time Coef Register (Default Value: 0x0000_77BF)

| Offset: 0x0210 | | | Register Name: AC_ADC_DRC_LPFLAT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x77BF | ADC_DRC_LPFLAT The left peak filter attack time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/ta). The format is 3.24. (The default value is 1 ms) |

8.3.6.70 0x0214 ADC DRC Right Peak Filter High Attack Time Coef Register (Default Value: 0x0000_000B)

| Offset: 0x0214 | | | Register Name: AC_ADC_DRC_RPFHAT |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x000B | ADC_DRC_RPFHAT The right peak filter attack time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/ta). The format is 3.24. (The default value is 1 ms) |

8.3.6.71 0x0218 ADC DRC Right Peak Filter Low Attack Time Coef Register (Default Value: 0x0000_77BF)

| Offset: 0x0218 | | | Register Name: AC_ADC_DRC_RPFLAT |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x77BF | ADC_DRC_RPFLAT The right peak filter attack time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/ta). The format is 3.24. (The default value is 1 ms) |

8.3.6.72 0x021C ADC DRC Left Peak Filter High Release Time Coef Register (Default Value: 0x0000_00FF)

| Offset: 0x021C | | | Register Name: AC_ADC_DRC_LPFHRT |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:11 | / | / | / |
| 10:0 | R/W | 0x00FF | ADC_DRC_LPFHRT The left peak filter release time parameter setting, which is determined by the equation that RT = exp (-2.2Ts/tr). The format is 3.24. (The default value is 100 ms) |

8.3.6.73 0x0220 ADC DRC Left Peak Filter Low Release Time Coef Register (Default Value: 0x0000_E1F8)

| Offset: 0x0220 | | | Register Name: AC_ADC_DRC_LPFLRT |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xE1F8 | ADC_DRC_LPFLRT The left peak filter release time parameter setting, which is determined by the equation that RT = exp (-2.2Ts/tr). The format is 3.24. (The default value is 100 ms) |

8.3.6.74 0x0224 ADC DRC Right Peak Filter High Release Time Coef Register (Default Value: 0x0000_00FF)

| Offset: 0x0224 | | | Register Name: AC_ADC_DRC_RPFHRT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x00FF | ADC_DRC_RPFHRT The right peak filter release time parameter setting, which is determined by the equation that RT = exp (-2.2Ts/tr). The format is 3.24. (The default value is 100 ms) |

8.3.6.75 0x0228 ADC DRC Right Peak Filter Low Release Time Coef Register (Default Value: 0x0000_E1F8)

| Offset: 0x0228 | | | Register Name: AC_ADC_DRC_RPFLRT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xE1F8 | ADC_DRC_RPFLRT The right peak filter release time parameter setting, which is determined by the equation that RT = exp (-2.2Ts/tr). The format is 3.24. (The default value is 100 ms) |

8.3.6.76 0x022C ADC DRC Left RMS Filter High Coef Register (Default Value: 0x0000_0001)

| Offset: 0x022C | | | Register Name: AC_ADC_DRC_LRMSHAT |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:11 | / | / | / |
| 10:0 | R/W | 0x0001 | ADC_DRC_LRMSHAT The left RMS filter average time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/tav). The format is 3.24. (The default value is 10 ms) |

8.3.6.77 0x0230 ADC DRC Left RMS Filter Low Coef Register (Default Value: 0x0000_2BAF)

| Offset: 0x0230 | | | Register Name: AC_ADC_DRC_LRMSLAT |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x2BAF | ADC_DRC_LRMSLAT The left RMS filter average time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/tav). The format is 3.24. (The default value is 10 ms) |

8.3.6.78 0x0234 ADC DRC Right RMS Filter High Coef Register (Default Value: 0x0000_0001)

| Offset: 0x0234 | | | Register Name: AC_ADC_DRC_RRMSHAT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:11 | / | / | / |
| 10:0 | R/W | 0x0001 | ADC_DRC_RRMSHAT The right RMS filter average time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/tav). The format is 3.24. (The default value is 10 ms) |

8.3.6.79 0x0238 ADC DRC Right RMS Filter Low Coef Register (Default Value: 0x0000_2BAF)

| Offset: 0x0238 | | | Register Name: AC_ADC_DRC_RRMSLAT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x2BAF | ADC_DRC_RRMSLAT The right RMS filter average time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/tav). The format is 3.24. (The default value is 10 ms) |

8.3.6.80 0x023C ADC DRC Compressor Threshold High Setting Register (Default Value: 0x0000_06A4)

| Offset: 0x023C | | | Register Name: AC_ADC_DRC_HCT |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x06A4 | ADC_DRC_HCT The compressor threshold setting, which is set by the equation that CTin = -CT/6.0206. The format is 8.24. (The default value is -40 dB) |

8.3.6.81 0x0240 ADC DRC Compressor Slope High Setting Register (Default Value: 0x0000_D3C0)

| Offset: 0x0240 | | | Register Name: AC_ADC_DRC_LCT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xD3C0 | ADC_DRC_LCT The compressor threshold setting, which is set by the equation that $CTin = -CT/6.0206$. The format is 8.24. (The default value is -40 dB) |

8.3.6.82 0x0244 ADC DRC Compressor Slope High Setting Register (Default Value: 0x0000_0080)

| Offset: 0x0244 | | | Register Name: AC_ADC_DRC_HKC |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x0080 | ADC_DRC_HKC The slope of the compressor which is determined by the equation that $Kc = 1/R$. R is the ratio of the compressor, which is always an integer. The format is 8.24. (The default value is <2:1>) |

8.3.6.83 0x0248 ADC DRC Compressor Slope Low Setting Register (Default Value: 0x0000_0000)

| Offset: 0x0248 | | | Register Name: AC_ADC_DRC_LKC |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x0000 | ADC_DRC_LKC The slope of the compressor, which is determined by the equation that $Kc = 1/R$. R is the ratio of the compressor, which is always an integer. The format is 8.24. (The default value is <2:1>) |

8.3.6.84 0x024C ADC DRC Compressor High Output at Compressor Threshold Register (Default Value: 0x0000_F95B)

| Offset: 0x024C | | | Register Name: AC_ADC_DRC_HOPC |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xF95B | ADC_DRC_HOPC The output of the compressor, which is determined by the equation $-OPC/6.0206$. The format is 8.24. (The default value is -40 dB) |

8.3.6.85 0x0250 ADC DRC Compressor Low Output at Compressor Threshold Register (Default Value: 0x0000_2C3F)

| Offset: 0x0250 | | | Register Name: AC_ADC_DRC_LOPC |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x2C3F | ADC_DRC_LOPC The output of the compressor, which is determined by the equation OPC/6.0206. The format is 8.24. (The default value is -40 dB) |

8.3.6.86 0x0254 ADC DRC Limiter Threshold High Setting Register (Default Value: 0x0000_01A9)

| Offset: 0x0254 | | | Register Name: AC_ADC_DRC_HLT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x01A9 | ADC_DRC_HLT The limiter threshold setting, which is set by the equation that LTin = -LT/6.0206. The format is 8.24. (The default value is -10 dB) |

8.3.6.87 0x0258 ADC DRC Limiter Threshold Low Setting Register (Default Value: 0x0000_34F0)

| Offset: 0x0258 | | | Register Name: AC_ADC_DRC_LLT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x34F0 | ADC_DRC_LLT The limiter threshold setting, which is set by the equation that LTin = -LT/6.0206. The format is 8.24. (The default value is -10 dB) |

8.3.6.88 0x025C ADC DRC Limiter Slope High Setting Register (Default Value: 0x0000_0005)

| Offset: 0x025C | | | Register Name: AC_ADC_DRC_HKI |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:14 | / | / | / |
| 13:0 | R/W | 0x0005 | ADC_DRC_HKI The slope of the limiter, which is determined by the equation that KI = 1/R. R is the ratio of the limiter, which is always an integer. The format is 8.24. (The default value is <50:1>) |

8.3.6.89 0x0260 ADC DRC Limiter Slope Low Setting Register (Default Value: 0x0000_1EB8)

| Offset: 0x0260 | | | Register Name: AC_ADC_DRC_LKI |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x1EB8 | ADC_DRC_LKI The slope of the limiter, which is determined by the equation that KI = 1/R. R is the ratio of the limiter, which is always an integer. The format is 8.24. (The default value is <50:1>) |

8.3.6.90 0x0264 ADC DRC Limiter High Output at Limiter Threshold Register (Default Value: 0x0000_FBD8)

| Offset: 0x0264 | | | Register Name: AC_ADC_DRC_HOPL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xFBD8 | ADC_DRC_HOPL The output of the limiter, which is determined by equation OPT/6.0206. The format is 8.24. (The default value is -25 dB) |

8.3.6.91 0x0268 ADC DRC Limiter Low Output at Limiter Threshold Register (Default Value: 0x0000_FBA7)

| Offset: 0x0268 | | | Register Name: AC_ADC_DRC_LOPL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xFBA7 | ADC_DRC_LOPL The output of the limiter which is determined by equation OPT/6.0206. The format is 8.24. (The default value is -25 dB) |

8.3.6.92 0x026C ADC DRC Expander Threshold High Setting Register (Default Value: 0x0000_0BA0)

| Offset: 0x026C | | | Register Name: AC_ADC_DRC_HET |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x0BA0 | ADC_DRC_HET The expander threshold setting, which is set by the equation that ETin = -ET/6.0206, The format is 8.24. (The default value is -70 dB) |

8.3.6.93 0x0270 ADC DRC Expander Threshold Low Setting Register (Default Value: 0x0000_7291)

| Offset: 0x0270 | | | Register Name: AC_ADC_DRC LET |
|----------------|------------|-------------|-------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |

| Offset: 0x0270 | | | Register Name: AC_ADC_DRC_LET |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 15:0 | R/W | 0x7291 | ADC_DRC_LET The expander threshold setting, which is set by the equation that ETin = -ET/6.0206, The format is 8.24. (The default value is -70 dB) |

8.3.6.94 0x0274 ADC DRC Expander Slope High Setting Register (Default Value: 0x0000_0500)

| Offset: 0x0274 | | | Register Name: AC_ADC_DRC_HKE |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x0500 | ADC_DRC_HKE The slope of the expander, which is determined by the equation that Ke = 1/R. R is the ratio of the expander, which is always an integer and the ke must larger than 50. The format is 8.24. (The default value is <1:5>) |

8.3.6.95 0x0278 ADC DRC Expander Slope Low Setting Register (Default Value: 0x0000_0000)

| Offset: 0x0278 | | | Register Name: AC_ADC_DRC_LKE |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x0000 | ADC_DRC_LKE The slope of the expander, which is determined by the equation that Ke = 1/R. R is the ratio of the expander, which is always an integer and the ke must be larger than 50. The format is 8.24. (The default value is <1:5>) |

8.3.6.96 0x027C ADC DRC Expander High Output at Expander Threshold Register (Default Value: 0x0000_F45F)

| Offset: 0x027C | | | Register Name: AC_ADC_DRC_HOPE |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xF45F | ADC_DRC_HOPE The output of the expander, which is determined by equation OPE/6.0206. The format is 8.24. (The default value is -70 dB) |

8.3.6.97 0x0280 ADC DRC Expander Low Output at Expander Threshold Register (Default Value: 0x0000_8D6E)

| Offset: 0x0280 | | | Register Name: AC_ADC_DRC_LOPE |
|----------------|------------|-------------|--------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |

| Offset: 0x0280 | | | Register Name: AC_ADC_DRC_LOPE |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 15:0 | R/W | 0x8D6E | ADC_DRC_LOPE The output of the expander, which is determined by equation OPE/6.0206. The format is 8.24. (The default value is -70 dB) |

8.3.6.98 0x0284 ADC DRC Linear Slope High Setting Register (Default Value: 0x0000_0100)

| Offset: 0x0284 | | | Register Name: AC_ADC_DRC_HKN |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x0100 | ADC_DRC_HKN The slope of the linear, which is determined by the equation that Kn = 1/R. R is the ratio of the linear, which is always an integer. The format is 8.24. (The default value is <1:1>) |

8.3.6.99 0x0288 ADC DRC Linear Slope Low Setting Register (Default Value: 0x0000_0000)

| Offset: 0x0288 | | | Register Name: AC_ADC_DRC_LKN |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x0000 | ADC_DRC_LKN The slope of the linear, which is determined by the equation that Kn = 1/R. R is the ratio of the linear, which is always an integer. The format is 8.24. (The default value is <1:1>) |

8.3.6.100 0x028C ADC DRC Smooth Filter Gain High Attack Time Coef Register (Default Value: 0x0000_0002)

| Offset: 0x028C | | | Register Name: AC_ADC_DRC_SFHAT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:11 | / | / | / |
| 10:0 | R/W | 0x0002 | ADC_DRC_SFHAT The smooth filter attack time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/tr). The format is 3.24. (The default value is 5 ms) |

8.3.6.101 0x0290 ADC DRC Smooth Filter Gain Low Attack Time Coef Register (Default Value: 0x0000_5600)

| Offset: 0x0290 | | | Register Name: AC_ADC_DRC_SFLAT |
|----------------|------------|-------------|---------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |

| Offset: 0x0290 | | | Register Name: AC_ADC_DRC_SFLAT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 15:0 | R/W | 0x5600 | ADC_DRC_SFLAT The smooth filter attack time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/tr). The format is 3.24. (The default value is 5 ms) |

8.3.6.102 0x0294 ADC DRC Smooth Filter Gain High Release Time Coef Register (Default Value: 0x0000_0000)

| Offset: 0x0294 | | | Register Name: AC_ADC_DRC_SFHRT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:11 | / | / | / |
| 10:0 | R/W | 0x0000 | ADC_DRC_SFHRT The gain smooth filter release time parameter setting, which is determined by the equation that RT = 1-exp (-2.2Ts/tr). The format is 3.24. (The default value is 200 ms) |

8.3.6.103 0x0298 ADC DRC Smooth Filter Gain Low Release Time Coef Register (Default Value: 0x0000_OF04)

| Offset: 0x0298 | | | Register Name: AC_ADC_DRC_SFLRT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xOF04 | ADC_DRC_SFLRT The gain smooth filter release time parameter setting, which is determined by the equation that RT = 1-exp (-2.2Ts/tr). The format is 3.24. (The default value is 200 ms) |

8.3.6.104 0x029C ADC DRC MAX Gain High Setting Register (Default Value: 0x0000_FE56)

| Offset: 0x029C | | | Register Name: AC_ADC_DRC_MXGHS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xFE56 | ADC_DRC_MXGHS The max gain setting, which is determined by equation MXGin=MXG/6.0206. The format is 8.24 and must -20 dB < MXG < 30 dB (The default value is -10 dB) |

8.3.6.105 0x02A0 ADC DRC MAX Gain Low Setting Register (Default Value: 0x0000_CB0F)

| Offset: 0x02A0 | | | Register Name: AC_ADC_DRC_MXGLS |
|----------------|------------|-------------|---------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |

| Offset: 0x02A0 | | | Register Name: AC_ADC_DRC_MXGLS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 15:0 | R/W | 0xCBOF | ADC_DRC_MXGLS The max gain setting, which is determined by equation MXGin=MXG/6.0206. The format is 8.24 and must -20 dB < MXG < 30 dB (The default value is -10 dB) |

8.3.6.106 0x02A4 ADC DRC MIN Gain High Setting Register (Default Value: 0x0000_F95B)

| Offset: 0x02A4 | | | Register Name: AC_ADC_DRC_MNGHS |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0xF95B | ADC_DRC_MNGHS The min gain setting, which is determined by equation MXGin=MXG/6.0206. The format is 8.24 and must -60 dB ≤ MNG ≤ -40 dB (The default value is -40 dB) |

8.3.6.107 0x02A8 ADC DRC MIN Gain Low Setting Register (Default Value: 0x0000_2C3F)

| Offset: 0x02A8 | | | Register Name: AC_ADC_DRC_MNGLS |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x2C3F | ADC_DRC_MNGLS The min gain setting, which is determined by equation MXGin=MNG/6.0206. The format is 8.24 and must -60 dB ≤ MNG ≤ -40 dB (The default value is -40 dB) |

8.3.6.108 0x02AC ADC DAP Expander Smooth Time High Coef Register (Default Value: 0x0000_0000)

| Offset: 0x02AC | | | Register Name: AC_ADC_DRC_EPSHC |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:11 | / | / | / |
| 10:0 | R/W | 0x0000 | ADC_DRC_EPSHC The gain smooth filter release and attack time parameter setting in expander region, which is determined by the equation that RT = 1-exp (-2.2Ts/tr). The format is 3.24. (The default value is 30 ms) |

8.3.6.109 0x02B0 ADC DRC Expander Smooth Time Low Coef Register (Default Value: 0x0000_640C)

| Offset: 0x02B0 | | | Register Name: AC_ADC_DRC_EPSLC |
|----------------|------------|-------------|---------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |

| Offset: 0x02B0 | | | Register Name: AC_ADC_DRC_EPSLC |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 15:0 | R/W | 0x640C | ADC_DRC_EPSLC The gain smooth filter release and attack time parameter setting in expander region, which is determined by the equation that RT = 1-exp (-2.2Ts/tr). The format is 3.24. (The default value is 30 ms) |

8.3.6.110 0x02B8 ADC DRC HPF Gain High Coef Register (Default Value: 0x0000_0100)

| Offset: 0x02B8 | | | Register Name: AC_ADC_DRC_HPFHGAIN |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:11 | / | / | / |
| 10:0 | R/W | 0x100 | ADC_DRC_HPFHGAIN The gain of HPF coefficient setting, which format is 3.24. (gain = 1) |

8.3.6.111 0x02BC ADC DRC HPF Gain Low Coef Register (Default Value: 0x0000_0000)

| Offset: 0x02BC | | | Register Name: AC_ADC_DRC_HPFLGAIN |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0x0000 | ADC_DRC_HPFLGAIN The gain of HPF coefficient setting, which format is 3.24. (gain = 1) |

8.3.6.112 0x0300 ADC1 Analog Control Register (Default Value: 0x001C_C055)

| Offset: 0x0300 | | | Register Name: ADC1_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31 | R/W | 0x0 | ADC1_EN ADC1 Channel Enable 0: Disabled 1: Enabled |
| 30 | R/W | 0x0 | MIC1_PGA_EN MIC1 PGA Enable 0: Disable 1: Enable |
| 29 | R/W | 0x0 | ADC1 Dither Control 0: New Dither Off 1: New Dither On |
| 28 | R/W | 0x0 | MIC1_SIN_EN MIC1 Single Input Enable 0:Disable 1:Enable |

| Offset: 0x0300 | | | Register Name: ADC1_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 27:26 | / | / | / |
| 25:24 | R/W | 0x0 | DSM_DITHER_LVL Dither Level Control (Dither level is positive ralated to the ctrl bits) 00: No Level 01: Min Level 10: Middle Level 11: Max Level |
| 23:22 | / | / | / |
| 21:20 | R/W | 0x1 | IOPBUFFER PGA Vcm Buffer OP Bias Current Select 00: 6 uA 01: 7 uA 10: 8 uA 11: 9 uA |
| 19:18 | R/W | 0x3 | ADC1_PGA_CTRL_RCM ADC1 PGA Common Mode Input Impedance Control for MICIN 00: 100 kΩ 01: 75 kΩ 10: 50 kΩ 11: 25 kΩ |
| 17:16 | R/W | 0x0 | ADC1_PGA_IN_VCM_CTRL ADC1 PGA Common-Mode Voltage Control 00: 900 mV 01: 800 mV 10: 750 mV 11: 700 mV |
| 15:14 | R/W | 0x3 | IOPADC ADC1-ADC2 Bias Current Select 00: 1 uA 01: 2 uA 10: 3 uA 11: 4 uA |
| 13 | R/W | 0x0 | ADC1_SINGLE_NOISE_CONTROL 0: Disable 1: Enable |

| Offset: 0x0300 | | | Register Name: ADC1_REG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 12:8 | R/W | 0x0 | <p>ADC1_PGA_GAIN_CTRL ADC1 PGA gain settings: 0x0: 0 dB 0x10: 21 dB 0x1: 6 dB 0x11: 22 dB 0x2: 6 dB 0x12: 23 dB 0x3: 6 dB 0x13: 24 dB 0x4: 9 dB 0x14: 25 dB 0x5: 10 dB 0x15: 26 dB 0x6: 11 dB 0x16: 27 dB 0x7: 12 dB 0x17: 28 dB 0x8: 13 dB 0x18: 29 dB 0x9: 14 dB 0x19: 30 dB 0xA: 15 dB 0x1A: 31 dB 0xB: 16 dB 0x1B: 32 dB 0xC: 17 dB 0x1C: 33 dB 0xD: 18 dB 0x1D: 34 dB 0xE: 19 dB 0x1E: 35 dB 0xF: 20 dB 0x1F: 36 dB</p> |
| 7:6 | R/W | 0x1 | <p>ADC1_IOPAAF ADC1 OP AAF Bias Current Select 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA. For example: ADC1_REG<15:14> = 11, IOPADC = 4 uA 00: 1.50*4 uA = 6 uA 01: 1.75*4 uA = 7 uA 10: 2.00*4 uA = 8 uA 11: 2.25*4 uA = 9 uA</p> |
| 5:4 | R/W | 0x1 | <p>ADC1_IOPSDM1 ADC1 OP SDM Bias Current Select 1 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA.</p> |

| Offset: 0x0300 | | | Register Name: ADC1_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 3:2 | R/W | 0x1 | ADC1_IOPSDM2 ADC1 OP SDM Bias Current Select 2 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA. |
| 1:0 | R/W | 0x1 | ADC1_IOPMIC ADC1 OP MIC Bias Current Select 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA. |

8.3.6.113 0x0304 ADC2 Analog Control Register (Default Value: 0x001C_0055)

| Offset: 0x0304 | | | Register Name: ADC2_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31 | R/W | 0x0 | ADC2_EN ADC2 Channel Enable 0: Disabled 1: Enabled |
| 30 | R/W | 0x0 | MIC2_PGA_EN MIC2 PGA Enable 0: Disable 1: Enable |
| 29 | R/W | 0x0 | ADC2 Dither Control 0: New Dither Off 1: New Dither On |
| 28 | R/W | 0x0 | MIC2_SIN_EN MIC2 Single Input Enable 0:Disable 1:Enable |
| 27:26 | / | / | / |
| 25:24 | R/W | 0x0 | DSM_DITHER_LVL Dither Level Control (Dither level is positive ralated to the ctrl bits) 00: No Level 01: Min Level 10: Middle Level 11: Max Level |
| 23:22 | / | / | / |

| Offset: 0x0304 | | | Register Name: ADC2_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 21:20 | R/W | 0x1 | IOPBUFFER PGA Vcm Buffer OP Bias Current Select 00: 6 uA 01: 7 uA 10: 8 uA 11: 9 uA |
| 19:18 | R/W | 0x3 | ADC2_PGA_CTRL_RCM ADC2 PGA Common Mode Input Impedance Control for MICIN 00: 100 kΩ 01: 75 kΩ 10: 50 kΩ 11: 25 kΩ |
| 17:16 | R/W | 0x0 | ADC2_PGA_IN_VCM_CTRL ADC2 PGA Common-Mode Voltage Control 00: 900 mV 01: 800 mV 10: 750 mV 11: 700 mV |
| 15:14 | / | / | / |
| 13 | R/W | 0x0 | ADC2_SINGLE_NOISE_CONTROL 0: Disable 1: Enable |
| 12:8 | R/W | 0x0 | ADC2_PGA_GAIN_CTRL ADC2 PGA Gain Settings 0x0: 0 dB 0x10: 21 dB 0x1: 6 dB 0x11: 22 dB 0x2: 6 dB 0x12: 23 dB 0x3: 6 dB 0x13: 24 dB 0x4: 9 dB 0x14: 25 dB 0x5: 10 dB 0x15: 26 dB 0x6: 11 dB 0x16: 27 dB 0x7: 12 dB 0x17: 28 dB 0x8: 13 dB 0x18: 29 dB 0x9: 14 dB 0x19: 30 dB 0xA: 15 dB 0x1A: 31 dB 0xB: 16 dB 0x1B: 32 dB 0xC: 17 dB 0x1C: 33 dB 0xD: 18 dB 0x1D: 34 dB 0xE: 19 dB 0x1E: 35 dB 0xF: 20 dB 0x1F: 36 dB |

| Offset: 0x0304 | | | Register Name: ADC2_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 7:6 | R/W | 0x1 | ADC2_IOPAAF ADC2 OP AAF Bias Current Select 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA. |
| 5:4 | R/W | 0x1 | ADC2_IOPSDM1 ADC2 OP SDM Bias Current Select 1 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA. |
| 3:2 | R/W | 0x1 | ADC2_IOPSDM2 ADC2 OP SDM Bias Current Select 2 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA. |
| 1:0 | R/W | 0x1 | ADC2_IOPMIC ADC2 OP MIC Bias Current Select 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA. |

8.3.6.114 0x0310 DAC Analog Control Register (Default Value: 0x0015_0000)

| Offset: 0x0310 | | | Register Name: DAC_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:26 | / | / | / |
| 25:24 | R/W | 0x0 | P_CURRENT_TEST_SELECT Internal Current Sink Test Enable (from LINEOUTP pin) 00: Normal 01: TEST IP20U_TEST1 |

| Offset: 0x0310 | | | Register Name: DAC_REG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 23:22 | R/W | 0x0 | <p>N_CURRENT_TEST_SELECT Internal Current Sink Test Enable (from LINEOUTP pin)</p> <p>00: Normal 01: TEST IN10U_TEST1 10: TEST INP10U_TVIN_CVBSIN 11: test two current together</p> |
| 21:20 | R/W | 0x1 | <p>IOPVRS VRA2 Buffer OP and Headphone Feedback Buffer OP Bias Current Select</p> <p>00: 6 uA 01: 7 uA 10: 8 uA 11: 9 uA</p> |
| 19:18 | R/W | 0x1 | <p>ILINEOUTAMPS LINEOUTLL/R AMP Bias Current Select</p> <p>00: 6 uA 01: 7 uA 10: 8 uA 11: 9 uA</p> |
| 17:16 | R/W | 0x1 | <p>IOPDACS OPDACL/R Bias Current Select</p> <p>00: 6 uA 01: 7 uA 10: 8 uA 11: 9 uA</p> |
| 15 | R/W | 0x0 | <p>DACL_EN DACL Enable</p> <p>0: Disabled 1: Enabled</p> |
| 14 | / | / | / |
| 13 | R/W | 0x0 | <p>LINEOUTLEN Left Channel LINEOUT Enable</p> <p>0: Disable 1: Enable</p> |
| 12 | R/W | 0x0 | <p>LMUTE DACL to Left Channel LINEOUT Mute Control</p> <p>0: Mute 1: Not mute</p> |
| 11:9 | / | / | / |

| Offset: 0x0310 | | | Register Name: DAC_REG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 8:7 | R/W | 0x0 | VRA2_OPVR_OI_CTRL VRA2 Feedback Buffer Output Current Select 00: 35I 01: 28I 10: 45I 11: 38I I=7uA |
| 6 | R/W | 0x0 | LINEOUTL_DIFFEN Left Channel LINEOUT Output Control 0: Single-End 1: Differential |
| 5 | / | / | / |
| 4:0 | R/W | 0x0 | LINEOUT_VOL_CTRL LINEOUT Volume Control. Total 30 level from 0x1F to 0x02 with the volume 0 dB to -43.5 dB, -1.5 dB/step, mute when 00000 & 00001. |

8.3.6.115 0x0318 MICBIAS Analog Control Register (Default Value: 0x0000_3030)

| Offset: 0x0318 | | | Register Name: MICBIAS_REG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7 | R/W | 0x0 | MMICBIASEN Master Microphone Bias Enable 0: Disabled 1: Enabled |
| 6:5 | R/W | 0x1 | MBIASSEL MMICBIAS Voltage Level Select 00: 1.88 V 01: 2.09 V 10: 2.33 V 11: 2.50 V |
| 4 | R/W | 0x1 | MMIC_BIAS_CHOPPER_EN MMIC BIAS Chopper Enable 0: Disabled 1: Enabled |
| 3:2 | R/W | 0x0 | MMIC_BIAS_CHOPPER_CLK_SEL MMIC BIAS Chopper Clock Select 00: 250 kHz 01: 500 kHz 10: 1 MHz 11: 2 MHz |

| Offset: 0x0318 | | | Register Name: MICBIAS_REG |
|----------------|------------|-------------|----------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 1:0 | / | / | / |

8.3.6.116 0x0320 BIAS Analog Control Register (Default Value: 0x0000_0080)

The register is not controlled by the clock and reset of Audio Codec, only controlled by the clock and reset of system bus.

| Offset: 0x0320 | | | Register Name: BIAS_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:0 | R/W | 0x80 | BIASDATA Bias Current Register Setting Data |

8.3.6.117 0x0348 POWER Analog Control Register (Default Value:0x8000_3019)

The register is not controlled by the clock and reset of Audio Codec, only controlled by the clock and reset of system bus.

| Offset: 0x0348 | | | Register Name: POWER_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31 | R/W | 0x1 | ALDO_EN ALDO Enable 0: Disabled 1: Enabled |
| 30 | / | / | / |
| 29 | R/W | 0x0 | VAR1SPEEDUP_DOWN_Further_CTRL VAR1 Speedup Down Further Control In Adda Analog 0: The digital logic signal input by the digital-analog interface pin controls the var1_speedup_down function (that is, the var1 signal is rapidly pulled up/down) 1: Writing 1 can finish the var1_speedup_down function (ignore the control of the digital-analog interface pin) |
| 28:16 | / | / | / |
| 15 | R | 0x0 | BG_BUFFER_DISABLE BG Output Buffer Disable Control 0: Enable 1: Disable |

| Offset: 0x0348 | | | Register Name: POWER_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 14:12 | R/W | 0x3 | <p>ALDO_OUTPUT_VOLTAGE ALDO Output Voltage Control 000: 2.03 V 001: 1.95 V 010: 1.87 V 011: 1.80 V 100: 1.73 V 101: 1.67 V 110: 1.61 V 111: 1.56 V</p> |
| 11:8 | R/W | 0x0 | <p>BG_ROUGH_TRIM BG Output Voltage Rough Trim Every step is 30mV. Output voltage settings: 0x0: 850mV 0x8: 940mV 0x1: 730mV 0x9: 970mV 0x2: 760mV 0xA: 1000mV 0x3: 790mV 0xB: 1030mV 0x4: 820mV 0xC: 1060mV 0x5: 700mV 0xD: 1090mV 0x6: 880mV 0xE: 1120mV 0x7: 910mV 0xF: 1150mV</p> |
| 7:0 | R/W | 0x25 | <p>BG_FINE_TRIM BG Output Voltage Fine Trim Only low 6-bit is used and every step is 0.8mV. The BG output voltage range is from rough trim value to rough trim value+0.8mV*63. Note: this register only controlled by system bus clock and reset.</p> |

8.3.6.118 0x034C ADC Current Analog Control Register (Default Value: 0x0000_1515)

| Offset: 0x034C | | | Register Name: ADC_CUR_REG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:14 | / | / | / |
| 13:12 | R/W | 0x1 | <p>ADC2_IOPMIC2 ADC2 OP MIC2 Bias Current Select 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA.</p> |

| Offset: 0x034C | | | Register Name: ADC_CUR_REG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 11:10 | R/W | 0x1 | ADC2_OUTPUT_CURRENT ADC2 OP MIC1 Output Current Select 00: 15I 01: 20I 10: 35I 11: 40I I=7 uA |
| 9:8 | R/W | 0x1 | ADC2_OUTPUT_CURRENT ADC2 OP MIC2 Output Current Select 00: 15I 01: 20I 10: 35I 11: 40I I=7 uA |
| 7:6 | / | / | / |
| 5:4 | R/W | 0x1 | ADC1_IOPMIC2 ADC1 OP MIC2 Bias Current Select 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA. |
| 3:2 | R/W | 0x1 | ADC1_OUTPUT_CURRENT ADC1 OP MIC1 Output Current Select 00: 15I 01: 20I 10: 35I 11: 40I I=7 uA |
| 1:0 | R/W | 0x1 | ADC1_OUTPUT_CURRENT ADC1 OP MIC2 Output Current Select 00: 15I 01: 20I 10: 35I 11: 40I I=7 uA |

Contents

| | | |
|--------|---|-----|
| 9 | EMAC..... | 722 |
| 9.1 | Overview | 722 |
| 9.2 | Block Diagram | 723 |
| 9.3 | Functional Description | 723 |
| 9.3.1 | External Signals | 723 |
| 9.3.2 | Clock Sources | 724 |
| 9.3.3 | Typical Application | 724 |
| 9.3.4 | EMAC RX/TX Descriptor | 724 |
| 9.3.5 | Transmit Descriptor | 725 |
| 9.3.6 | Receive Descriptor | 726 |
| 9.4 | Register List..... | 728 |
| 9.5 | Register Description..... | 729 |
| 9.5.1 | 0x0000 EMAC Basic Control Register0 (Default Value: 0x0000_0000) | 729 |
| 9.5.2 | 0x0004 EMAC Basic Control Register1 (Default Value: 0x0800_0000) | 729 |
| 9.5.3 | 0x0008 EMAC Interrupt Status Register (Default Value: 0x4000_0000) | 730 |
| 9.5.4 | 0x000C EMAC Interrupt Enable Register (Default Value: 0x0000_0000) | 732 |
| 9.5.5 | 0x0010 EMAC Transmit Control Register0 (Default Value: 0x0000_0000) | 733 |
| 9.5.6 | 0x0014 EMAC Transmit Control Register1 (Default Value: 0x0000_0000) | 734 |
| 9.5.7 | 0x001C EMAC Transmit Flow Control Register (Default Value: 0x0000_0000) | 734 |
| 9.5.8 | 0x0020 EMAC Transmit DMA Descriptor List Address Register (Default Value: 0x0000_0000)..... | 735 |
| 9.5.9 | 0x0024 EMAC Receive Control Register0 (Default Value: 0x0000_0000)..... | 735 |
| 9.5.10 | 0x0028 EMAC Receive Control Register1 (Default Value: 0x0000_0000)..... | 736 |
| 9.5.11 | 0x0034 EMAC Receive DMA Descriptor List Address Register (Default Value: 0x0000_0000) | 738 |
| 9.5.12 | 0x0038 EMAC Receive Frame Filter Register (Default Value: 0x0000_0000) | 738 |
| 9.5.13 | 0x0040 EMAC Receive Hash Table Register0 (Default Value: 0x0000_0000) | 739 |
| 9.5.14 | 0x0044 EMAC Receive Hash Table Register1 (Default Value: 0x0000_0000) | 739 |
| 9.5.15 | 0x0048 EMAC MII Command Register (Default Value: 0x0000_0000) | 740 |
| 9.5.16 | 0x004C EMAC MII Data Register (Default Value: 0x0000_0000) | 740 |
| 9.5.17 | 0x0050 EMAC MAC Address High Register0 (Default Value: 0x0000_FFFF) | 740 |

| | |
|--|-----|
| 9.5.18 0x0054 EMAC MAC Address Low Register0 (Default Value: 0xFFFF_FFFF) | 741 |
| 9.5.19 0x0050+0x8*N (N=1~7) EMAC MAC Address High RegisterN (Default Value: 0x0000_FFFF) | 741 |
| 9.5.20 0x0054+0x8*N (N=1~7) EMAC MAC Address Low Register N (Default Value: 0x0000_0000) | 741 |
| 9.5.21 0x00B0 EMAC Transmit DMA Status Register (Default Value: 0x0000_0000)..... | 741 |
| 9.5.22 0x00B4 EMAC Transmit DMA Current Descriptor Register (Default Value: 0x0000_0000)..... | 742 |
| 9.5.23 0x00B8 EMAC Transmit DMA Current Buffer Address Register (Default Value: 0x0000_0000) | 742 |
| 9.5.24 0x00C0 EMAC Receive DMA Status Register (Default Value: 0x0000_0000) | 742 |
| 9.5.25 0x00C4 EMAC Receive DMA Current Descriptor Register (Default Value: 0x0000_0000) | 743 |
| 9.5.26 0x00C8 EMAC Receive DMA Current Buffer Address Register (Default Value: 0x0000_0000) | 743 |
| 9.5.27 0x00D0 EMAC RGMII Status Register (Default Value: 0x0000_0000)..... | 743 |



Figures

| | |
|--|-----|
| Figure 9-1 EMAC Block Diagram | 723 |
| Figure 9-2 EMAC Typical Application | 724 |
| Figure 9-3 EMAC RX/TX Descriptor List..... | 725 |



Tables

| | |
|---|-----|
| Table 9-1 EMAC External Signals..... | 723 |
| Table 9-2 EMAC Clock Characteristics..... | 724 |



9 EMAC

9.1 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 both full and half duplex mode. The Internal DMA is designed for packet-oriented data transfers based on a linked list of descriptors. 2 KBytes TXFIFO and 8 KBytes RXFIFO are provided to keep continuous transmission and reception. Flow Control, CRC Pad & Stripping, and address filtering are also supported in this module.

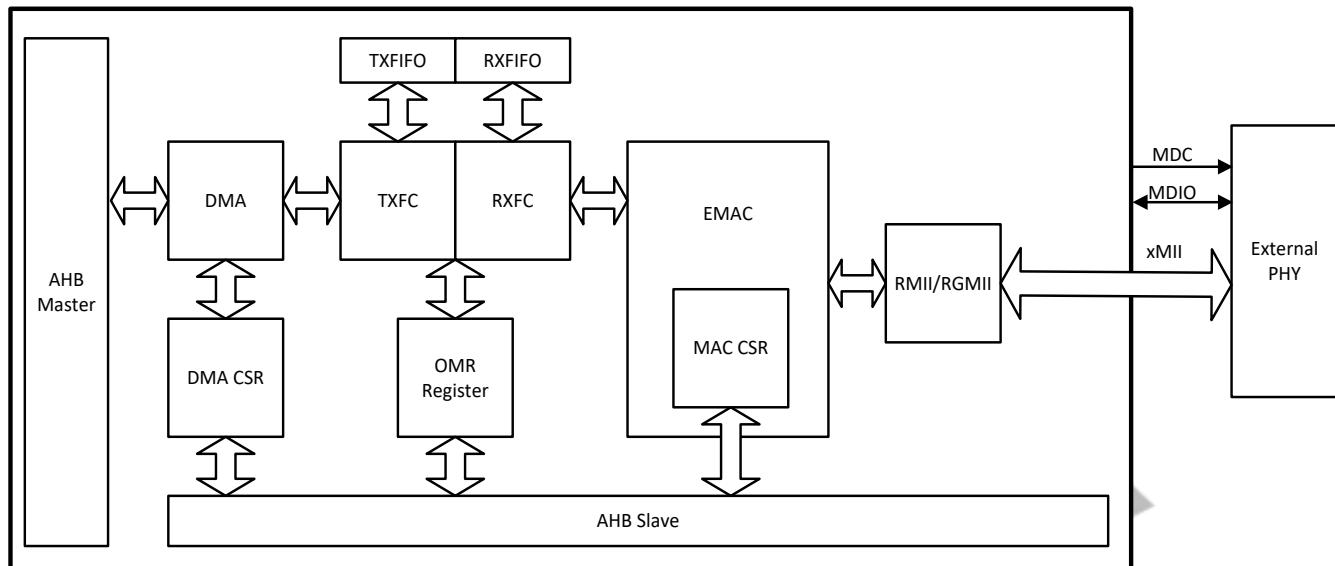
The EMAC has the following features:

- Supports 10/100/1000 Mbit/s data transfer rates
- Supports RMII/RGMII PHY interface
- Supports both full-duplex and half-duplex operation
- Programmable frame length to support Standard or Jumbo Ethernet frames with sizes up to 16 KBytes
- 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 KBytes of data
- Supports 2KB TXFIFO for transmission packets and 8KB RXFIFO for reception packets
- Programmable interrupt options for different operational conditions
- Support MDIO Interface

9.2 Block Diagram

The EMAC block diagram is shown below.

Figure 9-1 EMAC Block Diagram



9.3 Functional Description

9.3.1 External Signals

The following table describes the external signals of EMAC.

Table 9-1 EMAC External Signals

| Signal | Description | Type |
|-------------------------|--|------|
| RGMII-RXD0/RMII-RXD0 | RGMII/RMII Receive Data0 | I |
| RGMII-RXD1/ RMII-RXD1 | RGMII/RMII Receive Data1 | I |
| RGMII-RXD2 | RGMII Receive Data2 | I |
| RGMII-RXD3 | RGMII Receive Data3 | I |
| RGMII-RXCK | RGMII Receive Clock | I |
| RGMII-RXCTL/RMII-CRS-DV | RGMII Receive Control/RMII Carrier Sense Receive Data Valid | I |
| RGMII-TXD0/RMII-TXD0 | RGMII/RMII Transmit Data0 | O |
| RGMII-TXD1/RMII-TXD1 | RGMII/RMII Transmit Data1 | O |
| RGMII-TXD2 | RGMII Transmit Data2 | O |
| RGMII-TXD3 | RGMII Transmit Data3 | O |
| RGMII-TXCK/RMII-TXCK | RGMII/RMII Transmit Clock For RGMII, IO type is output; For RMII, IO type is input | I/O |

| Signal | Description | Type |
|-----------------------|---|------|
| RGMII-TXCTL/RMII-TXEN | RGMII Transmit Control/RMII Transmit Enable | O |
| RGMII-CLKIN/RMII-RXER | RGMII Transmit Clock from External | I |
| MDC | RGMII/RMII Management Data Clock | O |
| MDIO | RGMII/RMII Management Data Input/Output | I/O |
| EPHY-25M | 25MHz Output for EPHY | O |

9.3.2 Clock Sources

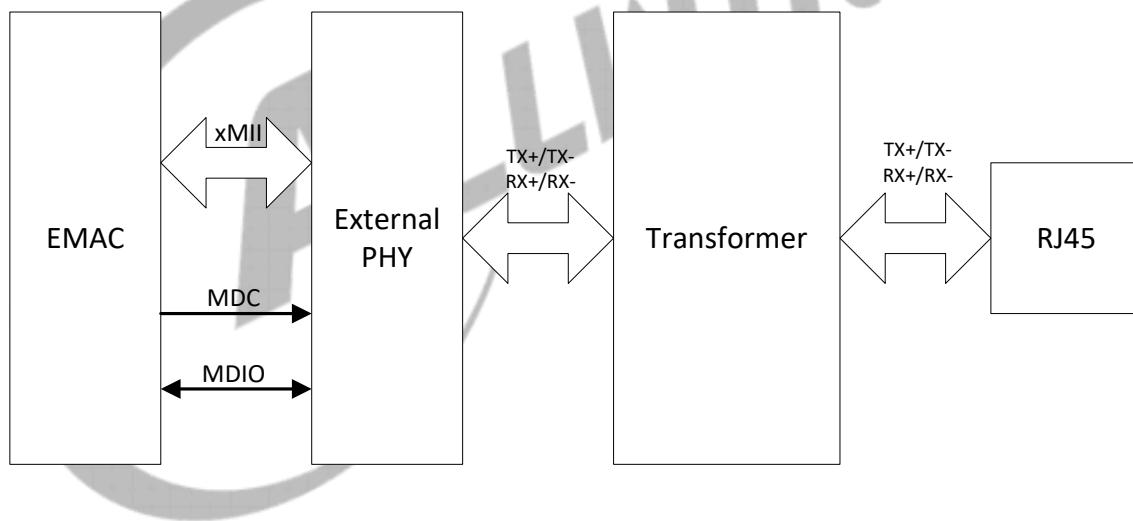
The following table describes the clock of EMAC.

Table 9-2 EMAC Clock Characteristics

| Module Clock | Source | Description |
|----------------|--------|---|
| EMAC_25M Clock | CCU | EMAC 25MHz clock source, refer to section 3.4 Clock Controller Unit (CCU) for detailed information. |

9.3.3 Typical Application

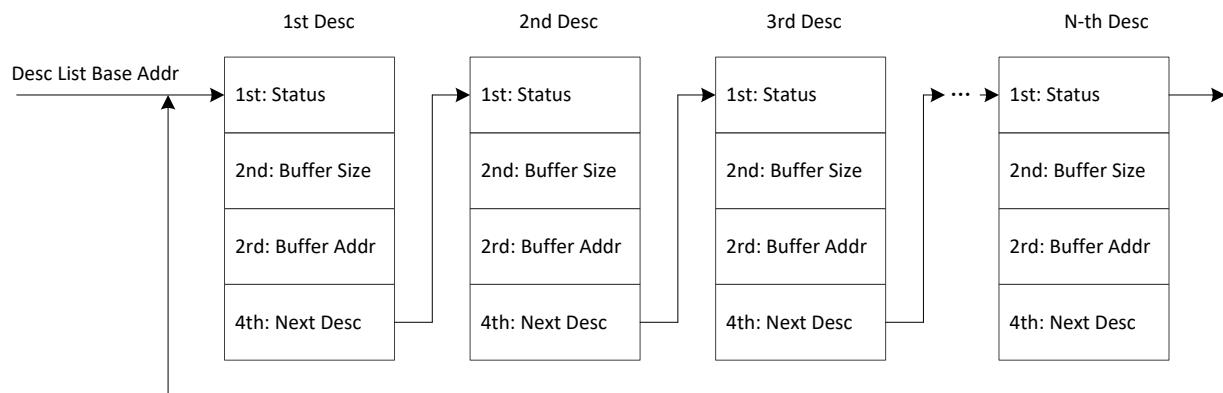
Figure 9-2 EMAC Typical Application



9.3.4 EMAC RX/TX Descriptor

The internal DMA of EMAC transfers data between host memory and internal RX/TX FIFO with a linked list of descriptors. Each descriptor is consisted of four words, and contains some necessary information to transfer TX and RX frames. The following figure shows descriptor list structure. The address of each descriptor must be 32-bit aligned.

Figure 9-3 EMAC RX/TX Descriptor List



9.3.5 Transmit Descriptor

9.3.5.1 1st Word of Transmit Descriptor

| Bits | Description |
|-------|--|
| 31 | TX_DESC_CTL When set, current descriptor can be used by DMA. This bit is cleared by DMA when the whole frame is transmitted or all data in current descriptor's buffer are transmitted. |
| 30:17 | Reserved |
| 16 | TX_HEADER_ERR When set, the checksum of transmitted frame's header is wrong. |
| 15 | Reserved |
| 14 | TX_LENGTH_ERR When set, the length of transmitted frame is wrong. |
| 13 | Reserved |
| 12 | TX_PAYLOAD_ERR When set, the checksum of transmitted frame's payload is wrong. |
| 11 | Reserved |
| 10 | TX_CRS_ERR When set, carrier is lost during transmission. |
| 9 | TX_COL_ERR_0 When set, the frame is aborted because of collision after contention period. |
| 8 | TX_COL_ERR_1 When set, the frame is aborted because of too many collisions. |
| 7 | Reserved. |
| 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 TX FIFO underflow error. |

| Bits | Description |
|------|---|
| 0 | TX_DEFER When set in Half-Duplex mode, the EMAC defers the frame transmission. |

9.3.5.2 2nd Word of Transmit Descriptor

| Bits | Description |
|-------|--|
| 31 | TX_INT_CTL When set and the current frame have been transmitted, the TX_INT in Interrupt Status Register will be set. |
| 30 | LAST_DESC When set, current descriptor is the last one for current frame. |
| 29 | FIR_DESC When set, current descriptor is the first one for current frame. |
| 28:27 | CHECKSUM_CTL These bits control to insert checksums in transmit frame. |
| 26 | CRC_CTL When set, CRC field is not transmitted. |
| 25:11 | Reserved |
| 10:0 | BUF_SIZE The size of buffer specified by current descriptor. |

9.3.5.3 3rd Word of Transmit Descriptor

| Bits | Description |
|------|--|
| 31:0 | BUF_ADDR The address of buffer specified by current descriptor. |

9.3.5.4 4th Word of Transmit Descriptor

| Bits | Description |
|------|--|
| 31:0 | NEXT_DESC_ADDR The address of next descriptor. It must be 32-bit aligned. |

9.3.6 Receive Descriptor

9.3.6.1 1st Word of Receive Descriptor

| Bits | Description |
|------|--|
| 31 | RX_DESC_CTL When set, current descriptor can be used by DMA. This bit is cleared by DMA when complete frame is received or current descriptor's buffer is full. |

| Bits | Description |
|-------|--|
| 30 | RX_DAF_FAIL When set, current frame don't pass 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 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 receive frame. |
| 15 | Reserved |
| 14 | RX_NO_ENOUGH_BUF_ERR When set, current frame is clipped because of no enough buffer. |
| 13 | RX_SAF_FAIL When set, current fame don't pass SA filter. |
| 12 | Reserved. |
| 11 | RX_OVERFLOW_ERR When set, a buffer overflow error occurred and current frame is wrong. |
| 10 | Reserved |
| 9 | FIR_DESC When set, current descriptor is the first descriptor for current frame. |
| 8 | LAST_DESC When set, current descriptor is the last descriptor for current frame. |
| 7 | RX_HEADER_ERR When set, the checksum of frame's header is wrong. |
| 6 | RX_COL_ERR When set, there is a late collision during reception in half-duplex mode. |
| 5 | Reserved. |
| 4 | RX_LENGTH_ERR When set, the length of current frame is wrong. |
| 3 | RX_PHY_ERR When set, the receive error signal from PHY is asserted during reception. |
| 2 | Reserved. |
| 1 | RX_CRC_ERR When set, the CRC filed of received frame is wrong. |
| 0 | RX_PAYLOAD_ERR When set, the checksum or length of received frame's payload is wrong. |

9.3.6.2 2nd Word of Receive Descriptor

| Bits | Description |
|-------|--|
| 31 | RX_INT_CTL When set and a frame have been received, the RX_INT will not be set. |
| 30:11 | Reserved |
| 10:0 | BUF_SIZE The size of buffer specified by current descriptor. |

9.3.6.3 3rd Word of Receive Descriptor

| Bits | Description |
|------|--|
| 31:0 | BUF_ADDR The address of buffer specified by current descriptor. |

9.3.6.4 4th Word of Receive Descriptor

| Bits | Description |
|------|--|
| 31:0 | NEXT_DESC_ADDR The address of next descriptor. This field must be 32-bit aligned. |

9.4 Register List

| Module Name | Base Address | Comments |
|-------------|--------------|-----------------|
| EMAC | 0x04500000 | EMAC Controller |

| Register Name | Offset | Description |
|-----------------------|--------------------|--|
| EMAC | | |
| EMAC_BASIC_CTL0 | 0x000 | EMAC Basic Control Register0 |
| EMAC_BASIC_CTL1 | 0x004 | EMAC Basic Control Register1 |
| EMAC_INT_STA | 0x008 | EMAC Interrupt Status Register |
| EMAC_INT_EN | 0x00C | EMAC Interrupt Enable Register |
| EMAC_TX_CTL0 | 0x010 | EMAC Transmit Control Register0 |
| EMAC_TX_CTL1 | 0x014 | EMAC Transmit Control Register1 |
| EMAC_TX_FLOW_CTL | 0x01C | EMAC Transmit Flow Control Register |
| EMAC_TX_DMA_DESC_LIST | 0x020 | EMAC Transmit DMA Descriptor List Address Register |
| EMAC_RX_CTL0 | 0x024 | EMAC Receive Control Register0 |
| EMAC_RX_CTL1 | 0x028 | EMAC Receive Control Register1 |
| EMAC_RX_DMA_DESC_LIST | 0x034 | EMAC Receive DMA Descriptor List Address Register |
| EMAC_RX_FRMFLT | 0x038 | EMAC Receive Frame Filter Register |
| EMAC_RX_HASH0 | 0x040 | EMAC Receive Hash Table Register0 |
| EMAC_RX_HASH1 | 0x044 | EMAC Receive Hash Table Register1 |
| EMAC_MII_CMD | 0x048 | EMAC MII Command Register |
| EMAC_MII_DATA | 0x04C | EMAC MII Data Register |
| EMAC_ADDR_HIGH0 | 0x050 | EMAC MAC Address High Register0 |
| EMAC_ADDR_LOW0 | 0x054 | EMAC MAC Address Low Register0 |
| EMAC_ADDR_HIGN | 0x050+0x8*N(N=1~7) | EMAC MAC Address High Register N |
| EMAC_ADDR_LOWN | 0x054+0x8*N(N=1~7) | EMAC MAC Address Low Register N |

| Register Name | Offset | Description |
|----------------------|--------|---|
| EMAC_TX_DMA_STA | 0x0B0 | EMAC Transmit DMA Status Register |
| EMAC_TX_DMA_CUR_DESC | 0x0B4 | EMAC Transmit DMA Current Descriptor Register |
| EMAC_TX_DMA_CUR_BUF | 0x0B8 | EMAC Transmit DMA Current Buffer Address Register |
| EMAC_RX_DMA_STA | 0x0C0 | EMAC Receive DMA Status Register |
| EMAC_RX_DMA_CUR_DESC | 0x0C4 | EMAC Receive DMA Current Descriptor Register |
| EMAC_RX_DMA_CUR_BUF | 0x0C8 | EMAC Receive DMA Current Buffer Address Register |
| EMAC_RGMII_STA | 0x0D0 | EMAC RGMII Status Register |

9.5 Register Description

9.5.1 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 00: 1000 Mbit/s 01: Reserved 10: 10 Mbit/s 11: 100 Mbit/s |
| 1 | R/W | 0x0 | LOOPBACK 0: Disable 1: Enable |
| 0 | R/W | 0x0 | DUPLEX 0: Half-duplex 1: Full-duplex |

9.5.2 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 | / | / | / |

| Offset: 0x0004 | | | Register Name: EMAC_BASIC_CTL1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 1 | R/W | 0x0 | <p>RX_TX_PRI RX TX DMA priority 0: Same priority 1: RX priority over TX</p> |
| 0 | R/W | 0x0 | <p>SOFT_RST Soft Reset all Registers and Logic 0: No valid 1: Reset</p> <p>Note: All clock inputs must be valid before soft rest. This bit is cleared internally when the reset operation is completed fully. Before write any register, this bit should read a 0.</p> |

9.5.3 0x0008 EMAC Interrupt Status Register (Default Value: 0x4000_0000)

| Offset: 0x0008 | | | Register Name: EMAC_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:17 | / | / | / |
| 16 | R/W1C | 0x0 | <p>RGMII_LINK_STA_P RGMII Link Status Changed Interrupt Pending 0: No Pending 1: Pending Write '1' to clear</p> |
| 15:14 | / | / | / |
| 13 | R/W1C | 0x0 | <p>RX_EARLY_P RX DMA Filled First data Buffer of the Receive Frame Interrupt Pending 0: No Pending 1: Pending Write '1' to clear</p> |
| 12 | R/W1C | 0x0 | <p>RX_OVERFLOW_P RX FIFO Overflow Error Interrupt Pending 0: No Pending 1: Pending Write '1' to clear</p> |
| 11 | R/W1C | 0x0 | <p>RX_TIMEOUT_P RX Timeout Interrupt Pending 0: No Pending 1: Pending Write '1' to clear .When this bit asserts, the length of receive frame is greater than 2048 bytes(10240 when JUMBO_FRM_EN is set)</p> |

| Offset: 0x0008 | | | Register Name: EMAC_INT_STA |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 10 | R/W1C | 0x0 | RX_DMA_STOPPED_P When this bit asserts, the RX DMA FSM is stopped. |
| 9 | R/W1C | 0x0 | RX_BUF_UA_P RX Buffer UA Interrupt Pending 0: No Pending 1: Pending Write '1' to clear .When this asserts, the RX DMA cannot acquire next RX descriptor and RX DMA FSM is suspended. The ownership of next RX descriptor should be changed to RX DMA. The RX DMA FSM will resume when writing to DMA_RX_START bit or next receive frame is coming. |
| 8 | R/W1C | 0x0 | RX_P Frame RX Completed Interrupt Pending 0: No Pending 1: Pending Write '1' to clear. When this bit asserts, a frame reception is completed. The RX DMA FSM remains in the running state. |
| 7:6 | / | / | / |
| 5 | R/W1C | 0x0 | TX_EARLY_P Frame Transmitted to FIFO totally Interrupt Pending 0: No Pending 1: Pending Write '1' to clear. |
| 4 | R/W1C | 0x0 | TX_UNDERFLOW_P TX FIFO Underflow Interrupt Pending 0: No Pending 1: Pending Write '1' to clear |
| 3 | R/W1C | 0x0 | TX_TIMEOUT_P Transmitter Timeout Interrupt Pending 0: No Pending 1: Pending Write '1' to clear |
| 2 | R/W1C | 0x0 | TX_BUF_UA_P TX Buffer UA Interrupt Pending 0: No Pending 1: Pending When this bit asserts, the TX DMA can not acquire next TX descriptor and TX DMA FSM is suspended. The ownership of next TX descriptor should be changed to TX DMA. The TX DMA FSM will resume when writing to DMA_TX_START bit. |

| Offset: 0x0008 | | | Register Name: EMAC_INT_STA |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 1 | R/W1C | 0x0 | <p>TX_DMA_STOPPED_P Transmission DMA Stopped Interrupt Pending 0: No Pending 1: Pending Write '1' to clear</p> |
| 0 | R/W1C | 0x0 | <p>TX_P Frame Transmission Interrupt Pending 0: No Pending 1: Pending Write '1' to clear</p> |

9.5.4 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 | <p>RX_EARLY_INT_EN Early Receive Interrupt 0: Disable 1: Enable</p> |
| 12 | R/W | 0x0 | <p>RX_OVERFLOW_INT_EN Receive Overflow Interrupt 0: Disable 1: Enable</p> |
| 11 | R/W | 0x0 | <p>RX_TIMEOUT_INT_EN Receive Timeout Interrupt 0: Disable 1: Enable</p> |
| 10 | R/W | 0x0 | <p>RX_DMA_STOPPED_INT_EN Receive DMA FSM Stopped Interrupt 0: Disable 1: Enable</p> |
| 9 | R/W | 0x0 | <p>RX_BUF_UA_INT_EN Receive Buffer Unavailable Interrupt 0: Disable 1: Enable</p> |
| 8 | R/W | 0x0 | <p>RX_INT_EN Receive Interrupt 0: Disable 1: Enable</p> |
| 7:6 | / | / | / |

| Offset: 0x000C | | | Register Name: EMAC_INT_EN |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 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 |
| 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 |

9.5.5 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 disable, transmit will continue until current transmit finish. |
| 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 | / | / | / |

9.5.6 0x0014 EMAC Transmit Control Register1 (Default Value: 0x0000_0000)

| Offset: 0x0014 | | | Register Name: EMAC_TX_CTL1 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31 | WAC | 0x0 | <p>TX_DMA_START Transmit DMA FSM Start 0: No valid 1: Start It is cleared internally and always read a 0</p> |
| 30 | R/W | 0x0 | <p>TX_DMA_EN 0: Stop TX DMA after the completion of current frame transmission. 1: Start and run TX DMA.</p> |
| 29:11 | / | / | / |
| 10:8 | R/W | 0x0 | <p>TX_TH The threshold value of TX DMA FIFO. When TX_MD is 0, transmission starts when the size of frame in TX DMA FIFO is greater than the threshold. In addition, full frames with a length less than the threshold are transferred automatically. 000: 64 001: 128 010: 192 011: 256 Others: Reserved</p> |
| 7:2 | / | / | / |
| 1 | R/W | 0x0 | <p>TX_MD Transmission Mode 0: TX start after TX DMA FIFO bytes is greater than TX_TH 1: TX start after TX DMA FIFO located a full frame</p> |
| 0 | R/WAC | 0x0 | <p>FLUSH_TX_FIFO This bit is set to flush the data in the TX FIFO, and cleared internally.</p> |

9.5.7 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 | <p>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 completion of transmission, this bit will be cleared automatically. Before write register TX_FLOW_CTRL, this bit must be read as 0.</p> |
| 30:22 | / | / | / |

| Offset: 0x001C | | | Register Name: EMAC_TX_FLOW_CTL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 21:20 | R/W | 0x0 | <p>TX_PAUSE_FRM_SLOT</p> <p>The threshold of the pause timer at which the input flow control signal is checked for automatic retransmission of pause frame.</p> <p>The threshold values should be always less than the PAUSE_TIME</p> |
| 19:4 | R/W | 0x0 | <p>PAUSE_TIME</p> <p>The pause time field in the transmitted control frame.</p> |
| 3:2 | / | / | / |
| 1 | R/W | 0x0 | <p>ZQP_FRM_EN</p> <p>0: Disable</p> <p>1: Enable</p> <p>When set, enable the functionality to generate Zero-Quanta Pause control frame.</p> |
| 0 | R/W | 0x0 | <p>TX_FLOW_CTL_EN</p> <p>TX Flow Control Enable</p> <p>0: Disable</p> <p>1: Enable</p> <p>When set, enable flow control operation to transmit pause frames in full-duplex mode, or enable the back-pressure operation in half-duplex mode.</p> |

9.5.8 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 | <p>TX_DESC_LIST</p> <p>The base address of transmit descriptor list. It must be 32-bit aligned.</p> |

9.5.9 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 | <p>RX_EN</p> <p>Enable Receiver</p> <p>0: Disable receiver after current reception</p> <p>1: Enable</p> |
| 30 | R/W | 0x0 | <p>RX_FRM_LEN_CTL</p> <p>Frame Receive Length Control</p> <p>0: Up to 2,048 bytes (JUMBO_FRM_EN==0)</p> <p>Up to 10,240 bytes (JUMBO_FRM_EN==1)</p> <p>1: Up to 16,384 bytes</p> <p>Any bytes after that is cut off</p> |

| Offset: 0x0024 | | | Register Name: EMAC_RX_CTL0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 29 | R/W | 0x0 | JUMBO_FRM_EN Jumbo Frame Enable 0: Disable 1: Enable Jumbo frames of 9,018 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's field value is less than or equal to 1,500 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 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 decode the received pause frame and disable its transmitter for a specified time by pause frame. |
| 15:0 | / | / | / |

9.5.10 0x0028 EMAC Receive Control Register1 (Default Value: 0x0000_0000)

| Offset: 0x0028 | | | Register Name: EMAC_RX_CTL1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31 | WAC | 0x0 | RX_DMA_START When set, the RX DMA will go to 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 finish receiving 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 |

| Offset: 0x0028 | | | Register Name: EMAC_RX_CTL1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 23:22 | R/W | 0x0 | <p>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.</p> |
| 21:20 | R/W | 0x0 | <p>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.</p> |
| 19:6 | / | / | / |
| 5:4 | R/W | 0x0 | <p>RX_TH Threshold for RX DMA FIFO Start 00: 64 01: 32 10: 96 11: 128 Only valid when RX_MD == 0, 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 set, forward undersized frames with no error and length less than 64bytes</p> |
| 1 | R/W | 0x0 | <p>RX_MD Receive Mode 0: RX start read after RX DMA FIFO bytes is greater than RX_TH 1: RX start read after RX DMA FIFO located a full frame</p> |
| 0 | R/W | 0x0 | <p>FLUSH_RX_FRM Flush Receive Frames 0: Enable when receive descriptors/buffers is unavailable 1: Disable</p> |

9.5.11 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 | RX_DESC_LIST The base address of receive descriptor list. It must be 32-bit aligned. |

9.5.12 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 pass 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 SA filter. In addition, if the SA field of 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 not matches 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 pass the address register filter or the hash filter(set by HASH_MULTICAST or HASH_UNICAST) |
| 0 | R/W | 0x0 | RX_ALL Receive All Frame Enable 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 |

9.5.13 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 receive frame filter. |

9.5.14 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 receive frame filter. |

9.5.15 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 Divide Ratio 000: 16 001: 32 010: 64 011: 128 Others: Reserved MDC Clock is divided from AHB clock |
| 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 0: Write no valid, read 0 indicate finish in read or write operation 1: Write start read or write operation and cleared internally, read 1 indicate busy. |

9.5.16 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. |

9.5.17 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. |

9.5.18 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. |

9.5.19 0x0050+0x8*N (N=1~7) EMAC MAC Address High RegisterN (Default Value: 0x0000_FFFF)

| Offset: 0x0050+0x8*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 |
| 29:24 | R/W | 0x0 | MAC_ADDR_BYTE_CTL MAC Address Byte Control Mask The lower bit of mask controls the lower byte of in 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 16bits of the MAC address. |

9.5.20 0x0054+0x8*N (N=1~7) EMAC MAC Address Low Register N (Default Value: 0x0000_0000)

| Offset: 0x0054+0x8*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 32bits of MAC address N (N: 1~7). |

9.5.21 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 | / | / | / |

| Offset: 0x00B0 | | | Register Name: EMAC_TX_DMA_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 2:0 | R | 0x0 | <p>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 frame from host memory to TX DMA FIFO 100: Reserved 101: Reserved 111: RUN_CLOSE_DESC, Closing TX descriptor 110: SUSPEND, TX descriptor unavailable or TX DMA FIFO underflow</p> |

9.5.22 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 | The address of current transmit descriptor. |

9.5.23 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 | The address of current transmit DMA buffer. |

9.5.24 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 | <p>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 frame. 100: SUSPEND, RX descriptor unavailable; 101: RUN_CLOSE_DESC, Closing RX descriptor. 110: Reserved 111: RUN_TRANS_DATA, Passing frame from host memory to RX DMA FIFO;</p> |

9.5.25 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 | The address of current receive descriptor |

9.5.26 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 | The address of current receive DMA buffer |

9.5.27 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 RGMII interface 0: down 1: up |
| 2:1 | R | 0x0 | RGMII_LINK_SPD The link speed of RGMII interface 00: 2.5 MHz 01: 25 MHz 10: 125 MHz 11: Reserved |
| 0 | R | 0x0 | RGMII_LINK_MD The link mode of RGMII interface 0: Half-Duplex 1: Full-Duplex |

Contents

| | |
|---|-----|
| 10 Interfaces | 750 |
| 10.1 The Two Wire Interface (TWI)..... | 750 |
| 10.1.1 Overview | 750 |
| 10.1.2 Block Diagram | 751 |
| 10.1.3 Functional Description | 751 |
| 10.1.4 Programming Guidelines..... | 757 |
| 10.1.5 Register List..... | 760 |
| 10.1.6 Register Description..... | 761 |
| 10.2 UART..... | 774 |
| 10.2.1 Overview | 774 |
| 10.2.2 Block Diagram | 775 |
| 10.2.3 Functional Description | 775 |
| 10.2.4 Programming Guidelines..... | 781 |
| 10.2.5 Register List..... | 784 |
| 10.2.6 Register Description | 784 |
| 10.3 Serial Peripheral Interface (SPI) | 805 |
| 10.3.1 Overview | 805 |
| 10.3.2 Block Diagram | 806 |
| 10.3.3 Functional Description | 807 |
| 10.3.4 Programming Guidelines..... | 813 |
| 10.3.5 Register List..... | 816 |
| 10.3.6 Register Description | 817 |
| 10.4 USB2.0 DRD | 835 |
| 10.4.1 Overview | 835 |
| 10.4.2 Block Diagram | 836 |
| 10.4.3 Functional Description | 836 |
| 10.4.4 USB_DRD_DEVICE Register List..... | 837 |
| 10.4.5 USB_DRD_DEVICE Register Description..... | 838 |
| 10.4.6 USB_DRD_HOST Register List..... | 862 |

| | |
|---|------|
| 10.4.7 USB_DRD_HOST Register Description..... | 863 |
| 10.5 GPIO | 908 |
| 10.5.1 Overview | 908 |
| 10.5.2 Block Diagram | 908 |
| 10.5.3 Functional Description | 909 |
| 10.5.4 Register List..... | 914 |
| 10.5.5 Register Description | 917 |
| 10.6 GPADC..... | 1039 |
| 10.6.1 Overview | 1039 |
| 10.6.2 Block Diagram | 1039 |
| 10.6.3 Functional Descriptions..... | 1040 |
| 10.6.4 Programming Guidelines..... | 1041 |
| 10.6.5 Register List..... | 1042 |
| 10.6.6 Register Description | 1043 |
| 10.7 PWM | 1059 |
| 10.7.1 Overview | 1059 |
| 10.7.2 Block Diagram | 1059 |
| 10.7.3 Functional Description | 1060 |
| 10.7.4 Programming Guidelines..... | 1070 |
| 10.7.5 Register List..... | 1071 |
| 10.7.6 Register Description | 1072 |
| 10.8 Wiegand Protocol..... | 1103 |
| 10.8.1 Overview | 1103 |
| 10.8.2 Block Diagram | 1103 |
| 10.8.3 Functional Description | 1104 |
| 10.8.4 Programming Guidelines..... | 1108 |
| 10.8.5 Register List..... | 1108 |
| 10.8.6 Register Description | 1109 |

Figures

| | |
|--|-----|
| Figure 10-1 TWI Block Diagram..... | 751 |
| Figure 10-2 Write Timing in 7-bit Standard Addressing Mode | 752 |
| Figure 10-3 Read Timing in 7-bit Standard Addressing Mode | 753 |
| Figure 10-4 Write Timing in 10-bit Extended Addressing Mode..... | 753 |
| Figure 10-5 Read Timing in 10-bit Extended Addressing Mode..... | 753 |
| Figure 10-6 TWI Driver Write Packet Transmission | 754 |
| Figure 10-7 TWI Driver Read Packet Transmission | 755 |
| Figure 10-8 TWI Programming State Diagram | 757 |
| Figure 10-9 UART Block Diagram | 775 |
| Figure 10-10 UART Serial Data Format | 776 |
| Figure 10-11 Application Diagram for RTS/CTS Autoflow Control..... | 777 |
| Figure 10-12 RTS/CTS Autoflow Control Data Format..... | 777 |
| Figure 10-13 Application Diagram for IrDA Transceiver | 777 |
| Figure 10-14 Serial IrDA Data Format | 778 |
| Figure 10-15 Application Diagram for RS-485 Transceiver | 778 |
| Figure 10-16 RS-485 Data Format..... | 778 |
| Figure 10-17 SPI Block Diagram | 806 |
| Figure 10-18 SPI Application Block Diagram | 808 |
| Figure 10-19 SPI Phase 0 Timing Diagram..... | 809 |
| Figure 10-20 SPI Phase 1 Timing Diagram..... | 810 |
| Figure 10-21 SPI 3-Wire Mode | 810 |
| Figure 10-22 SPI Dual-Input/Dual-Output Mode | 811 |
| Figure 10-23 SPI Dual I/O Mode..... | 811 |
| Figure 10-24 SPI Quad-Input/Quad-Output Mode | 812 |
| Figure 10-25 SPI Write/Read Data in CPU Mode | 814 |
| Figure 10-26 SPI Write/Read Data in DMA Mode | 815 |
| Figure 10-27 USB2.0 DRD Controller Block Diagram | 836 |
| Figure 10-28USB2.0 DRD Controller and PHY Connection Diagram | 837 |
| Figure 10-29 GPIO Block Diagram | 908 |

| | |
|--|------|
| Figure 10-30 Pull up/down Logic | 912 |
| Figure 10-31 IO Buffer Strength Diagram | 913 |
| Figure 10-32 GPADC Block Diagram | 1039 |
| Figure 10-33 GPADC Clock and Timing Requirement | 1041 |
| Figure 10-34 GPADC Initial Process..... | 1041 |
| Figure 10-35 PWM Block Diagram | 1060 |
| Figure 10-36 PWM Typical Application..... | 1061 |
| Figure 10-37 PWM01 Clock Controller Diagram | 1061 |
| Figure 10-38 PWM01 Output Logic Module Diagram..... | 1062 |
| Figure 10-39 PWM0 High Level Active State..... | 1063 |
| Figure 10-40 PWM0 Low Level Active State | 1064 |
| Figure 10-41 Phase of PWM0 High Level Active State | 1064 |
| Figure 10-42 PWM0 Output Waveform in Pulse Mode and Cycle Mode | 1065 |
| Figure 10-43 PWM01 Complementary Pair Output..... | 1066 |
| Figure 10-44 Dead-time Output Waveform..... | 1067 |
| Figure 10-45 Group 0–3 PWM Signal Output | 1068 |
| Figure 10-46 PWM01 Capture Logic Module Diagram | 1068 |
| Figure 10-47 PWM0 Channel Capture Timing..... | 1069 |
| Figure 10-48 Block Diagram of WIEGAND_CTRL Module | 1103 |
| Figure 10-49 Application Diagram of WIEGAND_CTRL Module..... | 1104 |
| Figure 10-50 Data Signal Oscillogram | 1106 |
| Figure 10-51 Diagram of WCLK Clock Module | 1107 |

Tables

| | |
|---|------|
| Table 10-1 TWI Sub-blocks..... | 751 |
| Table 10-2 TWI External Signals | 751 |
| Table 10-3 TWI Clock Sources..... | 752 |
| Table 10-4 UART External Signals | 775 |
| Table 10-5 UART Clock Sources..... | 776 |
| Table 10-6 UART Mode Baud and Error Rates | 779 |
| Table 10-7 IrDA Mode Baud and Error Rates..... | 780 |
| Table 10-8 RS485 Mode Baud and Error Rates..... | 780 |
| Table 10-9 SPI Sub-blocks | 806 |
| Table 10-10 SPI External Signals..... | 807 |
| Table 10-11 SPI Clock Sources..... | 808 |
| Table 10-12 SPI Transmit Format..... | 809 |
| Table 10-13 USB2.0 DRD External Signals..... | 836 |
| Table 10-14 Multi-function Port | 909 |
| Table 10-15 PA Multiplex Function..... | 909 |
| Table 10-16 PC Multiplex Function | 910 |
| Table 10-17 PD Multiplex Function..... | 910 |
| Table 10-18 PE Multiplex Function | 910 |
| Table 10-19 PF Multiplex Function | 911 |
| Table 10-20 PG Multiplex Function | 911 |
| Table 10-21 PH Multiplex Function..... | 911 |
| Table 10-22 PI Multiplex Function | 911 |
| Table 10-23 Port Function..... | 911 |
| Table 10-24GPADC External Signals..... | 1040 |
| Table 10-25 GPADC Clock Sources | 1040 |
| Table 10-26 PWM External Signals | 1060 |
| Table 10-27 External Signals of WIEGAND_CTRL..... | 1104 |
| Table 10-28 Data Format of Wiegand (26-bit)..... | 1104 |
| Table 10-29 Data Format of Wiegand (34-bit)..... | 1105 |

Table 10-30 Range Setting of WIEGAND_TP Register and WIEGAND_TW Register under different frequency dividing coefficients.....1107



10 Interfaces

10.1 The Two Wire Interface (TWI)

10.1.1 Overview

The Two Wire Interface (TWI) provides an interface between a CPU and any TWI-bus-compatible device that connects via the TWI bus. The TWI is designed to be compatible with the standard I2C bus protocol. The communication of the TWI is carried out by a byte-wise mode based on interrupt polled handshaking. Each device on the TWI bus is recognized by a unique address and can operate as either transmitter or receiver, a device connected to the TWI bus can be considered as master or slave when performing data transfers. Note that a master device is a device that initiates a data transfer on the bus and generates the clock signals to permit the transfer. During this transfer, any device addressed by this master is considered a slave.

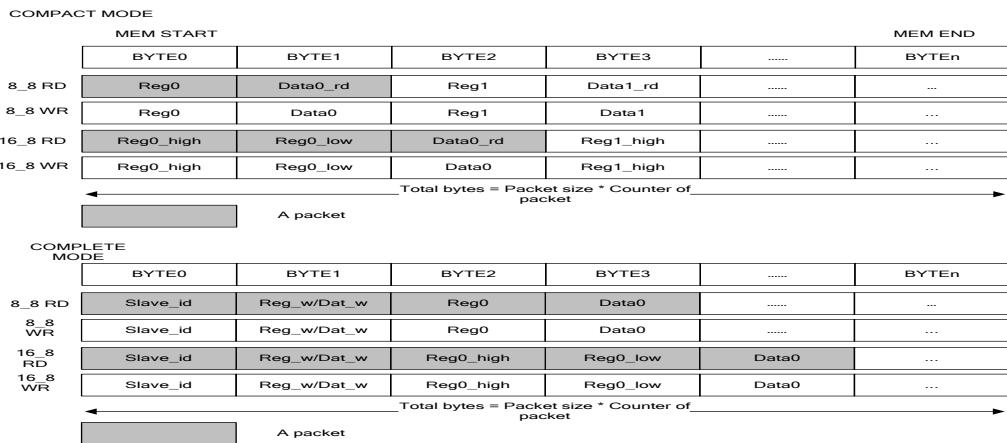
The TWI has the following features:

- Supports 5 TWIs
- Software programmability for slave or master
- Supports 7-bit and 10-bit device addressing modes
- Supports standard mode (up to 100 kbit/s) and fast mode (up to 400 kbit/s)
- Master mode features:
 - Supports the bus arbitration in the case of multiple master devices
 - Supports clock synchronization and bit and byte waiting
 - Supports packet transmission and DMA
- Slave mode features:
 - Interrupt on address detection

10.1.2 Block Diagram

The following figure shows the block diagram of TWI.

Figure 10-1 TWI Block Diagram



TWI contains the following sub-blocks:

Table 10-1 TWI Sub-blocks

| Term | Definition |
|---------|--------------------------------------|
| RESET | Module reset signal |
| INT | Module output interrupt signal |
| CFG_REG | Module configuration register in TWI |
| PE | Packet encoding/decoding |
| CCU | Module clock controller unit |

10.1.3 Functional Description

10.1.3.1 External Signals

The TWI controller has 5 TWI modules called TWI0, TWI1, TWI2, TWI3, and TWI4. The following table describes the external signals of the TWI. The TWIn-SCK and TWIn-SDA are bidirectional I/O, when the TWI is configured as a master device, the TWIn-SCK is an output pin; when the TWI is configurable as a slave device, the TWIn-SCK is an input pin. When using TWI, the corresponding PADs are selected as TWI function via section 10.5 GPIO.

Table 10-2 TWI External Signals

| Signal | Description | Type |
|----------|-------------------|------|
| TWI0 | | |
| TWI0_SCK | TWI0 Clock Signal | I/O |
| TWI0_SDA | TWI0 Serial Data | I/O |
| TWI1 | | |
| TWI1_SCK | TWI1 Clock Signal | I/O |
| TWI1_SDA | TWI1 Serial Data | I/O |

| Signal | Description | | Type |
|----------|-------------------|--|------|
| TWI2 | | | |
| TWI2_SCK | TWI2 Clock Signal | | I/O |
| TWI2_SDA | TWI2 Serial Data | | I/O |
| TWI3 | | | |
| TWI3_SCK | TWI3 Clock Signal | | I/O |
| TWI3_SDA | TWI3 Serial Data | | I/O |
| TWI4 | | | |
| TWI4_SCK | TWI4 Clock Signal | | I/O |
| TWI4_SDA | TWI4 Serial Data | | I/O |

10.1.3.2 Clock Sources

Each TWI controller has an input clock source. The following table describes the clock sources for TWI. After selecting a proper clock, users must open the gating of TWI and release the corresponding reset bit.

For more details on the clock setting, configuration, and gating information, see section 3.4 Clock Controller Unit (CCU).

Table 10-3 TWI Clock Sources

| Clock Sources | Description |
|---------------|---|
| APB1 Bus | TWI clock source. Refer to section 3.4 Clock Controller Unit (CCU) for details on APB1. |

10.1.3.3 Write/Read Timing in Standard and Extended Addressing Mode

This section is the 7-bit/10-bit addressing mode of the entire TWI protocol to read and write device registers. It can be achieved by directly using the TWI engine or using the TWI driver to control the TWI engine.

The following describes the write timing in 7-bit standard addressing mode.

Figure 10-2 Write Timing in 7-bit Standard Addressing Mode

Slave addr = 7-bit , register addr = 8-bit, data = 8-bit



Slave addr = 7-bit , register addr = 8-bit, data = n-byte



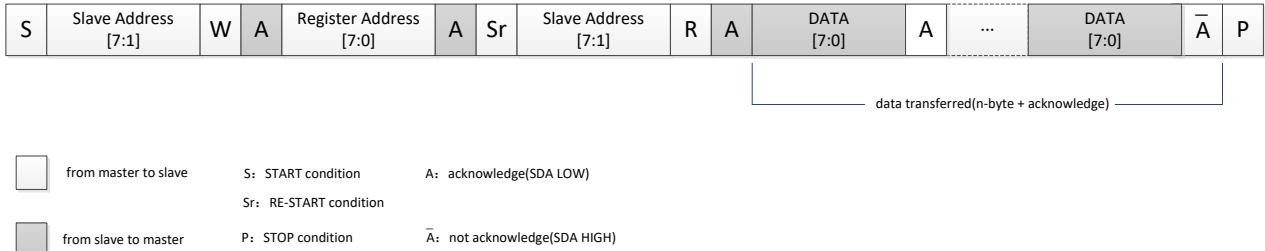
The following figure describes the read timing in 7-bit standard address mode.

Figure 10-3 Read Timing in 7-bit Standard Addressing Mode

Slave addr = 7-bit , register addr = 8-bit, data = 8-bit



Slave addr = 7-bit , register addr = 8-bit, data = n-byte



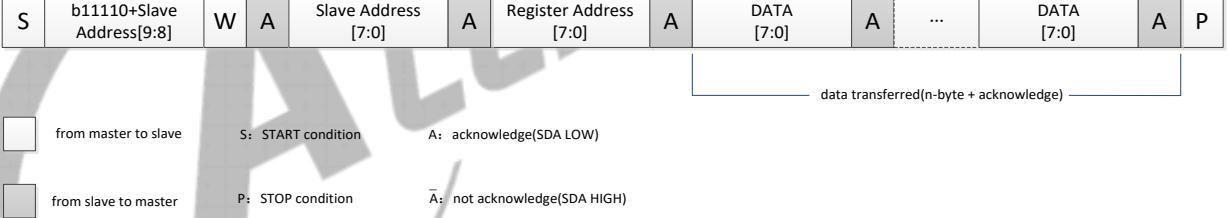
The following figure describes the write timing in 10-bit extended address mode.

Figure 10-4 Write Timing in 10-bit Extended Addressing Mode

Slave addr = 10-bit , register addr = 8-bit, data = 8-bit



Slave addr = 10-bit , register addr = 8-bit, data = n-byte



The following figure describes the read timing in 10-bit extended address mode.

Figure 10-5 Read Timing in 10-bit Extended Addressing Mode

Slave addr = 10-bit , register addr = 8-bit, data = 8-bit



Slave addr = 10-bit , register addr = 8-bit, data = n-byte



10.1.3.4 Write/Read Packet Transmission of TWI Driver

The TWI driver is only supported for master mode. When the TWI works in master mode, the TWI driver drives the TWI engine for one or more packet transmission instead of the CPU host. Packet transmission is defined in the following figures. The register address bytes and the written data bytes are buffered in SEND_FIFO, the read data bytes are buffered in RECV_FIFO.

Figure 10-6 TWI Driver Write Packet Transmission

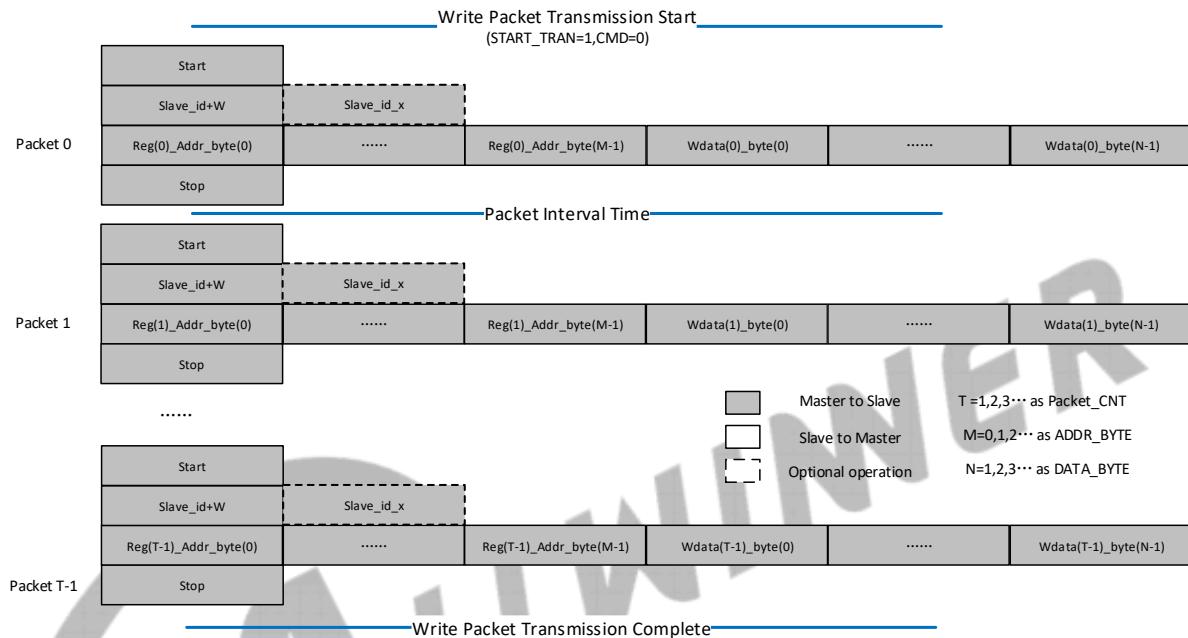
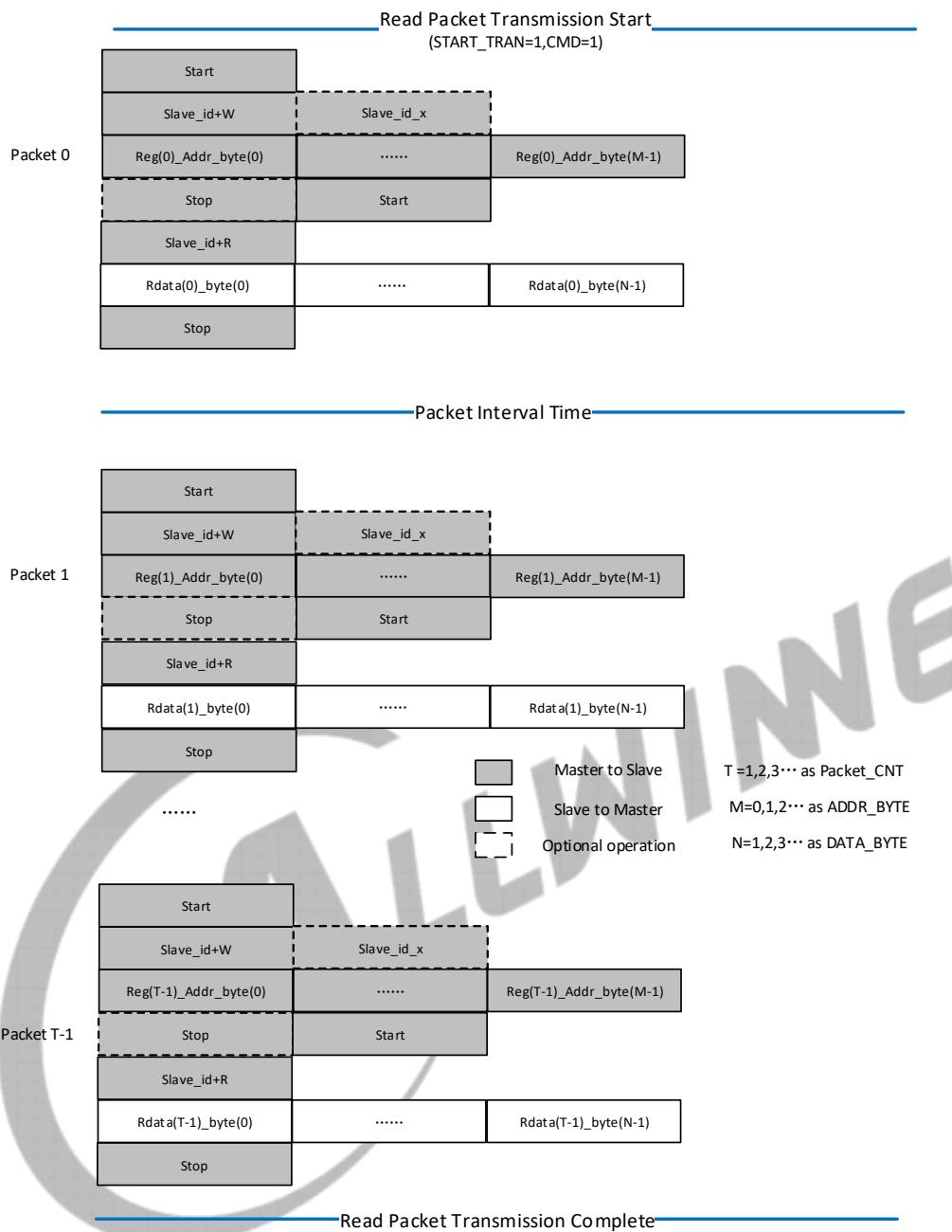


Figure 10-7 TWI Driver Read Packet Transmission

10.1.3.5 Master and Slave Mode of TWI Engine

There are four operation modes on the TWI bus which dictates the communications method. They are Master Transmit, Master Receive, Slave Transmit, and Slave Receive. In general, CPU host controls TWI engine by writing commands and data to its registers. TWI engine transmits an interrupt to CPU when each time a byte transfer is done or a START/STOP conditions is detected. The CPU host can also poll the status register for current status if the interrupt mechanism is not disabled by the CPU host.

When the CPU host wants to start a bus transfer, it initiates a bus START to enter the master mode by setting IM_STA bit in the [TWI_CNTR \(Offset: 0x000C\)](#) to high (before it must be low). The TWI Engine will assert INT line and INT_FLAG to indicate a completion for the START condition and each consequent byte transfer. At each

interrupt, the micro-processor needs to check the [TWI_STAT \(Offset: 0x0010\)](#) for current status. A transfer has to be concluded with STOP condition by setting M_STP bit high.

In Slave Mode, the TWI Engine also constantly samples the bus and look for its own slave address during addressing cycles. Once a match is found, it is addressed and interrupt the CPU host with the corresponding status. Upon request, the CPU host should read the status, read/write [TWI_DATA \(Offset: 0x0008\)](#) data register, and set the [TWI_CTR \(Offset: 0x000C\)](#). After each byte transfer, a slave device always halt the operation of remote master by holding the next low pulse on SCL line until the microprocessor responds to the status of previous transfer or START condition.

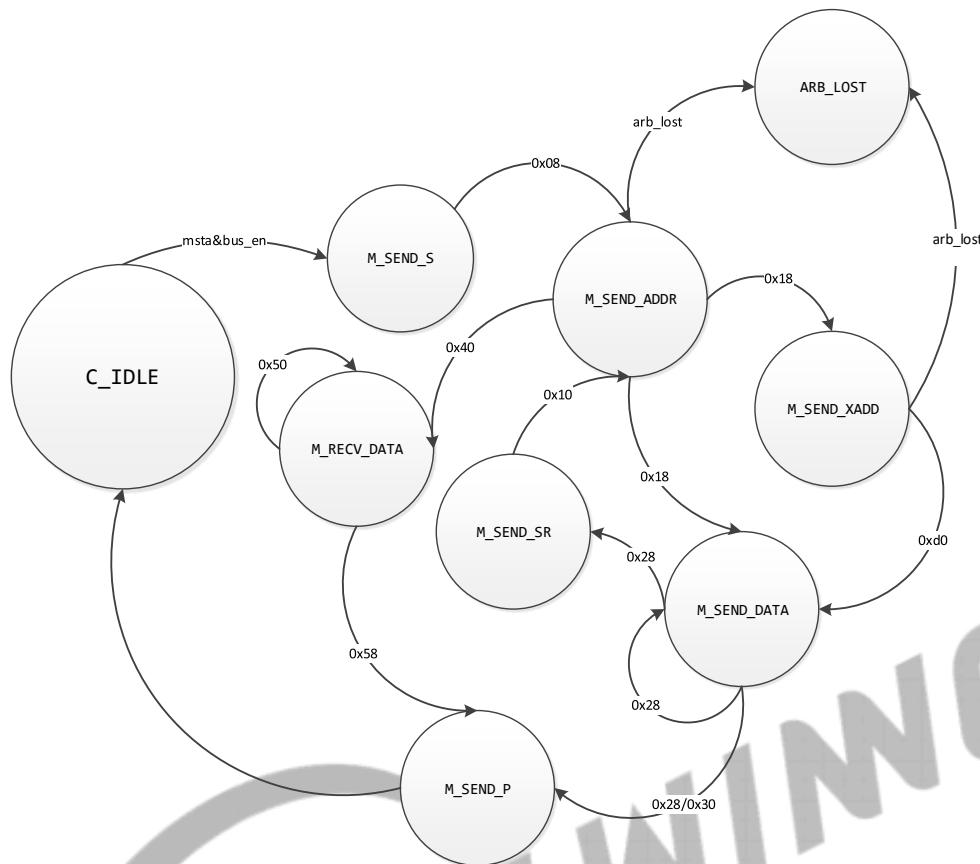
10.1.3.6 Generation of Repeated Start

After the data transfer, if the master still requires the bus, it can signal another Start followed by another slave address without signaling a Stop.

10.1.3.7 Programming State Diagram

The following figure shows the TWI programming state diagram. For the value between two states, see the [TWI_STAT \(Offset: 0x0010\)](#) register in section 10.1.6.5.

M_SEND_S: master sends START signal;
M_SEND_ADDR: master sends slave address;
M_SEND_XADD: master sends slave extended address;
M_SEND_SR: master repeated start;
M_SEND_DATA: master sends data;
M_SEND_P: master sends STOP signal;
M_RECV_DATA: master receives data;
ARB_LOST: Arbitration lost;
C_IDLE: Idle.

Figure 10-8 TWI Programming State Diagram

10.1.4 Programming Guidelines

The TWI controller operates in an 8-bit data format. The data on the TWI_SDA line is always 8 bits long. At first, the TWI controller sends a start condition. When in the addressing formats of 7-bit, the TWI sends out an 8-bit message which includes 7 MSB slave address and 1 LSB read/write flag. The least significant of the slave address indicates the direction of transmission. When the TWI works in 10-bit slave address mode, the operation will be divided into two steps, for details on the operation, refer to register description in Section 10.1.6.1 and 10.1.6.2.

The following takes the TWI module in the CPUX domain as an example.

10.1.4.1 Initialization for TWI Engine

To initialize the TWI engine, perform the following steps:

1. Configure corresponding GPIO multiplex function as TWI mode.
2. For TWIn, set [TWI_BGR_REG](#)[TWIn_GATING] in CCU module to 0 to close TWIn clock.
3. For TWIn, set [TWI_BGR_REG](#)[TWIn_RST] in CCU module to 0, then set to 1 to reset TWIn.
4. For TWIn, set [TWI_BGR_REG](#)[TWIn_GATING] in CCU module to 1 to open TWIn clock.
5. Configure [TWI_CCR](#)[CLK_M] and [TWI_CCR](#)[CLK_N] to get the needed rate (The clock source of TWI is from APB1).

6. Configure [TWI_CNTR\[BUS_EN\]](#) and [TWI_CNTR\[A_ACK\]](#), when using interrupt mode, set [TWI_CNTR\[INT_EN\]](#) to 1, and register the system interrupt through PLIC module. In slave mode, configure [TWI_ADDR](#) and [TWI_XADDR](#) registers to finish TWI initialization configuration.

10.1.4.2 Writing Data Operation for TWI Engine

To write data to the device, perform the following steps:

1. Clear [TWI_EFR](#) register, and configure [TWI_CNTR\[M_STA\]](#) to 1 to transmit the START signal.
2. After the START signal is transmitted, the first interrupt is triggered, then write device ID to [TWI_DATA](#) (For a 10-bit device ID, firstly write the first byte ID, secondly write the second byte ID in the next interrupt).
3. The Interrupt is triggered again after device ID transmission completes, write device data address to be read to [TWI_DATA](#) (For a 16-bit address, firstly write the first-byte address, secondly write the second-byte address).
4. Interrupt is triggered after data address transmission completes, write data to be transmitted to [TWI_DATA](#) (For consecutive write data operation, every byte transmission completion triggers interrupt, during interrupt write the next byte data to [TWI_DATA](#)).
5. After transmission completes, write [TWI_CNTR\[M_STP\]](#) to 1 to transmit the STOP signal and end this write-operation.

10.1.4.3 Reading Data Operation for TWI Engine

To read data from the device, perform the following steps:

1. Clear [TWI_EFR](#) register, and set [TWI_CNTR\[A_ACK\]](#) to 1, and configure [TWI_CNTR\[M_STA\]](#) to 1 to transmit the START signal.
2. After the START signal is transmitted, the first interrupt is triggered, then write device ID to [TWI_DATA](#) (For a 10-bit device ID, firstly write the first-byte ID, secondly write the second-byte ID in the next interrupt).
3. The Interrupt is triggered again after device ID transmission completes, write device data address to be read to [TWI_DATA](#) (For a 16-bit address, firstly write the first-byte address, secondly write the second-byte address).
4. The Interrupt is triggered after data address transmission completes, write [TWI_CNTR\[M_STA\]](#) to 1 to transmit new START signal, and after interrupt triggers, write device ID to [TWI_DATA](#) to start read-operation.
5. After device address transmission completes, each receive completion will trigger an interrupt, in turn, read [TWI_DATA](#) to get data, when receiving the previous interrupt of the last byte data, clear [A_ACK] to stop acknowledge signal of the last byte.
6. Write [TWI_CNTR\[M_STP\]](#) to 1 to transmit the STOP signal and end this read-operation.

10.1.4.4 Initialization for TWI Driver

To initialize the TWI driver, perform the following steps:

1. Configure corresponding GPIO multiplex function as TWI mode.
2. For TWIn, set [TWI_BGR_REG](#)[TWIn_GATING] in CCU module to 0 to close TWIn clock.
3. For TWIn, set [TWI_BGR_REG](#)[TWIn_RST] in CCU module to 0, then set to 1 to reset TWIn.
4. For TWIn, set [TWI_BGR_REG](#)[TWIn_GATING] in CCU module to 1 to open TWIn clock.
5. Set [TWI_DRV_CTRL](#)[TWI_DRV_EN] to 1 to enable the TWI driver.
6. Configure [TWI_DRV_BUS_CTRL](#)[CLK_M] and [TWI_DRV_BUS_CTRL](#)[CLK_N] to get the needed rate (The clock source of TWI is from APB1).
7. Set [TWI_DRV_CTRL](#)[RESTART_MODE] to 0 and [READ_TRAN_MODE] to 1, set [TWI_DRV_INT_CTRL](#)[TRAN_COM_INT_EN] to 1.
8. When using DMA for data transmission, set [TWI_DRV_DMA_CFG](#)[DMA_RX_EN] and [TWI_DRV_DMA_CFG](#)[DMA_TX_EN] to 1, and configure [TWI_DRV_DMA_CFG](#)[RX_TRIG] and [TWI_DRV_DMA_CFG](#)[TX_TRIG] to set the thresholds of RXFIFO and TXFIFO.

10.1.4.5 Writing Packet Transmission for TWI Driver

To write package to the device, perform the following steps:

1. Configure [TWI_DRV_SLV](#)[SLV_ID] to set the device ID, and configure [TWI_DRV_SLV](#)[CMD] to 0 to set the write operation.
2. Configure [TWI_DRV_FMT](#)[ADDR_BYTE] according to the address width of the device register, and [TWI_DRV_FMT](#)[DATA_BYTE] according to the written data count in a packet.
3. Configure [TWI_DRV_CFG](#)[PACKET_CNT] to set the written packet number.
4. Configure DMA channel, including TWI TXFIFO, device register address, and the written data.
5. Set [START_TRAN] to 1 to start TWI Driver transmission.
6. When TWI driver transmission completes, the interrupt is triggered, it indicates that the write packet transmission ends.

10.1.4.6 Reading Packet Transmission for TWI Driver

To read package from the device, perform the following steps:

1. Configure [TWI_DRV_SLV](#)[SLV_ID] to set the device ID, and configure [TWI_DRV_SLV](#)[CMD] to 1 to set the read operation.
2. Configure [TWI_DRV_FMT](#)[ADDR_BYTE] according to the address width of the device register, and [TWI_DRV_FMT](#)[DATA_BYTE] according to the read data count in a packet.
3. Configure [TWI_DRV_CFG](#)[PACKET_CNT] to set the read packet number.

4. Configure DMA channel, including TWI TXFIFO, TWI RXFIFO, device register address and the read data.
5. Set [START_TRAN] to 1 to start TWI Driver transmission.
6. When TWI driver transmission completes, the interrupt is triggered, it indicates that the read packet transmission ends.

10.1.5 Register List

| Module Name | Base Address | Comments |
|-------------|--------------|--------------------------------------|
| TWI0 | 0x0250_2000 | |
| TWI1 | 0x0250_2400 | TWI1 register is the same with TWI0. |
| TWI2 | 0x0250_2800 | TWI2 register is the same with TWI0. |
| TWI3 | 0x0250_2c00 | TWI3 register is the same with TWI0. |
| TWI4 | 0x0250_3000 | TWI4 register is the same with TWI0. |

| Register Name | Offset | Description |
|-----------------------|--------|---|
| TWI_ADDR | 0x0000 | TWI Slave address register |
| TWI_XADDR | 0x0004 | TWI Extend address register |
| TWI_DATA | 0x0008 | TWI Data register |
| TWI_CNTR | 0x000C | TWI Control register |
| TWI_STAT | 0x0010 | TWI Status register |
| TWI_CCR | 0x0014 | TWI Clock register |
| TWI_SRST | 0x0018 | TWI Soft reset register |
| TWI_EFR | 0x001C | TWI Enhance Feature register |
| TWI_LCR | 0x0020 | TWI Line Control register |
| TWI_DRV_CTRL | 0x0200 | TWI_DRV Control Register |
| TWI_DRV_CFG | 0x0204 | TWI_DRV Transmission Configuration Register |
| TWI_DRV_SLV | 0x0208 | TWI_DRV Slave ID Register |
| TWI_DRV_FMT | 0x020C | TWI_DRV Packet Format Register |
| TWI_DRV_BUS_CTRL | 0x0210 | TWI_DRV Bus Control Register |
| TWI_DRV_INT_CTRL | 0x0214 | TWI_DRV Interrupt Control Register |
| TWI_DRV_DMA_CFG | 0x0218 | TWI_DRV DMA Configure Register |
| TWI_DRV_FIFO_CON | 0x021C | TWI_DRV FIFO Content Register |
| TWI_DRV_SEND_FIFO_ACC | 0x0300 | TWI_DRV Send Data FIFO Access Register |
| TWI_DRV_RECV_FIFO_ACC | 0x0304 | TWI_DRV Receive Data FIFO Access Register |

10.1.6 Register Description

10.1.6.1 0x0000 TWI Slave Address Register (Default Value: 0x0000_0000)

| Offset: 0x0000 | | | Register Name: TWI_ADDR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:1 | R/W | 0x0 | SLA Slave Address For 7-bit addressing, the bit[7:1] indicates: SLA6, SLA5, SLA4, SLA3, SLA2, SLA1, SLA0 For 10-bit addressing, the bit[7:1] indicates: 1, 1, 1, 1, 0, SLAX[9:8] |
| 0 | R/W | 0x0 | GCE General Call Address Enable 0: Disable 1: Enable |



For 7-bit addressing:

SLA6 – SLA0 is the 7-bit address of the TWI in slave mode. When the TWI receives this address after a START condition, it will generate an interrupt and enter slave mode. (SLA6 corresponds to the first bit received from the TWI bus.) If GCE is set to ‘1’, the TWI will also recognize the general call address (00h).

For 10-bit addressing:

When the address received starts with b’11110, the TWI recognizes b’11110 as the first part of a 10-bit address and if the next two bits match ADDR[2:1] (For example, SLAX9 and SLAX8 for the extended address of the device), it sends an ACK. (The device does not generate an interrupt at this point.) If the next byte of the address matches the XADDR register (SLAX7 – SLAX0), the TWI generates an interrupt and goes into slave mode.

10.1.6.2 0x0004 TWI Extend Address Register (Default Value: 0x0000_0000)

| Offset: 0x0004 | | | Register Name: TWI_XADDR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:0 | R/W | 0x0 | SLAX Extend Slave Address SLAX[7:0] |

10.1.6.3 0x0008 TWI Data Register (Default Value: 0x0000_0000)

| Offset: 0x0008 | | | Register Name: TWI_DATA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:0 | R/W | 0x0 | TWI_DATA Data byte transmitted or received |

10.1.6.4 0x000C TWI Control Register (Default Value: 0x0000_0000)

| Offset: 0x000C | | | Register Name: TWI_CNTR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7 | R/W | 0x0 | INT_EN Interrupt Enable 0: The interrupt line always low 1: The interrupt line will go high when INT_FLAG is set. |
| 6 | R/W | 0x0 | BUS_EN TWI Bus Enable 0: The TWI bus SDA/SCL is ignored and the TWI controller will not respond to any address on the bus. 1: The TWI will respond to call to its slave address – and to the general call address if the GCE bit in the ADDR register is set. Note: In master operation mode, this bit should be set to '1'. |
| 5 | R/WAC | 0x0 | M_STA Master Mode Start When the M_STA is set to '1', the TWI controller enters master mode and will transmit a START condition on the bus when the bus is free. If the M_STA bit is set to '1' when the TWI controller is already in master mode and one or more bytes have been transmitted, then a repeated START condition will be sent. If the M_STA bit is set to '1' when the TWI is accessed in slave mode, the TWI will complete the data transfer in slave mode then enter master mode when the bus has been released. The M_STA bit is cleared automatically after a START condition has been sent. Writing a '0' to this bit has no effect. |

| Offset: 0x000C | | | Register Name: TWI_CNTR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 4 | R/W1C | 0x0 | <p>M_STP Master Mode Stop</p> <p>If the M_STP is set to '1' in master mode, a STOP condition is transmitted on the TWI bus. If the M_STP bit is set to '1' in slave mode, the TWI will indicate if a STOP condition has been received, but no STOP condition will be transmitted on the TWI bus. If both M_STA and M_STP bits are set, the TWI will first transmit the STOP condition (if in master mode), then transmit the START condition.</p> <p>The M_STP bit is cleared automatically. Writing a '0' to this bit has no effect.</p> |
| 3 | R/W1C | 0x0 | <p>INT_FLAG Interrupt Flag</p> <p>The INT_FLAG is automatically set to '1' when any of the 28 (out of the possible 29) states is entered (see 'STAT Register' below). The state that does not set INT_FLAG is state F8h. If the INT_EN bit is set, the interrupt line goes high when INT_FLAG is set to '1'. If the TWI is operating in slave mode, the data transfer is suspended when INT_FLAG is set and the low period of the TWI bus clock line (SCL) is stretched until '1' is written to INT_FLAG. The TWI clock line is then released and the interrupt line goes low.</p> |
| 2 | R/W | 0x0 | <p>A_ACK Assert Acknowledge</p> <p>When A_ACK is set to '1', an Acknowledge (low level on SDA) will be sent during the acknowledge clock pulse on the TWI bus if:</p> <ul style="list-style-type: none"> (1). Either the whole of a matching 7-bit slave address or the first or the second byte of a matching 10-bit slave address has been received. (2). The general call address has been received and the GCE bit in the ADDR register is set to '1'. (3). A data byte has been received in master or slave mode. <p>When A_ACK is '0', a Not Acknowledge (high level on SDA) will be sent when a data byte is received in master or slave mode.</p> <p>If A_ACK is cleared to '0' in slave transmitter mode, the byte in the DATA register is assumed to be the 'last byte'. After this byte has been transmitted, the TWI will enter state C8h then return to the idle state (status code F8h) when INT_FLAG is cleared.</p> <p>The TWI will not respond as a slave unless A_ACK is set.</p> |
| 1 | / | / | / |
| 0 | R/W | 0x0 | <p>CLK_COUNT_MODE</p> <p>0: scl clock high period count on oscl 1: scl clock high period count on iscl</p> |

10.1.6.5 0x0010 TWI Status Register (Default Value: 0x0000_00F8)

| Offset: 0x0010 | | | Register Name: TWI_STAT |
|----------------|------------|-------------|-------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |



| Offset: 0x0010 | | | Register Name: TWI_STAT |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 7:0 | R | 0xF8 | <p>STA</p> <p>Status Information Byte</p> <p>Code Status</p> <p>0x00: Bus error</p> <p>0x08: START condition transmitted</p> <p>0x10: Repeated START condition transmitted</p> <p>0x18: Address + Write bit transmitted, ACK received</p> <p>0x20: Address + Write bit transmitted, ACK not received</p> <p>0x28: Data byte transmitted in master mode, ACK received</p> <p>0x30: Data byte transmitted in master mode, ACK not received</p> <p>0x38: Arbitration lost in address or data byte</p> <p>0x40: Address + Read bit transmitted, ACK received</p> <p>0x48: Address + Read bit transmitted, ACK not received</p> <p>0x50: Data byte received in master mode, ACK transmitted</p> <p>0x58: Data byte received in master mode, not ACK transmitted</p> <p>0x60: Slave address + Write bit received, ACK transmitted</p> <p>0x68: Arbitration lost in the address as master, slave address + Write bit received, ACK transmitted</p> <p>0x70: General Call address received, ACK transmitted</p> <p>0x78: Arbitration lost in the address as master, General Call address received, ACK transmitted</p> <p>0x80: Data byte received after slave address received, ACK transmitted</p> <p>0x88: Data byte received after slave address received, not ACK transmitted</p> <p>0x90: Data byte received after General Call received, ACK transmitted</p> <p>0x98: Data byte received after General Call received, not ACK transmitted</p> <p>0xA0: STOP or repeated START condition received in slave mode</p> <p>0xA8: Slave address + Read bit received, ACK transmitted</p> <p>0xB0: Arbitration lost in the address as master, slave address + Read bit received, ACK transmitted</p> <p>0xB8: Data byte transmitted in slave mode, ACK received</p> <p>0xC0: Data byte transmitted in slave mode, ACK not received</p> <p>0xC8: The Last byte transmitted in slave mode, ACK received</p> <p>0xD0: Second Address byte + Write bit transmitted, ACK received</p> <p>0xD8: Second Address byte + Write bit transmitted, ACK not received</p> <p>0xF8: No relevant status information, INT_FLAG=0</p> <p>Others: Reserved</p> |

10.1.6.6 0x0014 TWI Clock Register (Default Value: 0x0000_0080)

| Offset: 0x0014 | | | Register Name: TWI_CCR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7 | R/W | 0x1 | CLK_DUTY Setting duty cycle of clock as master 0: 50% 1: 40% |
| 6:3 | R/W | 0x0 | CLK_M The TWI SCL output frequency, in master mode, is F1/10: $F1 = F0/(CLK_M + 1)$ $Fscl = F1/10 = Fin/(2^CLK_N * (CLK_M + 1)*10)$ Specially, Fscl = F1/11 when CLK_M=0 and CLK_DUTY=40% due to the delay of SCL sample debounce. |
| 2:0 | R/W | 0x0 | CLK_N The TWI bus is sampled by the TWI at the frequency defined by F0: $Fsamp = F0 = Fin/2^CLK_N$ The TWI SCL output frequency, in master mode, is F1/10: $F1 = F0/(CLK_M + 1)$ $Fscl = F1/10 = Fin/(2^CLK_N * (CLK_M + 1)*10)$ Specially, Fscl = F1/11 when CLK_M=0 and CLK_DUTY=40% due to the delay of SCL sample debounce. For Example: Fin = 24 MHz (APB clock input) For 400 kHz full speed 2-wire, CLK_N = 1, CLK_M = 2 $F0 = 24 \text{ MHz}/2^1 = 12 \text{ MHz}$, $F1 = F0/(10*(2+1)) = 0.4 \text{ MHz}$ For 100 kHz standard speed 2-wire, CLK_N = 1, CLK_M = 11 $F0 = 24 \text{ MHz}/2^1 = 12 \text{ MHz}$, $F1 = F0/(10*(11+1)) = 0.1 \text{ MHz}$ |

10.1.6.7 0x0018 TWI Soft Reset Register (Default Value: 0x0000_0000)

| Offset: 0x0018 | | | Register Name: TWI_SRST |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:1 | / | / | / |
| 0 | R/WAC | 0x0 | SOFT_RST Soft Reset Write '1' to this bit to reset the TWI and clear to '0' when completing Soft Reset operation. |

10.1.6.8 0x001C TWI Enhance Feature Register (Default Value: 0x0000_0000)

| Offset: 0x001C | | | Register Name: TWI_EFR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:2 | / | / | / |
| 1:0 | R/W | 0x0 | <p>DBN Data Byte Number Follow Read Command Control</p> <p>00: No data byte can be written after the read command</p> <p>01: Only 1-byte data can be written after the read command</p> <p>10: 2-bytes data can be written after the read command</p> <p>11: 3-bytes data can be written after the read command</p> |

10.1.6.9 0x0020 TWI Line Control Register (Default Value: 0x0000_003A)

| Offset: 0x0020 | | | Register Name: TWI_LCR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:6 | / | / | / |
| 5 | R | 0x1 | <p>SCL_STATE Current State of TWI_SCL</p> <p>0: Low 1: High</p> |
| 4 | R | 0x1 | <p>SDA_STATE Current State of TWI_SDA</p> <p>0: Low 1: High</p> |
| 3 | R/W | 0x1 | <p>SCL_CTL TWI_SCL Line State Control Bit</p> <p>When the line control mode is enabled (bit[2] is set), this bit decides the output level of TWI_SCL.</p> <p>0: Output low level 1: Output high level</p> |
| 2 | R/W | 0x0 | <p>SCL_CTL_EN TWI_SCL Line State Control Enable</p> <p>When this bit is set, the state of TWI_SCL is controlled by the value of bit[3].</p> <p>0: Disable TWI_SCL line control mode 1: Enable TWI_SCL line control mode</p> |
| 1 | R/W | 0x1 | <p>SDA_CTL TWI_SDA Line State Control Bit</p> <p>When the line control mode is enabled (bit[0] is set), this bit decides the output level of TWI_SDA.</p> <p>0: Output low level 1: Output high level</p> |

| Offset: 0x0020 | | | Register Name: TWI_LCR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 0 | R/W | 0x0 | <p>SDA_CTL_EN TWI_SDA Line State Control Enable When this bit is set, the state of TWI_SDA is controlled by the value of bit[1]. 0: Disable TWI_SDA line control mode 1: Enable TWI_SDA line control mode</p> |

10.1.6.10 0x0200 TWI_DRV Control Register (Default Value: 0x00F8_1000)

| Offset: 0x0200 | | | Register Name: TWI_DRV_CTRL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31 | R/WAC | 0x0 | <p>START_TRAN Start transmission 0: Transmission idle 1: Start transmission Automatically cleared to '0' when finished. If the slave is not responding for the expected status over the time defined by TIMEOUT, the current transmission will stop. All setting formats and data will be loaded from registers and FIFO when the transmission starts.</p> |
| 30 | / | / | / |
| 29 | R/W | 0x0 | <p>RESTART_MODE Restart mode 0: RESTART 1: STOP+START Define the TWI_DRV action after sending the register address.</p> |
| 28 | R/W | 0x0 | <p>READ_TRAN_MODE Read transition mode 0: Send slave_id+W 1: Not send slave_id+W Setting this bit to 1 if reading from a slave in which the register width is equal to 0.</p> |
| 27:24 | R | 0x0 | <p>TRAN_RESULT Transition result 000: OK 001: FAIL Other: Reserved</p> |

| Offset: 0x0200 | | | Register Name: TWI_DRV_CTRL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 23:16 | R | 0xf8 | <p>TWI_STA TWI status 0x00: bus error 0x08: START condition transmitted 0x10: Repeated START condition transmitted 0x18: Address + Write bit transmitted, ACK received 0x20: Address + Write bit transmitted, ACK not received 0x28: Data byte transmitted in master mode, ACK received 0x30: Data byte transmitted in master mode, ACK not received 0x38: Arbitration lost in address or data byte 0x40: Address + Read bit transmitted, ACK received 0x48: Address + Read bit transmitted, ACK not received 0x50: Data byte received in master mode, ACK received 0x58: Data byte received in master mode, ACK not received 0x01: Timeout when sending the 9th SCL clock Other: Reserved</p> |
| 15:8 | R/W | 0x10 | <p>TIMEOUT_N Timeout number When sending the 9th clock, assert fail signal when the slave device does not respond after N*FSCL cycles. And the software must do a reset to the TWI_DRV module and send a stop condition to slave.</p> |
| 7:2 | / | / | / |
| 1 | R/W | 0x0 | <p>SOFT_RESET Software reset 0: Normal 1: Reset</p> |
| 0 | R/W | 0x0 | <p>TWI_DRV_EN TWI driver enable 0: Module disable 1: Module enable (only use in TWI Master Mode)</p> |

10.1.6.11 0x0204 TWI_DRV Transmission Configuration Register (Default Value: 0x0000_0001)

| Offset: 0x0204 | | | Register Name: TWI_DRV_CFG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | R/W | 0x0 | <p>PKT_INTERVAL Define the interval between each packet for PKT_INTERVAL FSCL cycles.</p> |
| 15:0 | R/W | 0x1 | <p>PACKET_CNT The FIFO data is transmitted as PACKET_CNT packets in current format.</p> |

10.1.6.12 0x0208 TWI_DRV Slave ID Register (Default Value: 0x0000_0000)

| Offset: 0x0208 | | | Register Name: TWI_DRV_SLV |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:9 | R/W | 0x0 | SLV_ID Slave device ID For 7-bit addressing, the bit[7:1] indicates: SLA6, SLA5, SLA4, SLA3, SLA2, SLA1, SLA0 For 10-bit addressing, the bit[7:1] indicates: 1, 1, 1, 1, 0, SLAX[9:8] |
| 8 | R/W | 0x0 | CMD R/W operation to slave device 0: Write 1: Read |
| 7:0 | R/W | 0x0 | SLV_ID_X SLAX[7:0] The low 8 bits for slave device ID with 10-bit addressing. |

10.1.6.13 0x020C TWI_DRV Packet Format Register (Default Value: 0x0001_0001)

| Offset: 0x020C | | | Register Name: TWI_DRV_FMT |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:24 | / | / | / |
| 23:16 | R/W | 0x1 | ADDR_BYTEx How many bytes be sent as slave device reg address 0~255 |
| 15:0 | R/W | 0x1 | DATA_BYTEx How many bytes be sent/received as data 1~65535 |

10.1.6.14 0x0210 TWI_DRV Bus Control Register (Default Value: 0x0000_80C0)

| Offset: 0x0210 | | | Register Name: TWI_DRV_BUS_CTRL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:17 | / | / | / |
| 16 | W | 0x0 | CLK_COUNT_MODE Clock count mode 0: scl clock high period count on oscl 1: scl clock high period count on iscl |

| Offset: 0x0210 | | | Register Name: TWI_DRV_BUS_CTRL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 15 | R/W | 0x1 | CLK_DUTY Setting duty cycle of clock as Master 0: 50% 1: 40% |
| 14:12 | R/W | 0x0 | CLK_N TWI_DRV bus sampling clock F0=24MHz/2^CLK_N |
| 11:8 | R/W | 0x0 | CLK_M TWI_DRV output SCL frequency is FSCL=F1/10=(F0/(CLK_M+1))/10 Specially, Fosc = F1/11 when CLK_M=0 and CLK_DUTY=40% due to the delay of SCL sample debounce. |
| 7 | R | 0x1 | SCL_STA SCL current status |
| 6 | R | 0x1 | SDA_STA SDA current status |
| 5:4 | / | / | / |
| 3 | R/W | 0x0 | SCL_MOV SCL manual output value |
| 2 | R/W | 0x0 | SDA_MOV SDA manual output value |
| 1 | R/W | 0x0 | SCL_MOE SCL manual output enable |
| 0 | R/W | 0x0 | SDA_MOE SDA manual output enable |

10.1.6.15 0x0214 TWI_DRV Interrupt Control Register (Default Value: 0x0000_0000)

| Offset: 0x0214 | | | Register Name: TWI_DRV_INT_CTRL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:18 | / | / | / |
| 19 | R/W | 0x0 | RX_REQ_INT_EN If set, an interrupt is sent when RX_REQ_PD sets. |
| 18 | R/W | 0x0 | TX_REQ_INT_EN If set, an interrupt is sent when TX_REQ_PD sets. |
| 17 | R/W | 0x0 | TRAN_ERR_INT_EN If set, an interrupt is sent when TRAN_ERR_PD sets. |
| 16 | R/W | 0x0 | TRAN_COM_INT_EN If set, an interrupt is sent when TRAN_COM_PD sets. |
| 15:4 | / | / | / |
| 3 | R/W1C | 0x0 | RX_REQ_PD Set when the data byte number in RECV_FIFO reaches RX_TRIG. |

| Offset: 0x0214 | | | Register Name: TWI_DRV_INT_CTRL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 2 | R/W1C | 0x0 | TX_REQ_PD Set when there is no less than DMA_TX_TRIG empty byte number in SEND_FIFO. |
| 1 | R/W1C | 0x0 | TRAN_ERR_PD Packet transmission failure pending |
| 0 | R/W1C | 0x0 | TRAN_COM_PD Packet transmission completion pending |

10.1.6.16 0x0218 TWI_DRV DMA Configure Register (Default Value: 0x0010_0010)

| Offset: 0x0218 | | | Register Name: TWI_DRV_DMA_CFG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:25 | / | / | / |
| 24 | R/W | 0x0 | DMA_RX_EN DMA RX Enable |
| 23:22 | / | / | / |
| 21:16 | R/W | 0x10 | RX_TRIG RX trigger When DMA_RX_EN is set, send DMA RX Req when the data byte number in RECV_FIFO reaches RX_TRIG, or the read transmission is completed, the data of RECV_FIFO does not reach RX_TRIG but as long as the RECV_FIFO is not empty. |
| 15:9 | / | / | / |
| 8 | R/W | 0x0 | DMA_TX_EN DMA TX Enable |
| 7:6 | / | / | / |
| 5:0 | R/W | 0x10 | TX_TRIG TX trigger When DMA_TX_EN is set, send DMA TX Req when the space of SEND_FIFO (FIFO Level – data volume) reaches TX_TRIG. |

10.1.6.17 0x021C TWI_DRV FIFO Content Register (Default Value: 0x0000_0000)

| Offset: 0x021C | | | Register Name: TWI_DRV_FIFO_CON |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:23 | / | / | / |
| 22 | R/WAC | 0x0 | RECV_FIFO_CLEAR Set this bit to clear RECV_FIFO pointer, and this bit is cleared automatically. |
| 21:16 | R | 0x0 | RECV_FIFO_CONTENT The number of data in RECV_FIFO |

| Offset: 0x021C | | | Register Name: TWI_DRV_FIFO_CON |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 15:7 | / | / | / |
| 6 | R/WAC | 0x0 | SEND_FIFO_CLEAR Set this bit to clear SEND_FIFO pointer, and this bit is cleared automatically. |
| 5:0 | R | 0x0 | SEND_FIFO_CONTENT The number of data in SEND_FIFO |

10.1.6.18 0x0300 TWI_DRV Send Data FIFO Access Register (Default Value: 0x0000_0000)

| Offset: 0x0300 | | | Register Name: TWI_DRV_SEND_FIFO_ACC |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:0 | W | 0x0 | SEND_DATA_FIFO Address of a 32x8 SEND_FIFO, which stores reg address and data sending to the slave device. |

10.1.6.19 0x0304 TWI_DRV Receive Data FIFO Access Register (Default Value: 0x0000_0000)

| Offset: 0x0304 | | | Register Name: TWI_DRV_RECV_FIFO_ACC |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:0 | R | 0x0 | RECV_DATA_FIFO Address of a 32x8 RECV_FIFO, which stores data received from the slave device. |

10.2 UART

10.2.1 Overview

The universal asynchronous receiver transmitter (UART) provides an asynchronous serial communication with external devices, modem (data carrier equipment, DCE). It performs serial-to-parallel conversion on the data received from peripherals and transmits the converted data to the internal bus. It also performs parallel-to-serial conversion on the data that is transmitted to peripherals.

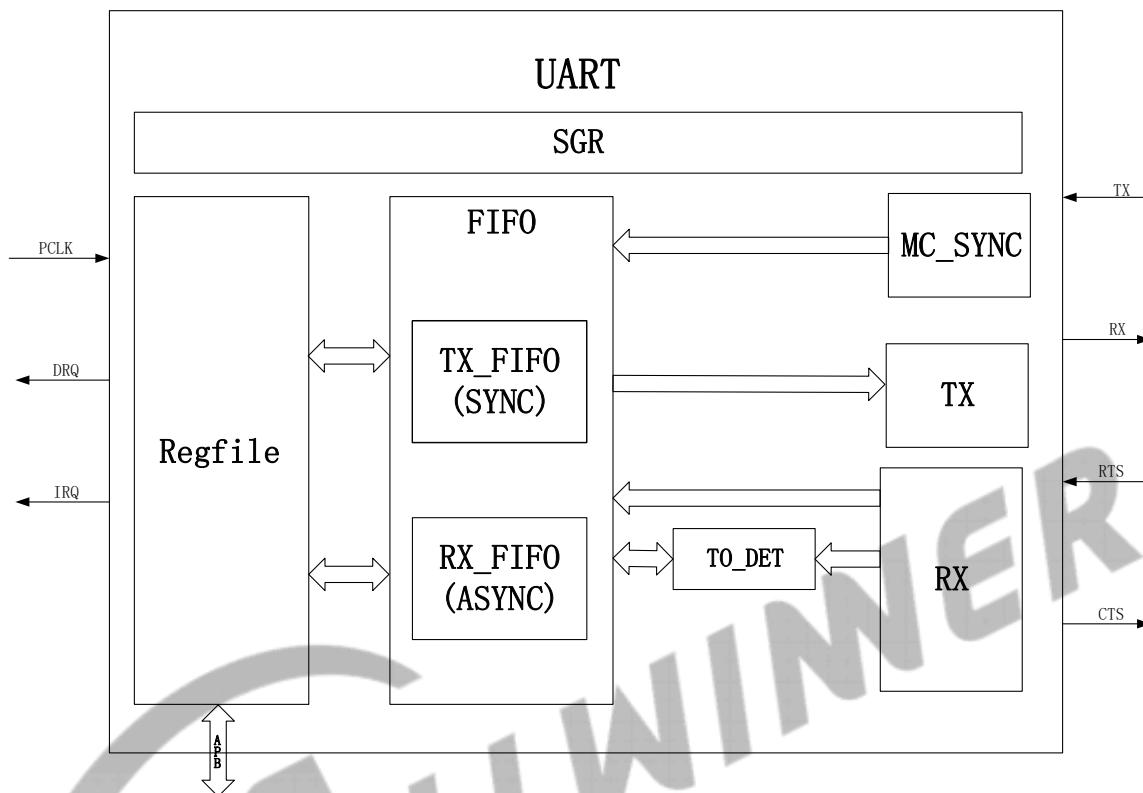
The UART has the following features:

- Compatible with industry-standard 16450/16550 UARTs
- Two separate FIFOs: one is RX FIFO, and the other is TX FIFO
 - Each of them is 64 bytes (for UART0)
 - Each of them is 128 bytes (for UART1, UART2, and UART3)
- The working reference clock is from the APB bus clock
 - Speed up to 4 Mbit/s with 64 MHz APB clock
 - Speed up to 1.5 Mbit/s with 24 MHz APB clock
- Supports programmable parity (even, odd, or no parity)
- Supports TX/RX DMA slave controller interface
- Supports software/hardware flow control
- Supports auto-flow by using CTS & RTS (Only for UART1/2/3)
- Support IrDA 1.0 SIR
- Support RS-485/8-bit mode
- Supports 5-8 data bits and 1/1.5/2 stop bits

10.2.2 Block Diagram

The following figure shows a block diagram of the UART.

Figure 10-9 UART Block Diagram



10.2.3 Functional Description

10.2.3.1 External Signals

The following table describes the external signals of UART.

Table 10-4 UART External Signals

| Signal | Description | Type |
|--------------|--------------------|------|
| UART0 | | |
| UART0_TX | Serial Data Output | O |
| UART0_RX | Serial Data Input | I |
| UART1 | | |
| UART1_TX | Serial Data Output | O |
| UART1_RX | Serial Data Input | I |
| UART1_CTS | Clear to Send | I |
| UART1_RTS | Request to Send | O |
| UART2 | | |
| UART2_TX | Serial Data Output | O |
| UART2_RX | Serial Data Input | I |
| UART2_CTS | Clear to Send | I |

| Signal | Description | Type |
|-----------|--------------------|------|
| UART2_RTS | Request to Send | O |
| UART3 | | |
| UART3_TX | Serial Data Output | O |
| UART3_RX | Serial Data Input | I |
| UART3_CTS | Clear to Send | I |
| UART3_RTS | Request to Send | O |

10.2.3.2 Clock Sources

The following table describes the clock sources of UART.

Table 10-5 UART Clock Sources

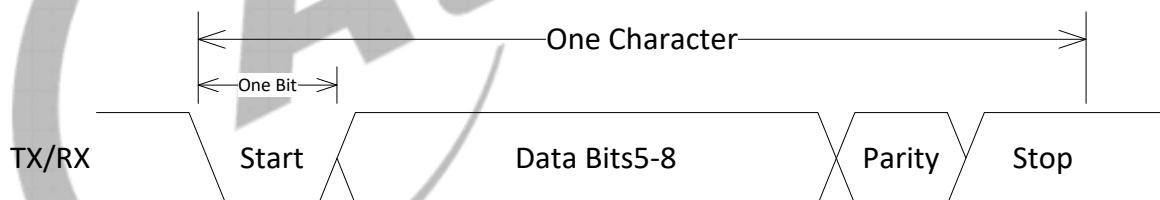
| Clock Sources | Description |
|---------------|--|
| APB1 Bus | UART clock source. Refer to section 3.4 Clock Controller Unit (CCU) for details on APB1. |

10.2.3.3 Typical Applications and Timing Diagram

UART Serial Data Format

The following figure shows the UART serial data format. The start bit, data bit, parity bit, and stop bit can be configured.

Figure 10-10 UART Serial Data Format



Using UART for RTS/CTS Autoflow Control

The following figure shows the typical application diagram for RTS/CTS autoflow control. Figure 9-12 shows the data format of the RTS/CTS autoflow control.

Figure 10-11 Application Diagram for RTS/CTS Autoflow Control

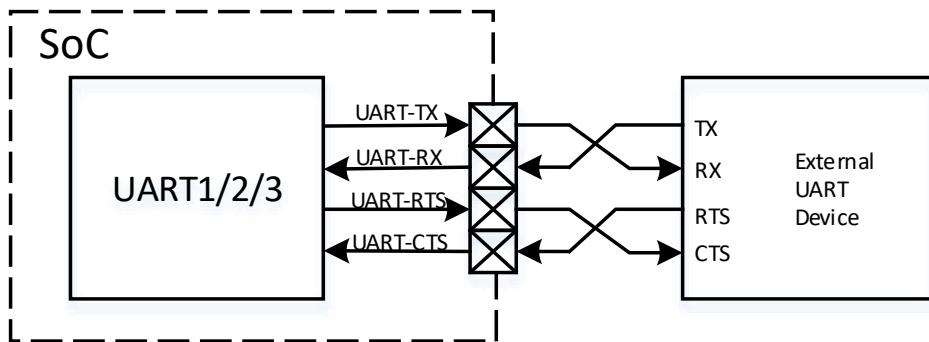
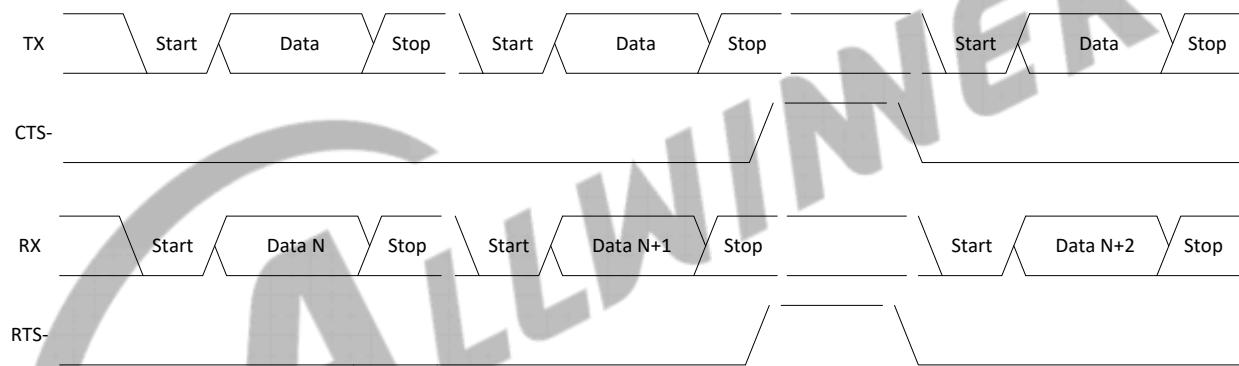


Figure 10-12 RTS/CTS Autoflow Control Data Format



Using UART for Serial IrDA

The following figure shows the application diagram for the IrDA transceiver. Figure 9-14 shows the data format of the serial IrDA.

Figure 10-13 Application Diagram for IrDA Transceiver

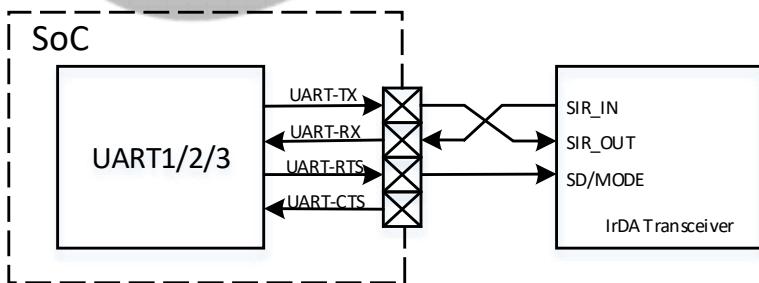
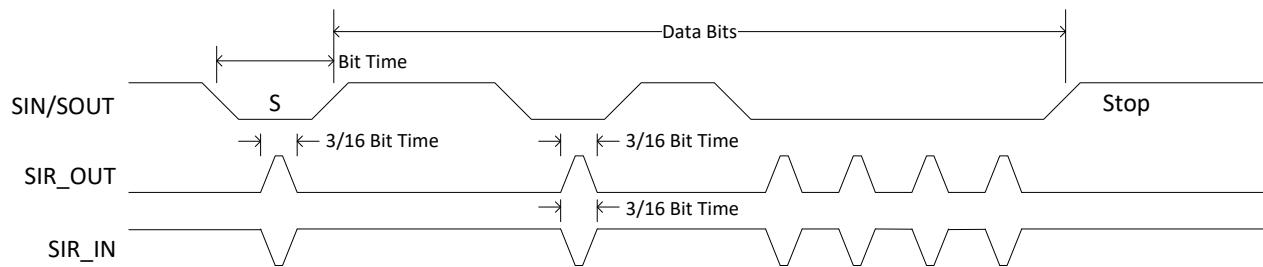
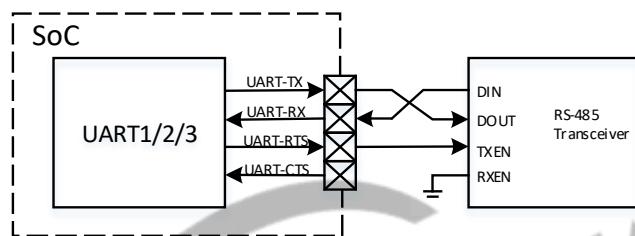
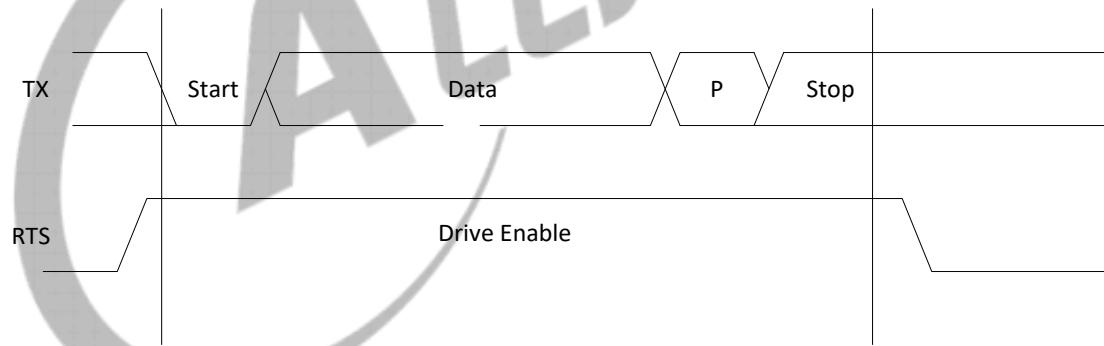


Figure 10-14 Serial IrDA Data Format

Using UART for RS-485

The following figure shows the application diagram for the RS-485 transceiver. Figure 9-16 shows the data format of the RS-485.

Figure 10-15 Application Diagram for RS-485 Transceiver**Figure 10-16 RS-485 Data Format**

10.2.3.4 UART Operating Mode

Data Frame Format

The [UART_LCR \(Offset: 0x000C\)](#) register can set the basic parameter of a data frame: data width (5 to 8 bits), stop bit number (1/1.5/2), parity type.

A frame transfer of the UART includes the start signal, data signal, parity bit, and stop signal. The LSB is transmitted first.

- Start signal (start bit): It is the start flag of a data frame. According to the UART protocol, the low level of the TXD signal indicates the start of a data frame. When the UART transmits data, the level needs to hold high.
- Data signal (data bit): The data bit width can be configured as 5-bit, 6-bit, 7-bit, and 8-bit through different applications.
- Parity bit: It is a 1-bit error correction signal. Parity bit includes odd parity, even parity. The UART can enable and disable the parity bit by setting the [UART_LCR \(Offset: 0x000C\)](#) register.
- Stop Signal (stop bit): It is the stop bit of a data frame. The stop bit can be set to 1-bit, 1.5-bit, and 2-bit by the [UART_LCR \(Offset: 0x000C\)](#) register. The high level of the TXD signal indicates the end of a data frame.

Baud and Error Rates

The baud rate is calculated as follows: Baud rate = SCLK/(16 * divisor).

The SCLK is usually APB2 or APS2 and can be set in section 3.4 Clock Controller Unit (CCU).

The divisor is a frequency divider of UART. The frequency divider has 16-bit, the low 8-bit is in the [UART_DLL \(Offset: 0x0000\)](#) register, the high 8-bit is in the [UART_DLH \(Offset: 0x0004\)](#) register.

The relationship between the different UART mode and the error rate is as follows.

Table 10-6 UART Mode Baud and Error Rates

| Clock source | Divisor | Baud rate | Over sampling | Error(%) |
|--------------|---------|-----------|---------------|----------|
| 24000000 | 5000 | 300 | 16 | 0 |
| 24000000 | 2500 | 600 | 16 | 0 |
| 24000000 | 1250 | 1200 | 16 | 0 |
| 24000000 | 625 | 2400 | 16 | 0 |
| 24000000 | 313 | 4800 | 16 | -0.16 |
| 24000000 | 156 | 9600 | 16 | 0.16 |
| 24000000 | 78 | 19200 | 16 | 0.16 |
| 24000000 | 39 | 38400 | 16 | 0.16 |
| 24000000 | 26 | 57600 | 16 | 0.16 |
| 24000000 | 13 | 115200 | 16 | 0.16 |
| 48000000 | 13 | 230400 | 16 | 0.16 |
| 64000000 | 7 | 576000 | 16 | -0.794 |
| 75000000 | 5 | 921600 | 16 | 1.725 |
| 48000000 | 3 | 1000000 | 16 | 0 |
| 24000000 | 1 | 1500000 | 16 | 0 |
| 48000000 | 1 | 3000000 | 16 | 0 |
| 64000000 | 1 | 4000000 | 16 | 0 |

Table 10-7 IrDA Mode Baud and Error Rates

| Clock source | Divisor | Baud rate | Encoding | Error(%) |
|--------------|---------|-----------|----------|----------|
| 24000000 | 5000 | 300 | 3/16 | 0 |
| 24000000 | 2500 | 600 | 3/16 | 0 |
| 24000000 | 1250 | 1200 | 3/16 | 0 |
| 24000000 | 625 | 2400 | 3/16 | 0 |
| 24000000 | 313 | 4800 | 3/16 | -0.16 |
| 24000000 | 156 | 9600 | 3/16 | 0.16 |
| 24000000 | 78 | 19200 | 3/16 | 0.16 |
| 24000000 | 39 | 38400 | 3/16 | 0.16 |
| 24000000 | 26 | 57600 | 3/16 | 0.16 |
| 24000000 | 13 | 115200 | 3/16 | 0.16 |

Table 10-8 RS485 Mode Baud and Error Rates

| Clock source | Divisor | Baud rate | Encoding | Error(%) |
|--------------|---------|-----------|----------|----------|
| 24000000 | 5000 | 300 | 16 | 0 |
| 24000000 | 2500 | 600 | 16 | 0 |
| 24000000 | 1250 | 1200 | 16 | 0 |
| 24000000 | 625 | 2400 | 16 | 0 |
| 24000000 | 313 | 4800 | 16 | -0.16 |
| 24000000 | 156 | 9600 | 16 | 0.16 |
| 24000000 | 78 | 19200 | 16 | 0.16 |
| 24000000 | 39 | 38400 | 16 | 0.16 |
| 24000000 | 26 | 57600 | 16 | 0.16 |
| 24000000 | 13 | 115200 | 16 | 0.16 |

DLAB Definition

The DLAB control bit ([UART_LCR \(Offset: 0x000C\)](#) [7]) is the access control bit of the divisor latch register.

If DLAB is 0, then the 0x00 offset address is the [UART_RBR \(Offset: 0x0000\)](#)/[UART_THR \(Offset: 0x0000\)](#) (RX/TX FIFO) register, and the 0x04 offset address is the [UART_IER \(Offset: 0x0004\)](#) register.

If DLAB is 1, then the 0x00 offset address is the [UART_DLL \(Offset: 0x0000\)](#) register, and the 0x04 offset address is the [UART_DLH \(Offset: 0x0004\)](#) register.

When the UART initials, the divisor needs to be set. That is, writing 1 to DLAB can access the [UART_DLL \(Offset: 0x0000\)](#) and [UART_DLH \(Offset: 0x0004\)](#) register, after finished the configuration, writing 0 to DLAB can access the [UART_RBR \(Offset: 0x0000\)](#)/[UART_THR \(Offset: 0x0000\)](#) register.

CHCFG_AT_BUSY Definition

The function of the CHCFG_AT_BUSY ([UART_HALT \(Offset: 0x00A4\)](#) [1]) and CHANGE_UPDATE ([UART_HALT \(Offset: 0x00A4\)](#) [2]) are as follows.

CHCFG_AT_BUSY: Enable the bit, the software can also set the UART controller when UART is busy, such as the [UART_LCR \(Offset: 0x000C\)](#), [UART_DLH \(Offset: 0x0004\)](#), [UART_DLL \(Offset: 0x0000\)](#) register.

CHANGE_UPDATE: If CHCFG_AT_BUSY is enabled, and CHANGE_UPDATE is written to 1, the configuration of the UART controller can be updated. After completed the update, the bit is cleared to 0 automatically.

Setting divisor performs the following steps:

Write 1 to CHCFG_AT_BUSY to enable “configure at busy”.

Write 1 to DLAB ([UART_LCR \(Offset: 0x000C\)](#) [7]) and set the [UART_DLH \(Offset: 0x0004\)](#) and [UART_DLL \(Offset: 0x0000\)](#) registers.

Write 1 to CHANGE_UPDATE to update the configuration. The bit is cleared to 0 automatically after completing the update.

UART Busy Flag

The [UART_USR \(Offset: 0x007C\)](#) [0] is a busy flag of the UART controller.

When the TX transmits data, or the RX receives data, or the TX FIFO is not empty, or the RX FIFO is not empty, then the busy flag bit can be set to 1 by hardware, which indicates the UART controller is busy.

10.2.4 Programming Guidelines

The following takes the UART module in the CPUX domain as an example.

10.2.4.1 Initialization

1. System Initialization

- Configure [APB1_CFG_REG \(Offset: 0x0524\)](#) in the CCU module to set the APB1 bus clock (The clock is 24MHz by default).
- Set [UART_BGR_REG \(Offset: 0x090C\)](#)[UARTx_GATING] to 1 to enable the module clock, and set [UART_BGR_REG \(Offset: 0x090C\)](#) [UARTx_RST] to 1 to de-assert the module.

2. UART Controller Initialization

- IO configuration: Configure GPIO multiplex as UART function, and set UART pins to internal pull-up mode (For detail, see the description in section 10.5 GPIO).
- Baud-rate configuration:
 - Set UART baud-rate (refer to section 10.2.3.4 UART Operating Mode);
 - Write [UART_FCR \(Offset: 0x0008\)](#) [FIFOE] to 1 to enable TX/RX FIFO;
 - Write [UART_HALT \(Offset: 0x00A4\)](#) [HALT_TX] to 1 to disable TX transfer;

- Set [UART_LCR \(Offset: 0x000C\)](#) [DLAB] to 1, remain default configuration for other bits; set 0x00 offset address to the [UART_DLL \(Offset: 0x0000\)](#) register, set 0x04 offset address to the [UART_DLH \(Offset: 0x0004\)](#) register;
- Write the high 8-bit of divisor to the [UART_DLH \(Offset: 0x0004\)](#) register, and write the low 8-bit of divisor to the [UART_DLL \(Offset: 0x0000\)](#) register;
- Set [UART_LCR \(Offset: 0x000C\)](#) [DLAB] to 0, remain default configuration for other bits; set 0x00 offset address to the [UART_RBR \(Offset: 0x0000\)](#)/[UART_THR \(Offset: 0x0000\)](#) register, set 0x04 offset address to the [UART_IER \(Offset: 0x0004\)](#) register;
- Set [UART_HALT \(0x00A4\)](#) [HALT_TX] to 0 to enable TX transfer.

3. Controller Parameter Configuration

- Set data width, stop bits, and even/odd parity type by writing the [UART_LCR \(Offset: 0x000C\)](#) register.
- Reset, enable FIFO and set FIFO trigger condition by writing the [UART_FCR \(Offset: 0x0008\)](#) register.
- Set the flow control parameter by writing the [UART_MCR \(Offset: 0x0010\)](#) register.

4. Interrupt Configuration

- Configure UART interrupt vector number to request UART interrupt.
- In DMA mode, write [UART_IER \(Offset: 0x0004\)](#) to 0 to disable interrupt; write [UART_HSK \(Offset: 0x0088\)](#) [Handshake configuration] to 0xE5 to set DMA handshake mode; write [UART_FCR \(Offset: 0x0008\)](#) [DMAM] to 1 to set DMA transmission/reception mode; set DMA parameter and request DMA interrupt according to DMA configuration process.
- In Interrupt mode, configure [UART_IER \(Offset: 0x0004\)](#) to enable the corresponding interrupt according to requirements: such as transmit (TX) interrupt, receive (RX) interrupt, receive line status interrupt, RS48 interrupt, etc. (Here TX/RX interrupt is usually used).

10.2.4.2 Transferring/Receiving Data in Query Mode

Data transfer

1. Write data to [UART_THR \(Offset: 0x0000\)](#) to start data transfer.
2. Check TX_FIFO status by reading [UART_USR \(Offset: 0x007C\)](#) [TFNF]. If the bit is 1, data can continue to be written; if the bit is 0, wait for data transfer, and data cannot continue to write until FIFO is not full.

Data receive

1. Check RX_FIFO status by reading [UART_USR \(Offset: 0x007C\)](#) [RFNE].
2. Read data from [UART_RBR \(Offset: 0x0000\)](#) if RX_FIFO is not empty.
3. If [UART_USR \(Offset: 0x007C\)](#) [RFNE] is 0, data is received completely.

10.2.4.3 Transferring/Receiving Data in Interrupt Mode

Data transfer

1. Set [UART_IER \(Offset: 0x0004\)](#) [ETBEI] to 1 to enable the UART transmission interrupt.
2. Write the data to be transmitted to [UART_THR \(Offset: 0x0000\)](#).
3. When the data of TX_FIFO meets trigger condition (such as FIFO/2, FIFO/4), the UART transfer interrupt is generated.
4. Check [UART_USR \(Offset: 0x007C\)](#) [TFE] and determine whether TX_FIFO is empty. If [UART_USR \(Offset: 0x007C\)](#) [TFE] is 1, it indicates that the data in TX_FIFO is transmitted completely.
5. Clear [UART_IER \(Offset: 0x0004\)](#) [ETBEI] to 0 to disable transfer interrupt.

Data receive

1. Set [UART_IER \(Offset: 0x0004\)](#) [ERBFI] to 1 to enable the UART reception interrupt.
2. When the received data from RX_FIFO meets trigger condition (such as FIFO/2, FIFO/4), the UART receive interrupt is generated.
3. Read data from [UART_RBR \(Offset: 0x0000\)](#).
4. Check RX_FIFO status by reading [UART_USR \(Offset: 0x007C\)](#) [RFNE] and determine whether to read data. If the bit is 1, continue to read data from [UART_RBR \(Offset: 0x0000\)](#) until [UART_USR \(Offset: 0x007C\)](#) [RFNE] is cleared to 0, which indicates data is received completely.

10.2.4.4 Transferring/Receiving Data in DMA Mode

Data transfer

1. Configure the UART DMA interrupt according to the initialization process.
2. Configure DMA data channel, including the transfer source address, the transfer destination address, the number of data to be transferred, and the transfer type, and so on. (For details, see section 3.10 Direct Memory Access Controller (DMAC)).
3. Enable the DMA transfer function of the UART by setting the register of the DMA module.
4. Determine whether UART data is transferred completely based on the DMA status. If all data is transferred completely, disable the DMA transfer function of the UART.

Data receive

1. Configure DMA data channel, including the transfer source address, the transfer destination address, the number of data to be transferred, and the transfer type, and so on. (For details, see section 3.10 Direct Memory Access Controller (DMAC)).
2. Enable the DMA receive function of the UART by setting the register of the DMA module.
3. Determine whether UART data is received completely based on the DMA status. If all data is received completely, disable the DMA receive function of the UART.

10.2.5 Register List

| Module Name | Base Address |
|-------------|--------------|
| UART0 | 0x02500000 |
| UART1 | 0x02500400 |
| UART2 | 0x02500800 |
| UART3 | 0x02500C00 |

| Register Name | Offset | Description |
|-----------------|--------|---|
| UART_RBR | 0x0000 | UART Receive Buffer Register |
| UART_THR | 0x0000 | UART Transmit Holding Register |
| UART_DLL | 0x0000 | UART Divisor Latch Low Register |
| UART_DLH | 0x0004 | UART Divisor Latch High Register |
| UART_IER | 0x0004 | UART Interrupt Enable Register |
| UART_IIR | 0x0008 | UART Interrupt Identity Register |
| UART_FCR | 0x0008 | UART FIFO Control Register |
| UART_LCR | 0x000C | UART Line Control Register |
| UART_MCR | 0x0010 | UART Modem Control Register |
| UART_LSR | 0x0014 | UART Line Status Register |
| UART_MSR | 0x0018 | UART Modem Status Register |
| UART_SCH | 0x001C | UART Scratch Register |
| UART_USR | 0x007C | UART Status Register |
| UART_TFL | 0x0080 | UART Transmit FIFO Level Register |
| UART_RFL | 0x0084 | UART Receive FIFO Level Register |
| UART_HSK | 0x0088 | UART DMA Handshake Configuration Register |
| UART_DMA_REQ_EN | 0x008C | UART DMA Request Enable Register |
| UART_HALT | 0x00A4 | UART Halt TX Register |
| UART_DBG_DLL | 0x00B0 | UART Debug DLL Register |
| UART_DBG_DLH | 0x00B4 | UART Debug DLH Register |
| UART_FCC | 0x00F0 | UART FIFO Clock Control Register |

10.2.6 Register Description

10.2.6.1 0x0000 UART Receiver Buffer Register (Default Value: 0x0000_0000)

| Offset: 0x0000 | | | Register Name: UART_RBR |
|----------------|------------|-------------|-------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |

| Offset: 0x0000 | | | Register Name: UART_RBR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 7:0 | R | 0x0 | <p>RBR Receiver Buffer Register Data byte received on the serial input port (sin) in UART mode, or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in UART_LCR is set. If in FIFO mode and FIFOs are enabled (The UART_FCR[0] is set to 1), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register can not read before the next data character arrives, then the data already in the FIFO are preserved, but any incoming data are lost and an overrun error occurs.</p> |

10.2.6.2 0x0000 UART Transmit Holding Register (Default Value: 0x0000_0000)

| Offset: 0x0000 | | | Register Name: UART_THR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:0 | W | 0x0 | <p>THR Transmit Holding Register Data is transmitted on the serial output port (SOUT) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the UART_THR when the THRE bit (UART_LSR[5]) is set. If in FIFO mode and FIFOs are enabled (UART_FCR[0] = 1) and THRE is set, the 16 number of characters data may be written to the UART_THR before the FIFO is full. When the FIFO is full, any written data results in the written data being lost.</p> |

10.2.6.3 0x0000 UART Divisor Latch Low Register (Default Value: 0x0000_0000)

| Offset: 0x0000 | | | Register Name: UART_DLL |
|----------------|------------|-------------|-------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |

| Offset: 0x0000 | | | Register Name: UART_DLL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 7:0 | R/W | 0x0 | <p>DLL</p> <p>Divisor Latch Low</p> <p>Lower 8 bits of 16 bits, read/write, Divisor Latch Register contains the baud rate divisor for the UART. This register may only be accessed when the DLAB bit (UART_LCR[7]) is set and the UART is not busy (UART_USR[0] is 0).</p> <p>The output baud rate is equal to the serial clock (SCLK) frequency divided by sixteen times the value of the baud rate divisor, as follows: baud rate = (serial clock freq)/(16 * divisor).</p> <p>Note that when the Divisor Latch Registers (UART_DLL and UART_DLH) are set to 0, the baud clock is disabled and no serial communications occur. Also, once the DLL is set, at least 8 clock cycles of the slowest UART clock should be allowed to pass before transmitting or receiving data.</p> |

10.2.6.4 0x0004 UART Divisor Latch High Register (Default Value: 0x0000_0000)

| Offset: 0x0004 | | | Register Name: UART_DLH |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:0 | R/W | 0x0 | <p>DLH</p> <p>Divisor Latch High</p> <p>Upper 8 bits of 16 bits, read/write, Divisor Latch Register contains the baud rate divisor for the UART. This register may only be accessed when the DLAB bit (UART_LCR[7]) is set and the UART is not busy (UART_USR[0] is 0).</p> <p>The output baud rate is equal to the serial clock (SCLK) frequency divided by sixteen times the value of the baud rate divisor, as follows: baud rate = (serial clock freq)/(16 * divisor).</p> <p>Note that when the Divisor Latch Registers (UART_DLL and UART_DLH) are set to 0, the baud clock is disabled and no serial communications occur. Also, once the DLH is set, at least 8 clock cycles of the slowest UART clock should be allowed to pass before transmitting or receiving data.</p> |

10.2.6.5 0x0004 UART Interrupt Enable Register (Default Value: 0x0000_0000)

| Offset: 0x0004 | | | Register Name: UART_IER |
|----------------|------------|-------------|-------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |

| Offset: 0x0004 | | | Register Name: UART_IER |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 7 | R/W | 0x0 | <p>PTIME</p> <p>Programmable THRE Interrupt Mode Enable</p> <p>This is used to enable/disable the generation of THRE Interrupt.</p> <p>0: Disable 1: Enable</p> |
| 6:5 | / | / | / |
| 4 | R/W | 0x0 | <p>RS485_INT_EN</p> <p>RS485 Interrupt Enable</p> <p>0: Disable 1: Enable</p> |
| 3 | R/W | 0x0 | <p>EDSSI</p> <p>Enable Modem Status Interrupt</p> <p>This is used to enable/disable the generation of Modem Status Interrupt. This is the fourth highest priority interrupt.</p> <p>0: Disable 1: Enable</p> |
| 2 | R/W | 0x0 | <p>ELSI</p> <p>Enable Receiver Line Status Interrupt</p> <p>This is used to enable/disable the generation of Receiver Line Status Interrupt. This is the highest priority interrupt.</p> <p>0: Disable 1: Enable</p> |
| 1 | R/W | 0x0 | <p>ETBEI</p> <p>Enable Transmit Holding Register Empty Interrupt</p> <p>This is used to enable/disable the generation of Transmitter Holding Register Empty Interrupt. This is the third-highest priority interrupt.</p> <p>0: Disable 1: Enable</p> |
| 0 | R/W | 0x0 | <p>ERBFI</p> <p>Enable Received Data Available Interrupt</p> <p>This is used to enable/disable the generation of Received Data Available Interrupt and the Character Timeout Interrupt (if in FIFO mode and FIFOs enabled). These are the second-highest priority interrupt.</p> <p>0: Disable 1: Enable</p> |

10.2.6.6 0x0008 UART Interrupt Identity Register (Default Value: 0x0000_0001)

| Offset: 0x0008 | | | Register Name: UART_IIR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:6 | R | 0x0 | <p>FEFLAG FIFOs Enable Flag This is used to indicate whether the FIFOs are enabled or disabled. 00: Disable 11: Enable</p> |
| 5:4 | / | / | / |
| 3:0 | R | 0x1 | <p>IID Interrupt ID This indicates the highest priority pending interrupt which can be one of the following types. 0000: modem status 0001: no interrupt pending 0010: THR empty 0011: RS485 Interrupt 0100: received data available 0110: receiver line status 0111: busy detect 1100: character timeout The bit 3 indicates an interrupt can only occur when the FIFOs are enabled and used to distinguish a Character Timeout condition interrupt.</p> |

| Interrupt ID | Priority Level | Interrupt Type | Interrupt Source | Interrupt Reset |
|--------------|----------------|-------------------------|--|--|
| 0001 | - | None | None | - |
| 0110 | Highest | Receiver line status | Overrun/parity/framing errors or break interrupt | Reading the line status register |
| 0011 | Second | RS485 Interrupt | In RS485 mode, receives address data and match setting address | Writes 1 to addr flag to reset |
| 0100 | Third | Received data available | Receiver data available (non-FIFO mode or FIFOs disabled) or RCVR FIFO trigger level reached (FIFO mode and FIFOs enabled) | Reading the receiver buffer register (non-FIFO mode or FIFOs disabled) or the FIFO drops below the trigger level (FIFO mode and FIFOs enabled) |

| Interrupt ID | Priority Level | Interrupt Type | Interrupt Source | Interrupt Reset |
|--------------|----------------|--|---|--|
| 1100 | Fourth | Character timeout indication | No characters in or out of the RCVR FIFO during the last 4 character times and there is at least 1character in it during This time | Reading the receiver buffer register |
| 0010 | Fifth | Transmitter holding register empty (Program THRE mode disabled) or XMIT FIFO at or below threshold (Program THRE mode enabled) | Transmitter holding register empty (Program THRE mode disabled) or XMIT FIFO at or below threshold (Program THRE mode enabled) | Reading the IIR register (if the source of interrupt); or, writing into THR (FIFOs or THRE mode not selected or disabled) or XMIT FIFO above threshold (FIFOs and THRE mode selected and enabled). |
| 0000 | Sixth | Modem status | Clear to send or data set ready or ring indicator or data carrier detect. Note that if the autoflow control mode is enabled, a change in CTS (that is, DCTS set) does not cause an interrupt. | Reading the Modem status register |
| 0111 | Seventh | Busy detect indication | UART_16550_COMPATIBLE = NO and master has tried to write to the Line Control Register while the UART is busy (USR[0] is set to one). | Reading the UART status register |

10.2.6.7 0x0008 UART FIFO Control Register (Default Value: 0x0000_0000)

| Offset: 0x0008 | | | Register Name: UART_FCR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:6 | W | 0x0 | <p>RT RCVR Trigger</p> <p>This is used to select the trigger level in the receiver FIFO when the Received Data Available Interrupt is generated. In the autoflow control mode, it is used to determine when the rts_n signal is de-asserted. It also determines when the dma_rx_req_n signal is asserted in certain modes of operation.</p> <p>00: 1 character in the FIFO 01: FIFO $\frac{1}{4}$ full 10: FIFO $\frac{1}{2}$ full 11: FIFO-2 less than full</p> |

| Offset: 0x0008 | | | Register Name: UART_FCR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 5:4 | W | 0x0 | <p>TFT TX Empty Trigger This is used to select the empty threshold level when the THRE Interrupts are generated and the mode is active. It also determines when the dma_tx_req_n signal is asserted in certain modes of operation.</p> <p>00: FIFO empty 01: 2 characters in the FIFO 10: FIFO ¼ full 11: FIFO ½ full</p> |
| 3 | W | 0x0 | <p>DMAM DMA Mode 0: Mode 0 In this mode, when the PTE in UART_HALT is high and TX FIFO is enabled, the TX DMA request will be set when the TFL in UART_TFL is less than or equal to FIFO Trigger Level (otherwise it will be cleared). When the PTE is high and TX FIFO is disabled, the TX DMA request will be set only if the THR in UART_THR is empty. If the PTE is low, the TX DMA request will be set only if the TX FIFO (TX FIFO enabled) or THR (TX FIFO disabled) is empty. When the DMA_PTE_RX in UART_HALT is high and RX FIFO is enabled, the RX DRQ will be set only if the RFL in UART_RFL is equal to or more than FIFO Trigger Level, otherwise, it will be cleared.</p> <p>1: Mode 1 In this mode, TX FIFO should be enabled. If the PTE in UART_HALT is high, the TX DMA request will be set when the TFL in UART_TFL is less than or equal to FIFO Trigger Level; If the PTE is low, the TX DMA request will be set when TX FIFO is empty. Once the request is set, it is cleared only when TX FIFO is full. If the RFL in UART_RFL is equal to or more than FIFO Trigger Level or there is a character timeout, the RX DRQ will be set; Once the RX DRQ is set, it is cleared only when RX FIFO (RX FIFO enabled) or RBR (RX FIFO disabled) is empty.</p> |
| 2 | W | 0x0 | <p>XFIFOR XMIT FIFO Reset The bit resets the control part of the transfer FIFO and treats the FIFO as empty. This also de-asserts the DMA TX request. It is 'self-cleared'. It is not necessary to clear this bit.</p> |

| Offset: 0x0008 | | | Register Name: UART_FCR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 1 | W | 0x0 | RFIFOR RCVR FIFO Reset The bit resets the control part of the receive FIFO and treats the FIFO as empty. This also de-asserts the DMA RX request. It is 'self-cleared'. It is not necessary to clear this bit. |
| 0 | W | 0x0 | FIFOE Enable FIFOs The bit enables/disables the transmitting (XMIT) and receiving (RCVR) FIFOs. Whenever the value of this bit is changed, both the XMIT and RCVR controller part of FIFOs is reset. |

10.2.6.8 0x000C UART Line Control Register (Default Value: 0x0000_0000)

| Offset: 0x000C | | | Register Name: UART_LCR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7 | R/W | 0x0 | DLAB Divisor Latch Access Bit It is writable only when UART is not busy (UART_USR[0] is 0) and always readable. This bit is used to enable the reading and writing of the Divisor Latch register (UART_DLL and UART_DLH) to set the baud rate of the UART. This bit must be cleared after the initial baud rate setup in order to access other registers. 0: Select RX Buffer Register (UART_RBR)/TX Holding Register (UART_THR) and Interrupt Enable Register (UART_IER) 1: Select Divisor Latch LS Register (UART_DLL) and Divisor Latch MS Register (UART_DLM) |
| 6 | R/W | 0x0 | BC Break Control Bit This is used to cause a break condition to be transmitted to the receiving device. If set to 0, the serial output is forced to the spacing (logic 0) state. When not in Loopback mode, as determined by UART_MCR[4], the SOUT line is forced low until the Break bit is cleared. If SIR_MODE is enabled and active (UART_MCR[6] is set to 1), the sir_out_n line is continuously pulsed. When in Loopback mode, the break condition is internally looped back to the receiver and the sir_out_n line is forced low. |

| Offset: 0x000C | | | Register Name: UART_LCR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 5:4 | R/W | 0x0 | <p>EPS Even Parity Select It is writable only when UART is not busy (UART_USR[0] is 0). This is used to select the even and odd parity when the PEN is enabled (the UART_LCR[3] is set to 1). Setting the UART_LCR[5] is unset to reverse the LCR[4].</p> <p>00: Odd Parity 01: Even Parity 1X: Reverse LCR[4] In RS485 mode, it is the 9th bit--address bit. 11: 9th bit = 0, indicates that this is a data byte. 10: 9th bit = 1, indicates that this is an address byte.</p> <p>Note: When using this function, the PEN(UART_LCR[3]) must set to 1.</p> |
| 3 | R/W | 0x0 | <p>PEN Parity Enable It is writable only when UART is not busy (UART_USR[0] is 0) and always readable. This bit is used to enable and disable parity generation and detection in transmitted and received serial characters respectively.</p> <p>0: Parity disabled 1: Parity enabled</p> |
| 2 | R/W | 0x0 | <p>STOP Number of stop bits It is writable only when UART is not busy (UART_USR[0] is 0) and always readable. This is used to select the number of stop bits per character that the peripheral transmits and receives. If set to 0, one stop bit is transmitted in the serial data. If set to 1 and the data bits are set to 5 (UART_LCR[1:0] is 0), one and a half stop bit is transmitted. Otherwise, two stop bits are transmitted. Note that regardless of the number of stop bits selected, the receiver checks only the first stop bit.</p> <p>0: 1 stop bit 1: 1.5 stop bits when DLS (LCR[1:0]) is zero, else 2 stop bit</p> |

| Offset: 0x000C | | | Register Name: UART_LCR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 1:0 | R/W | 0x0 | <p>DLS Data Length Select 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> |

10.2.6.9 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 00: UART Mode 01: IrDA SIR Mode 10: RS485 Mode 11: Reserved</p> |
| 5 | R/W | 0x0 | <p>AFCE Auto Flow Control Enable When FIFOs are enabled and the AFCE bit is set, the AutoFlow Control is enabled. 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 Loop Back Mode 0: Normal Mode 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 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) 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> |

10.2.6.10 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> |
| 5 | R | 0x1 | <p>THRE 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. 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> |

| Offset:0x0014 | | | Register Name: UART_LSR |
|---------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 4 | R | 0x0 | <p>BI Break Interrupt This is used to indicate the detection of a break sequence on the serial input data. If in UART mode (SIR_MODE == Disabled), it is set when the serial input, sir_in, is held in a logic '0' state for longer than the sum of start time + data bits + parity + stop bits. If in infrared mode (SIR_MODE == Enabled), it is set when the serial input, sir_in, is continuously pulsed to logic '0' for longer than the sum of start time + data bits + parity + stop bits. A break condition on serial input causes one and only one character, consisting of all zeros, to be received by the UART. 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> |
| 3 | RC | 0x0 | <p>FE Framing Error 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. 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]). 0: no framing error 1: framing error Reading the UART_LSR clears the FE bit.</p> |

| Offset:0x0014 | | | Register Name: UART_LSR |
|---------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 2 | RC | 0x0 | <p>PE 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> |
| 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>Note: Not use when the RXDMA master is enabled (RXDMA_CTRI[0] is set to 1).</p> |

10.2.6.11 0x0018 UART Modem Status Register (Default Value: 0x0000_0000)

| Offset: 0x0018 | | | Register Name: UART_MSR |
|----------------|------------|-------------|-------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |

| Offset: 0x0018 | | | Register Name: UART_MSR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 7 | R | 0x0 | <p>DCD Line State of Data Carrier Detect 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. 0: dcd_n input is de-asserted (logic 1) 1: dcd_n input is asserted (logic 0)</p> |
| 6 | R | 0x0 | <p>RI Line State of Ring Indicator 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. 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 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. 0: dsr_n input is de-asserted (logic 1) 1: dsr_n input is asserted (logic 0) 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 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. 0: cts_n input is de-asserted (logic 1) 1: cts_n input is asserted (logic 0) 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 This is used to indicate that the modem control line dcd_n has changed since the last time the UART_MSR was read. 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 Reading the UART_MSR clears the DDCD bit.</p> <p>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.</p> |
| 2 | RC | 0x0 | <p>TERI Trailing Edge Ring Indicator 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. 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 Reading the UART_MSR clears the TERI bit.</p> |
| 1 | RC | 0x0 | <p>DDSR Delta Data Set Ready This is used to indicate that the modem control line dsr_n has changed since the last time the UART_MSR was read. 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 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>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.</p> |
| 0 | RC | 0x0 | <p>DCTS Delta Clear to Send This is used to indicate that the modem control line cts_n has changed since the last time the UART_MSR was read. 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 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>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.</p> |

10.2.6.12 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 | SCRATCH_REG Scratch Register This register is for programmers to use as a temporary storage space. It has no defined purpose in the UART. |

10.2.6.13 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 | RFF RX FIFO Full This is used to indicate that the RX FIFO is completely full. 0: RX FIFO not full 1: RX FIFO Full This bit is cleared when the RX FIFO is no longer full. |
| 3 | R | 0x0 | RFNE RX FIFO Not Empty This is used to indicate that the RX FIFO contains one or more entries. 0: RX FIFO is empty 1: RX FIFO is not empty This bit is cleared when the RX FIFO is empty. |
| 2 | R | 0x1 | TFE TX FIFO Empty This is used to indicate that the TX FIFO is completely empty. 0: TX FIFO is not empty 1: TX FIFO is empty This bit is cleared when the TX FIFO is no longer empty. |
| 1 | R | 0x1 | TFNF TX FIFO Not Full This is used to indicate that the TX FIFO is not full. 0: TX FIFO is full 1: TX FIFO is not full This bit is cleared when the TX FIFO is full. |
| 0 | R | 0x0 | BUSY UART Busy Bit 0: Idle or inactive 1: Busy |

10.2.6.14 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 | TFL TX FIFO Level The bit indicates the number of data entries in the TX FIFO. |

10.2.6.15 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. Note: Not use when the RXDMA master is enabled (UART_RXDMA_CTRL[0] is set to 1). |

10.2.6.16 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 |

10.2.6.17 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 |

| Offset: 0x008C | | | Register Name: UART_DMA_REQ_EN |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 0 | R/W | 0x1 | DMA RX REQ Enable 0: Disable 1: Enable |

10.2.6.18 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 | PTE The sending of TX_REQ 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. 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. |
| 6 | R/W | 0x0 | DMA_PTE_RX The Transmission of RX_DRQ In DMA1 mode, when RFL is more than or equal to the trigger value, or a receive timeout has occurred, the controller sends DRQ. 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. |
| 5 | R/W | 0x0 | SIR_RX_INVERT SIR RX Pulse Polarity Invert 0: Not invert receiver signal 1: Invert receiver signal |
| 4 | R/W | 0x0 | SIR_TX_INVERT SIR TX Pulse Polarity Invert 0: Not invert transmit pulse 1: Invert transmit pulse |
| 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>Note: If FIFOs are not enabled, the setting has no effect on operation.</p> |

10.2.6.19 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 |

10.2.6.20 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 |

10.2.6.21 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 |

10.3 Serial Peripheral Interface (SPI)

10.3.1 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.

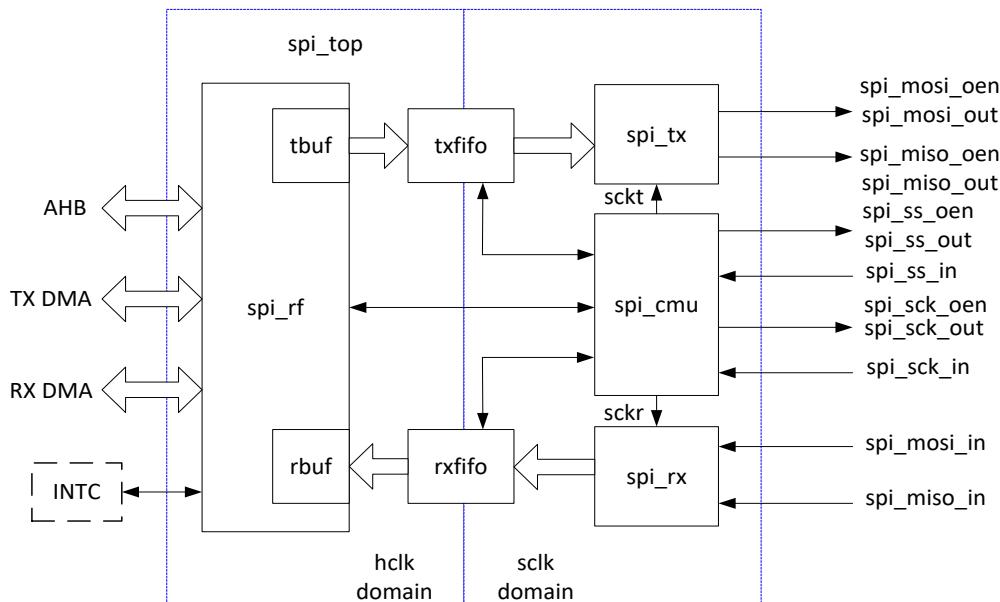
The SPI has the following features:

- Supports multiple SPI mode
 - Standard SPI (3-Wire/4-Wire SPI)
 - Dual-Output/Dual-Input SPI/ Dual I/O SPI
 - Quad-Output/Quad-Input SPI
 - Supports programmable serial data frame length: 0bit to 32bits
- Support the maximum IO rate of the mass production: 100 MHz
- Support TX/RX DMA slave interface
- Transmit and receive data FIFOs with 8-bit wide and 64-entry
- Supports data sample mode
 - mode0, mode1, mode2 and mode3
- Supports Controll singnal configutation
 - Four chip selects to support multiple peripherals
 - Polarity and phase of the Chip Select (SPI_SS) and SPI Clock (SPI_SCLK) are configurable

10.3.2 Block Diagram

The following figure shows a block diagram of the SPI.

Figure 10-17 SPI Block Diagram



SPI contains the following sub-blocks:

Table 10-9 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_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. |
| spi_rx | Responsible for implementing SPI data receive, the interface of the internal RXFIFO, and status register. |

10.3.3 Functional Description

10.3.3.1 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 10-10 SPI External Signals

| Signal | Description | Type |
|-----------|---|------|
| SPI0-CS0 | SPI0 Chip Select Signal0 (low active) | I/O |
| SPI0-CS1 | SPI0 Chip Select Signal1 (low active) | I/O |
| SPI0-CLK | SPI0 Clock Signal Provides serial interface timing | 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 | SPI0 Write Protect, Low Active Protects the memory area against all program or erase instructions. It also can be used for serial data input and output for SPI Quad Input or Quad Output mode. | I/O |
| SPI0-HOLD | SPI0 Hold Signal Pauses any serial communication with the device without deselecting or resetting it. It also can be used for serial data input and output for SPI Quad Input or Quad Output mode. | I/O |
| SPI1-CS0 | SPI1 Chip Select Signal0 (low active) | I/O |
| SPI1-CS1 | SPI1 Chip Select Signal1 (low active) | I/O |
| SPI1-CLK | SPI1 Clock Signal Provides serial interface timing. | I/O |
| SPI1-MOSI | SPI1 Master Data Out, Slave Data In | I/O |
| SPI1-MISO | SPI1 Master Data In, Slave Data Out | I/O |
| SPI1-WP | SPI1 Write Protect, Low Active Protects the memory area against all program or erase instructions. It also can be used for serial data input and output for SPI Quad Input or Quad Output mode. | I/O |
| SPI1-HOLD | SPI1 Hold Signal Pauses any serial communication with the device without resetting it. It also can be used for serial data input and output for SPI Quad Input or Quad Output mode. | I/O |
| SPI2-CS0 | SPI2 Chip Select Signal0 (low active) | I/O |
| SPI2-CS1 | SPI2 Chip Select Signal1 (low active) | I/O |
| SPI2-CLK | SPI2 Clock Signal Provides serial interface timing. | I/O |
| SPI2-MOSI | SPI2 Master Data Out, Slave Data In | I/O |
| SPI2-MISO | SPI2 Master Data In, Slave Data Out | I/O |

| Signal | Description | Type |
|----------|---|------|
| 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. The rising and falling edge is configurable. | 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 |

10.3.3.2 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 3.4 Clock Controller Unit (CCU).

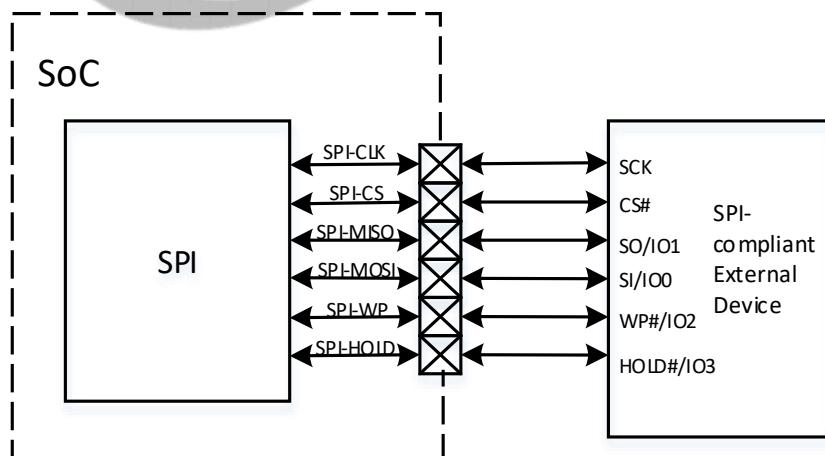
Table 10-11 SPI Clock Sources

| Clock Sources | Description |
|---------------|------------------|
| OSC24M | 24 MHz Crystal |
| PLL_PERI | Peripheral Clock |

10.3.3.3 Typical Application

The following figure shows the application block diagram when the SPI master device is connected to a slave device.

Figure 10-18 SPI Application Block Diagram



10.3.3.4 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 \(Offset: 0x0008\)](#). 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 10-12 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 |
| 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 |

The following figures describe four waveform for SPI_SCLK.

Figure 10-19 SPI Phase 0 Timing Diagram

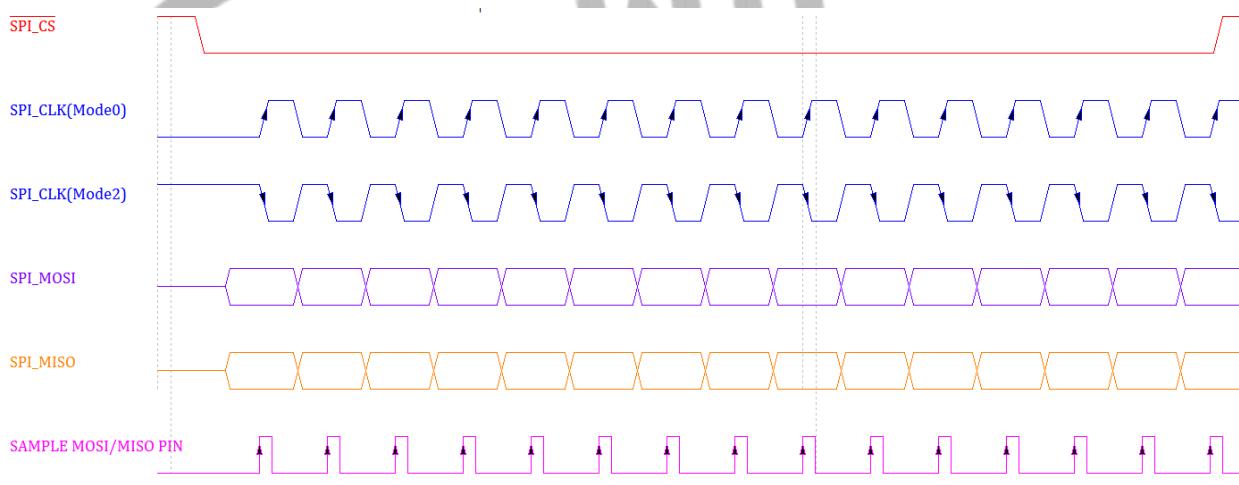
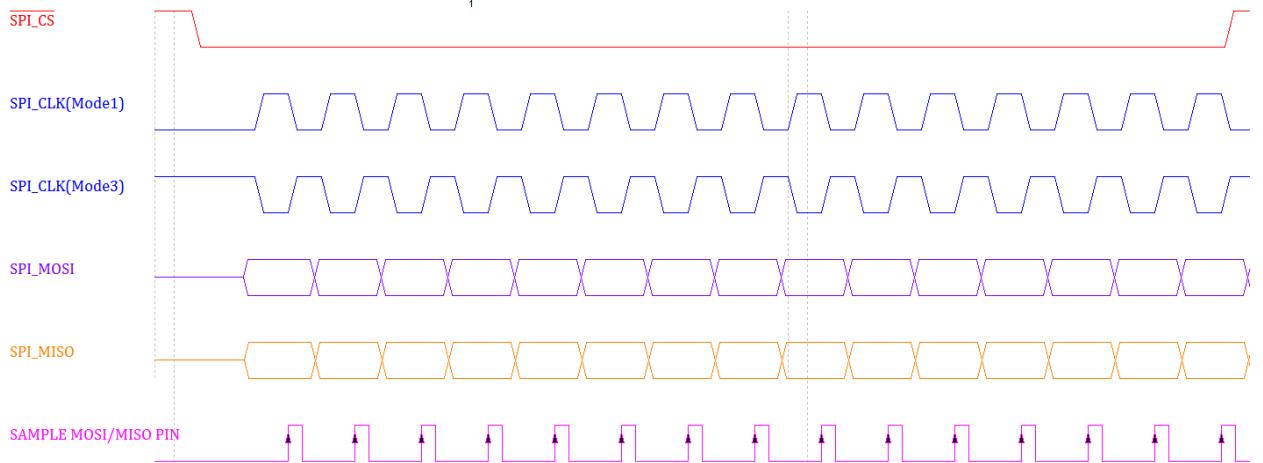
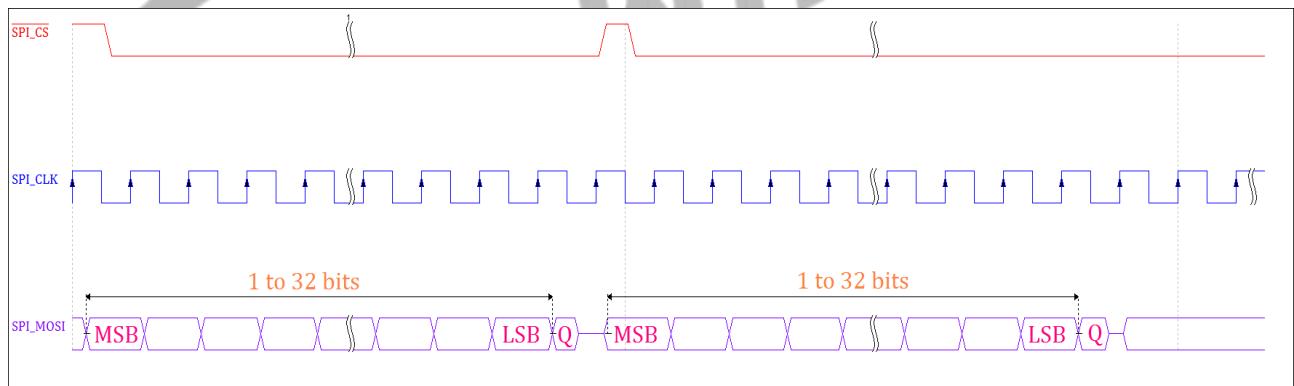


Figure 10-20 SPI Phase 1 Timing Diagram

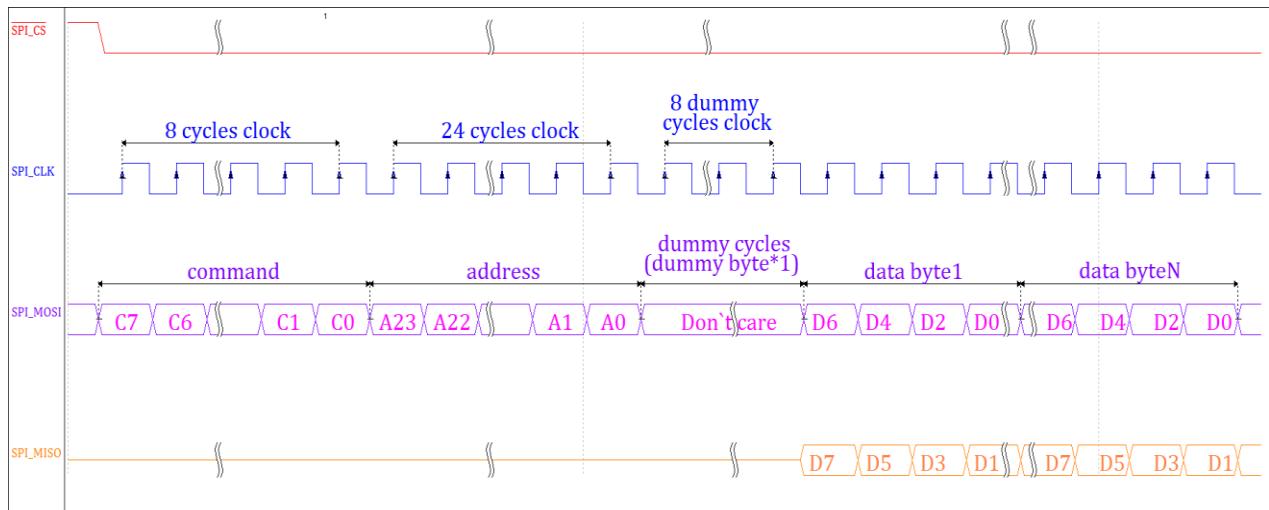
10.3.3.5 SPI 3-Wire Mode

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_BATCR\[1:0\]](#)) 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.

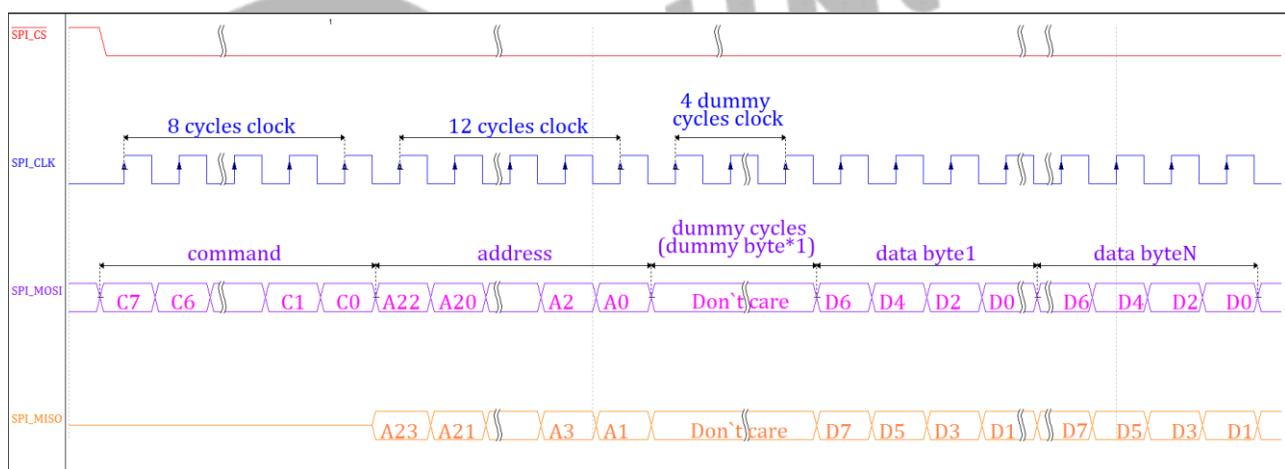
Figure 10-21 SPI 3-Wire Mode

10.3.3.6 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 \(Offset: 0x0038\)](#) [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 and the dual I/O SPI.

Figure 10-22 SPI Dual-Input/Dual-Output Mode

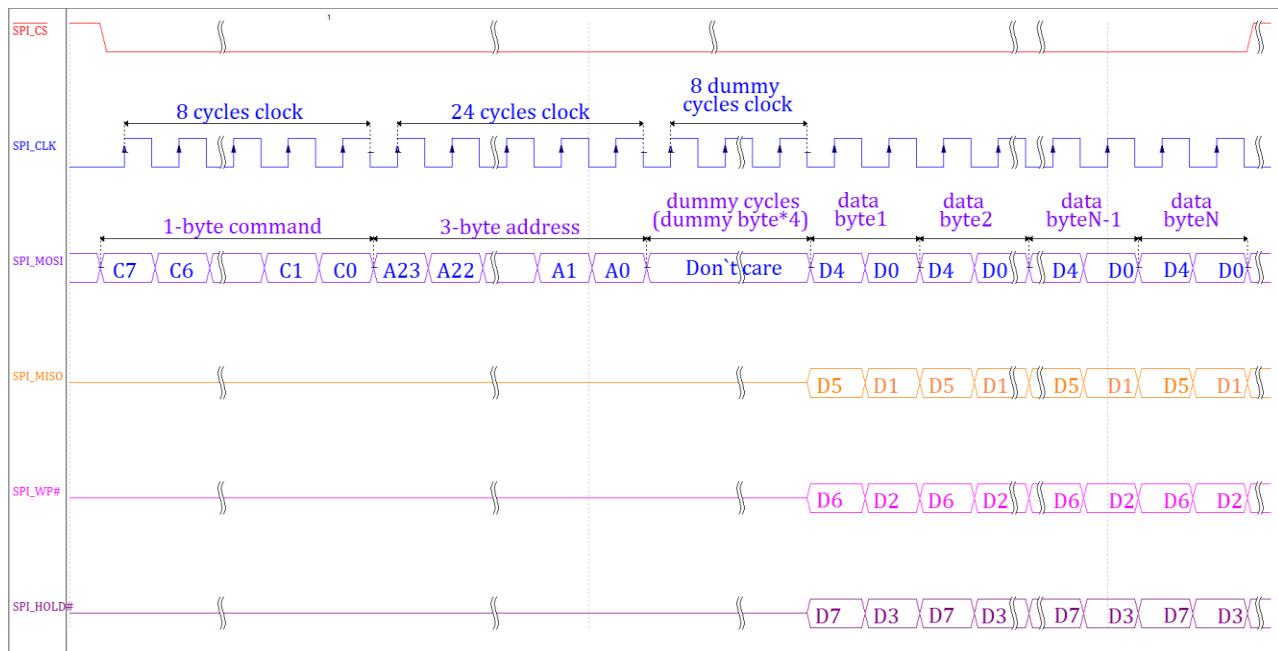
In the dual-input/dual-output SPI mode, the command, address, and the dummy bytes output in a unit of a single bit in serial mode through the SPI_MOSI line, only the data bytes are output (write) and input (read) in a unit of dual bits through the SPI_MOSI and SPI_MISO.

Figure 10-23 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.

10.3.3.7 SPI Quad-Input/Quad-Output Mode

The quad read mode (SPI x4) is selected when the Quad_EN is set in [SPI_BCC \(Offset: 0x0038\)](#) [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 10-24 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#.

10.3.3.8 SPI Error Conditions

If any error conditions occur, the hardware will set the corresponding status bits in the [SPI Interrupt Status Register \(Offset: 0x0014\)](#) and stop the transfer. For the SPI controller, the following error scenarios can happen.

- 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 \(Offset: 0x0014\)](#). 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 \(Offset: 0x0004\)](#).

- 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 \(Offset: 0x0014\)](#). 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 \(Offset: 0x0004\)](#).

- 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 \(Offset: 0x0014\)](#). 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 \(Offset: 0x0004\)](#).

- 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 \(Offset: 0x0014\)](#). 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 \(Offset: 0x0004\)](#).

10.3.4 Programming Guidelines

10.3.4.1 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.

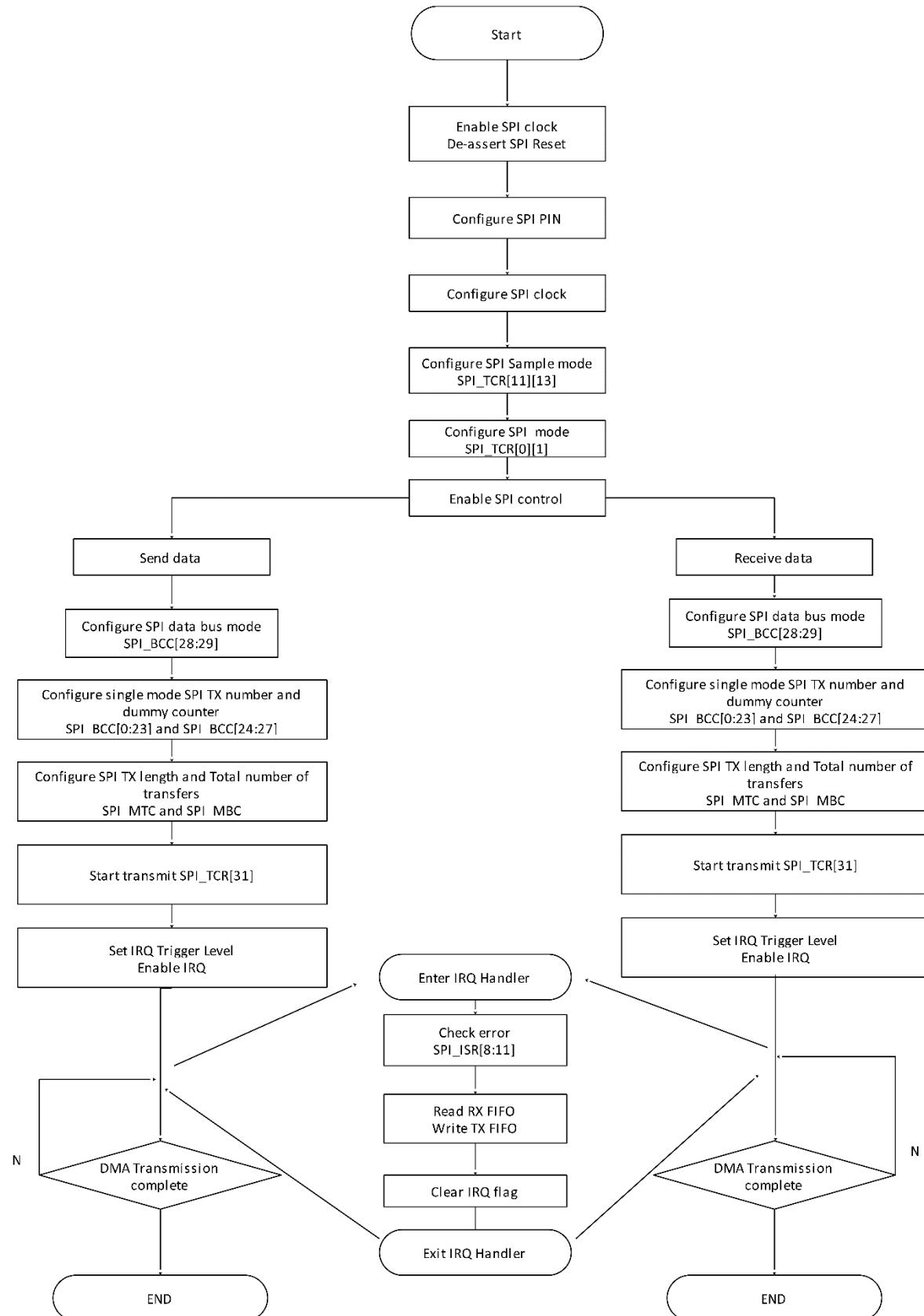
Write Data: The CPU or DMA must write data on the [SPI_TXD \(Offset: 0x0200\)](#), the data on the register are automatically moved to TX FIFO.

Read Data: To read data from RX FIFO, the CPU or DMA must access the [SPI_RXD \(Offset: 0x0300\)](#) and the data are automatically sent to the [SPI_RXD \(Offset: 0x0300\)](#).

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

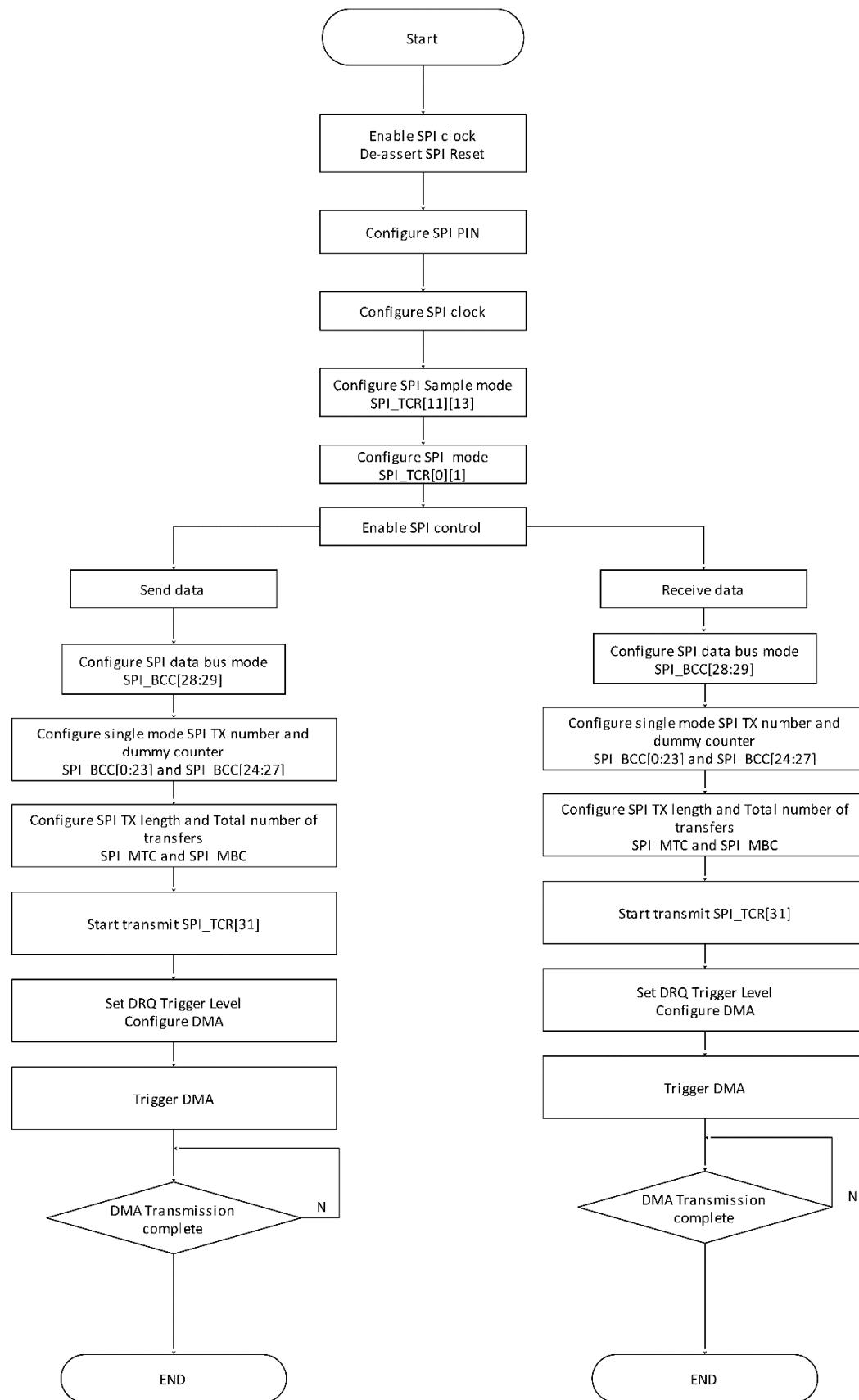
CPU Mode

Figure 10-25 SPI Write/Read Data in CPU Mode



DMA Mode

Figure 10-26 SPI Write/Read Data in DMA Mode



10.3.4.2 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:

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.
2. Configure a proper clock for SPI. The calibration delay chain is based on the clock for SPI from CCU.
3. Set proper initial delay value. Write 0xA0 to the [SPI Sample Delay Control Register \(Offset: 0x0028\)](#) to set initial delay value 0x20 to delay chain. Then write 0x0 to the [SPI Sample Delay Control Register \(Offset: 0x0028\)](#) to clear this value.
4. Write 0x8000 to the [SPI Sample Delay Control Register \(Offset: 0x0028\)](#) to start to calibrate the delay chain.
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 \(Offset: 0x0028\)](#). 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.
6. Calculate the delay time of one delay cell according to the cycle of the SPI clock and the result of calibration.

10.3.5 Register List

| Module Name | Base Address | Comments |
|-------------|--------------|----------------|
| SPI0 | 0x0402 5000 | SPI controller |
| SPI1 | 0x0402 6000 | SPI controller |
| SPI2 | 0x0402 7000 | SPI controller |
| SPI3 | 0x0402 8000 | SPI controller |

| 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 |
| SPI_TXD | 0x0200 | SPI TX Data Register |
| SPI_RXD | 0x0300 | SPI RX Data Register |

10.3.6 Register Description

10.3.6.1 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 | <p>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.</p> |
| 30:8 | / | / | / |
| 7 | R/W | 0x1 | <p>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.</p> |
| 6:3 | / | / | / |
| 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> |

| Offset:0x0004 | | | Register Name: SPI_GCR |
|---------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 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> |

10.3.6.2 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> |
| 14 | R/W | 0x0 | <p>SDDM</p> <p>Sending Data Delay Mode</p> <p>0: Normal sending</p> <p>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.</p> <p>Cannot be written when XCH=1.</p> |

| Offset: 0x0008 | | | Register Name: SPI_TCR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 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> |
| 12 | R/W | 0x0 | <p>FBS First Transmit Bit Select 0: MSB first 1: LSB first Cannot be written when XCH=1.</p> |
| 11 | R/W | 0x0 | <p>SDC Master Sample Data Control 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 Select rapid mode for high speed write. 0: Normal write mode 1: Rapid write mode Cannot be written when XCH=1.</p> |
| 9 | R/W | 0x0 | <p>DDB Dummy Burst Type 0: The bit value of dummy SPI burst is zero 1: The bit value of dummy SPI burst is one Cannot be written when XCH=1.</p> |
| 8 | R/W | 0x0 | <p>DHB Discard Hash Burst In master mode, it controls whether discarding unused SPI bursts 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. 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</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> |
| 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</p> <p>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> |
| 1 | R/W | 0x1 | <p>CPOL</p> <p>SPI Clock 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 |
| 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. |

10.3.6.3 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 | 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 |
| 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 |

| Offset: 0x0010 | | | Register Name: SPI_IER |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 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 |

10.3.6.4 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 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</p> |
| 10 | R/W1C | 0x0 | <p>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</p> |
| 9 | R/W1C | 0x0 | <p>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</p> |
| 8 | R/W1C | 0x0 | <p>RX_OVF RXFIFO Overflow When set, this bit indicates that RXFIFO has overflowed. Writing 1 to this bit clears it. 0: RXFIFO is not overflowed 1: RXFIFO is overflowed</p> |
| 7 | / | / | / |
| 6 | R/W1C | 0x0 | <p>TX_FULL TXFIFO Full This bit is set when the TXFIFO is full. Writing 1 to this bit clears it. 0: TXFIFO is not Full 1: TXFIFO is Full</p> |
| 5 | R/W1C | 0x1 | <p>TX_EMP TXFIFO Empty This bit is set when the TXFIFO is empty. Writing 1 to this bit clears it. 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 0: TX_WL > TX_TRIG_LEVEL 1: TX_WL <= TX_TRIG_LEVEL This bit will be immediately set to 1 if TX_WL <= 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 This bit is set when the RXFIFO is full. Writing 1 to this bit clears it. 0: Not Full 1: Full</p> |
| 1 | R/W1C | 0x1 | <p>RX_EMP RXFIFO Empty This bit is set when the RXFIFO is empty. Writing 1 to this bit clears it. 0: Not empty 1: empty</p> |
| 0 | R/W1C | 0x0 | <p>RX_RDY RXFIFO Ready 0: RX_WL < RX_TRIG_LEVEL 1: RX_WL >= RX_TRIG_LEVEL This bit is will be immediately set to 1 if RX_WL >= RX_TRIG_LEVEL. Writing “1” to this bit clears it. The RX_WL is the water level of RXFIFO.</p> |

10.3.6.5 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 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 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 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 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 0: Disable 1: Enable</p> |
| 7:0 | R/W | 0x1 | <p>RX_TRIG_LEVEL RX FIFO Ready Request Trigger Level</p> |

10.3.6.6 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 |

10.3.6.7 0x0020 SPI Wait Clock Register (Default Value: 0x0000_0000)

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

| Offset: 0x0020 | | | Register Name: SPI_WCR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 19:16 | R/W | 0x0 | <p>SWC 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>O: No wait states inserted n: n SPI_SCLK wait states inserted Cannot be written when XCH=1.</p> |
| 15:0 | R/W | 0x0 | <p>WCC 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>O: No wait states inserted n: n SPI_SCLK wait states inserted Cannot be written when XCH=1.</p> |

10.3.6.8 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 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 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> |
| 13:8 | R | 0x20 | <p>SAMP_DL 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> |

| Offset: 0x0028 | | | Register Name: SPI_SAMP_DL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 7 | R/W | 0x0 | SAMP_DL_SW_EN Sample Delay Software Enable When set, it indicates that enable sample delay specified at SAMP_DL_SW. Cannot be written when XCH=1. |
| 6 | / | / | / |
| 5:0 | R/W | 0x0 | SAMP_DL_SW Sample Delay Software The relative delay between the clock line and command line, data lines. 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. Cannot be written when XCH=1. |

10.3.6.9 0x0030 SPI Master Burst Counter Register (Default Value: 0x0000_0000)

| Offset: 0x0030 | | | Register Name: SPI_MBC |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:24 | / | / | / |
| 23:0 | R/W | 0x0 | MBC Master Burst Counter In master mode, this field specifies the total burst number. The total transfer data include the TXD, RXD, and dummy burst. 0: 0 burst 1: 1 burst ... N: N bursts Cannot be written when XCH=1. |

10.3.6.10 0x0034 SPI Master Transmit Counter Register (Default Value: 0x0000_0000)

| Offset: 0x0034 | | | Register Name: SPI_MTC |
|----------------|------------|-------------|------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:24 | / | / | / |

| Offset: 0x0034 | | | Register Name: SPI_MTC |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 23:0 | R/W | 0x0 | <p>MWTC Master Write Transmit Counter 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> |

10.3.6.11 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 | / | / | / |
| 29 | R/W | 0x0 | <p>Quad_EN Quad Mode Enable The quad mode includes Quad-Input and Quad-Output. 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 It is only valid when Quad_Mode_EN=0. 0: RX uses the single-bit mode 1: RX uses the dual-bit mode</p> <p>Cannot be written when XCH=1.</p> |
| 27:24 | R/W | 0x0 | <p>DBC Master Dummy Burst Counter 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> |

| Offset: 0x0038 | | | Register Name: SPI_BCC |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 23:0 | R/W | 0x0 | <p>STC Master Single Mode Transmit Counter 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> |

10.3.6.12 0x0040 SPI Bit-Aligned Transfer Configure Register (Default Value: 0x0000_00A0)

| Offset: 0x0040 | | | Register Name: SPI_BATCR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31 | R/WAC | 0x0 | <p>TCE Transfer Control Enable 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 1: Initiates transfer 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 Master Sample Standard 0: Delay Sample Mode 1: Standard Sample Mode In Standard Sample Mode, the SPI master samples the data at the standard rising edge of SCLK for each SPI mode; 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 | / | / | / |

| Offset: 0x0040 | | | Register Name: SPI_BATCR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 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 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 1: Enable</p> <p>It is only valid when Work Mode Select==0x10/0x11.</p> |
| 23:22 | / | / | / |
| 21:16 | R/W | 0x00 | <p>Configure the length of serial data frame (burst) of RX</p> <p>000000: 0 bit 000001: 1 bit ... 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 (bit31) is 1.</p> |
| 15:14 | / | / | / |
| 13:8 | R/W | 0x00 | <p>TX_FRM_LEN</p> <p>Configure the length of serial data frame (burst) of TX</p> <p>000000: 0 bit 000001: 1 bit ... 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 (bit31) is 1.</p> |
| 7 | R/W | 0x1 | <p>SS_LEVEL</p> <p>When control the SS signal manually, 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>It is only valid when Work Mode Select==0x10/0x11, and only work in Mode0, cannot be written when TCE (bit31) is 1.</p> |

| Offset: 0x0040 | | | Register Name: SPI_BATCR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 6 | R/W | 0x0 | <p>SS_OWNER 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.</p> |
| 5 | R/W | 0x1 | <p>SPOL SPI Chip Select Signal Polarity Control 0: Active high polarity (0 = Idle) 1: Active low polarity (1 = Idle) 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 SPI Chip Select Select one of four external SPI Master/Slave Devices 00: SPI_SS0 will be asserted 01: SPI_SS1 will be asserted 10: SPI_SS2 will be asserted 11: SPI_SS3 will be asserted 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 00: Data frame is byte aligned in standard SPI, dual-output/dual input SPI, dual IO SPI, and quad-output/quad-input SPI 01: Reserved 10: Data frame is bit aligned in 3-wire SPI 11: Data frame is bit aligned in standard SPI</p> |

10.3.6.13 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 Clock Divide Rate (Master Mode Only) The SPI_SCLK is determined according to the following equation: $SPI_CLK = Source_CLK / (2 * (CDR_N + 1))$. This register is only valid when Work Mode Select==0x10/0x11.</p> |

10.3.6.14 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 The Value of the Transmit Bits This register is used to store the value of the transmitted serial data frame. In the process of transmission, the LSB is transmitted first. This register is only valid when Work Mode Select==0x10/0x11.</p> |

10.3.6.15 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 The Value of the Receive Bits This register is used to store the value of the received serial data frame. In the process of transmission, the LSB is transmitted first. This register is only valid when Work Mode Select==0x10/0x11.</p> |

10.3.6.16 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 SPI NDMA Active Mode 00: dma_active is low 01: dma_active is high 10: dma_active is controlled by dma_request (DRQ) 11: dma_active is controlled by controller</p> |
| 5 | R/W | 0x1 | <p>SPI_ACK_M SPI NDMA Acknowledge Mode 0: active fall do not care ack 1: active fall must after detect ack is high</p> |
| 4:0 | R/W | 0x05 | <p>SPI_DMA_WAIT The counts of hold cycles from DMA last signal high to dma_active high</p> |

10.3.6.17 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>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.</p> |

10.3.6.18 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>Note: 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> |

10.4 USB2.0 DRD

10.4.1 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:

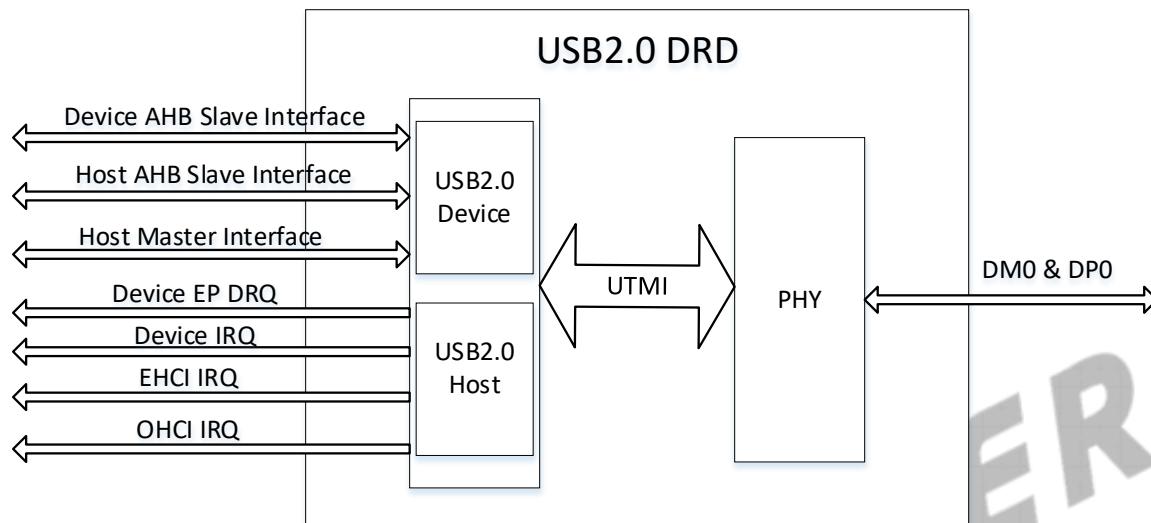
- Complies with USB2.0 Specification
- Supports USB Host function
 - Compatible with Enhanced Host Controller Interface (EHCI) Specification, Version 1.0
 - Compatible with Open Host Controller Interface (OHCI) Specification, Version 1.0a
 - Supports High-Speed (HS, 480 Mbit/s), Full-Speed (FS, 12 Mbit/s), 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

- Supports USB standby

10.4.2 Block Diagram

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

Figure 10-27 USB2.0 DRD Controller Block Diagram



10.4.3 Functional Description

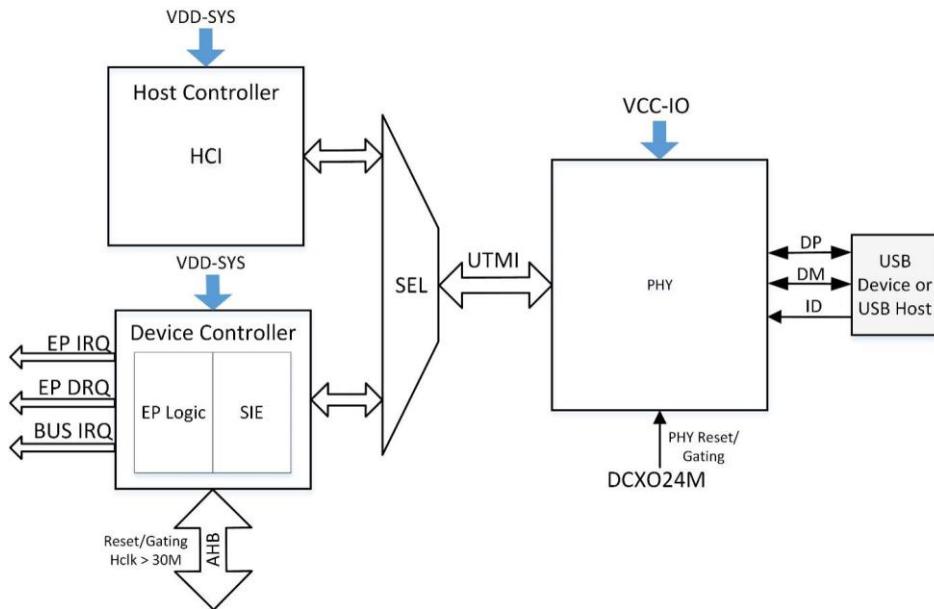
10.4.3.1 External Signals

Table 10-13 USB2.0 DRD External Signals

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

10.4.3.2 Controller and PHY Connection Diagram

Figure 10-28USB2.0 DRD Controller and PHY Connection Diagram



10.4.4 USB_DRD_DEVICE Register List

| Module Name | Base Address | Comments |
|----------------|--------------|-----------------------|
| USB_DRD_DEVICE | 0x04100000 | USB Device Controller |

| Register Name | Offset | Description |
|---------------|-------------------------------|--|
| USB_EPFIFOon | 0x0000+N*4 (N=0,1,2,3,4,5) | USB FIFO Entry for Endpoint N |
| USB_GCS | 0x0040 | USB Global Control and Status Register |
| USB_EPINTF | 0x0044 | USB Endpoint Interrupt Flag Register |
| USB_EPINTE | 0x0048 | USB Endpoint Interrupt Enable Register |
| USB_BUSINTF | 0x004C | USB Bus Interrupt Flag Register |
| USB_BUSINTE | 0x0050 | USB Bus Interrupt Enable Register |
| USB_FNUM | 0x0054 | USB Frame Number Register |
| USB_TESTC | 0x007C | USB Test Control Register |
| USB_CSR0 | 0x0080 | USB EP0 Control and Status Register |
| USB_TXCSR | 0x0080 | USB EP1~5 Tx Control and Status Register |
| USB_RXCSR | 0x0084 | USB EP1~5 Rx Control and Status Register |
| USB_COUNTO | 0x0088 | USB EP0 Rx Counter Register |
| USB_RXCOUNT | 0x0088 | USB EP1~5 Rx Counter Register |
| USB_ATTR0 | 0x008C | USB EP0 Attribute Register |
| USB_EPATTR | 0x008C | USB EP1~5 Attribute Register |
| USB_TXFIFO | 0x0090 | USB EP1~5 Tx FIFO Setting Register |
| USB_RXFIFO | 0x0094 | USB EP1~5 Rx FIFO Setting Register |

| Register Name | Offset | Description |
|---------------------|--------------------------|---|
| USB_FADDR | 0x0098 | USB Function Address Register |
| USB_ISCR | 0x0400 | USB Interface Status and Control Register |
| USB_PHY_CTL | 0x0404 | USB PHY Control Register |
| USB_PHY_TEST | 0x0414 | USB PHY Test Register |
| USB_PHY_TUNE | 0x0418 | USB PHY Tune Register |
| USB_PHY_SEL | 0x0420 | USB PHY Select Register |
| USB_PHY_STA | 0x0424 | USB PHY Status Register |
| USB_DMA_INTE | 0x0500 | USB DMA Interrupt Enable Register |
| USB_DMA_INTS | 0x0504 | USB DMA Interrupt Status Register |
| USB_DMA_CHAN_CFG | 0x0540+N*0x1 0(N=0~7) | USB DMA Channel Configuration Register |
| USB_DMA_SDRAM_ADD | 0x0544+N*0x1 0(N=0~7) | USB DMA SDRAM Start Address Register |
| USB_DMA_BC | 0x0548+N*0x1 0(N=0~7) | USB DMA Byte Counter Register |
| USB_DMA_RESIDUAL_BC | 0x0548+N*0x1 0(N=0~7) | USB DMA RESIDUAL Byte Counter Register |

10.4.5 USB_DRD_DEVICE Register Description

10.4.5.1 0x0000+N*0x04(N=0~5) USB FIFO Entry for Endpoint N (Default Value:0x0000_0000)

| Offset: 0x0000+N*0x04(N=0~5) | | | Register Name: USB_EPFIFO |
|------------------------------|------------|-------------|--------------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R/W | UDF | EPnFIFO FIFO Entry for Endpoint n |

10.4.5.2 0x0040 USB Global Control and Status Register (Default Value:0x0000_0020)

| Offset: 0x0040 | | | Register Name: USB_GCS |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31 | R/W | 0 | TX_EDMA 1'b0: DMA_REQ signal for all IN Endpoints will be de-asserted when MAXP bytes have been written to an endpoint. This is late mode. 1'b1: DMA_REQ signal for all IN Endpoints will be de-asserted when MAXP-8 bytes have been written to an endpoint. This is early mode. |

| Offset: 0x0040 | | | Register Name: USB_GCS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 30 | R/W | 0 | <p>RX_EDMA</p> <p>1'b0: DMA_REQ signal for all OUT Endpoints will be de-asserted when MAXP bytes have been read to an endpoint. This is late mode.</p> <p>1'b1: DMA_REQ signal for all OUT Endpoints will be de-asserted when MAXP-8 bytes have been read to an endpoint. This is early mode.</p> |
| 29 | / | / | / |
| 28:25 | R/W | 0 | <p>BUS_DRQ_SEL</p> <p>USB DMA Request Signal Source Select</p> <p>4'b0000: Select TX Endpoint 1 DRQ</p> <p>4'b0001: Select RX Endpoint 1 DRQ</p> <p>4'b0010: Select TX Endpoint 2 DRQ</p> <p>4'b0011: Select RX Endpoint 2 DRQ</p> <p>4'b0100: Select TX Endpoint 3 DRQ</p> <p>4'b0101: Select RX Endpoint 3 DRQ</p> <p>4'b0110: Select TX Endpoint 4 DRQ</p> <p>4'b0111: Select RX Endpoint 4 DRQ</p> |
| 24 | R/W | 0 | <p>FIFO_BUS_SEL</p> <p>0: CPU bus for FIFO Access,</p> <p>1: DMA bus for FIFO operation.</p> |
| 23:20 | / | / | / |
| 19:16 | R/W | 0x0 | <p>EPIND</p> <p>Endpoint Index</p> <p>Index is a 4-bit register that determines which endpoint control/status registers are accessed. Before accessing an endpoint's control/status registers at 0x0080~0x00BF, the endpoint number should be written to the Index register to ensure that correct control/status registers in the memory map.</p> <p>Note: The valid value for Index register is 0-4.</p> |
| 15 | R | 0 | <p>BDev</p> <p>B-Device</p> <p>0 => 'A' device;</p> <p>1 => 'B' device;</p> <p>Only valid while a session is in progress.</p> <p>Note: If the core is in Force_Host mode (i.e. a session has been started with USB_TMCTL.7=1), this bit will indicate the state of the HOSTDISCON input signal from the PHY.</p> |
| 14:13 | / | / | / |

| Offset: 0x0040 | | | Register Name: USB_GCS | | | | | | | | | | | | | | | |
|----------------|------------|--------------------------------|---|----|----|---------|---|---|------------------|---|---|--------------------------------|---|---|-------------------------------|---|---|-----------------|
| Bit | Read/Write | Default/Hex | Description | | | | | | | | | | | | | | | |
| 12:11 | R | 0x0 | <p>VBus</p> <p>These bits encode the current VBus level as follows:</p> <table border="1"> <thead> <tr> <th>D4</th><th>D3</th><th>Meaning</th></tr> </thead> <tbody> <tr> <td>0</td><td>0</td><td>Below SessionEnd</td></tr> <tr> <td>0</td><td>1</td><td>Above SessionEnd, below AValid</td></tr> <tr> <td>1</td><td>0</td><td>Above AValid, below VBusValid</td></tr> <tr> <td>1</td><td>1</td><td>Above VBusValid</td></tr> </tbody> </table> | D4 | D3 | Meaning | 0 | 0 | Below SessionEnd | 0 | 1 | Above SessionEnd, below AValid | 1 | 0 | Above AValid, below VBusValid | 1 | 1 | Above VBusValid |
| D4 | D3 | Meaning | | | | | | | | | | | | | | | | |
| 0 | 0 | Below SessionEnd | | | | | | | | | | | | | | | | |
| 0 | 1 | Above SessionEnd, below AValid | | | | | | | | | | | | | | | | |
| 1 | 0 | Above AValid, below VBusValid | | | | | | | | | | | | | | | | |
| 1 | 1 | Above VBusValid | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| 10 | R | 0 | <p>HostMode</p> <p>Host Mode</p> <p>This bit is set when the USB/DRD is acting as a Host.</p> | | | | | | | | | | | | | | | |
| 9 | / | / | / | | | | | | | | | | | | | | | |
| 8 | R/W | 0 | <p>Session</p> <p><i>When operating as an 'A' device</i>, this bit is set or cleared by the CPU to start or end a session.</p> <p><i>When operating as an 'B' device</i>, this bit is set/cleared by the USB/DRD when a session starts/ends. It is also set by the CPU to initiate the Session Request Protocol.</p> <p>When the USB/DRD is in Suspend mode, the bit may be cleared by the CPU to perform a software disconnect.</p> <p>Note: Clearing this bit when the core is not suspending will result in undefined behavior.</p> | | | | | | | | | | | | | | | |
| 7 | R/W | 0 | <p>IsoUpdateEn</p> <p>Isochronous Update Enable</p> <p>When set by the CPU, the USB/DRD will wait for an SOF token from the Tx packet ready before sending the packet. If an IN token is received before an SOF token, then a zero length data packet will be send.</p> <p>Note: This bit only affects endpoints performing Isochronous transfer.</p> | | | | | | | | | | | | | | | |
| 6 | R/W | 0 | <p>SoftConn</p> <p>Soft Connect</p> <p>The USB D+/D- line is enabled when this bit is set by CPU and tri-stated when this bit is cleared by CPU.</p> <p>Note: Only valid in Peripheral Mode (but not means 'B' Device).</p> | | | | | | | | | | | | | | | |
| 5 | R/W | 1 | <p>HSEN</p> <p>High-speed Mode Enable</p> <p>When set by CPU , the USB/DRD will negotiate for High-speed mode when the device is reset by host. If not set, the device will only operate in Full-speed mode.</p> | | | | | | | | | | | | | | | |

| Offset: 0x0040 | | | Register Name: USB_GCS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 4 | R | 0 | HSFLAG High-speed Mode Flag When set, this read-only bit indicates High-speed mode successfully negotiated during USB reset. And this bit becomes valid when USB Reset completes (as indicated by USB reset interrupt). |
| 3 | R | 0 | Reset This bit is set when Reset Signaling is present on the bus. |
| 2 | R/W | 0 | Resume Set by the CPU to generate Resume signaling when the function is in Suspend mode. The CPU should clear this bit after 10 ms (a maximum of 15 ms) to end Resume signaling. |
| 1 | R | 0 | SuspendM Suspend Mode This bit is set on entry into Suspend mode. |
| 0 | R/W | 0 | SuspendMEn Enable SuspendM Set by the CPU to enable the SUSPENDM output of UTMI+ bus. |

10.4.5.3 0x0044 USB Endpoint Interrupt Flag Register (Default Value:0x0000_0000)

| Offset: 0x0044 | | | Register Name: USB_EPINTF |
|----------------|------------|-------------|---------------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:21 | / | / | / |
| 20 | R | 0 | EP4Rx Rx Endpoint 4 interrupt flag |
| 19 | R | 0 | EP3Rx Rx Endpoint 3 interrupt flag |
| 18 | R | 0 | EP2Rx Rx Endpoint 2 interrupt flag |
| 17 | R | 0 | EP1Rx Rx Endpoint 1 interrupt flag |
| 16:5 | / | / | / |
| 4 | R | 0 | EP4Tx Tx Endpoint 4 interrupt flag |
| 3 | R | 0 | EP3Tx Tx Endpoint 3 interrupt flag |
| 2 | R | 0 | EP2Tx Tx Endpoint 2 interrupt flag |
| 1 | R | 0 | EP1Tx Tx Endpoint 1 interrupt flag |

| Offset: 0x0044 | | | Register Name: USB_EPINTF |
|----------------|------------|-------------|----------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 0 | R | 0 | EPO Endpoint 0 interrupt flag |

10.4.5.4 0x0048 USB Endpoint Interrupt Enable Register (Default Value:0x0000_0000)

| Offset: 0x0048 | | | Register Name: USB_EPINTE |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:21 | / | / | / |
| 20 | R/W | 0 | EP4Rx Rx Endpoint 4 interrupt enable |
| 19 | R/W | 0 | EP3Rx Rx Endpoint 3 interrupt enable |
| 18 | R/W | 0 | EP2Rx Rx Endpoint 2 interrupt enable |
| 17 | R/W | 0 | EP1Rx Rx Endpoint 1 interrupt enable |
| 16:5 | / | / | / |
| 4 | R/W | 0 | EP4Tx Tx Endpoint 4 interrupt enable |
| 3 | R/W | 0 | EP3Tx Tx Endpoint 3 interrupt enable |
| 2 | R/W | 0 | EP2Tx Tx Endpoint 2 interrupt enable |
| 1 | R/W | 0 | EP1Tx Tx Endpoint 1 interrupt enable |
| 0 | R/W | 0 | EPO Endpoint 0 interrupt enable |

10.4.5.5 0x004C USB Bus Interrupt Flag Register (Default Value:0x0000_0000)

| Offset: 0x004C | | | Register Name: USB_BUSINTF |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7 | R | 0 | VBusError Set when VBus drops below the VBus Valid threshold during a session. Note: Only valid when USB/DRD is 'A' device. |
| 6 | R | 0 | SessionRequest Set when Session Request signaling has been detected. Note: Only valid when USB/DRD is 'A' device. |

| Offset: 0x004C | | | Register Name: USB_BUSINTF |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 5 | R | 0 | <p>Disconnect Set in Host mode when a device disconnect is detected. Set in Peripheral mode when a session ends.</p> <p>Note: Valid at all transaction speeds.</p> |
| 4 | R | 0 | <p>Connect Set in host mode when a device connection is detected.</p> <p>Note: Only valid in Host mode. Valid at all transaction speeds.</p> |
| 3 | R | 0 | <p>SOF Set when a new frame starts.</p> |
| 2 | R | 0 | <p>ResetBabble Reset Set in Peripheral mode when Reset signaling is detected on the bus. Babble Set in Host mode when babble is detected.</p> <p>Note: Only active after first SOF has been sent.</p> |
| 1 | R | 0 | <p>Resume Set when Resume signaling is detected on the bus while the USB/DRD is in Suspend mode.</p> |
| 0 | R | 0 | <p>Suspend Set when Suspend signaling is detected on the bus.</p> <p>Note: Only valid in Peripheral mode.</p> |

10.4.5.6 0x0050 USB Bus Interrupt Enable Register (Default Value:0x0000_0000)

| Offset: 0x0050 | | | Register Name: USB_BUSINTE |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7 | R/W | 0 | VBusError VBusError interrupt enable |
| 6 | R/W | 0 | Session Request Session Request interrupt enable |
| 5 | R/W | 0 | Disconnect Disconnect interrupt enable |
| 4 | R/W | 0 | Connect Connect interrupt enable |
| 3 | R/W | 0 | SOF SOF interrupt enable |

| Offset: 0x0050 | | | Register Name: USB_BUSINTE |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 2 | R/W | 0 | ResetBabble Reset Reset interrupt enable Babble Babble interrupt enable |
| 1 | R/W | 0 | Resume Resume interrupt enable |
| 0 | R/W | 0 | Suspend Suspend interrupt enable |

10.4.5.7 0x0054 USB Frame Number Register (Default Value:0x0000_0000)

| Offset: 0x0054 | | | Register Name: USB_FNUM |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:11 | / | / | / |
| 10:0 | R | 0x000 | FRNUM Frame Number Hold the last received frame number. |

10.4.5.8 0x007C USB Test Control Register (Default Value:0x0000_0000)

| Offset: 0x007C | | | Register Name: USB_TESTC |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:24 | / | 0 | / |
| 23:16 | R | / | FSM USB Operation Finite State Machine for Debug |
| 15:11 | / | / | / |
| 10 | R/W | 0 | EXTRXACT Extend Rx Active Signal for safe |
| 9 | R/W | 0 | RESUME_SEO Resume from SEO Enable |
| 8 | R/W | 0 | TM1 Test Mode Enable for Simulation. |

| Offset: 0x007C | | | Register Name: USB_TESTC | | | | | | | | | | | | | | | |
|----------------|------------|-----------------|---|----------|----------|-----------------|---|---|-----------|---|---|------------|---|---|------------|---|---|-----------|
| Bit | Read/Write | Default/Hex | Description | | | | | | | | | | | | | | | |
| 7 | R/W | 0 | <p>Force_Host</p> <p>The CPU sets this bit to instruct the core to enter Host mode when the Session bit (<i>Bit 0 of USB_DEVCTL</i>) is set, regardless of whether it is connected to any peripheral. The state of the CID input, HostDisconnect and Linestate signals are ignored. The core will then remain in Host mode until the Session bit is cleared, even if a device is disconnected, and if the Force_Host bit remains set, will re-enter Host mode the next time the Session bit is set.</p> <p>While in this mode, the status if the HOSTDISCON signal from the PHY may be read from bit 7 of the USB_DEVCTL (<i>in 0x0060</i>) register.</p> <p>The operating speed is determined from the Force_HS and Force_FS bits as follows:</p> <table border="1"> <tr> <th>Force_HS</th><th>Force_FS</th><th>Operating Speed</th></tr> <tr> <td>0</td><td>0</td><td>Low Speed</td></tr> <tr> <td>0</td><td>1</td><td>Full Speed</td></tr> <tr> <td>1</td><td>0</td><td>High Speed</td></tr> <tr> <td>1</td><td>1</td><td>Undefined</td></tr> </table> | Force_HS | Force_FS | Operating Speed | 0 | 0 | Low Speed | 0 | 1 | Full Speed | 1 | 0 | High Speed | 1 | 1 | Undefined |
| Force_HS | Force_FS | Operating Speed | | | | | | | | | | | | | | | | |
| 0 | 0 | Low Speed | | | | | | | | | | | | | | | | |
| 0 | 1 | Full Speed | | | | | | | | | | | | | | | | |
| 1 | 0 | High Speed | | | | | | | | | | | | | | | | |
| 1 | 1 | Undefined | | | | | | | | | | | | | | | | |
| 6 | W | 0 | <p>FIFO_Access</p> <p>The CPU sets this bit to transfer the packet in the Endpoint 0 Tx FIFO to the Endpoint 0 Rx FIFO. It is cleared automatically.</p> <p>Note: Writing '0' to this bit will be ignored.</p> | | | | | | | | | | | | | | | |
| 5 | R/W | 0 | <p>Force_FS</p> <p>The CPU sets this bit either in conjunction with bit 7 above or to force the USB/DRD into Full-speed mode when it receive a USB reset.</p> | | | | | | | | | | | | | | | |
| 4 | R/W | 0 | <p>Force_HS</p> <p>The CPU sets this bit either in conjunction with bit 7 above or to force the USB/DRD into High-speed mode when it receive a USB reset.</p> | | | | | | | | | | | | | | | |
| 3 | R/W | 0 | <p>Test_Packet</p> <p>(High-speed mode) The CPU sets this bit to enter Test_Packet test mode. In the mode, the USB/DRD repetitively transmits on the bus a 53-byte test packet, the form of which is defined in the <i>Universal Serial Bus Specification</i> Revision 2.0, Section 7.1.20.</p> <p>Note: The test packet has a fixed format and must be loaded into the Endpoint 0 FIFO before the test mode is entered.</p> | | | | | | | | | | | | | | | |
| 2 | R/W | 0 | <p>Test_K</p> <p>(High-speed mode) The CPU sets this bit to enter the Test_K test mode. In this mode, the USB/DRD transmits a continuous K on the bus.</p> | | | | | | | | | | | | | | | |

| Offset: 0x007C | | | Register Name: USB_TESTC |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 1 | R/W | 0 | Test_J (High-speed mode) The CPU sets this bit to enter the Test_J test mode. In this mode, the USB/DRD transmits a continuous J on the bus. |
| 0 | R/W | 0 | Test_SE0_NAK (High-speed mode) The CPU sets this bit to enter the Test_SE0_NAK test mode. In this mode, the USB/DRD remains in High-speed mode but responds to any valid IN token with a NAK. |

10.4.5.9 0x0080 USB EP0 Control and Status Register (Default Value:0x0000_0000)

| Offset: 0x0080 | | | Register Name: USB_CSR0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:25 | / | / | / |
| 24 | W | 0 | FlushFIFO The CPU writes a '1' to this bit to flush the next packet to be transmitted/read from the Endpoint 0 FIFO. The FIFO pointer is reset, and the TxPktRdy/RxPktRdy bit (below) is cleared. Note: (1) Writing '0' to this bit is ignored. (2) Flush FIFO should only be used when TxPktRdy/RxPktRdy is set, at other times, it may cause data to be corrupted. |
| 23 | W | 0 | ServicedSetupEnd The CPU writes a '1' to this bit to clear the SetupEnd bit. It is cleared automatically. |
| 22 | W | 0 | ServicedRxPktRdy The CPU writes a 1 to this bit to clear the RxPktRdy bit. It is cleared automatically. |
| 21 | W | 0 | SendStall The CPU writes a '1' to this bit to terminate the current transaction. The STALL handshake will be transmitted and then this bit will be cleared automatically. Note: The FIFO should be flushed before SendStall is set. |
| 20 | R | 0 | SetupEnd This bit will be set when a control transaction ends before the DataEnd bit has been set. An interrupt will be generated and the FIFO flushed at this time. The bit is cleared by the CPU writing a '1' to the ServicedSetupEnd bit. |

| Offset: 0x0080 | | | Register Name: USB_CSRO |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 19 | W | 0 | <p>DataEnd</p> <p>The CPU sets this bit: When setting TxPktRdy for the last data packet. When clearing RxPktRdy after unloading the last data packet. When setting TxPktRdy for a zero length data packet. It is cleared automatically.</p> |
| 18 | R/W | 0 | <p>SentStall</p> <p>This bit is set when a STALL handshake is transmitted. The CPU should clear this bit.</p> |
| 17 | R/W | 0 | <p>TxPktRdy</p> <p>The CPU sets this bit after loading a data packet into the FIFO. It is cleared automatically when a data packet has been transmitted. An interrupt is also generated at this point (if enabled).</p> |
| 16 | R | 0 | <p>RxPktRdy</p> <p>This bit is set when a data packet has been received. An interrupt is generated (if enabled) when this bit is set. The CPU clears this bit by setting the ServicedRxPktRdy bit.</p> |
| 15:0 | / | / | / |

10.4.5.10 0x0080 USB EP1~5 Tx Control and Status Register (Default Value:0x0000_0000)

| Offset: 0x0080 | | | Register Name: USB_TXCSR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31 | R/W | 0 | <p>AutoSet</p> <p>If CPU sets this bit, TxPktRdy will be automatically set when data of maximum packet size (value in the USB_TXMAXP) is loaded into the Tx FIFO. If a packet of less than the maximum packet size is loaded, then TxPktRdy will have to be set manually.</p> <p>Note: Should not be set for high-bandwidth Isochronous/Interrupt endpoints.</p> |
| 30 | R/W | 0 | <p>ISO</p> <p>The CPU sets this bit to enable the Tx endpoint for Isochronous transfers, and clears it to enable the Tx endpoint for Bulk or Interrupt transfers.</p> <p>Note: This is only has any effect in Peripheral mode. In Host mode, it always returns zero.</p> |
| 29 | R/W | 0 | <p>Mode</p> <p>The CPU sets this bit to enable the endpoint direction as Tx, and clears the bit to enable it as Rx.</p> <p>Note: This bit only has any affect where the same endpoint FIFO is used for Tx and Rx transactions.</p> |

| Offset: 0x0080 | | | Register Name: USB_TXCSR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 28 | R/W | 0 | <p>DMAReqEnab</p> <p>The CPU sets this bit to enable the DMA request for the Tx endpoint.</p> |
| 27 | R/W | 0 | <p>FrcDataTog</p> <p>The CPU sets this bit to force the endpoint data toggle to switch and the data packet to be cleared from the FIFO, regardless of whether an ACK was received. This can be used by Interrupt Tx endpoints that are used to communicate rate feedback for Isochronous endpoints.</p> |
| 26 | R/W | 0 | <p>DMAReqMode</p> <p>The CPU sets this bit to select DMA Request Mode 1 and clears it to select DMA Request Mode 0.</p> <p>Note: This bit must not be cleared either before or in the same cycle as the above DMAReqEnab bit is cleared.</p> |
| 25:24 | / | / | / |
| 23 | R/W | 0 | <p>IncompTx</p> <p>When the endpoint is being used for high-bandwidth Isochronous/Interrupt transfers, this bit is set to indicate where a large packet has been split into 2 or 3 packets for transmission but insufficient IN tokens have been received to send all the parts.</p> <p>Note: In anything other than a high-bandwidth transfer, this bit will always return 0. And writing '1' to this bit is ignored.</p> |
| 22 | W | 0 | <p>ClrDataTog</p> <p>The CPU writes a 1 to this bit to reset the endpoint data toggle to 0. It is cleared automatically.</p> <p>Note: Writing '0' to this bit is ignored.</p> |
| 21 | R/W | 0 | <p>SentStall</p> <p>This bit is set when a STALL handshake is transmitted. The FIFO is flushed and the TxPktRdy bit is cleared. The CPU should clear this bit.</p> <p>Note: Writing '1' to this bit is ignored.</p> |
| 20 | R/W | 0 | <p>SendStall</p> <p>The CPU writes a 1 to this to issue a STALL handshake to an IN token. The CPU clears this bit to terminate the stall condition.</p> <p>Note:</p> <p>(1) The FIFO should be flushed before SendStall is set.</p> <p>(2) This bit has no effect where the endpoint is being used for Isochronous transfers.</p> |

| Offset: 0x0080 | | | Register Name: USB_TXCSR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 19 | W | 0 | <p>FlushFIFO</p> <p>The CPU writes a 1 to this bit to flush the latest packet from the endpoint Tx FIFO. The FIFO pointer is reset, the TxPktRdy bit (below) is cleared and an interrupt is generated. May be set simultaneously with TxPktRdy to abort the packet that is currently being loaded into the FIFO.</p> <p>Note:</p> <p>(1) Writing '0' to this bit is ignored.</p> <p>(2) Flush FIFO should only be used when TxPktRdy is set, at other times, it may cause data to be corrupted. Also note that, if the FIFO is double-buffered, FlushFIFO may need to be set twice to completely clear the FIFO.</p> |
| 18 | R/W | 0 | <p>UnderRun</p> <p>The USB sets this bit if an IN token is received when the TxPktRdy bit not set. The CPU should clear this bit.</p> <p>Note: Writing '1' to this bit is ignored.</p> |
| 17 | R/W | 0 | <p>FIFONotEmpty</p> <p>The USB sets this bit when there is at least 1 packet in the Tx FIFO.</p> <p>Note: Writing '1' to this bit is ignored.</p> |
| 16 | R/W | 0 | <p>TxPktRdy</p> <p>The CPU sets this bit after loading a data packet into the FIFO. It is cleared automatically when a data packet has been transmitted. An interrupt is also generated at this point (if enabled). TxPktRdy is also automatically cleared prior to loading a second packet to a double buffered FIFO.</p> <p>Note: Writing '0' to this bit is ignored.</p> |

| Offset: 0x0080 | | | Register Name: USB_TXCSR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 15:11 | R/W | 0x00 | <p>PacketCount</p> <p>In the case of Bulk endpoints with the packet splitting option enabled, the Packet Count can be up to 32 and defines the maximum number of USB packets of specified payload into which a single data packet placed in the FIFO should be split, prior to transfer. If the packet splitting option is not enabled, Packet Count is not implemented.</p> <p>For Isochronous/Interrupt endpoints operating in High-speed mode and with the High-bandwidth option enabled, Packet Count may only either 2 or 3 (corresponding to bit 11 or bit 12 set, respectively) and it specifies the maximum number of such transactions that can be take place in a single microframe. If either bit 11 or bit 12 is non-zero, the USB/DRD will automatically split any data packet written to the FIFO into up to 2 or 3 USB packet, each containing the specified payload (or less). For Isochronous/Interrupt transfers in Full-speed mode or if High-bandwidth is not enabled, bit 11 and 12 are ignored.</p> <p>Note: Value for this bits is (Packet Count - 1), but not Packet Count.</p> |
| 10:0 | R/W | 0x000 | <p>MaximumPayload</p> <p>These bits define the maximum payload (in bytes) transmitted in a single transaction. The value set can be up to 1024 bytes but is subject to the constraints placed by the USB Specification on packet sizes for Bulk, Interrupt and Isochronous transfers in Full-speed and High-speed operations.</p> <p>Note:</p> <p>(1) The value written to Maximum Payload (multiplied by Packet Count in the case of high-bandwidth Isochronous/Interrupt transfer) must match the value given in the wMaxPacketSize field of the Standard Endpoint Descriptor for the associated endpoint, a mismatch could cause unexpected result.</p> <p>(2) The total amount of data represented by the value written to this register (Maximum payload × Packet Count) must not exceed the FIFO size for Tx endpoint, and should not exceed half the FIFO size if double-buffering is required.</p> |

10.4.5.11 0x0084 USB EP1~5 Rx Control and Status Register (Default Value:0x0000_0000)

| Offset: 0x0084 | | | Register Name: USB_RXCSR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31 | R/W | 0 | <p>AutoClear If CPU sets this bit then the RxPktRdy will be automatically cleared when data of maximum packet size (value in the USB_TXMAXP) is unloaded from the Rx FIFO. If a packet of less than the maximum packet size is unloaded, then RxPktRdy will have to be cleared manually.</p> <p>Note: Should not be set for high-bandwidth Isochronous endpoints.</p> |
| 30 | R/W | 0 | <p>ISO The CPU sets this bit to enable the Rx endpoint for Isochronous transfers, and clears it to enable the Rx endpoint for Bulk or Interrupt transfers.</p> |
| 29 | R/W | 0 | <p>DMAReqEnab The CPU sets this bit to enable the DMA request for the Rx endpoint.</p> |
| 28 | R/W | 0 | <p>DisNyset_PIDError DisNyset <i>Bulk/Interrupt Transactions:</i> The CPU sets this bit to disable the sending of NYET handshakes. When set, all successfully received Rx packets are ACK'd including at the point at which the FIFO becomes full.</p> <p>Note: This bit only has any affect in High-speed mode, in which mode it should be set for all Interrupt endpoints.</p> <p>PIDError <i>ISO Transactions:</i> The core sets this bit to indicate a PID error in the received packet.</p> |
| 27 | R/W | 0 | <p>DMAReqMode The CPU sets this bit to select DMA Request Mode 1 and clears it to select DMA Request Mode 0.</p> <p>Note: This bit must not be cleared in the same cycle as the above RxPktRdy(or DMAReqEnab) bit is cleared.</p> |
| 26:25 | / | / | / |
| 24 | R/W | 0 | <p>IncompRx This bit will be set in a high-bandwidth Isochronous/Interrupt transfer if the packet received is incomplete. It will be cleared when RxPktRdy is cleared.</p> <p>Note:</p> <p>(1) Writing '1' to this bit is forbidden.</p> <p>(2) In anything other than a high-bandwidth transfer, this bit will always return 0.</p> |

| Offset: 0x0084 | | | Register Name: USB_RXCSR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 23 | W | 0 | <p>ClrDataTog The CPU writes a '1' to this bit to reset the endpoint data toggle to 0. It is cleared automatically.</p> <p>Note: Writing '0' to this bit is ignored.</p> |
| 22 | R/W | 0 | <p>SentStall This bit is set when a STALL handshake is transmitted. The CPU should clear this bit.</p> <p>Note: Writing '1' to this bit is ignored.</p> |
| 21 | R/W | 0 | <p>SendStall The CPU writes a '1' to this to issue a STALL handshake. The CPU clears this bit to terminate the stall condition.</p> <p>Note:</p> <p>(1) The FIFO should be flushed before SendStall is set.</p> <p>(2) This bit has no effect where the endpoint is being used for Isochronous transfers.</p> |
| 20 | W | 0 | <p>FlushFIFO The CPU writes a '1' to this bit to flush the next packet to be read from the endpoint Rx FIFO. The FIFO pointer is reset and the RxPktRdy bit (below) is cleared.</p> <p>Note:</p> <p>(1) Writing '0' to this bit is ignored.</p> <p>(2) Flush FIFO should only be used when RxPktRdy is set, at other times, it may cause data to be corrupted. Also note that, if the FIFO is double-buffered, FlushFIFO may need to be set twice to completely clear the FIFO.</p> |
| 19 | R | 0 | <p>DataError This bit is set when RxPktRdy is set if the data packet has a CRC or bit-stuff error. It is cleared when RxPktRdy is cleared.</p> <p>Note: This bit is only valid when the endpoint is operating in ISO mode. In bulk mode, it always returns zero.</p> |
| 18 | R/W | 0 | <p>OverRun The USB sets this bit if an OUT token can not be loaded into the Rx FIFO. The CPU should clear this bit.</p> <p>Note:</p> <p>(1) Writing '1' to this bit is ignored.</p> <p>(2) This bit is only valid when the endpoint is operating in ISO mode. In bulk mode, it always returns zero.</p> |
| 17 | R | 0 | FIFOFull The USB sets this bit when no more packets can be loaded into the Rx FIFO. |

| Offset: 0x0084 | | | Register Name: USB_RXCSR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 16 | R/W | 0 | <p>RxPktRdy</p> <p>This bit is set when a data packet has been received. The CPU should clear this bit when the packet has been unloaded from the Rx FIFO. An interrupt is generated when the bit is set.</p> <p>Note: Writing '1' to this bit is ignored.</p> |
| 15:11 | R/W | 0x00 | <p>PacketCount</p> <p>In the case of Bulk endpoints with the packet combining option enabled, the Packet Count can be up to 32 and defines the maximum number of USB packets of specified payload which are to be combined into a single data packet within the FIFO.</p> <p>For Isochronous/Interrupt endpoints operating in High-speed mode and with the High-bandwidth option enabled, Packet Count may only either 2 or 3 (corresponding to bit 11 or bit 12 set, respectively) and it specifies the maximum number of such transactions that can be take place in a single microframe. If either bit 11 or bit 12 is non-zero, the USB/DRD will automatically combine the separate USB packets received in any microframe into a single packet within the Rx FIFO. For Isochronous/Interrupt transfers in Full-speed mode or if High-bandwidth is not enabled, bit 11 and 12 are ignored.</p> <p>Note: Value for this bits is (Packet Count - 1), but not Packet Count.</p> |
| 10:0 | R/W | 0x000 | <p>MaximumPayload</p> <p>These bits define the maximum payload (in bytes) transmitted in a single transaction. The value set can be up to 1024 bytes but is subject to the constraints placed by the USB Specification on packet sizes for Bulk, Interrupt and Isochronous transfers in Full-speed and High-speed operations.</p> <p>Note:</p> <p>(1) The value written to Maximum Payload (multiplied by Packet Count in the case of high-bandwidth Isochronous/Interrupt transfer) must match the value given in the wMaxPacketSize field of the Standard Endpoint Descriptor for the associated endpoint, a mismatch could cause unexpected result.</p> <p>(2) The total amount of data represented by the value written to this register (Maximum payload × Packet Count) must not exceed the FIFO size for OUT endpoint, and should not exceed half the FIFO size if double-buffering is required.</p> |

10.4.5.12 0x0088 USB EP0 Rx Counter Register (Default Value:0x0000_0000)

| Offset: 0x0088 | | | Register Name: USB_COUNT0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:7 | / | / | / |
| 6:0 | R | 0x00 | <p>RxCount0 Endpoint 0 Rx Count These bits indicate the number of received data bytes in the Endpoint 0 FIFO.</p> <p>Note: The value returned changes as the FIFO is unloaded and is only valid while RxPktRdy (of USB_CSR0) is set.</p> |

10.4.5.13 0x0090 USB EP1~5 TxFIFO Setting Register (Default Value:0x0000_0000)

| Offset: 0x0090 | | | Register Name: USB_TXFIFO |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:29 | / | / | / |
| 28:16 | R/W | 0x0000 | <p>AD AD[12:0] Start address of the endpoint FIFO is in units of 8 bytes, and it equals (AD[12:0]*8).</p> |
| 15:5 | / | / | / |
| 4 | R/W | 0 | <p>DPB Defines whether double-packet buffering supported. When '1', double-packet buffering is supported. When '0', only single-packet buffering is supported.</p> |
| 3:0 | R/W | 0x0 | <p>SZ SZ[3:0] Maximum packet size to be allowed for (before any splitting within the FIFO of Bulk/High-Bandwidth packets prior to transmission) is $2^{(SZ[3:0]+3)}$ bytes, and the valid values for SZ[3:0] are 0x0–0x09. If DPB=0, the FIFO will also this size; if DPB=1, the FIFO will be twice this size.</p> |

10.4.5.14 0x0094 USB EP1~5 RxFIFO Setting Register (Default Value:0x0000_0000)

| Offset: 0x0094 | | | Register Name: USB_RXFIFO |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:29 | / | / | / |
| 28:16 | R/W | 0x0000 | <p>AD AD[12:0] Start address of the endpoint FIFO is in units of 8 bytes, and it equals (AD[12:0]*8).</p> |

| Offset: 0x0094 | | | Register Name: USB_RXFIFO |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 15:5 | / | / | / |
| 4 | R/W | 0 | DPB Defines whether double-packet buffering supported. When '1', double-packet buffering is supported. When '0', only single-packet buffering is supported. |
| 3:0 | R/W | 0x0 | SZ SZ[3:0] Maximum packet size to be allowed for (before any splitting within the FIFO of Bulk/High-Bandwidth packets prior to transmission) is $2^{(SZ[3:0]+3)}$ bytes, and the valid values for SZ[3:0] are 0x0–0x09. If DPB=0, the FIFO will also this size; if DPB=1, the FIFO will be twice this size. |

10.4.5.15 0x0098 USB Function Address Register (Default Value:0x0000_0000)

| Offset: 0x0098 | | | Register Name: USB_FADDR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:7 | / | / | / |
| 6:0 | RW | 0x00 | FADDR The function address in peripheral mode. This field is reset to zero after a USB bus reset, and should be updated by software after Set_Address Command during USB enumeration. |

10.4.5.16 0x0400 USB Interface Status and Control Register (Default Value:0x0000_0000)

| Offset: 0x0400 | | | Register Name: USB_ISCR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31 | / | 0 | / |
| 30 | R | - | VBUSLS USB VBUS Valid Status detected from Line State |
| 29 | R | - | VBUSEX USB VBUS Valid Status detected from external VBUS input |
| 28 | R | - | IDEX USB ID Status detected from external ID input |
| 27:26 | R | - | LS USB Line Status [27]—DM [26]—DP |
| 25 | R | - | VBUS USB VBUS Status merged from both internal and external |

| Offset: 0x0400 | | | Register Name: USB_ISCR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 24 | R | - | ID USB ID Status merged from both internal and external |
| 23:18 | / | 0 | / |
| 17 | R/W | 0 | IDPullupEn ID pull up enable 0: disable 1: enable ID pull up |
| 16 | R/W | 0 | DataPullupEn DP/DM pull up enable 0: DP/DM pull up disable 1: DP/DM pull up enable |
| 15:14 | R/W | 0 | ForceID Force ID 0x: use external ID Status 10: force ID to LOW 11: force ID to HIGH |
| 13:12 | R/W | 0 | ForceVBus Force VBUS Valid 0x: use external VBUS Valid Status from VBUS Input or Line State 10: Force VBUS Valid to LOW 11: Force VBUS Valid to HIGH |
| 11:10 | R/W | 0 | VBUSSSEL External VBUS Valid Source Select 0x: External VBUS Valid detected from VBUS Input 10: External VBUS Valid detected from DP/DM Input 11: External VBUS Valid detected from either VBUS or DP/DM input |
| 9:8 | / | 0 | / |
| 7 | R/W | 0 | WakeupEn USB Wakeup Enable 0: Disable 1: Enable |
| 6 | R/W | 0 | VBUSCDS VBUS Input Change Detect Status This bit is set by hardware after VBUS input changed when VBUS change detect is enable. Writing '1' will clear this bit. |
| 5 | R/W | 0 | IDCDS ID Input Change Detect Status This bit is set by hardware after ID input changed when ID change detect is enable. Writing '1' will clear this bit. |

| Offset: 0x0400 | | | Register Name: USB_ISCR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 4 | R/W | 0 | <p>DATAACDS DP/DM Input Change Detect Status This bit is set by hardware after DP/DM input changed when DP/DM change detect is enable. Writing '1' will clear this bit.</p> |
| 3 | R/W | 0 | <p>WakeupIE USB Wakeup IRQ Enable 1: Enable USB Wakeup IRQ 0: Disable USB Wakeup IRQ If this bit is set to zero, an USB wakeup event (VBUS/ID/DP/DM change) will generate an USB wakeup request to wakeup the system, but not generate an USB wakeup IRQ to CPU.</p> |
| 2 | R/W | 0 | <p>VBUSCDE VBUS Input Change Detect enable 0: Disable; 1: Enable</p> |
| 1 | RW | 0 | <p>IDCDE ID Input Change Detect enable 0: Disable; 1: Enable</p> |
| 0 | R/W | 0 | <p>DATACDE DP/DM Input Change Detect enable 0: Disable; 1: Enable</p> |

10.4.5.17 0x0410 USB PHY Control Register (Default Value:0x0000_0008)

| Offset: 0x0410 | | | Register Name: USB_PHY_CTL |
|----------------|------------|-------------|--|
| 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_DL |
| 6:4 | / | / | / |
| 3 | R/W | 1 | <p>SIDDQ 1: Write 1 to disable PHY. 0: Write 0 to enable PHY.</p> |
| 2 | / | / | / |
| 1 | R/W | 0 | <p>VC_EN 0: Write 0 to disable PHY VC bus 1: Write 1 to enable PHY VC bus. Note: only 1 VC bus of all the USB controllers could be enabled at the same time</p> |
| 0 | R/W | 0 | VC_CLK |

10.4.5.18 0x0414 USB PHY Test Register (Default Value:0x0000_0000)

| Offset: 0x0414 | | | Register Name: USB_PHY_TEST |
|----------------|------------|-------------|-----------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:15 | / | / | / |
| 14 | R/W | 0x0 | TESTBURNIN |
| 13 | R/W | 0x0 | TESTDATAOUTSEL |
| 12 | R/W | 0x0 | TESTCLK |
| 11:8 | R/W | 0x0 | TESTADDR |
| 7:0 | R/W | 0x0 | TESTDATAIN |

10.4.5.19 0x0418 USB PHY Tune Register (Default Value:0x0234_38E4)

| Offset: 0x0418 | | | Register Name: USB_PHY_TUNE |
|----------------|------------|-------------|-----------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:26 | / | / | / |
| 25:23 | R/W | 0x04 | COMPDISTUNE |
| 22:20 | R/W | 0x03 | SQRXTUNE |
| 19 | R/W | 0x0 | TXPREEMPPULSETUNE |
| 18:16 | R/W | 0x04 | DRDTUNE |
| 15:12 | R/W | 0x03 | TXFSLSTUNE |
| 11:8 | R/W | 0x08 | TXVREFTUNE |
| 7:6 | R/W | 0x03 | TXHSXVTUNE |
| 5:4 | R/W | 0x02 | TXRISETUNE |
| 3:2 | R/W | 0x01 | TXRESTUNE |
| 1:0 | R/W | 0x0 | TXPREEMPAMPTUNE |

10.4.5.20 0x420 USB PHY Select Register (Default Value:0x0000_0001)

| Offset: 0x420 | | | Register Name: USB_PHY_SEL |
|---------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:1 | / | / | Reserved |
| 0 | R/W | 0x1 | OTG_SEL 1: Phy is connected to OTG SIE 0: Phy is connected to HCI SIE |

10.4.5.21 0x424 USB PHY Status Register (Default Value: 0x0000_0000)

| Offset: 0x424 | | | Register Name: USB_PHY_STA |
|---------------|------------|-------------|----------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:18 | / | / | / |
| 17 | R | 0 | BIST_ERROR |

| Offset: 0x424 | | | Register Name: USB_PHY_STA |
|---------------|------------|-------------|----------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 16 | R | 0 | BIST_DONE |
| 15:1 | / | / | / |
| 0 | R | 0 | VC_DO |

10.4.5.22 0x0500 USB DMA Interrupt Enable Register (Default Value:0x0000_0000)

| Offset: 0x0500 | | | Register Name: USB_DMA_INTE |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | Reserved |
| 7 | R/W | 0x0 | USB_DMA7_PKG_INT_EN DMA7 Package End Transfer Interrupt Enable 0: Disable 1: Enable. |
| 6 | R/W | 0x0 | USB_DMA6_PKG_INT_EN DMA6 Package End Transfer Interrupt Enable 0: Disable 1: Enable. |
| 5 | R/W | 0x0 | USB_DMA5_PKG_INT_EN DMA5 Package End Transfer Interrupt Enable 0: Disable 1: Enable. |
| 4 | R/W | 0x0 | USB_DMA4_PKG_INT_EN DMA4 Package End Transfer Interrupt Enable 0: Disable 1: Enable. |
| 3 | R/W | 0x0 | USB_DMA3_PKG_INT_EN DMA3 Package End Transfer Interrupt Enable 0: Disable 1: Enable. |
| 2 | R/W | 0x0 | USB_DMA2_PKG_INT_EN DMA2 Package End Transfer Interrupt Enable 0: Disable 1: Enable. |
| 1 | R/W | 0x0 | USB_DMA1_PKG_INT_EN DMA1 Package End Transfer Interrupt Enable 0: Disable 1: Enable. |
| 0 | R/W | 0x0 | USB_DMA0_PKG_INT_EN DMA0 Package End Transfer Interrupt Enable 0: Disable 1: Enable. |

10.4.5.23 0x0504 USB DMA Interrupt Status Register (Default Value:0x0000_0000)

| Offset: 0x0504 | | | Register Name: USB_DMA_INTS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | Reserved |
| 7 | R/W | 0x0 | USB_DMA7_PKG_INT_STA DMA7 Package End Transfer Interrupt Status. Set 1 to the bit will clean it. 0: No effect 1: Pending |
| 6 | R/W | 0x0 | USB_DMA6_PKG_INT_STA DMA6 Package End Transfer Interrupt Status. Set 1 to the bit will clean it. 0: No effect 1: Pending |
| 5 | R/W | 0x0 | USB_DMA5_PKG_INT_STA DMA5 Package End Transfer Interrupt Status. Set 1 to the bit will clean it. 0: No effect 1: Pending |
| 4 | R/W | 0x0 | USB_DMA4_PKG_INT_STA DMA4 Package End Transfer Interrupt Status. Set 1 to the bit will clean it. 0: No effect 1: Pending |
| 3 | R/W | 0x0 | USB_DMA3_PKG_INT_STA DMA3 Package End Transfer Interrupt Status. Set 1 to the bit will clean it. 0: No effect 1: Pending |
| 2 | R/W | 0x0 | USB_DMA2_PKG_INT_STA DMA2 Package End Transfer Interrupt Status. Set 1 to the bit will clean it. 0: No effect 1: Pending |
| 1 | R/W | 0x0 | USB_DMA1_PKG_INT_STA DMA1 Package End Transfer Interrupt Status. Set 1 to the bit will clean it. 0: No effect 1: Pending |

| Offset: 0x0504 | | | Register Name: USB_DMA_INTS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 0 | R/W | 0x0 | USB_DMA0_PKG_INT_STA DMA0 Package End Transfer Interrupt Status. Set 1 to the bit will clean it. 0: No effect 1: Pending |

10.4.5.24 0x0540+N*0x10(N=0~7) USB DMA Channel Configuration Register (Default Value:0x0000_0000)

| Offset: 0x0540+N*0x10(N=0~7) | | | Register Name: USB_DMA_CHAN_CFG |
|------------------------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31 | R/W | 0x0 | DMA_EN DMA Channel Enable If set to 1, DMA will start the data transfer between the source and the destination. The bit will hold on until the DMA finished. It will be cleared automatically. Set 0 to this bit will stop the corresponding DMA channel and reset its state machine. |
| 30:27 | / | / | / |
| 26:16 | R/W | 0x0 | DMA_BST_LEN DMA Burst Length The value setting on this field should be equated to the usb max packet length of the corresponding endpoint. |
| 15:5 | / | / | / |
| 4 | R/W | 0x0 | DMA_DIR DMA Transfer Direction 0: SDRAM to USB FIFO 1: USB FIFO to SDRAM |
| 3:0 | R/W | 0x0 | DMA_FOR_EP DMA Channel for Endpoint The Endpoint number setting on this field selects the dma channel for the corresponding endpoint. |

10.4.5.25 0x0544+N*0x10(N=0~7) USB DMA SDRAM Start Address Register (Default Value:0x0000_0000)

| Offset: 0x0544+N*0x10(N=0~7) | | | Register Name: USB_DMA_SDRAM_ADD |
|------------------------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R/W | 0x0 | DMA_SDRAM_STR_ADDR DMA SDRAM Start Address The SDRAM start address for the DMA channel transfer between the SDRAM and USB FIFO. |

10.4.5.26 0x0548+N*0x10(N=0~7) USB DMA Byte Counter Register (Default Value:0x0000_0000)

| Offset: 0x0548+N*0x10(N=0~7) | | | Register Name: USB_DMA_BC |
|------------------------------|------------|-------------|----------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:18 | / | / | / |
| 17:0 | R/W | 0x0 | DMA_BC DMA Byte Counter |

10.4.5.27 0x0548+N*0x10(N=0~7) USB DMA RESIDUAL Byte Counter Register (Default Value:0x0000_0000)

| Offset: 0x0548+N*0x10(N=0~7) | | | Register Name: USB_DMA_RESIDUAL_BC |
|------------------------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:18 | / | / | / |
| 17:0 | R/W | 0x0 | DMA_RESIDUAL_BC DMA Residual Byte Counter This field contains the residual byte count in current transfer. |

10.4.6 USB_DRD_HOST Register List

| Module Name | Base Address | Comments |
|--------------|--------------|---------------------|
| USB_DRD_HOST | 0x04101000 | USB Host Controller |

| Register Name | Offset | Description |
|----------------------|--------|---|
| E_CAPLENGTH | 0x000 | EHCI Identification Register |
| E_HCIVERSION | 0x002 | EHCI Host Interface Version Number Register |
| E_HCSPARAMS | 0x004 | EHCI Host Control Structural Parameter Register |
| E_HCCPARAMS | 0x008 | EHCI Host Control Capability Parameter Register |
| E_HCSPPORTROUTE | 0x00c | EHCI Companion Port Route Description |
| E_USBCMD | 0x010 | EHCI USB Command Register |
| E_USBSTS | 0x014 | EHCI USB Status Register |
| E_USBINTR | 0x018 | EHCI USB Interrupt Enable Register |
| E_FRINDEX | 0x01c | EHCI USB Frame Index Register |
| E_PERIODICLISTBASE | 0x024 | EHCI Periodic Frame List Base Address Register |
| E_ASYNCLISTADDR | 0x028 | EHCI Current Asynchronous List Address Register |
| E_CONFIGFLAG | 0x050 | EHCI Configured Flag Register |
| E_PORTSC | 0x054 | EHCI Port Status and Control Register |
| O_HcRevision | 0x400 | OHCI Revision Register |
| O_HcControl | 0x404 | OHCI Control Register |
| O_HcCommandStatus | 0x408 | OHCI Command Status Register |
| O_HcInterruptStatus | 0x40c | OHCI Interrupt Status Register |
| O_HcInterruptEnable | 0x410 | OHCI Interrupt Enable Register |
| O_HcInterruptDisable | 0x414 | OHCI Interrupt Disable Register |

| Register Name | Offset | Description |
|----------------------|--------|---------------------------------------|
| O_HcHCCA | 0x418 | OHCI HCCA Register |
| O_HcPeriodCurrentED | 0x41c | OHCI Period Current ED Register |
| O_HcControlHeadED | 0x420 | OHCI Control Head ED Register |
| O_HcControlCurrentED | 0x424 | OHCI Control Current ED Register |
| O_HcBulkHeadED | 0x428 | OHCI Bulk Head ED Register |
| O_HcBulkCurrentED | 0x42c | OHCI Bulk Current ED Register |
| O_HcDoneHead | 0x430 | OHCI Done Head Register |
| O_HcFmInterval | 0x434 | OHCI Frame Interval Register |
| O_HcFmRemaining | 0x438 | OHCI Frame Remaining Register |
| O_HcFmNumber | 0x43c | OHCI Frame Number Register |
| O_HcPeriodicStart | 0x440 | OHCI Periodic Start Register |
| O_HcLSThreshold | 0x444 | OHCI LS Threshold Register |
| O_HcRhDescriptorA | 0x448 | OHCI Root Hub DescriptorA Register |
| O_HcRhDescriptorB | 0x44c | OHCI Root Hub DescriptorB Register |
| O_HcRhStatus | 0x450 | OHCI Root Hub Status Register |
| O_HcRhPortStatus | 0x454 | OHCI Root Hub Port Status Register |
| USB_CTRL | 0x800 | HCI Interface Register |
| HCI_CTRL3 | 0x808 | HCI Control 3 Register |
| PHY_CTRL | 0x810 | PHY Control Register |
| PHY_STA | 0x824 | PHY Status Register |
| USB_SPDCR | 0x828 | HCI SIE Port Disable Control Register |

10.4.7 USB_DRD_HOST Register Description

10.4.7.1 0x0000 EHCI Identification Register (Default Value:0x10)

| Offset:0x0000 | | | Register Name: E_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. |

10.4.7.2 0x0002 EHCI Host Interface Version Number Register (Default Value:0x0100)

| Offset: 0x0002 | | | Register Name: E_HCIVERSION |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 15:0 | R | 0x0100 | HCIVERSION This is a 16-bits 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. |

10.4.7.3 0x0004 EHCI Host Control Structural Parameter Register (Default Value:0x0000_1101)

| Offset: 0x0004 | | | Register Name: E_HCSPARAMS | | | | | | |
|----------------|---|-------------|--|-------|---------|---|---|---|--|
| Bit | Read/Write | Default/Hex | Description | | | | | | |
| 31:24 | / | / | / | | | | | | |
| 23:20 | R | 0 | <p>Debug Port Number 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. This field will always be '0'.</p> | | | | | | |
| 19:16 | / | / | / | | | | | | |
| 15:12 | R | 1 | <p>Number of Companion Controller (N_CC) 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). This field will always be '0'.</p> | | | | | | |
| 11:8 | R | 1 | <p>Number of Port per Companion Controller(N_PCC) 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. This field will always fix with '0'.</p> | | | | | | |
| 7 | R | 0 | <p>Port Routing Rules 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"> <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 | 1 | <p>N_PORTS 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. This field is always 1.</p> | | | | | | |

10.4.7.4 0x0008 EHCI Host Control Capability Parameter Register (Default Value:0x0000_a026)

| Offset: 0x0008 | | | Register Name: E_HCCPARAMS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:8 | R | 0xa0 | <p>EHCI Extended Capabilities Pointer (EECP) 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>Isochronous Scheduling Threshold 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 | 1 | <p>Asynchronous Schedule Park Capability 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> |

| Offset: 0x0008 | | | Register Name: E_HCCPARAMS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 1 | R | 1 | <p>Programmable Frame List Flag 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</p> <p>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</p> <p>USBCMD register Frame List Size field to configure the host controller.</p> <p>The frame list must always aligned on a 4K page boundary.This requirement ensures that the frame list is always physically contiguous.</p> |
| 0 | / | / | / |

10.4.7.5 0x000C EHCI Companion Port Route Description (Default Value:0x0000_0000)

| Offset: 0x000C | | | Register Name: E_HCSPPORTROUTE |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R | | <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 of 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> |

10.4.7.6 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 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"> <tr> <td>Value</td><td>Minimum Interrupt Interval</td></tr> <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> </table> <p>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.</p> | 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) | | | | | | | | | | | | | | | | | | | | |
| 15:12 | / | / | / | | | | | | | | | | | | | | | | | | |
| 11 | R | 1 | Asynchronous Schedule Park Mode Enable(OPTIONAL) If the Asynchronous Park Capability bit in the HCCPARAMS (Offset: 0x0008) 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 (Offset: 0x0008) 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. | | | | | | | | | | | | | | | | | | |

| Offset: 0x0010 | | | Register Name: USBCMD | | | | | | |
|----------------|--|-------------|---|-----------|---------|---|---|---|--|
| Bit | Read/Write | Default/Hex | Description | | | | | | |
| 7 | R/W | 0 | <p>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</p> | | | | | | |
| 6 | R/W | 0 | <p>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 USBINTR register is a one then the host controller will assert an interrupt at the next interrupt threshold. 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. 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 | 0 | <p>Asynchronous Schedule Enable This bit controls whether the host controller skips processing the Asynchronous Schedule. Values mean:</p> <table border="1"> <tr> <th>Bit Value</th><th>Meaning</th></tr> <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> </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. | | | | | | | | |

| Offset: 0x0010 | | | Register Name: USBCMD | | | | | | | | | | |
|----------------|--|-------------|--|-----------|---------|-----|---------------------------------------|-----|--|-----|---|-----|----------|
| Bit | Read/Write | Default/Hex | Description | | | | | | | | | | |
| 4 | R/W | 0 | <p>Periodic Schedule Enable 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 | 0 | <p>Frame List Size 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(2048bytes)</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(2048bytes) | 10b | 256 elements(1024bytes)For resource-constrained condition | 11b | reserved |
| Bits | Meaning | | | | | | | | | | | | |
| 00b | 1024 elements(4096bytes)Default value | | | | | | | | | | | | |
| 01b | 512 elements(2048bytes) | | | | | | | | | | | | |
| 10b | 256 elements(1024bytes)For resource-constrained condition | | | | | | | | | | | | |
| 11b | reserved | | | | | | | | | | | | |

| Offset: 0x0010 | | | Register Name: USBCMD |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 1 | R/W | 0 | <p>Host Controller Reset 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 | 0 | <p>Run/Stop 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> |

10.4.7.7 0x0014EHCI USB Status Register (Default Value:0x0000_1000)

| Offset: 0x0014 | | | Register Name: E_USBSTS |
|----------------|------------|-------------|-------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |

| Offset: 0x0014 | | | Register Name: E_USBSTS |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 15 | R | 0 | <p>Asynchronous Schedule Status</p> <p>The bit reports the current real status of Asynchronous Schedule. If this bit is a '0' then the status of the Asynchronous Schedule is disabled. If this bit is a '1', 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).</p> |
| 14 | R | 0 | <p>Periodic Schedule Status</p> <p>The bit reports the current real status of the Periodic Schedule. If this bit is a '0' 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 immediately disable or enable the Periodic Schedule when software transitions the Periodic Schedule Enable bit in the USBCMD register. When this bit and the Periodic Schedule Enable bit are the same value, the Periodic Schedule is either enabled (1) or disabled (0).</p> |
| 13 | R | 0 | <p>Reclamation</p> <p>This is a read-only status bit, which is used to detect an empty asynchronous schedule.</p> |
| 12 | R | 1 | <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 | 0 | <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 | 0 | <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> |

| Offset: 0x0014 | | | Register Name: E_USBSTS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 3 | R/WC | 0 | <p>Frame List Rollover 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.</p> |
| 2 | R/WC | 0 | <p>Port Change Detect 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 | 0 | <p>USB Error Interrupt(USBERRINT) 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. This bit and USBINT bit are set.</p> |
| 0 | R/WC | 0 | <p>USB Interrupt(USBINT) 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. 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> |

10.4.7.8 0x0018 EHCI USB Interrupt Enable Register (Default Value:0x0000_0000)

| Offset: 0x0018 | | | Register Name: E_USBINTR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:6 | / | / | / |
| 5 | R/W | 0 | <p>Interrupt on Async Advance Enable 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: E_USBINTR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 4 | R/W | 0 | Host System Error Enable 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. |
| 3 | R/W | 0 | Frame List Rollover Enable 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. |
| 2 | R/W | 0 | Port Change Interrupt Enable 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. |
| 1 | R/W | 0 | USB Error Interrupt Enable 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. The interrupt is acknowledged by software clearing the USBERRINT bit. |
| 0 | R/W | 0 | USB Interrupt Enable 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. The interrupt is acknowledged by software clearing the USBINT bit. |

10.4.7.9 0x001C EHCI Frame Index Register (Default Value:0x0000_0000)

| Offset: 0x001C | | | Register Name: E_FRINDEX |
|----------------|------------|-------------|--------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:14 | / | / | / |

| Offset: 0x001C | | | Register Name: E_FRINDEX | | | | | | | | | | | | | | | |
|-------------------------|-----------------|-------------|---|-------------------------|-----------------|---|-----|------|----|-----|-----|----|-----|-----|----|-----|----------|--|
| Bit | Read/Write | Default/Hex | Description | | | | | | | | | | | | | | | |
| 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.</p> <p>The following illustrates Values of N based on the value of the Frame List Size field in the USBCMD register.</p> <table border="1"> <thead> <tr> <th>USBCMD[Frame List Size]</th><th>Number Elements</th><th>N</th></tr> </thead> <tbody> <tr> <td>00b</td><td>1024</td><td>12</td></tr> <tr> <td>01b</td><td>512</td><td>11</td></tr> <tr> <td>10b</td><td>256</td><td>10</td></tr> <tr> <td>11b</td><td>Reserved</td><td></td></tr> </tbody> </table> | USBCMD[Frame List Size] | Number Elements | N | 00b | 1024 | 12 | 01b | 512 | 11 | 10b | 256 | 10 | 11b | Reserved | |
| USBCMD[Frame List Size] | Number Elements | N | | | | | | | | | | | | | | | | |
| 00b | 1024 | 12 | | | | | | | | | | | | | | | | |
| 01b | 512 | 11 | | | | | | | | | | | | | | | | |
| 10b | 256 | 10 | | | | | | | | | | | | | | | | |
| 11b | Reserved | | | | | | | | | | | | | | | | | |

**NOTE**

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

10.4.7.10 0x0024 EHCI Periodic Frame List Base Address Register (Default Value:0x0000_0000)

| Offset: 0x0024 | | | Register Name: E_PERIODICLISTBASE |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:12 | R/W | 0x0 | <p>Base Address</p> <p>These bits correspond to memory address signals [31:12], respectively.</p> <p>This register contains the beginning address of the Periodic Frame List in the system memory.</p> <p>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-K byte 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.</p> |
| 11:0 | / | / | / |

**NOTE**

Writes must be Dword Writes.

10.4.7.11 0x0028 EHCI Current Asynchronous List Address Register (Default Value:0x0000_0000)

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

**NOTE**

Write must be DWord Writes.

10.4.7.12 0x0050 EHCI Configure Flag Register (Default Value:0x0000_0000)

| Offset: 0x0050 | | | Register Name: E_CONFIGFLAG | | | | | | |
|----------------|--|-------------|--|-------|---------|---|--|---|---|
| Bit | Read/Write | Default/Hex | Description | | | | | | |
| 31:1 | / | / | / | | | | | | |
| 0 | R/W | 0 | <p>Configure Flag(CF) 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"> <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 use in the normal implementation.

10.4.7.13 0x0054 EHCI Port Status and Control Register (Default Value:0x0000_2000)

| Offset: 0x0054 | | | Register Name: E_PORTSC | | | | | | | | | | | | | | | | |
|----------------|---|-------------|--|------|-----------|-------|---|-------|--------------|-------|--------------|-------|--------------|-------|-------------|-------|-------------------|-------------|----------|
| Bit | Read/Write | Default/Hex | Description | | | | | | | | | | | | | | | | |
| 31:22 | / | / | / | | | | | | | | | | | | | | | | |
| 21 | R/W | 0 | <p>Wake on Disconnect Enable(WKDSCNNT_E) Writing this bit to a one enables the port to be sensitive to device disconnects as wake-up events. This field is zero if Port Power is zero. The default value in this field is '0'.</p> | | | | | | | | | | | | | | | | |
| 20 | R/W | 0 | <p>Wake on Connect Enable(WKCNNT_E) Writing this bit to a one enable the port to be sensitive to device connects as wake-up events. This field is zero if Port Power is zero. The default value in this field is '0'.</p> | | | | | | | | | | | | | | | | |
| 19:16 | R/W | 0 | <p>Port Test Control The value in this field specifies the test mode of the port. The encoding of the test mode bits are as follow:</p> <table border="1"><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 SE0_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 SE0_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 SE0_NAK | | | | | | | | | | | | | | | | | | |
| 0100b | Test Packet | | | | | | | | | | | | | | | | | | |
| 0101b | Test FORCE_ENABLE | | | | | | | | | | | | | | | | | | |
| 0110b-1111b | Reserved | | | | | | | | | | | | | | | | | | |
| 15:14 | / | / | / | | | | | | | | | | | | | | | | |

| Offset: 0x0054 | | | Register Name: E_PORTSC | | | | | | | | | | | | | | | |
|----------------|------------|--|--|------------|-----------|----------------|-----|-----|---|-----|---------|---|-----|---------|--|-----|-----------|---|
| Bit | Read/Write | Default/Hex | Description | | | | | | | | | | | | | | | |
| 13 | R/W | 1 | <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. 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 | / | / | / | | | | | | | | | | | | | | | |
| 11:10 | R | 0 | <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>SEO</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 | SEO | 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 | SEO | 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 | / | / | / | | | | | | | | | | | | | | | |

| Offset: 0x0054 | | | Register Name: E_PORTSC |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 8 | R/W | 0 | <p>Port Reset 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. Notes: when software writes this bit to a one, it must also write a zero to the Port Enable bit.</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> <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. This field is zero if Port Power is zero.</p> |

| Offset: 0x0054 | | | Register Name: E_PORTSC | | | | | | | | |
|-----------------------------|------------|-------------|--|-----------------------------|------------|----|---------|----|--------|----|---------|
| Bit | Read/Write | Default/Hex | Description | | | | | | | | |
| 7 | R/W | 0 | <p>Suspend</p> <p>Port Enabled Bit and Suspend bit of this register define the port states as follows:</p> <table border="1"> <thead> <tr> <th>Bits[Port Enables, Suspend]</th><th>Port State</th></tr> </thead> <tbody> <tr> <td>0x</td><td>Disable</td></tr> <tr> <td>10</td><td>Enable</td></tr> <tr> <td>11</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. Note 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> <ul style="list-style-type: none"> ① Software sets the Force Port Resume bit to a zero (from a one). ② Software sets the Port Reset bit to a one (from a zero). 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>This field is zero if Port Power is zero.</p> <p>The default value in this field is '0'.</p> | Bits[Port Enables, Suspend] | Port State | 0x | Disable | 10 | Enable | 11 | Suspend |
| Bits[Port Enables, Suspend] | Port State | | | | | | | | | | |
| 0x | Disable | | | | | | | | | | |
| 10 | Enable | | | | | | | | | | |
| 11 | Suspend | | | | | | | | | | |

| Offset: 0x0054 | | | Register Name: E_PORTSC |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 6 | R/W | 0 | <p>Force Port Resume 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 a 1 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 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 | 0 | <p>Over-current Change Default = 0. 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 | 0 | <p>Over-current Active 0 = This port does not have an over-current condition. 1 = This port currently has an over-current condition. This bit will automatically transition from a one to a zero when the over current condition is removed. The default value of this bit is '0'.</p> |

| Offset: 0x0054 | | | Register Name: E_PORTSC |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 3 | R/WC | 0 | <p>Port Enable/Disable Change Default = 0. 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. This field is zero if Port Power is zero.</p> |
| 2 | R/W | 0 | <p>Port Enabled/Disabled 1=Enable, 0=Disable. 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. 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 | 0 | <p>Connect Status Change 1=Change in Current Connect Status, 0=No change, Default=0. Indicates a change has occurred in the port's Current Connect Status. 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 | 0 | <p>Current Connect Status 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.

10.4.7.14 0x0400 OHCI Revision Register (Default Value:0x10)

| Offset: 0x0400 | | | | Register Name: O_HcRevision |
|----------------|------------|----|-------------|--|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 31:8 | / | / | / | / |
| 7:0 | R | R | 0x10 | <p>Revision</p> <p>This read-only field contains the BCD representation of the version of the HCI specification that is implemented by this HC. For example, a value of 0x11 corresponds to version 1.1. All of the HC implementations that are compliant with this specification will have a value of 0x10.</p> |

10.4.7.15 0x0404 OHCI Control Register (Default Value:0x0000_0000)

| Offset: 0x0404 | | | | Register Name: O_HcControl |
|----------------|------------|-----|-------------|--|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 31:11 | / | / | / | / |
| 10 | R/W | R | 0x0 | <p>RemoteWakeupEnable</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 ResumeDetected bit in HcInterruptStatus (Offset: 0x040C) 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>RemoteWakeupConnected</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>InterruptRouting</p> <p>This bit determines the routing of interrupts generated by events registered in HcInterruptStatus (Offset: 0x040C). If clear, all interrupts are routed to the normal host bus interrupt mechanism. If set interrupts are routed to the System</p> |

| Offset: 0x0404 | | | | Register Name: O_HcControl | | | | | | | | |
|----------------|----------------|-----|-------------|---|-----|----------|-----|-----------|-----|----------------|-----|------------|
| Bit | Read/Write | | Default/Hex | Description | | | | | | | | |
| | HCD | HC | | | | | | | | | | |
| | | | | 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. | | | | | | | | |
| 7:6 | R/W | R/W | 0x0 | <p>HostControllerFunctionalState for USB</p> <table border="1"> <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 StartOfFrame field of HcInterruptStatus (Offset: 0x040C).</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 Hub and asserts subsequent reset signaling to downstream ports.</p> | 00b | USBReset | 01b | USBResume | 10b | USBOperational | 11b | USBSuspend |
| 00b | USBReset | | | | | | | | | | | |
| 01b | USBResume | | | | | | | | | | | |
| 10b | USBOperational | | | | | | | | | | | |
| 11b | USBSuspend | | | | | | | | | | | |
| 5 | R/W | R | 0x0 | <p>BulkListEnable</p> <p>This bit is set to enable the processing of the Bulk list in the next Frame. If cleared by HCD, 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 HcBulkCurrentED (Offset: 0x42C) is pointing to an ED to be removed, HCD must advance the pointer by updating HcBulkCurrentED (Offset: 0x42C) before re-enabling processing of the list.</p> | | | | | | | | |
| 4 | R/W | R | 0x0 | <p>ControlListEnable</p> <p>This bit is set to enable the processing of the Control list in the next Frame. If cleared by HCD, 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 HcControlCurrentED (Offset: 0x424) is pointing to an ED to be removed, HCD must advance the pointer by updating HcControlCurrentED (Offset: 0x424) before re-enabling processing of the list.</p> | | | | | | | | |
| 3 | R/W | R | 0x0 | <p>IsochronousEnable</p> <p>This bit is used by HCD to enable/disable processing of isochronous EDs. While processing the periodic list in a Frame,</p> | | | | | | | | |

| Offset: 0x0404 | | | | Register Name: O_HcControl | | | | | | | | | | |
|----------------|---|----|-------------|--|------|---|---|-----|---|-----|---|-----|---|-----|
| Bit | Read/Write | | Default/Hex | Description | | | | | | | | | | |
| | HCD | HC | | | | | | | | | | | | |
| | | | | <p>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.</p> <p>Setting this bit is guaranteed to take effect in the next Frame (not the current Frame).</p> | | | | | | | | | | |
| 2 | R/W | R | 0x0 | <p>PeriodicListEnable</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>ControlBulkServiceRatio</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"> <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> <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> </tbody> </table> <p>The default value is 0x0.</p> | CBSR | No. of Control EDs Over Bulk EDs Served | 0 | 1:1 | 1 | 2:1 | 2 | 3:1 | 3 | 4:1 |
| CBSR | No. of Control EDs Over Bulk EDs Served | | | | | | | | | | | | | |
| 0 | 1:1 | | | | | | | | | | | | | |
| 1 | 2:1 | | | | | | | | | | | | | |
| 2 | 3:1 | | | | | | | | | | | | | |
| 3 | 4:1 | | | | | | | | | | | | | |

10.4.7.16 0x0408 OHCI Command Status Register (Default Value:0x0000_0000)

| Offset: 0x0408 | | | | Register Name: O_HcCommandStatus |
|----------------|------------|----|-------------|----------------------------------|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 31:18 | / | / | 0x0 | Reserved |

| Offset: 0x0408 | | | | Register Name: O_HcCommandStatus |
|----------------|------------|-----|-------------|---|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 17:16 | R | R/W | 0x0 | SchedulingOverrunCount 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 SchedulingOverrun in HcInterruptStatus (Offset: 0x040C) has already been set. This is used by HCD to monitor any persistent scheduling problem. |
| 15:4 | / | / | / | / |
| 3 | R/W | R/W | 0x0 | OwnershipChangeRequest This bit is set by an OS HCD to request a change of control of the HC. When set HC will set the OwnershipChange field in HcInterruptStatus (Offset: 0x040C) . After the changeover, this bit is cleared and remains so until the next request from OS HCD. |
| 2 | R/W | R/W | 0x0 | BulkListFilled 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 BulkListFilled is 0, HC will not start processing the Bulk list. If BulkListFilled 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 BulkListFilled to 1 causing the Bulk list processing to continue. If no TD is found on the Bulk list, and if HCD does not set BulkListFilled, then BulkListFilled will still be 0 when HC completes processing the Bulk list and Bulk list processing will stop. |
| 1 | R/W | R/W | 0x0 | ControlListFilled 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. When HC begins to process the head of the Control list, it checks CLF. As long as ControlListFilled is 0, HC will not start processing the Control list. If CF is 1, HC will start processing the Control list and will set ControlListFilled to 0. If HC finds a TD on the list, then HC will set ControlListFilled 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 ControlListFilled, then ControlListFilled will still be 0 when HC completes processing the Control list and Control list processing will stop. |

| Offset: 0x0408 | | | | Register Name: O_HcCommandStatus |
|----------------|------------|-----|-------------|--|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 0 | R/W | R/E | 0x0 | HostControllerReset 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 (Offset: 0x404) , 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 10ms. 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. |

10.4.7.17 0x040C OHCI Interrupt Status Register (Default Value:0x0000_0000)

| Offset: 0x040C | | | | Register Name: O_HcInterruptStatus |
|----------------|------------|-----|-------------|---|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 31:7 | / | / | / | / |
| 6 | R/W | R/W | 0x0 | RootHubStatusChange This bit is set when the content of HcRhStatus (Offset: 0x450) or the content of any of HcRhPortStatus [NumberofDownstreamPort] has changed. |
| 5 | R/W | R/W | 0x0 | FrameNumberOverflow This bit is set when the MSb of HcFmNumber (Offset: 0x43C,bit 15) changes value, from 0 to 1 or from 1 to 0, and after HccaFrameNumber (Offset:0x43C) has been updated. |
| 4 | R/W | R/W | 0x0 | UnrecoverableError 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 | ResumeDetected 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 resume signaling causing this bit to be set. This bit is not set when HCD sets the USBRseume state. |

| Offset: 0x040C | | | | Register Name: O_HcInterruptStatus |
|----------------|------------|-----|-------------|--|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 2 | R/W | R/W | 0x0 | <p>StartofFrame 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.</p> |
| 1 | R/W | R/W | 0x0 | <p>WritebackDoneHead 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.</p> |
| 0 | R/W | R/W | 0x0 | <p>SchedulingOverrun 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 (Offset: 0x408) to be Incremented.</p> |

10.4.7.18 0x0410 OHCI Interrupt Enable Register (Default Value:0x0000_0000)

| Offset: 0x0410 | | | | Register Name: O_HcInterruptEnable | | | | |
|----------------|--|----|-------------|---|---|---------|---|--|
| Bit | Read/Write | | Default/Hex | Description | | | | |
| | HCD | HC | | | | | | |
| 31 | R/W | R | 0x0 | <p>MasterInterruptEnable 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.</p> | | | | |
| 30:7 | / | / | / | / | | | | |
| 6 | R/W | R | 0x0 | <p>RootHubStatusChange Interrupt Enable</p> <table border="1"> <tr> <td>0</td><td>Ignore;</td></tr> <tr> <td>1</td><td>Enable interrupt generation due to Root Hub Status Change;</td></tr> </table> | 0 | Ignore; | 1 | Enable interrupt generation due to Root Hub Status Change; |
| 0 | Ignore; | | | | | | | |
| 1 | Enable interrupt generation due to Root Hub Status Change; | | | | | | | |
| 5 | R/W | R | 0x0 | <p>FrameNumberOverflow Interrupt Enable</p> <table border="1"> <tr> <td>0</td><td>Ignore;</td></tr> <tr> <td>1</td><td>Enable interrupt generation due to Frame Number Overflow;</td></tr> </table> | 0 | Ignore; | 1 | Enable interrupt generation due to Frame Number Overflow; |
| 0 | Ignore; | | | | | | | |
| 1 | Enable interrupt generation due to Frame Number Overflow; | | | | | | | |
| 4 | R/W | R | 0x0 | <p>UnrecoverableError Interrupt Enable</p> <table border="1"> <tr> <td>0</td><td>Ignore;</td></tr> <tr> <td>1</td><td>Enable interrupt generation due to Unrecoverable Error;</td></tr> </table> | 0 | Ignore; | 1 | Enable interrupt generation due to Unrecoverable Error; |
| 0 | Ignore; | | | | | | | |
| 1 | Enable interrupt generation due to Unrecoverable Error; | | | | | | | |

| Offset: 0x0410 | | | | Register Name: O_HcInterruptEnable |
|----------------|--|----|-------------|--|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 3 | R/W | R | 0x0 | ResumeDetected Interrupt Enable |
| | | | | <table border="1"> <tr> <td>0</td><td>Ignore;</td></tr> <tr> <td>1</td><td>Enable interrupt generation due to Resume Detected;</td></tr> </table> |
| 0 | Ignore; | | | |
| 1 | Enable interrupt generation due to Resume Detected; | | | |
| 2 | R/W | R | 0x0 | StartofFrame Interrupt Enable |
| | | | | <table border="1"> <tr> <td>0</td><td>Ignore;</td></tr> <tr> <td>1</td><td>Enable interrupt generation due to Start of Frame;</td></tr> </table> |
| 0 | Ignore; | | | |
| 1 | Enable interrupt generation due to Start of Frame; | | | |
| 1 | R/W | R | 0x0 | WritebackDoneHead Interrupt Enable |
| | | | | <table border="1"> <tr> <td>0</td><td>Ignore;</td></tr> <tr> <td>1</td><td>Enable interrupt generation due to Write back Done Head;</td></tr> </table> |
| 0 | Ignore; | | | |
| 1 | Enable interrupt generation due to Write back Done Head; | | | |
| 0 | R/W | R | 0x0 | SchedulingOverrun Interrupt Enable |
| | | | | <table border="1"> <tr> <td>0</td><td>Ignore;</td></tr> <tr> <td>1</td><td>Enable interrupt generation due to Scheduling Overrun;</td></tr> </table> |
| 0 | Ignore; | | | |
| 1 | Enable interrupt generation due to Scheduling Overrun; | | | |

10.4.7.19 0x0414 OHCI Interrupt Disable Register (Default Value:0x0000_0000)

| Offset: 0x0414 | | | | Register Name: O_HcInterruptDisable | | | | |
|----------------|---|----|---------|--|---|---------|---|---|
| Bit | Read/Write | | Default | Description | | | | |
| | HCD | HC | | | | | | |
| 31 | R/W | R | 0x0 | <p>MasterInterruptEnable</p> <p>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.</p> | | | | |
| 30:7 | / | / | / | / | | | | |
| 6 | R/W | R | 0x0 | <p>RootHubStatusChange Interrupt Disable</p> <table border="1"> <tr> <td>0</td><td>Ignore;</td></tr> <tr> <td>1</td><td>Disable interrupt generation due to Root Hub Status Change;</td></tr> </table> | 0 | Ignore; | 1 | Disable interrupt generation due to Root Hub Status Change; |
| 0 | Ignore; | | | | | | | |
| 1 | Disable interrupt generation due to Root Hub Status Change; | | | | | | | |
| 5 | R/W | R | 0x0 | <p>FrameNumberOverflow Interrupt Disable</p> <table border="1"> <tr> <td>0</td><td>Ignore;</td></tr> <tr> <td>1</td><td>Disable interrupt generation due to Frame Number Overflow;</td></tr> </table> | 0 | Ignore; | 1 | Disable interrupt generation due to Frame Number Overflow; |
| 0 | Ignore; | | | | | | | |
| 1 | Disable interrupt generation due to Frame Number Overflow; | | | | | | | |
| 4 | R/W | R | 0x0 | <p>UnrecoverableError Interrupt Disable</p> <table border="1"> <tr> <td>0</td><td>Ignore;</td></tr> <tr> <td>1</td><td>Disable interrupt generation due to Unrecoverable Error;</td></tr> </table> | 0 | Ignore; | 1 | Disable interrupt generation due to Unrecoverable Error; |
| 0 | Ignore; | | | | | | | |
| 1 | Disable interrupt generation due to Unrecoverable Error; | | | | | | | |
| 3 | R/W | R | 0x0 | <p>ResumeDetected Interrupt Disable</p> <table border="1"> <tr> <td>0</td><td>Ignore;</td></tr> <tr> <td>1</td><td>Disable interrupt generation due to Resume Detected;</td></tr> </table> | 0 | Ignore; | 1 | Disable interrupt generation due to Resume Detected; |
| 0 | Ignore; | | | | | | | |
| 1 | Disable interrupt generation due to Resume Detected; | | | | | | | |

| Offset: 0x0414 | | | | Register Name: O_HcInterruptDisable |
|----------------|---|----|---------|---|
| Bit | Read/Write | | Default | Description |
| | HCD | HC | | |
| 2 | R/W | R | 0x0 | StartofFrame Interrupt Disable |
| | | | | <table border="1"> <tr> <td>0</td><td>Ignore;</td></tr> <tr> <td>1</td><td>Disable interrupt generation due to Start of Flame;</td></tr> </table> |
| 0 | Ignore; | | | |
| 1 | Disable interrupt generation due to Start of Flame; | | | |
| 1 | R/W | R | 0x0 | WritebackDoneHead Interrupt Disable |
| | | | | <table border="1"> <tr> <td>0</td><td>Ignore;</td></tr> <tr> <td>1</td><td>Disable interrupt generation due to Write back Done Head;</td></tr> </table> |
| 0 | Ignore; | | | |
| 1 | Disable interrupt generation due to Write back Done Head; | | | |
| 0 | R/w | R | 0x0 | SchedulingOverrun Interrupt Disable |
| | | | | <table border="1"> <tr> <td>0</td><td>Ignore;</td></tr> <tr> <td>1</td><td>Disable interrupt generation due to Scheduling Overrun;</td></tr> </table> |
| 0 | Ignore; | | | |
| 1 | Disable interrupt generation due to Scheduling Overrun; | | | |

10.4.7.20 0x0418 OHCI HCCA Register (Default Value:0x0000_0000)

| Offset: 0x0418 | | | | Register Name: O_HcHCCA |
|----------------|------------|----|-------------|---|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 31:8 | R/W | R | 0x0 | <p>HCCA[31:8]</p> <p>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.</p> |
| 7:0 | R | R | 0x0 | <p>HCCA[7:0]</p> <p>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.</p> |

10.4.7.21 0x041C OHCI Period Current ED Register (Default Value:0x0000_0000)

| Offset: 0x041C | | | | Register Name: O_HcPeriodCurrentED |
|----------------|------------|-----|-------------|--|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 31:4 | R | R/W | 0x0 | <p>PCED[31:4]</p> <p>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.</p> |

| Offset: 0x041C | | | | Register Name: O_HcPeriodCurrentED |
|----------------|------------|----|-------------|--|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 3:0 | R | R | 0x0 | PCED[3:0] 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. |

10.4.7.22 0x0420 OHCI Control Head ED Register (Default Value:0x0000_0000)

| Offset: 0x0420 | | | | Register Name: O_HcControlHeadED |
|----------------|------------|----|-------------|---|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 31:4 | R/W | R | 0x0 | EHCD[31:4] 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 | EHCD[3:0] 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. |

10.4.7.23 0x0424 OHCI Control Current ED Register (Default Value:0x0000_0000)

| Offset: 0x0424 | | | | Register Name: HcControlCurrentED |
|----------------|------------|-----|-------------|--|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 31:4 | R/W | R/W | 0x0 | <p>CCED[31:4]</p> <p>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 ControlListFilled of in HcCommandStatus (Offset: 0x0408). If set, it copies the content of HcControlHeadED (Offset: 0x0420) to HcControlCurrentED and clears the bit. If not set, it does nothing.</p> <p>HCD is allowed to modify this register only when the ControlListEnable of HcControl (Offset: 0x0404) 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.</p> |
| 3:0 | R | R | 0x0 | <p>CCED[3:0]</p> <p>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.</p> |

10.4.7.24 0x0428 OHCI Bulk Head ED Register (Default Value:0x0000_0000)

| Offset: 0x0428 | | | | Register Name: O_HcBulkHeadED |
|----------------|------------|----|-------------|---|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 31:4 | R/W | R | 0x0 | <p>BHED[31:4]</p> <p>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.</p> |
| 3:0 | R | R | 0x0 | <p>BHED[3:0]</p> <p>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.</p> |

10.4.7.25 0x042C OHCI Bulk Current ED Register (Default Value:0x0000_0000)

| Offset: 0x042C | | | | Register Name: O_HcBulkCurrentED |
|----------------|------------|-----|-------------|--|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 31:4 | R/W | R/W | 0x0 | BulkCurrentED[31:4] 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 ControlListFilled of HcControl (Offset: 0x0404) . If set, it copies the content of HcBulkHeadED (Offset: 0x0428) to HcBulkCurrentED and clears the bit. If it is not set, it does nothing. HCD is only allowed to modify this register when the BulkListEnable of HcControl (Offset: 0x0404) 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 | BulkCurrentED [3:0] 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. |

10.4.7.26 0x0430 OHCI Done Head Register (Default Value:0x0000_0000)

| Offset: 0x0430 | | | | Register Name: O_HcDoneHead |
|----------------|------------|-----|-------------|--|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 31:4 | R | R/W | 0x0 | HcDoneHead[31:4] When a TD is completed, HC writes the content of HcDoneHead to the NextTD field of the TD. HC then overwrites the content of HcDoneHead 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 WritebackDoneHead of HcInterruptStatus (Offset: 0x040C) . |
| 3:0 | R | R | 0x0 | HcDoneHead[3:0] 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. |

10.4.7.27 0x0434 OHCI Frame Interval Register (Default Value:0x0000_2EDF)

| Offset: 0x0434 | | | | Register Name: O_HcFmInterval |
|----------------|------------|----|-------------|--|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 31 | R/W | R | 0x0 | FrameIntervalToggler HCD toggles this bit whenever it loads a new value to FrameInterval. |
| 30:16 | R/W | R | 0x0 | FSLargestDataPacket 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 | FrameInterval 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 HostControllerReset field of HcCommandStatus (Offset: 0x408) 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. |

10.4.7.28 0x0438 OHCI Frame Remaining Register (Default Value:0x0000_0000)

| Offset: 0x0438 | | | | Register Name: O_HcFmRemaining |
|----------------|------------|-----|-------------|---|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 31 | R | R/W | 0x0 | FrameRemaining Toggle This bit is loaded from the FrameIntervalToggle field of HcFmInterval (Offset: 0x0434) whenever FrameRemaining reaches 0. This bit is used by HCD for the synchronization between FrameInterval and FrameRemaining. |
| 30:14 | / | / | / | / |
| 13:0 | R | RW | 0x0 | FramRemaining This counter is decremented at each bit time. When it reaches zero, it is reset by loading the FrameInterval value specified in HcFmInterval (Offset: 0x0434) at the next bit time boundary. When entering the USBOPERATIONAL state, HC re-loads the content with the FrameInterval of HcFmInterval (Offset: 0x0434) and uses the updated value from the next SOF. |

10.4.7.29 0x043C OHCI Frame Number Register (Default Value:0x0000_0000)

| Offset: 0x043C | | | | Register Name: O_HcFmNumber |
|----------------|------------|-----|-------------|---|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 31:16 | / | / | / | / |
| 15:0 | R | R/W | 0x0 | <p>FrameNumber</p> <p>This is incremented when HcFmRemaining (Offset: 0x0438) is re-loaded. It will be rolled over to 0x0 after 0xffff. When entering the USBOPERATIONAL state, this will be incremented automatically. The content will be written to HCCA after HC has incremented the FrameNumber 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 StartofFrame in HcInterruptStatus (Offset: 0x040C).</p> |

10.4.7.30 0x0440 OHCI Periodic Start Register (Default Value:0x0000_0000)

| Offset: 0x0440 | | | | Register Name: O_HcPeriodicStatus |
|----------------|------------|----|-------------|---|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 31:14 | / | / | / | / |
| 13:0 | R/W | R | 0x0 | <p>PeriodicStart</p> <p>After a hardware reset, this field is cleared. This is then set by HCD during the HC initialization. The value is calculated roughly as 10% off from HcFmInterval (Offset: 0x0434). A typical value will be 0x2A3F (0x3e67??). When HcFmRemaining (Offset: 0x0438) 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.</p> |

10.4.7.31 0x0444 OHCI LS Threshold Register (Default Value:0x0000_0628)

| Offset: 0x0444 | | | | Register Name: O_HcLSThreshold |
|----------------|------------|----|-------------|--------------------------------|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 31:12 | | | | Reserved |

| Offset: 0x0444 | | | | Register Name: O_HcLSThreshold |
|----------------|------------|----|-------------|--|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 11:0 | R/W | R | 0x0628 | LSThreshold This field contains a value which is compared to the FrameRemaining field prior to initiating a Low Speed transaction. The transaction is started only if FrameRemaining ³ this field. The value is calculated by HCD with the consideration of transmission and setup overhead. |

10.4.7.32 0x0448 OHCI Root Hub DescriptorA Register (Default Value:0x0200_1201)

| Offset: 0x0448 | | | | Register Name: O_HcRhDescriptorA | | | | |
|----------------|--|----|-------------|--|---|--|---|--|
| Bit | Read/Write | | Default/Hex | Description | | | | |
| | HCD | HC | | | | | | |
| 31:24 | R/W | R | 0x2 | PowerOnToPowerGoodTime[POTPGT] 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. | | | | |
| 23:13 | / | / | / | / | | | | |
| 12 | R/W | R | 1 | NoOverCurrentProtection This bit describes how the overcurrent status for the Root Hub ports are reported. When this bit is cleared, the OverCurrentProtectionMode field specifies global or per-port reporting. <table border="1" data-bbox="690 1370 1437 1493"> <tr> <td>0</td><td>Over-current status is reported collectively for all downstream ports.</td></tr> <tr> <td>1</td><td>No overcurrent protection supported.</td></tr> </table> | 0 | Over-current status is reported collectively for all downstream ports. | 1 | No overcurrent protection supported. |
| 0 | Over-current status is reported collectively for all downstream ports. | | | | | | | |
| 1 | No overcurrent protection supported. | | | | | | | |
| 11 | R/W | R | 0 | OverCurrentProtectionMode This bit describes how the overcurrent status for the Root Hub ports are reported. At reset, these fields should reflect the same mode as PowerSwitchingMode. This field is valid only if the NoOverCurrentProtection field is cleared. <table border="1" data-bbox="690 1718 1437 1841"> <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 | Device Type 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. | | | | |

| Offset: 0x0448 | | | | Register Name: O_HcRhDescriptorA | | | | |
|----------------|---|----|-------------|---|---|---|---|---|
| Bit | Read/Write | | Default/Hex | Description | | | | |
| | HCD | HC | | | | | | |
| 9 | R/W | R | 1 | PowerSwitchingMode 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 NoPowerSwitching field is cleared. | | | | |
| | | | | <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 PortPowerControlMask bit is set, the port responds only to port power commands (Set/ClearPortPower). If the port mask is cleared, then the port is controlled only by the global power switch (Set/ClearGlobalPower).</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 PortPowerControlMask bit is set, the port responds only to port power commands (Set/ClearPortPower). If the port mask is cleared, then the port is controlled only by the global power switch (Set/ClearGlobalPower). |
| 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 PortPowerControlMask bit is set, the port responds only to port power commands (Set/ClearPortPower). If the port mask is cleared, then the port is controlled only by the global power switch (Set/ClearGlobalPower). | | | | | | | |
| 8 | R/W | R | 0 | NoPowerSwithcing 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 PowerSwitchingMode specifies global or per-port switching. | | | | |
| | | | | <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 | NumberDownstreamPorts 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. | | | | |

10.4.7.33 0x044C OHCI Root Hub DescriptorB Register (Default Value:0x0000_0000)

| Offset: 0x044C | | | | Register Name: O_HcRhDescriptorB | | | | | | | | | | |
|----------------|--------------------------------|----|-------------|---|------|----------|------|-------------------------------|------|-------------------------------|-----|--|-------|--------------------------------|
| Bit | Read/Write | | Default/Hex | Description | | | | | | | | | | |
| | HCD | HC | | | | | | | | | | | | |
| 31:16 | R/W | R | 0x0 | <p>PortPowerControlMask</p> <p>Each bit indicates if a port is affected by a global power control command when PowerSwitchingMode is set. When set, the port's power state is only affected by per-port power control (Set/ClearPortPower). When cleared, the port is controlled by the global power switch (Set/ClearGlobalPower). If the device is configured to global switching mode (PowerSwitchingMode = 0), 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>DeviceRemovable</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. | | | | | | | | | | | | | |

10.4.7.34 0x0450 OHCI Root Hub Status Register (Default Value:0x0000_0000)

| Offset: 0x0450 | | | | Register Name: O_HcRhStatus |
|----------------|------------|----|-------------|--|
| Bit | Read/Write | | Default/Hex | Description |
| | HCD | HC | | |
| 31 | W | R | 0 | <p>(write)ClearRemoteWakeUpEnable</p> <p>Write a '1' clears DeviceRemoteWakeUpEnable. Write a '0' has no effect.</p> |
| 30:18 | / | / | / | / |
| 17 | R/W | R | 0 | <p>OverCurrentIndicatorChange</p> <p>This bit is set by hardware when a change has occurred to the OverCurrentIndicator field of this register. The HCD clears this bit by writing a '1'. Writing a '0' has no effect.</p> |

| Offset: 0x0450 | | | | Register Name: O_HcRhStatus | | | | |
|----------------|---|-----|-------------|--|---|---|---|---|
| Bit | Read/Write | | Default/Hex | Description | | | | |
| | HCD | HC | | | | | | |
| 16 | R/W | R | 0x0 | <p>(read)LocalPowerStartusChange The Root Hub does not support the local power status features, thus, this bit is always read as '0'.</p> <p>(write)SetGlobalPower In global power mode (PowerSwitchingMode=0), This bit is written to '1' to turn on power to all ports (clear PortPowerStatus). In per-port power mode, it sets PortPowerStatus only on ports whose PortPowerControlMask bit is not set. Writing a '0' has no effect.</p> | | | | |
| 15 | R/W | R | 0x0 | <p>(read)DeviceRemoteWakeupEnable This bit enables a ConnectStatusChange bit as a resume event, causing a USBSUSPEND to USBRESUME state transition and setting the ResumeDetected interrupt.</p> <table border="1"> <tr> <td>0</td><td>ConnectStatusChange is not a remote wakeup event.</td></tr> <tr> <td>1</td><td>ConnectStatusChange is a remote wakeup event.</td></tr> </table> <p>(write)SetRemoteWakeupEnable Writing a '1' sets DeviceRemoveWakeupEnable. Writing a '0' has no effect.</p> | 0 | ConnectStatusChange is not a remote wakeup event. | 1 | ConnectStatusChange is a remote wakeup event. |
| 0 | ConnectStatusChange is not a remote wakeup event. | | | | | | | |
| 1 | ConnectStatusChange is a remote wakeup event. | | | | | | | |
| 14:2 | / | / | / | / | | | | |
| 1 | R | R/W | 0x0 | <p>OverCurrentIndicator This bit reports overcurrent conditions when the global reporting is implemented. When set, an overcurrent condition exists. When cleared, all power operations are normal.</p> <p>If per-port overcurrent protection is implemented this bit is always '0'</p> | | | | |
| 0 | R/W | R | 0x0 | <p>(Read)LocalPowerStatus 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>(Write)ClearGlobalPower When write, this bit is operated as the ClearGlobalPower. In global power mode (PowerSwitchingMode=0), This bit is written to '1' to turn off power to all ports (clear PortPowerStatus). In per-port power mode, it clears PortPowerStatus only on ports whose PortPowerControlMask bit is not set. Writing a '0' has no effect.</p> | | | | |

10.4.7.35 0x0454 OHCI Root Hub Port Status Register (Default Value:0x0000_0100)

| Offset: 0x0454 | | | | Register Name: O_HcRhPortStatus | | | | |
|----------------|---------------------------------------|-----|-------------|--|---|---------------------------------------|---|--------------------------------------|
| Bit | Read/Write | | Default/Hex | Description | | | | |
| | HCD | HC | | | | | | |
| 31:21 | / | / | 0x0 | Reserved | | | | |
| 20 | R/W | R/W | 0x0 | <p>PortResetStatusChange 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.</p> <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 |
| 0 | port reset is not complete | | | | | | | |
| 1 | port reset is complete | | | | | | | |
| 19 | R/W | R/W | 0x0 | <p>PortOverCurrentIndicatorChange This bit is valid only if overcurrent conditions are reported on a per-port basis. This bit is set when Root Hub changes the PortOverCurrentIndicator bit. The HCD writes a '1' to clear this bit. Writing a '0' has no effect.</p> <table border="1"> <tr> <td>0</td><td>no change in PortOverCurrentIndicator</td></tr> <tr> <td>1</td><td>PortOverCurrentIndicator has changed</td></tr> </table> | 0 | no change in PortOverCurrentIndicator | 1 | PortOverCurrentIndicator has changed |
| 0 | no change in PortOverCurrentIndicator | | | | | | | |
| 1 | PortOverCurrentIndicator has changed | | | | | | | |
| 18 | R/W | R/W | 0x0 | <p>PortSuspendStatusChange 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 ResetStatusChange is set.</p> <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 |
| 0 | resume is not completed | | | | | | | |
| 1 | resume completed | | | | | | | |
| 17 | R/W | R/W | 0x0 | <p>PortEnableStatusChange This bit is set when hardware events cause the PortEnableStatus 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.</p> <table border="1"> <tr> <td>0</td><td>no change in PortEnableStatus</td></tr> <tr> <td>1</td><td>change in PortEnableStatus</td></tr> </table> | 0 | no change in PortEnableStatus | 1 | change in PortEnableStatus |
| 0 | no change in PortEnableStatus | | | | | | | |
| 1 | change in PortEnableStatus | | | | | | | |

| Offset: 0x0454 | | | | Register Name: O_HcRhPortStatus | | | | |
|----------------|-------------------------------|-----|-------------|--|---|-------------------------------|---|----------------------------|
| Bit | Read/Write | | Default/Hex | Description | | | | |
| | HCD | HC | | | | | | |
| 16 | R/W | R/W | 0x0 | <p>ConnectStatusChange 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 CurrentConnectStatus is cleared when a SetPortReset, SetPortEnable, or SetPortSuspend 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.</p> <table border="1"> <tr> <td>0</td><td>no change in PortEnableStatus</td></tr> <tr> <td>1</td><td>change in PortEnableStatus</td></tr> </table> <p>Note: If the DeviceRemovable[NDP] bit is set, this bit is set only after a Root Hub reset to inform the system that the device is attached.</p> | 0 | no change in PortEnableStatus | 1 | change in PortEnableStatus |
| 0 | no change in PortEnableStatus | | | | | | | |
| 1 | change in PortEnableStatus | | | | | | | |
| 15:10 | / | / | / | / | | | | |
| 9 | R/W | R/W | - | <p>(read)LowSpeedDeviceAttached 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 CurrentConnectStatus 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)ClearPortPower The HCD clears the PortPowerStatus 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 | | | | | | | |

| Offset: 0x0454 | | | | Register Name: O_HcRhPortStatus | | | | |
|----------------|---------------------------------|-----|-------------|--|---|---------------------------------|---|-----------------------------|
| Bit | Read/Write | | Default/Hex | Description | | | | |
| | HCD | HC | | | | | | |
| 8 | R/W | R/W | 0x1 | <p>(read)PortPowerStatus</p> <p>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 SetPortPower or SetGlobalPower. HCD clears this bit by writing ClearPortPower or ClearGlobalPower. Which power control switches are enabled is determined by PowerSwitchingMode and PortPortControlMask[NumberDownstreamPort]. In global switching mode(PowerSwitchingMode=0), only Set/ClearGlobalPower controls this bit. In per-port power switching (PowerSwitchingMode=1), if the PortPowerControlMask[NDP] bit for the port is set, only Set/ClearPortPower commands are enabled. If the mask is not set, only Set/ClearGlobalPower commands are enabled. When port power is disabled, CurrentConnectStatus, PortEnableStatus, PortSuspendStatus, and PortResetStatus 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)SetPortPower</p> <p>The HCD writes a '1' to set the PortPowerStatus bit. Writing a '0' has no effect.</p> <p>Note: This bit is always reads '1b' if power switching is not supported.</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 | <p>(read)PortResetStatus</p> <p>When this bit is set by a write to SetPortReset, port reset signaling is asserted. When reset is completed, this bit is cleared when PortResetStatusChange is set. This bit cannot be set if CurrentConnectStatus 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)SetPortReset</p> <p>The HCD sets the port reset signaling by writing a '1' to this bit. Writing a '0' has no effect. If CurrentConnectStatus is cleared, this write does not set PortResetStatus, but instead sets ConnectStatusChange. 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 | | | | | | | |

| Offset: 0x0454 | | | | Register Name: O_HcRhPortStatus | | | | |
|----------------|---------------------------------|-----|-------------|--|---|---------------------------|---|---------------------------------|
| Bit | Read/Write | | Default/Hex | Description | | | | |
| | HCD | HC | | | | | | |
| 3 | R/W | R/W | 0x0 | <p>(read)PortOverCurrentIndicator 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)ClearSuspendStatus The HCD writes a '1' to initiate a resume. Writing a '0' has no effect. A resume is initiated only if PortSuspendStatus 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)PortSuspendStatus This bit indicates the port is suspended or in the resume sequence. It is set by a SetSuspendState write and cleared when PortSuspendStatusChange is set at the end of the resume interval. This bit cannot be set if CurrentConnectStatus is cleared. This bit is also cleared when PortResetStatusChange 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)SetPortSuspend The HCD sets the PortSuspendStatus bit by writing a '1' to this bit. Writing a '0' has no effect. If CurrentConnectStatus is cleared, this write does not set PortSuspendStatus; instead it sets ConnectStatusChange. This informs the driver that it attempted to suspend a disconnected port.</p> | 0 | port is not suspended | 1 | port is suspended |
| 0 | port is not suspended | | | | | | | |
| 1 | port is suspended | | | | | | | |

| Offset: 0x0454 | | | | Register Name: O_HcRhPortStatus | | | | |
|----------------|---------------------|-----|-------------|--|---|---------------------|---|------------------|
| Bit | Read/Write | | Default/Hex | Description | | | | |
| | HCD | HC | | | | | | |
| 1 | R/W | R/W | 0x0 | <p>(read)PortEnableStatus 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 PortEnabledStatusChange to be set. HCD sets this bit by writing SetPortEnable and clears it by writing ClearPortEnable. This bit cannot be set when CurrentConnectStatus is cleared. This bit is also set, if not already, at the completion of a port reset when ResetStatusChange is set or port suspend when SuspendStatusChange 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)SetPortEnable The HCD sets PortEnableStatus by writing a '1'. Writing a '0' has no effect. If CurrentConnectStatus is cleared, this write does not set PortEnableStatus, but instead sets ConnectStatusChange. 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)CurrentConnectStatus 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)ClearPortEnable The HCD writes a '1' to clear the PortEnableStatus bit. Writing '0' has no effect. The CurrentConnectStatus is not affected by any write.</p> <p>Note: This bit is always read '1' when the attached device is nonremovable (DviceRemoveable[NumberDownstreamPort]).</p> | 0 | No device connected | 1 | Device connected |
| 0 | No device connected | | | | | | | |
| 1 | Device connected | | | | | | | |

10.4.7.36 0x0800 HCI Interface Register (Default Value:0x0000_0000)

| Offset: 0x0800 | | | Register Name: USB_CTRL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31 | R/W | 0x0 | USB Standby Clock Sel 0: normal mode USB clock as usual 1: standby mode USB clock switch to RC 16M clock |
| 30:29 | / | / | 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 than real time 0: Normal mode, the counters will count full time |
| 24 | R/W | 0 | Simulation mode 1: Set PHY in a non-driving mode so the EHCI can detect device connection, this is used only for simulation 0: No effect |
| 23:21 | / | / | / |
| 20 | R/W | 0 | EHCI HS force Set 1 to this field force the EHCI enter the high speed mode during bus reset. This field only valid when the bit 1 is set. |
| 19 | / | / | / |
| 18 | R/W | 0 | 1: within 2us 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 from 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 |

| Offset: 0x0800 | | | Register Name: USB_CTRL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 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 11:9 is enabled. |
| 7:4 | / | / | / |
| 3 | R/W | 0x0 | RC Clock Gating 0x0: clock gated 0x1: clock ungated |
| 2 | R/W | 0x0 | RC Generation Enable 0x0: disable 0x1: enable |
| 1 | / | / | / |
| 0 | R/W | 0 | ULPI bypass enable. 1: Enable UTMI interface, disable ULPI interface 0: Enable ULPI interface, disable UTMI interface |

10.4.7.37 0x0808 HCI Control 3 Register (Default Value:0x0000_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 dected. 1: Linestate change dected. 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 |

10.4.7.38 0x0810 PHY Control Register (Default Value: 0x0000_0008)

| Offset: 0x0810 | | | Register Name: PHY_CTRL(Default Value: 0x0000_0008) |
|----------------|------------|-------------|---|
| 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 | 1 | SIDDQ 1: Write 1 to disable PHY. 0: Write 0 to enable PHY. |
| 2 | / | / | / |
| 1 | R/W | 0 | VC_EN 0: Write 0 to disable PHY VC bus 1: Write 1 to enable PHY VC bus. Note: Only 1 VC bus of all the USB controllers could be enabled at the same time. |
| 0 | R/W | 0 | VC_CLK |

10.4.7.39 0x0824 PHY Status Register(USB1) (Default Value: 0x0000_0000)

| Offset: 0x0824 | | | Register Name: PHY_STA |
|----------------|------------|-------------|------------------------|
| 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 |

10.4.7.40 0x0828 HCI SIE Port Disable Control Register (Default Value:0x0000_0003)

| Offset: 0x0828 | | | Register Name: USB_SPDCR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:17 | / | / | / |
| 16 | R/W | 0 | SEO 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 2us, setting this bit to 1, which is cooperated with SS_UTMI_BACKWARD_ENB_I. |
| 3:2 | / | / | / |

| Offset: 0x0828 | | | Register Name: USB_SPDCR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 1:0 | R/W | 0x3 | <p>Port Disable Control</p> <p>00: Port Disable when no-se0 detect before SOF</p> <p>01: Port Disable when no-se0 detect before SOF</p> <p>10: No Port Disable when no-se0 detect before SOF</p> <p>11: Port Disable when no-se0 3 time detect before SOF during 8 Frames</p> |



10.5 GPIO

10.5.1 Overview

The general purpose input/output (GPIO) is one of the blocks controlling the chip multiplexing pins. The V853 supports 8 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.

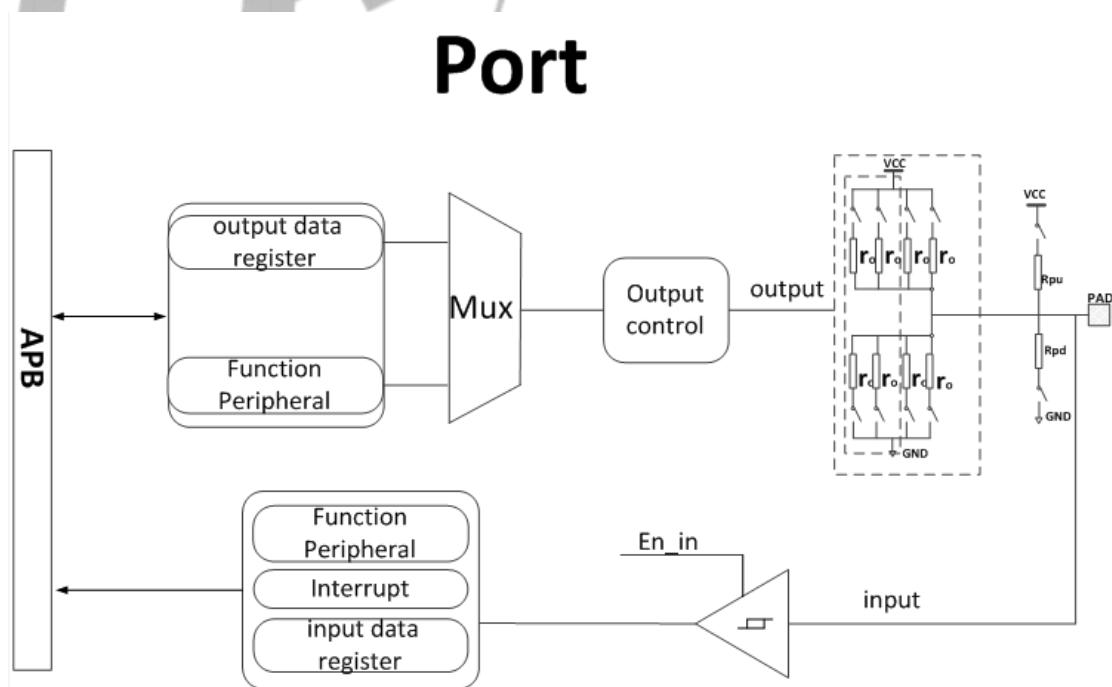
The port controller has the following features:

- 8 ports (PA, PC, PD, PE, PF, PG, PH, PI)
- Software control for each signal pin
- Each GPIO peripheral can produce an interrupt
- Pull_up/Pull_down/no_Pull register control
- Controls the direction of every signal
- 4 drive strengths in each operating mode
- Up to 111 interrupts
- Configurable interrupt edges

10.5.2 Block Diagram

The following figure shows the block diagram of the GPIO.

Figure 10-29 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.

10.5.3 Functional Description

10.5.3.1 Multi-function Port

The V853/V853S includes 111 multi-functional input/output port pins. There are 8 ports as listed below.

Table 10-14 Multi-function Port

| Port Name | Number of Pins | Input Driver | Output Driver | Multiplex Pins | Typical Power Supply |
|-----------|----------------|--------------|---------------|---|----------------------|
| PA | 22 | Schmitt | CMOS | CSI/PWM/UART/LCD/PA-EINT | 3.3V/1.8V |
| PC | 12 | Schmitt | CMOS | SDC/SPI/PWM/TWI/PC-EINT | 3.3V/1.8V |
| PD | 23 | Schmitt | CMOS | LCD/SPI/DSI/DMIC/RMII/TWI/UART/PD-EINT | 3.3V/1.8V |
| PE | 18 | Schmitt | CMOS | CSI/I2S/LCD/PWM/RGMII/RMII/SDC/TWI/UART/WIEGAND /PE-EINT | 3.3V/1.8V |
| PF | 7 | Schmitt | CMOS | SDC/SPI/JTAG//PF-EINT | 3.3V/1.8V |
| PG | 8 | Schmitt | CMOS | LCD/SDC/TWI/UART/PG-EINT | 3.3V/1.8V |
| PH | 16 | Schmitt | CMOS | DMIC/EPHY/I2S/JTAG/SPI/PWM/RMII/TWI/UART/WIEGAN D/PH-EINT | 3.3V |
| PI | 5 | Schmitt | CMOS | CSI/LCD/SPI/TWI/PI-EINT | 3.3V/1.8V |

10.5.3.2 GPIO Multiplex Function

The following tables show the multiplex function pins of the V853/V853S.



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

Table 10-15 PA Multiplex Function

| GPIO Port | Function 2 | Function 3 | Function 4 | Function 5 | Function 6 | Function 7 | Function 8 | Function 14 |
|-----------|-----------------------------|------------|----------------|-------------|-------------|------------|------------|-------------|
| PA0 | MIPIA_CSI_CKOP | NCSI_D8 | | | | | | PA_EINT0 |
| PA1 | MIPIA_CSI_CKON | NCSI_D9 | | | | | | PA_EINT1 |
| PA2 | MIPIA_CSI_D1N | NCSI_D10 | | | | | | PA_EINT2 |
| PA3 | MIPIA_CSI_D1P | NCSI_D11 | | | | | | PA_EINT3 |
| PA4 | MIPIA_CSI_D0P | NCSI_D12 | | | | | | PA_EINT4 |
| PA5 | MIPIA_CSI_D0N | NCSI_D13 | | | | | | PA_EINT5 |
| PA6 | MIPIB_CSI_D0N /MIPIA-CI-D2N | NCSI_D14 | TWI1_SCK | PWM0 | | | | PA_EINT6 |
| PA7 | MIPIB_CSI_D0P /MIPIA-CI-D2P | NCSI_D15 | TWI1_SDA | PWM1 | | | | PA_EINT7 |
| PA8 | MIPIB_CSI_D1N /MIPIA-CI-D3N | TWI4_SCK | TWI3_SCK | PWM2 | UART2_TX | | | PA_EINT8 |
| PA9 | MIPIB_CSI_D1P /MIPIA-CI-D3P | TWI4_SDA | TWI3_SDA | PWM3 | UART2_RX | | | PA_EINT9 |
| PA10 | MIPIB_CSI_CKON | NCSI_HSYNC | MIPI_CSI_MCLK0 | TWI0_SCK | CLK_FANOUT0 | | | PA_EINT10 |
| PA11 | MIPIB_CSI_CKOP | NCSI_VSYNC | MIPI_CSI_MCLK1 | TWI0_SDA | CLK_FANOUT1 | | | PA_EINT11 |
| PA12 | | NCSI_D0 | MIPI_CSI_MCLK0 | UART0_TX | | | | PA_EINT12 |
| PA13 | | NCSI_D1 | MIPI_CSI_MCLK1 | UART0_RX | | | | PA_EINT13 |
| PA14 | | NCSI_D2 | TWI1_SCK | CLK_FANOUT0 | | | | PA_EINT14 |
| PA15 | | NCSI_D3 | TWI1_SDA | CLK_FANOUT1 | | | | PA_EINT15 |
| PA16 | | NCSI_D4 | TWI0_SCK | UART3_TX | | | | PA_EINT16 |
| PA17 | | NCSI_D5 | TWI0_SDA | UART3_RX | | | | PA_EINT17 |
| PA18 | | NCSI_D6 | WIEGAND_D0 | UART3_RTS | | | | PA_EINT18 |
| PA19 | | NCSI_D7 | WIEGAND_D1 | UART3_CTS | | | | PA_EINT19 |
| PA20 | | NCSI_MCLK | CSI_SM_VS | TCON_TRIG | | | | PA_EINT20 |
| PA21 | | NCSI_PCLK | | | | | | PA_EINT21 |

Table 10-16 PC Multiplex Function

| GPIO Port | Function 2 | Function 3 | Function 4 | Function 5 | Function 6 | Function 7 | Function 8 | Function 14 |
|-----------|------------|------------|------------|------------|------------|------------|------------|-------------|
| PC0 | | SDC2_CLK | SPI0_CLK | | | | | PC_EINT0 |
| PC1 | | SDC2_CMD | SPI0_CS0 | | | | | PC_EINT1 |
| PC2 | | SDC2_D2 | SPI0_MOSI | BOOT_SEL0 | | | | PC_EINT2 |
| PC3 | | SDC2_D1 | SPI0_MISO | BOOT_SEL1 | | | | PC_EINT3 |
| PC4 | | SDC2_D0 | SPI0_WP | PWM4 | TWI1_SCK | | | PC_EINT4 |
| PC5 | | SDC2_D3 | SPI0_HOLD | PWM5 | TWI1_SDA | | | PC_EINT5 |
| PC6 | | SDC2_D4 | SPI0_CS1 | | | | | PC_EINT6 |
| PC7 | | SDC2_D5 | | | | | | PC_EINT7 |
| PC8 | | SDC2_D6 | | | | | | PC_EINT8 |
| PC9 | | SDC2_D7 | | | | | | PC_EINT9 |
| PC10 | | SDC2_DS | | | | | | PC_EINT10 |
| PC11 | | SDC2_RST | | | | | | PC_EINT11 |

Table 10-17 PD Multiplex Function

| GPIO Port | Function 2 | Function 3 | Function 4 | Function 5 | Function 6 | Function 7 | Function 8 | Function 14 |
|-----------|------------|------------|-------------|------------|--|-------------|------------|-------------|
| PD0 | LCD_D2 | | | | | | | PD_EINT0 |
| PD1 | LCD_D3 | PWM0 | RMII_RXD1 | DSI_D0N | SPI1_CS0/DBI_CSX | RMII_TXD0 | | PD_EINT1 |
| PD2 | LCD_D4 | PWM1 | RMII_RXD0 | DSI_D0P | SPI1_CLK/DBI_SCLK | RMII_TXD1 | | PD_EINT2 |
| PD3 | LCD_D5 | PWM2 | RMII_CRS_DV | DSI_D1N | SPI1_MOSI/DBI_SDO | RMII_RXER | | PD_EINT3 |
| PD4 | LCD_D6 | PWM3 | RMII_RXER | DSI_D1P | SPI1_MISO/ DBI_SDI/ DBI_TE/DBI_DCX | RMII_CRS_DV | | PD_EINT4 |
| PD5 | LCD_D7 | PWM4 | RMII_TXD1 | DSI_CKN | SPI1_HOLD/DBI_DCX/ DBI_WRX | RMII_RXD1 | | PD_EINT5 |
| PD6 | LCD_D10 | PWM5 | RMII_TXD0 | DSI_CKP | SPI1_WP/DBI_TE | RMII_RXD0 | | PD_EINT6 |
| PD7 | LCD_D11 | PWM6 | RMII_TXCK | DSI_D2N | SPI1_CS1 | MDC | | PD_EINT7 |
| PD8 | LCD_D12 | PWM7 | RMII_TXEN | | | | | PD_EINT8 |
| PD9 | LCD_D13 | PWM8 | | DSI_D2P | | MDIO | | PD_EINT9 |
| PD10 | LCD_D14 | I2S1_MCLK | | DSI_D3N | | RMII_TXEN | | PD_EINT10 |
| PD11 | LCD_D15 | I2S1_BCLK | | DSI_D3P | | RMII_TXCK | | PD_EINT11 |
| PD12 | LCD_D18 | I2S1_LRCK | | DMIC_DATA3 | | PWM11 | | PD_EINT12 |
| PD13 | LCD_D19 | I2S1_DOUT0 | | DMIC_DATA2 | | | | PD_EINT13 |
| PD14 | LCD_D20 | I2S1_DOUT1 | I2S1_DIN1 | DMIC_DATA1 | | | | PD_EINT14 |
| PD15 | LCD_D21 | I2S1_DOUT2 | I2S1_DIN2 | DMIC_DATA0 | | | | PD_EINT15 |
| PD16 | LCD_D22 | I2S1_DOUT3 | I2S1_DIN3 | DMIC_CLK | | | | PD_EINT16 |
| PD17 | LCD_D23 | I2S1_DIN0 | | | | | | PD_EINT17 |
| PD18 | LCD_CLK | | EPHY_25M | SPI2_CLK | TWI3_SCK | UART2_TX | | PD_EINT18 |
| PD19 | LCD_DE | PWM9 | TCON_TRIG | SPI2_MOSI | TWI3_SDA | UART2_RX | | PD_EINT19 |
| PD20 | LCD_HSYNC | PWM10 | MDC | SPI2_MISO | TWI2_SCK | UART2_RTS | | PD_EINT20 |
| PD21 | LCD_VSYNC | | MDIO | SPI2_CS0 | TWI2_SDA | UART2_CTS | | PD_EINT21 |
| PD22 | PWM_9 | | | | | | | PD_EINT22 |

Table 10-18 PE Multiplex Function

| GPIO Port | Function 2 | Function 3 | Function 4 | Function 5 | Function 6 | Function 7 | Function 8 | Function 14 |
|-----------|------------|-------------------------|------------|----------------|------------|------------|------------|-------------|
| PE0 | NCSI_PCLK | RGMII_RXD1/RMII_RXD1 | I2S1_MCLK | PWM0 | SDC1_CLK | UART3_TX | TWI3_SCK | PE_EINT0 |
| PE1 | NCSI_MCLK | RGMII_TXCK/RMII_TXCK | I2S1_BCLK | PWM1 | SDC1_CMD | UART3_RX | TWI3_SDA | PE_EINT1 |
| PE2 | NCSI_HSYNC | RGMII_RXCTL/RMII_CRS_DV | I2S1_LRCK | PWM2 | SDC1_D0 | UART3_CTS | TWI1_SCK | PE_EINT2 |
| PE3 | NCSI_VSYNC | RGMII_RXD0/RMII_RXD0 | I2S1_DIN0 | PWM3 | SDC1_D1 | UART3_RTS | TWI1_SDA | PE_EINT3 |
| PE4 | NCSI_D0 | RGMII_RXD0/RMII_RXD0 | I2S1_DOUT0 | PWM4 | SDC1_D2 | TWI3_SCK | TWI0_SCK | PE_EINT4 |
| PE5 | NCSI_D1 | RGMII_RXD1/RMII_RXD1 | | PWM5 | SDC1_D3 | TWI3_SDA | TWI0_SDA | PE_EINT5 |
| PE6 | NCSI_D2 | RGMII_TXCTL/RMII_TXEN | LCD_D2 | PWM6 | UART1_TX | | TWI4_SCK | PE_EINT6 |
| PE7 | NCSI_D3 | RGMII_CLKIN/RMII_RXER | LCD_D15 | PWM7 | UART1_RX | I2S1_DOUT0 | TWI4_SDA | PE_EINT7 |
| PE8 | NCSI_D4 | MDC | LCD_D18 | PWM8 | WIEGAND_D0 | I2S1_DIN0 | TWI1_SCK | PE_EINT8 |
| PE9 | NCSI_D5 | MDIO | LCD_D19 | PWM9 | WIEGAND_D1 | I2S1_LRCK | TWI1_SDA | PE_EINT9 |
| PE10 | NCSI_D6 | EPHY_25M | LCD_D20 | PWM10 | UART2_RTS | I2S1_BCLK | WIEGAND_D0 | PE_EINT10 |
| PE11 | NCSI_D7 | RGMII_RXD3 | LCD_D21 | CSI_SM_VS | UART2_CTS | I2S1_MCLK | WIEGAND_D1 | PE_EINT11 |
| PE12 | NCSI_D8 | RGMII_RXD2 | LCD_D22 | MIPI_CSI_MCLK0 | UART2_TX | UART3_TX | | PE_EINT12 |
| PE13 | NCSI_D9 | RGMII_RXCK | LCD_D23 | MIPI_CSI_MCLK1 | UART2_RX | UART3_RX | | PE_EINT13 |
| PE14 | NCSI_D10 | RGMII_RXD3 | | | | | | PE_EINT14 |
| PE15 | NCSI_D11 | RGMII_RXD2 | | | | | | PE_EINT15 |

| GPIO Port | Function 2 | Function 3 | Function 4 | Function 5 | Function 6 | Function 7 | Function 8 | Function 14 |
|-----------|------------|------------|------------|------------|------------|------------|------------|-------------|
| PE16 | TWI0_SCK | TWI4_SCK | | | | | | PE_EINT16 |
| PE17 | TWI0_SDA | TWI4_SDA | | | | | | PE_EINT17 |

Table 10-19 PF Multiplex Function

| GPIO Port | Function 2 | Function 3 | Function 4 | Function 5 | Function 6 | Function 7 | Function 8 | Function 14 |
|-----------|------------|------------|------------|------------|------------|------------|------------|-------------|
| PF0 | SDC0_D1 | JTAG_MS | SPI0_CLK | SPI2_CLK | R_JTAG_MS | CPU_BIST0 | | PF_EINT0 |
| PF1 | SDC0_D0 | JTAG_DI | SPI0_MOSI | SPI2_MOSI | R_JTAG_DI | CPU_BIST1 | | PF_EINT1 |
| PF2 | SDC0_CLK | UART0_TX | SPI0_MISO | SPI2_MISO | | | | PF_EINT2 |
| PF3 | SDC0_CMD | JTAG_DO | SPI0_CS0 | SPI2_CS0 | R_JTAG_DO | | | PF_EINT3 |
| PF4 | SDC0_D3 | UART0_RX | SPI0_CS1 | SPI2_CS1 | | | | PF_EINT4 |
| PF5 | SDC0_D2 | JTAG_CK | | | R_JTAG_CK | | | PF_EINT5 |
| PF6 | DBG_CLK | | | | | | | PF_EINT6 |

Table 10-20 PG Multiplex Function

| GPIO Port | Function 2 | Function 3 | Function 4 | Function 5 | Function 6 | Function 7 | Function 8 | Function 14 |
|-----------|------------|-------------|------------|------------|------------|------------|------------|-------------|
| PG0 | SDC1_CLK | LCD_D0 | UART3_TX | | | | | PG_EINT0 |
| PG1 | SDC1_CMD | LCD_D1 | UART3_RX | | | | | PG_EINT1 |
| PG2 | SDC1_D0 | LCD_D8 | UART3_CTS | | | | | PG_EINT2 |
| PG3 | SDC1_D1 | LCD_D9 | UART3_RTS | | | | | PG_EINT3 |
| PG4 | SDC1_D2 | LCD_D16 | UART1_RTS | | | | | PG_EINT4 |
| PG5 | SDC1_D3 | LCD_D17 | UART1_CTS | | | | | PG_EINT5 |
| PG6 | TWI4_SCK | CLK_FANOUT0 | UART1_TX | | | | | PG_EINT6 |
| PG7 | TWI4_SDA | CLK_FANOUT1 | UART1_RX | | | | | PG_EINT7 |

Table 10-21 PH Multiplex Function

| GPIO Port | Function 2 | Function 3 | Function 4 | Function 5 | Function 6 | Function 7 | Function 8 | Function 14 |
|-----------|-------------|-------------|------------|-------------|------------|-------------|------------|-------------|
| PH0 | PWM0 | I2S0_MCLK | SPI1_CLK | UART3_TX | DMIC_DATA3 | | | PH_EINT0 |
| PH1 | PWM1 | I2S0_BCLK | SPI1_MOSI | UART3_RX | DMIC_DATA2 | | | PH_EINT1 |
| PH2 | PWM2 | I2S0_LRCK | SPI1_MISO | UART3_CTS | DMIC_DATA1 | | | PH_EINT2 |
| PH3 | PWM3 | I2S0_DOUT0 | SPI1_CS0 | UART3_RTS | DMIC_DATA0 | | | PH_EINT3 |
| PH4 | PWM4 | I2S0_DIN0 | SPI1_CS1 | CLK_FANOUT2 | DMIC_CLK | | | PH_EINT4 |
| PH5 | PWM5 | RMII_RXD1 | TWI2_SCK | UART2_TX | | | | PH_EINT5 |
| PH6 | PWM6 | RMII_RXD0 | TWI2_SDA | UART2_RX | I2S1_MCLK | DMIC_DATA3 | | PH_EINT6 |
| PH7 | PWM7 | RMII_CRS_DV | UART0_TX | UART2_RTS | I2S1_BCLK | DMIC_DATA2 | | PH_EINT7 |
| PH8 | PWM8 | RMII_RXER | UART0_RX | UART2_CTS | I2S1_LRCK | DMIC_DATA1 | | PH_EINT8 |
| PH9 | PWM9 | RMII_TXD1 | TWI3_SCK | UART0_TX | I2S1_DIN0 | DMIC_DATA0 | | PH_EINT9 |
| PH10 | PWM10 | RMII_TXD0 | TWI3_SDA | UART0_RX | I2S1_DOUT0 | DMIC_CLK | | PH_EINT10 |
| PH11 | JTAG_MS | RMII_TXCK | R_JTAG_MS | TWI2_SCK | SPI3_CLK | CLK_FANOUT0 | PWM4 | PH_EINT11 |
| PH12 | JTAG_CK | RMII_TXEN | R_JTAG_CK | TWI2_SDA | SPI3_MOSI | CLK_FANOUT1 | PWM5 | PH_EINT12 |
| PH13 | JTAG_DO | MDC | R_JTAG_DO | TWI3_SCK | SPI3_MISO | WIEGAND_D0 | PWM6 | PH_EINT13 |
| PH14 | JTAG_DI | MDIO | R_JTAG_DI | TWI3_SDA | SPI3_CS0 | WIEGAND_D1 | PWM7 | PH_EINT14 |
| PH15 | CLK_FANOUT2 | EPHY_25M | | | SPI3_CS1 | | PWM8 | PH_EINT15 |

Table 10-22 PI Multiplex Function

| GPIO Port | Function 2 | Function 3 | Function 4 | Function 5 | Function 6 | Function 7 | Function 8 | Function 14 |
|-----------|----------------|------------|------------|------------|------------|-------------|------------|-------------|
| PI0 | MIPI_CSI_MCLK0 | | | | | | | PI_EINT0 |
| PI1 | CSI_SM_HS | | SPI2_CLK | TWI1_SCK | TWI4_SCK | | | PI_EINT1 |
| PI2 | CSI_SM_VS | TCON_TRIG | SPI2_MOSI | TWI1_SDA | TWI4_SDA | | | PI_EINT2 |
| PI3 | | | SPI2_MISO | TWI0_SCK | TWI3_SCK | CLK_FANOUT0 | | PI_EINT3 |
| PI4 | | | SPI2_CS0 | TWI0_SDA | TWI3_SDA | CLK_FANOUT1 | | PI_EINT4 |

10.5.3.3 Port Function

The Port Controller supports 8 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 10-23 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

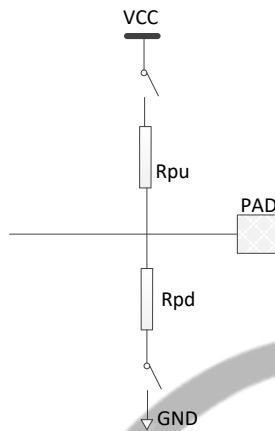
X: Select configuration according to the actual situation

N: Forbid to configure

10.5.3.4 Pull Up/Down and High-Impedance Logic

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

Figure 10-30 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.

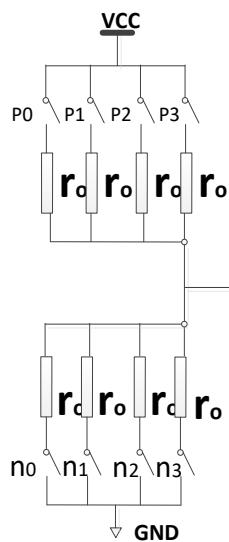
The pull-up/down of each IO is weak pull-up/down.

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

10.5.3.5 Buffer Strength

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

Figure 10-31 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 r0. When the buffer strength is set to 1, only the p0 and p1 is on, the output impedance is equivalent to two r0 in parallel, the impedance value is r0/2. When the buffer strength is 2, only the p0, p1, and p2 is on, the output impedance is equivalent to three r0 in parallel, the impedance value is r0/3. When buffer strength is 3, the p0, p1, p2, and p3 is on, the output impedance is equivalent to four r0 in parallel, the impedance value is r0/4.

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 r0. When the buffer strength is set to 1, only the n0 and n1 is on, the output impedance is equivalent to two r0 in parallel, the impedance value is r0/2. When the buffer strength is 2, only the n0, n1, and n2 is on, the output impedance is equivalent to three r0 in parallel, the impedance value is r0/3. When the buffer strength is 3, the n0, n1, n2, and n3 is on, the output impedance is equivalent to four r0 in parallel, the impedance value is r0/4.

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

10.5.3.6 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.

10.5.4 Register List

| Module Name | Base Address |
|-------------|--------------|
| GPIO | 0x02000000 |

| Register Name | Offset | Description |
|---------------|--------|-----------------------------|
| PA_CFG0 | 0x0000 | PA Configure Register 0 |
| PA_CFG1 | 0x0004 | PA Configure Register 1 |
| PA_CFG2 | 0x0008 | PA Configure Register 2 |
| PA_DAT | 0x0010 | PA Data Register |
| PA_DRV0 | 0x0014 | PA Multi_Driving Register 0 |
| PA_DRV1 | 0x0018 | PA Multi_Driving Register 1 |
| PA_DRV2 | 0x001C | PA Multi_Driving Register 2 |
| PA_PUL0 | 0x0024 | PA Pull Register 0 |
| PA_PUL1 | 0x0028 | PA Pull Register 1 |
| PC_CFG0 | 0x0060 | PC Configure Register 0 |
| PC_CFG1 | 0x0064 | PC Configure Register 1 |
| PC_DAT | 0x0070 | PC Data Register |
| PC_DRV0 | 0x0074 | PC Multi_Driving Register 0 |
| PC_DRV1 | 0x0078 | PC Multi_Driving Register 1 |
| PC_PUL | 0x0084 | PC Pull Register |
| 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 |

| Register Name | Offset | Description |
|---------------|--------|--|
| PD_DRV1 | 0x00A8 | PD Multi_Driving Register 1 |
| PD_DRV2 | 0x00AC | PD Multi_Driving Register 2 |
| PD_PUL0 | 0x00B4 | PD Pull Register 0 |
| PD_PUL1 | 0x00B8 | PD Pull Register 1 |
| PE_CFG0 | 0x00C0 | PE Configure Register 0 |
| PE_CFG1 | 0x00C4 | PE Configure Register 1 |
| PE_CFG2 | 0x00C8 | PE Configure Register 2 |
| PE_DAT | 0x00D0 | PE Data Register |
| PE_DRV0 | 0x00D4 | PE Multi_Driving Register 0 |
| PE_DRV1 | 0x00D8 | PE Multi_Driving Register 1 |
| PE_DRV2 | 0x00DC | PE Multi_Driving Register 2 |
| PE_PUL0 | 0x00E4 | PE Pull Register 0 |
| PE_PUL1 | 0x00E8 | PE Pull Register 1 |
| PF_CFG | 0x00F0 | PF Configure Register |
| PF_DAT | 0x0100 | PF Data Register |
| PF_DRV | 0x0104 | PF Multi_Driving Register |
| PF_PUL | 0x0114 | PF Pull Register |
| PG_CFG | 0x0120 | PG Configure Register |
| PG_DAT | 0x0130 | PG Data Register |
| PG_DRV | 0x0134 | PG Multi_Driving Register |
| PG_PUL | 0x0144 | PG Pull Register |
| PH_CFG0 | 0x0150 | PH Configure Register 0 |
| PH_CFG1 | 0x0154 | PH Configure Register 1 |
| PH_DAT | 0x0160 | PH Data Register |
| PH_DRV0 | 0x0164 | PH Multi_Driving Register 0 |
| PH_DRV1 | 0x0168 | PH Multi_Driving Register 1 |
| PH_PUL | 0x0174 | PH Pull Register |
| PI_CFG | 0x0180 | PI Configure Register |
| PI_DAT | 0x0190 | PI Data Register |
| PI_DRV | 0x0194 | PI Multi_Driving Register |
| PI_PUL | 0x01A4 | PI Pull Register |
| PA_INT_CFG0 | 0x0200 | PA External Interrupt Configure Register 0 |
| PA_INT_CFG1 | 0x0204 | PA External Interrupt Configure Register 1 |
| PA_INT_CFG2 | 0x0208 | PA External Interrupt Configure Register 2 |
| PA_INT_CTL | 0x0210 | PA External Interrupt Control Register |
| PA_INT_STA | 0x0214 | PA External Interrupt Status Register |
| PA_INT_DEB | 0x0218 | PA External Debounce Configure Register |
| PC_INT_CFG0 | 0x0240 | PC External Interrupt Configure Register 0 |
| PC_INT_CFG1 | 0x0244 | PC External Interrupt Configure Register 1 |
| PC_INT_CTL | 0x0250 | PC External Interrupt Control Register |
| PC_INT_STA | 0x0254 | PC External Interrupt Status Register |
| PC_INT_DEB | 0x0258 | PC External Debounce Configure Register |

| Register Name | Offset | Description |
|---------------------|--------|--|
| PD_INT_CFG0 | 0x0260 | PD External Interrupt Configure Register 0 |
| PD_INT_CFG1 | 0x0264 | PD External Interrupt Configure Register 1 |
| PD_INT_CFG2 | 0x0268 | PD External Interrupt Configure Register 2 |
| PD_INT_CTL | 0x0270 | PD External Interrupt Control Register |
| PD_INT_STA | 0x0274 | PD External Interrupt Status Register |
| PD_INT_DEB | 0x0278 | PD External Debounce Configure Register |
| PE_INT_CFG0 | 0x0280 | PE External Interrupt Configure Register 0 |
| PE_INT_CFG1 | 0x0284 | PE External Interrupt Configure Register 1 |
| PE_INT_CFG2 | 0x0288 | PE External Interrupt Configure Register 2 |
| PE_INT_CFG3 | 0x028C | PE External Interrupt Configure Register 3 |
| PE_INT_CTL | 0x0290 | PE External Interrupt Control Register |
| PE_INT_STA | 0x0294 | PE External Interrupt Status Register |
| PE_INT_DEB | 0x0298 | PE External Debounce Configure Register |
| PF_INT_CFG | 0x02A0 | PF External Interrupt Configure Register |
| PF_INT_CTL | 0x02B0 | PF External Interrupt Control Register |
| PF_INT_STA | 0x02B4 | PF External Interrupt Status Register |
| PF_INT_DEB | 0x02B8 | PF External Debounce Configure Register |
| PG_INT_CFG0 | 0x02C0 | PG External Interrupt Configure Register 0 |
| PG_INT_CTL | 0x02D0 | PG External Interrupt Control Register |
| PG_INT_STA | 0x02D4 | PG External Interrupt Status Register |
| PG_INT_DEB | 0x02D8 | PG External Debounce Configure Register |
| PH_INT_CFG0 | 0x02E0 | PH External Interrupt Configure Register 0 |
| PH_INT_CFG1 | 0x02E4 | PH External Interrupt Configure Register 1 |
| PH_INT_CTL | 0x02F0 | PH External Interrupt Control Register |
| PH_INT_STA | 0x02F4 | PH External Interrupt Status Register |
| PH_INT_DEB | 0x02F8 | PH External Debounce Configure Register |
| PI_INT_CFG | 0x0300 | PI External Interrupt Configure Register |
| PI_INT_CTL | 0x0310 | PI External Interrupt Control Register |
| PI_INT_STA | 0x0314 | PI External Interrupt Status Register |
| PI_INT_DEB | 0x0318 | PI External Debounce Configure 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 |
| PIO_POW_VAL_SET_CTL | 0x0350 | PIO Group Power Voltage Select Control Register |

10.5.5 Register Description

10.5.5.1 0x0000 PA Configure Register 0 (Default Value: 0xFFFF_FFFF)

| Offset: 0x0000 | | | Register Name: PA_CFG0 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | R/W | 0xF | <p>PA7_SELECT PA7 Select. 0000: Input 0001: Output 0010: MIPIB_CSI_D0P /MIPIA-CSI-D2P 0011: NCSI_D15 0100: TWI1_SDA 0101: PWM1 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT7 1111: IO Disable</p> |
| 27:24 | R/W | 0xF | <p>PA6_SELECT PA6 Select. 0000: Input 0001: Output 0010: MIPIB_CSI_D0N /MIPIA-CSI-D2N 0011: NCSI_D14 0100: TWI1_SCK 0101: PWM0 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT6 1111: IO Disable</p> |
| 23:20 | R/W | 0xF | <p>PA5_SELECT PA5 Select. 0000: Input 0001: Output 0010: MIPIA_CSI_D0N 0011: NCSI_D13 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT5 1111: IO Disable</p> |

| Offset: 0x0000 | | | Register Name: PA_CFG0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 19:16 | R/W | 0xF | PA4_SELECT PA4 Select. 0000: Input 0001: Output 0010: MIPIA_CSI_D0P 0011: NCSI_D12 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT4 1111: IO Disable |
| 15:12 | R/W | 0xF | PA3_SELECT PA3 Select. 0000: Input 0001: Output 0010: MIPIA_CSI_D1P 0011: NCSI_D11 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT3 1111: IO Disable |
| 11:8 | R/W | 0xF | PA2_SELECT PA2 Select. 0000: Input 0001: Output 0010: MIPIA_CSI_D1N 0011: NCSI_D10 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT2 1111: IO Disable |
| 7:4 | R/W | 0xF | PA1_SELECT PA1 Select. 0000: Input 0001: Output 0010: MIPIA_CSI_CK0N 0011: NCSI_D9 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT1 1111: IO Disable |

| Offset: 0x0000 | | | Register Name: PA_CFG0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 3:0 | R/W | 0xF | PA0_SELECT PA0 Select. 0000: Input 0001: Output 0010: MIPIA_CSI_CKOP 0011: NCSI_D8 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT0 1111: IO Disable |

10.5.5.2 0x0004 PA Configure Register 1 (Default Value: 0xFFFF_FFFF)

| Offset: 0x0004 | | | Register Name: PA_CFG1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | R/W | 0xF | PA15_SELECT PA15 Select. 0000: Input 0001: Output 0010: Reserved 0011: NCSI_D3 0100: TWI1_SDA 0101: CLK_FANOUT1 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT15 1111: IO Disable |
| 27:24 | R/W | 0xF | PA14_SELECT PA14 Select. 0000: Input 0001: Output 0010: Reserved 0011: NCSI_D2 0100: TWI1_SCK 0101: CLK_FANOUT0 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT14 1111: IO Disable |

| Offset: 0x0004 | | | Register Name: PA_CFG1 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 23:20 | R/W | 0xF | PA13_SELECT PA13 Select. 0000: Input 0001: Output 0010: Reserved 0011: NCSI_D1 0100: MIPI_CSI_MCLK1 0101: UART0_RX 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT13 1111: IO Disable |
| 19:16 | R/W | 0xF | PA12_SELECT PA12 Select. 0000: Input 0001: Output 0010: Reserved 0011: NCSI_D0 0100: MIPI_CSI_MCLK0 0101: UART0_TX 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT12 1111: IO Disable |
| 15:12 | R/W | 0xF | PA11_SELECT PA11 Select. 0000: Input 0001: Output 0010: MIPIB_CSI_CK0P 0011: NCSI_VSYNC 0100: MIPI_CSI_MCLK1 0101: TWI0_SDA 0110: CLK_FANOUT1 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT11 1111: IO Disable |
| 11:8 | R/W | 0xF | PA10_SELECT PA10 Select. 0000: Input 0001: Output 0010: MIPIB_CSI_CK0N 0011: NCSI_HSYNC 0100: MIPI_CSI_MCLK0 0101: TWI0_SCK 0110: CLK_FANOUT0 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT10 1111: IO Disable |

| Offset: 0x0004 | | | Register Name: PA_CFG1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 7:4 | R/W | 0xF | PA9_SELECT PA9 Select. 0000: Input 0001: Output 0010: MIPIB_CSI_D1P /MIPIA-CSI-D3P 0011: TWI4_SDA 0100: TWI3_SDA 0101: PWM3 0110: UART2_RX 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT9 1111: IO Disable |
| 3:0 | R/W | 0xF | PA8_SELECT PA8 Select. 0000: Input 0001: Output 0010: MIPIB_CSI_D1N /MIPIA-CSI-D3N 0011: TWI4_SCK 0100: TWI3_SCK 0101: PWM2 0110: UART2_TX 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT8 1111: IO Disable |

10.5.5.3 0x0008 PA Configure Register 2 (Default Value: 0x00FF_FFFF)

| Offset: 0x0008 | | | Register Name: PA_CFG2 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:24 | / | / | / |
| 23:20 | R/W | 0xF | PA21_SELECT PA21 Select. 0000: Input 0001: Output 0010: Reserved 0011: NCSI_PCLK 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT21 1111: IO Disable |

| Offset: 0x0008 | | | Register Name: PA_CFG2 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 19:16 | R/W | 0xF | PA20_SELECT PA20 Select. 0000: Input 0001: Output 0010: Reserved 0011: NCSI_MCLK 0100: CSI_SM_VS 0101: TCON_TRIG 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT20 1111: IO Disable |
| 15:12 | R/W | 0xF | PA19_SELECT PA19 Select. 0000: Input 0001: Output 0010: Reserved 0011: NCSI_D7 0100: WIEGAND_D1 0101: UART3_CTS 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT19 1111: IO Disable |
| 11:8 | R/W | 0xF | PA18_SELECT PA18 Select. 0000: Input 0001: Output 0010: Reserved 0011: NCSI_D6 0100: WIEGAND_D0 0101: UART3_RTS 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT18 1111: IO Disable |
| 7:4 | R/W | 0xF | PA17_SELECT PA17 Select. 0000: Input 0001: Output 0010: Reserved 0011: NCSI_D5 0100: TWIO_SDA 0101: UART3_RX 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT17 1111: IO Disable |

| Offset: 0x0008 | | | Register Name: PA_CFG2 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 3:0 | R/W | 0xF | PA16_SELECT PA16 Select. 0000: Input 0001: Output 0010: Reserved 0011: NCSI_D4 0100: TWI0_SCK 0101: UART3_TX 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA_EINT16 1111: IO Disable |

10.5.5.4 0x0010 PA Data Register (Default Value: 0x0000_0000)

| Offset: 0x0010 | | | Register Name: PA_DAT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:22 | / | / | / |
| 21:0 | R/W | 0 | PA_DAT PA 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. |

10.5.5.5 0x0014 PA Multi_Driving Register 0 (Default Value: 0x1111_1111)

| Offset: 0x0014 | | | Register Name: PA_DRV0 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:30 | / | / | / |
| 29:28 | R/W | 0x1 | PA7_DRV PA7 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 27:26 | / | / | / |
| 25:24 | R/W | 0x1 | PA6_DRV PA6 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 23:22 | / | / | / |

| Offset: 0x0014 | | | Register Name: PA_DRV0 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 21:20 | R/W | 0x1 | PA5_DRV PA5 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 19:18 | / | / | / |
| 17:16 | R/W | 0x1 | PA4_DRV PA4 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 15:14 | / | / | / |
| 13:12 | R/W | 0x1 | PA3_DRV PA3 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 11:10 | / | / | / |
| 9:8 | R/W | 0x1 | PA2_DRV PA2 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 7:6 | / | / | / |
| 5:4 | R/W | 0x1 | PA1_DRV PA1 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 3:2 | / | / | / |
| 1:0 | R/W | 0x1 | PA0_DRV PA0 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |

10.5.5.6 0x0018 PA Multi_Driving Register 1 (Default Value: 0x1111_1111)

| Offset: 0x0018 | | | Register Name: PA_DRV1 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:30 | / | / | / |
| 29:28 | R/W | 0x1 | PA15_DRV PA15 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 27:26 | / | / | / |

| Offset: 0x0018 | | | Register Name: PA_DRV1 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 25:24 | R/W | 0x1 | PA14_DRV PA14 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 23:22 | / | / | / |
| 21:20 | R/W | 0x1 | PA13_DRV PA13 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 19:18 | / | / | / |
| 17:16 | R/W | 0x1 | PA12_DRV PA12 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 15:14 | / | / | / |
| 13:12 | R/W | 0x1 | PA11_DRV PA11 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 11:10 | / | / | / |
| 9:8 | R/W | 0x1 | PA10_DRV PA10 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 7:6 | / | / | / |
| 5:4 | R/W | 0x1 | PA9_DRV PA9 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 3:2 | / | / | / |
| 1:0 | R/W | 0x1 | PA8_DRV PA8 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |

10.5.5.7 0x001C PA Multi_Driving Register 2 (Default Value: 0x0011_1111)

| Offset: 0x001C | | | Register Name: PA_DRV2 |
|----------------|------------|-------------|------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:22 | / | / | / |

| Offset: 0x001C | | | Register Name: PA_DRV2 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 21:20 | R/W | 0x1 | PA21_DRV PA21 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 19:18 | / | / | / |
| 17:16 | R/W | 0x1 | PA20_DRV PA20 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 15:14 | / | / | / |
| 13:12 | R/W | 0x1 | PA19_DRV PA19 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 11:10 | / | / | / |
| 9:8 | R/W | 0x1 | PA18_DRV PA18 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 7:6 | / | / | / |
| 5:4 | R/W | 0x1 | PA17_DRV PA17 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 3:2 | / | / | / |
| 1:0 | R/W | 0x1 | PA16_DRV PA16 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |

10.5.5.8 0x0024 PA Pull Register 0 (Default Value: 0x0000_0000)

| Offset: 0x0024 | | | Register Name: PA_PULO |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:30 | R/W | 0x0 | PA15_PULL PA15 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 29:28 | R/W | 0x0 | PA14_PULL PA14 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |

| Offset: 0x0024 | | | Register Name: PA_PUL0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 27:26 | R/W | 0x0 | PA13_PULL PA13 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 25:24 | R/W | 0x0 | PA12_PULL PA12 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 23:22 | R/W | 0x0 | PA11_PULL PA11 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 21:20 | R/W | 0x0 | PA10_PULL PA10 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 19:18 | R/W | 0x0 | PA9_PULL PA9 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 17:16 | R/W | 0x0 | PA8_PULL PA8 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 15:14 | R/W | 0x0 | PA7_PULL PA7 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 13:12 | R/W | 0x0 | PA6_PULL PA6 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 11:10 | R/W | 0x0 | PA5_PULL PA5 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 9:8 | R/W | 0x0 | PA4_PULL PA4 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |

| Offset: 0x0024 | | | Register Name: PA_PUL0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 7:6 | R/W | 0x0 | PA3_PULL PA3 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 5:4 | R/W | 0x0 | PA2_PULL PA2 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 3:2 | R/W | 0x0 | PA1_PULL PA1 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 1:0 | R/W | 0x0 | PA0_PULL PA0 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |

10.5.5.9 0x0028 PA Pull Register 1 (Default Value: 0x0000_0000)

| Offset: 0x0028 | | | Register Name: PA_PUL1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:12 | / | / | / |
| 11:10 | R/W | 0x0 | PA21_PULL PA21 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 9:8 | R/W | 0x0 | PA20_PULL PA20 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 7:6 | R/W | 0x0 | PA19_PULL PA19 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 5:4 | R/W | 0x0 | PA18_PULL PA18 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 3:2 | R/W | 0x0 | PA17_PULL PA17 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |

| Offset: 0x0028 | | | Register Name: PA_PUL1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 1:0 | R/W | 0x0 | PA16_PULL PA16 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |

10.5.5.10 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 0010: Reserved 0100: Reserved 0110: Reserved 1000: Reserved 1010: Reserved 1100: Reserved 1110: PC_EINT7 0001: Output 0011: SDC2_D5 0101: Reserved 0111: Reserved 1001: Reserved 1011: Reserved 1101: Reserved 1111: IO Disable |
| 27:24 | R/W | 0xF | PC6_SELECT PC6 Select. 0000: Input 0010: Reserved 0100: SPI0_CS1 0110: Reserved 1000: Reserved 1010: Reserved 1100: Reserved 1110: PC_EINT6 0001: Output 0011: SDC2_D4 0101: Reserved 0111: Reserved 1001: Reserved 1011: Reserved 1101: Reserved 1111: IO Disable |
| 23:20 | R/W | 0xF | PC5_SELECT PC5 Select. 0000: Input 0010: Reserved 0100: SPI0_HOLD 0110: TWI1_SDA 1000: Reserved 1010: Reserved 1100: Reserved 1110: PC_EINT5 0001: Output 0011: SDC2_D3 0101: PWM5 0111: Reserved 1001: Reserved 1011: Reserved 1101: Reserved 1111: IO Disable |

| Offset: 0x0060 | | | Register Name: PC_CFG0 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 19:16 | R/W | 0xF | PC4_SELECT PC4 Select. 0000: Input 0001: Output 0010: Reserved 0011: SDC2_D0 0100: SPI0_WP 0101: PWM4 0110: TWI1_SCK 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PC_EINT4 1111: IO Disable |
| 15:12 | R/W | 0xF | PC3_SELECT PC3 Select. 0000: Input 0001: Output 0010: Reserved 0011: SDC2_D1 0100: SPI0_MISO 0101: BOOT_SEL1 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PC_EINT3 1111: IO Disable |
| 11:8 | R/W | 0xF | PC2_SELECT PC2 Select. 0000: Input 0001: Output 0010: Reserved 0011: SDC2_D2 0100: SPI0_MOSI 0101: BOOT_SEL0 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PC_EINT2 1111: IO Disable |
| 7:4 | R/W | 0xF | PC1_SELECT PC1 Select. 0000: Input 0001: Output 0010: Reserved 0011: SDC2_CMD 0100: SPI0_CS0 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PC_EINT1 1111: IO Disable |

| Offset: 0x0060 | | | Register Name: PC_CFG0 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 3:0 | R/W | 0xF | PC0_SELECT PC0 Select. 0000: Input 0001: Output 0010: Reserved 0011: SDC2_CLK 0100: SPI0_CLK 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PC_EINT0 1111: IO Disable |

10.5.5.11 0x0064 PC Configure Register 1 (Default Value: 0x0000_FFFF)

| Offset: 0x0064 | | | Register Name: PC_CFG1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:12 | R/W | 0xF | PC11_SELECT PC11 Select. 0000: Input 0001: Output 0010: Reserved 0011: SDC2_RST 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PC_EINT11 1111: IO Disable |
| 11:8 | R/W | 0xF | PC10_SELECT PC10 Select. 0000: Input 0001: Output 0010: SPF_DQS 0011: SDC2_DS 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PC_EINT10 1111: IO Disable |

| Offset: 0x0064 | | | Register Name: PC_CFG1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 7:4 | R/W | 0xF | PC9_SELECT PC9 Select. 0000: Input 0001: Output 0010: SPF_IO7 0011: SDC2_D7 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PC_EINT9 1111: IO Disable |
| 3:0 | R/W | 0xF | PC8_SELECT PC8 Select. 0000: Input 0001: Output 0010: SPF_IO6 0011: SDC2_D6 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PC_EINT8 1111: IO Disable |

10.5.5.12 0x0070 PC Data Register (Default Value: 0x0000_0000)

| Offset: 0x0070 | | | Register Name: PC_DAT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:12 | / | / | / |
| 11:0 | R/W | 0 | PC_DAT PC 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. |

10.5.5.13 0x0074 PC Multi_Driving Register 0 (Default Value: 0x1111_1111)

| Offset: 0x0074 | | | Register Name: PC_DRV0 |
|----------------|------------|-------------|------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:30 | / | / | / |

| Offset: 0x0074 | | | Register Name: PC_DRV0 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 29:28 | R/W | 0x1 | PC7_DRV PC7 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 27:26 | / | / | / |
| 25:24 | R/W | 0x1 | PC6_DRV PC6 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 23:22 | / | / | / |
| 21:20 | R/W | 0x1 | PC5_DRV PC5 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 19:18 | / | / | / |
| 17:16 | R/W | 0x1 | PC4_DRV PC4 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 15:14 | / | / | / |
| 13:12 | R/W | 0x1 | PC3_DRV PC3 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 11:10 | / | / | / |
| 9:8 | R/W | 0x1 | PC2_DRV PC2 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 7:6 | / | / | / |
| 5:4 | R/W | 0x1 | PC1_DRV PC1 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 3:2 | / | / | / |
| 1:0 | R/W | 0x1 | PC0_DRV PC0 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |

10.5.5.14 0x0078 PC Multi_Driving Register 1 (Default Value: 0x0000_1111)

| Offset: 0x0078 | | | Register Name: PC_DRV1 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:14 | / | / | / |
| 13:12 | R/W | 0x1 | PC11_DRV PC11 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 11:10 | / | / | / |
| 9:8 | R/W | 0x1 | PC10_DRV PC10 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 7:6 | / | / | / |
| 5:4 | R/W | 0x1 | PC9_DRV PC9 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 3:2 | / | / | / |
| 1:0 | R/W | 0x1 | PC8_DRV PC8 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |

10.5.5.15 0x0084 PC Pull Register (Default Value: 0x0000_0050)

| Offset: 0x0084 | | | Register Name: PC_PUL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:24 | / | / | / |
| 23:22 | R/W | 0x0 | PC11_PULL PC11 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 21:20 | R/W | 0x0 | PC10_PULL PC10 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 19:18 | R/W | 0x0 | PC9_PULL PC9 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |

| Offset: 0x0084 | | | Register Name: PC_PUL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 17:16 | R/W | 0x0 | PC8_PULL PC8 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 15:14 | R/W | 0x0 | PC7_PULL PC7 Pull_up or 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 or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 11:10 | R/W | 0x0 | PC5_PULL PC5 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 9:8 | R/W | 0x0 | PC4_PULL PC4 Pull_up or 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 or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 5:4 | R/W | 0x1 | PC2_PULL PC2 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 3:2 | R/W | 0x0 | PC1_PULL PC1 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 1:0 | R/W | 0x0 | PC0_PULL PC0 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |

10.5.5.16 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: LCD_D11 0011: PWM6 0100: RMII_TXCK 0101: DSI_D2N 0110: SPI1_CS1 0111: MDC 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT7 1111: IO Disable |
| 27:24 | R/W | 0xF | PD6_SELECT PD6 Select. 0000: Input 0001: Output 0010: LCD_D10 0011: PWM5 0100: RMII_RXD0 0101: DSI_CKP 0110: SPI1_WP/DBI_TE 0111: RMII_RXD0 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT6 1111: IO Disable |
| 23:20 | R/W | 0xF | PD5_SELECT PD5 Select. 0000: Input 0001: Output 0010: LCD_D7 0011: PWM4 0100: RMII_RXD1 0101: DSI_CKN 0110: SPI1_HOLD/DBI_DCX/DBI_WRX0111: RMII_RXD1 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT5 1111: IO Disable |

| Offset: 0x0090 | | | Register Name: PD_CFG0 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 19:16 | R/W | 0xF | PD4_SELECT PD4 Select. 0000: Input 0001: Output 0010: LCD_D6 0011: PWM3 0100: RMII_RXER 0101: DSI_D1P 0110: SPI1_MISO/DBI_SDI/DBI_TE/DBI_DCX0111: RMII_CRS_DV 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT4 1111: IO Disable |
| 15:12 | R/W | 0xF | PD3_SELECT PD3 Select. 0000: Input 0001: Output 0010: LCD_D5 0011: PWM2 0100: RMII_CRS_DV 0101: DSI_D1N 0110: SPI1_MOSI/DBI_SDO 0111: RMII_RXER 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT3 1111: IO Disable |
| 11:8 | R/W | 0xF | PD2_SELECT PD2 Select. 0000: Input 0001: Output 0010: LCD_D4 0011: PWM1 0100: RMII_RXD0 0101: DSI_D0P 0110: SPI1_CLK/DBI_SCLK 0111: RMII_TXD1 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT2 1111: IO Disable |
| 7:4 | R/W | 0xF | PD1_SELECT PD1 Select. 0000: Input 0001: Output 0010: LCD_D3 0011: PWM0 0100: RMII_RXD1 0101: DSI_D0N 0110: SPI1_CS0/DBI_CSX 0111: RMII_TXD0 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT1 1111: IO Disable |

| Offset: 0x0090 | | | Register Name: PD_CFG0 | | | | | | | | | | | | | | | | |
|----------------|------------------|-------------|---|-------------|--------------|--------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|------------------|
| Bit | Read/Write | Default/Hex | Description | | | | | | | | | | | | | | | | |
| 3:0 | R/W | 0xF | <p>PDO_SELECT PDO Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: LCD_D2</td><td>0011: Reserved</td></tr> <tr><td>0100: Reserved</td><td>0101: Reserved</td></tr> <tr><td>0110: Reserved</td><td>0111: Reserved</td></tr> <tr><td>1000: Reserved</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PD_EINT0</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: LCD_D2 | 0011: Reserved | 0100: Reserved | 0101: Reserved | 0110: Reserved | 0111: Reserved | 1000: Reserved | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PD_EINT0 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | |
| 0010: LCD_D2 | 0011: Reserved | | | | | | | | | | | | | | | | | | |
| 0100: Reserved | 0101: Reserved | | | | | | | | | | | | | | | | | | |
| 0110: Reserved | 0111: Reserved | | | | | | | | | | | | | | | | | | |
| 1000: Reserved | 1001: Reserved | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | |
| 1110: PD_EINT0 | 1111: IO Disable | | | | | | | | | | | | | | | | | | |

10.5.5.17 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 | <p>PD15_SELECT PD15 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: LCD_D21</td><td>0011: I2S1_DOUT2</td></tr> <tr><td>0100: I2S1_DIN2</td><td>0101: DMIC_DATA0</td></tr> <tr><td>0110: Reserved</td><td>0111: Reserved</td></tr> <tr><td>1000: Reserved</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PD_EINT15</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: LCD_D21 | 0011: I2S1_DOUT2 | 0100: I2S1_DIN2 | 0101: DMIC_DATA0 | 0110: Reserved | 0111: Reserved | 1000: Reserved | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PD_EINT15 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | |
| 0010: LCD_D21 | 0011: I2S1_DOUT2 | | | | | | | | | | | | | | | | | | |
| 0100: I2S1_DIN2 | 0101: DMIC_DATA0 | | | | | | | | | | | | | | | | | | |
| 0110: Reserved | 0111: Reserved | | | | | | | | | | | | | | | | | | |
| 1000: Reserved | 1001: Reserved | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | |
| 1110: PD_EINT15 | 1111: IO Disable | | | | | | | | | | | | | | | | | | |
| 27:24 | R/W | 0xF | <p>PD14_SELECT PD14 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: LCD_D20</td><td>0011: I2S1_DOUT1</td></tr> <tr><td>0100: I2S1_DIN1</td><td>0101: DMIC_DATA1</td></tr> <tr><td>0110: Reserved</td><td>0111: Reserved</td></tr> <tr><td>1000: Reserved</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PD_EINT14</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: LCD_D20 | 0011: I2S1_DOUT1 | 0100: I2S1_DIN1 | 0101: DMIC_DATA1 | 0110: Reserved | 0111: Reserved | 1000: Reserved | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PD_EINT14 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | |
| 0010: LCD_D20 | 0011: I2S1_DOUT1 | | | | | | | | | | | | | | | | | | |
| 0100: I2S1_DIN1 | 0101: DMIC_DATA1 | | | | | | | | | | | | | | | | | | |
| 0110: Reserved | 0111: Reserved | | | | | | | | | | | | | | | | | | |
| 1000: Reserved | 1001: Reserved | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | |
| 1110: PD_EINT14 | 1111: IO Disable | | | | | | | | | | | | | | | | | | |

| Offset: 0x0094 | | | Register Name: PD_CFG1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 23:20 | R/W | 0xF | PD13_SELECT PD13 Select. 0000: Input 0001: Output 0010: LCD_D19 0011: I2S1_DOUT0 0100: Reserved 0101: DMIC_DATA2 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT13 1111: IO Disable |
| 19:16 | R/W | 0xF | PD12_SELECT PD12 Select. 0000: Input 0001: Output 0010: LCD_D18 0011: I2S1_LRCK 0100: Reserved 0101: DMIC_DATA3 0110: Reserved 0111: PWM11 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT12 1111: IO Disable |
| 15:12 | R/W | 0xF | PD11_SELECT PD11 Select. 0000: Input 0001: Output 0010: LCD_D15 0011: I2S1_BCLK 0100: Reserved 0101: DSI_D3P 0110: Reserved 0111: RMII_TXCK 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT11 1111: IO Disable |
| 11:8 | R/W | 0xF | PD10_SELECT PD10 Select. 0000: Input 0001: Output 0010: LCD_D14 0011: I2S1_MCLK 0100: Reserved 0101: DSI_D3N 0110: Reserved 0111: RMII_TXEN 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT10 1111: IO Disable |

| Offset: 0x0094 | | | Register Name: PD_CFG1 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 7:4 | R/W | 0xF | PD9_SELECT PD9 Select. 0000: Input 0001: Output 0010: LCD_D13 0011: PWM8 0100: Reserved 0101: DSI_D2P 0110: Reserved 0111: MDIO 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT9 1111: IO Disable |
| 3:0 | R/W | 0xF | PD8_SELECT PD8 Select. 0000: Input 0001: Output 0010: LCD_D12 0011: PWM7 0100: RMII_TXEN 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT8 1111: IO Disable |

10.5.5.18 0x0098 PD Configure Register 2 (Default Value: 0xFFFF_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: PWM_9 0011: Reserved 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT22 1111: IO Disable |

| Offset: 0x0098 | | | Register Name: PD_CFG2 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 23:20 | R/W | 0xF | PD21_SELECT PD21 Select. 0000: Input 0001: Output 0010: LCD_VSYNC 0011: Reserved 0100: MDIO 0101: SPI2_CS0 0110: TWI2_SDA 0111: UART2_CTS 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT21 1111: IO Disable |
| 19:16 | R/W | 0xF | PD20_SELECT PD20 Select. 0000: Input 0001: Output 0010: LCD_HSYNC 0011: PWM10 0100: MDC 0101: SPI2_MISO 0110: TWI2_SCK 0111: UART2_RTS 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT20 1111: IO Disable |
| 15:12 | R/W | 0xF | PD19_SELECT PD19 Select. 0000: Input 0001: Output 0010: LCD_DE 0011: PWM9 0100: TCON_TRIG 0101: SPI2_MOSI 0110: TWI3_SDA 0111: UART2_RX 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT19 1111: IO Disable |
| 11:8 | R/W | 0xF | PD18_SELECT PD18 Select. 0000: Input 0001: Output 0010: LCD_CLK 0011: Reserved 0100: EPHY_25M 0101: SPI2_CLK 0110: TWI3_SCK 0111: UART2_TX 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT18 1111: IO Disable |

| Offset: 0x0098 | | | Register Name: PD_CFG2 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 7:4 | R/W | 0xF | PD17_SELECT PD17 Select. 0000: Input 0001: Output 0010: LCD_D23 0011: I2S1_DINO 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT17 1111: IO Disable |
| 3:0 | R/W | 0xF | PD16_SELECT PD16 Select. 0000: Input 0001: Output 0010: LCD_D22 0011: I2S1_DOUT3 0100: I2S1_DIN3 0101: DMIC_CLK 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD_EINT16 1111: IO Disable |

10.5.5.19 0x00A0 PD Data Register (Default Value: 0x0000_0000)

| Offset: 0x00A0 | | | Register Name: PD_DAT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:23 | / | / | / |
| 22:0 | R/W | 0 | PD_DAT PD 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. |

10.5.5.20 0x00A4 PD Multi_Driving Register 0 (Default Value: 0x1111_1111)

| Offset: 0x00A4 | | | Register Name: PD_DRV0 |
|----------------|------------|-------------|------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:30 | / | / | / |

| Offset: 0x00A4 | | | Register Name: PD_DRV0 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 29:28 | R/W | 0x1 | PD7_DRV PD7 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 27:26 | / | / | / |
| 25:24 | R/W | 0x1 | PD6_DRV PD6 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 23:22 | / | / | / |
| 21:20 | R/W | 0x1 | PD5_DRV PD5 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 19:18 | / | / | / |
| 17:16 | R/W | 0x1 | PD4_DRV PD4 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 15:14 | / | / | / |
| 13:12 | R/W | 0x1 | PD3_DRV PD3 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 11:10 | / | / | / |
| 9:8 | R/W | 0x1 | PD2_DRV PD2 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 7:6 | / | / | / |
| 5:4 | R/W | 0x1 | PD1_DRV PD1 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 3:2 | / | / | / |
| 1:0 | R/W | 0x1 | PDO_DRV PDO Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |

10.5.5.21 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: Level0 01: Level1 10: Level2 11: Level3 |
| 27:26 | / | / | / |
| 25:24 | R/W | 0x1 | PD14_DRV PD14 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 23:22 | / | / | / |
| 21:20 | R/W | 0x1 | PD13_DRV PD13 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 19:18 | / | / | / |
| 17:16 | R/W | 0x1 | PD12_DRV PD12 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 15:14 | / | / | / |
| 13:12 | R/W | 0x1 | PD11_DRV PD11 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 11:10 | / | / | / |
| 9:8 | R/W | 0x1 | PD10_DRV PD10 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 7:6 | / | / | / |
| 5:4 | R/W | 0x1 | PD9_DRV PD9 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 3:2 | / | / | / |
| 1:0 | R/W | 0x1 | PD8_DRV PD8 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |

10.5.5.22 0x00AC PD Multi_Driving Register 2 (Default Value: 0x0111_1111)

| Offset: 0x00AC | | | Register Name: PD_DRV2 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:26 | / | / | / |
| 25:24 | R/W | 0x1 | PD22_DRV PD22 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 23:22 | / | / | / |
| 21:20 | R/W | 0x1 | PD21_DRV PD21 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 19:18 | / | / | / |
| 17:16 | R/W | 0x1 | PD20_DRV PD20 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 15:14 | / | / | / |
| 13:12 | R/W | 0x1 | PD19_DRV PD19 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 11:10 | / | / | / |
| 9:8 | R/W | 0x1 | PD18_DRV PD18 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 7:6 | / | / | / |
| 5:4 | R/W | 0x1 | PD17_DRV PD17 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 3:2 | / | / | / |
| 1:0 | R/W | 0x1 | PD16_DRV PD16 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |

10.5.5.23 0x00B4 PD Pull Register 0 (Default Value: 0x0000_0000)

| Offset: 0x00B4 | | | Register Name: PD_PUL0 |
|----------------|------------|-------------|---|
| 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 |
| 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 |

| Offset: 0x00B4 | | | Register Name: PD_PUL0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 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 | PDO_PULL PDO Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |

10.5.5.24 0x00B8 PD Pull Register 1 (Default Value: 0x0000_0000)

| Offset: 0x00B8 | | | Register Name: PD_PUL1 |
|----------------|------------|-------------|---|
| 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 |

| Offset: 0x00B8 | | | Register Name: PD_PUL1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 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 |

10.5.5.25 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: NCSI_D3 0011: RGMII_CLKIN/RMII_RXER 0100: LCD_D15 0101: PWM7 0110: UART1_RX 0111: I2S1_DOUT0 1000: TWI4_SDA 1001: Reserved 1010: Reserved 1100: Reserved 1110: PE_EINT7 1111: IO Disable |

| Offset: 0x00C0 | | | Register Name: PE_CFG0 | | | | | | | | | | | | | | | | | | |
|-----------------------------|------------------|-------------|---|-------------|--------------|---------------|--|-----------------------------|--|------------------|------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|------------------|
| Bit | Read/Write | Default/Hex | Description | | | | | | | | | | | | | | | | | | |
| 27:24 | R/W | 0xF | <p>PE6_SELECT PE6 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: NCSI_D2</td><td></td></tr> <tr><td>0011: RGMII_TXCTL/RMII_TXEN</td><td></td></tr> <tr><td>0100: LCD_D2</td><td>0101: PWM6</td></tr> <tr><td>0110: UART1_TX</td><td>0111: Reserved</td></tr> <tr><td>1000: TWI4_SCK</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PE_EINT6</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: NCSI_D2 | | 0011: RGMII_TXCTL/RMII_TXEN | | 0100: LCD_D2 | 0101: PWM6 | 0110: UART1_TX | 0111: Reserved | 1000: TWI4_SCK | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PE_EINT6 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | | | |
| 0010: NCSI_D2 | | | | | | | | | | | | | | | | | | | | | |
| 0011: RGMII_TXCTL/RMII_TXEN | | | | | | | | | | | | | | | | | | | | | |
| 0100: LCD_D2 | 0101: PWM6 | | | | | | | | | | | | | | | | | | | | |
| 0110: UART1_TX | 0111: Reserved | | | | | | | | | | | | | | | | | | | | |
| 1000: TWI4_SCK | 1001: Reserved | | | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | | | |
| 1110: PE_EINT6 | 1111: IO Disable | | | | | | | | | | | | | | | | | | | | |
| 23:20 | R/W | 0xF | <p>PE5_SELECT PE5 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: NCSI_D1</td><td></td></tr> <tr><td>0011: RGMII_RXD1/RMII_RXD1</td><td></td></tr> <tr><td>0100: Reserved</td><td>0101: PWM5</td></tr> <tr><td>0110: SDC1_D3</td><td>0111: TWI3_SDA</td></tr> <tr><td>1000: TWI0_SDA</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PE_EINT5</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: NCSI_D1 | | 0011: RGMII_RXD1/RMII_RXD1 | | 0100: Reserved | 0101: PWM5 | 0110: SDC1_D3 | 0111: TWI3_SDA | 1000: TWI0_SDA | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PE_EINT5 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | | | |
| 0010: NCSI_D1 | | | | | | | | | | | | | | | | | | | | | |
| 0011: RGMII_RXD1/RMII_RXD1 | | | | | | | | | | | | | | | | | | | | | |
| 0100: Reserved | 0101: PWM5 | | | | | | | | | | | | | | | | | | | | |
| 0110: SDC1_D3 | 0111: TWI3_SDA | | | | | | | | | | | | | | | | | | | | |
| 1000: TWI0_SDA | 1001: Reserved | | | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | | | |
| 1110: PE_EINT5 | 1111: IO Disable | | | | | | | | | | | | | | | | | | | | |
| 19:16 | R/W | 0xF | <p>PE4_SELECT PE4 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: NCSI_D0</td><td></td></tr> <tr><td>0011: RGMII_RXD0/RMII_RXD0</td><td></td></tr> <tr><td>0100: I2S1_DOUT0</td><td>0101: PWM4</td></tr> <tr><td>0110: SDC1_D2</td><td>0111: TWI3_SCK</td></tr> <tr><td>1000: TWI0_SCK</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PE_EINT4</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: NCSI_D0 | | 0011: RGMII_RXD0/RMII_RXD0 | | 0100: I2S1_DOUT0 | 0101: PWM4 | 0110: SDC1_D2 | 0111: TWI3_SCK | 1000: TWI0_SCK | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PE_EINT4 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | | | |
| 0010: NCSI_D0 | | | | | | | | | | | | | | | | | | | | | |
| 0011: RGMII_RXD0/RMII_RXD0 | | | | | | | | | | | | | | | | | | | | | |
| 0100: I2S1_DOUT0 | 0101: PWM4 | | | | | | | | | | | | | | | | | | | | |
| 0110: SDC1_D2 | 0111: TWI3_SCK | | | | | | | | | | | | | | | | | | | | |
| 1000: TWI0_SCK | 1001: Reserved | | | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: 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: NCSI_VSYNC 0011: RGMII_RXD0/RMII_RXD0 0100: I2S1_DINO 0101: PWM3 0110: SDC1_D1 0111: UART3_RTS 1000: TWI1_SDA 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE_EINT3 1111: IO Disable |
| 11:8 | R/W | 0xF | PE2_SELECT PE2 Select. 0000: Input 0001: Output 0010: NCSI_HSYNC 0011: RGMII_RXCTL/RMII_CRS_DV 0100: I2S1_LRCK 0101: PWM2 0110: SDC1_D0 0111: UART3_CTS 1000: TWI1_SCK 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE_EINT2 1111: IO Disable |
| 7:4 | R/W | 0xF | PE1_SELECT PE1 Select. 0000: Input 0001: Output 0010: NCSI_MCLK 0011: RGMII_TXCK/RMII_TXCK 0100: I2S1_BCLK 0101: PWM1 0110: SDC1_CMD 0111: UART3_RX 1000: TWI3_SDA 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE_EINT1 1111: IO Disable |

| Offset: 0x00C0 | | | Register Name: PE_CFG0 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 3:0 | R/W | 0xF | PEO_SELECT PEO Select. 0000: Input 0001: Output 0010: NCSI_PCLK 0011: RGMII_RXD1/RMII_RXD1 0100: I2S1_MCLK 0101: PWM0 0110: SDC1_CLK 0111: UART3_TX 1000: TWI3_SCK 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE_EINT0 1111: IO Disable |

10.5.5.26 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 | PE15_SELECT PE15 Select. 0000: Input 0001: Output 0010: NCSI_D11 0011: RGMII_TXD2 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE_EINT15 1111: IO Disable |
| 27:24 | R/W | 0xF | PE14_SELECT PE14 Select. 0000: Input 0001: Output 0010: NCSI_D10 0011: RGMII_TXD3 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE_EINT14 1111: IO Disable |

| Offset: 0x00C4 | | | Register Name: PE_CFG1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 23:20 | R/W | 0xF | PE13_SELECT PE13 Select. 0000: Input 0001: Output 0010: NCSI_D9 0011: RGMII_RXCK 0100: LCD_D23 0101: MIPI_CSI_MCLK1 0110: UART2_RX 0111: UART3_RX 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE_EINT13 1111: IO Disable |
| 19:16 | R/W | 0xF | PE12_SELECT PE12 Select. 0000: Input 0001: Output 0010: NCSI_D8 0011: RGMII_RXD2 0100: LCD_D22 0101: MIPI_CSI_MCLK0 0110: UART2_TX 0111: UART3_TX 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE_EINT12 1111: IO Disable |
| 15:12 | R/W | 0xF | PE11_SELECT PE11 Select. 0000: Input 0001: Output 0010: NCSI_D7 0011: RGMII_RXD3 0100: LCD_D21 0101: CSI_SM_VS 0110: UART2_CTS 0111: I2S1_MCLK 1000: WIEGAND_D1 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE_EINT11 1111: IO Disable |
| 11:8 | R/W | 0xF | PE10_SELECT PE10 Select. 0000: Input 0001: Output 0010: NCSI_D6 0011: EPHY_25M 0100: LCD_D20 0101: PWM10 0110: UART2_RTS 0111: I2S1_BCLK 1000: WIEGAND_D0 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: 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: NCSI_D5 0011: MDIO 0100: LCD_D19 0101: PWM9 0110: WIEGAND_D1 0111: I2S1_LRCK 1000: TWI1_SDA 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE_EINT9 1111: IO Disable |
| 3:0 | R/W | 0xF | PE8_SELECT PE8 Select. 0000: Input 0001: Output 0010: NCSI_D4 0011: MDC 0100: LCD_D18 0101: PWM8 0110: WIEGAND_D0 0111: I2S1_DIN0 1000: TWI1_SCK 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE_EINT8 1111: IO Disable |

10.5.5.27 0x00C8 PE Configure Register 2 (Default Value: 0x0000_0OFF)

| Offset: 0x00C8 | | | Register Name: PE_CFG2 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:4 | R/W | 0xF | PE17_SELECT PE17 Select. 0000: Input 0001: Output 0010: TWI0_SDA 0011: TWI4_SDA 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE_EINT17 1111: IO Disable |

| Offset: 0x00C8 | | | Register Name: PE_CFG2 | | | | | | | | | | | | | | | | |
|-----------------|------------------|-------------|--|-------------|--------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|-----------------|------------------|
| Bit | Read/Write | Default/Hex | Description | | | | | | | | | | | | | | | | |
| 3:0 | R/W | 0xF | <p>PE16_SELECT PE16 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: TWI0_SCK</td><td>0011: TWI4_SCK</td></tr> <tr><td>0100: Reserved</td><td>0101: Reserved</td></tr> <tr><td>0110: Reserved</td><td>0111: Reserved</td></tr> <tr><td>1000: Reserved</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PE_EINT16</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: TWI0_SCK | 0011: TWI4_SCK | 0100: Reserved | 0101: Reserved | 0110: Reserved | 0111: Reserved | 1000: Reserved | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PE_EINT16 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | |
| 0010: TWI0_SCK | 0011: TWI4_SCK | | | | | | | | | | | | | | | | | | |
| 0100: Reserved | 0101: Reserved | | | | | | | | | | | | | | | | | | |
| 0110: Reserved | 0111: Reserved | | | | | | | | | | | | | | | | | | |
| 1000: Reserved | 1001: Reserved | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | |
| 1110: PE_EINT16 | 1111: IO Disable | | | | | | | | | | | | | | | | | | |

10.5.5.28 0x00D0 PE Data Register (Default Value: 0x0000_0000)

| Offset: 0x00D0 | | | Register Name: PE_DAT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:18 | / | / | / |
| 17:0 | R/W | 0 | <p>PE_DAT PE Data.</p> <p>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.</p> |

10.5.5.29 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 | <p>PE7_DRV PE7 Multi_Driving Select.</p> <table> <tr><td>00: Level0</td><td>01: Level1</td></tr> <tr><td>10: Level2</td><td>11: Level3</td></tr> </table> | 00: Level0 | 01: Level1 | 10: Level2 | 11: Level3 |
| 00: Level0 | 01: Level1 | | | | | | |
| 10: Level2 | 11: Level3 | | | | | | |
| 27:26 | / | / | / | | | | |
| 25:24 | R/W | 0x1 | <p>PE6_DRV PE6 Multi_Driving Select.</p> <table> <tr><td>00: Level0</td><td>01: Level1</td></tr> <tr><td>10: Level2</td><td>11: Level3</td></tr> </table> | 00: Level0 | 01: Level1 | 10: Level2 | 11: Level3 |
| 00: Level0 | 01: Level1 | | | | | | |
| 10: Level2 | 11: Level3 | | | | | | |
| 23:22 | / | / | / | | | | |

| Offset: 0x00D4 | | | Register Name: PE_DRV0 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 21:20 | R/W | 0x1 | PE5_DRV PE5 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 19:18 | / | / | / |
| 17:16 | R/W | 0x1 | PE4_DRV PE4 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 15:14 | / | / | / |
| 13:12 | R/W | 0x1 | PE3_DRV PE3 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 11:10 | / | / | / |
| 9:8 | R/W | 0x1 | PE2_DRV PE2 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 7:6 | / | / | / |
| 5:4 | R/W | 0x1 | PE1_DRV PE1 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 3:2 | / | / | / |
| 1:0 | R/W | 0x1 | PE0_DRV PE0 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |

10.5.5.30 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 | PE15_DRV PE15 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 27:26 | / | / | / |

| Offset: 0x00D8 | | | Register Name: PE_DRV1 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 25:24 | R/W | 0x1 | PE14_DRV PE14 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 23:22 | / | / | / |
| 21:20 | R/W | 0x1 | PE13_DRV PE13 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 19:18 | / | / | / |
| 17:16 | R/W | 0x1 | PE12_DRV PE12 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 15:14 | / | / | / |
| 13:12 | R/W | 0x1 | PE11_DRV PE11 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 11:10 | / | / | / |
| 9:8 | R/W | 0x1 | PE10_DRV PE10 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 7:6 | / | / | / |
| 5:4 | R/W | 0x1 | PE9_DRV PE9 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 3:2 | / | / | / |
| 1:0 | R/W | 0x1 | PE8_DRV PE8 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |

10.5.5.31 0x00DC PE Multi_Driving Register 2 (Default Value: 0x0000_0011)

| Offset: 0x00DC | | | Register Name: PE_DRV2 |
|----------------|------------|-------------|------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:6 | / | / | / |

| Offset: 0x00DC | | | Register Name: PE_DRV2 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 5:4 | R/W | 0x1 | PE17_DRV PE17 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 3:2 | / | / | / |
| 1:0 | R/W | 0x1 | PE16_DRV PE16 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |

10.5.5.32 0x00E4 PE Pull Register 0 (Default Value: 0x0000_0000)

| Offset: 0x00E4 | | | Register Name: PE_PUL0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:30 | R/W | 0x0 | PE15_PULL PE15 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 29:28 | R/W | 0x0 | PE14_PULL PE14 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: 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 |
| 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 |

| Offset: 0x00E4 | | | Register Name: PE_PUL0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 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 |
| 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 |

10.5.5.33 0x00E8 PE Pull Register 1 (Default Value: 0x0000_0000)

| Offset: 0x00E8 | | | Register Name: PE_PUL1 |
|----------------|------------|-------------|------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:4 | / | / | / |

| Offset: 0x00E8 | | | Register Name: PE_PUL1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 3:2 | R/W | 0x0 | PE17_PULL PE17 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 1:0 | R/W | 0x0 | PE16_PULL PE16 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |

10.5.5.34 0x00F0 PF Configure Register (Default Value: 0xFFFF_FFFF)

| Offset: 0x00F0 | | | Register Name: PF_CFG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | / | / | / |
| 27:24 | R/W | 0xF | PF6_SELECT PF6 Select. 0000: Input 0010: DBG_CLK 0100: Reserved 0110: Reserved 1000: Reserved 1010: Reserved 1100: Reserved 1110: PF_EINT6 0001: Output 0011: Reserved 0101: Reserved 0111: Reserved 1001: Reserved 1011: Reserved 1101: Reserved 1111: IO Disable |
| 23:20 | R/W | 0xF | PF5_SELECT PF5 Select. 0000: Input 0010: SDC0_D2 0100: Reserved 0110: R_JTAG_CK 1000: Reserved 1010: Reserved 1100: Reserved 1110: PF_EINT5 0001: Output 0011: JTAG_CK 0101: Reserved 0111: Reserved 1001: Reserved 1011: Reserved 1101: Reserved 1111: IO Disable |

| Offset: 0x00F0 | | | Register Name: PF_CFG |
|----------------|------------|-------------|---|
| 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: SPI0_CS1 0101: SPI2_CS1 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PF_EINT4 1111: IO Disable |
| 15:12 | R/W | 0xF | PF3_SELECT PF3 Select. 0000: Input 0001: Output 0010: SDC0_CMD 0011: JTAG_DO 0100: SPI0_CS0 0101: SPI2_CS0 0110: R_JTAG_DO 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: 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: SPI0_MISO 0101: SPI2_MISO 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PF_EINT2 1111: IO Disable |
| 7:4 | R/W | 0xF | PF1_SELECT PF1 Select. 0000: Input 0001: Output 0010: SDC0_D0 0011: JTAG_DI 0100: SPI0莫斯I 0101: SPI2莫斯I 0110: R_JTAG_DI 0111: CPU_BIST1 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PF_EINT1 1111: IO Disable |

| Offset: 0x00F0 | | | Register Name: PF_CFG | | | | | | | | | | | | | | | | |
|-----------------|------------------|-------------|---|-------------|--------------|---------------|---------------|----------------|----------------|-----------------|-----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|------------------|
| Bit | Read/Write | Default/Hex | Description | | | | | | | | | | | | | | | | |
| 3:0 | R/W | 0xF | <p>PFO_SELECT PFO Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: SDC0_D1</td><td>0011: JTAG_MS</td></tr> <tr><td>0100: SPI0_CLK</td><td>0101: SPI2_CLK</td></tr> <tr><td>0110: R_JTAG_MS</td><td>0111: CPU_BIST0</td></tr> <tr><td>1000: Reserved</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PF_EINT0</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: SDC0_D1 | 0011: JTAG_MS | 0100: SPI0_CLK | 0101: SPI2_CLK | 0110: R_JTAG_MS | 0111: CPU_BIST0 | 1000: Reserved | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PF_EINT0 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | |
| 0010: SDC0_D1 | 0011: JTAG_MS | | | | | | | | | | | | | | | | | | |
| 0100: SPI0_CLK | 0101: SPI2_CLK | | | | | | | | | | | | | | | | | | |
| 0110: R_JTAG_MS | 0111: CPU_BIST0 | | | | | | | | | | | | | | | | | | |
| 1000: Reserved | 1001: Reserved | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | |
| 1110: PF_EINT0 | 1111: IO Disable | | | | | | | | | | | | | | | | | | |

10.5.5.35 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 | <p>PF_DAT PF Data.</p> <p>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.</p> |

10.5.5.36 0x0104 PF Multi_Driving Register (Default Value: 0x0111_1111)

| Offset: 0x0104 | | | Register Name: PF_DRV | | | | |
|----------------|------------|-------------|---|------------|------------|------------|------------|
| Bit | Read/Write | Default/Hex | Description | | | | |
| 31:26 | / | / | / | | | | |
| 25:24 | R/W | 0x1 | <p>PF6_DRV PF6 Multi_Driving Select.</p> <table> <tr><td>00: Level0</td><td>01: Level1</td></tr> <tr><td>10: Level2</td><td>11: Level3</td></tr> </table> | 00: Level0 | 01: Level1 | 10: Level2 | 11: Level3 |
| 00: Level0 | 01: Level1 | | | | | | |
| 10: Level2 | 11: Level3 | | | | | | |
| 23:22 | / | / | / | | | | |
| 21:20 | R/W | 0x1 | <p>PF5_DRV PF5 Multi_Driving Select.</p> <table> <tr><td>00: Level0</td><td>01: Level1</td></tr> <tr><td>10: Level2</td><td>11: Level3</td></tr> </table> | 00: Level0 | 01: Level1 | 10: Level2 | 11: Level3 |
| 00: Level0 | 01: Level1 | | | | | | |
| 10: Level2 | 11: Level3 | | | | | | |
| 19:18 | / | / | / | | | | |

| Offset: 0x0104 | | | Register Name: PF_DRV |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 17:16 | R/W | 0x1 | PF4_DRV PF4 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 15:14 | / | / | / |
| 13:12 | R/W | 0x1 | PF3_DRV PF3 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 11:10 | / | / | / |
| 9:8 | R/W | 0x1 | PF2_DRV PF2 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 7:6 | / | / | / |
| 5:4 | R/W | 0x1 | PF1_DRV PF1 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 3:2 | / | / | / |
| 1:0 | R/W | 0x1 | PFO_DRV PFO Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |

10.5.5.37 0x0114 PF Pull Register (Default Value: 0x0000_0000)

| Offset: 0x0114 | | | Register Name: PF_PUL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:14 | / | / | / |
| 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 |

| Offset: 0x0114 | | | Register Name: PF_PUL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 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 | PFO_PULL PFO Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |

10.5.5.38 0x0120 PG Configure Register (Default Value: 0xFFFF_FFFF)

| Offset: 0x0120 | | | Register Name: PG_CFG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | R/W | 0xF | PG7_SELECT PG7 Select. 0000: Input 0001: Output 0010: TWI4_SDA 0011: CLK_FANOUT1 0100: UART1_RX 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PG_EINT7 1111: IO Disable |
| 27:24 | R/W | 0xF | PG6_SELECT PG6 Select. 0000: Input 0001: Output 0010: TWI4_SCK 0011: CLK_FANOUT0 0100: UART1_TX 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PG_EINT6 1111: IO Disable |

| Offset: 0x0120 | | | Register Name: PG_CFG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 23:20 | R/W | 0xF | PG5_SELECT PG5 Select. 0000: Input 0001: Output 0010: SDC1_D3 0011: LCD_D17 0100: UART1_CTS 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PG_EINT5 1111: IO Disable |
| 19:16 | R/W | 0xF | PG4_SELECT PG4 Select. 0000: Input 0001: Output 0010: SDC1_D2 0011: LCD_D16 0100: UART1_RTS 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PG_EINT4 1111: IO Disable |
| 15:12 | R/W | 0xF | PG3_SELECT PG3 Select. 0000: Input 0001: Output 0010: SDC1_D1 0011: LCD_D9 0100: UART3_RTS 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PG_EINT3 1111: IO Disable |
| 11:8 | R/W | 0xF | PG2_SELECT PG2 Select. 0000: Input 0001: Output 0010: SDC1_D0 0011: LCD_D8 0100: UART3_CTS 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PG_EINT2 1111: IO Disable |

| Offset: 0x0120 | | | Register Name: PG_CFG | | | | | | | | | | | | | | | | |
|----------------|------------------|-------------|---|-------------|--------------|----------------|--------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|------------------|
| Bit | Read/Write | Default/Hex | Description | | | | | | | | | | | | | | | | |
| 7:4 | R/W | 0xF | <p>PG1_SELECT PG1 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: SDC1_CMD</td><td>0011: LCD_D1</td></tr> <tr><td>0100: UART3_RX</td><td>0101: Reserved</td></tr> <tr><td>0110: Reserved</td><td>0111: Reserved</td></tr> <tr><td>1000: Reserved</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PG_EINT1</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: SDC1_CMD | 0011: LCD_D1 | 0100: UART3_RX | 0101: Reserved | 0110: Reserved | 0111: Reserved | 1000: Reserved | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PG_EINT1 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | |
| 0010: SDC1_CMD | 0011: LCD_D1 | | | | | | | | | | | | | | | | | | |
| 0100: UART3_RX | 0101: Reserved | | | | | | | | | | | | | | | | | | |
| 0110: Reserved | 0111: Reserved | | | | | | | | | | | | | | | | | | |
| 1000: Reserved | 1001: Reserved | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | |
| 1110: PG_EINT1 | 1111: IO Disable | | | | | | | | | | | | | | | | | | |
| 3:0 | R/W | 0xF | <p>PG0_SELECT PG0 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: SDC1_CLK</td><td>0011: LCD_D0</td></tr> <tr><td>0100: UART3_TX</td><td>0101: Reserved</td></tr> <tr><td>0110: Reserved</td><td>0111: Reserved</td></tr> <tr><td>1000: Reserved</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PG_EINT0</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: SDC1_CLK | 0011: LCD_D0 | 0100: UART3_TX | 0101: Reserved | 0110: Reserved | 0111: Reserved | 1000: Reserved | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PG_EINT0 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | |
| 0010: SDC1_CLK | 0011: LCD_D0 | | | | | | | | | | | | | | | | | | |
| 0100: UART3_TX | 0101: Reserved | | | | | | | | | | | | | | | | | | |
| 0110: Reserved | 0111: Reserved | | | | | | | | | | | | | | | | | | |
| 1000: Reserved | 1001: Reserved | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | |
| 1110: PG_EINT0 | 1111: IO Disable | | | | | | | | | | | | | | | | | | |

10.5.5.39 0x0130 PG Data Register (Default Value: 0x0000_0000)

| Offset: 0x0130 | | | Register Name: PG_DAT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:0 | R/W | 0 | <p>PG_DAT PG Data.</p> <p>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.</p> |

10.5.5.40 0x0134 PG Multi_Driving Register (Default Value: 0x1111_1111)

| Offset: 0x0134 | | | Register Name: PG_DRV |
|----------------|------------|-------------|-----------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:30 | / | / | / |

| Offset: 0x0134 | | | Register Name: PG_DRV |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 29:28 | R/W | 0x1 | PG7_DRV PG7 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 27:26 | / | / | / |
| 25:24 | R/W | 0x1 | PG6_DRV PG6 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 23:22 | / | / | / |
| 21:20 | R/W | 0x1 | PG5_DRV PG5 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 19:18 | / | / | / |
| 17:16 | R/W | 0x1 | PG4_DRV PG4 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 15:14 | / | / | / |
| 13:12 | R/W | 0x1 | PG3_DRV PG3 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 11:10 | / | / | / |
| 9:8 | R/W | 0x1 | PG2_DRV PG2 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 7:6 | / | / | / |
| 5:4 | R/W | 0x1 | PG1_DRV PG1 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 3:2 | / | / | / |
| 1:0 | R/W | 0x1 | PG0_DRV PG0 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |

10.5.5.41 0x0144 PG Pull Register (Default Value: 0x0000_0000)

| Offset: 0x0144 | | | Register Name: PG_PUL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 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 |

10.5.5.42 0x0150 PH Configure Register 0 (Default Value: 0xFFFF_FFFF)

| Offset: 0x0150 | | | Register Name: PH_CFG0 | | | | | | | | | | | | | | | | |
|-----------------|-------------------|-------------|--|-------------|--------------|------------|-------------------|----------------|-------------------|-----------------|------------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|------------------|
| Bit | Read/Write | Default/Hex | Description | | | | | | | | | | | | | | | | |
| 31:28 | R/W | 0xF | <p>PH7_SELECT PH7 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: PWM7</td><td>0011: RMII_CRS_DV</td></tr> <tr><td>0100: UART0_TX</td><td>0101: UART2_RTS</td></tr> <tr><td>0110: I2S1_BCLK</td><td>0111: DMIC_DATA2</td></tr> <tr><td>1000: Reserved</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PH_EINT7</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: PWM7 | 0011: RMII_CRS_DV | 0100: UART0_TX | 0101: UART2_RTS | 0110: I2S1_BCLK | 0111: DMIC_DATA2 | 1000: Reserved | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PH_EINT7 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | |
| 0010: PWM7 | 0011: RMII_CRS_DV | | | | | | | | | | | | | | | | | | |
| 0100: UART0_TX | 0101: UART2_RTS | | | | | | | | | | | | | | | | | | |
| 0110: I2S1_BCLK | 0111: DMIC_DATA2 | | | | | | | | | | | | | | | | | | |
| 1000: Reserved | 1001: Reserved | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | |
| 1110: PH_EINT7 | 1111: IO Disable | | | | | | | | | | | | | | | | | | |
| 27:24 | R/W | 0xF | <p>PH6_SELECT PH6 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: PWM6</td><td>0011: RMII_RXD0</td></tr> <tr><td>0100: TWI2_SDA</td><td>0101: UART2_RX</td></tr> <tr><td>0110: I2S1_MCLK</td><td>0111: DMIC_DATA3</td></tr> <tr><td>1000: Reserved</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PH_EINT6</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: PWM6 | 0011: RMII_RXD0 | 0100: TWI2_SDA | 0101: UART2_RX | 0110: I2S1_MCLK | 0111: DMIC_DATA3 | 1000: Reserved | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PH_EINT6 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | |
| 0010: PWM6 | 0011: RMII_RXD0 | | | | | | | | | | | | | | | | | | |
| 0100: TWI2_SDA | 0101: UART2_RX | | | | | | | | | | | | | | | | | | |
| 0110: I2S1_MCLK | 0111: DMIC_DATA3 | | | | | | | | | | | | | | | | | | |
| 1000: Reserved | 1001: Reserved | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | |
| 1110: PH_EINT6 | 1111: IO Disable | | | | | | | | | | | | | | | | | | |
| 23:20 | R/W | 0xF | <p>PH5_SELECT PH5 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: PWM5</td><td>0011: RMII_RXD1</td></tr> <tr><td>0100: TWI2_SCK</td><td>0101: UART2_TX</td></tr> <tr><td>0110: Reserved</td><td>0111: Reserved</td></tr> <tr><td>1000: Reserved</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PH_EINT5</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: PWM5 | 0011: RMII_RXD1 | 0100: TWI2_SCK | 0101: UART2_TX | 0110: Reserved | 0111: Reserved | 1000: Reserved | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PH_EINT5 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | |
| 0010: PWM5 | 0011: RMII_RXD1 | | | | | | | | | | | | | | | | | | |
| 0100: TWI2_SCK | 0101: UART2_TX | | | | | | | | | | | | | | | | | | |
| 0110: Reserved | 0111: Reserved | | | | | | | | | | | | | | | | | | |
| 1000: Reserved | 1001: Reserved | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | |
| 1110: PH_EINT5 | 1111: IO Disable | | | | | | | | | | | | | | | | | | |
| 19:16 | R/W | 0xF | <p>PH4_SELECT PH4 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: PWM4</td><td>0011: I2S0_DINO</td></tr> <tr><td>0100: SPI1_CS1</td><td>0101: CLK_FANOUT2</td></tr> <tr><td>0110: DMIC_CLK</td><td>0111: Reserved</td></tr> <tr><td>1000: Reserved</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PH_EINT4</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: PWM4 | 0011: I2S0_DINO | 0100: SPI1_CS1 | 0101: CLK_FANOUT2 | 0110: DMIC_CLK | 0111: Reserved | 1000: Reserved | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PH_EINT4 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | |
| 0010: PWM4 | 0011: I2S0_DINO | | | | | | | | | | | | | | | | | | |
| 0100: SPI1_CS1 | 0101: CLK_FANOUT2 | | | | | | | | | | | | | | | | | | |
| 0110: DMIC_CLK | 0111: Reserved | | | | | | | | | | | | | | | | | | |
| 1000: Reserved | 1001: Reserved | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | |
| 1110: PH_EINT4 | 1111: IO Disable | | | | | | | | | | | | | | | | | | |

| Offset: 0x0150 | | | Register Name: PH_CFG0 | | | | | | | | | | | | | | | | |
|------------------|------------------|-------------|--|-------------|--------------|------------|------------------|-----------------|-----------------|------------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|------------------|
| Bit | Read/Write | Default/Hex | Description | | | | | | | | | | | | | | | | |
| 15:12 | R/W | 0xF | <p>PH3_SELECT PH3 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: PWM3</td><td>0011: I2S0_DOUT0</td></tr> <tr><td>0100: SPI1_CS0</td><td>0101: UART3_RTS</td></tr> <tr><td>0110: DMIC_DATA0</td><td>0111: Reserved</td></tr> <tr><td>1000: Reserved</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PH_EINT3</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: PWM3 | 0011: I2S0_DOUT0 | 0100: SPI1_CS0 | 0101: UART3_RTS | 0110: DMIC_DATA0 | 0111: Reserved | 1000: Reserved | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PH_EINT3 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | |
| 0010: PWM3 | 0011: I2S0_DOUT0 | | | | | | | | | | | | | | | | | | |
| 0100: SPI1_CS0 | 0101: UART3_RTS | | | | | | | | | | | | | | | | | | |
| 0110: DMIC_DATA0 | 0111: Reserved | | | | | | | | | | | | | | | | | | |
| 1000: Reserved | 1001: Reserved | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | |
| 1110: PH_EINT3 | 1111: IO Disable | | | | | | | | | | | | | | | | | | |
| 11:8 | R/W | 0xF | <p>PH2_SELECT PH2 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: PWM2</td><td>0011: I2S0_LRCK</td></tr> <tr><td>0100: SPI1_MISO</td><td>0101: UART3_CTS</td></tr> <tr><td>0110: DMIC_DATA1</td><td>0111: Reserved</td></tr> <tr><td>1000: Reserved</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PH_EINT2</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: PWM2 | 0011: I2S0_LRCK | 0100: SPI1_MISO | 0101: UART3_CTS | 0110: DMIC_DATA1 | 0111: Reserved | 1000: Reserved | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PH_EINT2 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | |
| 0010: PWM2 | 0011: I2S0_LRCK | | | | | | | | | | | | | | | | | | |
| 0100: SPI1_MISO | 0101: UART3_CTS | | | | | | | | | | | | | | | | | | |
| 0110: DMIC_DATA1 | 0111: Reserved | | | | | | | | | | | | | | | | | | |
| 1000: Reserved | 1001: Reserved | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | |
| 1110: PH_EINT2 | 1111: IO Disable | | | | | | | | | | | | | | | | | | |
| 7:4 | R/W | 0xF | <p>PH1_SELECT PH1 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: PWM1</td><td>0011: I2S0_BCLK</td></tr> <tr><td>0100: SPI1_MOSI</td><td>0101: UART3_RX</td></tr> <tr><td>0110: DMIC_DATA2</td><td>0111: Reserved</td></tr> <tr><td>1000: Reserved</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PH_EINT1</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: PWM1 | 0011: I2S0_BCLK | 0100: SPI1_MOSI | 0101: UART3_RX | 0110: DMIC_DATA2 | 0111: Reserved | 1000: Reserved | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PH_EINT1 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | |
| 0010: PWM1 | 0011: I2S0_BCLK | | | | | | | | | | | | | | | | | | |
| 0100: SPI1_MOSI | 0101: UART3_RX | | | | | | | | | | | | | | | | | | |
| 0110: DMIC_DATA2 | 0111: Reserved | | | | | | | | | | | | | | | | | | |
| 1000: Reserved | 1001: Reserved | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | |
| 1110: PH_EINT1 | 1111: IO Disable | | | | | | | | | | | | | | | | | | |
| 3:0 | R/W | 0xF | <p>PH0_SELECT PH0 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: PWM0</td><td>0011: I2S0_MCLK</td></tr> <tr><td>0100: SPI1_CLK</td><td>0101: UART3_TX</td></tr> <tr><td>0110: DMIC_DATA3</td><td>0111: Reserved</td></tr> <tr><td>1000: Reserved</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PH_EINT0</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: PWM0 | 0011: I2S0_MCLK | 0100: SPI1_CLK | 0101: UART3_TX | 0110: DMIC_DATA3 | 0111: Reserved | 1000: Reserved | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PH_EINT0 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | |
| 0010: PWM0 | 0011: I2S0_MCLK | | | | | | | | | | | | | | | | | | |
| 0100: SPI1_CLK | 0101: UART3_TX | | | | | | | | | | | | | | | | | | |
| 0110: DMIC_DATA3 | 0111: Reserved | | | | | | | | | | | | | | | | | | |
| 1000: Reserved | 1001: Reserved | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | |
| 1110: PH_EINT0 | 1111: IO Disable | | | | | | | | | | | | | | | | | | |

10.5.5.43 0x0154 PH Configure Register 1 (Default Value: 0xFFFF_FFFF)

| Offset: 0x0154 | | | Register Name: PH_CFG1 | | | | | | | | | | | | | | | | |
|-------------------|-------------------|-------------|---|-------------|--------------|-------------------|-----------------|-----------------|----------------|-----------------|-------------------|------------|----------------|----------------|----------------|----------------|----------------|-----------------|------------------|
| Bit | Read/Write | Default/Hex | Description | | | | | | | | | | | | | | | | |
| 31:28 | R/W | 0xF | <p>PH15_SELECT PH15 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: CLK_FANOUT2</td><td>0011: EPHY_25M</td></tr> <tr><td>0100: Reserved</td><td>0101: Reserved</td></tr> <tr><td>0110: SPI3_CS1</td><td>0111: Reserved</td></tr> <tr><td>1000: PWM8</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PH_EINT15</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: CLK_FANOUT2 | 0011: EPHY_25M | 0100: Reserved | 0101: Reserved | 0110: SPI3_CS1 | 0111: Reserved | 1000: PWM8 | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PH_EINT15 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | |
| 0010: CLK_FANOUT2 | 0011: EPHY_25M | | | | | | | | | | | | | | | | | | |
| 0100: Reserved | 0101: Reserved | | | | | | | | | | | | | | | | | | |
| 0110: SPI3_CS1 | 0111: Reserved | | | | | | | | | | | | | | | | | | |
| 1000: PWM8 | 1001: Reserved | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | |
| 1110: PH_EINT15 | 1111: IO Disable | | | | | | | | | | | | | | | | | | |
| 27:24 | R/W | 0xF | <p>PH14_SELECT PH14 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: JTAG_DI</td><td>0011: MDIO</td></tr> <tr><td>0100: R_JTAG_DI</td><td>0101: TWI3_SDA</td></tr> <tr><td>0110: SPI3_CS0</td><td>0111: WIEGAND_D1</td></tr> <tr><td>1000: PWM7</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PH_EINT14</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: JTAG_DI | 0011: MDIO | 0100: R_JTAG_DI | 0101: TWI3_SDA | 0110: SPI3_CS0 | 0111: WIEGAND_D1 | 1000: PWM7 | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PH_EINT14 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | |
| 0010: JTAG_DI | 0011: MDIO | | | | | | | | | | | | | | | | | | |
| 0100: R_JTAG_DI | 0101: TWI3_SDA | | | | | | | | | | | | | | | | | | |
| 0110: SPI3_CS0 | 0111: WIEGAND_D1 | | | | | | | | | | | | | | | | | | |
| 1000: PWM7 | 1001: Reserved | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | |
| 1110: PH_EINT14 | 1111: IO Disable | | | | | | | | | | | | | | | | | | |
| 23:20 | R/W | 0xF | <p>PH13_SELECT PH13 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: JTAG_DO</td><td>0011: MDC</td></tr> <tr><td>0100: R_JTAG_DO</td><td>0101: TWI3_SCK</td></tr> <tr><td>0110: SPI3_MISO</td><td>0111: WIEGAND_D0</td></tr> <tr><td>1000: PWM6</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PH_EINT13</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: JTAG_DO | 0011: MDC | 0100: R_JTAG_DO | 0101: TWI3_SCK | 0110: SPI3_MISO | 0111: WIEGAND_D0 | 1000: PWM6 | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PH_EINT13 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | |
| 0010: JTAG_DO | 0011: MDC | | | | | | | | | | | | | | | | | | |
| 0100: R_JTAG_DO | 0101: TWI3_SCK | | | | | | | | | | | | | | | | | | |
| 0110: SPI3_MISO | 0111: WIEGAND_D0 | | | | | | | | | | | | | | | | | | |
| 1000: PWM6 | 1001: Reserved | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | |
| 1110: PH_EINT13 | 1111: IO Disable | | | | | | | | | | | | | | | | | | |
| 19:16 | R/W | 0xF | <p>PH12_SELECT PH12 Select.</p> <table> <tr><td>0000: Input</td><td>0001: Output</td></tr> <tr><td>0010: JTAG_CK</td><td>0011: RMII_TXEN</td></tr> <tr><td>0100: R_JTAG_CK</td><td>0101: TWI2_SDA</td></tr> <tr><td>0110: SPI3_MOSI</td><td>0111: CLK_FANOUT1</td></tr> <tr><td>1000: PWM5</td><td>1001: Reserved</td></tr> <tr><td>1010: Reserved</td><td>1011: Reserved</td></tr> <tr><td>1100: Reserved</td><td>1101: Reserved</td></tr> <tr><td>1110: PH_EINT12</td><td>1111: IO Disable</td></tr> </table> | 0000: Input | 0001: Output | 0010: JTAG_CK | 0011: RMII_TXEN | 0100: R_JTAG_CK | 0101: TWI2_SDA | 0110: SPI3_MOSI | 0111: CLK_FANOUT1 | 1000: PWM5 | 1001: Reserved | 1010: Reserved | 1011: Reserved | 1100: Reserved | 1101: Reserved | 1110: PH_EINT12 | 1111: IO Disable |
| 0000: Input | 0001: Output | | | | | | | | | | | | | | | | | | |
| 0010: JTAG_CK | 0011: RMII_TXEN | | | | | | | | | | | | | | | | | | |
| 0100: R_JTAG_CK | 0101: TWI2_SDA | | | | | | | | | | | | | | | | | | |
| 0110: SPI3_MOSI | 0111: CLK_FANOUT1 | | | | | | | | | | | | | | | | | | |
| 1000: PWM5 | 1001: Reserved | | | | | | | | | | | | | | | | | | |
| 1010: Reserved | 1011: Reserved | | | | | | | | | | | | | | | | | | |
| 1100: Reserved | 1101: Reserved | | | | | | | | | | | | | | | | | | |
| 1110: PH_EINT12 | 1111: IO Disable | | | | | | | | | | | | | | | | | | |

| Offset: 0x0154 | | | Register Name: PH_CFG1 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 15:12 | R/W | 0xF | PH11_SELECT PH11 Select. 0000: Input 0001: Output 0010: JTAG_MS 0011: RMII_TXCK 0100: R_JTAG_MS 0101: TWI2_SCK 0110: SPI3_CLK 0111: CLK_FANOUT0 1000: PWM4 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PH_EINT11 1111: IO Disable |
| 11:8 | R/W | 0xF | PH10_SELECT PH10 Select. 0000: Input 0001: Output 0010: PWM10 0011: RMII_RXD0 0100: TWI3_SDA 0101: UART0_RX 0110: I2S1_DOUT0 0111: DMIC_CLK 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PH_EINT10 1111: IO Disable |
| 7:4 | R/W | 0xF | PH9_SELECT PH9 Select. 0000: Input 0001: Output 0010: PWM9 0011: RMII_RXD1 0100: TWI3_SCK 0101: UART0_TX 0110: I2S1_DIN0 0111: DMIC_DATA0 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PH_EINT9 1111: IO Disable |
| 3:0 | R/W | 0xF | PH8_SELECT PH8 Select. 0000: Input 0001: Output 0010: PWM8 0011: RMII_RXER 0100: UART0_RX 0101: UART2_CTS 0110: I2S1_LRCK 0111: DMIC_DATA1 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PH_EINT8 1111: IO Disable |

10.5.5.44 0x0160 PH Data Register (Default Value: 0x0000_0000)

| Offset: 0x0160 | | | Register Name: PH_DAT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R/W | 0 | PH_DAT PH 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. |

10.5.5.45 0x0164 PH Multi_Driving Register 0 (Default Value: 0x1111_1111)

| Offset: 0x0164 | | | Register Name: PH_DRV0 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:30 | / | / | / |
| 29:28 | R/W | 0x1 | PH7_DRV PH7 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 27:26 | / | / | / |
| 25:24 | R/W | 0x1 | PH6_DRV PH6 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 23:22 | / | / | / |
| 21:20 | R/W | 0x1 | PH5_DRV PH5 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 19:18 | / | / | / |
| 17:16 | R/W | 0x1 | PH4_DRV PH4 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 15:14 | / | / | / |
| 13:12 | R/W | 0x1 | PH3_DRV PH3 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 11:10 | / | / | / |

| Offset: 0x0164 | | | Register Name: PH_DRV0 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 9:8 | R/W | 0x1 | PH2_DRV PH2 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 7:6 | / | / | / |
| 5:4 | R/W | 0x1 | PH1_DRV PH1 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 3:2 | / | / | / |
| 1:0 | R/W | 0x1 | PH0_DRV PH0 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |

10.5.5.46 0x0168 PH Multi_Driving Register 1 (Default Value: 0x1111_1111)

| Offset: 0x0168 | | | Register Name: PH_DRV1 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:30 | / | / | / |
| 29:28 | R/W | 0x1 | PH15_DRV PH15 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 27:26 | / | / | / |
| 25:24 | R/W | 0x1 | PH14_DRV PH14 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 23:22 | / | / | / |
| 21:20 | R/W | 0x1 | PH13_DRV PH13 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 19:18 | / | / | / |
| 17:16 | R/W | 0x1 | PH12_DRV PH12 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 15:14 | / | / | / |

| Offset: 0x0168 | | | Register Name: PH_DRV1 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 13:12 | R/W | 0x1 | PH11_DRV PH11 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 11:10 | / | / | / |
| 9:8 | R/W | 0x1 | PH10_DRV PH10 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 7:6 | / | / | / |
| 5:4 | R/W | 0x1 | PH9_DRV PH9 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 3:2 | / | / | / |
| 1:0 | R/W | 0x1 | PH8_DRV PH8 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |

10.5.5.47 0x0174 PH Pull Register (Default Value: 0x0000_0000)

| Offset: 0x0174 | | | Register Name: PH_PUL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:30 | R/W | 0x0 | PH15_PULL PH15 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 29:28 | R/W | 0x0 | PH14_PULL PH14 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 27:26 | R/W | 0x0 | PH13_PULL PH13 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 25:24 | R/W | 0x0 | PH12_PULL PH12 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |

| Offset: 0x0174 | | | Register Name: PH_PUL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 23:22 | R/W | 0x0 | PH11_PULL PH11 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 21:20 | R/W | 0x0 | PH10_PULL PH10 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 19:18 | R/W | 0x0 | PH9_PULL PH9 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 17:16 | R/W | 0x0 | PH8_PULL PH8 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 15:14 | R/W | 0x0 | PH7_PULL PH7 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 13:12 | R/W | 0x0 | PH6_PULL PH6 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 11:10 | R/W | 0x0 | PH5_PULL PH5 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 9:8 | R/W | 0x0 | PH4_PULL PH4 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 7:6 | R/W | 0x0 | PH3_PULL PH3 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 5:4 | R/W | 0x0 | PH2_PULL PH2 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |

| Offset: 0x0174 | | | Register Name: PH_PUL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 3:2 | R/W | 0x0 | PH1_PULL PH1 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 1:0 | R/W | 0x0 | PH0_PULL PH0 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |

10.5.5.48 0x0180 PI Configure Register (Default Value: 0x000F_FFFF)

| Offset: 0x0180 | | | Register Name: PI_CFG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:20 | / | / | / |
| 19:16 | R/W | 0xF | PI4_SELECT PI4 Select. 0000: Input 0010: Reserved 0100: SPI2_CS0 0110: TWI3_SDA 1000: Reserved 1010: Reserved 1100: Reserved 1110: PI_EINT4 0001: Output 0011: Reserved 0101: TWI0_SDA 0111: CLK_FANOUT1 1001: Reserved 1011: Reserved 1101: Reserved 1111: IO Disable |
| 15:12 | R/W | 0xF | PI3_SELECT PI3 Select. 0000: Input 0010: Reserved 0100: SPI2_MISO 0110: TWI3_SCK 1000: Reserved 1010: Reserved 1100: Reserved 1110: PI_EINT3 0001: Output 0011: Reserved 0101: TWI0_SCK 0111: CLK_FANOUT0 1001: Reserved 1011: Reserved 1101: Reserved 1111: IO Disable |

| Offset: 0x0180 | | | Register Name: PI_CFG |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 11:8 | R/W | 0xF | PI2_SELECT PI2 Select. 0000: Input 0001: Output 0010: CSI_SM_VS 0011: TCON_TRIG 0100: SPI2_MOSI 0101: TWI1_SDA 0110: TWI4_SDA 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PI_EINT2 1111: IO Disable |
| 7:4 | R/W | 0xF | PI1_SELECT PI1 Select. 0000: Input 0001: Output 0010: CSI_SM_HS 0011: Reserved 0100: SPI2_CLK 0101: TWI1_SCK 0110: TWI4_SCK 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PI_EINT1 1111: IO Disable |
| 3:0 | R/W | 0xF | PIO_SELECT PIO Select. 0000: Input 0001: Output 0010: MIPI_CSI_MCLK0 0011: Reserved 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PI_EINT0 1111: IO Disable |

10.5.5.49 0x0190 PI Data Register (Default Value: 0x0000_0000)

| Offset: 0x0190 | | | Register Name: PI_DAT |
|----------------|------------|-------------|-----------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:5 | / | / | / |

| Offset: 0x0190 | | | Register Name: PI_DAT |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 4:0 | R/W | 0 | PI_DAT PI 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. |

10.5.5.50 0x0194 PI Multi_Driving Register (Default Value: 0x0001_1111)

| Offset: 0x0194 | | | Register Name: PI_DRV |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:18 | / | / | / |
| 17:16 | R/W | 0x1 | PI4_DRV PI4 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 15:14 | / | / | / |
| 13:12 | R/W | 0x1 | PI3_DRV PI3 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 11:10 | / | / | / |
| 9:8 | R/W | 0x1 | PI2_DRV PI2 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 7:6 | / | / | / |
| 5:4 | R/W | 0x1 | PI1_DRV PI1 Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |
| 3:2 | / | / | / |
| 1:0 | R/W | 0x1 | PIO_DRV PIO Multi_Driving Select. 00: Level0 01: Level1 10: Level2 11: Level3 |

10.5.5.51 0x01A4 PI Pull Register (Default Value: 0x0000_0000)

| Offset: 0x01A4 | | | Register Name: PI_PUL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:10 | / | / | / |
| 9:8 | R/W | 0x0 | PI4_PULL PI4 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 7:6 | R/W | 0x0 | PI3_PULL PI3 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 5:4 | R/W | 0x0 | PI2_PULL PI2 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 3:2 | R/W | 0x0 | PI1_PULL PI1 Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |
| 1:0 | R/W | 0x0 | PIO_PULL PIO Pull_up or down Select. 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved |

10.5.5.52 0x0200 PA External Interrupt Configure Register 0 (Default Value: 0x0000_0000)

| Offset: 0x0200 | | | Register Name: PA_INT_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 |

| Offset: 0x0200 | | | Register Name: PA_INT_CFG0 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 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 |
| 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: 0x0200 | | | Register Name: PA_INT_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 |

10.5.5.53 0x0204 PA External Interrupt Configure Register 1 (Default Value: 0x0000_0000)

| Offset: 0x0204 | | | Register Name: PA_INT_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: 0x0204 | | | Register Name: PA_INT_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 |
| 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: 0x0204 | | | Register Name: PA_INT_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 |

10.5.5.54 0x0208 PA External Interrupt Configure Register 2 (Default Value: 0x0000_0000)

| Offset: 0x0208 | | | Register Name: PA_INT_CFG2 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:24 | / | / | / |
| 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: 0x0208 | | | Register Name: PA_INT_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 |

10.5.5.55 0x0210 PA External Interrupt Control Register (Default Value: 0x0000_0000)

| Offset: 0x0210 | | | Register Name: PA_INT_CTL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:22 | / | / | / |
| 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 |

| Offset: 0x0210 | | | Register Name: PA_INT_CTL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 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 |
| 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: 0x0210 | | | Register Name: PA_INT_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 |

10.5.5.56 0x0214 PA External Interrupt Status Register (Default Value: 0x0000_0000)

| Offset: 0x0214 | | | Register Name: PA_INT_STA |
|----------------|------------|-------------|---------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:22 | / | / | / |

| Offset: 0x0214 | | | Register Name: PA_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 21 | R/W | 0x0 | EINT21_STATUS External INT21 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 20 | R/W | 0x0 | EINT20_STATUS External INT20 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 19 | R/W | 0x0 | EINT19_STATUS External INT19 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 18 | R/W | 0x0 | EINT18_STATUS External INT18 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 17 | R/W | 0x0 | EINT17_STATUS External INT17 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 16 | R/W | 0x0 | EINT16_STATUS External INT16 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 15 | R/W | 0x0 | EINT15_STATUS External INT15 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 14 | R/W | 0x0 | EINT14_STATUS External INT14 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

| Offset: 0x0214 | | | Register Name: PA_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 13 | R/W | 0x0 | EINT13_STATUS External INT13 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 12 | R/W | 0x0 | EINT12_STATUS External INT12 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 11 | R/W | 0x0 | EINT11_STATUS External INT11 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 10 | R/W | 0x0 | EINT10_STATUS External INT10 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 9 | R/W | 0x0 | EINT9_STATUS External INT9 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 8 | R/W | 0x0 | EINT8_STATUS External INT8 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 7 | R/W | 0x0 | EINT7_STATUS External INT7 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 6 | R/W | 0x0 | EINT6_STATUS External INT6 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

| Offset: 0x0214 | | | Register Name: PA_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 5 | R/W | 0x0 | EINT5_STATUS External INT5 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 4 | R/W | 0x0 | EINT4_STATUS External INT4 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 3 | R/W | 0x0 | EINT3_STATUS External INT3 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 2 | R/W | 0x0 | EINT2_STATUS External INT2 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 1 | R/W | 0x0 | EINT1_STATUS External INT1 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 0 | R/W | 0x0 | EINT0_STATUS External INT0 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

10.5.5.57 0x0218 PA External Debounce Configure Register (Default Value: 0x0000_0000)

| Offset: 0x0218 | | | Register Name: PA_INT_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: 0x0218 | | | Register Name: PA_INT_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 |

10.5.5.58 0x0240 PC External Interrupt Configure Register 0 (Default Value: 0x0000_0000)

| Offset: 0x0240 | | | Register Name: PC_INT_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: 0x0240 | | | Register Name: PC_INT_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 | 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 |

10.5.5.59 0x0244 PC External Interrupt Configure Register 1 (Default Value: 0x0000_0000)

| Offset: 0x0244 | | | Register Name: PC_INT_CFG1 |
|----------------|------------|-------------|----------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |

| Offset: 0x0244 | | | Register Name: PC_INT_CFG1 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 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 |
| 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 |

10.5.5.60 0x0250 PC External Interrupt Control Register (Default Value: 0x0000_0000)

| Offset: 0x0250 | | | Register Name: PC_INT_CTL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:12 | / | / | / |
| 11 | R/W | 0x0 | EINT11_CTL External INT11 Enable. 0: Disable 1: Enable |

| Offset: 0x0250 | | | Register Name: PC_INT_CTL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 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 |
| 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 |

| Offset: 0x0250 | | | Register Name: PC_INT_CTL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 0 | R/W | 0x0 | EINT0_CTL External INT0 Enable. 0: Disable 1: Enable |

10.5.5.61 0x0254 PC External Interrupt Status Register (Default Value: 0x0000_0000)

| Offset: 0x0254 | | | Register Name: PC_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:12 | / | / | / |
| 11 | R/W | 0x0 | EINT11_STATUS External INT11 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 10 | R/W | 0x0 | EINT10_STATUS External INT10 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 9 | R/W | 0x0 | EINT9_STATUS External INT9 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 8 | R/W | 0x0 | EINT8_STATUS External INT8 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 7 | R/W | 0x0 | EINT7_STATUS External INT7 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 6 | R/W | 0x0 | EINT6_STATUS External INT6 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

| Offset: 0x0254 | | | Register Name: PC_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 5 | R/W | 0x0 | EINT5_STATUS External INT5 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 4 | R/W | 0x0 | EINT4_STATUS External INT4 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 3 | R/W | 0x0 | EINT3_STATUS External INT3 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 2 | R/W | 0x0 | EINT2_STATUS External INT2 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 1 | R/W | 0x0 | EINT1_STATUS External INT1 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 0 | R/W | 0x0 | EINT0_STATUS External INT0 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

10.5.5.62 0x0258 PC External Debounce Configure Register (Default Value: 0x0000_0000)

| Offset: 0x0258 | | | Register Name: PC_INT_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: 0x0258 | | | Register Name: PC_INT_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 |

10.5.5.63 0x0260 PD External Interrupt Configure Register 0 (Default Value: 0x0000_0000)

| Offset: 0x0260 | | | Register Name: PD_INT_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: 0x0260 | | | Register Name: PD_INT_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 | 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 |

10.5.5.64 0x0264 PD External Interrupt Configure Register 1 (Default Value: 0x0000_0000)

| Offset: 0x0264 | | | Register Name: PD_INT_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 |
| 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_INT_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 |

10.5.5.65 0x0268 PD External Interrupt Configure Register 2 (Default Value: 0x0000_0000)

| Offset: 0x0268 | | | Register Name: PD_INT_CFG2 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | / | / | / |
| 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 |

| Offset: 0x0268 | | | Register Name: PD_INT_CFG2 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 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 |
| 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 |

| Offset: 0x0268 | | | Register Name: PD_INT_CFG2 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 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 |

10.5.5.66 0x0270 PD External Interrupt Control Register (Default Value: 0x0000_0000)

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

| Offset: 0x0270 | | | Register Name: PD_INT_CTL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 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 |
| 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 |

| Offset: 0x0270 | | | Register Name: PD_INT_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 | EINT1_CTL External INT1 Enable. 0: Disable 1: Enable |
| 0 | R/W | 0x0 | EINT0_CTL External INT0 Enable. 0: Disable 1: Enable |

10.5.5.67 0x0274 PD External Interrupt Status Register (Default Value: 0x0000_0000)

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

| Offset: 0x0274 | | | Register Name: PD_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 20 | R/W | 0x0 | EINT20_STATUS External INT20 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 19 | R/W | 0x0 | EINT19_STATUS External INT19 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 18 | R/W | 0x0 | EINT18_STATUS External INT18 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 17 | R/W | 0x0 | EINT17_STATUS External INT17 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 16 | R/W | 0x0 | EINT16_STATUS External INT16 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 15 | R/W | 0x0 | EINT15_STATUS External INT15 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 14 | R/W | 0x0 | EINT14_STATUS External INT14 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 13 | R/W | 0x0 | EINT13_STATUS External INT13 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

| Offset: 0x0274 | | | Register Name: PD_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 12 | R/W | 0x0 | EINT12_STATUS External INT12 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 11 | R/W | 0x0 | EINT11_STATUS External INT11 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 10 | R/W | 0x0 | EINT10_STATUS External INT10 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 9 | R/W | 0x0 | EINT9_STATUS External INT9 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 8 | R/W | 0x0 | EINT8_STATUS External INT8 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 7 | R/W | 0x0 | EINT7_STATUS External INT7 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 6 | R/W | 0x0 | EINT6_STATUS External INT6 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 5 | R/W | 0x0 | EINT5_STATUS External INT5 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

| Offset: 0x0274 | | | Register Name: PD_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 4 | R/W | 0x0 | EINT4_STATUS External INT4 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 3 | R/W | 0x0 | EINT3_STATUS External INT3 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 2 | R/W | 0x0 | EINT2_STATUS External INT2 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 1 | R/W | 0x0 | EINT1_STATUS External INT1 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 0 | R/W | 0x0 | EINT0_STATUS External INT0 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

10.5.5.68 0x0278 PD External Debounce Configure Register (Default Value: 0x0000_0000)

| Offset: 0x0278 | | | Register Name: PD_INT_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 |

10.5.5.69 0x0280 PE External Interrupt Configure Register 0 (Default Value: 0x0000_0000)

| Offset: 0x0280 | | | Register Name: PE_INT_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 |
| 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 |

| Offset: 0x0280 | | | Register Name: PE_INT_CFG0 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 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 |
| 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 |

10.5.5.70 0x0284 PE External Interrupt Configure Register 1 (Default Value: 0x0000_0000)

| Offset: 0x0284 | | | Register Name: PE_INT_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: 0x0284 | | | Register Name: PE_INT_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 |
| 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: 0x0284 | | | Register Name: PE_INT_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 |

10.5.5.71 0x0288 PE External Interrupt Configure Register 2 (Default Value: 0x0000_0000)

| Offset: 0x0288 | | | Register Name: PE_INT_CFG2 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 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 |

10.5.5.72 0x028C PE External Interrupt Configure Register 3 (Default Value: 0x0000_0000)

| Offset: 0x028C | | | Register Name: PE_INT_CFG3 |
|----------------|------------|-------------|----------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | / | / | / |

10.5.5.73 0x0290 PE External Interrupt Control Register (Default Value: 0x0000_0000)

| Offset: 0x0290 | | | Register Name: PE_INT_CTL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:18 | / | / | / |
| 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 |
| 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 |

| Offset: 0x0290 | | | Register Name: PE_INT_CTL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 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 |
| 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 |

10.5.5.74 0x0294 PE External Interrupt Status Register (Default Value: 0x0000_0000)

| Offset: 0x0294 | | | Register Name: PE_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:18 | / | / | / |
| 17 | R/W | 0x0 | EINT17_STATUS External INT17 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 16 | R/W | 0x0 | EINT16_STATUS External INT16 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 15 | R/W | 0x0 | EINT15_STATUS External INT15 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 14 | R/W | 0x0 | EINT14_STATUS External INT14 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 13 | R/W | 0x0 | EINT13_STATUS External INT13 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 12 | R/W | 0x0 | EINT12_STATUS External INT12 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 11 | R/W | 0x0 | EINT11_STATUS External INT11 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

| Offset: 0x0294 | | | Register Name: PE_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 10 | R/W | 0x0 | EINT10_STATUS External INT10 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 9 | R/W | 0x0 | EINT9_STATUS External INT9 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 8 | R/W | 0x0 | EINT8_STATUS External INT8 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 7 | R/W | 0x0 | EINT7_STATUS External INT7 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 6 | R/W | 0x0 | EINT6_STATUS External INT6 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 5 | R/W | 0x0 | EINT5_STATUS External INT5 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 4 | R/W | 0x0 | EINT4_STATUS External INT4 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 3 | R/W | 0x0 | EINT3_STATUS External INT3 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

| Offset: 0x0294 | | | Register Name: PE_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 2 | R/W | 0x0 | EINT2_STATUS External INT2 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 1 | R/W | 0x0 | EINT1_STATUS External INT1 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 0 | R/W | 0x0 | EINT0_STATUS External INT0 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

10.5.5.75 0x0298 PE External Debounce Configure Register (Default Value: 0x0000_0000)

| Offset: 0x0298 | | | Register Name: PE_INT_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 |

10.5.5.76 0x02A0 PF External Interrupt Configure Register (Default Value: 0x0000_0000)

| Offset: 0x02A0 | | | Register Name: PF_INT_CFG |
|----------------|------------|-------------|---------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | / | / | / |

| Offset: 0x02A0 | | | Register Name: PF_INT_CFG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 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 |
| 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_INT_CFG |
|----------------|------------|-------------|--|
| 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 |

10.5.5.77 0x02B0 PF External Interrupt Control Register (Default Value: 0x0000_0000)

| Offset: 0x02B0 | | | Register Name: PF_INT_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 |
| 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 |

| Offset: 0x02B0 | | | Register Name: PF_INT_CTL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 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 |

10.5.5.78 0x02B4 PF External Interrupt Status Register (Default Value: 0x0000_0000)

| Offset: 0x02B4 | | | Register Name: PF_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:7 | / | / | / |
| 6 | R/W | 0x0 | EINT6_STATUS External INT6 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 5 | R/W | 0x0 | EINT5_STATUS External INT5 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 4 | R/W | 0x0 | EINT4_STATUS External INT4 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 3 | R/W | 0x0 | EINT3_STATUS External INT3 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 2 | R/W | 0x0 | EINT2_STATUS External INT2 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

| Offset: 0x02B4 | | | Register Name: PF_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 1 | R/W | 0x0 | EINT1_STATUS External INT1 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 0 | R/W | 0x0 | EINT0_STATUS External INT0 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

10.5.5.79 0x02B8 PF External Debounce Configure Register (Default Value: 0x0000_0000)

| Offset: 0x02B8 | | | Register Name: PF_INT_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 |

10.5.5.80 0x02C0 PG External Interrupt Configure Register (Default Value: 0x0000_0000)

| Offset: 0x02C0 | | | Register Name: PG_INT_CFGd |
|----------------|------------|-------------|--|
| 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 |

| Offset: 0x02C0 | | | Register Name: PG_INT_CFGd |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 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 |
| 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: 0x02C0 | | | Register Name: PG_INT_CFGd |
|----------------|------------|-------------|--|
| 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 |

10.5.5.81 0x02D0 PG External Interrupt Control Register (Default Value: 0x0000_0000)

| Offset: 0x02D0 | | | Register Name: PG_INT_CTL |
|----------------|-------------|-------------|---|
| Bit | sRead/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 |
| 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 |

| Offset: 0x02D0 | | | Register Name: PG_INT_CTL |
|----------------|-------------|-------------|---|
| Bit | sRead/Write | Default/Hex | Description |
| 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 |

10.5.5.82 0x02D4 PG External Interrupt Status Register (Default Value: 0x0000_0000)

| Offset: 0x02D4 | | | Register Name: PG_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7 | R/W | 0x0 | EINT7_STATUS External INT7 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 6 | R/W | 0x0 | EINT6_STATUS External INT6 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 5 | R/W | 0x0 | EINT5_STATUS External INT5 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 4 | R/W | 0x0 | EINT4_STATUS External INT4 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

| Offset: 0x02D4 | | | Register Name: PG_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 3 | R/W | 0x0 | EINT3_STATUS External INT3 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 2 | R/W | 0x0 | EINT2_STATUS External INT2 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 1 | R/W | 0x0 | EINT1_STATUS External INT1 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 0 | R/W | 0x0 | EINT0_STATUS External INT0 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

10.5.5.83 0x02D8 PG External Debounce Configure Register (Default Value: 0x0000_0000)

| Offset: 0x02D8 | | | Register Name: PG_INT_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 |

10.5.5.84 0x02E0 PH External Interrupt Configure Register 0 (Default Value: 0x0000_0000)

| Offset: 0x02E0 | | | Register Name: PH_INT_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 |
| 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 |

| Offset: 0x02E0 | | | Register Name: PH_INT_CFG0 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 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 |
| 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 |

10.5.5.85 0x02E4 PH External Interrupt Configure Register 1 (Default Value: 0x0000_0000)

| Offset: 0x02E4 | | | Register Name: PH_INT_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: 0x02E4 | | | Register Name: PH_INT_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 |
| 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: 0x02E4 | | | Register Name: PH_INT_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 |

10.5.5.86 0x02F0 PH External Interrupt Control Register (Default Value: 0x0000_0000)

| Offset: 0x02F0 | | | Register Name: PH_INT_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 |
| 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 |

| Offset: 0x02F0 | | | Register Name: PH_INT_CTL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 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 |
| 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 |

| Offset: 0x02F0 | | | Register Name: PH_INT_CTL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 0 | R/W | 0x0 | EINT0_CTL External INT0 Enable. 0: Disable 1: Enable |

10.5.5.87 0x02F4 PH External Interrupt Status Register (Default Value: 0x0000_0000)

| Offset: 0x02F4 | | | Register Name: PH_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15 | R/W | 0x0 | EINT15_STATUS External INT15 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 14 | R/W | 0x0 | EINT14_STATUS External INT14 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 13 | R/W | 0x0 | EINT13_STATUS External INT13 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 12 | R/W | 0x0 | EINT12_STATUS External INT12 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 11 | R/W | 0x0 | EINT11_STATUS External INT11 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 10 | R/W | 0x0 | EINT10_STATUS External INT10 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

| Offset: 0x02F4 | | | Register Name: PH_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 9 | R/W | 0x0 | EINT9_STATUS External INT9 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 8 | R/W | 0x0 | EINT8_STATUS External INT8 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 7 | R/W | 0x0 | EINT7_STATUS External INT7 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 6 | R/W | 0x0 | EINT6_STATUS External INT6 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 5 | R/W | 0x0 | EINT5_STATUS External INT5 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 4 | R/W | 0x0 | EINT4_STATUS External INT4 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 3 | R/W | 0x0 | EINT3_STATUS External INT3 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 2 | R/W | 0x0 | EINT2_STATUS External INT2 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

| Offset: 0x02F4 | | | Register Name: PH_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 1 | R/W | 0x0 | EINT1_STATUS External INT1 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 0 | R/W | 0x0 | EINT0_STATUS External INT0 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

10.5.5.88 0x02F8 PH External Debounce Configure Register (Default Value: 0x0000_0000)

| Offset: 0x02F8 | | | Register Name: PH_INT_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 |

10.5.5.89 0x0300 PI External Interrupt Configure Register (Default Value: 0x0000_0000)

| Offset: 0x0300 | | | Register Name: PI_INT_CFG |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:20 | / | / | / |
| 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: 0x0300 | | | Register Name: PI_INT_CFG |
|----------------|------------|-------------|--|
| 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 | 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 |

10.5.5.90 0x0310 PI External Interrupt Control Register (Default Value: 0x0000_0000)

| Offset: 0x0310 | | | Register Name: PI_INT_CTL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:5 | / | / | / |
| 4 | R/W | 0x0 | EINT4_CTL External INT4 Enable. 0: Disable 1: Enable |

| Offset: 0x0310 | | | Register Name: PI_INT_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 |

10.5.5.91 0x0314 PI External Interrupt Status Register (Default Value: 0x0000_0000)

| Offset: 0x0314 | | | Register Name: PI_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:5 | / | / | / |
| 4 | R/W | 0x0 | EINT4_STATUS External INT4 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 3 | R/W | 0x0 | EINT3_STATUS External INT3 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 2 | R/W | 0x0 | EINT2_STATUS External INT2 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |
| 1 | R/W | 0x0 | EINT1_STATUS External INT1 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

| Offset: 0x0314 | | | Register Name: PI_INT_STA |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 0 | R/W | 0x0 | EINT0_STATUS External INT0 Pending Bit. 0: No IRQ pending 1: IRQ pending Write '1' to clear |

10.5.5.92 0x0318 PI External Debounce Configure Register (Default Value: 0x0000_0000)

| Offset: 0x0318 | | | Register Name: PI_INT_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 |

10.5.5.93 0x0340 PIO Group Withstand Voltage Mode Select Register (Default Value: 0x0000_0000)

| 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: 3.3V 1: 1.8V |
| 11:9 | / | / | / |
| 8 | R/W | 0x0 | PI_PWR_MOD_SEL PI_POWER MODE Select. 0: 3.3V 1: 1.8V If PI_Port Power Source select VCC_IO, this bit is invalid |
| 7 | / | / | / |

| Offset: 0x0340 | | | Register Name: PIO_POW_MOD_SEL |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 6 | R/W | 0x0 | PG_PWR_MOD_SEL PG_POWER MODE Select. 0: 3.3V 1: 1.8V If PG Port Power Source select VCC_IO, this bit is invalid |
| 5 | R/W | 0x0 | PF_PWR_MOD_SEL PF_POWER MODE Select. 0: 3.3V 1: 1.8V If PF Port Power Source select VCC_IO, this bit is invalid |
| 4 | R/W | 0x0 | PE_PWR_MOD_SEL PE_POWER MODE Select. 0: 3.3V 1: 1.8V If PE Port Power Source select VCC_IO, this bit is invalid |
| 3 | R/W | 0x0 | PD_PWR_MOD_SEL PD_POWER MODE Select. 0: 3.3V 1: 1.8V If PD Port Power Source select VCC_IO, this bit is invalid |
| 2 | R/W | 0x0 | PC_PWR_MOD_SEL PC_POWER MODE Select. 0: 3.3V 1: 1.8V If PC Port Power Source select VCC_IO, this bit is invalid |
| 1 | / | / | / |
| 0 | R/W | 0x0 | PA_PWR_MOD_SEL PA_POWER MODE Select. 0: 3.3V 1: 1.8V If PA Port Power Source select VCC_IO, this bit is invalid |

10.5.5.94 0x0344 PIO Group Withstand Voltage Mode Select Control Register (Default Value: 0x0000_0000)

| 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 |
| 11:9 | / | / | / |
| 8 | R/W | 0x0 | VCC_PI_WS_VOL_MOD_SEL VCC_PI Withstand Voltage Mode Select Control. 0: Enable 1: Disable |
| 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 | R/W | 0x0 | VCC_PA_WS_VOL_MOD_SEL VCC_PA Withstand Voltage Mode Select Control. 0: Enable 1: Disable |

10.5.5.95 0x0348 PIO Group Power Value Register (Default Value: 0x0000_0000)

| 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. |
| 15:9 | / | / | / |
| 8 | R | 0x0 | PI_PWR_VAL PI_Port Power Value. If PI_Port Power Source select VCC_IO, this bit is invalid |
| 7 | / | / | / |
| 6 | R | 0x0 | PG_PWR_VAL PG_Port Power Value. If PG_Port Power Source select VCC_IO, this bit is invalid |
| 5 | R | 0x0 | PF_PWR_VAL PF_Port Power Value. If PF_Port Power Source select VCC_IO, this bit is invalid |
| 4 | R | 0x0 | PE_PWR_VAL PE_Port Power Value. If PE_Port Power Source select VCC_IO, this bit is invalid |
| 3 | R | 0x0 | PD_PWR_VAL PD_Port Power Value. If PD_Port Power Source select VCC_IO, this bit is invalid |
| 2 | R | 0x0 | PC_PWR_VAL PC_Port Power Value. If PC_Port Power Source select VCC_IO, this bit is invalid |
| 1 | / | / | / |
| 0 | R | 0x0 | PA_PWR_VAL PA_Port Power Value. If PA_Port Power Source select VCC_IO, this bit is invalid |

10.5.5.96 0x0350 PIO Group Power Voltage Select Control Register (Default Value: 0x0000_0001)

| Offset: 0x0350 | | | Register Name: PIO_POW_VAL_SET_CTL |
|----------------|------------|-------------|------------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:1 | / | / | / |

| Offset: 0x0350 | | | Register Name: PIO_POW_VAL_SET_CTL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 0 | R/W | 0x1 | VCC_PF_PWR_VOL_SEL VCC_PF Power Voltage Select Control. 0: 1.8V 1: 3.3V When configuring this register to switch PF port power, you need to configure bit [5] of PIO_POW_MOD_SEL (Offset: 0x0340) to '1', configure bit [5] of PIO_POW_MS_CTL (Offset: 0x0344) to '0' and set the voltage mode of GPIO group to compatible mode. |



10.6 GPADC

10.6.1 Overview

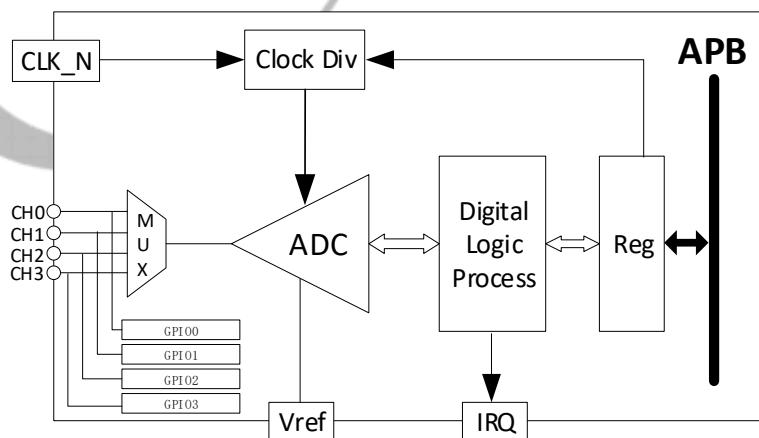
The General Purpose ADC (GPADC) can convert the external signal into a certain proportion of digital value, to realize the measurement of analog signal, which can be applied to power detection and key detection. This ADC is a 4-channel successive approximation register (SAR) A/D converter.

- 12-bit resolution and 8-bit effective SAR type A/D converter
- 4-channel multiplexer
- 64 FIFO depth of data register
- Power Supply Voltage: 1.8 V, Analog Input Range: 0 to 1.8 V
- Maximum sampling frequency: 1 MHz
- Supports data compare and interrupt
- Supports three operation modes
 - Single conversion mode
 - Continuous conversion mode
 - Burst conversion mode

10.6.2 Block Diagram

The following figure shows a block diagram of the GPADC.

Figure 10-32 GPADC Block Diagram



10.6.3 Functional Descriptions

10.6.3.1 External Signals

The following table describes the external signals of GPADC.

Table 10-24GPADC External Signals

| Signal | Description | Type |
|--------|--------------------|------|
| GPADC0 | ADC Input Channel0 | AI |
| GPADC1 | ADC Input Channel1 | AI |
| GPADC2 | ADC Input Channel2 | AI |
| GPADC3 | ADC Input Channel3 | AI |

10.6.3.2 Clock Sources

The following table describes the clock source for GPADC. Users can see section 3.4 Clock Controller Unit (CCU) for clock setting, configuration, and gating information.

Table 10-25 GPADC Clock Sources

| Clock Sources | Description |
|---------------|-------------|
| OSC24M | 24 MHz |

10.6.3.3 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.

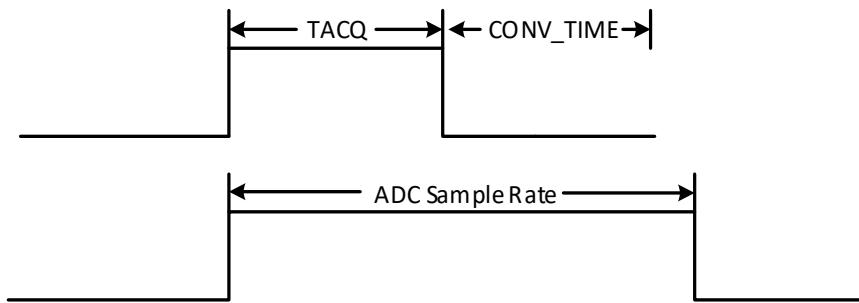
10.6.3.4 Clock and Timing Requirements

CLK_IN = 24 MHz.

CONV_TIME(Conversion Time) = 1/(24MHz/14Cycles) =0.583 (us).

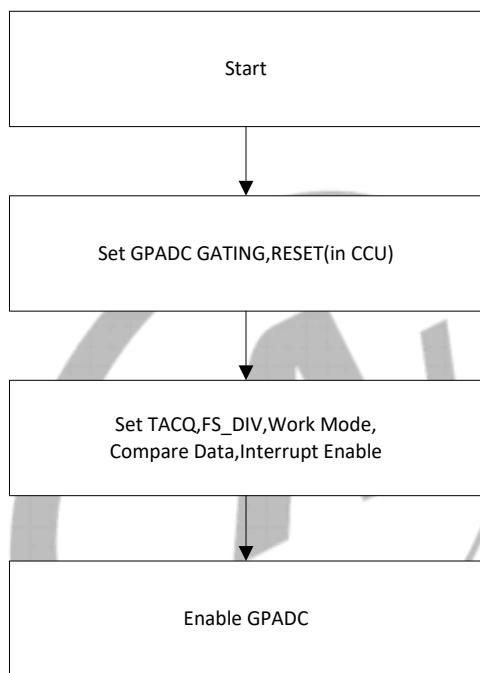
TACQ>10RC (R is output impedance of ADC sample circuit, C = 6.4 pF)

ADC Sample Frequency > TACQ+CONV_TIME.

Figure 10-33 GPADC Clock and Timing Requirement

10.6.4 Programming Guidelines

The GPADC initial process is as follows.

Figure 10-34 GPADC Initial Process

Take channel 0 as an example:

Query Mode

1. Write 0x1 to the bit[16] of [GPADC_BGR_REG \(Offset: 0x09EC\)](#) to dessert reset.
2. Write 0x1 to the bit[0] of [GPADC_BGR_REG \(Offset: 0x09EC\)](#) to enable the GPADC clock.
3. Write 0x2F to the bit[15:0] of [GP_SR_CON \(Offset: 0x0000\)](#) to set the acquiring time of ADC.
4. Write 0x1DF to the bit[31:16] of [GP_SR_CON \(Offset: 0x0000\)](#) to set the ADC sample frequency divider.
5. Write 0x2 to the bit[19:18] of [GP_CTRL \(Offset: 0x0004\)](#) to set the continuous conversion mode.
6. Write 0x1 to the bit[0] of [GP_CS_EN \(Offset: 0x0008\)](#) to enable the analog input channel.
7. Write 0x1 to the bit[16] of [GP_CTRL \(Offset: 0x0004\)](#) to enable the ADC function.

8. Read the bit[0] of [GP_DATA_INTS \(Offset: 0x0038\)](#), if the bit is 1, then data conversion is complete.
9. Read the bit[11:0] of [GP_CHO_DATA \(Offset: 0x0080\)](#), and calculate voltage value based on GPADC formula.

Interrupt Mode

1. Write 0x1 to the bit[16] of [GPADC_BGR_REG \(Offset: 0x09EC\)](#) to dessert reset.
2. Write 0x1 to the bit[0] of [GPADC_BGR_REG \(Offset: 0x09EC\)](#) to enable the GPADC clock.
3. Write 0x2F to the bit[15:0] of [GP_SR_CON \(Offset: 0x0000\)](#) to set the acquiring time of ADC.
4. Write 0x1DF to the bit[31:16] of [GP_SR_CON \(Offset: 0x0000\)](#) to set the ADC sample frequency divider.
5. Write 0x2 to the bit[19:18] of [GP_CTRL \(Offset: 0x0004\)](#) to set the continuous conversion mode.
6. Write 0x1 to the bit[0] of [GP_CS_EN \(Offset: 0x0008\)](#) to enable the analog input channel.
7. Write 0x1 to the bit[0] of [GP_DATA_INTC \(Offset: 0x0028\)](#) to enable the GPADC data interrupt.
8. Set interrupt based on PLIC module.
9. Put interrupt handler address into interrupt vector table.
10. Write 0x1 to the bit16 of [GP_CTRL \(Offset: 0x0004\)](#) to enable the ADC function.
11. Read the bit[11:0] of [GP_CHO_DATA \(Offset: 0x0080\)](#) from the interrupt handler, calculate voltage value based on GPADC formula.

10.6.5 Register List

| Module Name | Base Address |
|-------------|--------------|
| GPADC | 0x02009000 |

| Register Name | Offset | Description |
|-----------------|--------|---|
| GP_SR_CON | 0x0000 | GPADC Sample Rate Config 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_DATAL_INTC | 0x0020 | GPADC Data Low Interrupt Config Register |
| GP_DATAH_INTC | 0x0024 | GPADC Data High Interrupt Config Register |
| GP_DATA_INTC | 0x0028 | GPADC Data Interrupt Config Register |
| GP_DATAL_INTS | 0x0030 | GPADC Data Low Interrupt Status Register |
| GP_DATAH_INTS | 0x0034 | GPADC Data High Interrupt Status Register |
| GP_DATA_INTS | 0x0038 | GPADC Data Interrupt Status Register |
| GP_CHO_CMP_DATA | 0x0040 | GPADC CHO Compare Data Register |

| Register Name | Offset | Description |
|-----------------|--------|---|
| GP_CH1_CMP_DATA | 0x0044 | GPADC CH1 Compare Data Register |
| GP_CH2_CMP_DATA | 0x0048 | GPADC CH2 Compare Data Register |
| GP_CH3_CMP_DATA | 0x004C | GPADC CH3 Compare Data Register |
| GP_CH0_DATA | 0x0080 | GPADC CH0 Data Register |
| GP_CH1_DATA | 0x0084 | GPADC CH1 Data Register |
| GP_CH2_DATA | 0x0088 | GPADC CH2 Data Register |
| GP_CH3_DATA | 0x008C | GPADC CH3 Data Register |
| GP_OPA_EN | 0x00F0 | GPADC OPAMP Enable Register |
| GP_OPA_TUNE0 | 0x00F4 | GPADC OPAMP Tune0 Register |
| IO_CON | 0x0100 | IO Configure Register |
| GP_GPIO_IO_CON | 0x0110 | GPADC GPADC IO Configure Register |
| GP_GPIO_DATA | 0x0120 | GPADC GPIO Data Register |
| GP_GPIO_IRQ_CON | 0x0130 | GPADC GPIO Interrupt Configure Register |
| GP_GPIO_IRQ_EN | 0x0138 | GPADC GPIO Interrupt Enable Register |
| GP_GPIO_IRQ | 0x013C | GPADC GPIO Interrupt Select Register |
| GP_GPIO_PULL | 0x0140 | GPADC GPIO Pull Register |
| GP_GPIO_MDR | 0x0150 | GPADC GPIO Multi-driver Register |

10.6.6 Register Description

10.6.6.1 0x0000 GPADC Sample Rate Config Register (Default Value: 0x01DF_002F)

| Offset: 0x0000 | | | Register Name: GP_SR_CON |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31: 16 | R/W | 0x1DF(50K) | FS_DIV. ADC Sample Frequency Divider CLK_IN/(n+1) |
| 15:0 | R/W | 0x2F(2uS) | TACQ. ADC acquire time (n+1)/CLK_IN |

10.6.6.2 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 | / | / | / |

| Offset: 0x0004 | | | Register Name: GP_CTRL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 21:20 | R/W | 0x0 | ADC_OP_BIAS.(Adjust the bandwidth of the ADC amplifier) ADC OP Bias |
| 19:18 | R/W | 0x0 | GPADC Work Mode 00: Single conversion mode 10: Continuous conversion mode 11: Burst conversion mode |
| 17 | R/W | 0x0 | ADC_CALI_EN. ADC Calibration 1: start Calibration, it is clear to 0 after calibration |
| 16 | R/W | 0x0 | ADC_EN. (Before enabling this bit, configure the parameters such as ADC working mode and channels) ADC Function Enable 0: Disable 1: Enable Note: when the single conversion mode is selected, this bit be automatically cleared to 0 after the conversion is finished. |
| 15:0 | / | / | / |

10.6.6.3 0x0008 GPADC Compare and Select Enable Register (Default Value: 0x0000_0000)

| Offset: 0x0008 | | | Register Name: GP_CS_EN |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:20 | / | / | / |
| 19 | R/W | 0x0 | ADC_CH3_CMP_EN Channel 3 Compare Enable 0: Disable 1: Enable |
| 18 | R/W | 0x0 | ADC_CH2_CMP_EN Channel 2 Compare Enable 0: Disable 1: Enable |
| 17 | R/W | 0x0 | ADC_CH1_CMP_EN Channel 1 Compare Enable 0: Disable 1: Enable |
| 16 | R/W | 0x0 | ADC_CH0_CMP_EN Channel 0 Compare Enable 0: Disable 1: Enable |
| 15:4 | / | / | / |

| Offset: 0x0008 | | | Register Name: GP_CS_EN |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 3 | R/W | 0x0 | ADC_CH3_SELECT. Analog input channel 3 Select 0: Disable 1: Enable |
| 2 | R/W | 0x0 | ADC_CH2_SELECT. Analog input channel 2 Select 0: Disable 1: Enable |
| 1 | R/W | 0x0 | ADC_CH1_SELECT. Analog input channel 1 Select 0: Disable 1: Enable |
| 0 | R/W | 0x0 | ADC_CH0_SELECT. Analog input channel 0 Select 0: Disable 1: Enable |

10.6.6.4 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 Date DRQ Enable 0: Disable 1: Enable |
| 17 | R/W | 0x0 | FIFO_OVERRUN_IRQ_EN. ADC FIFO Over Run 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_TRIGGER_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, self clear to '0' |

| Offset: 0x000C | | | Register Name: GP_FIFO_INTC |
|-----------------------|-------------------|--------------------|------------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 3:0 | / | / | / |

10.6.6.5 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 Over Run IRQ pending 0: No Pending IRQ 1: FIFO Overrun Pending IRQ Write '1' to clear this interrupt |
| 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 |
| 15:14 | / | / | / |
| 13:8 | R | 0x0 | RXA_CNT. ADC FIFO available Sample Word Counter |
| 7:0 | / | / | / |

10.6.6.6 0x0014 GPADC FIFO Data Register (Default Value: 0x0000_0xxx)

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

10.6.6.7 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 | 0x000 | GP_CDATA GPADC Calibration Data |

10.6.6.8 0x0020 GPADC Low Interrupt Configuration Register (Default Value: 0x0000_0000)

| Offset: 0x0020 | | | Register Name: GP_DATAL_INTC |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:4 | / | / | / |
| 3 | R/W | 0x0 | CH3_LOW_IRQ_EN 0: Disable 1: Enable |
| 2 | R/W | 0x0 | CH2_LOW_IRQ_EN 0: Disable 1: Enable |
| 1 | R/W | 0x0 | CH1_LOW_IRQ_EN 0: Disable 1: Enable |
| 0 | R/W | 0x0 | CHO_LOW_IRQ_EN 0: Disable 1: Enable |

10.6.6.9 0x0024 GPADC HIGH Interrupt Configuration Register (Default Value: 0x0000_0000)

| Offset: 0x0024 | | | Register Name: GP_DATAH_INTC |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:4 | / | / | / |
| 3 | R/W | 0x0 | CH3_HIG_IRQ_EN 0: Disable 1: Enable |
| 2 | R/W | 0x0 | CH2_HIG_IRQ_EN 0: Disable 1: Enable |
| 1 | R/W | 0x0 | CH1_HIG_IRQ_EN 0: Disable 1: Enable |
| 0 | R/W | 0x0 | CHO_HIG_IRQ_EN 0: Disable 1: Enable |

10.6.6.10 0x0028 GPADC DATA Interrupt Configuration Register (Default Value: 0x0000_0000)

| Offset: 0x0028 | | | Register Name: GP_DATA_INTC |
|----------------|------------|-------------|-----------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:4 | / | / | / |

| Offset: 0x0028 | | | Register Name: GP_DATA_INTC |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 3 | R/W | 0x0 | CH3_DATA_IRQ_EN 0: Disable 1: Enable |
| 2 | R/W | 0x0 | CH2_DATA_IRQ_EN 0: Disable 1: Enable |
| 1 | R/W | 0x0 | CH1_DATA_IRQ_EN 0: Disable 1: Enable |
| 0 | R/W | 0x0 | CHO_DATA_IRQ_EN 0: Disable 1: Enable |

10.6.6.11 0x0030 GPADC Low Interrupt Status Register (Default Value: 0x0000_0000)

| Offset: 0x0030 | | | Register Name: GP_DATAL_INTS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:4 | / | / | / |
| 3 | R/W1C | 0x0 | CH3_LOW_PENGDING 1: Channel 3 Voltage Low Available Pending IRQ Write '1' to clear this interrupt or automatic clear if interrupt condition fails |
| 2 | R/W1C | 0x0 | CH2_LOW_PENGDING 1: Channel 2 Voltage Low Available Pending IRQ Write '1' to clear this interrupt or automatic clear if interrupt condition fails |
| 1 | R/W1C | 0x0 | CH1_LOW_PENGDING 1: Channel 1 Voltage Low Available Pending IRQ Write '1' to clear this interrupt or automatic clear if interrupt condition fails |
| 0 | R/W1C | 0x0 | CHO_LOW_PENGDING 1: Channel 0 Voltage Low Available Pending IRQ Write '1' to clear this interrupt or automatic clear if interrupt condition fails |

10.6.6.12 0x0034 GPADC High Interrupt Status Register (Default Value: 0x0000_0000)

| Offset: 0x0034 | | | Register Name: GP_DATAH_INTS |
|----------------|------------|-------------|------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:4 | / | / | / |

| Offset: 0x0034 | | | Register Name: GP_DATAH_INTS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 3 | R/W1C | 0x0 | CH3_HIG_PENGding 0: NO Pending IRQ 1: Channel 3 Voltage High Available Pending IRQ Write '1' to clear this interrupt or automatic clear if interrupt condition fails |
| 2 | R/W1C | 0x0 | CH2_HIG_PENGding 0: NO Pending IRQ 1: Channel 2 Voltage High Available Pending IRQ Write '1' to clear this interrupt or automatic clear if interrupt condition fails |
| 1 | R/W1C | 0x0 | CH1_HIG_PENGding 0: NO Pending IRQ 1: Channel 1 Voltage High Available Pending IRQ Write '1' to clear this interrupt or automatic clear if interrupt condition fails |
| 0 | R/W1C | 0x0 | CH0_HIG_PENGding 0: NO Pending IRQ 1: Channel 0 Voltage High Available Pending IRQ Write '1' to clear this interrupt or automatic clear if interrupt condition fails |

10.6.6.13 0x0038 GPADC Data Interrupt Status Register (Default Value: 0x0000_0000)

| Offset: 0x0038 | | | Register Name: GP_DATA_INTS |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:4 | / | / | / |
| 3 | R/W1C | 0x0 | CH3_DATA_PENGding 0: NO Pending IRQ 1: Channel 3 Data Available Pending IRQ Write '1' to clear this interrupt or automatic clear if interrupt condition fails |
| 2 | R/W1C | 0x0 | CH2_DATA_PENGding 0: NO Pending IRQ 1: Channel 2 Data Available Pending IRQ Write '1' to clear this interrupt or automatic clear if interrupt condition fails |
| 1 | R/W1C | 0x0 | CH1_DATA_PENGding 0: NO Pending IRQ 1: Channel 1 Data Available Pending IRQ Write '1' to clear this interrupt or automatic clear if interrupt condition fails |

| Offset: 0x0038 | | | Register Name: GP_DATA_INTS |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 0 | R/W1C | 0x0 | CHO_DATA_PENGDING 0: NO Pending IRQ 1: Channel 0 Data Available Pending IRQ Write '1' to clear this interrupt or automatic clear if interrupt condition fails |

10.6.6.14 0x0040 GPADC CH0 Compare Data Register (Default Value: 0xBFF_0400)

| Offset: 0x0040 | | | Register Name: GP_CH0_CMP_DATA |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | / | / | / |
| 27:16 | R/W | 0xBFF | CH0_CMP_HIG_DATA Channel 0 Voltage High Value |
| 15:12 | / | / | / |
| 11:0 | R/W | 0x400 | CH0_CMP_LOW_DATA Channel 0 Voltage Low Value |

10.6.6.15 0x0044 GPADC CH1 Compare Data Register (Default Value: 0xBFF_0400)

| Offset: 0x0044 | | | Register Name: GP_CH1_CMP_DATA |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | / | / | / |
| 27:16 | R/W | 0xBFF | CH1_CMP_HIG_DATA Channel 1 Voltage High Value |
| 15:12 | / | / | / |
| 11:0 | R/W | 0x400 | CH1_CMP_LOW_DATA Channel 1 Voltage Low Value |

10.6.6.16 0x0048 GPADC CH2 Compare Data Register (Default Value: 0xBFF_0400)

| Offset: 0x0048 | | | Register Name: GP_CH2_CMP_DATA |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | / | / | / |
| 27:16 | R/W | 0xBFF | CH2_CMP_HIG_DATA Channel 2 Voltage High Value |
| 15:12 | / | / | / |
| 11:0 | R/W | 0x400 | CH2_CMP_LOW_DATA Channel 2 Voltage Low Value |

10.6.6.17 0x004C GPADC CH3 Compare Data Register (Default Value: 0xOBFF_0400)

| Offset: 0x004C | | | Register Name: GP_CH3_CMP_DATA |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | / | / | / |
| 27:16 | R/W | 0xBFF | CH3_CMP_HIG_DATA Channel 3 Voltage High Value |
| 15:12 | / | / | / |
| 11:0 | R/W | 0x400 | CH3_CMP_LOW_DATA Channel 3 Voltage Low Value |

10.6.6.18 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 |

10.6.6.19 0x0084 GPADC CH1 Data Register (Default Value: 0x0000_0000)

| Offset: 0x0084 | | | Register Name: GP_CH1_DATA |
|----------------|------------|-------------|-------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:12 | / | / | / |
| 11:0 | R | 0x000 | GP_CH1_DATA Channel 1 Data |

10.6.6.20 0x0088 GPADC CH2 Data Register (Default Value: 0x0000_0000)

| Offset: 0x0088 | | | Register Name: GP_CH2_DATA |
|----------------|------------|-------------|-------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:12 | / | / | / |
| 11:0 | R | 0x000 | GP_CH2_DATA Channel 2 Data |

10.6.6.21 0x008C GPADC CH3 Data Register (Default Value: 0x0000_0000)

| Offset: 0x008C | | | Register Name: GP_CH3_DATA |
|----------------|------------|-------------|-------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:12 | / | / | / |
| 11:0 | R | 0x000 | GP_CH3_DATA Channel 3 Data |

10.6.6.22 0x00F0 GPADC OPAMP Enable Register Description (Default Value: 0x0000_0000)

| Offset: 0x00F0 | | | Register Name: GP_OPA_EN |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:20 | / | / | / |
| 19 | R/W | 0x0 | GP_CH3_OPA_SEL 0: ADC input bypass operation amplifier 1: ADC input from operation amplifier |
| 18 | R/W | 0x0 | GP_CH2_OPA_SEL 0: ADC input bypass operation amplifier 1: ADC input from operation amplifier |
| 17 | R/W | 0x0 | GP_CH1_OPA_SEL 0: ADC input bypass operation amplifier 1: ADC input from operation amplifier |
| 16 | R/W | 0x0 | GP_CH0_OPA_SEL 0: ADC input bypass operation amplifier 1: ADC input from operation amplifier |
| 15:4 | / | / | / |
| 3 | R/W | 0x0 | GP_CH3_OPA_EN 0: Disable 1:enable operation amplifier for GPADC Channel 3 |
| 2 | R/W | 0x0 | GP_CH2_OPA_EN 0: Disable 1:enable operation amplifier for GPADC Channel 2 |
| 1 | R/W | 0x0 | GP_CH1_OPA_EN 0: Disable 1:enable operation amplifier for GPADC Channel 1 |
| 0 | R/W | 0x0 | GP_CH0_OPA_EN 0: Disable 1:enable operation amplifier for GPADC Channel 0 |

10.6.6.23 0x00F4 GPADC OPAMP Tune0 Register Description (Default Value 0x0000_8888)

| Offset: 0x00F4 | | | Register Name: GP_OPA_TUNE0 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:12 | R/W | 0x8 | GP_CH3_OPA_TUNE Indicates the magnification of operation amplifier for GPADC Channel 3 0: 1 times 1: 2 times 15: 16 times |

| Offset: 0x00F4 | | | Register Name: GP_OPA_TUNE0 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 11:8 | R/W | 0x8 | <p>GP_CH2_OPA_TUNE Indicates the magnification of operation amplifier for GPADC Channel 2 0: 1 times 1: 2 times 15: 16 times</p> |
| 7:4 | R/W | 0x8 | <p>GP_CH1_OPA_TUNE Indicates the magnification of operation amplifier for GPADC Channel 1 0: 1 times 1: 2 times 15: 16 times</p> |
| 3:0 | R/W | 0x8 | <p>GP_CH0_OPA_TUNE Indicates the magnification of operation amplifier for GPADC Channel 0 0: 1 times 1: 2 times 15: 16 times</p> |

10.6.6.24 0x0100 6.24. IO Configure Register Description (Default Value 0x0000_0000)

| Offset: 0x0100 | | | Register Name: IO_CON |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:4 | / | / | / |
| 3 | R/W | 0x0 | <p>CH3_IO_CON Channel 3 GPADC or GPIO IO Configure 0: Used IO as GPADC Pin 1: Used IO as GPIO Pin</p> |
| 2 | R/W | 0x0 | <p>CH2_IO_CON Channel 2 GPADC or GPIO IO Configure 0: Used IO as GPADC Pin 1: Used IO as GPIO Pin</p> |
| 1 | R/W | 0x0 | <p>CH1_IO_CON Channel 1 GPADC or GPIO IO Configure 0: Used IO as GPADC Pin 1: Used IO as GPIO Pin</p> |

| Offset: 0x0100 | | | Register Name: IO_CON |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 0 | R/W | 0x0 | CH0_IO_CON Channel 0 GPADC or GPIO IO Configure 0: Used IO as GPADC Pin 1: Used IO as GPIO Pin |

10.6.6.25 0x0110 GPADC GPIO IO Configure Register Description (Default Value 0x0000_0000)

| Offset: 0x0110 | | | Register Name: GP_GPIO_IO_CON |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:6 | R/W | 0x0 | CH3_GPIO_IO_CON Channel 3 GPIO IO Configure 00: Disable 01: Input 10: Output 11: Interrupt |
| 5:4 | R/W | 0x0 | CH2_GPIO_IO_CON Channel 2 GPIO IO Configure 00: Disable 01: Input 10: Output 11: Interrupt |
| 3:2 | R/W | 0x0 | CH1_GPIO_IO_CON Channel 1 GPIO IO Configure 00: Disable 01: Input 10: Output 11: Interrupt |
| 1:0 | R/W | 0x0 | CH0_GPIO_IO_CON Channel 0 GPIO IO Configure 00: Disable 01: Input 10: Output 11: Interrupt |

10.6.6.26 0x0120 GPADC GPIO Data Register Description (Default Value 0x0000_0000)

| Offset: 0x0120 | | | Register Name: GP_GPIO_DATA |
|----------------|------------|-------------|-----------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:4 | / | / | / |

| Offset: 0x0120 | | | Register Name: GP_GPIO_DATA |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 3 | R/W | 0x0 | CH3_GPIO_DATA Channel 3 GPIO DATA Show the value of input or output data |
| 2 | R/W | 0x0 | CH2_GPIO_DATA Channel 2 GPIO DATA Show the value of input or output data |
| 1 | R/W | 0x0 | CH1_GPIO_DATA Channel 1 GPIO DATA Show the value of input or output data |
| 0 | R/W | 0x0 | CH0_GPIO_DATA Channel 0 GPIO DATA Show the value of input or output data |

10.6.6.27 0x0130 GPADC GPIO Interrupt Configure0 Register Description (Default Value 0x0000_0000)

| Offset: 0x0130 | | | Register Name: GP_GPIO_IRQ_CON0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | /GPIO |
| 15:12 | R/W | 0x0 | CH3_GP_GPIO_IRQ_CON Channel 3 GPADC GPIO Interrupt Configure. 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved |
| 11:8 | R/W | 0x0 | CH2_GP_GPIO_IRQ_CON Channel 2 GPADC GPIO Interrupt Configure. 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved |
| 7:4 | R/W | 0x0 | CH1_GP_GPIO_IRQ_CON Channel 1 GPADC GPIO Interrupt Configure. 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved |

| Offset: 0x0130 | | | Register Name: GP_GPIO_IRQ_CON0 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 3:0 | R/W | 0x0 | CH0_GPIO_IRQ_CON Channel 0 GPADC GPIO Interrupt Configure. 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved |

10.6.6.28 0x0138 GPADC GPIO Interrupt Enable Register Description (Default Value 0x0000_0000)

| Offset: 0x0138 | | | Register Name: GP_GPIO_IRQ_EN |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:4 | / | / | / |
| 3 | R/W | 0x0 | CH3_GPIO_IRQ_EN Channel 3 GPIO Interrupt Enable 0: Disable 1: Enable |
| 2 | R/W | 0x0 | CH2_GPIO_IRQ_EN Channel 2 GPIO Interrupt Enable 0: Disable 1: Enable |
| 1 | R/W | 0x0 | CH1_GPIO_IRQ_EN Channel 1 GPIO Interrupt Enable 0: Disable 1: Enable |
| 0 | R/W | 0x0 | CH0_GPIO_IRQ_EN Channel 0 GPIO Interrupt Enable 0: Disable 1: Enable |

10.6.6.29 0x013C GPADC GPIO Interrupt Pending Register Description (Default Value 0x0000_0000)

| Offset: 0x013C | | | Register Name: GP_GPIO_IRQ_PENDING |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:4 | / | / | / |
| 3 | R/W1C | 0x0 | CH3_GPIO_IRQ_PENDING Channel 3 GPIO Interrupt Pending 0: NO interrupt pending 1: GPIO has at least one interrupt pending |

| Offset: 0x013C | | | Register Name: GP_GPIO_IRQ_PENDING |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 2 | R/W1C | 0x0 | CH2_GPIO_IRQ_PENDING Channel 2 GPIO Interrupt Pending 0: NO interrupt pending 1: GPIO has at least one interrupt pending |
| 1 | R/W1C | 0x0 | CH1_GPIO_IRQ_PENDING Channel 1 GPIO Interrupt Pending 0: NO interrupt pending 1: GPIO has at least one interrupt pending |
| 0 | R/W1C | 0x0 | CH0_GPIO_IRQ_PENDING Channel 0 GPIO Interrupt Pending 0: NO interrupt pending 1: GPIO has at least one interrupt pending |

10.6.6.30 0x0140 GPADC GPIO Pull Register Description (Default Value 0x0000_0000)

| Offset: 0x0140 | | | Register Name: GP_GPIO_PULL |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:6 | R/W | 0x0 | CH3_GPIO_PULL Channel 3 GPIO Pull Register 00: Both pull up and pull down are disabled 01: Pull up enable 10: Pull down enable 00: Reserved |
| 5:4 | R/W | 0x0 | CH2_GPIO_PULL Channel 2GPIO Pull Register 00: Both pull up and pull down are disabled 01: Pull up enable 10: Pull down enable 00: Reserved |
| 3:2 | R/W | 0x0 | CH1_GPIO_PULL Channel 1 GPIO Pull Register 00: Both pull up and pull down are disabled 01: Pull up enable 10: Pull down enable 00: Reserved |
| 1:0 | R/W | 0x0 | CH0_GPIO_PULL Channel 0 GPIO Pull Register 00: Both pull up and pull down are disabled 01: Pull up enable 10: Pull down enable 00: Reserved |

10.6.6.31 0x0150 GPADC GPIO Multi-drive Register Description (Default Value 0x0000_0000)

| Offset: 0x0150 | | | Register Name: GP_GPIO_MDR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:6 | R/W | 0x0 | CH3_GPIO_SEL Channel 3 GPIO Selectors 00: Level 0 01: Level 1 10: Level 2 11: Level 3 |
| 5:4 | R/W | 0x0 | CH2_GPIO_SEL Channel 2 GPIO Selectors 00: Level 0 01: Level 1 10: Level 2 11: Level 3 |
| 3:2 | R/W | 0x0 | CH1_GPIO_SEL Channel 1 GPIO Selectors 00: Level 0 01: Level 1 10: Level 2 11: Level 3 |
| 1:0 | R/W | 0x0 | CH0_GPIO_SEL Channel 0 GPIO Selectors 00: Level 0 01: Level 1 10: Level 2 11: Level 3 |

10.7 PWM

10.7.1 Overview

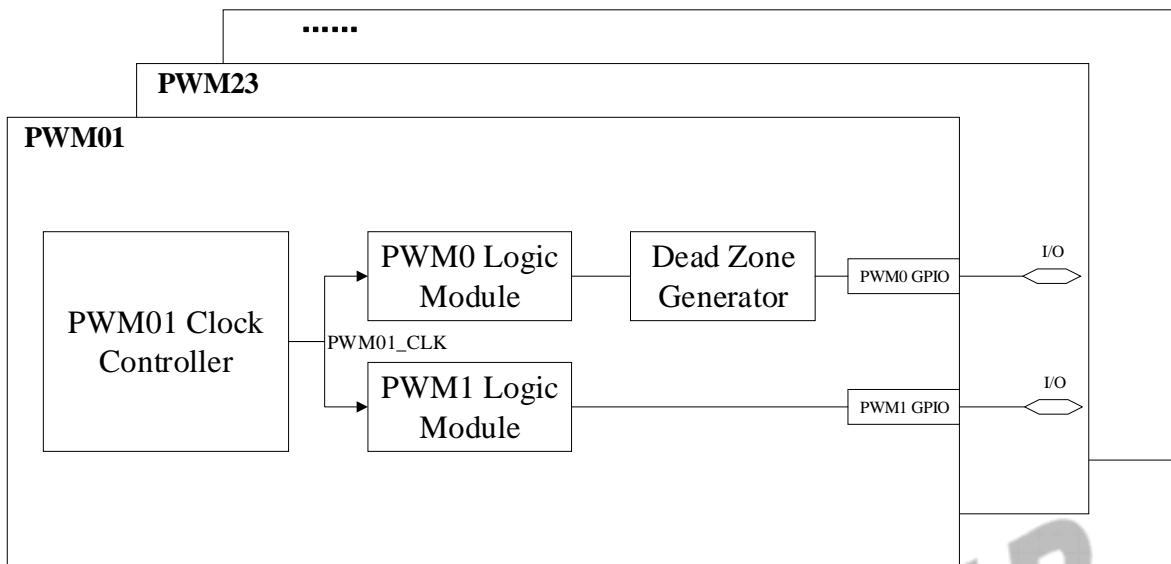
The Pulse Width Modulation (PWM) module can output the configurable PWM waveforms and measure the external input waveforms.

The PWM has the following features:

- Supports 12 independent PWM channels (PWM0 to PWM11)
 - 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 6 complementary pairs output
 - PWM01 pair (PWM0 + PWM1), PWM23 pair (PWM2 + PWM3), PWM45 pair (PWM4 + PWM5), PWM67 pair (PWM6 + PWM7), PWM89 pair (PWM8+PWM9),, and PWMab pair (PWM10+PWM11)
 - 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 12 channels capture input
 - Supports rising edge detection and falling edge detection for input waveform pulse
 - Supports pulse-width measurement for input waveform pulse

10.7.2 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 10-35 PWM Block Diagram

Each PWM pair consists of 1 clock module, 2 timer logic module, and 1 programmable dead-zone generator.

10.7.3 Functional Description

10.7.3.1 External Signals

The following table describes the external signals of the PWM.

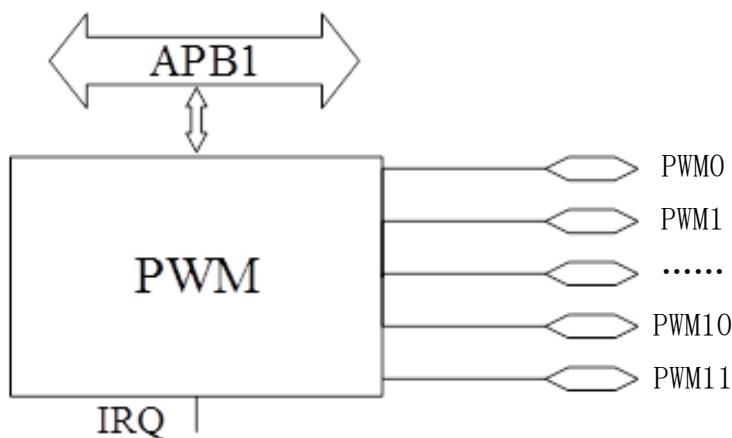
Table 10-26 PWM External Signals

| 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 |
| PWM8 | Pulse Width Module Channel8 | I/O |
| PWM9 | Pulse Width Module Channel9 | I/O |
| PWM10 | Pulse Width Module Channel10 | I/O |
| PWM11 | Pulse Width Module Channel11 | I/O |

10.7.3.2 Typical Application

The PWM module configures the registers through APB1, outputs the PWM waveform and generates the interrupt. The typical application is as follows:

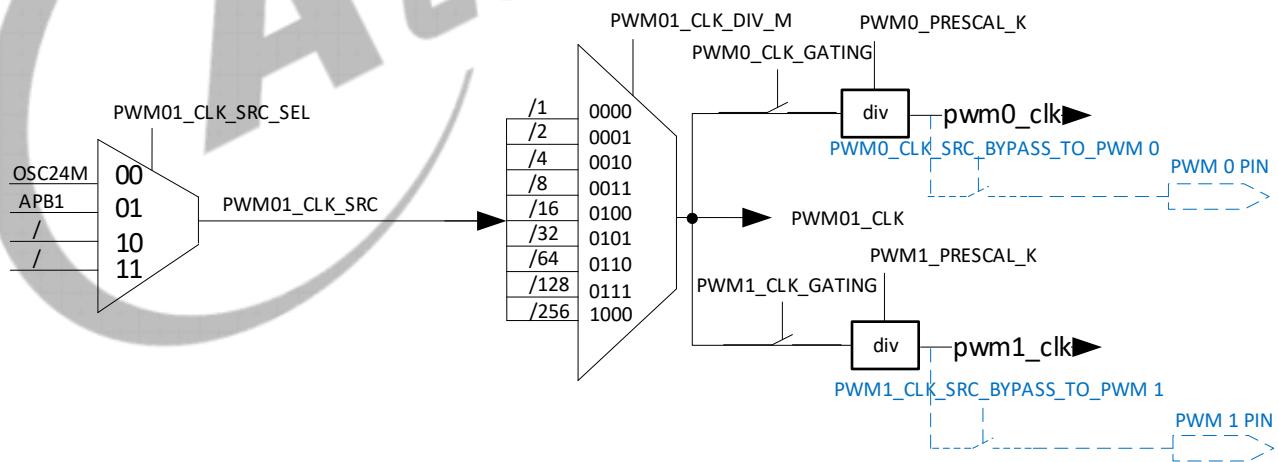
Figure 10-36 PWM Typical Application



10.7.3.3 Clock Controller

Using PWM01 as an example. The other PWM pairs are the same as PWM01.

Figure 10-37 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 ([PWMx_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 10-38 for the details about implement. At last, the output clock of the clock controller is sent to the PWM logic module.

10.7.3.4 PWM Output

Taking PWM01 as an example, the following figure 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).

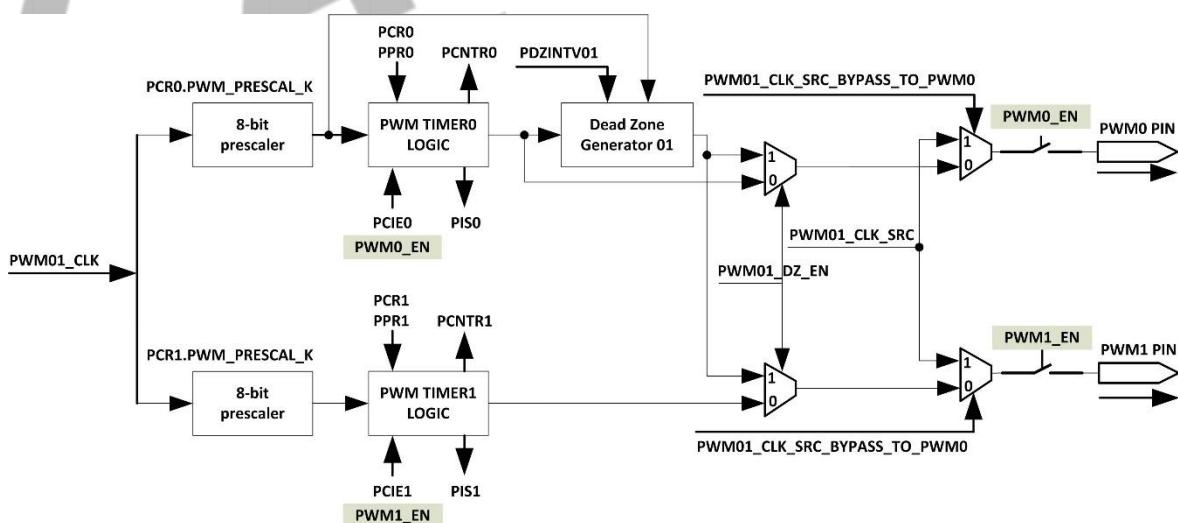
The [PWM_ENTIRE_CYCLE](#) and the [PWM_ACT_CYCLE](#) support the cache load, after PWM output is enabled, the register values of the [PWM_ENTIRE_CYCLE](#) and the [PWM_ACT_CYCLE](#) can be changed anytime, the changed value caches into the cache register. When the PCNTR counter outputs a period of PWM 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 10-38 PWM01 Output Logic Module Diagram



10.7.3.5 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.

- $\text{PCNTR} = (\text{PCNTR} == \text{PWM_ENTIRE_CYCLE}) ? 0 : \text{PCNTR} + 1$
- PCNTR starts to count by [PWM COUNTER START](#), the counter of a PWM period is $(\text{PWM_ENTIRE_CYCLE} + 1)$.
- $\text{PCNTR} > (\text{PWM_ENTIRE_CYCLE} - \text{PWM_ACT_CYCLE})$, output “active state”
- $\text{PCNTR} \leq (\text{PWM_ENTIRE_CYCLE} - \text{PWM_ACT_CYCLE})$, output “~(active state)”

Active state of PWM0 channel is high level (PCR0. PWM_ACT_STA = 1)

When $\text{PCNTR} > (\text{PPR0. PWM_ENTIRE_CYCLE} - \text{PPR0.PWM_ACT_CYCLE})$, then PWM0 outputs 1 (high level).

When $\text{PCNTR} \leq (\text{PPR0. PWM_ENTIRE_CYCLE} - \text{PPR0.PWM_ACT_CYCLE})$, then PWM0 outputs 0 (low level).

The formula of the output period and the duty-cycle for PWM are as follows.

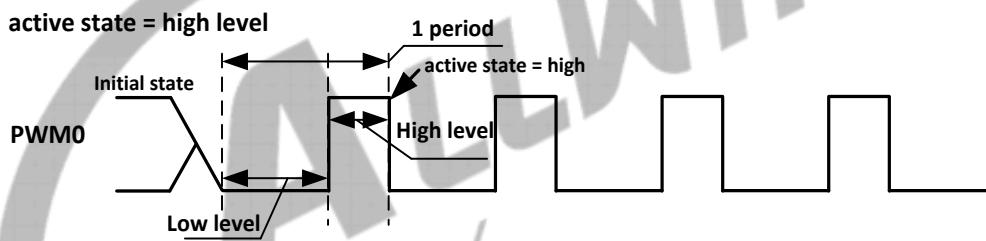
$$T_{\text{period}} = (\text{PWM01_CLK} / \text{PWM0_PREScale_K})^{-1} * (\text{PPR0.PWM_ENTIRE_CYCLE} + 1)$$

$$T_{\text{high-level}} = (\text{PWM01_CLK} / \text{PWM0_PREScale_K})^{-1} * \text{PPR0.PWM_ACT_CYCLE}$$

$$T_{\text{low-level}} = (\text{PWM01_CLK} / \text{PWM0_PREScale_K})^{-1} * (\text{PPR0.PWM_ENTIRE_CYCLE} + 1 - \text{PPR0.PWM_ACT_CYCLE})$$

$$\text{Duty-cycle} = (\text{high level time}) / (\text{1 period time}) = T_{\text{high-level}} / T_{\text{period}}$$

Figure 10-39 PWM0 High Level Active State



Active state of PWM0 channel is low level (PCR0. PWM_ACT_STA = 0)

When $\text{PCNTR} > (\text{PPR0.PWM_ENTIRE_CYCLE} - \text{PPR0.PWM_ACT_CYCLE})$, then PWM0 outputs 0.

When $\text{PCNTR} \leq (\text{PPR0.PWM_ENTIRE_CYCLE} - \text{PPR0.PWM_ACT_CYCLE})$, then PWM0 outputs 1.

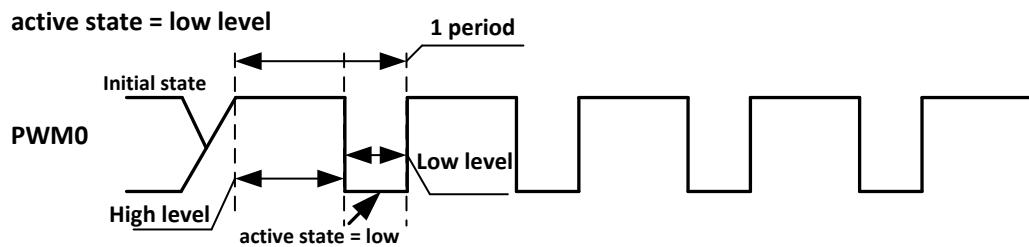
The formula of the output period and the duty-cycle for PWM are as follows.

$$T_{\text{period}} = (\text{PWM01_CLK} / \text{PWM0_PREScale_K})^{-1} * (\text{PPR0.PWM_ENTIRE_CYCLE} + 1)$$

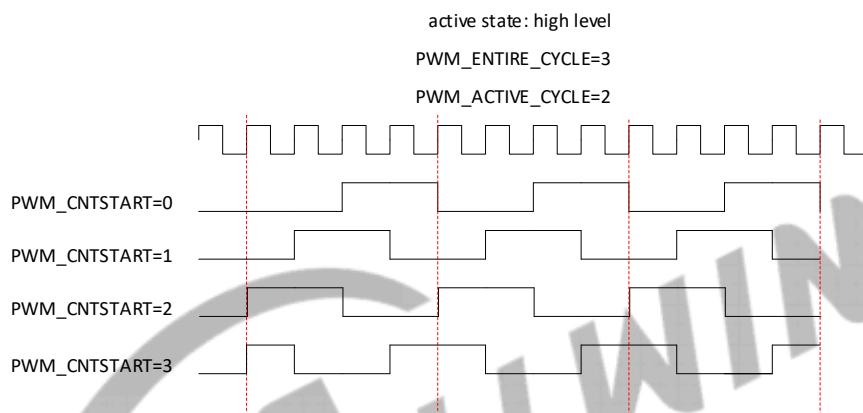
$$T_{\text{high-level}} = (\text{PWM01_CLK} / \text{PWM0_PREScale_K})^{-1} * (\text{PPR0.PWM_ENTIRE_CYCLE} + 1 - \text{PPR0.PWM_ACT_CYCLE})$$

$$T_{\text{low-level}} = (\text{PWM01_CLK} / \text{PWM0_PREScale_K})^{-1} * \text{PPR0.PWM_ACT_CYCLE}$$

$$\text{Duty-cycle} = (\text{low level time}) / (\text{1 period time}) = T_{\text{low-level}} / T_{\text{period}}$$

Figure 10-40 PWM0 Low Level Active State

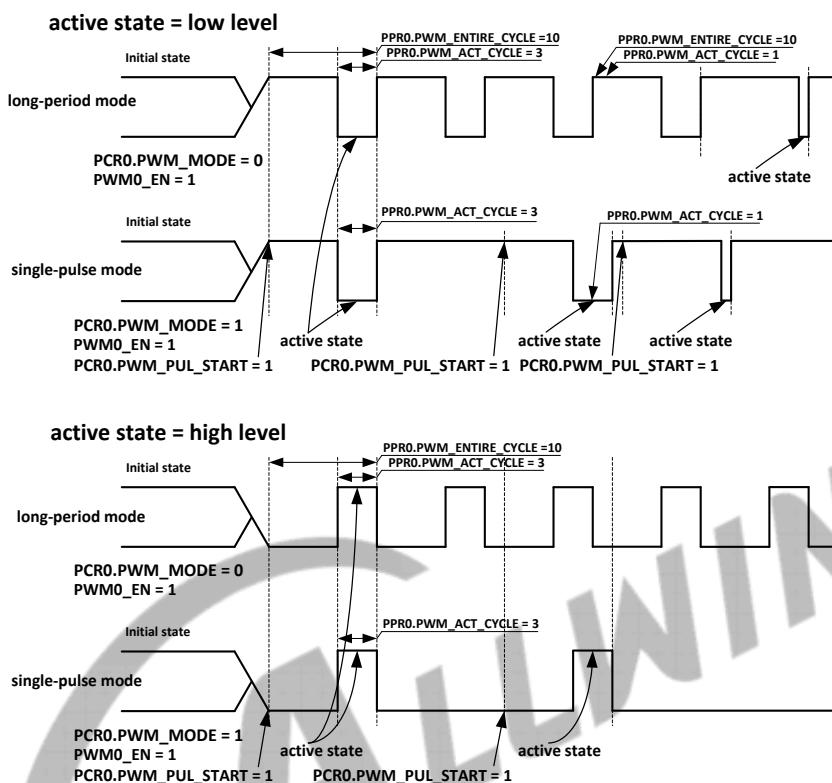
The counter of PCNTR starts from 0 by default, it can output the pulse control of the waveform by setting [PWM_COUNTER_START](#). The figure is as follows.

Figure 10-41 Phase of PWM0 High Level Active State

10.7.3.6 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. The following figure shows the PWM output waveform in pulse mode and cycle mode.

Figure 10-42 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.

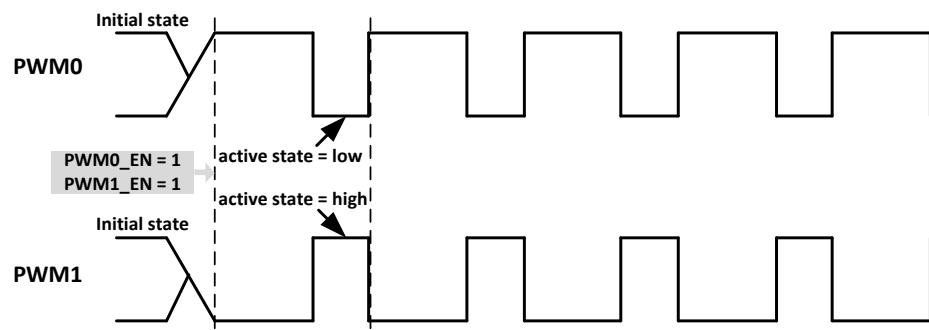
When [PCR0\[PWM_MODE\]](#) is 0, the PWMO outputs in cycle mode. When [PCR0\[PWM_MODE\]](#) is 1, the PWMO outputs in pulse mode.

Specifically, in pulse mode, after the PWMO channel enabled, [PCR0\[PWM_PUL_START\]](#) needs to be set to 1 when the PWMO 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.

10.7.3.7 Complementary Pair Output

Every PWM pair supports complementary pair output and PWM pair with dead-time. the following figure shows the complementary pair output of PWM01.

Figure 10-43 PWM01 Complementary Pair Output



The complementary pair output needs to satisfy the following conditions:

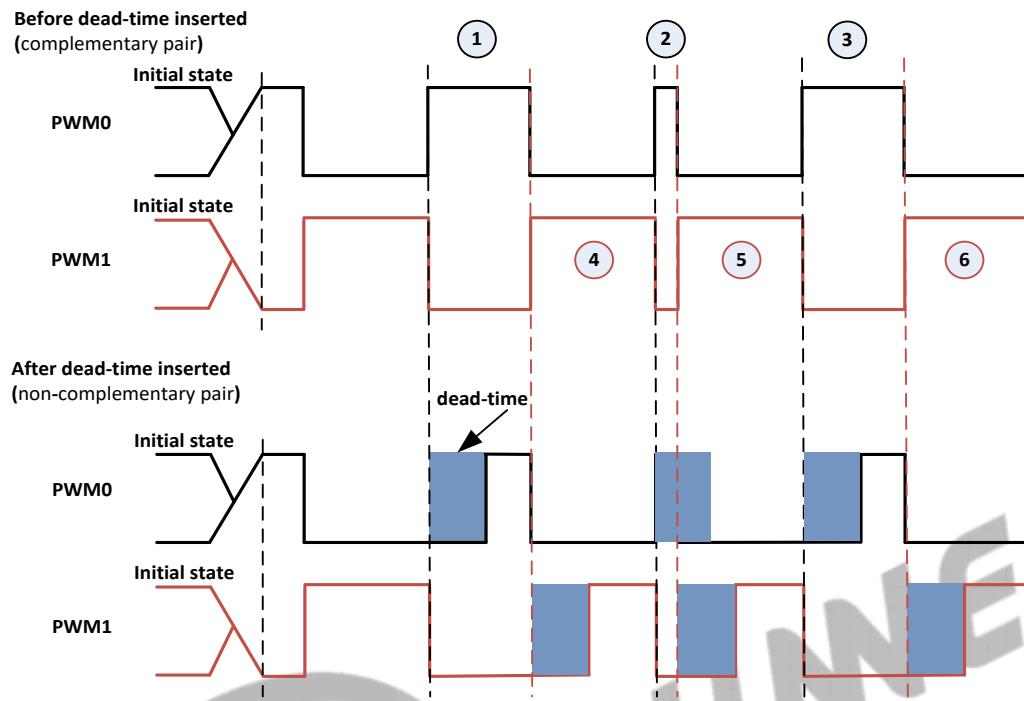
- PWM0 and PWM1 have the same clock divider, frequency, duty-cycle, and phase
- PWM0 and PWM1 have an opposite active state
- Enable the clock gating of PWM0 and PWM1 at the same time
- Enable the waveform output of PWM0 and PWM1 at the same time

10.7.3.8 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.

The following figure shows the output waveform.

Figure 10-44 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.

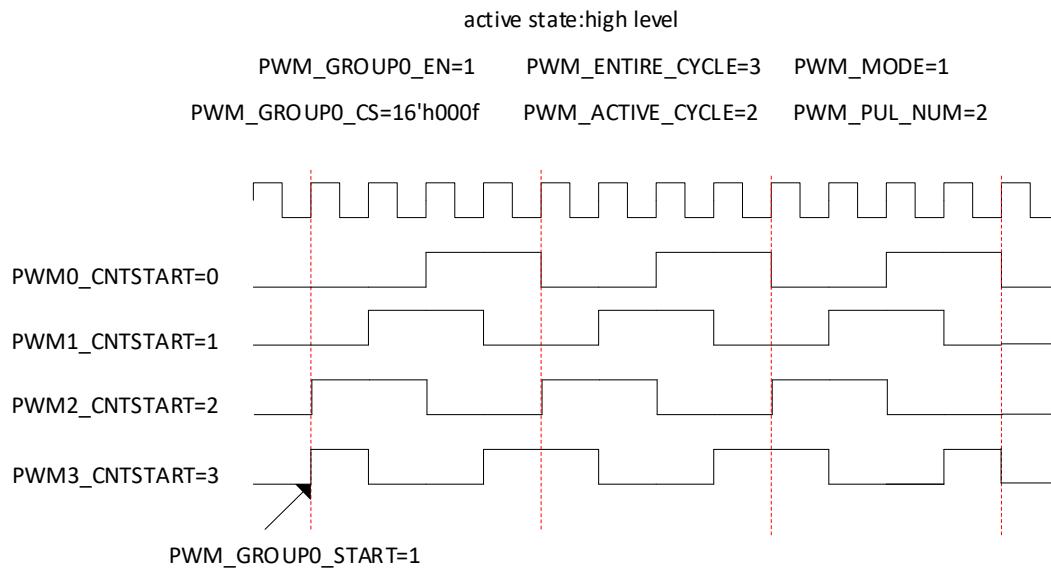
For the complementary pair of Dead Zone Generator 01, the principle of inserting dead-time is that to insert dead-time as soon as the rising edge came. If the high level time for mark② in the above figure is less than dead-time, then dead-time will override the high level. The setting of dead-time needs to consider the period and the duty-cycle of the output waveform. The dead-time formula is defined as follows:

$$\text{Dead-time} = (\text{PWM01_CLK} / \text{PWM0_PRESCALE_K}) - 1 * \text{PDZINTV01}$$

10.7.3.9 PWM Group Mode

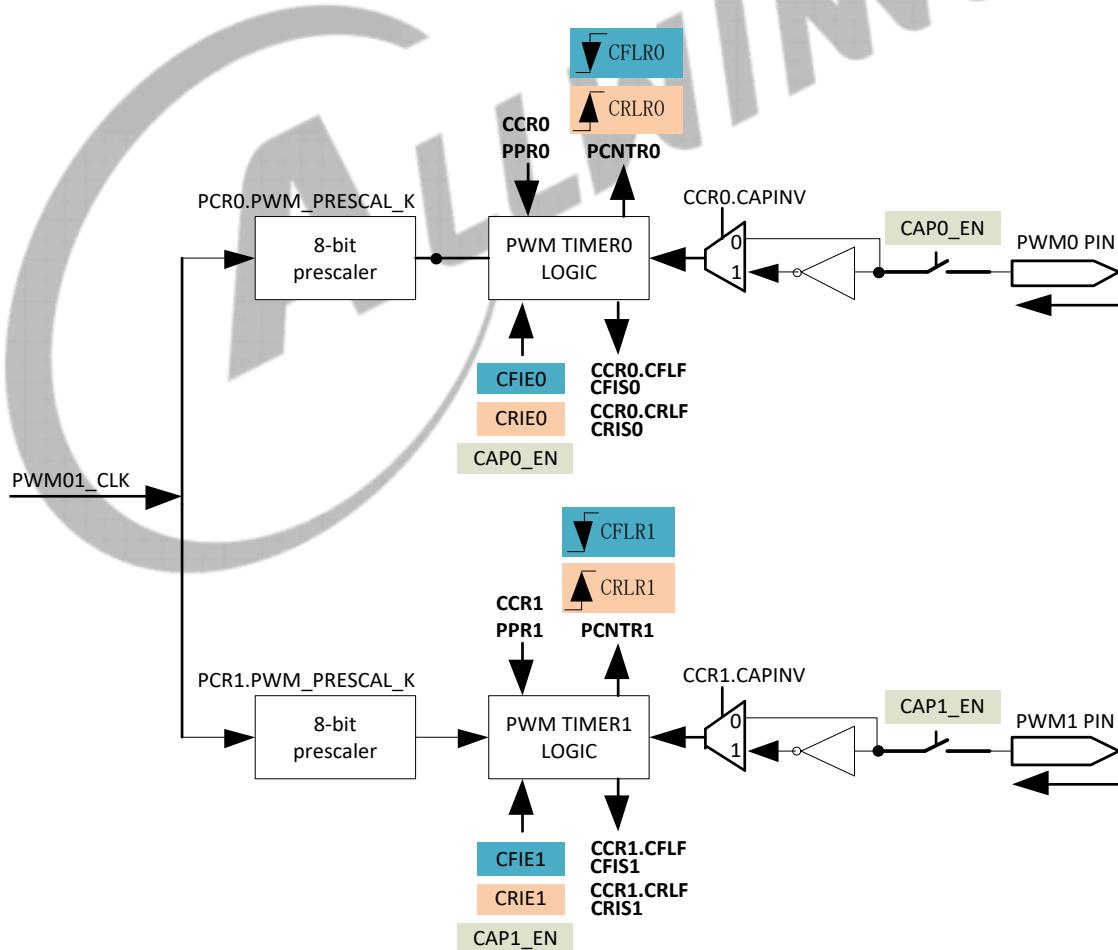
Taking PWM Group0 as an example. The same group of PWM channel is selected to work by PGR0.CS; the same [PWM_ENTIRE_CYCLE](#), [PWM_ACT_CYCLE](#) are set by the same clock configuration; the different [PWM_COUNTER_START](#) can output PWM group signals with the same duty-cycle and the different phase.

Figure 10-45 Group 0–3 PWM Signal Output



10.7.3.10 Capture Input

Figure 10-46 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

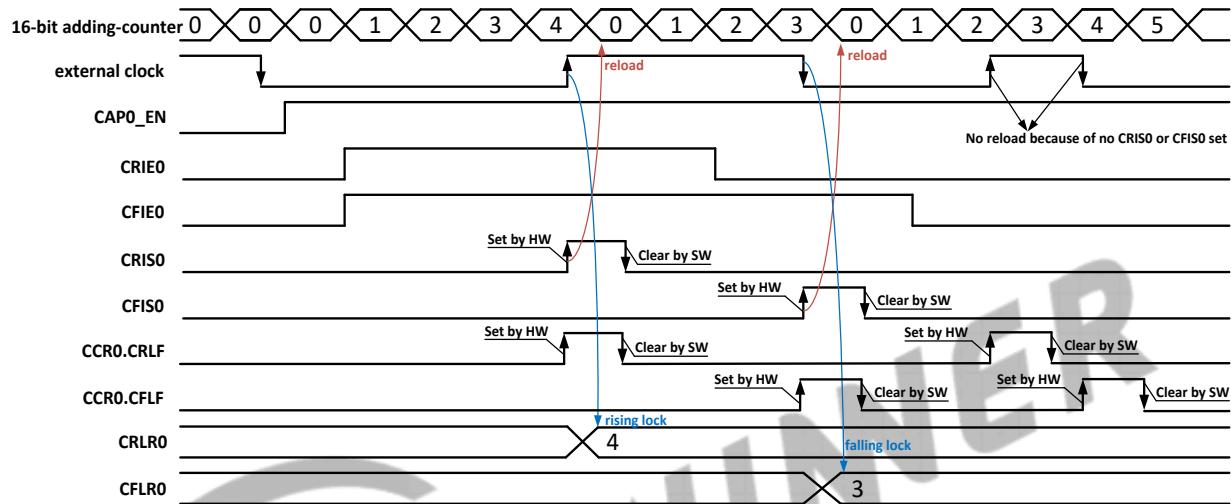
channel has one [CFLR0](#) and one [CRLR0](#) for capturing up-counter value on the falling edge and rising edge, respectively. You can calculate the period of the external clock by [CFLR0](#) and [CRLR0](#).

$$\text{Thigh-level} = (\text{PWM01_CLK} / \text{PWM0_PRESCALE_K}) - 1 * \text{CRLR0}$$

$$\text{Tlow-level} = (\text{PWM01_CLK} / \text{PWM0_PRESCALE_K}) - 1 * \text{CFLR0}$$

$$\text{Tperiod} = \text{Thigh-level} + \text{Tlow-level}$$

Figure 10-47 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 [CRIEO](#) 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 [CRIEO](#) 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\[CCLF\]](#) is set to 1. If [CFIEO](#) 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 [CFIEO](#) 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.

10.7.3.11 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.



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 [CRIEQ](#) is 1, then [CRISO](#) is set to 1; when the timer logic module of the capture channel0 captures falling edge, and [CFIEQ](#) is 1, then [CFISO](#) is set to 1.

10.7.4 Programming Guidelines

The following working mode takes PWM01 as an example, other PWM pairs and PWM01 are consistent.

10.7.4.1 Configuring Clock

1. PWM gating: When using PWM, write 1 to [PCGR](#)[PWMx_CLK_GATING].
2. PWM clock source select: Set [PCCR01](#)[PWM01_CLK_SRC] to select HOSC or APB0 clock.
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).
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.
5. PWM internal clock configuration: Set [PCR](#)[PWM_PRESCAL_K] to select any frequency division coefficient from 1 to 256.



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 regroups.

10.7.4.2 Configuring PWM

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.
2. PWM active level: Set [PCR](#)[PWM_ACT_STA] to select a low level or high level.

3. PWM duty-cycle: Configure [PPR\[PWM_ENTIRE_CYCLE\]](#) and [PPR\[PWM_ACT_CYCLE\]](#) after clock gating is opened.
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\[PWM_COUNTER_STATUS\]](#).
5. Enable PWM: Configure PER to select the corresponding PWM enable bit; when selecting pulse mode, [PCR\[PWM_PUL_START\]](#) needs to be enabled.

10.7.4.3 Configuring Deadzone

1. Set initial value: set [PDZINTV01].
2. Enable Deadzone: set [PWM01_DZ_CN].

10.7.4.4 Configuring Capture Input

1. Enable capture: Configure [CER](#) to enable the corresponding channel.
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.

10.7.5 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 |
| <u>PCCR89</u> | 0x0030 | PWM89 Clock Configuration Register |
| <u>PCCRab</u> | 0x0034 | PWMab 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 |
| <u>PDZCR89</u> | 0x0070 | PWM89 Dead Zone Control Register |

| Register Name | Offset | Description |
|---------------|---------------------------------|----------------------------------|
| PDZCRab | 0x0074 | PWMab 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= 0~11) | PWM Control Register |
| PPR | 0x0100+0x0004+N*0x0020(N= 0~11) | PWM Period Register |
| PCNTR | 0x0100+0x0008+N*0x0020(N= 0~11) | PWM Counter Register |
| PPCNTR | 0x0100+0x000C+N*0x0020(N= 0~11) | PWM Pulse Counter Register |
| CCR | 0x0100+0x0010+N*0x0020(N= 0~11) | Capture Control Register |
| CRLR | 0x0100+0x0014+N*0x0020(N= 0~11) | Capture Rise Lock Register |
| CFLR | 0x0100+0x0018+N*0x0020(N= 0~11) | Capture Fall Lock Register |

10.7.6 Register Description

10.7.6.1 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 |
| 15:12 | / | / | / |

| Offset:0x0000 | | | Register Name: PIER |
|---------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 11 | R/W | 0x0 | PCIE11 PWM channel 11 Interrupt Enable. 0: PWM channel 11 Interrupt Disable; 1: PWM channel 11 Interrupt Enable. |
| 10 | R/W | 0x0 | PCIE10 PWM channel 10 Interrupt Enable. 0: PWM channel 10 Interrupt Disable; 1: PWM channel 10 Interrupt Enable. |
| 9 | R/W | 0x0 | PCIE9 PWM channel 9 Interrupt Enable. 0: PWM channel 9 Interrupt Disable; 1: PWM channel 9 Interrupt Enable. |
| 8 | R/W | 0x0 | PCIE8 PWM channel 8 Interrupt Enable. 0: PWM channel 8 Interrupt Disable; 1: PWM channel 8 Interrupt Enable. |
| 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. |
| 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. |

| Offset:0x0000 | | | Register Name: PIER |
|---------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 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. |

10.7.6.2 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 3 Interrupt Status |
| 17 | R/W1C | 0x0 | PGIS1 PWM group 1 Interrupt Status |
| 16 | R/W1C | 0x0 | PGISO PWM group 0 Interrupt Status |
| 15:12 | / | / | / |
| 11 | R/W1C | 0x0 | PIS11 PWM channel 11 Interrupt Status. When PWM channel 11 counter reaches Entire Cycle Value, this bit is set 1 by hardware. Write 1 to clear this bit. Reads 0: PWM channel 11 interrupt is not pending. Reads 1: PWM channel 11 interrupt is pending. Writes 0: No effect. Writes 1: Clear PWM channel 11 interrupt status. |
| 10 | R/W1C | 0x0 | PIS10 PWM channel 10 Interrupt Status. When PWM channel 10 counter reaches Entire Cycle Value, this bit is set 1 by hardware. Write 1 to clear this bit. Reads 0: PWM channel 10 interrupt is not pending. Reads 1: PWM channel 10 interrupt is pending. Writes 0: No effect. Writes 1: Clear PWM channel 10 interrupt status. |

| Offset:0x0004 | | | Register Name: PISR |
|---------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 9 | R/W1C | 0x0 | <p>PIS9</p> <p>PWM channel 9 Interrupt Status. When PWM channel 9 counter reaches Entire Cycle Value, this bit is set 1 by hardware. Write 1 to clear this bit.</p> <p>Reads 0: PWM channel 9 interrupt is not pending.</p> <p>Reads 1: PWM channel 9 interrupt is pending.</p> <p>Writes 0: no effect.</p> <p>Writes 1: Clear PWM channel 9 interrupt status.</p> |
| 8 | R/W1C | 0x0 | <p>PIS8</p> <p>PWM channel 8 Interrupt Status. When PWM channel 8 counter reaches Entire Cycle Value, this bit is set 1 by hardware. Write 1 to clear this bit.</p> <p>Reads 0: PWM channel 8 interrupt is not pending.</p> <p>Reads 1: PWM channel 8 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear PWM channel 8 interrupt status.</p> |
| 7 | R/W1C | 0x0 | <p>PIS7</p> <p>PWM channel 7 Interrupt Status. When PWM channel 7 counter reaches Entire Cycle Value, this bit is set 1 by hardware. Write 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. When PWM channel 6 counter reaches Entire Cycle Value, this bit is set 1 by hardware. Write 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. When PWM channel 5 counter reaches Entire Cycle Value, this bit is set 1 by hardware. Write 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> |

| Offset:0x0004 | | | Register Name: PISR |
|---------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 4 | R/W1C | 0x0 | <p>PIS4</p> <p>PWM channel 4 Interrupt Status. When PWM channel 4 counter reaches Entire Cycle Value, this bit is set 1 by hardware. Write 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> |
| 3 | R/W1C | 0x0 | <p>PIS3</p> <p>PWM channel 3 Interrupt Status. When PWM channel 3 counter reaches Entire Cycle Value, this bit is set 1 by hardware. Write 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. When PWM channel 2 counter reaches Entire Cycle Value, this bit is set 1 by hardware. Write 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. When PWM channel 1 counter reaches Entire Cycle Value, this bit is set 1 by hardware. Write 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>PISO</p> <p>PWM channel 0 Interrupt Status. When PWM channel 0 counter reaches Entire Cycle Value, this bit is set 1 by hardware. Write 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> |

10.7.6.3 0x0010 Capture IRQ Enable Register (Default Value: 0x0000_0000)

| Offset:0x0010 | | | Register Name: CIER |
|---------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:24 | / | / | / |
| 23 | R/W | 0x0 | CFIE11 If this enable bit is set 1, when capture channel 11 captures falling edge, it generates a capture channel 11 pending. 0: Capture channel 11 fall lock Interrupt disable; 1: Capture channel 11 fall lock Interrupt enable. |
| 22 | R/W | 0x0 | CRIE11 If this enable bit is set 1, when capture channel 11 captures rising edge, it generates a capture channel 11 pending. 0: Capture channel 11 rise lock Interrupt disable; 1: Capture channel 11 rise lock Interrupt enable. |
| 21 | R/W | 0x0 | CFIE10 If this enable bit is set 1, when capture channel 10 captures falling edge, it generates a capture channel 10 pending. 0: Capture channel 10 fall lock Interrupt disable; 1: Capture channel 10 fall lock Interrupt enable. |
| 20 | R/W | 0x0 | CRIE10 If this enable bit is set 1, when capture channel 10 captures rising edge, it generates a capture channel 10 pending. 0: Capture channel 10 rise lock Interrupt disable; 1: Capture channel 10 rise lock Interrupt enable. |
| 19 | R/W | 0x0 | CFIE9 If this enable bit is set 1, when capture channel 9 captures falling edge, it generates a capture channel 9 pending. 0: Capture channel 9 fall lock Interrupt disable; 1: Capture channel 9 fall lock Interrupt enable. |
| 18 | R/W | 0x0 | CRIE9 If this enable bit is set 1, when capture channel 9 captures rising edge, it generates a capture channel 9 pending. 0: Capture channel 9 rise lock Interrupt disable; 1: Capture channel 9 rise lock Interrupt enable. |
| 17 | R/W | 0x0 | CFIE8 If this enable bit is set 1, when capture channel 8 captures falling edge, it generates a capture channel 8 pending. 0: Capture channel 8 fall lock Interrupt disable; 1: Capture channel 8 fall lock Interrupt enable. |

| Offset:0x0010 | | | Register Name: CIER |
|---------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 16 | R/W | 0x0 | CRIE8 If this enable bit is set 1, when capture channel 8 captures rising edge, it generates a capture channel 8 pending. 0: Capture channel 8 rise lock Interrupt disable; 1: Capture channel 8 rise lock Interrupt enable. |
| 15 | R/W | 0x0 | CFIE7 If this enable bit is set 1, when capture channel 7 captures falling edge, it generates a capture channel 7 pending. 0: Capture channel 7 fall lock Interrupt disable; 1: Capture channel 7 fall lock Interrupt enable. |
| 14 | R/W | 0x0 | CRIE7 If this enable bit is set 1, when capture channel 7 captures rising edge, it generates a capture channel 7 pending. 0: Capture channel 7 rise lock Interrupt disable; 1: Capture channel 7 rise lock Interrupt enable. |
| 13 | R/W | 0x0 | CFIE6 If this enable bit is set 1, when capture channel 6 captures falling edge, it generates a capture channel 6 pending. 0: Capture channel 6 fall lock Interrupt disable; 1: Capture channel 6 fall lock Interrupt enable. |
| 12 | R/W | 0x0 | CRIE6 If this enable bit is set 1, when capture channel 6 captures rising edge, it generates a capture channel 6 pending. 0: Capture channel 6 rise lock Interrupt disable; 1: Capture channel 6 rise lock Interrupt enable. |
| 11 | R/W | 0x0 | CFIE5 If this enable bit is set 1, when capture channel 5 captures falling edge, it generates a capture channel 5 pending. 0: Capture channel 5 fall lock Interrupt disable; 1: Capture channel 5 fall lock Interrupt enable. |
| 10 | R/W | 0x0 | CRIE5 If this enable bit is set 1, when capture channel 5 captures rising edge, it generates a capture channel 5 pending. 0: Capture channel 5 rise lock Interrupt disable; 1: Capture channel 5 rise lock Interrupt enable. |
| 9 | R/W | 0x0 | CFIE4 If this enable bit is set 1, when capture channel 4 captures falling edge, it generates a capture channel 4 pending. 0: Capture channel 4 fall lock Interrupt disable; 1: Capture channel 4 fall lock Interrupt enable. |

| Offset:0x0010 | | | Register Name: CIER |
|---------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 8 | R/W | 0x0 | CRIE4 If this enable bit is set 1, when capture channel 4 captures rising edge, it generates a capture channel 4 pending. 0: Capture channel 4 rise lock Interrupt disable; 1: Capture channel 4 rise lock Interrupt enable. |
| 7 | R/W | 0x0 | CFIE3 If this enable bit is set 1, when capture channel 3 captures falling edge, it generates a capture channel 3 pending. 0: Capture channel 3 fall lock Interrupt disable; 1: Capture channel 3 fall lock Interrupt enable. |
| 6 | R/W | 0x0 | CRIE3 If this enable bit is set 1, when capture channel 3 captures rising edge, it generates a capture channel 3 pending. 0: Capture channel 3 rise lock Interrupt disable; 1: Capture channel 3 rise lock Interrupt enable. |
| 5 | R/W | 0x0 | CFIE2 If this enable bit is set 1, when capture channel 2 captures falling edge, it generates a capture channel 2 pending. 0: Capture channel 2 fall lock Interrupt disable; 1: Capture channel 2 fall lock Interrupt enable. |
| 4 | R/W | 0x0 | CRIE2 If this enable bit is set 1, when capture channel 2 captures rising edge, it generates a capture channel 2 pending. 0: Capture channel 2 rise lock Interrupt disable; 1: Capture channel 2 rise lock Interrupt enable. |
| 3 | R/W | 0x0 | CFIE1 If this enable bit is set 1, when capture channel 1 captures falling edge, it generates a capture channel 1 pending. 0: Capture channel 1 fall lock Interrupt disable; 1: Capture channel 1 fall lock Interrupt enable. |
| 2 | R/W | 0x0 | CRIE1 If this enable bit is set 1, when capture channel 1 captures rising edge, it generates a capture channel 1 pending. 0: Capture channel 1 rise lock Interrupt disable; 1: Capture channel 1 rise lock Interrupt enable. |
| 1 | R/W | 0x0 | CFIE0 If this enable bit is set 1, when capture channel 0 captures falling edge, it generates a capture channel 0 pending. 0: Capture channel 0 fall lock Interrupt disable; 1: Capture channel 0 fall lock Interrupt enable. |

| Offset:0x0010 | | | Register Name: CIER |
|---------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 0 | R/W | 0x0 | <p>CRIEO</p> <p>If this enable bit is set 1, when capture channel 0 captures rising edge, it generates a capture channel 0 pending.</p> <p>0: Capture channel 0 rise lock Interrupt disable; 1: Capture channel 0 rise lock Interrupt enable.</p> |

10.7.6.4 0x0014 Capture IRQ Status Register (Default Value: 0x0000_0000)

| Offset:0x0014 | | | Register Name: CISR |
|---------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:24 | / | / | / |
| 23 | R/W1C | 0x0 | <p>CFIS11</p> <p>Capture channel 11 falling lock interrupt status.</p> <p>When capture channel 11 captures falling edge, if capture channel 11 fall lock interrupt (CFIE11) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 11 interrupt is not pending.</p> <p>Reads 1: Capture channel 11 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 11 interrupt status.</p> |
| 22 | R/W1C | 0x0 | <p>CRIS11</p> <p>Capture channel 11 rising lock interrupt status.</p> <p>When capture channel 11 captures rising edge, if capture channel 11 rise lock interrupt (CRIE11) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 11 interrupt is not pending.</p> <p>Reads 1: Capture channel 11 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 11 interrupt status.</p> |
| 21 | R/W1C | 0x0 | <p>CFIS10</p> <p>Capture channel 10 falling lock interrupt status.</p> <p>When capture channel 10 captures falling edge, if capture channel 10 fall lock interrupt (CFIE10) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 10 interrupt is not pending.</p> <p>Reads 1: Capture channel 10 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 10 interrupt status.</p> |

| Offset:0x0014 | | | Register Name: CISR |
|---------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 20 | R/W1C | 0x0 | <p>CRIS10</p> <p>Capture channel 10 rising lock interrupt status.</p> <p>When capture channel 10 captures rising edge, if capture channel 10 rise lock interrupt (CRIE10) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 10 interrupt is not pending.</p> <p>Reads 1: Capture channel 10 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 10 interrupt status.</p> |
| 19 | R/W1C | 0x0 | <p>CFIS9</p> <p>Capture channel 9 falling lock interrupt status.</p> <p>When capture channel 9 captures falling edge, if capture channel 9 fall lock interrupt (CFIE9) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 9 interrupt is not pending.</p> <p>Reads 1: Capture channel 9 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 9 interrupt status.</p> |
| 18 | R/W1C | 0x0 | <p>CRIS9</p> <p>Capture channel 9 rising lock interrupt status.</p> <p>When capture channel 9 captures rising edge, if capture channel 9 rise lock interrupt (CRIE9) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 9 interrupt is not pending.</p> <p>Reads 1: Capture channel 9 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 9 interrupt status.</p> |
| 17 | R/W1C | 0x0 | <p>CFIS8</p> <p>Capture channel 8 falling lock interrupt status.</p> <p>When capture channel 8 captures falling edge, if capture channel 8 fall lock interrupt (CFIE8) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 8 interrupt is not pending.</p> <p>Reads 1: Capture channel 8 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 8 interrupt status.</p> |

| Offset:0x0014 | | | Register Name: CISR |
|---------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 16 | R/W1C | 0x0 | <p>CRIS8</p> <p>Capture channel 8 rising lock interrupt status.</p> <p>When capture channel 8 captures rising edge, if capture channel 8 rise lock interrupt (CRIE8) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 8 interrupt is not pending.</p> <p>Reads 1: Capture channel 8 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 8 interrupt status.</p> |
| 15 | R/W1C | 0x0 | <p>CFIS7</p> <p>Capture channel 7 falling lock interrupt status.</p> <p>When capture channel 7 captures falling edge, if capture channel 7 fall lock interrupt (CFIE7) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 7 interrupt is not pending.</p> <p>Reads 1: Capture channel 7 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 7 interrupt status.</p> |
| 14 | R/W1C | 0x0 | <p>CRIS7</p> <p>Capture channel 7 rising lock interrupt status.</p> <p>When capture channel 7 captures rising edge, if capture channel 7 rise lock interrupt (CRIE7) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 7 interrupt is not pending.</p> <p>Reads 1: Capture channel 7 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 7 interrupt status.</p> |
| 13 | R/W1C | 0x0 | <p>CFIS6</p> <p>Capture channel 6 falling lock interrupt status.</p> <p>When capture channel 6 captures falling edge, if capture channel 6 fall lock interrupt (CFIE6) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 6 interrupt is not pending.</p> <p>Reads 1: Capture channel 6 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 6 interrupt status.</p> |

| Offset:0x0014 | | | Register Name: CISR |
|---------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 12 | R/W1C | 0x0 | <p>CRIS6</p> <p>Capture channel 6 rising lock interrupt status.</p> <p>When capture channel 6 captures rising edge, if capture channel 6 rise lock interrupt (CRIE6) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 6 interrupt is not pending.</p> <p>Reads 1: Capture channel 6 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 6 interrupt status.</p> |
| 11 | R/W1C | 0x0 | <p>CFIS5</p> <p>Capture channel 5 falling lock interrupt status.</p> <p>When capture channel 5 captures falling edge, if capture channel 5 fall lock interrupt (CFIE5) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 5 interrupt is not pending.</p> <p>Reads 1: Capture channel 5 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 5 interrupt status.</p> |
| 10 | R/W1C | 0x0 | <p>CRIS5</p> <p>Capture channel 5 rising lock interrupt status.</p> <p>When capture channel 5 captures rising edge, if capture channel 5 rise lock interrupt (CRIE5) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 5 interrupt is not pending.</p> <p>Reads 1: Capture channel 5 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 5 interrupt status.</p> |
| 9 | R/W1C | 0x0 | <p>CFIS4</p> <p>Capture channel 4 falling lock interrupt status.</p> <p>When capture channel 4 captures falling edge, if capture channel 4 fall lock interrupt (CFIE4) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 4 interrupt is not pending.</p> <p>Reads 1: Capture channel 4 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 4 interrupt status.</p> |

| Offset:0x0014 | | | Register Name: CISR |
|---------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 8 | R/W1C | 0x0 | <p>CRIS4</p> <p>Capture channel 4 rising lock interrupt status.</p> <p>When capture channel 4 captures rising edge, if capture channel 4 rise lock interrupt (CRIE4) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 4 interrupt is not pending.</p> <p>Reads 1: Capture channel 4 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 4 interrupt status.</p> |
| 7 | R/W1C | 0x0 | <p>CFIS3</p> <p>Capture channel 3 falling lock interrupt status.</p> <p>When capture channel 3 captures falling edge, if capture channel 3 fall lock interrupt (CFIE3) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 3 interrupt is not pending.</p> <p>Reads 1: Capture channel 3 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 3 interrupt status.</p> |
| 6 | R/W1C | 0x0 | <p>CRIS3</p> <p>Capture channel 3 rising lock interrupt status.</p> <p>When capture channel 3 captures rising edge, if capture channel 3 rise lock interrupt (CRIE3) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 3 interrupt is not pending.</p> <p>Reads 1: Capture channel 3 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 3 interrupt status.</p> |
| 5 | R/W1C | 0x0 | <p>CFIS2</p> <p>Capture channel 2 falling lock interrupt status.</p> <p>When capture channel 2 captures falling edge, if capture channel 2 fall lock interrupt (CFIE2) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 2 interrupt is not pending.</p> <p>Reads 1: Capture channel 2 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 2 interrupt status.</p> |

| Offset:0x0014 | | | Register Name: CISR |
|---------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 4 | R/W1C | 0x0 | <p>CRIS2.</p> <p>Capture channel 2 rising lock interrupt status.</p> <p>When capture channel 2 captures rising edge, if capture channel 2 rise lock interrupt (CRIE2) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 2 interrupt is not pending.</p> <p>Reads 1: Capture channel 2 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 2 interrupt status.</p> |
| 3 | R/W1C | 0x0 | <p>CFIS1</p> <p>Capture channel 1 falling lock interrupt status.</p> <p>When capture channel 1 captures falling edge, if capture channel 1 fall lock interrupt (CFIE1) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 1 interrupt is not pending.</p> <p>Reads 1: Capture channel 1 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 1 interrupt status.</p> |
| 2 | R/W1C | 0x0 | <p>CRIS1</p> <p>Capture channel 1 rising lock interrupt status.</p> <p>When capture channel 1 captures rising edge, if capture channel 1 rise lock interrupt (CRIE1) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 1 interrupt is not pending.</p> <p>Reads 1: Capture channel 1 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 1 interrupt status.</p> |
| 1 | R/W1C | 0x0 | <p>CFISO</p> <p>Capture channel 0 falling lock interrupt status.</p> <p>When capture channel 0 captures falling edge, if capture channel 0 fall lock interrupt (CFIE0) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit.</p> <p>Reads 0: Capture channel 0 interrupt is not pending.</p> <p>Reads 1: Capture channel 0 interrupt is pending.</p> <p>Writes 0: No effect.</p> <p>Writes 1: Clear capture channel 0 interrupt status.</p> |

| Offset:0x0014 | | | Register Name: CISR |
|---------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 0 | R/W1C | 0x0 | <p>CRISO Capture channel 0 rising lock interrupt status. When capture channel 0 captures rising edge, if capture channel 0 rise lock interrupt (CRIE0) is enabled, this bit is set 1 by hardware.</p> <p>Write 1 to clear this bit. Reads 0: Capture channel 0 interrupt is not pending. Reads 1: Capture channel 0 interrupt is pending. Writes 0: No effect. Writes 1: Clear capture channel 0 interrupt status.</p> |

10.7.6.5 0x0020 PWM01 Clock Configuration Register (Default Value: 0x0000_0000)

| Offset: 0x0020 | | | Register Name: PCCR01 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:9 | / | / | / |
| 8:7 | R/W | 0x00 | <p>PWM01_CLK_SRC Select PWM01 clock source 00: OSC24M 01: APB1 Others: /</p> |
| 6:4 | / | / | / |
| 3:0 | R/W | 0x0000 | <p>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: /</p> |

10.7.6.6 0x0024 PWM23 Clock Configuration Register (Default Value: 0x0000_0000)

| Offset: 0x0024 | | | Register Name: PCCR23 |
|----------------|------------|-------------|-----------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:9 | / | / | / |

| Offset: 0x0024 | | | Register Name: PCCR23 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 8:7 | R/W | 0x00 | PWM23_CLK_SRC_SEL Select PWM23 clock source 00: OSC24M 01: APB1 Others: / |
| 6:4 | / | / | / |
| 3:0 | R/W | 0x0000 | 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: / |

10.7.6.7 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 | 0x00 | PWM45_CLK_SRC_SEL Select PWM45 clock source 00: OSC24M 01: APB1 Others: / |
| 6:4 | / | / | / |
| 3:0 | R/W | 0x0000 | 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: / |

10.7.6.8 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 | 0x00 | PWM67_CLK_SRC_SEL Select PWM67 clock source 00: OSC24M 01: APB1 Others: / |
| 6:4 | / | / | / |
| 3:0 | R/W | 0x0000 | 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: / |

10.7.6.9 0x0030 PWM89 Clock Configuration Register (Default Value: 0x0000_0000)

| Offset: 0x0030 | | | Register Name: PCCR89 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:9 | / | / | / |
| 8:7 | R/W | 0x00 | PWM89_CLK_SRC_SEL Select PWM89 clock source 00: OSC24M 01: APB1 Others: / |
| 6:4 | / | / | / |

| Offset: 0x0030 | | | Register Name: PCCR89 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 3:0 | R/W | 0x0000 | PWM89_CLK_DIV_M PWM89 clock divide M 0000: /1 0001: /2 0010: /4 0011: /8 0100: /16 0101: /32 0110: /64 0111: /128 1000: /256 others: / |

10.7.6.10 0x0034 PWMab Clock Configuration Register (Default Value: 0x0000_0000)

| Offset: 0x0034 | | | Register Name: PCCRab |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:9 | / | / | / |
| 8:7 | R/W | 0x00 | PWMab_CLK_SRC_SEL Select PWMab clock source 00: OSC24M 01: APB1 Others: / |
| 6:4 | / | / | / |
| 3:0 | R/W | 0x0000 | PWMab_CLK_DIV_M PWMab clock divide M 0000: /1 0001: /2 0010: /4 0011: /8 0100: /16 0101: /32 0110: /64 0111: /128 1000: /256 others: / |

10.7.6.11 0x0040 PWM Clock Gating Register (Default Value: 0x0000_0000)

| Offset: 0x0040 | | | Register Name: PCGR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:28 | / | / | / |
| 27 | R/W | 0x0 | PWM11_CLK_BYPASS Bypass clock source (after pre-scale) to PWM11 output 0: not bypass 1: bypass |
| 26 | R/W | 0x0 | PWM10_CLK_BYPASS Bypass clock source (after pre-scale) to PWM10 output 0: not bypass 1: bypass |
| 25 | R/W | 0x0 | PWM9_CLK_BYPASS Bypass clock source (after pre-scale) to PWM9 output 0: not bypass 1: bypass |
| 24 | R/W | 0x0 | PWM8_CLK_BYPASS Bypass clock source (after pre-scale) to PWM8 output 0: not bypass 1: bypass |
| 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 |
| 19 | R/W | 0x0 | PWM3_CLK_BYPASS Bypass clock source (after pre-scale) to PWM3 output 0: not bypass 1: bypass |

| Offset: 0x0040 | | | Register Name: PCGR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 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 |
| 15:12 | / | / | / |
| 11 | R/W | 0x0 | PWM11_CLK_GATING Gating clock for PWM11 0: Mask 1: Pass |
| 10 | R/W | 0x0 | PWM10_CLK_GATING Gating clock for PWM10 0: Mask 1: Pass |
| 9 | R/W | 0x0 | PWM9_CLK_GATING Gating clock for PWM9 0: Mask 1: Pass |
| 8 | R/W | 0x0 | PWM8_CLK_GATING Gating clock for PWM8 0: Mask 1: Pass |
| 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 |

| Offset: 0x0040 | | | Register Name: PCGR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 4 | R/W | 0x0 | PWM4_CLK_GATING Gating clock for PWM4 0: Mask 1: Pass |
| 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 |

10.7.6.12 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 | PDZINTV01 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. |

10.7.6.13 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. |

| Offset: 0x0064 | | | Register Name: PDZCR23 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 7:1 | / | / | / |
| 0 | R/W | 0x0 | PWM23_DZ_EN PWM23 Dead Zone enable 0: Dead Zone disable 1: Dead Zone enable. |

10.7.6.14 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. |

10.7.6.15 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 | / | / | / |
| 0 | R/W | 0x0 | PWM67_DZ_EN PWM67 Dead Zone enable 0: Dead Zone disable 1: Dead Zone enable. |

10.7.6.16 0x0070 PWM89 Dead Zone Control Register (Default Value: 0x0000_0000)

| Offset: 0x0070 | | | Register Name: PDZCR89 |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:8 | R/W | 0x0 | PWM89_DZ_INTV PWM89 Dead Zone interval value. |
| 7:1 | / | / | / |

| Offset: 0x0070 | | | Register Name: PDZCR89 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 0 | R/W | 0x0 | PWM89_DZ_EN PWM89 Dead Zone enable 0: Dead Zone disable 1: Dead Zone enable. |

10.7.6.17 0x0074 PWMab Dead Zone Control Register (Default Value: 0x0000_0000)

| Offset: 0x0074 | | | Register Name: PDZCRab |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:8 | R/W | 0x0 | PWMab_DZ_INTV PWMab Dead Zone interval value. |
| 7:1 | / | / | / |
| 0 | R/W | 0x0 | PWMab_DZ_EN PWMab Dead Zone enable 0: Dead Zone disable 1: Dead Zone enable. |

10.7.6.18 0x0080 PWM Enable Register (Default Value: 0x0000_0000)

| Offset: 0x0080 | | | Register Name: PER |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:12 | / | / | / |
| 11 | R/W | 0x0 | PWM11_EN PWM Channel 11 Enable 0: PWM disable 1: PWM enable. |
| 10 | R/W | 0x0 | PWM10_EN PWM Channel 10 Enable 0: PWM disable 1: PWM enable. |
| 9 | R/W | 0x0 | PWM9_EN PWM Channel 9 Enable 0: PWM disable 1: PWM enable. |
| 8 | R/W | 0x0 | PWM8_EN PWM Channel 8 Enable 0: PWM disable 1: PWM enable. |

| Offset: 0x0080 | | | Register Name: PER |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 7 | R/W | 0x0 | PWM7_EN PWM Channel 7 Enable 0: PWM disable 1: PWM enable. |
| 6 | R/W | 0x0 | PWM6_EN PWM Channel 6 Enable 0: PWM disable 1: PWM enable. |
| 5 | R/W | 0x0 | PWM5_EN PWM Channel 5 Enable 0: PWM disable 1: PWM enable. |
| 4 | R/W | 0x0 | PWM4_EN PWM Channel 4 Enable 0: PWM disable 1: PWM enable. |
| 3 | R/W | 0x0 | PWM3_EN PWM Channel 3 Enable 0: PWM disable 1: PWM enable. |
| 2 | R/W | 0x0 | PWM2_EN PWM Channel 2 Enable 0: PWM disable 1: PWM enable. |
| 1 | R/W | 0x0 | PWM1_EN PWM Channel 1 Enable 0: PWM disable 1: PWM enable. |
| 0 | R/W | 0x0 | PWM0_EN PWM Channel 0 Enable 0: PWM disable 1: PWM enable. |

10.7.6.19 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 | PWMG0_START PWM channels selected in PWMG0_CS start to output PWM waveform at the same time. |

| Offset: 0x0090 | | | Register Name: PGR0 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 16 | R/W | 0x0 | PWMG0_EN PWM Group0 Enable. |
| 15:0 | R/W | 0x0 | PWMG0_CS If bit[i] is set, PWM i is selected as one channel of PWM Group0. |

10.7.6.20 0x0094 PWM Group1 Register (Default Value: 0x0000_0000)

| Offset: 0x0094 | | | Register Name: PGR1 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:18 | / | / | / |
| 17 | R/WAC | 0x0 | PWMG1_START PWM channels selected in PWMG1_CS 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, PWM i is selected as one channel of PWM Group1. |

10.7.6.21 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 PWM channels selected in PWMG2_CS 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, PWM i is selected as one channel of PWM Group2. |

10.7.6.22 0x009C PWM Group3 Register (Default Value: 0x0000_0000)

| Offset: 0x009C | | | Register Name: PGR3 |
|----------------|------------|-------------|---------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:18 | / | / | / |

| Offset: 0x009C | | | Register Name: PGR3 |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 17 | R/WAC | 0x0 | PWMG3_START PWM channels selected in PWMG3_CS 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, PWM i is selected as one channel of PWM Group3. |

10.7.6.23 0x00C0 Capture Enable Register (Default Value: 0x0000_0000)

| Offset: 0x00C0 | | | Register Name: CER |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:12 | / | / | / |
| 11 | R/W | 0x0 | CAP11_EN When enable capture function, the 16-bit up-counter starts working and capture channel11 is permitted to capture external falling edge or rising edge. 0: Capture disable 1: Capture enable. |
| 10 | R/W | 0x0 | CAP10_EN When enable capture function, the 16-bit up-counter starts working and capture channel10 is permitted to capture external falling edge or rising edge. 0: Capture disable 1: Capture enable. |
| 9 | R/W | 0x0 | CAP9_EN When enable capture function, the 16-bit up-counter starts working and capture channel9 is permitted to capture external falling edge or rising edge. 0: Capture disable 1: Capture enable. |
| 8 | R/W | 0x0 | CAP8_EN When enable capture function, the 16-bit up-counter starts working and capture channel8 is permitted to capture external falling edge or rising edge. 0: Capture disable 1: Capture enable. |

| Offset: 0x00C0 | | | Register Name: CER |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 7 | R/W | 0x0 | CAP7_EN When enable capture function, the 16-bit up-counter starts working and capture channel7 is permitted to capture external falling edge or rising edge. 0: Capture disable 1: Capture enable. |
| 6 | R/W | 0x0 | CAP6_EN When enable capture function, the 16-bit up-counter starts working and capture channel6 is permitted to capture external falling edge or rising edge. 0: Capture disable 1: Capture enable. |
| 5 | R/W | 0x0 | CAP5_EN When enable capture function, the 16-bit up-counter starts working and capture channel5 is permitted to capture external falling edge or rising edge. 0: Capture disable 1: Capture enable. |
| 4 | R/W | 0x0 | CAP4_EN When enable capture function, the 16-bit up-counter starts working and capture channel4 is permitted to capture external falling edge or rising edge. 0: Capture disable 1: Capture enable. |
| 3 | R/W | 0x0 | CAP3_EN When enable capture function, the 16-bit up-counter starts working and capture channel3 is permitted to capture external falling edge or rising edge. 0: Capture disable 1: Capture enable. |
| 2 | R/W | 0x0 | CAP2_EN When enable capture function, the 16-bit up-counter starts working and capture channel2 is permitted to capture external falling edge or rising edge. 0: Capture disable 1: Capture enable. |
| 1 | R/W | 0x0 | CAP1_EN When enable capture function, the 16-bit up-counter starts working and capture channel1 is permitted to capture external falling edge or rising edge. 0: Capture disable 1: Capture enable. |

| Offset: 0x00C0 | | | Register Name: CER |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 0 | R/W | 0x0 | <p>CAPO_EN.</p> <p>When enable capture function, the 16-bit up-counter starts working and capture channel is permitted to capture external falling edge or rising edge.</p> <p>0: Capture disable 1: Capture enable.</p> |

10.7.6.24 0x0100+N*0x0020(N= 0~11) PWM Control Register (Default Value: 0x0000_0000)

| Offset: 0x0100+0x0000+N*0x0020(N= 0~11) | | | Register Name: PCR |
|---|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | R/W | 0x0 | <p>PWM_PUL_NUM</p> <p>In pulse mode, PWM output pulse for PWM_CYCLE_NUM+1 times and then stop.</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> |
| 10 | R/WAC | 0x0 | <p>PWM_PUL_START</p> <p>PWM pulse output start</p> <p>0: No effect 1: output pulse for PWM_CYCLE_NUM+1</p> <p>After finishing configuration for outputting pulse, set this bit once and then PWM would output waveform. After the waveform is finished, the bit will be cleared automatically.</p> |
| 9 | R/W | 0x0 | <p>PWM_MODE</p> <p>PWM output mode select</p> <p>0: cycle mode 1: pulse mode.</p> |
| 8 | R/W | 0x0 | <p>PWM_ACT_STA</p> <p>PWM active state</p> <p>0: Low Level 1: High Level.</p> |

| Offset: 0x0100+0x0000+N*0x0020(N=0~11) | | | Register Name: PCR |
|---|-------------------|--------------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 7:0 | R/W | 0x0 | <p>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.</p> |

10.7.6.25 0x0104+N*0x0020(N= 0~11) PWM Period Register (Default Value: 0x0000_0000)

| Offset:0x0100+0x0004+N*0x0020(N=0~11) | | | Register Name: PPR |
|--|-------------------|--------------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | R/W | 0x0 | <p>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 need to be modified dynamically, the PCLK should be faster than the PWM CLK .</p> |
| 15:0 | R/W | 0x0 | <p>PWM_ACT_CYCLE Number of the active cycles in the PWM clock 0 = 0 cycle 1 = 1 cycles N = N cycles</p> |

10.7.6.26 0x0108+N*0x0020(N= 0~11) PWM Counter Register (Default Value: 0x0000_0000)

| Offset: 0x0100+0x0008+N*0x0020(N=0~11) | | | 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 16bit up-counter. |

10.7.6.27 0x010C+N*0x0020(N= 0~11) PWM Pulse Counter Register (Default Value: 0x0000_0000)

| Offset: 0x0100+0x000C+N*0x0020(N= 0~11) | | | 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. |

10.7.6.28 0x0110+N*0x0020(N= 0~11) Capture Control Register (Default Value: 0x0000_0000)

| Offset: 0x0100+0x0010+N*0x0020(N= 0~11) | | | Register Name: CCR |
|---|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:5 | / | / | / |
| 4 | R/W1C | 0x0 | CRLF When capture channel captures rising edge, the 16-bit up-counter's current value 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 capture channel captures falling edge, the 16-bit up-counter's current value 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 Inversing the signal inputted from capture channel before capture channel's 16bit counter. 0: not inverse 1: inverse |

10.7.6.29 0x0114+N*0x0020(N= 0~11) Capture Rise Lock Register (Default Value: 0x0000_0000)

| Offset: 0x0100+0x0014+N*0x0020(N= 0~11) | | | Register Name: CRLR |
|---|------------|-------------|---------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |

| Offset: 0x0100+0x0014+N*0x0020(N=0~11) | | | Register Name: CRLR |
|--|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 15:0 | R | 0x0 | CRLR When capture channel captures rising edge, the 16-bit up-counter's current value is latched to this register. |

10.7.6.30 0x0118+N*0x0020(N= 0~11) Capture Fall Lock Register (Default Value: 0x0000_0000)

| Offset: 0x0100+0x0018+N*0x0020(N=0~11) | | | Register Name: CFLR |
|--|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:16 | / | / | / |
| 15:0 | R | 0x0 | CFLR When capture channel captures falling edge, the 16-bit up-counter's current value is latched to this register. |

10.8 Wiegand Protocol

10.8.1 Overview

The Wiegand Protocol is a communication protocol developed by Motorola, which is commonly used in the card readers and IC cards of access control systems. The WIEGAND_CTRL is a data reading control module for external Wiegand devices. CPU can configure this module through the APB bus to receive the signals of DATA0 and DATA1 sent by external Wiegand devices.

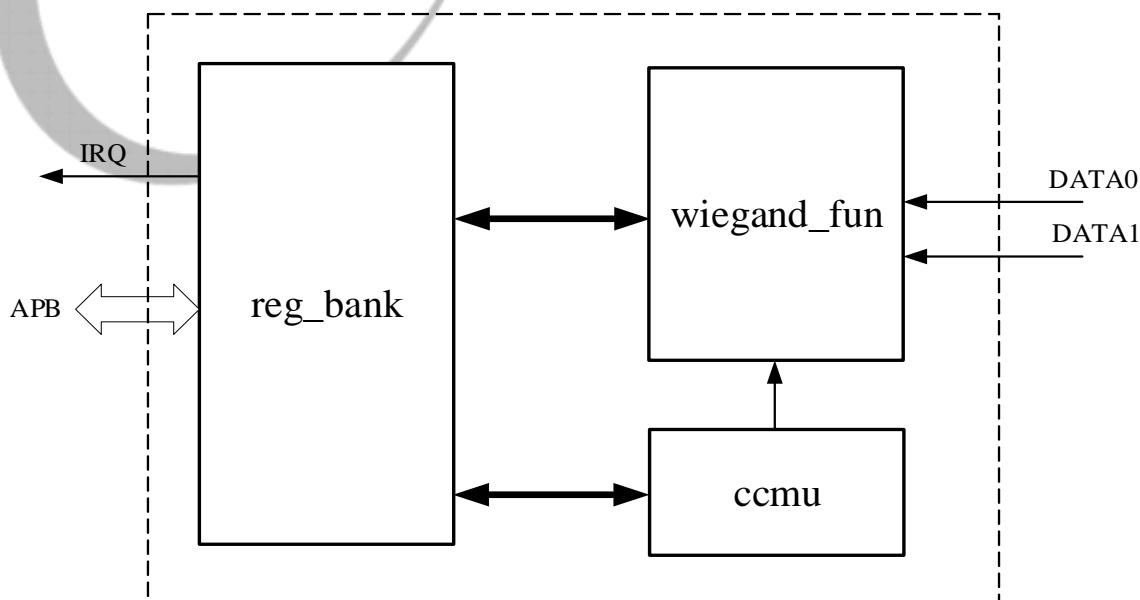
The features of The Wiegand Protocol are as follows:

- Support 26-bit & 34-bit Wiegand protocol formats.
- Support the configurable duration and period of Wiegand data waveforms.
- Support CPU to access the configuration register and data transmission through the APB bus.
- Support receive completion interrupt, parity error reporting interrupt and receive timeout interrupt.
- Support observing the number and contents of received data during the timeout interrupt.
- Support configuring the highest and lowest parity bit polarity.
- Support configuring the polarity of DATA0 and DATA1.
- Support adjusting the frequency dividing coefficients of internal counting reference clock.

10.8.2 Block Diagram

As shown in Figure 10-48, the WIEGAND_CTRL module is composed of the register file REG_BANK, the Wiegand protocol function module WIEGAND_FUN, the clock module ccmu and some interfaces.

Figure 10-48 Block Diagram of WIEGAND_CTRL Module



10.8.3 Functional Description

10.8.3.1 External Signals

The following table describes the external signals of WIEGAND_CTRL.

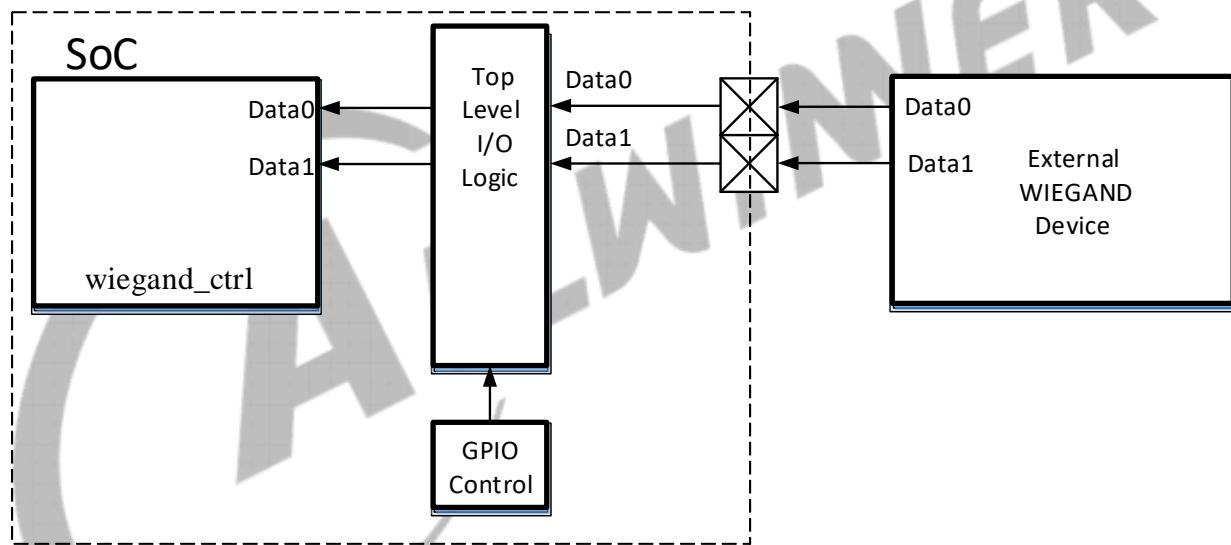
Table 10-27 External Signals of WIEGAND_CTRL

| Wiegand Interface Signal | Description | Type | Width |
|--------------------------|---|------|-------|
| DATA0 | Input Signal Line 0 of Wiegand Protocol | I | 1 |
| DATA1 | Input Signal Line 1 of Wiegand Protocol | I | 1 |

10.8.3.2 Typical Application

As shown in Figure 10-49, the GPIO controller must be correctly configured to connect the WIEGAND_CTRL module to external devices.

Figure 10-49 Application Diagram of WIEGAND_CTRL Module



26-bit Wiegand Protocol

As shown in Table 10-28, the bits 1 to 8 are group codes (HID codes of electronic card), which have 8 binary data and 256 states.

The bits 9 to 24 are identification codes (PID codes of electronic card), which have 16 binary data and 65536 states.

Bit 0 is the even parity bit of bits 1 to 12; bit 25 is the odd parity bit of bits 13 to 24.

HID code is the abbreviation of Hidden ID Code; PID code is the abbreviation of Public ID Code.

Table 10-28 Data Format of Wiegand (26-bit)

| Bits | Meaning |
|-------|---------------------------------|
| Bit 0 | Even parity bit of bits 1 to 12 |

| Bits | Meaning |
|-----------|---|
| Bits 1-8 | Group Code (PID code of electronic card) |
| Bits 9-24 | Identification Code (PID code of electronic card) |
| Bit 25 | Odd parity bit of bits 13 to 24 |

The Wiegand transmitter outputs 26-bit data through DATA0 and DATA1, which are kept at a high level of +5V when there is no data. If the output is 0, the DATA0 will be at a low level for some time; if the output is 1, the DATA1 will be at a low level for some time. To prevent the data reading conflicts, the minimum interval between the Wiegand outputs of two electronic cards is 0.25 seconds. For example, in the bus or subway, if multiple cards are stacked together, any of them may not be successfully swiped.

To ensure that card numbers are correct and not mixed up, they are verified by means of odd parity or even parity.

Odd Parity: During the bit transmission, an additional bit is taken as a check bit. If the number of "1" in the total data is even, the check bit is "1", otherwise the check bit is "0". The received data will be checked according to the requirements of odd parity. If the number of "1" is odd, it indicates the transmission is correct, otherwise the transmission is wrong.

Even Parity: During the bit transmission, an additional bit is taken as a check bit. If the number of "1" in the total data is odd, the check bit is "1", otherwise the check bit is "0". The received data will be checked according to the requirements of even parity. If the number of "1" is even, it indicates the transmission is correct, otherwise the transmission is wrong.

34-bit Wiegand Protocol

As shown in Table 10-29, the bits 1 to 16 are group codes (HID code of electronic card), which have 8 binary data and 256 states.

The bits 17 to 32 are identification codes (PID code of electronic card), which have 16 binary data and 65536 states.

Bit 0 is the even party bit of bits 1 to 16; bit 33 is the odd parity bit of bits 17 to 32.

HID code is the abbreviation of Hidden ID Code; PID code is the abbreviation of Public ID Code.

Table 10-29 Data Format of Wiegand (34-bit)

| Bits | Meaning |
|------------|---|
| Bit 0 | Even party bit of bits 1 to 16 |
| Bits 1-16 | Group Code (PID code of electronic card) |
| Bits 17-32 | Identification Code (PID code of electronic card) |
| Bit 33 | Odd parity bit of bits 17 to 32 |

Configurable Parity Check Polarity and Input Port Polarity

This module supports configurable parity check polarity and input port polarity for better compatibility. As stipulated by the 26-bit & 34-bit Wiegand Protocol, high bit is odd check bit while low bit is even check bit. This

module supports configuring the parity type of high and low parity bits. Please see the Section 6.10 for more details.

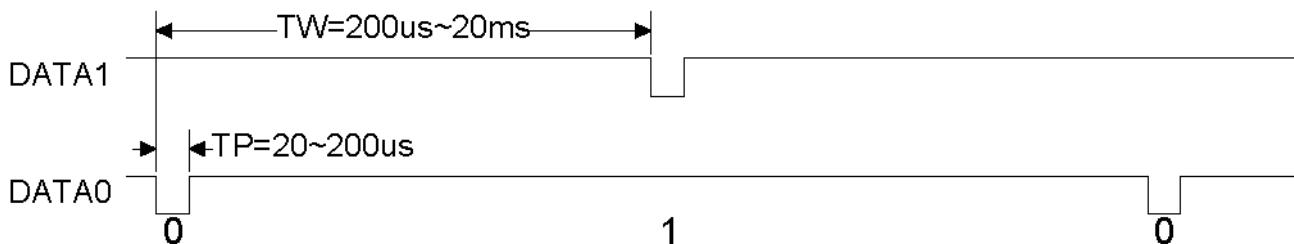
The input signal polarity can be configured by the WIEGAND_DBP register to cope with the situation that the signals of DATA0 and DATA1 are reversely connected.

10.8.3.3 Functions

Wiegand protocol, also known as Wiegand code, only needs two cables during data transmission. One is DATA0, and the other is DATA1. The protocol stipulates that the two cables are at a high level when there is no data. As shown in Figure 10-50, the DATA0 at a low level represents 0, and the DATA1 at a low level represents 1. (Low level signal is less than 1V, and high level signal is more than 4V) The pulse width ranges are 20us-200us, and the intervals of two pulses are 200us-20ms.

TP is the duration of low level, and TW is the period of low level. Both of them are flexible in a range, among which different devices may have their TP and TW parameter configuration. Therefore, to achieve more reasonable signal detection, this module provides register port for parameter configuration in accordance with peripheral values of TP and TW. Please see the Chapter 6 for more details.

Figure 10-50 Data Signal Oscilloscope



10.8.3.4 Cautions and Notes

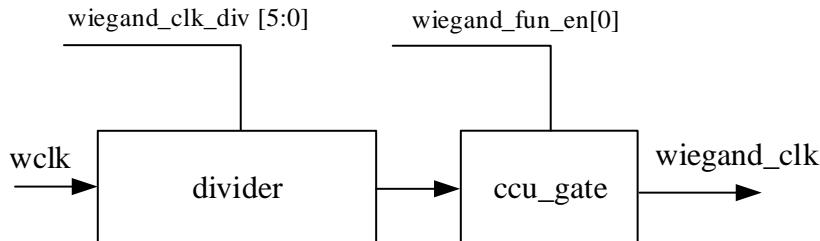
SCLK Clock Frequency Division Configuration

The bit [5:0] of WIEGAND_CLK_DIV register is used for configuring the frequency dividing coefficients. The equation of the frequency-divided WIEGAND_CLK is $\text{WIEGAND_CLK} = (\text{WCLK}) / (\text{WIEGAND_CLK_DIV}[5:0])$. The output of a frequency divider is controlled by clock gating, and the control signal sources from the data that is written from the APB interface to WIEGAND_FUN_EN register. When CPU configures WIEGAND_FUN_EN register as 1 through the APB bus, the clock gating of frequency divider will be turned on and output the frequency-divided WIEGAND_CLK. The logic diagram of clock gating is shown in Figure 10-51, in which CCU_GATE is the clock logic unit of CCMU library.

This clock divider supports the two frequency divisions at least. If the value of WIEGAND_CLK_DIV is 0 or 1, it equals 2, namely, two frequency divisions. This clock divider supports 48 frequency divisions at most. If the value of WIEGAND_CLK_DIV is larger than 48, it equals 48 frequency divisions. This is because that when the frequency dividing coefficient equals 48, the WCLK before frequency division is 24MHz, and the WIEGAND_CLK after frequency division is 0.5MHz with a period of 2us. As stipulated by Wiegand protocol, the minimum pulse width of Wiegand data is 20us. Therefore, the larger frequency dividing coefficients (more than 48 frequency

divisions) will lead to clock frequencies not enough to collect Wiegand data. Therefore, the value of WIEGAND_CLK_DIV register ranges from 0 to 48.

Figure 10-51 Diagram of WCLK Clock Module



Error Flag of Parity Check

When receiving complete data, the module will conduct internal parity check and read the original parity information received this time by using the WIEGAND_RXC register. If a check error is found when the module checks the received data and parity bit, the corresponding bit of the WIEGAND_CBSR register will be set to 1. The high parity bit and low parity bit respectively corresponds to the CBS_H and CBS_L of WIEGAND_CBSR register.

Timeout Interrupt

As the Wiegand protocol stipulates that the interval of two adjacent data waveforms is 200us-30ms, a timeout interrupt will be triggered if the next waveform beyond this interval isn't detected. This interrupt flag corresponds to the TRIS bit of WIEGAND_ISR register. The threshold of triggering this interrupt is set by the WIEGAND_TW register. When this interrupt is triggered, the module can check the number of receive completion bits and the data value by reading the WIEGAND_TDNR and the WIEGAND_TDR. The interrupt flag can be cleared by writing 1.

Range Setting of TP and TW

The following table is the data in case of WCLK=24MHz.

Table 10-30 Range Setting of WIEGAND_TP Register and WIEGAND_TW Register under different frequency dividing coefficients

| Frequency Dividing Coefficients | WIEGAND_CLK | Period | TP Min Value (20us) | TP Max Value (200us) | TW Min Value (200us) | TW Max Value (20ms) |
|---------------------------------|-------------|---------|---------------------|----------------------|----------------------|---------------------|
| 0 | 12MHz | 0.083us | 240 | 2400 | 2400 | 240000 |
| 1 | 12MHz | 0.083us | 240 | 2400 | 2400 | 240000 |
| 2 | 12MHz | 0.083us | 240 | 2400 | 2400 | 240000 |
| ... | ... | ... | ... | ... | ... | ... |
| 24 | 1MHz | 1us | 20 | 200 | 200 | 20000 |
| ... | ... | ... | ... | ... | ... | ... |
| 48 | 0.5MHz | 2us | 10 | 100 | 100 | 10000 |

10.8.4 Programming Guidelines

10.8.4.1 Initialization Process

1. Reset release, and turn on the clock gating.
2. Configure the IO interface of DATA0 and DATA1 as the input port.
3. Configure the WIEGAND_WMR register to set the format of Wiegand Protocol as 26-bit or 34-bit.
4. Configure the WIEGAND_CLK_DIV register to set the frequency dividing coefficients of internal divided clock. As the frequency of input oscillator clock is 24MHz, it is recommended to set WIEGAND_CLK_DIV as 24 frequency division to get the 1MHz divided clock.
5. Configure WIEGAND_TP register to set the detection time of TP in microseconds (us). The detection counter clock is the 1MHz divided clock. The value of WIEGAND_TP should be smaller than the actual duration of its data pulse.
6. Configure the WIEGAND_TW register to set the detection time of TW in microseconds (us). The detection counter clock is the 1MHz divided clock. The value of WIEGAND_TW should be smaller than the actual period of its data pulse.
7. Configure the WIEGAND_ICR register to set the interrupt enable.
8. Configure the WIEGAND_CBP register to set the detection polarity of parity bit.
9. Configure the WIEGAND_DBP register to set the data polarity of DATA0 and DATA1.
10. After completing the above configurations, configure the WIEGAND_FUN_EN register to enable the WIEGAND_CTRL module.

10.8.4.2 Data Reading Process

1. When the initialization process finishes, the WIEGAND_CTRL module will continuously monitor the varying level of DATA0 and DATA1. When receiving the specified number of bits (26bit or 34bit), the module will trigger the signal of receive completion interrupt and start the parity check.
2. WIEGAND_ISR register is an interrupt flag register. CPU decides the next operations based on the IRQ interrupt signals and the different interrupt flags of WIEGAND_ISR register.
3. WIEGAND_RXD register read data while WIEGAND_RXC register read parity bits. When the format of Wiegand protocol is 26bit, the least significant bit of WIEGAND_RXD register is 24 (not included two check bits). When the format of Wiegand protocol is 34bit, the least significant bit of WIEGAND_RXD register is 32 (not included two check bits).
4. Write 1 to clear the specified bits of WIEGAND_ISR register.

10.8.5 Register List

| Module Name | Base Address |
|--------------|--------------|
| WIEGAND_CTRL | 0x02020000 |

| Register Name | Offset | Description |
|-----------------|--------|--------------------------------------|
| WIEGAND_WMR | 0x0004 | WIEGAND Work Mode Register |
| WIEGAND_TP | 0x0008 | WIEGAND TP Control Register |
| WIEGAND_TW | 0x000C | WIEGAND TW Control Register |
| WIEGAND_CLK_DIV | 0x0010 | WIEGAND Clock Divide Register |
| WIEGAND_FUN_EN | 0x0014 | WIEGAND Function Enable Register |
| WIEGAND_ICR | 0x001C | WIEGAND Interrupt Control Register |
| WIEGAND_ISR | 0x0020 | WIEGAND Interrupt Status Register |
| WIEGAND_CBSR | 0x0024 | WIEGAND Check Bit Status Register |
| WIEGAND_CBP | 0x0028 | WIEGAND Check Bit Polar Register |
| WIEGAND_DBP | 0x002C | WIEGAND Data Bit Polar Register |
| WIEGAND_DSR | 0x0034 | WIEGAND Data Status Register |
| WIEGAND_TDNR | 0x0038 | WIEGAND Timeout Data Number Register |
| WIEGAND_TDR | 0x003C | WIEGAND Timeout Data Register |
| WIEGAND_RXD | 0x0100 | WIEGAND RX DATA Register |
| WIEGAND_RXC | 0x0200 | WIEGAND RX CHECK Register |

10.8.6 Register Description

10.8.6.1 0x0004 WIEGAND Work Mode Register (Default Value: 0x0000_0000)

| Offset: 0x0004 | | | Register Name: WIEGAND_WMR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:1 | / | / | / |
| 0 | R/W | 0x0 | WM Work Mode 0: wiegand-26bit 1: wiegand-34bit |

10.8.6.2 0x0008 WIEGAND TP Control Register (Default Value: 0x0000_0000)

| Offset: 0x0008 | | | Register Name: WIEGAND_TP |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R/W | 0x0 | <p>TPV TP Value Before setting the TP Value, work out its time unit according to the WIEGAND_CLK_DIV. The equation is as below: $\text{Unit(TPV)} = 1/(\text{WCLK}/\text{WIEGAND_CLK_DIV})$ For example, if the WCLK is 24MHz, and the WIEGAND_CLK_DIV is set to 24. Then, the divided clock frequency will be 1MHz, and the unit of the WIEGAND_TP will be $1/1\text{MHz}=1\text{us}$.</p> |

10.8.6.3 0x000C WIEGAND TW Control Register (Default Value: 0x0000_0000)

| Offset: 0x000C | | | Register Name: WIEGAND_TW |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R/W | 0x0 | <p>TWV TW Value Before setting the TW Value, work out its time unit according to the WIEGAND_CLK_DIV. The equation is as below: $\text{Unit(TWV)} = 1/(\text{WCLK}/\text{WIEGAND_CLK_DIV})$ For example, if the WCLK is 24MHz, and the WIEGAND_CLK_DIV is set to 24, the divided clock frequency will be 1MHz, and the unit of the WIEGAND_TW will be $1/1\text{MHz}=1\text{us}$.</p> |

10.8.6.4 0x0010 WIEGAND Clock Divide Register (Default Value: 0x0000_0000)

| Offset: 0x0010 | | | Register Name: WIEGAND_CLK_DIV |
|----------------|------------|-------------|--------------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:6 | / | / | / |

| Offset: 0x0010 | | | Register Name: WIEGAND_CLK_DIV |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 5:0 | R/W | 0x0 | <p>RCD Rate of WCLK divider The actual clock equals the WCLK frequency divided by the value of the WIEGAND_CLK_DIV. For example, if the WCLK is 24MHz, and the register is set to 24, the frequency of divided clock will be 1MHz.</p> <p>Note:</p> <p>(1) The divider does not support one divided-frequency. That means if the value of RCD is set to 0 or 1, it equals 2.</p> <p>(2) The upper limit of RCD is 48. That means the frequency divider only supports 2-48 divided frequencies.</p> |

10.8.6.5 0x0014 WIEGAND Function Enable Register (Default Value: 0x0000_0000)

| Offset: 0x0014 | | | Register Name: WIEGAND_FUN_EN |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:1 | / | / | / |
| 0 | R/W | 0x0 | <p>FUN_EN Function Enable This is used to enable/disable the function module. 0: Disable, the function module will be reset. 1: Enable</p> |

10.8.6.6 0x001C WIEGAND Interrupt Control Register (Default Value: 0x0000_0000)

| Offset: 0x001C | | | Register Name: WIEGAND_ICR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:2 | / | / | / |
| 1 | R/W | 0x0 | <p>TRIE Timeout Report Interrupt Enable This is used to enable/disable the generation of TRIE Interrupt. 0: Disable 1: Enable</p> |
| 0 | R/W | 0x0 | <p>RIE Receive Interrupt Enable This is used to enable/disable the generation of RIE Interrupt. 0: Disable 1: Enable</p> |

10.8.6.7 0x0020 WIEGAND Interrupt Status Register (Default Value: 0x0000_0000)

| Offset: 0x0020 | | | Register Name: WIEGAND_ISR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:2 | / | / | /. |
| 1 | R/W1C | 0x0 | <p>TRIS Timeout Report Interrupt Status Reads 0: Timeout Report Interrupt is not pending. 1: Timeout Report Interrupt is pending. Writes 0: No Effect. 1: Clear Timeout Report Interrupt status.</p> |

| Offset: 0x0024 | | | Register Name: WIEGAND_CBSR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:2 | / | / | /. |
| 1 | R | 0x0 | <p>CBS_H Check the Bit Status of Identification Code This is used to indicate whether the Check Bit Status of Identification Code is right or wrong. 0: Right 1: Wrong</p> |
| 0 | R | 0x0 | <p>CBS_L Check the Bit Status of Group Code This is used to indicate whether the Check Bit Status of Group Code is right or wrong. 0: Right 1: Wrong</p> |

10.8.6.8 0x0024 WIEGAND Check Bit Status Register (Default Value: 0x0000_0000)

| Offset: 0x0024 | | | Register Name: WIEGAND_CBSR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:2 | / | / | /. |
| 1 | R | 0x0 | <p>CBS_H Check the Bit Status of Identification Code This is used to indicate whether the Check Bit Status of Identification Code is right or wrong. 0: Right 1: Wrong</p> |
| 0 | R | 0x0 | <p>CBS_L Check the Bit Status of Group Code This is used to indicate whether the Check Bit Status of Group Code is right or wrong. 0: Right 1: Wrong</p> |

10.8.6.9 0x0028 WIEGAND Check Bit Polar Register (Default Value: 0x0000_0000)

| Offset: 0x0028 | | | Register Name: WIEGAND_CB_P |
|----------------|------------|-------------|-----------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:2 | / | / | /. |

| Offset: 0x0028 | | | Register Name: WIEGAND_CBP |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 1 | R/W | 0x0 | <p>CBP_H Check the Bit Polar of High Check Bit This is used to set the check polar of high bit 0: even check 1: odd check</p> |
| 0 | R/W | 0x0 | <p>CBP_L Check the Bit Polar of Low Check Bit This is used to set the check polar of low bit 0: even check 1: odd check</p> |

10.8.6.10 0x002C WIEGAND Data Bit Polar Register (Default Value: 0x0000_0000)

| Offset: 0x002C | | | Register Name: WIEGAND_DBP |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:1 | / | / | /. |
| 0 | R/W | 0x0 | <p>DBP Data Bit Polar This is used to set whether the Data 0 and DATA1 is inversed or not. 0: normal 1: inverse</p> |

10.8.6.11 0x0034 WIEGAND Data Status Register (Default Value: 0x0000_0003)

| Offset: 0x0034 | | | Register Name: WIEGAND_DSR |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:2 | / | / | /. |
| 1 | R | 0x1 | <p>DS1 DATA1 Status This is used to read the value of DATA1 directly.</p> |
| 0 | R | 0x1 | <p>DS0 DATA0 Status This is used to read the value of DATA0 directly.</p> |

10.8.6.12 0x0038 WIEGAND Timeout DATA Number Register (Default Value: 0x0000_0000)

| Offset: 0x0038 | | | Register Name: WIEGAND_TDNR |
|----------------|------------|-------------|-----------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | /. |

| Offset: 0x0038 | | | Register Name: WIEGAND_TDNR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 7:0 | R | 0x0 | TDN Timeout Data Number In case of timeout error, TDN indicates the bit number that already received by the serial-to-parallel buffer. |

10.8.6.13 0x003C WIEGAND Timeout DATA Register (Default Value: 0x0000_0000)

| Offset: 0x003C | | | Register Name: WIEGAND_TDR |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R | 0x0 | TD Timeout Data In case of timeout error, TD indicates the bit data that already received by the serial-to-parallel buffer. Note that only bit[31:0] of the serial-to-parallel buffer can be read by the TD. |

10.8.6.14 0x0100 WIEGAND RX DATA Register (Default Value: 0x0000_0000)

| Offset: 0x0100 | | | Register Name: WIEGAND_RXD |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R | 0x0 | RD Rx Data The data received from Wiegand interface |

10.8.6.15 0x0200 WIEGAND RX Check Register (Default Value: 0x0000_0000)

| Offset: 0x0200 | | | Register Name: WIEGAND_RXC |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:2 | / | / | / |
| 1 | R | 0x0 | RCB_H Rx Check bit of Identification Code The high Check bit received from Wiegand interface |
| 0 | R | 0x0 | RCB_L Rx Check bit of Group Code The low Check bit received from Wiegand interface |

Contents

| | |
|-------------------------------------|------|
| 11 Security System..... | 1118 |
| 11.1 Crypto Engine..... | 1118 |
| 11.1.1 Overview | 1118 |
| 11.1.2 Block Diagram | 1120 |
| 11.1.3 Functional Description | 1121 |
| 11.1.4 Operation Guides | 1131 |
| 11.1.5 Register List..... | 1143 |
| 11.1.6 Register Description..... | 1144 |
| 11.2 Security ID | 1158 |



Figures

| | |
|---|------|
| Figure 11-1 CE Block Diagram | 1120 |
| Figure 11-2 DES Encryption and Decryption | 1121 |
| Figure 11-3 3DES Encryption and Decryption of a 3-key Operation and a 2-key Operation | 1122 |
| Figure 11-4 ECB Mode Encryption and Decryption..... | 1123 |
| Figure 11-5 CBC Mode Encryption and Decryption | 1123 |
| Figure 11-6 CTR Mode Encryption and Decryption..... | 1124 |
| Figure 11-7 CFB Mode Encryption and Decryption..... | 1125 |
| Figure 11-8 OFB Mode Encryption and Decryption | 1126 |
| Figure 11-9 CTS Mode Encryption and Decryption | 1127 |
| Figure 11-10 Word Address of Message | 1128 |
| Figure 11-11 Byte Order..... | 1128 |
| Figure 11-12 Bit Order | 1128 |
| Figure 11-13 The Storage Method of 32-bit IV | 1129 |
| Figure 11-14 The Storage Method of 64-bit IV | 1129 |
| Figure 11-15 Task Chaining of Hash Algorithms and Random Bit Generator Algorithms..... | 1132 |
| Figure 11-16 CE Block Diagram | 1136 |

Tables

| | |
|---|------|
| Table 11-1 Symmetric Algorithm Configuration Properties..... | 1129 |
| Table 11-2 Hash Algorithm Configuration Properties..... | 1130 |
| Table 11-3 RNG Configuration Properties..... | 1130 |
| Table 11-4 Asymmetric Algorithm Configuration Properties..... | 1130 |



11 Security System

11.1 Crypto Engine

11.1.1 Overview

The Crypto Engine (CE) module is one encryption/decryption algorithms accelerator. It supports kinds of symmetric, asymmetric, HASH, and RBG algorithms. There are two software interfaces for secure and non-secure world each. Algorithm control information is written in memory by task descriptor, then CE automatically reads it when executing request. It supports parallel requests from 4 channels each world, and has an internal DMA controller to transfer data between CE and memory. It supports parallel running for symmetric, HASH, asymmetric algorithms.

The CE has the following features:

- Supports symmetrical algorithm for encryption and decryption: AES, XTS-AES, DES, 3DES, SM4
 - Supports ECB, CBC, CTS, CTR, CFB, OFB, CBC-MAC, GCM mode for AES
 - Supports 128/192/256-bit key for AES
 - Supports 256/512-bit key for XTS-AES
 - Supports ECB, CBC, CTR, CBC-MAC mode for DES
- Hash Algorithms
 - Support MD5, SHA1, SHA224, SHA256, SHA384, SHA512, SM3
 - Support HMAC-SHA1, HMAC-SHA256
 - Support multi-package¹ mode for these ones
 - Support hardware padding
- Random bit generate Algorithms
 - Support PRNG, 15 bits seed width, and output with multiple of 5 words
 - Support TRNG, post-process by hardware with SHA256, output with multiple of 8 words
 - Support Instantiate/Reseed/Generate/Uninstantiate 4 process
 - Support prediction resistance requests
 - Support 8 separate suits of Internal State

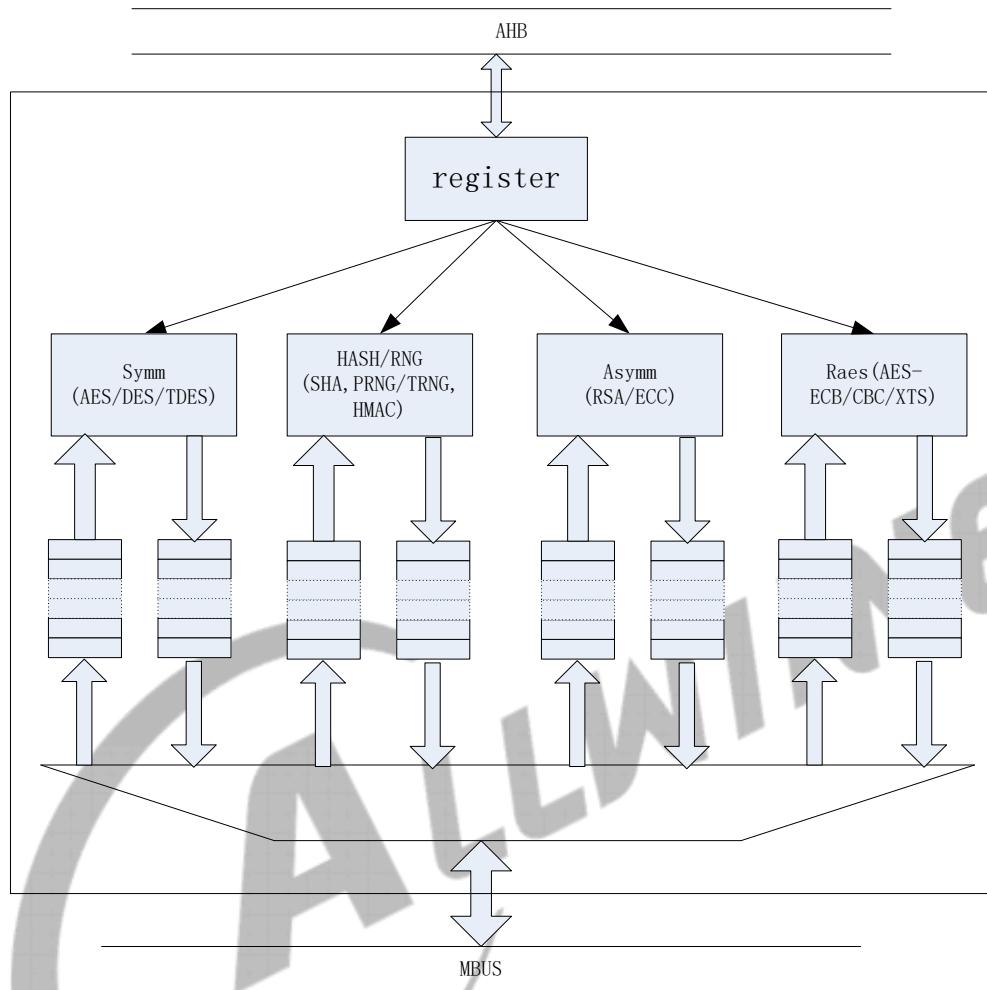
¹ If not last package, input should be aligned with computation block, namely 512bits or 1024bit

- Maxim 2^32 BYTE length of Entropy input, Nonce, Personalization, Additional input. And length is multiple of word
- Public Key Algorithms
 - Support RSA Public Key Algorithms, include 512/1024/2048/3072/4096 bit width
 - Support ECC Public Key Algorithms, include 160/224/256/384/521 bit width
 - Support SM2 Algorithms
- Security Strategy and System Feature
 - Symmetric, asymmetric, HASH/RBG ctrl logics are separate, can handle task simultaneously. Symmetric logic can select instantiate 2 suits at implementation time.
 - Support task chain mode for each request. Task or task chain are executed at request order.
 - 8 scatter group(sg) are supported for both input and output data
 - Support secure and non-secure interfaces respectively, each world issues task request through its own interface, don't know each other's existence.
 - Each world has 4 channels for software request, each channel has an interrupt control and status bit, and channels are independent with each other.
 - Supports byte-aligned address for all configurations

11.1.2 Block Diagram

The following figure shows a block diagram of CE.

Figure 11-1 CE Block Diagram

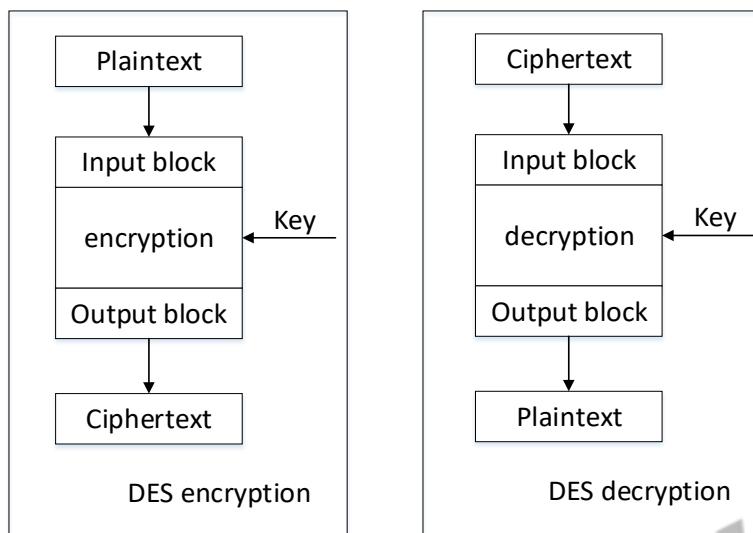


11.1.3 Functional Description

11.1.3.1 DES Algorithm

The following figure shows the DES encryption and decryption operation.

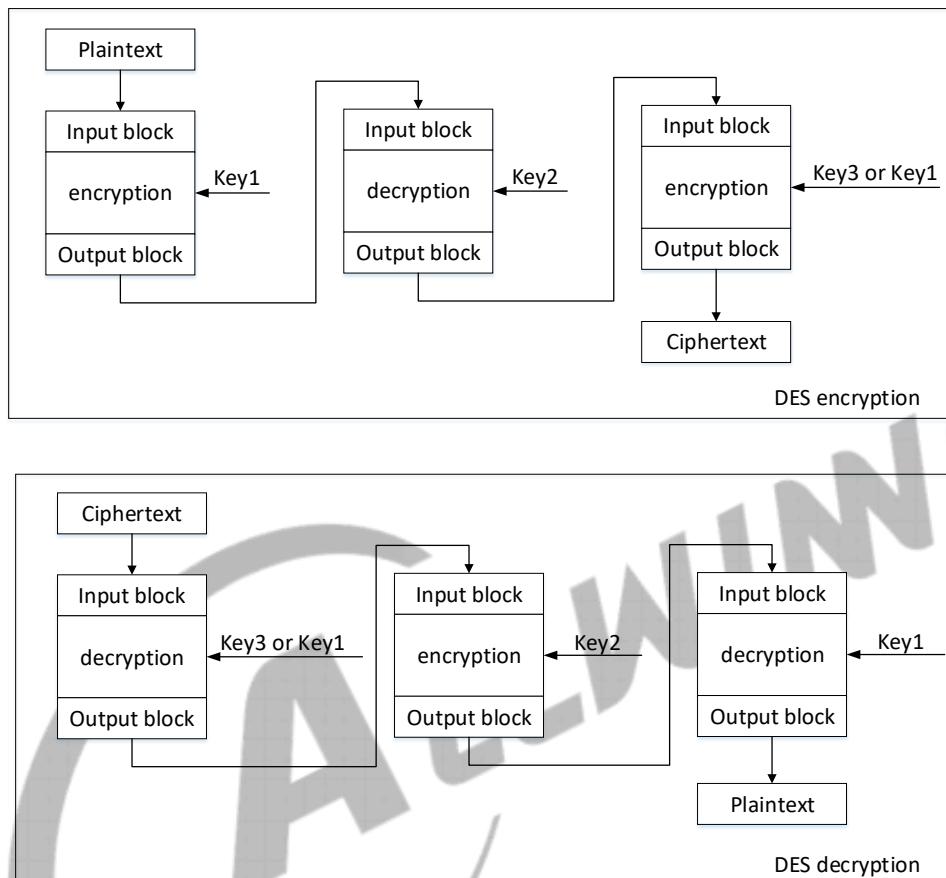
Figure 11-2 DES Encryption and Decryption



11.1.3.2 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. The following figure shows the 3DES encryption and decryption operation of a 3-key operation and a 2-key operation.

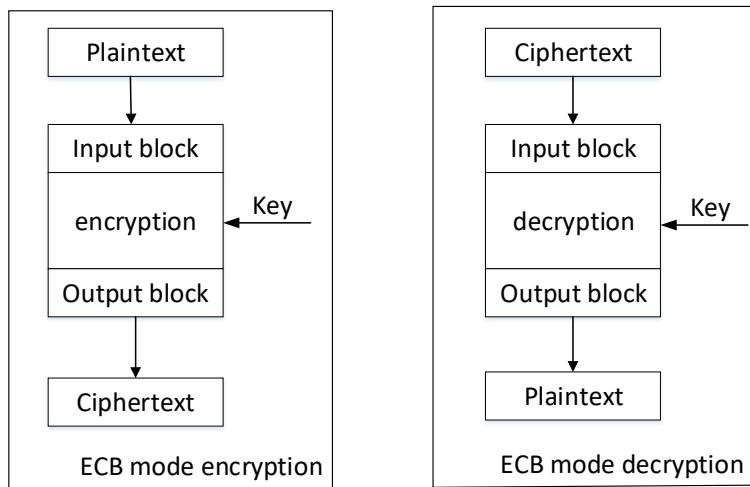
Figure 11-3 3DES Encryption and Decryption of a 3-key Operation and a 2-key Operation



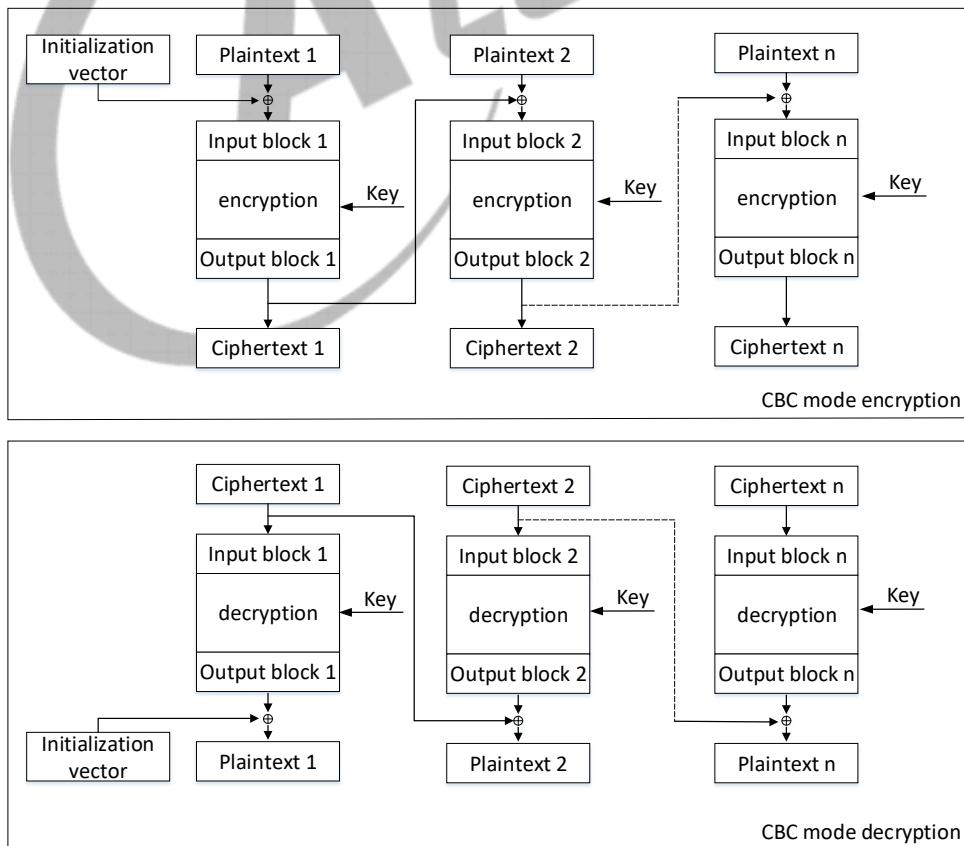
11.1.3.3 ECB Mode

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 11-4 ECB Mode Encryption and Decryption**11.1.3.4 CBC Mode**

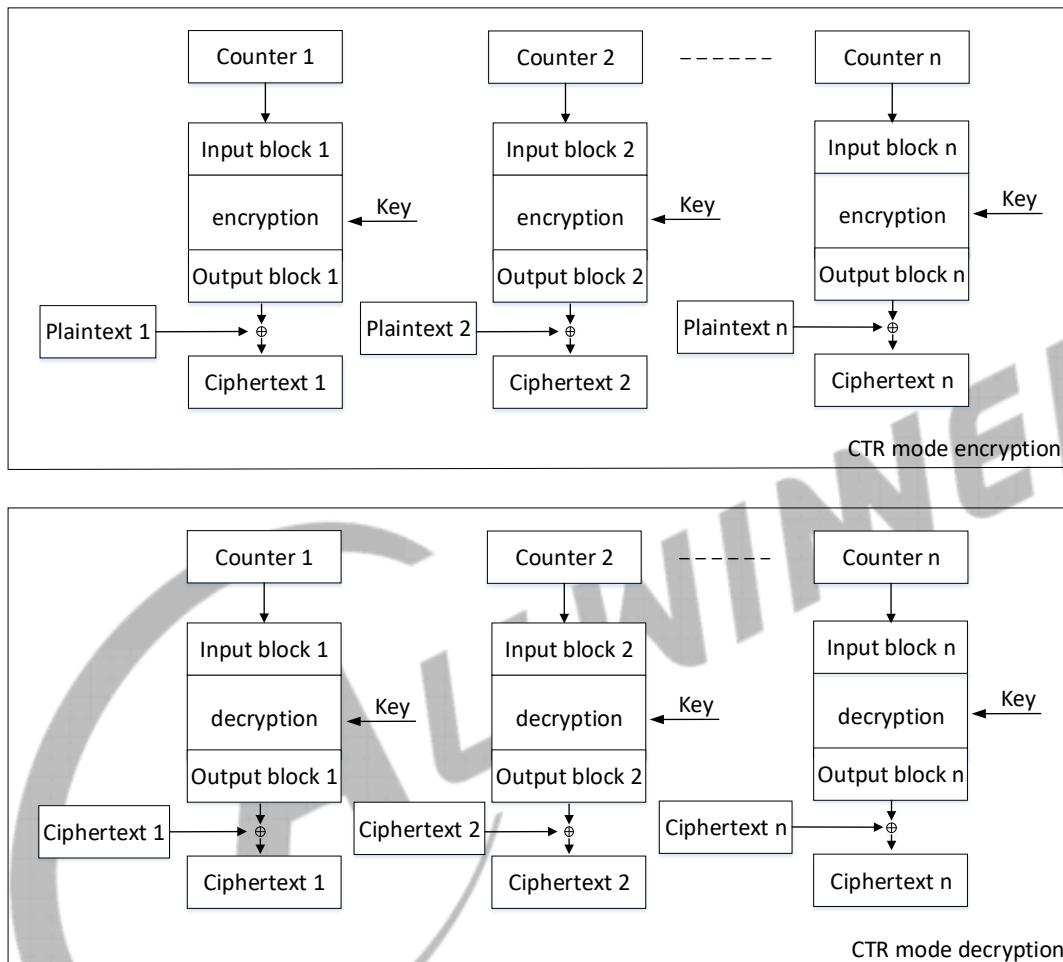
The CBC mode is a confidentiality mode whose encryption process features the combining of the plaintext blocks with the previous ciphertext blocks. The CBC mode requires an initialization vector (IV) to combine with the first plaintext block. The encryption process of each plaintext block is related 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 11-5 CBC Mode Encryption and Decryption

11.1.3.5 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 11-6 CTR Mode Encryption and Decryption

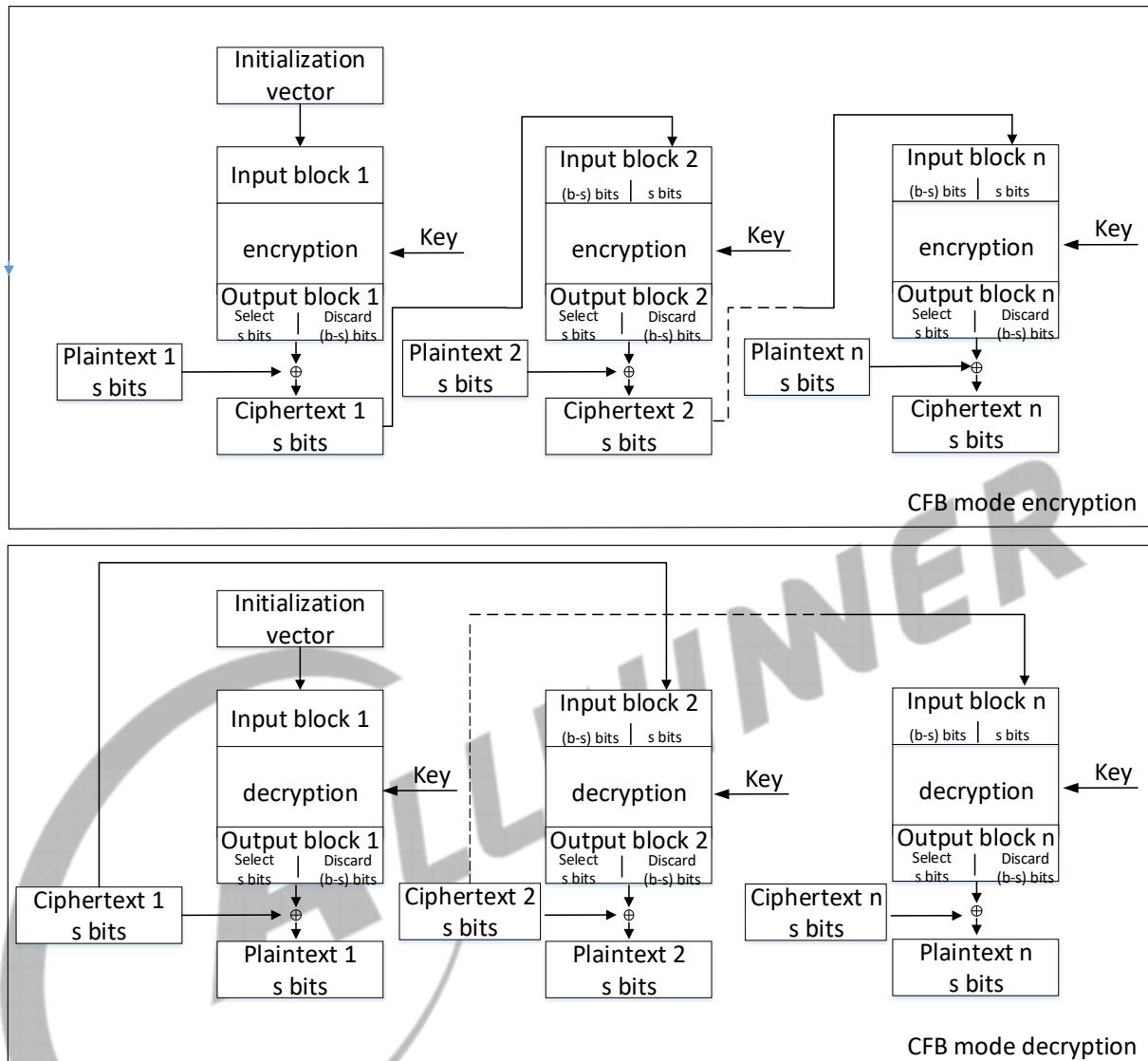


11.1.3.6 CFB Mode

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.

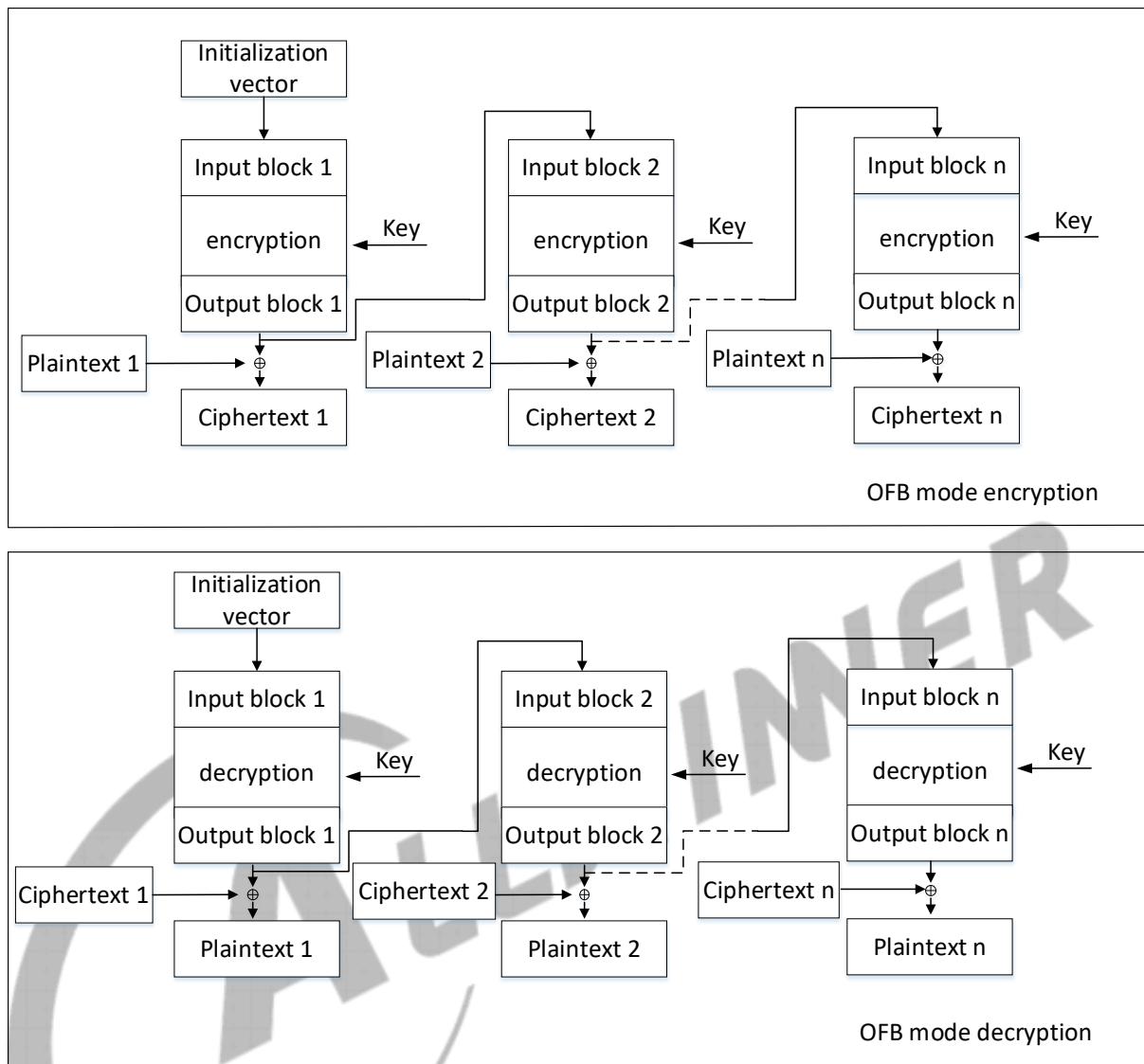
The following figure shows the s-bit CFB mode of the AES algorithms.

Figure 11-7 CFB Mode Encryption and Decryption



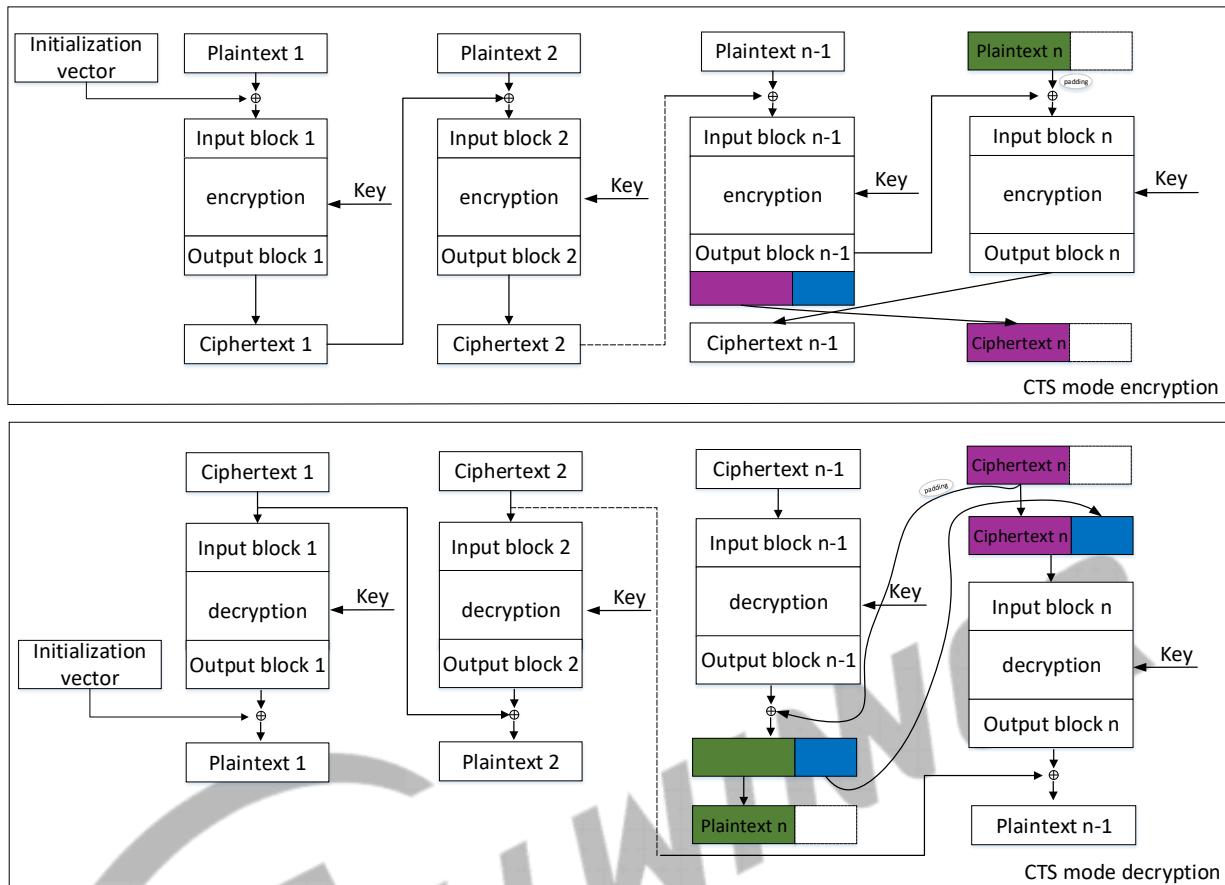
11.1.3.7 OFB Mode

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 11-8 OFB Mode Encryption and Decryption

11.1.3.8 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 11-9 CTS Mode Encryption and Decryption

11.1.3.9 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.

11.1.3.10 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: $M = C^D \text{ mod } N$.

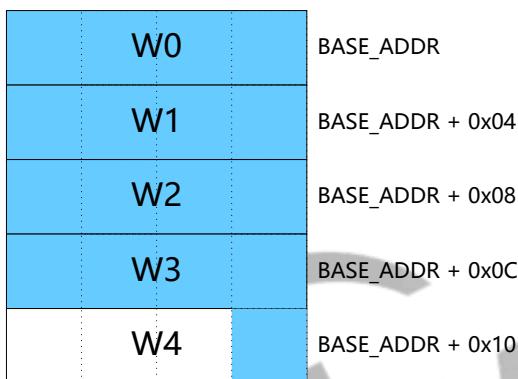
M indicates the plaintext, C indicates the ciphertext, (N, E) indicates the public key, and (N, D) indicates the private key.

11.1.3.11 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.

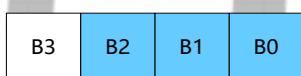
Each task supports up to 8 message segments, and the data volume of each message segment supports up to 4 GWord (AES-CTS is 1 GByte). The total amount of all segments in a task (that is a package) supports up to 4 GWord (AES-CTS is 1 GByte). If a message is divided into multiple packages, all others 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 11-10 Word Address of Message



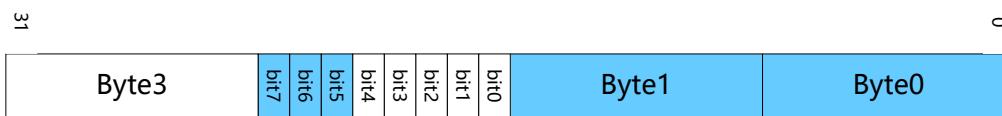
Byte order: low byte first, high byte last. When the data is less than one word, the low byte is filled first. The following figure shows the byte order structure (blue means it is filled by the message).

Figure 11-11 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 11-12 Bit Order



11.1.3.12 Storing Key

The length of KEY must be an integer multiple of word, refer to section 11.1.3.14 Algorithm Length Properties.

11.1.3.13 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 11-13 The Storage Method of 32-bit IV

| | |
|-----------|------------------|
| IV0[31:0] | BASE_ADDR |
| IV1[31:0] | BASE_ADDR + 0x04 |
| | |
| IV7[31:0] | BASE_ADDR + 0x1C |

The following figure shows the storage method of 64-bit IV value.

Figure 11-14 The Storage Method of 64-bit IV

| | |
|-----------|------------------|
| IV0[31:0] | BASE_ADDR |
| IV1[31:0] | BASE_ADDR + 0x04 |
| | |
| IV7[31:0] | BASE_ADDR + 0x1C |

11.1.3.14 Algorithm Length Properties

The algorithm length has different requirements for different algorithms.

Table 11-1 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 |

| Algorithm | Length Setting | | | | Alignment | Software Padding |
|-----------|----------------|------------------|--------|--------|--------------|------------------|
| | Source Size | Destination Size | KEY | IV | | |
| 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 11-2 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 11-3 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 11-4 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 |

| Algorithm | Length Setting | | | | Alignment | Software Padding |
|-----------|----------------|------------------|---------|------------|--------------|------------------|
| | Source Size | Destination Size | KEY | IV | | |
| RSA2048 | 64 Word | 64 Word | 64 Word | Not use IV | Word-aligned | need |

11.1.3.15 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.

11.1.3.16 Clock Requirement

| Clock Name | Description | Requirement |
|------------|-------------------|------------------|
| AHB_CLK | AHB bus clock | 24 MHz ~ 200 MHz |
| M_CLK | MBUS clock | 24 MHz ~ 400 MHz |
| CE_CLK | CE work clock | 24 MHz ~ 400 MHz |
| OSL_CLK | TRNG sample clock | <3 MHz |

11.1.4 Operation Guides

11.1.4.1 Task Descriptor of Hash Algorithms and RBG Algorithms

The task descriptor is data written by software to a contiguous space in memory. The data describes the various properties of a task, such as algorithm type, mode, subcommand, key address, data source address, the data size read from data source, abstract destination address, the written destination data size, and the information of other tasks. First, we configure the task descriptor by software; then we operate the registers of CE to start this task. After the task starts, CE will read task descriptor based on the address of the task descriptor configured in register, and perform the task one time based on the described properties.

In applications, the “NEXT TASK ADDR” field can be configured as the starting address of the next task descriptor, to concatenate multi task descriptors into a task chain. After starting the first task, CE will perform every task in order until the “NEXT TASK ADDR” field is invalid (that is 0).

The HASH/RBG algorithms and Symmetrical/Asymmetrical algorithms use the different descriptor structure, separately.

Figure 11-15 Task Chaining of Hash Algorithms and Random Bit Generator Algorithms

The detail structures are as follows.

| No. | Descriptor | Name | Width | Description |
|-----|------------|----------|---------|---|
| 0 | CTRL | CHN | [1:0] | Channel ID |
| | | IVE | [8] | IV mode enable, active high |
| | | LPKG | [12] | Last package flag, active high |
| | | DLAV | [13] | Data length valid For last package, the bit needs be configured. For non last package, the bit needs not be configured. (Please configure it as 0 in PRNG/TRNG/DRBG.) 1: DLA means the WORD address where data total length (by bits) is saved. 0: DLA means the value of message total length (by bits). |
| | | IE | [16] | Interrupt enable for current task, active high |
| 1 | CMD | HASH SEL | [3:0] | Hash algorithms select 0: MD5 1: SHA1 2: SHA224 3: SHA256 4: SHA384 5: SHA512 6: SM3 Other: Reserved |
| | | HME | [4] | HMAC mode enable, active high |
| | | RGB SEL | [11:8] | RGB algorithms select 0: No RGB use 1: PRNG 2: TRNG Other: Reserved |
| | | SUB CMD | [31:16] | Sub-command in a specific algorithms When using PRNG, sub_cmd[15] means PRNG seed reload; sub_cmd[14:0] means PRNG linearly shifted seed |
| 2 | DLA | DLA | [31:0] | Data length OR its address. For last package, the field needs be configured. For non last package, the field needs not be configured. (Not used in PRNG/TRNG) When DLAV=1, here is the WORD address where data total length (by bits) is saved. When DLAV=0, here is the value of message total length (by bits) |

| No. | Descriptor | Name | Width | Description |
|--------|------------|-----------|---------|---|
| 3 | DLA | DLA | [7:0] | When DLAV=1, here is the byte address bit[39:32] where data total length (by bits) is saved. |
| | KA | KA | [31:8] | KEY Address: The byte address bit[23:0].where HMAC KEY or PRNG KEY is saved. |
| 4 | KA | KA | [15:0] | KEY Address: The byte address bit[39:24].where HMAC KEY or PRNG KEY is saved. |
| | IVA | IVA | [31:16] | IV Address: The byte address bit[15:0] where IV is saved. |
| 5 | IVA | IVA | [23:0] | IV Address: The byte address bit[39:16] where IV is saved. |
| | Reversed | Reversed | [31:24] | / |
| 6+5*x | SGx_W0 | SGx_WORD0 | [31:0] | Source Data Address x: The byte address bit[31:0] where Source Data is saved. |
| 7+5*x | SGx_W1 | SGx_WORD1 | [7:0] | Source Data Address x: The byte address bit[39:32] where Source Data is saved. |
| | | | [31:8] | Output Data Address x: The byte address bit[23:0] where Output Data to be saved. |
| 8+5*x | SGx_W2 | SGx_WORD2 | [15:0] | Output Data Address x: The byte address bit[39:24] where Output Data to be saved. |
| | Reversed | Reversed | [31:16] | / |
| 9+5*x | SGx_W3 | SGx_WORD3 | [31:0] | Source Data length x: The Length (by bytes) of Source Data. |
| 10+5*x | SGx_W4 | SGx_WORD4 | [31:0] | Output Data length x: The Length (by bytes) of output Data. |
| 46 | NSA | NSA | [31:0] | Next SG Address: The byte address bit[31:0].where the descriptor of the next 8 sg in a task is saved. If this is the only one group sg or the last group of a task, NSA must be 32'h0. |
| 47 | NSA | NSA | [7:0] | Next SG Address: The byte address bit[39:32].where the descriptor of the next 8 sg in a task is saved. If this is the only one group sg or the last group of a task, NSA must be 8'h0. |

| No. | Descriptor | Name | Width | Description |
|-----|------------|----------|--------|---|
| | NTA | NTA | [31:8] | Next task Address: The byte address bit[23:0] where the descriptor of the next task in a task-chain is saved. If this is the only task or the last task of a task-chain, NTA must be 24'h0. |
| 48 | NTA | NTA | [7:0] | Next task Address: The byte address bit[39:24] where the descriptor of the next task in a task-chain is saved. If this is the only task or the last task of a task-chain, NTA must be 16'h0. |
| | Reversed | Reversed | [31:8] | / |
| 49 | Reversed | Reversed | / | / |
| 50 | Reversed | Reversed | / | / |
| 51 | Reversed | Reversed | / | / |

11.1.4.2 Other Algorithms Task Descriptor

Software make request through task descriptor, providing algorithm type, mode, key address, source/destination sg address and size, etc. The task descriptor is as follows.

Figure 11-16 CE Block Diagram



Channel id supports 0-3 for each world.

- Common ctrl:

| Bit | Read/Write | Default | Description |
|-------|------------|---------|---|
| 31 | R/W | 0 | interrupt enable for current task 0: disable interrupt 1: enable interrupt |
| 30:25 | / | / | / |
| 24:17 | R/W | 0 | cbc_mac_len the outcome bit length of CBC-MAC when in CBC-MAC mode. The part also be used as gcm/ocb mode tag_len. |
| 16:9 | / | / | / |
| 8 | R/W | 0 | OP DIR Algorithm Operation Direction 0: Encryption 1: Decryption |
| 7 | / | / | / |
| 6:0 | R/W | 0 | Algorithm type 0x0: AES 0x1: DES 0x2: Triple DES (3DES) 0x3: SM4 others: reserved 0x20: RSA 0x21: ECC 0x22: SM2 others: reserved 0x30: RAES Others: reserved |

- Symmetric ctrl

| Bit | Read/Write | Default | Description |
|-------|------------|---------|--|
| 31:30 | / | / | / |
| 29:28 | R/W | 0 | SCK_SEL 0: use sck0/maskkey0 1: use sck1/maskkey1 2: use sck2/maskkey2 3: reserved |

| Bit | Read/Write | Default | Description |
|------------|-------------------|----------------|--|
| 27 | R/W | 0 | <p>ESCK_EN</p> <p>0: the process ESCK to SCK not need 1: the process ESCK to SCK need</p> <p>*Anyway, when firstly starting the keyladder as after reset,it will be the process ESCK to SCK whether the bit is 1 or not.When once again ,the process ESCK to SCK will see the bit.</p> |
| 26 | R/W | 0 | <p>no_modk</p> <p>0: have module key derivation function 1: no module key derivation function</p> |
| 25:24 | R/W | 0 | <p>Key ladder stage</p> <p>0: no key ladder 1: 3-stage key ladder 2: 5-stage key ladder 3: reserved</p> |
| 23:20 | R/W | 0 | <p>KEY Select</p> <p>key select for AES/SM4/TDES (TDES only configured as 0/8-15)</p> <p>0: Select input CE_KEYx (Normal Mode) 1: Select {SSK} 2: Select {HUK} 3: Select {RSSK}, used for decrypt HDCP key, EK, BSSK 4-7: Reserved 8-15: Select internal Key n (n from 0 to 7)</p> |
| 19:18 | R/W | 0 | <p>cfb_width</p> <p>For AES-CFB width</p> <p>0: CFB1 1: CFB8 2: CFB64 3: CFB128</p> |
| 17 | / | / | / |
| 16 | R/W | 0 | <p>AES CTS last package flag</p> <p>When set to '1', it means this is the last package for AES-CTS mode(the size of the last package >128bit).</p> <p>The part also be used as gcm/ocb mode gcm_last/ocb_last .</p> |
| 15:14 | / | / | / |
| 13 | R/W | 0 | <p>xts_last</p> <p>0: not last block for XTS 1: last block for XTS</p> |
| 12 | R/W | 0 | <p>xts_first</p> <p>0: not first block for XTS 1: first block for XTS</p> |

| Bit | Read/Write | Default | Description |
|------|------------|---------|--|
| 11:8 | R/W | 0 | AES/DES/3DES/RAES modes. DES/3DES only supports ECB/CBC/CTR. RAES only supports ECB/CBC/XTS. operation mode for symmetric 0: Electronic Code Book (ECB) mode 1: Cipher Block Chaining (CBC) mode 2: Counter (CTR) mode 3: CipherText Stealing (CTS) mode 4: Output feedback (OFB) mode 5: Cipher feedback (CFB) mode 6: CBC-MAC mode 7: OCB mode 8: GCM mode 9: XTS mode Other: reserved |
| 7:6 | / | / | / |
| 5:4 | R/W | 0 | gcm_iv_mode[1:0] gcm_iv_mode[0]: value 1 show the last req for iv calculate gcm_iv_mode[1]: 0: no GHASH calculate mode 1: GHASH calculate mode gcm_iv_mode[1:0]: 00: IDLE state, this calculate do not have the process from iv to J0. 01: by iv padding generating J0. On the mode, iv padding is 96 bits, so iv_length will be 96bits. 10: by GHASH calculate for iv generating J0, and this is not the last req for iv calculate. 11: by GHASH calculate for iv generating J0, and this is the last req for iv calculate |
| 3:2 | R/W | 0 | CTR Width Counter Width for CTR Mode 0: 16-bits Counter 1: 32-bits Counter, gcm mode always use this setting without software 2: 64-bits Counter 3: 128-bits Counter |
| 1:0 | R/W | 0 | AES Key Size 0: 128-bits 1: 192-bits 2: 256-bits 3: Reserved |

- Asymmetric ctrl

| Bit | Read/Write | Default | Description |
|-------|------------|---------|--|
| 31:21 | / | / | / |
| 20:16 | R/W | 0 | PKC algorithm mode. For modular computation: 00000: modular exponent(RSA) 00001: modular add 00010: modular minus 00011: modular multiplication others: reserved For ECC: 00000: point add 00001: point double 00010: point multiplication 00011: point verification 00100: encryption 00101: decryption 00110: sign 00111: sign verify others: reserved For SM2: 00000: encryption 00001: decryption 00010: sign 00011: sign verify 00100: key exchange |
| 15:8 | / | / | / |
| 7:0 | R/W | 0 | Asymmetric algorithms operation width field. It indicates how much width this request apply, as words. |

key addr field is address for each algorithm's key, also for extension feature micro codes address.(By byte)

iv addr field is tweak value address for XTS.

ctr addr is address for next block's IV. (By byte)

src/dst sgX addr field indicate 40bits address for source and destination data. (By byte)

src/dst sgX size field indicates size for each sg respectively(by byte)

For SG, the detail as flow:

| by te3 | by te2 | by te1 | by te0 | |
|-------------------|-------------------|-------------------|-------------------|----------|
| SRC_ADDR0 [B3] | SRC_ADDR0 [B2] | SRC_ADDR0 [B1] | SRC_ADDR0 [B0] | SG_WORD0 |
| DST_ADDR0 [B2] | DST_ADDR0 [B1] | DST_ADDR0 [B0] | SRC_ADDR0 [B4] | SG_WORD1 |
| | | DST_ADDR0 [B4] | DST_ADDR0 [B3] | SG_WORD2 |
| SRC_SIZE0 [B3] | SRC_SIZE0 [B2] | SRC_SIZE0 [B1] | SRC_SIZE0 [B0] | SG_WORD3 |
| DST_SIZE0 [B3] | DST_SIZE0 [B2] | DST_SIZE0 [B1] | DST_SIZE0 [B0] | SG_WORD4 |

1 group SG has 8 sg, each sg has 5 words, the ADDR is 40 bits and byte-addr; the SIZE is 32bits nad byte-unit. We will support unlimited SG number, but the 1860 just use for test. This can has many group SG in a task, using the next_sg_addr to create the new SG information in the task.

Next sg field should be set to 0 when no next group sg, else set to next sg's descriptor.

next task field should be set to 0 when no next task, else set to next task's descriptor.

11.1.4.3 Security Operations

CE has two sets of complete registers, the base address of one register is CE_NS, is called non-secure world; the base address of the other register is CS_S, is called secure world. The configurations of two registers are independent each other. The task started by configuring the non-secure world register is called non-secure task; otherwise, called secure task. The difference between secure world and non-secure world has mainly two aspects.

- Non-secure CPU only can access the registers of non-secure world; secure CPU can access the registers of non-secure world and the registers of secure world;
- Non-secure task cannot access any secure domain in system, such as KEYSRAM and SID. But secure task can.

For example, if the EK stored in KEYSRAM needs to do a MD5 operation, the task requires CE to read the EK of KEYSRAM during execution, so it can only access secure-world registers by secure CPU to configure and start the task.

11.1.4.4 Task Parallel

Algorithms are divided into 4 types: symmetric, HASH/RBG, asymmetric, and RAES. These four algorithms can run in parallel. Among these 4 types, the task request and complete time are not sure. If one type uses the outcome of another type, software should make sure to start one type after another type is finished.

Each type has a task queue with 8 elements for requests. Tasks in each queue are handled in sequence. If software makes several requests with the same type, these tasks will be executed in request sequence. If software makes several requests with different types, these tasks will be executed in parallel.

CE supports 4 virtual channels with ID 0-3 in each world. Because parallel tasks would finish out of order, software should make different type request with different channel id, which results in generating different interrupt status bit.

11.1.4.5 PKC Microcode

PKC module supports RSA, ECC, SM2 algorithms in the form of microcode. It implements basic modular add, minus, multiplication, point add, point double, and logic computing, etc. Complete RSA/ECC/SM2 encryption, decryption, sign, verify are implemented with these microcode.

Asymmetric algorithms RSA/ECC/SM2 are implemented as microcode in PKC module. The encryption, decryption, sign, verify operations of asymmetric algorithms are composed with certain fixed microcode with hardware.

11.1.4.6 PKC Configuration

Before starting PKC, task description must be configured. Parameters to PKC are assigned to source sg, outcome is put to destination sg.

For RSA, parameters should be at the order of key, modulus, plaintext.

For ECC point add $P_2 = P_0 + P_1$, parameters should be at the order of p, P_{0x} , P_{0y} , P_{1x} , P_{1y} . Output is at the order of P_{2x} , P_{2y} .

For ECC point double $P_2 = 2 * P_0$, parameters should be at the order of p, a, P_{0x} , P_{0y} . Output is at the order of P_{2x} , P_{2y} .

For ECC point multiplication $P_2 = k * P_0$, parameters should be at the order of p, k, a, P_{0x} , P_{0y} . Output is at the order of P_{2x} , P_{2y} .

For ECC point verification, parameters should be at the order of p, a, P_{0x} , P_{0y} , b. Output is 1 or 0.

For ECC encryption, parameters should be at the order of random k, p, a, G_x , G_y , Q_x , Q_y , m. Output is at the order of R_x , R_y , c.

For ECC decryption, parameters should be at the order of random k, p, a, R_x , R_y , c. Output is m.

For ECC signature, parameters should be at the order of random k, p, a, G_x , G_y , n, d, e. Output is at the order of r, s.

For ECC signature verification, parameters should be at the order of n, s, e, r, p, a, G_x , G_y , Q_x , Q_y , n, r. Output is 1 or 0.

11.1.4.7 Error Check

After CE reads the task descriptor, CE can monitor error during algorithm operation. When the error is monitored, CE will do the following operations:

The task will pause immediately

Generates interrupt

The corresponding channel of the task status register is Fail

The corresponding channel bit of error status register can be read error number

The error number has the following types.

| Code | Name | Description | Algorithms Type |
|------|--------------------------------|---|-----------------|
| 0x01 | algorithm not support | The algorithm type is not supported. | All |
| 0x11 | KEYSRAM access error | In AES decryption task, RSSK is used as plaintext, the DST address is not in KEYSRAM space. | AES decryption |
| 0x21 | key ladder configuration error | / | KL |
| 0x31 | data length error | Input size or output size configuration size error. | All |

11.1.5 Register List

| Module Name | Base Address | Comments |
|-------------|--------------|-----------------|
| CE_NS | 0x03040000 | Non-Security CE |
| CE_S | 0x03040800 | Security CE |

| Register Name | Offset | Description |
|---------------|--------|---|
| CE_NS | | |
| CE_TDAO_NS | 0x0000 | Non-Security CE Task Descriptor Address0 Register |
| CE_TDA1_NS | 0x0004 | Non-Security CE Task Descriptor Address1 Register |
| CE_ICR_NS | 0x0008 | Non-Security CE Interrupt Control Register |
| CE_ISR_NS | 0x000C | Non-Security CE Interrupt Status Register |
| CE_TLR_NS | 0x0010 | Non-Security CE Task Load Register |
| CE_TSR_NS | 0x0014 | Non-Security CE Task Status Register |
| CE_ESR_NS | 0x0018 | Non-Security CE Error Status Register |
| CE_DRL_NS | 0x001C | Non-Security CE DRBG Request limit Register |
| CE_S | | |
| CE_TDAO_S | 0x0000 | Security CE Task Descriptor Address0 Register |
| CE_TDA1_S | 0x0004 | Security CE Task Descriptor Address1 Register |
| CE_ICR_S | 0x0008 | Security CE Interrupt Control Register |
| CE_ISR_S | 0x000C | Security CE Interrupt Status Register |
| CE_TLR_S | 0x0010 | Security CE Task Load Register |
| CE_TSR_S | 0x0014 | Security CE Task Status Register |
| CE_ESR_S | 0x0018 | Security CE Error Status Register |
| CE_DRL_S | 0x001C | Security CE DRBG Request limit Register |
| CE_SCSAO_S | 0x0020 | Security CE Symmetric algorithm DMA Current Source Address0 Register |
| CE_SCSA1_S | 0x0024 | Security CE Symmetric algorithm DMA Current Source Address1 Register |
| CE_SCDAO_S | 0x0028 | Security CE Symmetric algorithm DMA Current Destination Address0 Register |

| Register Name | Offset | Description |
|---------------|--------|--|
| CE_SCDA1_S | 0x002C | Security CE Symmetric algorithm DMA Current Destination Address1 Register |
| CE_HCSA0_S | 0x0030 | Security CE HASH algorithm DMA Current Source Address0 Register |
| CE_HCSA1_S | 0x0034 | Security CE HASH algorithm DMA Current Source Address1 Register |
| CE_HCDA0_S | 0x0038 | Security CE HASH algorithm DMA Current Destination Address0 Register |
| CE_HCDA1_S | 0x003C | Security CE HASH algorithm DMA Current Destination Address1 Register |
| CE_ACSA0_S | 0x0040 | Security CE Asymmetric algorithm DMA Current Source Address0 Register |
| CE_ACSA1_S | 0x0044 | Security CE Asymmetric algorithm DMA Current Source Address1 Register |
| CE_ACDA0_S | 0x0048 | Security CE Asymmetric algorithm DMA Current Destination Address0 Register |
| CE_ACDA1_S | 0x004C | Security CE Asymmetric algorithm DMA Current Destination Address1 Register |
| CE_XCSA0_S | 0x0050 | Security CE XTS algorithm DMA Current Source Address0 Register |
| CE_XCSA1_S | 0x0054 | Security CE XTS algorithm DMA Current Source Address1 Register |
| CE_XCDA0_S | 0x0058 | Security CE XTS algorithm DMA Current Destination Address0 Register |
| CE_XCDA1_S | 0x005C | Security CE XTS algorithm DMA Current Destination Address1 Register |

11.1.6 Register Description

11.1.6.1 0x0000 Non-Security CE Task Descriptor Address0 Register (Default Value: 0x0000_0000)

| Offset: 0x0000 | | | Register Name: CE_TDA0_NS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R/W | 0x0 | TASK_DES_Address0 Task Descriptor Address0 is bit[31:0] (byte address) |

11.1.6.2 0x0004 Non-Security CE Task Descriptor Address1 Register (Default Value: 0x0000_0000)

| Offset: 0x0004 | | | Register Name: CE_TDA1_NS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:0 | R/W | 0x0 | TASK_DES_Address1 Task Descriptor Address is bit[39:32] (byte address) |

11.1.6.3 0x0008 Non-Security CE Interrupt Control Register (Default Value: 0x0000_0000)

| Offset: 0x0008 | | | Register Name: CE_ICR_NS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:4 | / | / | / |
| 3 | R/W | 0x0 | TASK_CHAN3_INT_EN Channel 3 Task Interrupt Enable 0: interrupt disable 1: interrupt enable |

| Offset: 0x0008 | | | Register Name: CE_ICR_NS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 2 | R/W | 0x0 | TASK_CHAN2_INT_EN Channel 2 Task Interrupt Enable 0: interrupt disable 1: interrupt enable |
| 1 | R/W | 0x0 | TASK_CHAN1_INT_EN Channel 1 Task Interrupt Enable 0: interrupt disable 1: interrupt enable |
| 0 | R/W | 0x0 | TASK_CHAN0_INT_EN Channel 0 Task Interrupt Enable 0: interrupt disable 1: interrupt enable |

11.1.6.4 0x000C Non-Security CE Interrupt Status Register (Default Value: 0x0000_0000)

| Offset: 0x000C | | | Register Name: CE_ISR_NS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:6 | R/W1C | 0x0 | TASK_CHAN3_STA Channel 3 task Status: Read 0x1: task completed successfully Read 0x2: task fail and stop Write 0x3 to clear it. |
| 5:4 | R/W1C | 0x0 | TASK_CHAN2_STA Channel 2 task Status: Read 0x1: task completed successfully Read 0x2: task fail and stop Write 0x3 to clear it. |
| 3:2 | R/W1C | 0x0 | TASK_CHAN1_STA Channel 1 task Status: Read 0x1: task completed successfully Read 0x2: task fail and stop Write 0x3 to clear it. |
| 1:0 | R/W1C | 0x0 | TASK_CHAN0_STA Channel 0 task Status: Read 0x1: task completed successfully Read 0x2: task fail and stop Write 0x3 to clear it. |

11.1.6.5 0x0010 Non-Security CE Task Load Register (Default Value: 0x0000_0000)

| Offset: 0x0010 | | | Register Name: CE_TLR_NS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:4 | / | / | / |
| 3 | R/W | 0x0 | <p>TASK_LOAD_RAES Task Load For Channel RAES type Read as 1: CE is busy to load another task, or the task queue is full. Write this bit is not allowed now. Read as 0: CE is ready to load another task Write 1: Load task and start to run. Write 0: No effect</p> |
| 2 | R/W | 0x0 | <p>TASK_LOAD_ASYM Task Load For Channel ASYM type Read as 1: CE is busy to load another task, or the task queue is full. Write this bit is not allowed now. Read as 0: CE is ready to load another task Write 1: Load task and start to run. Write 0: No effect</p> |
| 1 | R/W | 0x0 | <p>TASK_LOAD_HR Task Load For Channel HASH/RBG type Read as 1: CE is busy to load another task, or the task queue is full. Write this bit is not allowed now. Read as 0: CE is ready to load another task Write 1: Load task and start to run. Write 0: No effect</p> |
| 0 | R/W | 0x0 | <p>TASK_LOAD_SYMM Task Load For Channel SYMM type Read as 1: CE is busy to load another task, or the task queue is full. Write this bit is not allowed now. Read as 0: CE is ready to load another task Write 1: Load task and start to run. Write 0: No effect</p> |

11.1.6.6 0x0014 Non-Security CE Task Status Register (Default Value: 0x0000_0000)

| Offset: 0x0014 | | | Register Name: CE_TSR_NS |
|----------------|------------|-------------|--------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |

| Offset: 0x0014 | | | Register Name: CE_TSR_NS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 7:6 | R | 0x0 | <p>TASK_CHAN_XTS indicate which channel is run for XTS.</p> <p>0: task channel0 1: task channel1 2: task channel2 3: task channel3</p> |
| 5:4 | R | 0x0 | <p>TASK_CHAN_ASYMM indicate which channel is run for asymmetric.</p> <p>0: task channel0 1: task channel1 2: task channel2 3: task channel3</p> |
| 3:2 | R | 0x0 | <p>TASK_CHAN_DIG indicate which channel is run for digest.</p> <p>0: task channel0 1: task channel1 2: task channel2 3: task channel3</p> |
| 1:0 | R | 0x0 | <p>TASK_CHAN_SYMM indicate which channel is run for symmetric.</p> <p>0: task channel0 1: task channel1 2: task channel2 3: task channel3</p> |

11.1.6.7 0x0018 Non-Security CE Error Status Register (Default Value: 0x0000_0000)

| Offset: 0x0018 | | | Register Name: CE_ESR_NS |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:24 | R | 0x0 | <p>CHAN3_ERR_STATE Error code for task channel 3</p> <p>0x00: No error 0x01: algorithm not support 0x11: KEYSRAM access error 0x31: data length error 0x42: DRBG internal state invalid 0x44: DRBG Entropy or Nonce missing 0x46: DRBG do not support Prediction Resistance in current Instantiate other: reserved</p> |

| Offset: 0x0018 | | | Register Name: CE_ESR_NS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 23:16 | R | 0x0 | CHAN2_ERR_STATE Error code for task channel 2 0x00: No error 0x01: algorithm not support 0x11: KEYSRAM access error 0x31data length error 0x42 DRBG internal state invalid 0x44 DRBG Entropy or Nonce missing 0x46 DRBG do not support Prediction Resistance in current Instantiate other: reserved |
| 15:8 | R | 0x0 | CHAN1_ERR_STATE Error code for task channel 1 0x00: No error 0x01: algorithm not support 0x11: KEYSRAM access error 0x31data length error 0x42 DRBG internal state invalid 0x44 DRBG Entropy or Nonce missing 0x46 DRBG do not support Prediction Resistance in current Instantiate other: reserved |
| 7:0 | R | 0x0 | CHAN0_ERR_STATE Error code for task channel 0 0x00: No error 0x01: algorithm not support 0x11: KEYSRAM access error 0x31data length error 0x42 DRBG internal state invalid 0x44 DRBG Entropy or Nonce missing 0x46 DRBG do not support Prediction Resistance in current Instantiate other: reserved |

11.1.6.8 0x001C Non-Security CE DRBG Request Limit Register (Default Value: 0x0000_0000)

| Offset: 0x001C | | | Register Name: CE_DRL_NS |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R/W | 0x0 | MAX_NUM_REQ Maxim number of requests between reseeds in DRBG. Note: This register can only be configured when there is no DRBG task to be completed. |

11.1.6.9 0x0000 Security CE Task Descriptor Address0 Register (Default Value: 0x0000_0000)

| Offset: 0x0000 | | | Register Name: CE_TDA0_S |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R/W | 0x0 | TASK_DES_ADDR0 Task Descriptor Address is bit[31:0] (byte address) |

11.1.6.10 0x0004 Security CE Task Descriptor Address1 Register (Default Value: 0x0000_0000)

| Offset: 0x0004 | | | Register Name: CE_TDA1_S |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:0 | R/W | 0x0 | TASK_DES_ADDR1 Task Descriptor Address is bit[39:32] (byte address) |

11.1.6.11 0x0008 Security CE Interrupt Control Register (Default Value: 0x0000_0000)

| Offset: 0x0008 | | | Register Name: CE_ICR_S |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:4 | / | / | / |
| 3 | R/W | 0x0 | TASK_CHAN3_INT_EN Channel 3 Task Interrupt Enable 0: interrupt disable 1: interrupt enable |
| 2 | R/W | 0x0 | TASK_CHAN2_INT_EN Channel 2 Task Interrupt Enable 0: interrupt disable 1: interrupt enable |
| 1 | R/W | 0x0 | TASK_CHAN1_INT_EN Channel 1 Task Interrupt Enable 0: interrupt disable 1: interrupt enable |
| 0 | R/W | 0x0 | TASK_CHAN0_INT_EN Channel 0 Task Interrupt Enable 0: interrupt disable 1: interrupt enable |

11.1.6.12 0x000C Security CE Interrupt Status Register (Default Value: 0x0000_0000)

| Offset: 0x000C | | | Register Name: CE_ISR_S |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:6 | R/W1C | 0x0 | <p>TASK_CHAN3_STA Channel 3 task Status : Read 0x1: task completed successfully Read 0x2: task fail and stop Write 0x3 to clear it.</p> |
| 5:4 | R/W1C | 0x0 | <p>TASK_CHAN2_STA Channel 2 task Status : Read 0x1: task completed successfully Read 0x2: task fail and stop Write 0x3 to clear it.</p> |
| 3:2 | R/W1C | 0x0 | <p>TASK_CHAN1_STA Channel 1 task Status : Read 0x1: task completed successfully Read 0x2: task fail and stop Write 0x3 to clear it.</p> |
| 1:0 | R/W1C | 0x0 | <p>TASK_CHAN0_STA Channel 0 task Status : Read 0x1: task completed successfully Read 0x2: task fail and stop Write 0x3 to clear it.</p> |

11.1.6.13 0x0010 Security CE Task Load Register (Default Value: 0x0000_0000)

| Offset: 0x0010 | | | Register Name: CE_TLR_S |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:4 | / | / | / |
| 3 | R/W | 0x0 | <p>TASK_LOAD_RAES Task Load For Channel RAES type Read as 1: CE is busy to load another task, or the task queue is full. Write this bit is not allowed now. Read as 0: CE is ready to load another task Write 1: Load task and start to run. Write 0: No effect</p> |

| Offset: 0x0010 | | | Register Name: CE_TLR_S |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 2 | R/W | 0x0 | <p>TASK_LOAD_ASYM Task Load For Channel ASYM type Read as 1: CE is busy to load another task, or the task queue is full. Write this bit is not allowed now. Read as 0: CE is ready to load another task Write 1: Load task and start to run. Write 0: No effect</p> |
| 1 | R/W | 0x0 | <p>TASK_LOAD_HR Task Load For Channel HASH/RBG type Read as 1: CE is busy to load another task, or the task queue is full. Write this bit is not allowed now. Read as 0: CE is ready to load another task Write 1: Load task and start to run. Write 0: No effect</p> |
| 0 | R/W | 0x0 | <p>TASK_LOAD_SYMM Task Load For Channel SYMM type Read as 1: CE is busy to load another task, or the task queue is full. Write this bit is not allowed now. Read as 0: CE is ready to load another task Write 1: Load task and start to run. Write 0: No effect</p> |

11.1.6.14 0x0014 Security CE Task Status Register (Default Value: 0x0000_0000)

| Offset: 0x0014 | | | Register Name: CE_TSR_S |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:6 | R | 0x0 | <p>TASK_CHAN_XTS indicate which channel is run for XTS. 0: task channel0 1: task channel1 2: task channel2 3: task channel3</p> |
| 5:4 | R | 0x0 | <p>TASK_CHAN_ASYMM indicate which channel is run for asymmetric. 0: task channel0 1: task channel1 2: task channel2 3: task channel3</p> |

| Offset: 0x0014 | | | Register Name: CE_TSR_S |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 3:2 | R | 0x0 | <p>TASK_CHAN_DIG indicate which channel in run for digest.</p> <p>0: task channel0 1: task channel1 2: task channel2 3: task channel3</p> |
| 1:0 | R | 0x0 | <p>TASK_CHAN_SYMM indicate which channel in run for symmetric.</p> <p>0: task channel0 1: task channel1 2: task channel2 3: task channel3</p> |

11.1.6.15 0x0018 Security CE Error Status Register (Default Value: 0x0000_0000)

| Offset: 0x0018 | | | Register Name: CE_ESR_S |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:24 | R | 0x0 | <p>CHAN3_ERR_STATE Error code for task channel 3</p> <p>0x00: No error 0x01: algorithm not support 0x11: KEYSRAM access error 0x31data length error 0x42 DRBG internal state invalid 0x44 DRBG Entropy or Nonce missing 0x46 DRBG do not support Prediction Resistance in current Instantiate other: reserved</p> |
| 23:16 | R | 0x0 | <p>CHAN2_ERR_STATE Error code for task channel 2</p> <p>0x00: No error 0x01: algorithm not support 0x11: KEYSRAM access error 0x31data length error 0x42 DRBG internal state invalid 0x44 DRBG Entropy or Nonce missing 0x46 DRBG do not support Prediction Resistance in current Instantiate other: reserved</p> |

| Offset: 0x0018 | | | Register Name: CE_ESR_S |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 15:8 | R | 0x0 | CHAN1_ERR_STATE Error code for task channel 1 0x00: No error 0x01: algorithm not support 0x11: KEYSRAM access error 0x31: data length error 0x42: DRBG internal state invalid 0x44: DRBG Entropy or Nonce missing 0x46: DRBG do not support Prediction Resistance in current Instantiate other: reserved |
| 7:0 | R | 0x0 | CHAN0_ERR_STATE Error code for task channel 0 0x00: No error 0x01: algorithm not support 0x11: KEYSRAM access error 0x31: data length error 0x42: DRBG internal state invalid 0x44: DRBG Entropy or Nonce missing 0x46: DRBG do not support Prediction Resistance in current Instantiate other: reserved |

11.1.6.16 0x001C Security CE DRBG Request Limit Register (Default Value: 0x0000_0000)

| Offset: 0x001C | | | Register Name: CE_DRL_S |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R/W | 0x0 | MAX_NUM_REQ Maxim number of requests between reseeds in DRBG. Note: This register can only be configured after all tasks have been completed. |

11.1.6.17 0x0020 Security CE Symmetric algorithm DMA Current Source Address0 Register (Default Value: 0x0000_0000)

| Offset: 0x0020 | | | Register Name: CE_SCSA0_S |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R | 0x0 | SYMM_CUR_SRC_ADDR0 Symmetric algorithm current source address DMA reads. Bit[31:0], byte address. |

11.1.6.18 0x0024 Security CE Symmetric algorithm DMA Current Source Address1 Register (Default Value: 0x0000_0000)

| Offset: 0x0024 | | | Register Name: CE_SCSA1_S |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:0 | R | 0x0 | SYMM_CUR_SRC_ADDR1 Symmetric algorithm current source address DMA reads. Bit[39:32], byte address. |

11.1.6.19 0x0028 Security CE Symmetric algorithm DMA Current Destination Address0 Register (Default Value: 0x0000_0000)

| Offset: 0x0028 | | | Register Name: CE_SCDA0_S |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R | 0x0 | SYMM_CUR_DST_ADDR0 Symmetric algorithm current destination address DMA writes. Bit[31:0], byte address. |

11.1.6.20 0x002C Security CE Symmetric algorithm DMA Current Destination Address1 Register (Default Value: 0x0000_0000)

| Offset: 0x002C | | | Register Name: CE_SCDA1_S |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:0 | R | 0x0 | SYMM_CUR_DST_ADDR1 Symmetric algorithm current destination address DMA writes. Bit[39:32], byte address. |

11.1.6.21 0x0030 Security CE HASH algorithm DMA Current Source Address0 Register (Default Value: 0x0000_0000)

| Offset: 0x0030 | | | Register Name: CE_HCSA0_S |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R | 0x0 | HASH_CUR_SRC_ADDR0 HASH algorithm current source address DMA reads. Bit[31:0], byte address. |

11.1.6.22 0x0034 Security CE HASH algorithm DMA Current Source Address1 Register (Default Value: 0x0000_0000)

| Offset: 0x0034 | | | Register Name: CE_HCSA1_S |
|----------------|------------|-------------|---------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |

| Offset: 0x0034 | | | Register Name: CE_HCSA1_S |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 7:0 | R | 0x0 | HASH_CUR_SRC_ADDR1 HASH algorithm current source address DMA reads. Bit[39:32], byte address. |

11.1.6.23 0x0038 Security CE HASH algorithm DMA Current Destination Address0 Register (Default Value: 0x0000_0000)

| Offset: 0x0038 | | | Register Name: CE_HCDA0_S |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R | 0x0 | HASH_CUR_DST_ADDR0 HASH algorithm current destination address DMA writes. Bit[31:0], byte address. |

11.1.6.24 0x003C Security CE HASH algorithm DMA Current Destination Address1 Register (Default Value: 0x0000_0000)

| Offset: 0x003C | | | Register Name: CE_HCDA1_S |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:0 | R | 0x0 | HASH_CUR_DST_ADDR1 HASH algorithm current destination address DMA writes. Bit[39:32], byte address. |

11.1.6.25 0x0040 Security CE Asymmetric algorithm DMA Current Source Address0 Register (Default Value: 0x0000_0000)

| Offset: 0x0040 | | | Register Name: CE_ACSA0_S |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R | 0x0 | ASYMM_CUR_SRC_ADDR0 Asymmetric algorithm current source address DMA reads. Bit[31:0], byte address. |

11.1.6.26 0x0044 Security CE Asymmetric algorithm DMA Current Source Address1 Register (Default Value: 0x0000_0000)

| Offset: 0x0044 | | | Register Name: CE_ACSA1_S |
|----------------|------------|-------------|---------------------------|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |

| Offset: 0x0044 | | | Register Name: CE_ACSA1_S |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 7:0 | R | 0x0 | ASYMM_CUR_SRC_ADDR1 Asymmetric algorithm current source address DMA reads. Bit[39:32], byte address. |

11.1.6.27 0x0048 Security CE Asymmetric algorithm DMA Current Destination Address0 Register (Default Value: 0x0000_0000)

| Offset: 0x0048 | | | Register Name: CE_ACDA0_S |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R | 0x0 | ASYMM_CUR_DST_ADDR0 Asymmetric algorithm current destination address DMA writes. Bit[31:0], byte address. |

11.1.6.28 0x004C Security CE Asymmetric algorithm DMA Current Destination Address1 Register (Default Value: 0x0000_0000)

| Offset: 0x004C | | | Register Name: CE_ACDA1_S |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:0 | R | 0x0 | ASYMM_CUR_DST_ADDR1 Asymmetric algorithm current destination address DMA writes. Bit[39:32], byte address. |

11.1.6.29 0x0050 Security CE XTS algorithm DMA Current Source Address0 Register (Default Value: 0x0000_0000)

| Offset: 0x0050 | | | Register Name: CE_XCSA0_S |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R | 0x0 | XTS_CUR_CUR_ADDR0 XTS algorithm current source address DMA reads. |

11.1.6.30 0x0054 Security CE XTS algorithm DMA Current Source Address1 Register (Default Value: 0x0000_0000)

| Offset: 0x0054 | | | Register Name: CE_XCSA1_S |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:0 | R | 0x0 | XTS_CUR_CUR_ADDR1 XTS algorithm current source address DMA reads. |

11.1.6.31 0x0058 Security CE XTS algorithm DMA Current Destination Address0 Register (Default Value: 0x0000_0000)

| Offset: 0x0058 | | | Register Name: CE_XCDA0_S |
|----------------|------------|-------------|--|
| Bit | Read/Write | Default/Hex | Description |
| 31:0 | R | 0x0 | XTS_CUR_DST_ADDR0 XTS algorithm current destination address DMA writes. Bit[31:0], byte address. |

11.1.6.32 0x005C Security CE XTS algorithm DMA Current Destination Address1 Register (Default Value: 0x0000_0000)

| Offset: 0x005C | | | Register Name: CE_XCDA1_S |
|----------------|------------|-------------|---|
| Bit | Read/Write | Default/Hex | Description |
| 31:8 | / | / | / |
| 7:0 | R | 0x0 | XTS_CUR_DST_ADDR1 XTS algorithm current destination address DMA writes. Bit[39:32], byte address. |

11.2 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
- The module register is non-secure forever, the eFuse has secure zone and non-secure zone



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.

Contents

| | |
|--|------|
| 12 Carrier, Storage and Baking Information | 1162 |
| 12.1 Carrier | 1162 |
| 12.1.1 Matrix Tray Information..... | 1162 |
| 12.2 Storage | 1164 |
| 12.2.1 Moisture Sensitivity Level (MSL) | 1164 |
| 12.2.2 Bagged Storage Conditions | 1164 |
| 12.2.3 Out-of-bag Duration..... | 1164 |
| 12.2.4 Baking..... | 1165 |



Figures

Figure 12-1 V853/V853S Tray Dimension Drawing.....1163



Tables

| | |
|--|------|
| Table 12-1 Matrix Tray Carrier Information..... | 1162 |
| Table 12-2 V853/V853S Packing Quantity Information | 1162 |
| Table 12-3 MSL Summary | 1164 |
| Table 12-4 Bagged Storage Conditions | 1164 |
| Table 12-5 Out-of-bag Duration..... | 1164 |



12 Carrier, Storage and Baking Information

12.1 Carrier

12.1.1 Matrix Tray Information

The following table shows the V853/V853S matrix tray carrier information.

Table 12-1 Matrix Tray Carrier Information

| Item | Color | Size | Note |
|---|---------------|--|--|
| Tray | Black | 322.6 mm x 135.9 mm x 7.62 mm | 168 Qty/Tray |
| Aluminum foil bags | Silvery white | 540 mm x 300 mm x 0.14 mm | Vacuum packing Including HIC and desiccant Printing: RoHS symbol |
| Pearl cotton cushion (Vacuum bag) | White | 12 mm x 680 mm x 185 mm | |
| Pearl cotton cushion (The Gap between vacuum bag and inner box) | White | Left-Right: 12 mm x 180 mm x 85 mm Front-Back: 12 mm x 350 mm x 70 mm | |
| Inner Box | White | 396 mm x 196 mm x 96 mm | Printing: RoHS symbol 10 Tray/Inner box |
| Carton | White | 420 mm x 410 mm x 320 mm | 6 Inner box/Carton |

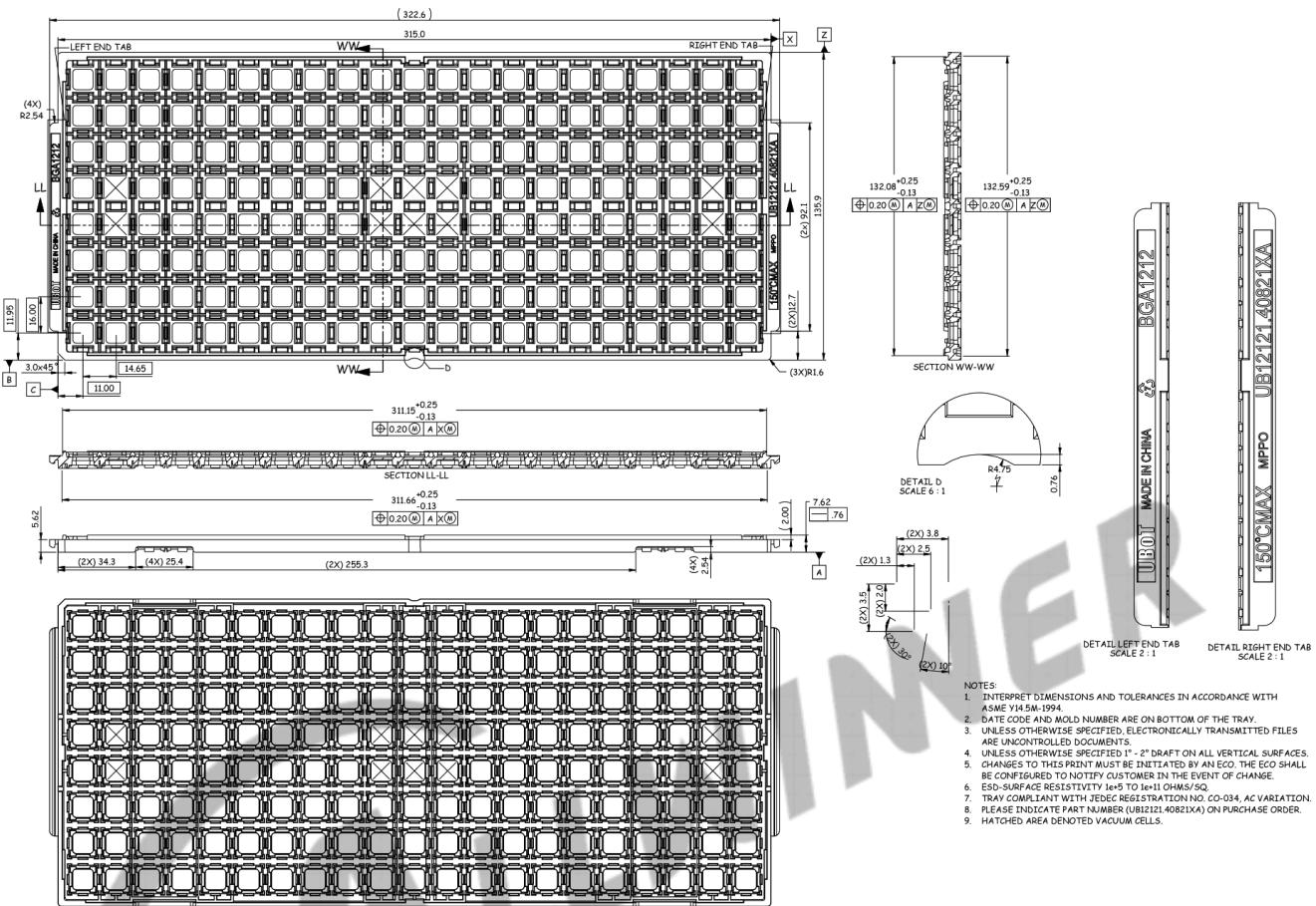
The following table shows the V853/V853S packing quantity.

Table 12-2 V853/V853S Packing Quantity Information

| Sample | Size (mm) | Qty/Tray | Tray/Inner Box | Full Inner Box Qty | Inner Box/Carton | Full Carton Qty |
|------------|-----------|----------|----------------|--------------------|------------------|-----------------|
| V853/V853S | 12x12 | 168 | 10 | 1680 | 6 | 10080 |

The following figure shows the tray dimension drawing of V853/V853S.

Figure 12-1 V853/V853S Tray Dimension Drawing



12.2 Storage

The reliability will be affected if any condition specified in the section 12.2.2 and the section 12.2.3 is exceeded.

12.2.1 Moisture Sensitivity Level (MSL)

A package's MSL indicates its ability to withstand the exposure after it is removed from its shipment bag. A low MSL device sample can be exposed on the factory floor longer than a high MSL device sample. Table 8 4 defines all MSL.



NOTE

V853/V853S device samples are classified as MSL3.

Table 12-3 MSL Summary

| MSL | Out-of-bag Floor Life | Comments |
|-----|-----------------------|---------------|
| 1 | Unlimited | ≤30°C / 85%RH |
| 2 | 1 year | ≤30°C / 60%RH |
| 2a | 4 weeks | ≤30°C / 60%RH |
| 3 | 168 hours | ≤30°C / 60%RH |
| 4 | 72 hours | ≤30°C / 60%RH |
| 5 | 48 hours | ≤30°C / 60%RH |
| 5a | 24 hours | ≤30°C / 60%RH |
| 6 | Time on Label (TOL) | ≤30°C / 60%RH |

12.2.2 Bagged Storage Conditions

The following table defines the shelf life of the V853/V853S device samples.

Table 12-4 Bagged Storage Conditions

| Packing mode | Vacuum packing |
|---------------------|----------------|
| Storage temperature | 20–26°C |
| Storage humidity | 40%–60%RH |
| Shelf life | 12 months |

12.2.3 Out-of-bag Duration

It is defined by the device MSL rating. The out-of-bag duration of V853/V853S is as follows.

Table 12-5 Out-of-bag Duration

| | |
|--------------------------------|-----------|
| Storage temperature | 20–26°C |
| Storage humidity | 40%–60%RH |
| Moisture sensitive level (MSL) | 3 |
| Floor life | 168 hours |

For the storage rules not mentioned in this document, refer to the latest **IPC/JEDEC J-STD-020C**.

12.2.4 Baking

It is not necessary to bake V853/V853S if the conditions specified in the section 12.2.2 and the section 12.2.3 have not been exceeded. It is necessary to bake V853/V853S if any condition specified in section 12.2.2 and section 12.2.3 has been exceeded.

It is necessary to bake V853/V853S if the storage humidity condition has been exceeded. If the device sample is removed from its shipment bag for more than 2 days, it shall be baked to guarantee the production.

Baking conditions: 125°C, 8 hours, nitrogen protection. Note that the baking should not exceed 1 time due to a risk of the deformation.



Figures

| | |
|--|------|
| Figure 13-1 Lead-free Reflow Profile | 1168 |
| Figure 13-2 Measuring the Reflow Soldering Process | 1169 |



Tables

Table 13-1 Lead-free Reflow Profile Conditions 1168



13 Reflow Profile

All Allwinner chips provided for clients are lead-free RoHS-compliant products.

The reflow profile recommended in this document is lead-free, it is suitable for the pure lead-free technology of lead-free solder paste. If customers need to use the lead solder paste, contact Allwinner FAE.

The following figure shows the appropriate reflow profile.

Figure 13-1 Lead-free Reflow Profile

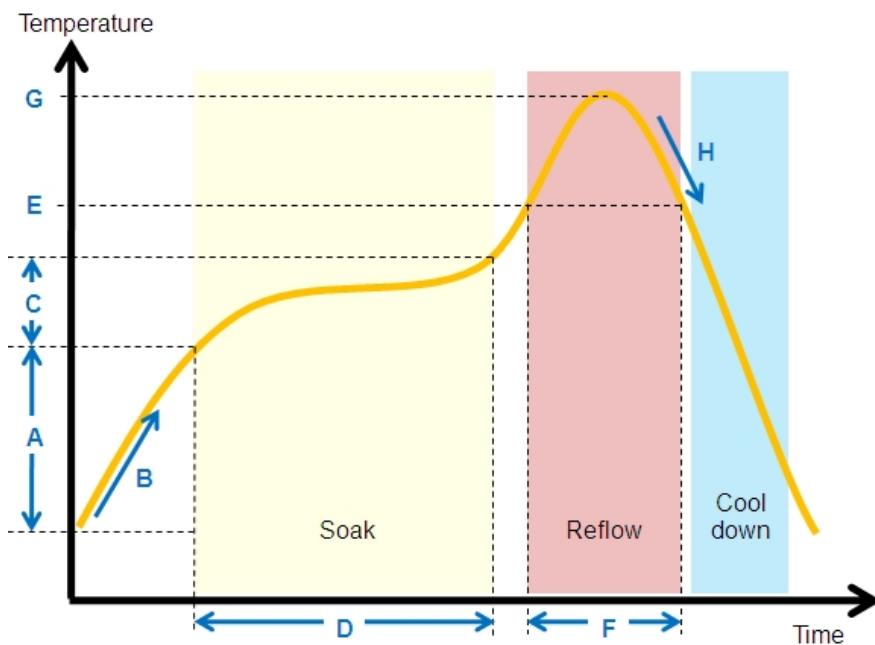


Table 13-1 Lead-free Reflow Profile Conditions

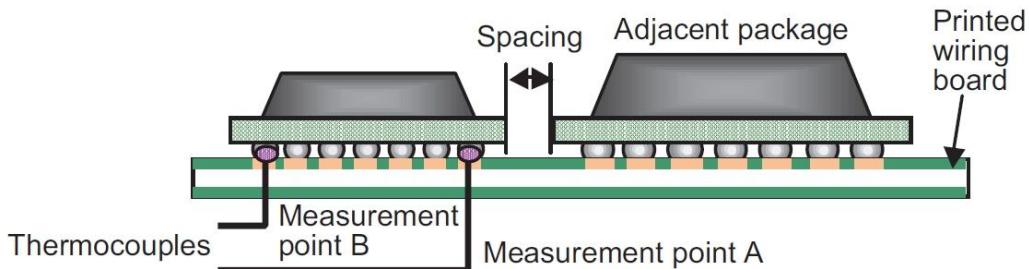
| QTI Typical SMT Reflow Profile Conditions (for Reference Only) | | |
|--|-----------------------------------|--------------------|
| | Step | Reflow Condition |
| Environment | N2 purge reflow usage (yes/no) | Yes, N2 purge used |
| | If yes, O2 ppm level | O2 < 1500 ppm |
| A | Preheat ramp up temperature range | 25°C -> 150°C |
| B | Preheat ramp up rate | 1.5~2.5 °C/sec |
| C | Soak temperature range | 150°C -> 190°C |
| D | Soak time | 80~110 sec |
| E | Liquidus temperature | 217°C |
| F | Time above liquidus | 60~90 sec |
| G | Peak temperature | 240~250°C |
| H | Cool down temperature rate | ≤4°C/sec |

The method of measuring the reflow soldering process is as follows.

Fix the thermocouple probe of the temperature measuring line at the connection point between the pin (solderable end) of the packaged device and the pad by using the high-temperature solder wire or the high-

temperature tape. Fix the packaged device at the pad by using the high-temperature tape or other methods, and cover over the thermocouple probe as shown in the following figure

Figure 13-2 Measuring the Reflow Soldering Process



 **NOTE**

To measure the temperature of the QFP-packaged chip, place the temperature probe directly at the pin.

If possible, the more accurate measuring way is to drill the packaged device, or drill the PCB, and fix the thermocouple probe through the drilled hole at the pad.

Contents

| | |
|------------------------|------|
| 14 FT/QA/QC Test | 1171 |
| 14.1 FT Test | 1171 |
| 14.2 QA Test..... | 1171 |
| 14.3 QC Test..... | 1171 |



14 FT/QA/QC Test

14.1 FT Test

FT test is the finished product testing after the chip is packaged, and it is a functional test of all modules for each produced chip.

14.2 QA Test

QA test is a system-level sampling test for good-quality chips. According to the application level of the chip, a certain percentage of good-quality chips are selected for system-level testing to make the chip work in a typical application scenario, and judge whether the chip works normally in this scenario.

14.3 QC Test

QC test is a module-level sampling test for good-quality chips. According to the chip application level, a certain percentage of good-quality chips are selected for module-level functional testing to monitor whether the chip production process is normal.

Figures

| | |
|--------------------------------|------|
| Figure 15-1 V853 Marking..... | 1174 |
| Figure 15-2 V853S Marking..... | 1175 |



Tables

| | |
|---|------|
| Table 15-1 V853 Marking Definitions..... | 1174 |
| Table 15-2 V853S Marking Definitions..... | 1175 |

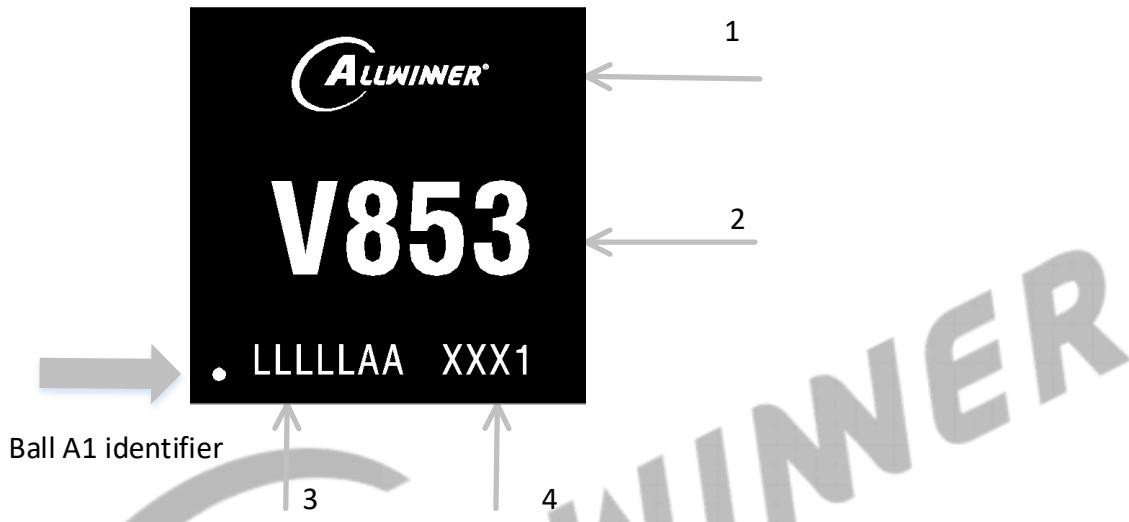


15 Part Marking

15.1 V853

The following figure shows the V853 marking.

Figure 15-1 V853 Marking



The following figure describes the V853 marking definitions.

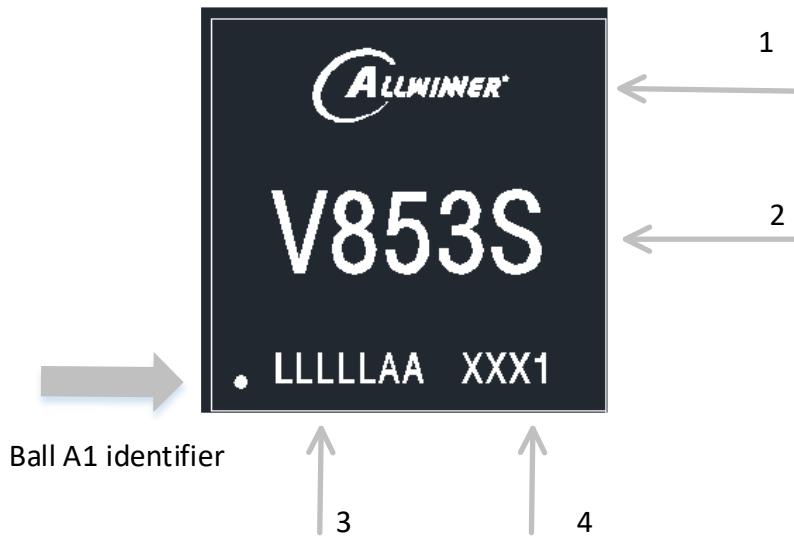
Table 15-1 V853 Marking Definitions

| No. | Marking | Description | Fixed/Dynamic |
|-----|-----------|------------------------|---------------|
| 1 | ALLWINNER | Allwinner logo or name | Fixed |
| 2 | V853 | Product name | Fixed |
| 3 | LLLLA | Lot number | Dynamic |
| 4 | XXX1 | Date code | Dynamic |

15.2 V853S

The following figure shows the V853S marking.

Figure 15-2 V853S Marking



The following figure describes the V853S marking definitions.

Table 15-2 V853S Marking Definitions

| No. | Marking | Description | Fixed/Dynamic |
|-----|-----------|------------------------|---------------|
| 1 | ALLWINNER | Allwinner logo or name | Fixed |
| 2 | V853S | Product name | Fixed |
| 3 | LLLLLAA | Lot number | Dynamic |
| 4 | XXX1 | Date code | Dynamic |

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.



Copyright © 2022 Allwinner Technology Co.,Ltd. All Rights Reserved.

Allwinner Technology Co.,Ltd.

No.9 Technology Road 2, High-Tech Zone,

Zhuhai, Guangdong Province, China

Contact US:

Service@allwinnertech.com

www.allwinnertech.com