



F102MX The Cost-Effective Multimedia Platform SoC

User Manual

Revision 0.10
November 19, 2025

Revision History

Revision	Date	Author	Description
0.10	November 19, 2025	AWA1233	Drafted Version



Contents

Revision History	i
Contents	ii
Figures	x
Tables	xvi
About This Document	xix
1 Product Description.....	22
1.1 Overview.....	22
1.2 Block Diagram.....	23
2 Memory Map	24
3 System Boot.....	27
3.1 Boot ROM (BROM).....	27
3.1.1 Overview.....	27
3.1.2 Boot Medium Select.....	27
3.1.3 FEL Process	29
4 Power, Reset, and Clock Management	31
4.1 Clock Controller Unit (CCU)	31
4.1.1 Overview.....	31
4.1.2 Block Diagram.....	31
4.1.3 Functional Description.....	32
4.1.4 Programming Guidelines.....	35
4.1.5 Register List.....	41
4.1.6 Register Description	43
5 Processors	103
5.1 RISC-V System.....	103
5.1.1 Overview.....	103
5.1.2 Block Diagram	104
6 Thermal Sensor.....	105
6.1 Thermal Sensor Controller (THS).....	105
6.1.1 Overview.....	105
6.1.2 Block Diagram	105

6.1.3	Functional Description.....	106
6.1.4	Register List.....	107
6.1.5	Register Description.....	107
7	Timers.....	112
7.1	Timer0.....	112
7.1.1	Overview.....	112
7.1.2	Block Diagram	113
7.1.3	Functional Description.....	113
7.1.4	Programming Guidelines.....	117
7.1.5	Register List.....	118
7.1.6	Register Description.....	118
7.2	High Speed Timer (HSTimer).....	129
7.2.1	Overview.....	129
7.2.2	Block Diagram	129
7.2.3	Functional Description.....	129
7.2.4	Programming Guidelines.....	132
7.2.5	Register List.....	133
7.2.6	Register Description.....	133
7.3	RISC-V Watchdog.....	140
7.3.1	Overview.....	140
7.3.2	Block Diagram	140
7.3.3	Functional Description.....	140
7.3.4	Programming Guidelines.....	142
7.3.5	Register List.....	142
7.3.6	Register Description.....	143
8	RTC.....	146
8.1	RTC.....	146
8.1.1	Overview.....	146
8.1.2	Block Diagram	147
8.1.3	External Signals	147
8.1.4	Functional Descriptions.....	147
8.1.5	Programming Guidelines.....	149
8.1.6	Register List.....	150

8.1.7	Register Description	151
9	Memory Controllers	160
9.1	SDRAM Controller (DRAMC)	160
9.2	SD/MMC Host Controller (SMHC).....	161
9.2.1	Overview.....	161
9.2.2	Block Diagram	162
9.2.3	External Signals	162
9.2.4	Functional Description.....	163
9.2.5	Programming Guidelines.....	170
9.2.6	Register List.....	176
9.2.7	Register Description	177
9.3	SPI Flash Controller (SPIF)	204
9.3.1	Overview.....	204
9.3.2	Block Diagram	205
9.3.3	External Signals	206
9.3.4	Functional Description.....	206
9.3.5	Programming Guidelines.....	220
9.3.6	Register List.....	225
9.3.7	Register Description.....	226
10	Interrupts.....	254
10.1	Platform-Level Interrupt Controller (PLIC).....	254
10.1.1	Overview.....	254
10.1.2	Functional Description.....	254
10.1.3	Register List.....	258
10.1.4	Register Description	259
11	DMA.....	263
11.1	DMA Controller (DMAC)	263
11.1.1	Overview.....	263
11.1.2	Block Diagram	264
11.1.3	Functional Description.....	265
11.1.4	Programming Guidelines.....	272
11.1.5	Register List.....	275
11.1.6	Register Description	276

12	Video&Graphic Engine	293
12.1	Video Encoder	293
12.2	Video Decoder	294
12.3	Graphics Processing	294
12.3.1	Display Engine (DE)	294
12.3.2	De-interlacer (DI)	295
12.3.3	Graphic 2D (G2D)	295
13	Video Output	297
13.1	LVDS	297
13.1.1	Overview	297
13.1.2	Block Diagram	297
13.1.3	External Signals	297
13.1.4	Functional Description	298
13.1.5	Programming Guidelines	300
13.1.6	Register List	301
13.1.7	Register Description	301
13.2	MIPI DSI	305
13.2.1	Overview	305
13.2.2	Block Diagram	306
13.2.3	External Signals	306
13.2.4	Functional Description	307
13.3	TCON LCD	307
13.3.1	Overview	307
13.3.2	Block Diagram	308
13.3.3	External Signals	309
13.3.4	Functional Description	312
13.3.5	Programming Guidelines	318
13.3.6	Register List	323
13.3.7	Register Description	324
13.4	VIDEO_OUT SYS	345
13.4.1	Overview	345
13.4.2	Block Diagram	345
13.4.3	Register List	345

13.4.4 Register Description	346
14 Audio Interfaces.....	347
14.1 Audio Codec	347
14.1.1 Overview.....	347
14.1.2 Block Diagram	348
14.1.3 External Signals	348
14.1.4 Functional Description.....	349
14.1.5 Programming Guidelines.....	356
14.1.6 Register List.....	357
14.1.7 Register Description	361
14.2 I2S/PCM	416
14.2.1 Overview.....	416
14.2.2 Block Diagram	417
14.2.3 External Signals	418
14.2.4 Functional Description.....	418
14.2.5 Programming Guidelines.....	420
14.2.6 Register List.....	421
14.2.7 Register Description	422
14.3 One Wire Audio (OWA).....	444
14.3.1 Overview.....	444
14.3.2 Block Diagram	445
14.3.3 External Signals	445
14.3.4 Functional Description.....	445
14.3.5 Programming Guidelines.....	450
14.3.6 Register List.....	452
14.3.7 Register Description	452
15 Peripherals.....	460
15.1 USB2.0 DRD (USB0)	460
15.1.1 Overview.....	460
15.1.2 Block Diagram	461
15.1.3 External Signals	461
15.1.4 Functional Description.....	462
15.2 PWM	463

15.2.1 Overview.....	463
15.2.2 Block Diagram	463
15.2.3 External Signals	464
15.2.4 Functional Description.....	464
15.2.5 Programming Guidelines.....	472
15.2.6 Register List.....	473
15.2.7 Register Description	474
15.3 PWM_BL.....	493
15.3.1 Overview.....	493
15.3.2 Block Diagram	494
15.3.3 External Signals	495
15.3.4 Typical application	495
15.3.5 Functional Description.....	495
15.3.6 Programming Guidelines.....	497
15.3.7 Register list.....	500
15.3.8 Register Description.....	502
15.4 Two Wire Interface (TWI).....	514
15.4.1 Overview.....	514
15.4.2 Block Diagram	514
15.4.3 External Signals	515
15.4.4 Functional Description.....	516
15.4.5 Programming Guidelines.....	520
15.4.6 Register List.....	523
15.4.7 Register Description	524
15.5 General Purpose ADC (GPADC).....	539
15.5.1 Overview.....	539
15.5.2 Block Diagram	539
15.5.3 External Signals	540
15.5.4 Functional Description.....	540
15.5.5 Programming Guidelines.....	541
15.5.6 Register List.....	542
15.5.7 Register Description	543
15.6 Touch Panel ADC (TPADC)	554
15.6.1 Overview.....	554

15.6.2	Block Diagram	554
15.6.3	External Signals	555
15.6.4	Functional Description.....	555
15.6.5	Register List.....	560
15.6.6	Register Description	561
15.7	GPIO	568
15.7.1	Overview.....	568
15.7.2	Block Diagram	569
15.7.3	Functional Description.....	569
15.7.4	Register List.....	576
15.7.5	Register Description	577
15.8	UART	661
15.8.1	Overview.....	661
15.8.2	Block Diagram	662
15.8.3	External Signals	662
15.8.4	Functional Description.....	663
15.8.5	Programming Guidelines.....	669
15.8.6	Register List.....	672
15.8.7	Register Description.....	673
15.9	SPI	700
15.9.1	Overview.....	700
15.9.2	Block Diagram	701
15.9.3	External Signals	702
15.9.4	Functional Description.....	702
15.9.5	Programming Guidelines.....	708
15.9.6	Register List.....	711
15.9.7	Register Description	712
15.10	SPI_DBI.....	731
15.10.1	Overview	731
15.10.2	Block Diagram	732
15.10.3	External Signals.....	733
15.10.4	Functional Description	734
15.10.5	Programming Guidelines	748
15.10.6	Register List	752

15.10.7	Register Description	753
15.11	CIR Receiver (IR_RX)	783
15.11.1	Overview	783
15.11.2	Block Diagram	783
15.11.3	External Signals.....	783
15.11.4	Functional Description	784
15.11.5	Operation Guidelines.....	787
15.11.6	Register List	788
15.11.7	Register Description	788
16	Security.....	793
16.1	Crypto Engine(CE)	793
16.1.1	Overview.....	793
16.1.2	Block Diagram	793
16.1.3	Functional Discription.....	794
16.1.4	Operation Guide	795
16.1.5	Register List.....	795
16.1.6	Register Description.....	796

Figures

Figure 1-1 System Block Diagram	23
Figure 3-1 USB FEL Process Diagram.....	30
Figure 4-1 CCU Block Diagram	31
Figure 4-2 PLL Distribution	32
Figure 5-1 RISC-V System Block Diagram	104
Figure 6-1 THS Block Diagram.....	105
Figure 6-2 Thermal Sensor Timing Requirement	106
Figure 6-3 Thermal Sensor Controller Interrupt Source	106
Figure 7-1 Timer Block Diagram.....	113
Figure 7-2 Block Diagram for the Timer.....	113
Figure 7-3 Block Diagram for the Watchdog	115
Figure 7-4 Block Diagram for the AVS	116
Figure 7-5 HSTimer Block Diagram.....	129
Figure 7-6 Typical Application for HSTimer	130
Figure 7-7 HSTimer Initialization Process	131
Figure 7-8 Block Diagram for the RISC-V Watchdog.....	140
Figure 7-9 RISC-V Watchdog Application Diagram	140
Figure 8-1 RTC Block Diagram.....	147
Figure 8-2 RTC Counter.....	148
Figure 8-3 RTC 1 kHz Counter Step Structure	148
Figure 9-1 SMHC Block Diagram.....	162
Figure 9-2 Command Token Format.....	164
Figure 9-3 Response Token Format	164
Figure 9-4 Data Packet Format for SDR.....	165
Figure 9-5 Data Packet Format for DDR.....	165
Figure 9-6 Single-Block and Multi-Block Read Operation.....	166
Figure 9-7 Single-Block and Multi-Block Read Operation.....	166
Figure 9-8 IDMAC Descriptor Structure Diagram	167
Figure 9-9 Phase Offset of Command and Data in SDR Mode	169
Figure 9-10 Phase Offset of Command and Data in DDR4 Mode (SMHC_NTSR [31] = 0)	169
Figure 9-11 Phase Offset of Command and Data in DDR4 (2x Mode) (SMHC_NTSR [31] = 1).....	170

Figure 9-12 SPI_Flash Block Diagram	205
Figure 9-13 Typical Application.....	206
Figure 9-14 SPI Transfer Mode	209
Figure 9-15 SPI STR Transfer	209
Figure 9-16 SPI DTR Transfer Example, 1-4-4.....	210
Figure 9-17 SPI DTR Transfer Example, 4-4-4.....	210
Figure 9-18 SPI CS Select Timing Relation	211
Figure 9-19 SPI 3-Wire Mode.....	212
Figure 9-20 SPI CMD with Single IO.....	212
Figure 9-21 SPI Write CMD and DATA with One Wire	212
Figure 9-22 SPI Write CMD and Read DATA with One Wire (No Dummy)	213
Figure 9-23 SPI Write CMD and Read DATA with One Wire (with Dummy)	213
Figure 9-24 SPI Dual Input/Dual Output Mode (1-1-2)	214
Figure 9-25 SPI Dual Input/Dual Output Mode (1-2-2)	214
Figure 9-26 SPI Quad Input/Quad Output Mode (1-1-4).....	215
Figure 9-27 SPI Quad Input/ Quad Output Mode (1-4-4).....	215
Figure 9-28 SPI Quad Input/ Quad Output Mode (4-4-4).....	215
Figure 9-29 Descriptor Structure Diagram	216
Figure 9-30 SPI Programming flow	221
Figure 9-31 SPI Transfer Phase Flow Diagram	223
Figure 11-1 DMAC Block Diagram.....	264
Figure 11-2 DMA Descriptor	267
Figure 11-3 DMA Linked List Transfer	268
Figure 11-4 Workflow of the DMAC Handshake Mode	271
Figure 11-5 DMAC Transfer Process	273
Figure 13-1 LVDS Block Diagram.....	297
Figure 13-2 VESA 6-Bit format	298
Figure 13-3 Single LVDS VSEA Mode	299
Figure 13-4 Single LVDS JEDIA Mode	299
Figure 13-5 Interface Mapping	300
Figure 13-6 MIPI DSI Block Diagram.....	306
Figure 13-7 DISPLL Distribution	307
Figure 13-8 TCON_LCD Block Diagram.....	308
Figure 13-9 HV Interface Vertical Timing	314

Figure 13-10 HV Interface Horizontal Timing.....	315
Figure 13-11 Single-channel Transmission Timing	316
Figure 13-12 CPU interface timing (left read, right write)	317
Figure 13-13 The Data Timing of MIPI DSI Video Mode	318
Figure 13-14 HV Mode Initial Process.....	318
Figure 13-15 i8080 Mode Initial Process	320
Figure 13-16 VIDEO_OUT SYS Block Diagram	345
Figure 14-1 Audio Codec Block Diagram	348
Figure 14-2 Audio Codec Clock Diagram	349
Figure 14-3 Audio Codec Digital Part Reset System	350
Figure 14-4 Audio Codec Analog Part Reset System	350
Figure 14-5 Audio Codec Data Path Diagram.....	351
Figure 14-6 Headphone Output Application.....	352
Figure 14-7 Audio Codec Interrupt System	352
Figure 14-8 DAP Data Flow	353
Figure 14-9 HPF Logic Structure	353
Figure 14-10 DRC static Curve Parameters.....	354
Figure 14-11 DRC Block Diagram	354
Figure 14-12 Energy Filter Structure.....	355
Figure 14-13 Gain Smooth Filter	356
Figure 14-14 I2S/PCM Interface System Block Diagram.....	417
Figure 14-15 Typical Application of I2S/PCM Interface.....	417
Figure 14-16 I2S Standard Mode Timing	418
Figure 14-17 Left-Justified Mode Timing.....	419
Figure 14-18 Right-Justified Mode Timing	419
Figure 14-19 PCM Long Frame Mode Timing	419
Figure 14-20 PCM Short Frame Mode Timing.....	420
Figure 14-21 I2S/PCM Operation Flow.....	420
Figure 14-22 OWA Block Diagram	445
Figure 14-23 OWA Biphase-Mark Code	446
Figure 14-24 OWA Sub-Frame Format	446
Figure 14-25 OWA Frame/Block Format	447
Figure 14-26 Data-Burst Format.....	448
Figure 14-27 Data-burst Preamble.....	448

Figure 14-28 Data-burst Preamble words	449
Figure 14-29 Fields of Burst-information	449
Figure 14-30 Length of the Burst-Payload Specified by Pd	450
Figure 14-31 OWA Operation Flow	451
Figure 15-1 USB2.0 DRD Controller Block Diagram	461
Figure 15-2 USB2.0 DRD Controller and PHY Connection Diagram	462
Figure 15-3 PWM Block Diagram	464
Figure 15-4 PWM01 Clock Controller Diagram	465
Figure 15-5 PWM01 Output Logic Module Diagram.....	465
Figure 15-6 PWM01 Complementary Pair Output	467
Figure 15-7 Output Waveform with Dead-time Insertion.....	468
Figure 15-8 PWM0 High Level Active State	469
Figure 15-9 PWM0 Low Level Active State	469
Figure 15-10 Phase of PWM0 High Level Active State.....	470
Figure 15-11 Group 0-3 PWM Signal Output.....	470
Figure 15-12 PWM01 Capture Logic Module Diagram	471
Figure 15-13 PWM0 Channel Capture Timing	471
Figure 15-14 PWM_BL Block Diagram.....	494
Figure 15-15 PWM_BL0 Working Diagram	495
Figure 15-16 PWM_BL0 Duty Cycle Adjustment Step	496
Figure 15-17 PWM_BL0 Duty Cycle Adjustment Step Interval	496
Figure 15-18 PWM_BL0 Abnormal Protection.....	496
Figure 15-19 TWI Block Diagram	514
Figure 15-20 Write Timing in 7-bit Standard Addressing Mode.....	516
Figure 15-21 Read Timing in 7-bit Standard Addressing Mode	517
Figure 15-22 Write Timing in 10-bit Extended Addressing Mode	517
Figure 15-23 Read Timing in 10-bit Extended Addressing Mode.....	517
Figure 15-24 TWI Driver Write Packet Transmission.....	518
Figure 15-25 TWI Driver Read Packet Transmission	519
Figure 15-26 TWI Programming State Diagram	520
Figure 15-27 GPADC Block Diagram.....	539
Figure 15-28 GPADC Timing Requirement	540
Figure 15-29 GPADC Initial Process.....	541
Figure 15-30 TPADC Block Diagram	554

Figure 15-31 TPADC Single-Ended Mode for AUX ADC	555
Figure 15-32 TPADC Differential Mode for Touch Panel.....	556
Figure 15-33 Single Touch X-Coordinate Measurement for Touch Panel.....	556
Figure 15-34 Dual Touch Detection for Touch Panel.....	557
Figure 15-35 Touch Pressure Measurement for Touch Panel.....	558
Figure 15-36 Pen Down Detection for Touch Panel.....	559
Figure 15-37 GPIO Block Diagram	569
Figure 15-38 Pull-up, Pull-down, and High-impedance Input Mode	570
Figure 15-39 IO Buffer Strength Diagram	571
Figure 15-40 UART Block Diagram	662
Figure 15-41 I/O Signals of UART Controllers (2/4-wire)	663
Figure 15-42 UART Serial Data Format	664
Figure 15-43 Application Diagram for RTS/CTS Auto flow Control	664
Figure 15-44 RTS/CTS Auto Flow Control Data Format.....	665
Figure 15-45 Application Diagram for IrDA Transceiver	665
Figure 15-46 Serial IrDA Data Format	665
Figure 15-47 Application Diagram for RS-485 Transceiver	666
Figure 15-48 RS-485 Data Format	666
Figure 15-49 SPI Block Diagram	701
Figure 15-50 SPI Application Block Diagram.....	703
Figure 15-51 SPI Phase 0 Timing Diagram.....	704
Figure 15-52 SPI Phase 1 Timing Diagram.....	704
Figure 15-53 SPI 3-Wire Mode.....	705
Figure 15-54 SPI Dual-Input/Dual-Output Mode	705
Figure 15-55 SPI Dual I/O Mode.....	706
Figure 15-56 SPI Quad-Input/Quad-Output Mode.....	706
Figure 15-57 Data Transfer in Query Mode(including CPU and DMA mode)	709
Figure 15-58 SPI_DBI Block Diagram	732
Figure 15-59 SPI Application Block Diagram.....	734
Figure 15-60 DBI Application Block Diagram	735
Figure 15-61 DBI 3-Line Display Bus Serial Interface Write Operation Format	736
Figure 15-62 DBI 3-Line Display Bus Serial Interface 8-bit Read Operation Format.....	736
Figure 15-63 DBI 3-Line Display Bus Serial Interface 24-bit Read Operation Format.....	737
Figure 15-64 DBI 3-Line Display Bus Serial Interface 32-bit Read Operation Format.....	737

Figure 15-65 DBI 4-Line Display Bus Serial Interface Write Operation Format	738
Figure 15-66 DBI 4-Line Display Bus Serial Interface 8-bit Read Operation Format.....	738
Figure 15-67 DBI 4-Line Display Bus Serial Interface 24-bit Read Operation Format.....	739
Figure 15-68 DBI 4-Line Display Bus Serial Interface 32-bit Read Operation Format.....	739
Figure 15-69 RGB111 3-Line Interface Transmit Video Format	740
Figure 15-70 RGB444 3-Line Interface Transmit Video Format	740
Figure 15-71 RGB565 3-Line Interface Transmit Video Format	741
Figure 15-72 RGB666 3-Line Interface Transmit Video Format	741
Figure 15-73 RGB111 4-Line Interface Transmit Video Format	742
Figure 15-74 RGB444 4-Line Interface Transmit Video Format	742
Figure 15-75 RGB565 4-Line Interface Transmit Video Format	743
Figure 15-76 RGB666 4-Line Interface Transmit Video Format	743
Figure 15-77 RGB444 2 Data Lane Interface Transmit Video Format.....	744
Figure 15-78 RGB565 2 Data Lane Interface Transmit Video Format.....	745
Figure 15-79 RGB666 2 Data Lane Interface Transmit Video Format 0.....	745
Figure 15-80 RGB666 2 Data Lane Interface Transmit Video Format 1 (ilitek)	746
Figure 15-81 RGB666 2 Data Lane Interface Transmit Video Format 2 (New vision)	746
Figure 15-82 RGB888 2 Data Lane Interface Transmit Video Format.....	746
Figure 15-83 CIR Receiver Block Diagram	783
Figure 15-84 CIR Receiver Application Diagram	784
Figure 15-85 A Typical Pulse Train of the NEC	784
Figure 15-86 Logical '0' and Logical '1' of NEC Protocol	785
Figure 15-87 ATHR Definition	785
Figure 15-88 ITHR Definition.....	786
Figure 15-89 NTHR Definition	786
Figure 15-90 APAM Definition	786
Figure 15-91 CIR Receiver Process	787
Figure 16-1 CE Block Diagram	794

Tables

Table 3-1 BOOT_MODE Setting	27
Table 3-2 GPIO Boot Select Configuration	28
Table 3-3 Groups of eFuse_Boot_Select	28
Table 3-4 eFuse Boot Select Setting	28
Table 4-1 PLL Typical Application.....	32
Table 4-2 PLL Specifications	34
Table 4-3 Recommended Values for Configuration Factors of PLL_AUDIO1	38
Table 8-1 RTC External Signals.....	147
Table 8-2 RTC Counter Changing Range	148
Table 9-1 SMHC Sub-blocks.....	162
Table 9-2 SMHC External Signals	162
Table 9-3 DES0 Definition	167
Table 9-4 DES1 Definition	168
Table 9-5 DES2 Definition	168
Table 9-6 DES3 Definition	168
Table 9-7 Phase Offset of Command and Data in SDR Mode	169
Table 9-8 Phase Offset of Command and Data in DDR4 Mode	170
Table 9-9 Phase Offset of Command and Data in DDR4 (2x) Mode	170
Table 9-10 SPI_Flash Sub-blocks	205
Table 9-11 SPI_Flash External Signals.....	206
Table 9-12 SPI_Flash Support List	207
Table 9-13 SPI Flash Modes with Clock Polarity and Phase	208
Table 10-1 Interrupt Sources.....	254
Table 11-1 DMAC Sub-blocks.....	264
Table 11-2 DMA DRQ Types	265
Table 11-3 DMA Address Descriptor.....	268
Table 13-1 LVDS External Signals.....	298
Table 13-2 MIPI DSI External Signals	306
Table 13-3 DISPPLL Specifications.....	307
Table 13-4 TCON_LCD Sub-blocks.....	308
Table 13-5 TCON_LCD External Signals.....	309

Table 13-6 The Correspondence between LCD and RGB	309
Table 13-7 The Correspondence among LCD, LVDS, and DSI.....	310
Table 13-8 HV Panel Signals	314
Table 13-9 Analysis of XY Signal	316
Table 13-10 CPU Panel Signals.....	316
Table 13-11 HV Mode Configuration Example	319
Table 13-12 BT.656 Mode Configuration Example	319
Table 13-13 i8080 Mode Configuration Example	321
Table 14-1 Audio Codec External Signals.....	348
Table 14-2 I2S/PCM External Signals	418
Table 14-3 OWA External Signals	445
Table 14-4 Biphase-Mark Encoder	446
Table 14-5 Preamble Codes.....	447
Table 14-6 Bit Allocation of Data-Burst in IEC 60958 Subframes	448
Table 14-7 The Relation between Different System Clock and Sample Ratio	450
Table 15-1 USB2.0 DRD External Signals.....	461
Table 15-2 PWM External Signals	464
Table 15-3 PWM_BL External Signals	495
Table 15-4 PWM_BL_100 System Source Configuration.....	497
Table 15-5 PWM_BL0 Output Configuration without GPADC Voltage Quantization Mode.....	498
Table 15-6 PWM_BL0 Output Configuration with GPADC Voltage Quantization Mode	499
Table 15-7 TWI Sub-blocks	515
Table 15-8 TWI External Signals	515
Table 15-9 GPADC External Signals.....	540
Table 15-10 TPADC External Signals.....	555
Table 15-11 Median and Averaging Filter Size	559
Table 15-12 Median and Averaging Filter Example	560
Table 15-13 Port Function	570
Table 15-14 Multi-function Port	572
Table 15-15 Port A Multiplex Function.....	574
Table 15-16 Port B Multiplex Function	574
Table 15-17 Port C Multiplex Function	574
Table 15-18 Port D Multiplex Function	574
Table 15-19 Port E Multiplex Function.....	575

Table 15-20 Port F Multiplex Function	575
Table 15-21 UART External Signals	662
Table 15-22 Errors of Baud Rate in UART Mode	666
Table 15-23 Errors of Baud Rate in IrDA Mode	667
Table 15-24 Errors of Baud Rate in RS485 Mode	667
Table 15-25 Errors of Baud Rate in 19.2M and 26M Clock Source	668
Table 15-26 SPI Sub-blocks	701
Table 15-27 SPI External Signals	702
Table 15-28 SPI Transmit Format	703
Table 15-29 SPI Old Sample Mode	710
Table 15-30 SPI New Sample Mode	710
Table 15-31 SPI_DBI Sub-blocks	732
Table 15-32 GPIO multiplexing of SPI1 and DBI	733
Table 15-33 SPI_DBI External Signals	733
Table 15-34 2 data lane sclk frequency	747
Table 15-35 3 line sclk frequency	747
Table 15-36 4 line sclk frequency	747
Table 15-37 CIR Receiver External Signals	783

About This Document

Purpose and Target Readers

This document details the features, logical structures, functional description, programming guidelines, and related registers of each module about the SoC. A basic knowledge of digital circuits, logical circuits, SoCs, and so on is necessary to use this manual. Particular attention should be paid to the cautions and notes within the body of the text.

Related Documentation

For a complete listing of related documentation and development-support tools for the device, contact the Allwinner FAE or access the Allwinner Customer Service Platform by visiting <https://open.allwinnertech.com/>.

Revision Number Definition

Revision 0.10-0.1x

This document is released based on the experimental sample(s). Therefore, it may be modified, calibrated and supplemented hereafter.

All statements, information and recommendations in this document do not constitute any express or implied representation or warranty (including, but not limited to, the warranties of fitness for a particular purpose, merchantability, non-infringement, and accuracy and completeness of the document). Allwinner shall not be liable for any person's use of such information or/and this document.

If you have any questions about the document, please contact us to confirm and obtain the latest version.

Symbol Conventions

The symbols that may be found in this document are defined as follows.

Symbol	Description
 CAUTION	A caution means that damage to equipment is possible.
 NOTE	Provides additional information to emphasize or supplement important points of the main text.

Table Content Conventions

The table content conventions that may be found in this document are defined as follows.

Symbol	Description
-	The cell is blank.

Notation of Symbols and Numbers

The expressions of the data capacity, the frequency, and the data rate are described as follows.

Type	Symbol	Value
Data capacity	K	1024
	M	1,048,576
	G	1,073,741,824
Frequency, data rate	k	1000
	M	1,000,000
	G	1,000,000,000

The expressions of addresses and data are described as follows.

Number	Example	Description
0x	0x0200,0x79	Address or data in hexadecimal
0b	0b010,0b00 000 111	Data or sequence in binary (register description is excluded.)
X	00X,XX1	In data expression, X indicates 0 or 1. For example, 00X indicates 000 or 001, XX1 indicates 001,011,101 or 111.

Register Attributes

The register attributes that may be found in this document are defined as follows.

Abbr. & Symbol	Description
R	Read Only
R/W	Read/Write
R/WAC	Read/Write-Automatic-Clear, clear the bit automatically when the operation of complete. Writing 0 has no effect.
R/WC	Read/Write-Clear
R/W0C	Read/Write 0 to Clear. Writing 1 has no effect
R/W1C	Read/Write 1 to Clear. Writing 0 has no effect
R/W1S	Read/Write 1 to Set. Writing 0 has no effect
W	Write Only

Abbr. & Symbol	Description
/	If the default value in the row of a bit or multiple bits is “/” , this bit is unused.
UDF	If the default value of a bit or multiple bits is “UDF” , this default value is undefined.



1 Product Description

1.1 Overview

The F102MX is an advanced application processor designed specifically for video decoding market, integrating a domestic RISC-V CPU instruction architecture.

This advanced SoC supports a wide range of decoding formats, including H.264, MPEG-1/2/4, JPEG, PNG, etc. Additionally, its independent hardware encoder can perform encoding in JPEG or MJPEG.

This advanced SoC also supports rich display interfaces such as RGB, LVDS, and MIPI. Integrated multi-channel ADCs/DACs and audio interfaces (I2S/PCM/OWA) provide an excellent voice interaction solution.

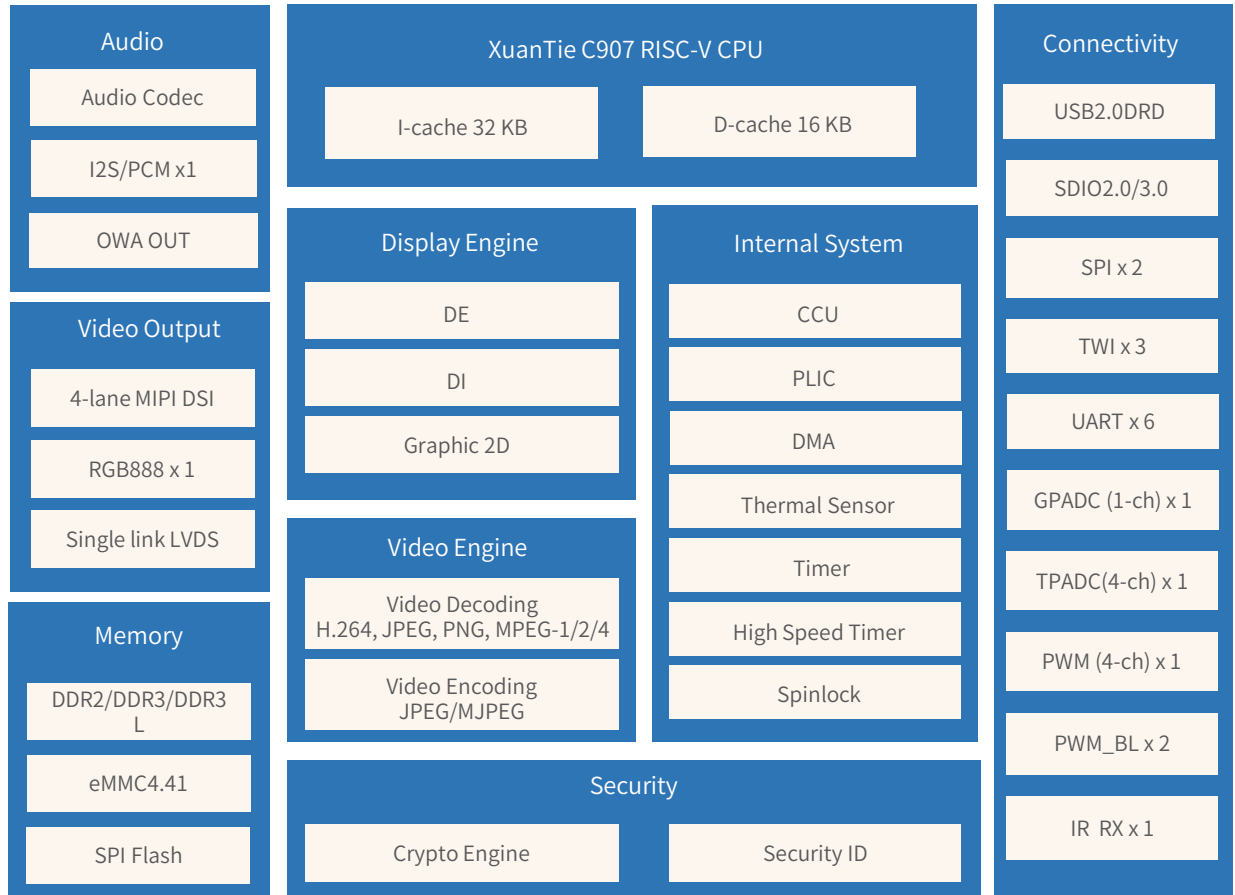
The F102MX can be used in network video players, digital photo frames, smart vision transmission screen, and so on.



1.2 Block Diagram

The following figure shows the system block diagram.

Figure 1-1 System Block Diagram



2 Memory Map

The following table shows the memory map viewed from RISC-V.

Region	Address	Size (Bytes)
SRAM_SHARE		
VE_SHARE_SRAM	0x0002 0000---0x0002 AFFF	44 KB Only when the relative bit of BOOT SRAM Remap Register is set to 1.
DE_SHARE_SRAM	0x0002 B000---0x0002 EFFF	16 KB Only when the relative bit of BOOT SRAM Remap Register is set to 1.
USB_SHARE_SRAM0	0x0002 F000---0x0002 FFFF	4 KB Only when the relative bit of BOOT SRAM Remap Register is set to 1.
USB_SHARE_SRAM1	0x0003 0000---0x0003 0FFF	4 KB Only when the relative bit of BOOT SRAM Remap Register is set to 1.
SHRAM_VE_SYS		
VE	0x01C0 E000---0x01C0 FFFF	8 KB
SP0		
GPIO	0x0200 0000---0x0200 07FF	2 KB
PWM	0x0200 0C00---0x0200 0FFF	1 KB
CCU	0x0200 1000---0x0200 1FFF	4 KB
GPADC	0x0200 9000---0x0200 93FF	1 KB
THS	0x0200 9400---0x0200 97FF	1 KB
TPADC	0x0200 9C00---0x0200 9FFF	1 KB
PWM_BL0	0x0200 A000---0x0200 A3FF	1 KB
AUDIODCODEC	0x0203 0000---0x0203 0FFF	4 KB
I2S0	0x0203 2000---0x0203 2FFF	4 KB
OWA	0x0203 6000---0x0203 63FF	1 KB
IR_RX0	0x0208 7000---0x0208 7FFF	4 KB
TIMER	0x0205 0000---0x0205 0FFF	4 KB
SP1		
UART0	0x0250 0000---0x0250 03FF	1 KB
UART1	0x0250 0400---0x0250 07FF	1 KB
UART2	0x0250 0800---0x0250 0BFF	1 KB
UART3	0x0250 0C00---0x0250 0FFF	1 KB
UART4	0x0250 1000---0x0250 13FF	1 KB
UART5	0x0250 1400---0x0250 17FF	1 KB
SP2		
TWI0	0x0250 2000---0x0250 23FF	1 KB

Region	Address	Size (Bytes)
TWI1	0x0250 2400---0x0250 27FF	1 KB
TWI2	0x0250 2800---0x0250 2BFF	1 KB
SH0		
SYSCTRL	0x0300 0000---0x0300 0FFF	4 KB
DMAC	0x0300 2000---0x0300 2FFF	4 KB
SPINLOCK	0x0300 5000---0x0300 5FFF	4 KB
SID	0x0300 6000---0x0300 6FFF	4 KB
HSTIMER	0x0300 8000---0x0300 8FFF	4 KB
DCU	0x0301 0000---0x0301 FFFF	64 KB
CE_NS	0x0304 0800---0x0304 0FFF	2 KB
MSI+MEMC	0x0310 2000---0x0330 1FFF	2 MB
RTC	0x0309 0000---0x0309 03FF	1 KB
SH2		
SMHC0	0x0402 0000---0x0402 0FFF	4 KB
SMHC2	0x0402 2000---0x0402 2FFF	4 KB
SPI0	0x0402 5000---0x0402 5FFF	4 KB
SPI1	0x0402 6000---0x0402 6FFF	4 KB
USB0	0x0410 0000---0x041F FFFF	1 MB
SH3-SPI_FLASH		
SPI_FLASH0	0x0C00 0000---0x0C00 0FFF	4 KB
SPI_FLASH_XIP	0x0E00 0000---0x0FFF FFFF	32 MB The SRAM can only be accessed when SPI_FLASH is set to XIP mode.
SH4_VIDEO_OUT_SYS		
DE	0x0500 0000---0x052F FFFF	3 MB
DI	0x0540 0000---0x0540 FFFF	64 KB
G2D	0x0541 0000---0x0544 FFFF	256 KB
DSI	0x0545 0000---0x0545 1FFF	8 KB
DISPLAY_TOP	0x0546 0000---0x0546 0FFF	4 KB
TCON_LCD0	0x0546 1000---0x0546 1FFF	4 KB
SH4_RISCV_SYS		
RISCV_BROM	0x0600 0000---0x0600 FFFF	64 KB Address mapping which RISC-V core accesses BROM: 0x0000 0000---0x0000 FFFF
RISCV_CFG	0x0601 0000---0x0601 0FFF	4 KB
RISCV_WDG	0x0601 1000---0x0601 1FFF	4 KB
RISCV_TIMESTAMP	0x0601 2000---0x0601 2FFF	4 KB
RISCV_PMC	0x0601 8000---0x0601 FFFF	32 KB

Region	Address	Size (Bytes)
RISCV_PLIC	0x1000_0000---0x13FF_FFFF	64 MB Only RISC-V core access, address region which RISC-V core accesses interrupts.
RISCV_CLIC	0x1400_0000---0x1400_FFFF	64 KB Only RISC-V core access, address region which RISC-V core accesses interrupts.
DRAM Space		
DRAM SPACE	0x4000 0000---0x4FFF FFFF	256 MB



3 System Boot

3.1 Boot ROM (BROM)

3.1.1 Overview

The BROM system is actually split up into two parts: FEL and Media Boot. The task of FEL is to write the external data to the local NVM, and the task of the Media Boot is to load from NVM to obtain and run an effective and legitimate BOOT0.

The BROM system includes the following features:

- Supports RISC-V boot process
- Support Mandatory Upgrade Process through USB or SD Card
- Support GPIO to select the kind of boot media to boot
- Support eFuse to select the kind of boot media to boot

3.1.2 Boot Medium Select

The BROM system supports the following boot media:

- SD Card
- eMMC
- SPI NOR Flash (Quad Mode)
- SPI NAND Flash (Quad Mode)

There are two ways to select the boot medium: GPIO Select and eFuse Select. The BROM will read the state of BOOT_MODE first, and then select the boot medium to boot according to the state of BOOT_MODE. The BOOT_MODE is the BROM_Config in the eFuse mapping.

The following table shows the BOOT_MODE setting:

Table 3-1 BOOT_MODE Setting

BOOT_MODE[0]	Boot Select type
0	GPIO boot select, indicates that the boot medium is decided by the value of the GPIO pin.
1	eFuse boot select, indicates that the boot medium is decided by the value of the eFuse type.

3.1.2.1 GPIO Boot Select

If the state of the BOOT_MODE is 0, the boot medium is decided by the value of the GPIO pin.

Table 3-2 GPIO Boot Select Configuration

Pin_Boot_Select[1:0]	Boot media
00	SPI0 NOR->SPI0 NAND
01	SMHC0->SPI0 NOR
10	SMHC0->SPI0 NAND
11	SMHC0-> SMHC2->SPI0 NOR->SPI NAND



NOTE

The status of the GPIO boot select pin can be read by the bit[12:11] of the system configuration module (register: 0x03000024).

When trying SPI0 NOR and SPI0 NAND, BROM try 4 wire mode first, then 1 wire mode

3.1.2.2 eFuse Boot Select

If the state of the BOOT_MODE is 1, select the eFuse Boot.

The eFuse_Boot_Select_Cfg is divided into 4 groups and each group is 3-bit. The following table shows the groups of eFuse_Boot_Select.

Table 3-3 Groups of eFuse_Boot_Select

eFuse_Boot_Select_Cfg [11:0]	Description
eFuse_Boot_Select[2:0]	eFuse_Boot_Select_1
eFuse_Boot_Select[5:3]	eFuse_Boot_Select_2
eFuse_Boot_Select[8:6]	eFuse_Boot_Select_3
eFuse_Boot_Select[11:9]	eFuse_Boot_Select_4

These four groups take effect with the following priority:

eFuse_Boot_Select_1->eFuse_Boot_Select_2->eFuse_Boot_Select_3->eFuse_Boot_Select_4

For example, eFuse_Boot_Select_2 will not take effect unless eFuse_Boot_Select_1 is set as 0b111(skip), eFuse_Boot_Select_3 will not take effect unless eFuse_Boot_Select_2 is set as 0b111(skip), etc. If all three groups are set to 0b111, no other groups can be used for boot select, BROM assume “try” is selected.

In the Try mode, the BROM follows the order below to select the boot medium:

SD Card -> SPI NOR -> SPI NAND -> eMMC

The following table shows the boot medium priority for the different values of eFuse_Boot_Select_n, where n = [4:1].

Table 3-4 eFuse Boot Select Setting

eFuse_Boot_Select_n	Boot Medium Priority
000	Select the boot medium in Try mode.
001	SMHC0->SPI0 NOR(1 wire)->SPI0 NOR(4 wire)

eFuse_Boot_Select_n	Boot Medium Priority
010	SMHC0->EMMC_USER->EMMC_BOOT
011	SMHC0->SPI0 NOR(4 wire)->SPI0 NOR(1 wire)
100	SMHC0->SPI0 NAND(4 wire)->SPI0 NAND(1 wire)
101	SMHC0->SPI0 NAND(1 wire)->SPI0 NAND(4 wire)
110	SMHC0->EMMC_BOOT->EMMC_USER
111	The next a group of the eFuse_Boot_Select,But when the n is equal to 4,it will be a way of Try.



NOTE

1. The SHMC2 is also called EMMC.
2. The try process is SMHC0->SPI0 NOR->SPI0 NAND-> EMMC2_USER->EMMC2_BOOT.
3. When trying SPI0 NOR and SPI0 NAND, BROM try 4 wire mode first, then 1 wire mode.

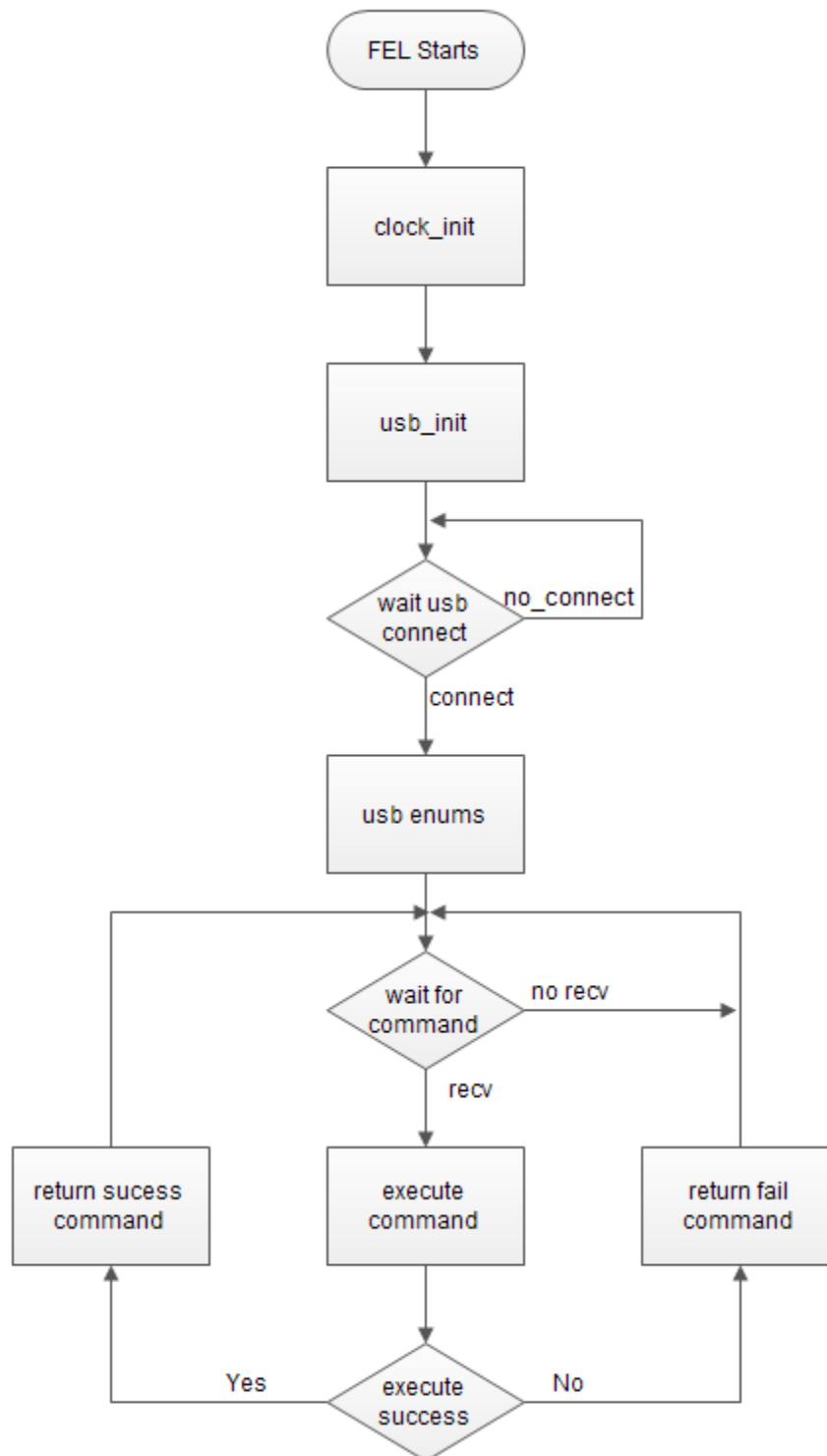
3.1.3 FEL Process

3.1.3.1 Mandatory Upgrade Process

When the system chooses to enter Mandatory Upgrade Process,then jump to the FEL Process.

The following figure shows the FEL process.

Figure 3-1 USB FEL Process Diagram



4 Power, Reset, and Clock Management

4.1 Clock Controller Unit (CCU)

4.1.1 Overview

The clock controller unit (CCU) controls the PLL configurations and most of the clock generation, division, distribution, synchronization, and gating in system. The input signals of the CCU include the external clock for the reference frequency (24 MHz). The outputs from the CCU are mostly clocks to other blocks in the system.

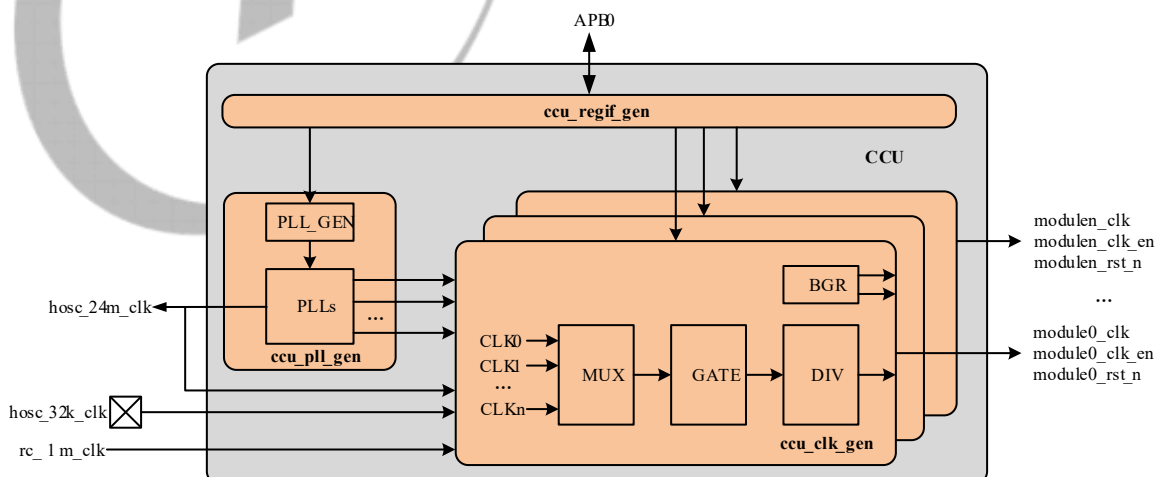
The CCU includes the following features:

- Clock sources: DCXO and RC 24MHz
- Bus source and divisions
- Clock output control
- Configurable modules clock
- Bus clock gating
- Bus software reset

4.1.2 Block Diagram

The following figure shows the functional block diagram of the CCU.

Figure 4-1 CCU Block Diagram



4.1.3 Functional Description

4.1.3.1 PLL Distribution

The following figure shows the PLL distribution for buses and modules.

Figure 4-2 PLL Distribution

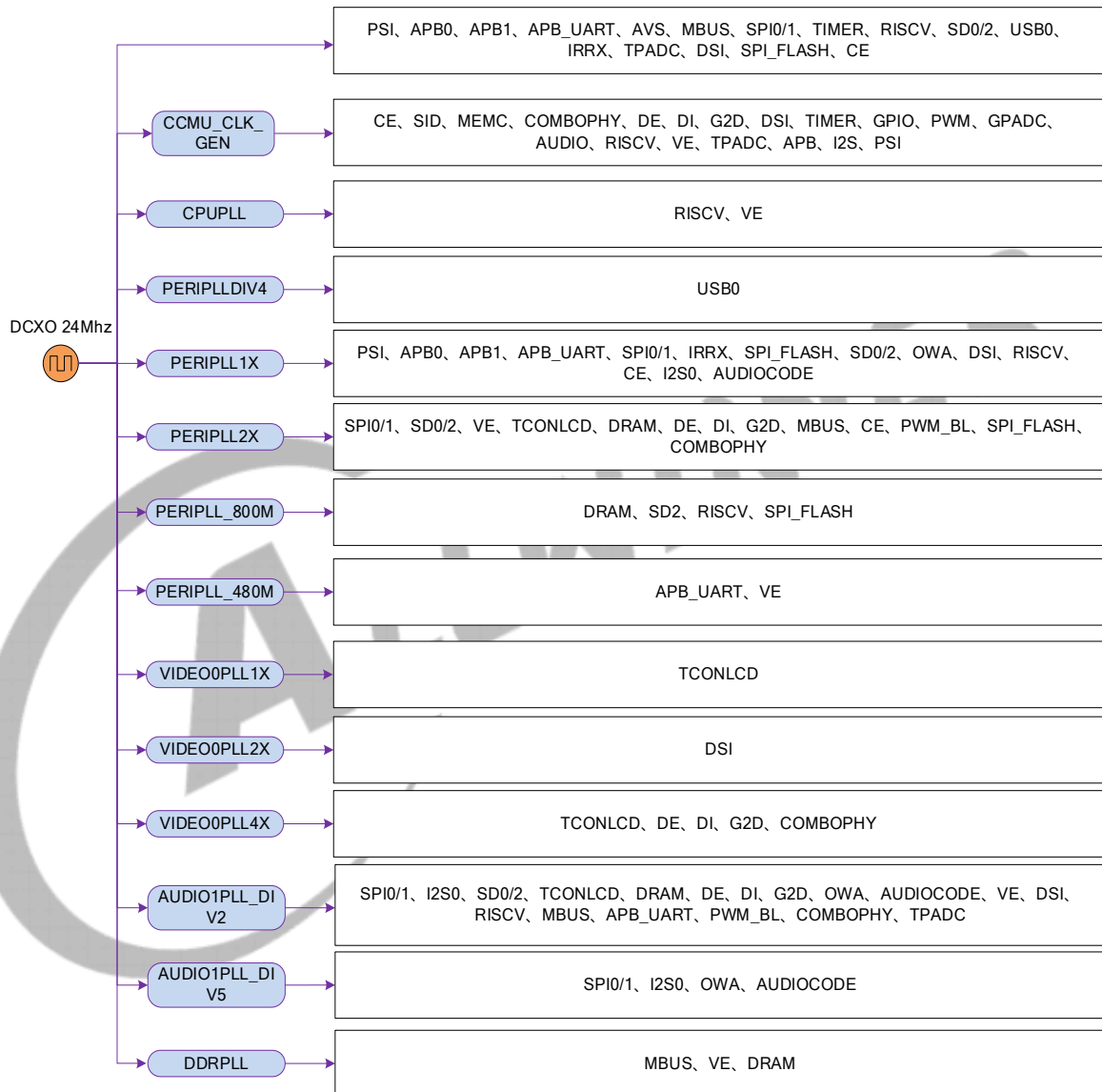


Table 4-1 PLL Typical Application

PLL Type	Application Module	Notes
PLL_CPU	RISC-V CPU, VE	Support DVFS
PLL_PERI(2X)	MBUS, SPI0/1, SMHC0/2, VE, TCON_LCD0, DRAM, DE, DI, G2D, CE, PWM_BL, SPI_FLASH, COMBOPHY	No support dynamic FM

PLL Type	Application Module	Notes
PLL_PERI(1X)	PSI, APB0, APB1, APB_UART, SPI0/1, IR_RX, SPI_FLASH, SMH0/2, OWA, DSI, RISC-V CPU, CE, I2S0, AUDIOCODEC	No support dynamic FM
PLL_PERI(800M)	DRAM, SMHC2, RISC-V CPU, SPI_FLASH	No support dynamic FM
PLL_PERI(480M)	APB_UART, VE	No support dynamic FM
PLL_PERI(DIV4)	USB0	No support DVFS
PLL_VIDEO0(4X)	TCON_LCD0, DE, DI, G2D, COMBOPHY	No support DVFS
PLL_VIDEO0(2X)	DSI	No support DVFS
PLL_VIDEO0(1X)	TCON_LCD0	No support DVFS
PLL_AUDIO1(DIV2)	SPI0/1, I2S0, SMHC0/2, TCON_LCD0, DRAM, DE, DI, G2D, OWA, AUDIOCODEC, VE, DSI, RISC-V CPU, MBUS, APB_UART, PWM_BL, COMBOPHY, TPADC	No support DVFS
PLL_AUDIO1(DIV5)	SPI0/1, OWA, I2S0, AUDIOCODEC	No support DVFS
PLL_DDR	MBUS, VE, DRAM	Support spread spectrum No support linear FM

4.1.3.2 PLL Specifications

The specifications of each PLL are shown in the following table.

Table 4-2 PLL Specifications

PLL	Stable Operating Frequency	Actual Operating Frequency	Default Frequency	Spread Spectrum	Linear FM	Rate Control	Pk-Pk	Lock Time
PLL_CPU	480 MHz to 2.016 GHz (24*N/P)	480 MHz to 1.5 GHz	480 MHz	No	Yes	Yes	< 50 ps	0.1 ms
PLL_AUDIO1	1.52 GHz to 3.1 GHz (24MHz*N/M)	3.072 GHz	Integer mode: 1/2x: 1.536 GHz 1/5x: 614.4 MHz Decimal mode: 1/2x: 1.1179648 GHz 1/5x: 471.8592 MHz	Yes	No	No	< 30 ps	0.1 ms
PLL_PERI	1.26 GHz to 2.52 GHz (24*N/M)	2.4 GHz	1/2x: 1.2 GHz 1/3x: 800 MHz 1/5x: 480 MHz	Yes	No	No	< 125 ps	0.1 ms
PLL_Video0(4X)	1.26 GHz to 2.52 GHz (24*N/M)	1188 MHz	1x: 297 MHz 2x:594 MHz 4x:1188 MHz	Yes	No	No	< 70 ps	0.1 ms
PLL_DDR	480 MHz to 2.016 GHz (24*N/M1)	480 MHz to 2.016 GHz	480 MHz	Yes	No	No	< 50 ps	0.1 ms



4.1.4 Programming Guidelines

4.1.4.1 Ordinary PLL Configuration

Enable the PLL

- Step 1** Write 1 to LDO_EN of [PLL_CTRL_REG1](#).
- Step 2** Configure the N, M1, M0, and, P of PLL_XXX_CTRL_REG.
- Step 3** Write 1 to the PLL_EN and LDO_EN of PLL_XXX_CTRL_REG. Write 0 to the PLL_OUTPUT_GATE of PLL_XXX_CTRL_REG.
- Step 4** Write 1 to the LOCK_ENABLE of PLL_XXX_CTRL_REG.
- Step 5** Read LOCK_STATUS every 3 us for total three times. The PLL is locked if the value of every read is 1.
- Step 6** Delay 20us.
- Step 7** Write 1 to the PLL_OUTPUT_GATE of PLL_XXX_CTRL_REG.

Disable the PLL

- Step 1** Ensure that the clock using the PLL to be disabled is switched to other clock sources, or the clock is disabled.
- Step 2** Write 0 to the PLL_OUTPUT_GATE of PLL_XXX_CTRL_REG.
- Step 3** Write 0 to the LOCK_ENABLE of PLL_XXX_CTRL_REG.
- Step 4** Write 0 to the PLL_EN of PLL_XXX_CTRL_REG.
- Step 5** Write 0 to the LDO_EN of the [PLL_CTRL_REG1](#) register.



NOTE

- Frequent PLL enable/disable is not recommended, as the PLL enable/disable will arise the mutual interference of PLLs and system instability. To disable PLL, it is recommended to write 0 to the PLL_OUTPUT_GATE.
- The LDO_EN of [PLL_CTRL_REG1](#) register controls the enable of all system PLL's LDO. The LDO_EN needs to be configured as 1, unless system is power-off or all PLLs are disabled.

Configure the PLL Frequency

- Step 1** Write 0 to the PLL_OUTPUT_GATE of PLL_XXX_CTRL_REG.
- Step 2** Modify the values of N and M factors. (It is not recommended to configure M1 factor.)
- Step 3** Write 0 and then write 1 to the LOCK_ENABLE of PLL_XXX_CTRL_REG.

- Step 4** Read LOCK_STATUS every 3 us for total three times. The PLL is locked if the value of all three reads is 1.
- Step 5** Delay 20us.
- Step 6** Write 1 to the PLL_OUTPUT_GATE of PLL_XXX_CTRL_REG.

4.1.4.2 Linear FM PLL Configuration

Enable the PLL

- Step 1** Configure the N, M0, M1, and P factors of the PLL_CPU_CTRL_REG register.
- Step 2** Write 1 to the PLL_PLL_EN and PLL_LDO_EN of the PLL_CPU_CTRL_REG register.
- Step 3** Write 1 to the LOCK_ENABLE of the PLL_CPU_CTRL_REG register.
- Step 4** Write 1 to the PLL_UPDATE of the PLL_CPU_CTRL_REG register.
- Step 5** Wait for the PLL_UPDATE of the PLL_CPU_CTRL_REG register to change to 0.
- Step 6** Read LOCK_STATUS every 3 us for total three times. The PLL is locked if the value of all three reads is 1.
- Step 7** Delay 20 us

Disable the PLL

- Step 1** Write 0 to the LOCK_ENABLE of the PLL_CPU_CTRL_REG register.
- Step 2** Write 0 to the PLL_PLL_EN and PLL_LDO_EN of the PLL_CPU_CTRL_REG register.

Configure Linear Frequency Modulation (for dynamic configuration)

- Step 1** Configure the PLL_SSC_STEP bit of the PLL_CPU_SSC_REG register to select required frequency modulation slope.
- Step 2** Configure the PLL_SSC bit of the PLL_CPU_SSC_REG register to set the SSC amplitude.
- Step 3** Write 1 to the PLL_SSC_MODE bit of the PLL_CPU_SSC_REG register to enable linear frequency modulation.
- Step 4** Configure the N factor of the PLL_CPU_CTRL_REG register.
- Step 5** Write 1 to the PLL_UPDATE bit of the PLL_CPU_CTRL_REG register to update PLL configuration parameters.
- Step 6** Wait for the value of the PLL_UPDATE bit to change to 0.
- Step 7** Write 0 to the PLL_SSC_MODE bit of the PLL_CPU_SSC_REG register to disable linear frequency modulation.
- Step 8** Write 1 to the PLL_UPDATE bit of the PLL_CPU_CTRL_REG register to update PLL configuration parameters.

Step 9 Wait for the value of the PLL_UPDATE bit to change to 0.

Configure Ordinary Frequency Modulation (for static configuration)



NOTE

It is not suggested to enable or disable the PLLs during usage.

Step 1 Make sure the PLL is enabled. If not, refer to Section Enable the PLL to enable the PLL.

Step 2 Configure the N and M factors of the [PLL_CPU_CTRL_REG](#) register.

Step 3 Write 0 and then write 1 to the LOCK_ENABLE of the [PLL_CPU_CTRL_REG](#) register.

Step 4 Write 1 to the PLL_UPDATE of the [PLL_CPU_CTRL_REG](#) register.

Step 5 Read LOCK_STATUS every 3 us for total three times. The PLL is locked if the value of all three reads is 1.

Step 6 Delay 20 us and then PLL is available.

4.1.4.3 PLL_AUDIO Configuration

The frequency configuration formula of PLL_AUDIO:

$$\text{PLL\_AUDIO} = 24 \text{ MHz} * N / M0 / M1 / P$$

PLL_AUDIO includes PLL_AUDIO0 and PLL_AUDIO1.

PLL_AUDIO does not support dynamic adjustment because changing any parameter of N, M0, M1, and P will affect the normal work of PLL, and the PLL will need to be relocked.

Generally, PLL_AUDIO only needs two frequency points: 24.576*4 MHz or 22.5792*4 MHz. For these two frequencies, there are usually special recommended matching factors. To implement the desired frequency point of PLL_AUDIO, you need to use the decimal frequency-division function, so follow the steps below:

Step 1 Configure the N, M0, M1 and P factors.

Step 2 Configure PLL_AUDIOx Control Register[PLL_SDM_ENABLE] to 1.

Step 3 Configure PLL_AUDIOx Pattern0 Control Register to enable the digital spread spectrum.

Step 4 Write PLL_AUDIOx Control Register [Lock Enable] to 0 and then to 1.

Step 5 Wait for PLL_AUDIOx Control Register[Lock] to 1.



NOTE

When the P factor of PLL_AUDIO is an odd number, the clock output is an unequal-duty-cycle signal.

The recommended values for configuration factors of PLL_AUDIO1 are as follows.

Table 4-3 Recommended Values for Configuration Factors of PLL_AUDIO1

Mode	Clock Source (MHz)	Frequency-doubling N	VCO (MHz)	Post Frequency-Division	PLL Output (MHz)	Divisor	Actual Operating Frequency (MHz)	Description
Integer divider	24	128	3072	2	1536	4	384	Provide clock source for peripherals
						8	192	
						16	96	
								
				5	614.4	25	24.576	For audio-related modules
Decimal divider	24	98.304	2359.296	2	1179.648	2	589.824	Provides clock source for peripheral devices
						3	393.216	
						6	196.608	
						12	98.304	
						48	24.576	For audio-related modules
				5	471.8592			

4.1.4.4 Implement Spread Spectrum

The spread spectrum technology is to convert a narrowband signal into a wideband signal. It helps to reduce the effect of electromagnetic interference (EMI) associated with the fundamental frequency of the signal.

For the general PLL frequency, the calculation formula is as follows:

$$f = \frac{N+1+X}{P \cdot (M0+1) \cdot (M1+1)} \cdot 24MHz, 0 < X < 1$$

Where,

P is the frequency division factor of module or PLL;

M0 is the post-frequency division factor of PLL;

M1 is the pre-frequency division factor of PLL;

N is the frequency doubling factor of PLL;

X is the amplitude coefficient of the spread spectrum.

The parameters N, P, M1, and M0 are for the frequency division.

When M1 = 0, M0 = 0, and P = 1 (no frequency division), the calculation formula of PLL frequency can be simplified as follows:

$$f = (N + 1 + X) \cdot 24\text{MHz}, 0 < X < 1$$

$$[f_1, f_2] = (N + 1 + [X_1, X_2]) \cdot 24\text{MHz}$$

$$\text{WAVE\_BOT} = 2^{17} \cdot X_1$$

$$\text{WAVE\_STEP} = 2^{17} \cdot (X_2 - X_1) / (24\text{MHz} / \text{FREQ}) \cdot 2$$

Where, WAVE_BOT and WAVE_STEP are bits of the PLL pattern control register, and FREQ is the frequency of the spread spectrum.



NOTE

Different PLLs have different calculate formulas, refer to the CTRL register of the corresponding PLL in Section 4.1.6 Register Description.

Follow the steps below to implement the spread spectrum:

Step 1 Configure the control register of the corresponding PLL

- Calculate the factor N and decimal value X according to the PLL frequency and PLL frequency formula. Refer to the control register of the corresponding PLL (named PLL_XXX_CTRL_REG, where xxx is the module name) in Section 4.1.6 Register Description for the corresponding PLL frequency formula.
- Write M0, M1, N, and PLL frequency to the PLL_XXX_CTRL_REG.
- Configure the PLL_SDM_EN bit (bit [24]) of the PLL_XXX_CTRL_REG to 1 to enable the spread spectrum function.

Step 2 Configure the pattern control register of the corresponding PLL

- Calculate the WAVE_BOT and WAVE_STEP of the pattern control register according to decimal value X and spread spectrum frequency.
- Write WAVE_BOT and WAVE_STEP to the PLL_XXX_PAT0_CTRL_REG register.
- Configure the FREQ of PLL_XXX_PAT0_CTRL_REG register.
- Configure the spread spectrum mode (SPR_FREQ_MODE) to 2 or 3.
- If the PLL_INPUT_DIV2 of the PLL_XXX_CTRL_REG is 1, configure the spread spectrum clock source select bit (SDM_CLK_SEL) of the PLL pattern control register to 1. Otherwise, configure SDM_CLK_SEL to the default value 0.
- Configure the SIG_DELT_PAT_EN bit (bit [31]) of this register to 1.

Step 3 Delay 20 us.

4.1.4.5 Glitch-free Switch Configuration

To use glitch-free switch, follow the rules below:

- Both the original clock source and the target clock source are enabled and output stably.
- In the frequency modulation process, ensure that the output frequency does not exceed the target frequency (during frequency increasing) or original frequency (during frequency reducing).
- The switching interval needs to be longer than two clock periods of the slowest clock source.
- To switch from a lower frequency to a higher frequency: set the frequency division factor first, and then configure the clock source.
- To switch from a higher frequency to a lower frequency: configure the clock source first, and then set the frequency division factor.

4.1.4.6 Bus Clock Configuration

The Bus clock and CPU clock must use glitch-free switch. See section 4.1.4.5 for configuration rules. Besides, ensure that there will be no frequency exceeding the highest frequency clock of the bus when switching bus clock.

The bus frequency for each bus is as follows:

- **RISCV_AXI**: It is suggested to be configured as the RISC-V CPU clock frequency divided by 1
- **PSI (AHB)**: 200 MHz at max.
- **APB0**: 100 MHz at max.
- **APB1**: 100 MHz at max.
- **APB_UART**: 100 MHz at max
- **MBUS**: 400 MHz at max

4.1.4.7 Module Clock Configuration

For the bus gating reset register of a module, the reset bit is de-asserted first, and then the gating clock is passed to avoid potential problems caused by the asynchronous release of the reset signal.

For all module clocks except the DRAM clocks, configure the clock source and frequency division factor first, and then release the clock gating (that is, set to 1). DRAM clocks use glitch-free switch, see Section 4.1.4.5 for configuration rules.

4.1.5 Register List

Module Name	Base Address

Module Name	Base Address
CCU	0x02001000

Register Name	Offset	Description
PLL_CPU_CTRL_REG	0x0000	PLL_CPU Control Register
PLL_DDR_CTRL_REG	0x0010	PLL_DDR Control Register
PLL_PERI_CTRL_REG	0x0020	PLL_PERI Control Register
PLL_VIDEO0_CTRL_REG	0x0040	PLL_VIDEO0 Control Register
PLL_AUDIO1_CTRL_REG	0x0080	PLL_AUDIO1 Control Register
PLL_CPU_PAT0_CTRL_REG	0x0100	PLL_CPU Pattern0 Control Register
PLL_CPU_PAT1_CTRL_REG	0x0104	PLL_CPU Pattern1 Control Register
PLL_DDR_PAT0_CTRL_REG	0x0110	PLL_DDR Pattern0 Control Register
PLL_DDR_PAT1_CTRL_REG	0x0114	PLL_DDR Pattern1 Control Register
PLL_PERI_PAT0_CTRL_REG	0x0120	PLL_PERI Pattern0 Control Register
PLL_PERI_PAT1_CTRL_REG	0x0124	PLL_PERI Pattern1 Control Register
PLL_VIDEO0_PAT0_CTRL_REG	0x0140	PLL_VIDEO0 Pattern0 Control Register
PLL_VIDEO0_PAT1_CTRL_REG	0x0144	PLL_VIDEO0 Pattern1 Control Register
PLL_AUDIO1_PAT0_CTRL_REG	0x0180	PLL_AUDIO1 Pattern0 Control Register
PLL_AUDIO1_PAT1_CTRL_REG	0x0184	PLL_AUDIO1 Pattern1 Control Register
PLL_CPU_SSC_REG	0x0200	PLL_CPU SSC Register
PLL_CPU_BIAS_REG	0x0300	PLL_CPU Bias Register
PLL_DDR_BIAS_REG	0x0310	PLL_DDR Bias Register
PLL_PERI_BIAS_REG	0x0320	PLL_PERI Bias Register
PLL_VIDEO0_BIAS_REG	0x0340	PLL_VIDEO0 Bias Register
PLL_AUDIO1_BIAS_REG	0x0380	PLL_AUDIO1 Bias Register
PLL_CPU_TUN1_REG	0x0404	PLL_CPU Tuning1 Register
PSI_CLK_REG	0x0510	PSI Clock Register
APB0_CLK_REG	0x0520	APB0 Clock Register
APB1_CLK_REG	0x0524	APB1 Clock Register
APB_UART_CLK_REG	0x0528	APB_UART Clock Register
MBUS_BGR_REG	0x0540	MBUS Bus Gating Reset Register
MBUS_CLK_REG	0x0544	MBUS Clock Register
DE_CLK_REG	0x0600	DE Clock Register
DE_BGR_REG	0x060C	DE Bus Gating Reset Register
DI_CLK_REG	0x0620	DI Clock Register
DI_BGR_REG	0x062C	DI Bus Gating Reset Register
G2D_CLK_REG	0x0630	G2D Clock Register
G2D_BGR_REG	0x063C	G2D Bus Gating Reset Register
CE_CLK_REG	0x0680	CE Clock Register
CE_BGR_REG	0x068C	CE Bus Gating Reset Register
VE_CLK_REG	0x0690	VE Clock Register
VE_BGR_REG	0x069C	VE Bus Gating Reset Register

Register Name	Offset	Description
DMA_BGR_REG	0x070C	DMA Bus Gating Reset Register
HSTIMER_BGR_REG	0x073C	HSTIMER Bus Gating Reset Register
AVS_CLK_REG	0x0740	AVS Clock Register
DBGSYS_BGR_REG	0x078C	DBGSYS Bus Gating Reset Register
PWM_BL_CLK_REG	0x0790	PWM_BL Clock Register
PWM_BL_BGR_REG	0x0794	PWM_BL Bus Gating Reset Register
PWM_BGR_REG	0x07AC	PWM Bus Gating Reset Register
DRAM_CLK_REG	0x0800	DRAM Clock Register
MBUS_MAT_CLK_GATING_REG	0x0804	MBUS Master Clock Gating Register
DRAM_BGR_REG	0x080C	DRAM Bus Gating Reset Register
SMHC0_CLK_REG	0x0830	SMHC0 Clock Register
SMHC2_CLK_REG	0x0838	SMHC2 Clock Register
SMHC_BGR_REG	0x084C	SMHC Bus Gating Reset Register
UART_BGR_REG	0x090C	UART Bus Gating Reset Register
TWI_BGR_REG	0x091C	TWI Bus Gating Reset Register
SPI0_CLK_REG	0x0940	SPI0 Clock Register
SPI1_CLK_REG	0x0944	SPI1 Clock Register
SPI_BGR_REG	0x096C	SPI Bus Gating Reset Register
SPIF_CLK_REG	0x0980	SPIF Clock Register
SPIF_BGR_REG	0x0984	SPIF Bus Gating Reset Register
IR_RX_CLK_REG	0x09C0	IR_RX Clock Register
IR_RX_BGR_REG	0x09CC	IR_RX Bus Gating Reset Register
GPADC_BGR_REG	0x09EC	GPADC Bus Gating Reset Register
THS_BGR_REG	0x09FC	THS Bus Gating Reset Register
I2S0_CLK_REG	0x0A10	I2S0 Clock Register
I2S_BGR_REG	0x0A20	I2S Bus Gating Reset Register
OWA_TX_CLK_REG	0x0A24	OWA_TX Clock Register
OWA_RX_CLK_REG	0x0A28	OWA_RX Clock Register
OWA_BGR_REG	0x0A2C	OWA Bus Gating Reset Register
AUDIO_CODEC_DAC_CLK_REG	0x0A50	AUDIO_CODEC_DAC Clock Register
AUDIO_CODEC_ADC_CLK_REG	0x0A54	AUDIO_CODEC_ADC Clock Register
AUDIO_CODEC_BGR_REG	0x0A5C	AUDIO_CODEC Bus Gating Reset Register
USB0_CLK_REG	0x0A70	USB0 Clock Register
USB_BGR_REG	0x0A8C	USB Bus Gating Reset Register
COMBOPHY0_CLK_REG	0x0AA0	COMBOPHY0 Clock Register
COMBOPHY0_BGR_REG	0x0AA4	COMBOPHY0 Bus Gating Reset Register
DPSS_TOP_BGR_REG	0x0ABC	DPSS_TOP Bus Gating Reset Register
DSI_CLK_REG	0x0B24	DSI Clock Register
DSI_BGR_REG	0x0B4C	DSI Bus Gating Reset Register
TCONLCD_CLK_REG	0x0B60	TCONLCD Clock Register
TCONLCD_BGR_REG	0x0B7C	TCONLCD Bus Gating Reset Register
LVDS_BGR_REG	0x0BAC	LVDS Bus Gating Reset Register

Register Name	Offset	Description
TPADC_CLK_REG	0x0C50	TPADC Clock Register
TPADC_BGR_REG	0x0C5C	TPADC Bus Gating Reset Register
RISCV_CLK_REG	0x0D00	RISCV Clock Register
RISCV_CFG_BGR_REG	0x0D0C	RISCV_CFG Bus Gating Reset Register
CCU_SEC_SWITCH_REG	0x0F00	CCU Security Switch Register
PLL_LOCK_DBG_CTRL_REG	0x0F04	PLL Lock Debug Control Register
CCU_FAN_GATE_REG	0x0F30	CCU Fanout Clock Gate Register
CLK27M_FAN_REG	0x0F34	CLK27M FANOUT Register
PCLK_FAN_REG	0x0F38	PCLK FANOUT Register
CCU_FAN_REG	0x0F3C	CCU FANOUT Register
PLL_CTRL_REG0	0x0F40	PLL Control Register0
PLL_CTRL_REG1	0x0F44	PLL Control Register1
HOSC_32K_SEL_REG	0x0F48	HOSC And 32k Select Register

4.1.6 Register Description

4.1.6.1 0x0000 PLL_CPU Control Register (Default Value: 0x4000_1400)

Offset: 0x0000			Register Name: PLL_CPU_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	PLL_PLL_EN PLL Enable 0: Disable 1: Enable The PLL_CPU= InputFreq*N/P/(M0*M1) The PLL_CPU's VCO frequency range (InputFreq*N/P) is from 480 MHz to 3.0 GHz. Lower frequency can be obtained by dividing M0 and M1. When the crystal oscillator is 24 MHz, the default frequency of PLL_CPU is 480 MHz.
30	R/W	0x1	PLL_LDO_EN LDO enable 0: Disable 1: Enable
29	R/W	0x0	LOCK_ENABLE Lock Enable 0: Disable 1: Enable

Offset: 0x0000			Register Name: PLL_CPU_CTRL_REG
Bit	Read/Write	Default/Hex	Description
28	R	0x0	LOCK PLL Lock info 0: Unlocked 1: Locked (It indicates that the PLL has been stable.)
27	R/W	0x0	PLL_OUTPUT_GATE PLL Output Gating Enable 0: Disable 1: Enable The bit is used to control the output enable of the PLL.
26	R/WAC	0x0	PLL_UPDATE Write 1 update the cfg to PLL, auto clear.
25:22	/	/	/
21:20	R/W	0x0	PLL_M0 PLL_M0 $M0 = PLL\_M0 + 1$ PLL_FACTOR_M0 is from 0 to 3.
19	/	/	/
18:16	R/W	0x0	PLL_P PLL PREDIV P $P = PLL\_P + 1$ PLL_P is from 0 to 7
15:8	R/W	0x14	PLL_N PLL N $N = PLL\_N$ In application, PLL_N shall be more than or equal to 12.
7:6	R/W	0x0	PLL_UNLOCK_MDSEL PLL Unlock Level 00: 21-29 Clock Cycles 01: 22-28 Clock Cycles 1X: 20-30 Clock Cycles
5	R/W	0x0	PLL_LOCK_MDSEL PLL Lock Level 0: 24-26 Clock Cycles 1: 23-27 Clock Cycles
4	/	/	/
3:0	R/W	0x0	PLL_M1 PLL_M1 $M1 = PLL\_M1 + 1$ PLL_M1 is from 0 to 15.

4.1.6.2 0x0010 PLL_DDR Control Register (Default Value: 0x4800_2301)

Offset: 0x0010			Register Name: PLL_DDR_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	PLL_EN PLL Enable 0: Disable 1: Enable The PLL_DDR= InputFreq*N/M1/M0 When the crystal oscillator is 24 MHz, the default frequency of PLL_DDR is 432 MHz.
30	R/W	0x1	PLL_LDO_EN LDO enable 0: Disable 1: Enable
29	R/W	0x0	LOCK_ENABLE Lock Enable 0: Disable 1: Enable
28	R	0x0	LOCK PLL Lock Info 0: Unlocked 1: Locked (It indicates that the PLL has been stable.)
27	R/W	0x1	PLL_OUTPUT_GATE PLL Output Gating Enable 0: Disable 1: Enable The bit is used to control the output enable of the PLL.
26:25	/	/	/
24	R/W	0x0	PLL_SDM_EN PLL SDM Enable 0: Disable 1: Enable
23:16	/	/	/
15:8	R/W	0x23	PLL_N PLL N N= PLL_N +1 PLL_N is from 0 to 254. In application, PLL_N shall be more than or equal to 11.

Offset: 0x0010			Register Name: PLL_DDR_CTRL_REG
Bit	Read/Write	Default/Hex	Description
7:6	R/W	0x0	PLL_UNLOCK_MDSEL PLL Unlock Level 00: 21-29 Clock Cycles 01: 22-28 Clock Cycles 1X: 20-30 Clock Cycles
5	R/W	0x0	PLL_LOCK_MDSEL PLL Lock Level 0: 24-26 Clock Cycles 1: 23-27 Clock Cycles
4:2	/	/	/
1	R/W	0x0	PLL_INPUT_DIV2 PLL Input Div M1 $M1 = PLL\_INPUT\_DIV2 + 1$ PLL_INPUT_DIV2 is from 0 to 1.
0	R/W	0x1	Reserved

4.1.6.3 0x0020 PLL_PERI Control Register (Default Value: 0x4021_6310)

Offset: 0x0020			Register Name: PLL_PERI_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	PLL_EN PLL Enable 0: Disable 1: Enable The $PLL\_PERI(2X) = 24MHz * N/M/P0$ The $PLL\_PERI(1X) = 24MHz * N/M/P0/2$ The $PLL\_PERI(800M) = 24MHz * N/M/P1$ The $PLL\_PERI(480M) = 24MHz * N/M/P2$ When the crystal oscillator is 24 MHz, the default frequency of PLL_PERI(2X) is 1.2 GHz, the default frequency of PLL_PERI(1X) is 600 MHz, and the default frequency of PLL_PERI(800M) is 800 MHz. The output clock of PERI1PLL2X is fixed to 1.2 GHz and not suggested to change the parameter.
30	R/W	0x1	PLL_LDO_EN LDO enable 0: Disable 1: Enable

Offset: 0x0020			Register Name: PLL_PERI_CTRL_REG
Bit	Read/Write	Default/Hex	Description
29	R/W	0x0	LOCK_ENABLE Lock Enable 0: Disable 1: Enable
28	R	0x0	LOCK PLL Lock Info 0: Unlocked 1: Locked (It indicates that the PLL has been stable.)
27	R/W	0x0	PLL_OUTPUT_GATE PLL Output Gating Enable 0: Disable 1: Enable The bit is used to control the output enable of the PLL.
26:25	/	/	/
24	R/W	0x0	PLL_SDM_EN PLL SDM Enable 0: Disable 1: Enable Enable spread spectrum and decimal division.
23	/	/	/
22:20	R/W	0x2	PLL_P1 PLL Output Div P1 $P1 = PLL\_OUTPUT\_DIV\_P1 + 1$ PLL_OUTPUT_DIV_P1 is from 0 to 7.
19	/	/	/
18:16	R/W	0x1	PLL_P0 PLL Output Div P0 $P0 = PLL\_OUTPUT\_DIV\_P0 + 1$ PLL_OUTPUT_DIV_P0 is from 0 to 7.
15:8	R/W	0x63	PLL_N PLL N $N = PLL\_N + 1$ PLL_N is from 0 to 254. In application, PLL_N shall be more than or equal to 11.
7:6	R/W	0x0	PLL_UNLOCK_MDSEL PLL Unlock Level 00: 21-29 Clock Cycles 01: 22-28 Clock Cycles 1X: 20-30 Clock Cycles

Offset: 0x0020			Register Name: PLL_PERI_CTRL_REG
Bit	Read/Write	Default/Hex	Description
5	R/W	0x0	PLL_LOCK_MDSEL PLL Lock Level 0: 24-26 Clock Cycles 1: 23-27 Clock Cycles
4:2	/	/	/
1	R/W	0x0	PLL_INPUT_DIV2 PLL Input Div M $M = \text{PLL\_INPUT\_DIV2} + 1$ PLL_INPUT_DIV2 is from 0 to 1.
0	/	/	/

4.1.6.4 0x0040 PLL_VIDEO0 Control Register (Default Value: 0x4000_6203)

Offset: 0x0040			Register Name: PLL_VIDEO0_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	PLL_EN PLL Enable 0: Disable 1: Enable For application, $\text{PLL\_VIDEO0}(4X) = \text{InputFreq} * N/M$ $\text{PLL\_VIDEO0}(2X) = \text{InputFreq} * N/M/2$ $\text{PLL\_VIDEO0}(1X) = \text{InputFreq} * N/M/4$ When the crystal oscillator is 24 MHz, The default value of PLL_VIDEO0(4X) is 1188MHz=24*99/2.
30	R/W	0x1	PLL_LDO_EN LDO enable 0: Disable 1: Enable
29	R/W	0x0	LOCK_ENABLE Lock Enable 0: Disable 1: Enable
28	R	0x0	LOCK PLL Lock Info 0: Unlocked 1: Locked (It indicates that the PLL has been stable.)

Offset: 0x0040			Register Name: PLL_VIDEO0_CTRL_REG
Bit	Read/Write	Default/Hex	Description
27	R/W	0x0	PLL_OUTPUT_GATE PLL Output Gating Enable 0: Disable 1: Enable The bit is used to control the output enable of the PLL.
26:25	/	/	/
24	R/W	0x0	PLL_SDM_EN PLL SDM Enable 0: Disable 1: Enable.
23:16	/	/	/
15:8	R/W	0x62	PLL_N PLL N N= PLL_N +1 PLL_N is from 0 to 254. In application, PLL_N shall be more than or equal to 11.
7:6	R/W	0x0	PLL_UNLOCK_MDSEL PLL Unlock Level 00: 21-29 Clock Cycles 01: 22-28 Clock Cycles 1X: 20-30 Clock Cycles
5	R/W	0x0	PLL_LOCK_MDSEL PLL Lock Level 0: 24-26 Clock Cycles 1: 23-27 Clock Cycles
4:2	/	/	/
1	R/W	0x1	PLL_INPUT_DIV2 PLL Input Div M M1=PLL_INPUT_DIV_M + 1 PLL_INPUT_DIV_M is from 0 to 1.
0	R/W	0x1	Reserved

4.1.6.5 0x0080 PLL_AUDIO1 Control Register (Default Value: 0x4041_7F00)

Offset: 0x0080			Register Name: PLL_AUDIO1_CTRL_REG
Bit	Read/Write	Default/Hex	Description

Offset: 0x0080			Register Name: PLL_AUDIO1_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	PLL_EN PLL Enable 0: Disable 1: Enable The PLL_AUDIO1 = 24MHz*N/M The PLL_AUDIO1(DIV2) = 24MHz*N/M /P0 The PLL_AUDIO1(DIV5) = 24MHz*N/M /P1 The working frequency range of 24 MHz/M*N is from 180 MHz to 3.5 GHz. The default frequency of PLL_AUDIO1 is 3072 MHz. The default frequency of PLL_AUDIO1(DIV2) is 1536 MHz. The default frequency of PLL_AUDIO1(DIV5) is 614.4 MHz (24.576 MHz*25).
30	R/W	0x1	PLL_LDO_EN LDO enable 0: Disable 1: Enable
29	R/W	0x0	LOCK_ENABLE Lock Enable 0: Disable 1: Enable
28	R	0x0	LOCK PLL Lock Info 0: Unlocked 1: Locked (It indicates that the PLL has been stable.) The bit is only valid when the bit29 is set to 1.
27	R/W	0x0	PLL_OUTPUT_GATE PLL Output Gating Enable 0: Disable 1: Enable The bit is used to control the output enable of the PLL.
26:25	/	/	/
24	R/W	0x0	PLL_SDM_EN PLL SDM Enable 0: Disable 1: Enable Enable spread spectrum and decimal division.
23	/	/	/

Offset: 0x0080			Register Name: PLL_AUDIO1_CTRL_REG
Bit	Read/Write	Default/Hex	Description
22:20	R/W	0x4	PLL_P1 PLL Output Div P1 $P1 = PLL\_OUTPUT\_DIV\_P1 + 1$ PLL_OUTPUT_DIV_P1 is from 0 to 7.
19	/	/	/
18:16	R/W	0x1	PLL_P0 PLL Output Div P0 $P0 = PLL\_OUTPUT\_DIV\_P0 + 1$ PLL_OUTPUT_DIV_P0 is from 0 to 7.
15:8	R/W	0x7F	PLL_FACTOR_N PLL Factor N $N = PLL\_FACTOR\_N + 1$ PLL_FACTOR_N is from 0 to 254. In application, PLL_FACTOR_N shall be more than or equal to 11.
7:6	R/W	0x0	PLL_UNLOCK_MDSEL PLL Unlock Level 00: 21-29 Clock Cycles 01: 22-28 Clock Cycles 1X: 20-30 Clock Cycles
5	R/W	0x0	PLL_LOCK_MDSEL PLL Lock Level 0: 24-26 Clock Cycles 1: 23-27 Clock Cycles
4:2	/	/	/
1	R/W	0x0	PLL_INPUT_DIV2 PLL Input Div M $M = PLL\_INPUT\_DIV\_M + 1$ PLL_INPUT_DIV_M is from 0 to 1.
0	/	/	/

4.1.6.6 0x0100 PLL_CPU Pattern0 Control Register (Default Value: 0x0000_0000)

Offset: 0x0100			Register Name: PLL_CPU_PAT0_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SIG_DELT_PAT_EN Sigma Delta Pattern Enable.

Offset: 0x0100			Register Name: PLL_CPU_PAT0_CTRL_REG
Bit	Read/Write	Default/Hex	Description
30:29	R/W	0x0	SPR_FREQ_MODE Spread Frequency Mode 00: SDM 01: MASH-1(8 bit) 10: MASH-2(8 bit) 11: MASH-3(8 bit)
28:17	R/W	0x0	WAVE_STEP Wave Step
16:0	R/W	0x0	WAVE_BOT Wave Bottom

4.1.6.7 0x0104 PLL_CPU Pattern1 Control Register (Default Value: 0x0000_0000)

Offset: 0x0104			Register Name: PLL_CPU_PAT1_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:22	R/W	0x0	SDM_CYCLE SDM Cycle $SDM\_CYCLE = SDM\_FRE / CLK\_SDM / 2$
21	/	/	/
20	R/W	0x0	SDM_DIRECTION SDM direction 0: UP 1: DOWM
19	/	/	/
18	R/W	0x0	DITHER_EN Dither Enable.
17	R/W	0x0	FRAC_EN Fraction Enable.
16:0	R/W	0x0	FRAC_IN Fraction In.

4.1.6.8 0x0110 PLL_DDR Pattern0 Control Register (Default Value: 0x0000_0000)

Offset: 0x0110			Register Name: PLL_DDR_PAT0_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SIG_DELT_PAT_EN Sigma Delta Pattern Enable.

Offset: 0x0110			Register Name: PLL_DDR_PAT0_CTRL_REG
Bit	Read/Write	Default/Hex	Description
30:29	R/W	0x0	SPR_FREQ_MODE Spread Frequency Mode 00: DC=0 01: DC=1 10: Triangular(1-bit) 11: Triangular(n-bit)
28:20	R/W	0x0	WAVE_STEP Wave Step.
19	R/W	0x0	SDM_CLK_SEL SDM Clock Select 0: 24MHz 1: 12MHz When the PLL_INPUT_DIV2 is 1, the bit needs to be set to 1.
18:17	R/W	0x0	FREQ Frequency 00: 31.5KHz 01: 32KHz 10: 32.5KHz 11: 33KHz
16:0	R/W	0x0	WAVE_BOT Wave Bottom.

4.1.6.9 0x0114 PLL_DDR Pattern1 Control Register (Default Value: 0x0000_0000)

Offset: 0x0114			Register Name: PLL_DDR_PAT1_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:25	/	/	/
24	R/W	0x0	DITHER_EN Dither Enable.
23:21	/	/	/
20	R/W	0x0	FRAC_EN Fraction Enable.
19:17	/	/	/
16:0	R/W	0x0	FRAC_IN Fraction In.

4.1.6.10 0x0120 PLL_PERI Pattern0 Control Register (Default Value: 0x0000_0000)

Offset: 0x0120			Register Name: PLL_PERI_PAT0_CTRL_REG

Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SIG_DELT_PAT_EN Sigma Delta Pattern Enable.
30:29	R/W	0x0	SPR_FREQ_MODE Spread Frequency Mode 00: DC=0 01: DC=1 10: Triangular(1-bit) 11: Triangular(n-bit)
28:20	R/W	0x0	WAVE_STEP Wave Step.
19	R/W	0x0	SDM_CLK_SEL SDM Clock Select 0: 24MHz 1: 12MHz When the PLL_INPUT_DIV2 is 1, the bit needs to be set to 1.
18:17	R/W	0x0	FREQ Frequency 00: 31.5KHz 01: 32KHz 10: 32.5KHz 11: 33KHz
16:0	R/W	0x0	WAVE_BOT Wave Bottom.

4.1.6.11 0x0124 PLL_PERI Pattern1 Control Register (Default Value: 0x0000_0000)

Offset: 0x0124			Register Name: PLL_PERI_PAT1_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:25	/	/	/
24	R/W	0x0	DITHER_EN Dither Enable.
23:21	/	/	/
20	R/W	0x0	FRAC_EN Fraction Enable.
19:17	/	/	/
16:0	R/W	0x0	FRAC_IN Fraction In.

4.1.6.12 0x0140 PLL_VIDEO0 Pattern0 Control Register (Default Value: 0x0000_0000)

Offset: 0x0140	Register Name: PLL_VIDEO0_PAT0_CTRL_REG

Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SIG_DELT_PAT_EN Sigma Delta Pattern Enable.
30:29	R/W	0x0	SPR_FREQ_MODE Spread Frequency Mode 00: DC=0 01: DC=1 10: Triangular(1-bit) 11: Triangular(n-bit)
28:20	R/W	0x0	WAVE_STEP Wave Step.
19	R/W	0x0	SDM_CLK_SEL SDM Clock Select 0: 24MHz 1: 12MHz When the PLL_INPUT_DIV2 is 1, the bit needs to be set to 1.
18:17	R/W	0x0	FREQ Frequency 00: 31.5KHz 01: 32KHz 10: 32.5KHz 11: 33KHz
16:0	R/W	0x0	WAVE_BOT Wave Bottom.

4.1.6.13 0x0144 PLL_VIDEO0 Pattern1 Control Register (Default Value: 0x0000_0000)

Offset: 0x0144			Register Name: PLL_VIDEO0_PAT1_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:25	/	/	/
24	R/W	0x0	DITHER_EN Dither Enable.
23:21	/	/	/
20	R/W	0x0	FRAC_EN Fraction Enable.
19:17	/	/	/
16:0	R/W	0x0	FRAC_IN Fraction In.

4.1.6.14 0x0180 PLL_AUDIO1 Pattern0 Control Register (Default Value: 0x0000_0000)

Offset: 0x0180	Register Name: PLL_AUDIO1_PAT0_CTRL_REG

Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SIG_DELT_PAT_EN Sigma Delta Pattern Enable.
30:29	R/W	0x0	SPR_FREQ_MODE Spread Frequency Mode 00: DC=0 01: DC=1 10: Triangular(1-bit) 11: Triangular(n-bit)
28:20	R/W	0x0	WAVE_STEP Wave Step.
19	R/W	0x0	SDM_CLK_SEL SDM Clock Select 0: 24MHz 1: 12MHz When the PLL_INPUT_DIV2 is 1, the bit needs to be set to 1.
18:17	R/W	0x0	FREQ Frequency 00: 31.5KHz 01: 32KHz 10: 32.5KHz 11: 33KHz
16:0	R/W	0x0	WAVE_BOT Wave Bottom.

4.1.6.15 0x0184 PLL_AUDIO1 Pattern1 Control Register (Default Value: 0x0000_0000)

Offset: 0x0184			Register Name: PLL_AUDIO1_PAT1_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:25	/	/	/
24	R/W	0x0	DITHER_EN Dither Enable.
23:21	/	/	/
20	R/W	0x0	FRAC_EN Fraction Enable.
19:17	/	/	/
16:0	R/W	0x0	FRAC_IN Fraction In.

4.1.6.16 0x0200 PLL_CPU SSC Register (Default Value: 0x4CCC_A000)

Offset: 0x0200	Register Name: PLL_CPU_SSC_REG

Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	PLL_SSC_MODE PLL_SSC_MODE 0: Normal Mode 1: Continuously Frequency Scale
30	R/W	0x1	PLL_SSC_RSTN SSC RSTN bake up.
29	R/W	0x0	PLL_SSC_CLK_SEL SSC CLK Select 0: PLL_CLK_SDM 1: REF_CLK
28:12	R/W	0xCCCA	PLL_SSC SSC amplitude must be an integer multiple of 2^{step} . spread spectrum amplitude = (SSC amplitude + 2^{step}) * 24 / (2^{17}), unit: MHz.
11:7	/	/	/
6:4	R/W	0x0	PLL_PHASE_COMPENSATE PLL_PHASE_COMPENSATE The value of bit[6:4] is based on 24M clock, then the default PLL phase compensate is (3/24000000)s
3:0	R/W	0x0	PLL_SSC_STEP PLL_SSC_STEP 0000:0.00439MHz/us ($576/2^{17}$) 0001:0.00879MHz/us ($576/2^{16}$) 0010:0.01758MHz/us ($576/2^{15}$) 0011:0.03516MHz/us ($576/2^{14}$) 0100:0.07031MHz/us ($576/2^{13}$) 0101:0.14062MHz/us ($576/2^{12}$) 0110:0.28125MHz/us ($576/2^{11}$) 0111:0.56250MHz/us ($576/2^{10}$) 1000:1.12500MHz/us ($576/2^9$) 1001:2.25000MHz/us ($576/2^8$) 1010:4.50000MHz/us ($576/2^7$) 1011:9.00000MHz/us ($576/2^6$) Others:0.00439MHz/us($576/2^{17}$)

4.1.6.17 0x0300 PLL_CPU Bias Register (Default Value: 0x0004_0000)

Offset: 0x0300			Register Name: PLL_CPU_BIAS_REG
Bit	Read/Write	Default/Hex	Description
31:21	/	/	/

Offset: 0x0300			Register Name: PLL_CPU_BIAS_REG
Bit	Read/Write	Default/Hex	Description
20:16	R/W	0x4	PLL_CP PLL current bias control.
15:0	/	/	/

4.1.6.18 0x0310 PLL_DDR Bias Register (Default Value: 0x0002_0000)

Offset: 0x0310			Register Name: PLL_DDR_BIAS_REG
Bit	Read/Write	Default/Hex	Description
31:21	/	/	/
20:16	R/W	0x2	PLL_CP PLL bias control.
15:0	/	/	/

4.1.6.19 0x0320 PLL_PERI Bias Register (Default Value: 0x0004_0000)

Offset: 0x0320			Register Name: PLL_PERI_BIAS_REG
Bit	Read/Write	Default/Hex	Description
31:21	/	/	/
20:16	R/W	0x4	PLL_CP PLL bias control .
15:0	/	/	/

4.1.6.20 0x0340 PLL_VIDEO0 Bias Register (Default Value: 0x0004_0000)

Offset: 0x0340			Register Name: PLL_VIDEO0_BIAS_REG
Bit	Read/Write	Default/Hex	Description
31:21	/	/	/
20:16	R/W	0x4	PLL_CP PLL bias control .
15:0	/	/	/

4.1.6.21 0x0380 PLL_AUDIO1 Bias Register (Default Value: 0x0005_0000)

Offset: 0x0380			Register Name: PLL_AUDIO1_BIAS_REG
Bit	Read/Write	Default/Hex	Description
31:21	/	/	/
20:16	R/W	0x5	PLL_CP PLL bias control.
15:0	/	/	/

4.1.6.22 0x0404 PLL_CPU Tuning1 Register (Default Value: 0x0000_0000)

Offset: 0x0404			Register Name: PLL_CPU_TUN1_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	PLL_SDM_EN PLL_SDM_EN.
30:0	/	/	/

4.1.6.23 0x0510 PSI Clock Register (Default Value: 0x0000_0000)

Offset: 0x0510			Register Name: PSI_CLK_REG
Bit	Read/Write	Default/Hex	Description
31:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: HOSC 001: CLK32K 010: CLK1M_RC 011: PERIPLL1X PSI_CLK = Clock Source/M/N The clock select is lack of glitch switching and supports dynamic configuration.
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8 The clock division is lack of glitch switching and supports dynamic configuration.
7:2	/	/	/
1:0	R/W	0x0	FACTOR_M Factor M M= FACTOR_M +1 FACTOR_M is from 0 to 3 The clock division is lack of glitch switching and supports dynamic configuration.

4.1.6.24 0x0520 APB0 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0520		Register Name: APB0_CLK_REG

Bit	Read/Write	Default/Hex	Description
31:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: HOSC 001: CLK32K 010: PSI_CLK 011: PERIPLL1X APB0_CLK = Clock Source/M/N The clock select is lack of glitch switching and supports dynamic configuration.
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8 The clock division is lack of glitch switching and supports dynamic configuration.
7:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M M= FACTOR_M +1 FACTOR_M is from 0 to 31. The clock division is lack of glitch switching and supports dynamic configuration.

4.1.6.25 0x0524 APB1 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0524			Register Name: APB1_CLK_REG
Bit	Read/Write	Default/Hex	Description
31:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: HOSC 001: CLK32K 010: PSI_CLK 011: PERIPLL1X APB1_CLK = Clock Source/M/N The clock select is lack of glitch switching and supports dynamic configuration.
23:10	/	/	/

Offset: 0x0524			Register Name: APB1_CLK_REG
Bit	Read/Write	Default/Hex	Description
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8 The clock division is lack of glitch switching and supports dynamic configuration.
7:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M M= FACTOR_M +1 FACTOR_M is from 0 to 31. The clock division is lack of glitch switching and supports dynamic configuration.

4.1.6.26 0x0528 APB_UART Clock Register (Default Value: 0x0000_0000)

Offset: 0x0528			Register Name: APB_UART_CLK_REG
Bit	Read/Write	Default/Hex	Description
31:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: HOSC 001: CLK32K 010: PERIPLL_480M 011: PERIPLL1X 100: AUDIO1PLL_DIV2 Note: This clock MUX is glitch-free switching, and supports dynamic configuration. APB_UART_CLK = Clock Source/M/N.
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8 Note: This clock DIV is glitch-free switching, and supports dynamic configuration.
7:5	/	/	/

Offset: 0x0528			Register Name: APB_UART_CLK_REG
Bit	Read/Write	Default/Hex	Description
4:0	R/W	0x0	FACTOR_M Factor M M= FACTOR_M +1. FACTOR_M is from 0 to 31. Note: This clock DIV is glitch-free switching, and supports dynamic configuration.

4.1.6.27 0x0540 MBUS Bus Gating Reset Register (Default Value: 0x4000_0000)

Offset: 0x0540			Register Name: MBUS_BGR_REG
Bit	Read/Write	Default/Hex	Description
31	/	/	/
30	R/W	0x1	MBUS_RST. MBUS Reset. 0: Assert 1: De-assert
29:0	/	/	/

4.1.6.28 0x0544 MBUS Clock Register (Default Value: 0x8000_0000)

Offset: 0x0544			Register Name: MBUS_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x1	MBUS_CLK_GATING. Gating Clock. 0: Clock is OFF 1: Clock is ON DE_CLK = Clock Source/M.
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL. Clock Source Select 00:HOSC 01:PERIPLL2X 10: AUDIO1PLL_DIV2 11:DDRPLL
23:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M M= FACTOR_M +1. FACTOR_M is from 0 to 31.

4.1.6.29 0x0600 DE Clock Register (Default Value: 0x0000_0000)

Offset: 0x0600			Register Name: DE_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	DE_CLK_GATING Gating Clock 0: Clock is OFF 1: Clock is ON DE_CLK = Clock Source/M.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: PERIPLL2X 001: VIDEO0PLL4X 010: AUDIO1PLL_DIV2
23:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M M= FACTOR_M +1 FACTOR_M is from 0 to 31.

4.1.6.30 0x060C DE Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x060C			Register Name: DE_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	DE_RST DE Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	DE_GATING Gating Clock for DE 0: Mask 1: Pass

4.1.6.31 0x0620 DI Clock Register (Default Value: 0x0000_0000)

Offset: 0x0620			Register Name: DI_CLK_REG
Bit	Read/Write	Default/Hex	Description

Offset: 0x0620			Register Name: DI_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	DI_CLK_GATING Gating Clock 0: Clock is OFF 1: Clock is ON DI_CLK = Clock Source/M.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: PERIPLL2X 001: VIDEO0PLL4X 010: AUDIO1PLL_DIV2
23:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M M= FACTOR_M +1 FACTOR_M is from 0 to 31.

4.1.6.32 0x062C DI Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x062C			Register Name: DI_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	DI_RST DI Reset 0: Assert 1: DE-assert
15:1	/	/	/
0	R/W	0x0	DI_GATING Gating Clock for DI 0: Mask 1: Pass

4.1.6.33 0x0630 G2D Clock Register (Default Value: 0x0000_0000)

Offset: 0x0630			Register Name: G2D_CLK_REG
Bit	Read/Write	Default/Hex	Description

Offset: 0x0630			Register Name: G2D_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	G2D_CLK_GATING Gating Clock 0: Clock is OFF 1: Clock is ON G2D_CLK = Clock Source/M.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: PERIPLL2X 001: VIDEO0PLL4X 010: AUDIO1PLL_DIV2
23:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M M= FACTOR_M +1 FACTOR_M is from 0 to 31.

4.1.6.34 0x063C G2D Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x063C			Register Name: G2D_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	G2D_RST G2D Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	G2D_GATING Gating Clock for G2D 0: Mask 1: Pass

4.1.6.35 0x0680 CE Clock Register (Default Value: 0x0000_0000)

Offset: 0x0680			Register Name: CE_CLK_REG
Bit	Read/Write	Default/Hex	Description

Offset: 0x0680			Register Name: CE_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	CE_CLK_GATING Gating Clock 0: Clock is OFF 1: Clock is ON CE_CLK = Clock Source/M/N.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: HOSC 01: PERIPLL2X 10: PERIPLL1X
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M M= FACTOR_M +1 FACTOR_M is from 0 to 15.

4.1.6.36 0x068C CE Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x068C			Register Name: CE_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	CE_RST CE Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	CE_GATING Gating Clock for CE 0: Mask 1: Pass

4.1.6.37 0x0690 VE Clock Register (Default Value: 0x0000_0000)

Offset: 0x0690			Register Name: VE_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	VE_CLK_GATING Gating Clock 0: Clock is OFF 1: Clock is ON VE_CLK = Clock Source/M.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: PERIPLL_480M 001: PERIPLL2X 010: AUDIO1PLL_DIV2 011: CPUPLL 100: DDRPLL
23:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M M= FACTOR_M +1 FACTOR_M is from 0 to 31.

4.1.6.38 0x069C VE Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x069C			Register Name: VE_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	VE_RST VE Reset 0: Assert 1: DE-assert
15:1	/	/	/
0	R/W	0x0	VE_GATING Gating Clock for VE 0: Mask 1: Pass

4.1.6.39 0x070C DMA Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x070C			Register Name: DMA_BGR_REG
Bit	Read/Write	Default/Hex	Description

Offset: 0x070C			Register Name: DMA_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	DMA_RST DMA Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	DMA_GATING Gating Clock for DMA 0: Mask 1: Pass

4.1.6.40 0x073C HSTIMER Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x073C			Register Name: HSTIMER_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	HSTIMER_RST HSTIMER Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	HSTIMER_GATING Gating Clock for HSTIMER 0: Mask 1: Pass

4.1.6.41 0x0740 AVS Clock Register (Default Value: 0x0000_0000)

Offset: 0x0740			Register Name: AVS_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	AVS_CLK_GATING Gating Clock 0: Clock is OFF 1: Clock is ON AVS_CLK = HOSC.
30:0	/	/	/

4.1.6.42 0x078C DBGSYS Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x078C			Register Name: DBGSYS_BGR_REG

Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	DBGSYS_RST DBGSYS Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	DBGSYS_GATING Gating Clock for DBGSYS 0: Mask 1: Pass

4.1.6.43 0x0790 PWM_BL Clock Register (Default Value: 0x0000_0000)

Offset: 0x0790			Register Name: PWM_BL_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	PWM_BL_CLK_GATING. Gating Clock. 0: Clock is OFF 1: Clock is ON PWM_BL_CLK = Clock Source/M.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL. Clock Source Select 000: PERIPLL2X 001: AUDIO1PLL_DIV2
23:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M. M= FACTOR_M +1. FACTOR_M is from 0 to 31.

4.1.6.44 0x0794 PWM_BL Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x0794			Register Name: PWM_BL_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	PWM_BL_RST. PWM_BL Reset. 0: Assert 1: De-assert
15:1	/	/	/

Offset: 0x0794			Register Name: PWM_BL_BGR_REG
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	PWM_BL_GATING. Gating Clock For PWM_BL. 0: Mask 1: Pass

4.1.6.45 0x07AC PWM Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x07AC			Register Name: PWM_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	PWM_RST PWM Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	PWM_GATING Gating Clock for PWM 0: Mask 1: Pass

4.1.6.46 0x0800 DRAM Clock Register (Default Value: 0x0000_0000)

Offset: 0x0800			Register Name: DRAM_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	DRAM_CLK_GATING Gating Clock 0: Clock is OFF 1: Clock is ON DRAM_CLK = Clock Source/M/N.
30	R/W	0x0	DRAM_MODULE_RST Module Reset 0: Assert 1: De-assert
29:28	/	/	/

Offset: 0x0800			Register Name: DRAM_CLK_REG
Bit	Read/Write	Default/Hex	Description
27	R/WAC	0x0	DRAM_UPD SDRCLK Configuration 0 update 0: Invalid 1: Valid Setting 1 will validate the bit. It will be automatically cleared after the bit is valid. Here supports dram req/ack signal. When dram_update is set to 1, dram_clk_sel/dram_div2/dram_clk1 will be updated.
26:24	R/W	0x0	DRAM_CLK_SEL Clock Source Select 00: PERIPLL2X 01: DDRPLL 10: PERIPLL_800M 11: AUDIO1PLL_DIV2 The clock select is lack of glitch switching and supports dynamic configuration.
23:10	/	/	/
9:8	R/W	0x0	DRAM_DIV2 Factor N 00: 1 01: 2 10: 4 11: 8
7:2	/	/	/
1:0	R/W	0x0	DRAM_DIV1 Factor M M= FACTOR_M +1 FACTOR_M is from 0 to 3. The clock select is lack of glitch switching and supports dynamic configuration.

4.1.6.47 0x0804 MBUS Master Clock Gating Register (Default Value: 0x0000_0000)

Offset: 0x0804			Register Name: MBUS_MAT_CLK_GATING_REG
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11	R/W	0x0	Reserved.

Offset: 0x0804			Register Name: MBUS_MAT_CLK_GATING_REG
Bit	Read/Write	Default/Hex	Description
10	R/W	0x0	G2D_MCLK_EN Gating MBUS Clock for G2D 0: Mask 1: Pass
9:7	/	/	/
6:2	/	/	/
1	R/W	0x0	VE_MCLK_EN Gating MBUS Clock for VE 0: Mask 1: Pass
0	R/W	0x0	DMA_MCLK_EN Gating MBUS Clock for DMA 0: Mask 1: Pass

4.1.6.48 0x080C DRAM Bus Gating Reset Register (Default Value: 0x0000_0001)

Offset: 0x080C			Register Name: DRAM_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	DRAM_RST DRAM Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x1	DRAM_GATING Gating Clock for DRAM 0: Mask 1: Pass

4.1.6.49 0x0830 SMHC0 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0830			Register Name: SMHC0_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SMHC0_CLK_GATING Gating Clock 0: Clock is OFF 1: Clock is ON SMHC0_CLK = Clock Source/M/N.
30:27	/	/	/

Offset: 0x0830			Register Name: SMHC0_CLK_REG
Bit	Read/Write	Default/Hex	Description
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: HOSC 01: PERIPLL1X 10: PERIPLL2X 11: AUDIO1PLL_DIV2
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M M= FACTOR_M +1 FACTOR_M is from 0 to 15.

4.1.6.50 0x0838 SMHC2 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0838			Register Name: SMHC2_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SMHC2_CLK_GATING Gating Clock 0: Clock is OFF 1: Clock is ON SMHC2_CLK = Clock Source/M/N.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: HOSC 001: PERIPLL1X 010: PERIPLL2X 011: PERIPLL_800M 100: AUDIO1PLL_DIV2
23:10	/	/	/

Offset: 0x0838			Register Name: SMHC2_CLK_REG
Bit	Read/Write	Default/Hex	Description
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M M= FACTOR_M +1 FACTOR_M is from 0 to 15.

4.1.6.51 0x084C SMHC Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x084C			Register Name: SMHC_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:19	/	/	/
18	R/W	0x0	SMHC2_RST SMHC2 Reset 0: Assert 1: De-assert
17	/	/	/
16	R/W	0x0	SMHC0_RST SMHC0 Reset 0: Assert 1: De-assert
15:3	/	/	/
2	R/W	0x0	SMHC2_GATING Gating Clock for SMHC2 0: Mask 1: Pass
1	/	/	/
0	R/W	0x0	SMHC0_GATING Gating Clock for SMHC0 0: Mask 1: Pass

4.1.6.52 0x090C UART Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x090C	Register Name: UART_BGR_REG

Bit	Read/Write	Default/Hex	Description
31:22	/	/	/
21	R/W	0x0	UART5_RST UART5 Reset 0: Assert 1: De-assert
20	R/W	0x0	UART4_RST UART4 Reset 0: Assert 1: De-assert
19	R/W	0x0	UART3_RST UART3 Reset 0: Assert 1: De-assert
18	R/W	0x0	UART2_RST UART2 Reset 0: Assert 1: De-assert
17	R/W	0x0	UART1_RST UART1 Reset 0: Assert 1: De-assert
16	R/W	0x0	UART0_RST UART0 Reset 0: Assert 1: De-assert
15:6	/	/	/
5	R/W	0x0	UART5_GATING Gating Clock for UART5 0: Mask 1: Pass
4	R/W	0x0	UART4_GATING Gating Clock for UART0 0: Mask 1: Pass
3	R/W	0x0	UART3_GATING Gating Clock for UART3 0: Mask 1: Pass
2	R/W	0x0	UART2_GATING Gating Clock for UART2 0: Mask 1: Pass

1	R/W	0x0	UART1_GATING Gating Clock for UART1 0: Mask 1: Pass
0	R/W	0x0	UART0_GATING Gating Clock for UART0 0: Mask 1: Pass

4.1.6.53 0x091C TWI Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x091C			Register Name: TWI_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:19	/	/	/
18	R/W	0x0	TWI2_RST TWI2 Reset 0: Assert 1: De-assert
17	R/W	0x0	TWI1_RST TWI1 Reset 0: Assert 1: De-assert
16	R/W	0x0	TWI0_RST TWI0 Reset 0: Assert 1: De-assert
15:3	/	/	/
2	R/W	0x0	TWI2_GATING Gating Clock for TWI2 0: Mask 1: Pass
1	R/W	0x0	TWI1_GATING Gating Clock for TWI1 0: Mask 1: Pass
0	R/W	0x0	TWI0_GATING Gating Clock for TWI0 0: Mask 1: Pass

4.1.6.54 0x0940 SPI0 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0940	Register Name: SPI0_CLK_REG

Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SPI0_CLK_GATING Gating Clock 0: Clock is OFF 1: Clock is ON SPI0_CLK = Clock Source/M/N.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: HOSC 001: PERIPLL1X 010: PERIPLL2X 011: AUDIO1PLL_DIV2 100: AUDIO1PLL_DIV5
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M M= FACTOR_M +1 FACTOR_M is from 0 to 15.

4.1.6.55 0x0944 SPI1 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0944			Register Name: SPI1_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SPI1_CLK_GATING Gating Clock 0: Clock is OFF 1: Clock is ON SPI1_CLK = Clock Source/M/N.
30:27	/	/	/

Offset: 0x0944			Register Name: SPI1_CLK_REG
Bit	Read/Write	Default/Hex	Description
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: HOSC 001: PERIPLL1X 010: PERIPLL2X 011: AUDIO1PLL_DIV2 100: AUDIO1PLL_DIV5
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M M= FACTOR_M +1 FACTOR_M is from 0 to 15.

4.1.6.56 0x096C SPI Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x096C			Register Name: SPI_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17	R/W	0x0	SPI1_RST SPI1 Reset 0: Assert 1: De-assert
16	R/W	0x0	SPI0_RST SPI0 Reset 0: Assert 1: De-assert
15:2	/	/	/
1	R/W	0x0	SPI1_GATING Gating Clock for SPI1 0: Mask 1: Pass

Offset: 0x096C			Register Name: SPI_BGR_REG
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	SPI0_GATING Gating Clock for SPI0 0: Mask 1: Pass

4.1.6.57 0x0980 SPIF Clock Register (Default Value: 0x0000_0000)

Offset: 0x0980			Register Name: SPIF_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SPIF_CLK_GATING. Gating Clock. 0: Clock is OFF 1: Clock is ON SPIF_CLK = Clock Source/M/N.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL. Clock Source Select 000: HOSC 001: PERIPLL1X 010: PERIPLL2X 011: PERIPLL_800M
23:13	/	/	/
12:8	R/W	0x0	FACTOR_N. Factor N N= FACTOR_N +1. FACTOR_N is from 0 to 31.
7:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M M= FACTOR_M +1. FACTOR_M is from 0 to 31.

4.1.6.58 0x0984 SPIF Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x0984			Register Name: SPIF_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/

Offset: 0x0984			Register Name: SPIF_BGR_REG
Bit	Read/Write	Default/Hex	Description
16	R/W	0x0	SPIF_RST. SPIF Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	SPIF_GATING. Gating Clock For SPIF 0: Mask 1: Pass

4.1.6.59 0x09C0 IR_RX Clock Register (Default Value: 0x0000_0000)

Offset: 0x09C0			Register Name: IR_RX_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	IR_RX_CLK_GATING. Gating Clock. 0: Clock is OFF 1: Clock is ON IR_RX_CLK = Clock Source/M/N.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL. Clock Source Select 0: HOSC 1: PERIPLL1X
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N. Factor N 00: 1 01: 2 10: 4 11: 8
7:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M M= FACTOR_M +1. FACTOR_M is from 0 to 15.

4.1.6.60 0x09CC IR_RX Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x09CC	Register Name: IR_RX_BGR_REG

Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	IR_RX_RST. IR_RX Reset. 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	IR_RX_GATING. Gating Clock For IR_RX. 0: Mask 1: Pass

4.1.6.61 0x09EC GPADC Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x09EC			Register Name: GPADC_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	GPADC_RST GPADC Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	GPADC_GATING Gating Clock for GPADC 0: Mask 1: Pass

4.1.6.62 0x09FC THS Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x09FC			Register Name: THS_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	THS_RST THS Reset 0: Assert 1: DE-assert
15:1	/	/	/
0	R/W	0x0	THS_GATING Gating Clock for THS 0: Mask 1: Pass

4.1.6.63 0x0A10 I2S0 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0A10			Register Name: I2S0_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	I2S0_CLK_GATING Gating Clock 0: Clock is OFF 1: Clock is ON I2S0_CLK = Clock Source/M/N.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: AUDIO1PLL_DIV2 01: AUDIO1PLL_DIV5 10: PERIPLL1X
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: /1 01: /2 10: /4 11: /8
7:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M. M= FACTOR_M +1 FACTOR_M is from 0 to 31.

4.1.6.64 0x0A20 I2S Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x0A20			Register Name: I2S_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	I2S0_RST I2S0 Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	I2S0_GATING Gating Clock for I2S0 0: Mask 1: Pass

4.1.6.65 0x0A24 OWA_TX Clock Register (Default Value: 0x0000_0000)

Offset: 0x0A24			Register Name: OWA_TX_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	OWA_TX_CLK_GATING Gating Clock 0: Clock is OFF 1: Clock is ON OWA_TX_CLK = Clock Source/M/N.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: AUDIO1PLL_DIV2 01: AUDIO1PLL_DIV5 10: PERIPLL1X
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: /1 01: /2 10: /4 11: /8
7:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M. M= FACTOR_M +1 FACTOR_M is from 0 to 31.

4.1.6.66 0x0A28 OWA_RX Clock Register (Default Value: 0x0000_0000)

Offset: 0x0A28			Register Name: OWA_RX_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	OWA_RX_CLK_GATING Gating Clock 0: Clock is OFF 1: Clock is ON OWA_RX_CLK = Clock Source/M/N.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: PERIPLL1X 01: AUDIO1PLL_DIV2 10: AUDIO1PLL_DIV5

Offset: 0x0A28			Register Name: OWA_RX_CLK_REG
Bit	Read/Write	Default/Hex	Description
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: /1 01: /2 10: /4 11: /8
7:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M. M= FACTOR_M +1 FACTOR_M is from 0 to 31.

4.1.6.67 0x0A2C OWA Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x0A2C			Register Name: OWA_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	OWA_RST OWA Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	OWA_GATING Gating Clock for OWA 0: Mask 1: Pass

4.1.6.68 0x0A50 AUDIO_CODEC_DAC Clock Register (Default Value: 0x0000_0000)

Offset: 0x0A50			Register Name: AUDIO_CODEC_DAC_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	AUDIO_CODEC_DAC_CLK_GATING Gating Clock 0: Clock is OFF 1: Clock is ON AUDIO_CODEC_DAC_CLK = Clock Source/M/N.
30:27	/	/	/

Offset: 0x0A50			Register Name: AUDIO_CODEC_DAC_CLK_REG
Bit	Read/Write	Default/Hex	Description
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: AUDIO1PLL_DIV2 01: AUDIO1PLL_DIV5 10: PERIPLL1X
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: /1 01: /2 10: /4 11: /8
7:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M. M= FACTOR_M +1 FACTOR_M is from 0 to 31.

4.1.6.69 0x0A54 AUDIO_CODEC_ADC Clock Register (Default Value: 0x0000_0000)

Offset: 0x0A54			Register Name: AUDIO_CODEC_ADC_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	AUDIO_CODEC_ADC_CLK_GATING Gating Clock 0: Clock is OFF 1: Clock is ON AUDIO_CODEC_ADC_CLK = Clock Source/M/N.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: AUDIO1PLL_DIV2 01: AUDIO1PLL_DIV5 10: PERIPLL1X
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: /1 01: /2 10: /4 11: /8
7:5	/	/	/

Offset: 0x0A54			Register Name: AUDIO_CODEC_ADC_CLK_REG
Bit	Read/Write	Default/Hex	Description
4:0	R/W	0x0	FACTOR_M Factor M. M= FACTOR_M +1 FACTOR_M is from 0 to 31.

4.1.6.70 0x0A5C AUDIO_CODEC Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x0A5C			Register Name: AUDIO_CODEC_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	AUDIO_CODEC_RST AUDIO_CODEC Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	AUDIO_CODEC_GATING Gating Clock for AUDIO_CODEC 0: Mask 1: Pass

4.1.6.71 0x0A70 USB0 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0A70			Register Name: USB0_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	USB0_CLKEN Gating Clock for OHCI0 0: Clock is OFF 1: Clock is ON
30	R/W	0x0	USBPHY0_RSTN USB PHY0 Reset 0: Assert 1: De-assert
29:26	/	/	/
25:24	R/W	0x0	USB0_CLK12M_SEL OHCI0 12M Source Select 00: 12M divided from 48MHz 01: 12M divided from 24MHz 10: RTC_32K 11: /
23:0	/	/	/

4.1.6.72 0x0A8C USB Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x0A8C			Register Name: USB_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:25	/	/	/
24	R/W	0x0	USBOTG0_RST USBOTG0 Reset 0: Assert 1: De-assert
23:21	/	/	/
20	R/W	0x0	USBEHCI0_RST USBEHCI0 Reset 0: Assert 1: De-assert
19:17	/	/	/
16	R/W	0x0	USBOHCI0_RST USBOHCI0 Reset 0: Assert 1: De-assert
15:9	/	/	/
8	R/W	0x0	USBOTG0_GATING Gating Clock for USBOTG0 0: Mask 1: Pass
7:5	/	/	/
4	R/W	0x0	USBEHCI0_GATING Gating Clock for USBEHCI0 0: Mask 1: Pass
3:1	/	/	/
0	R/W	0x0	USBOHCI0_GATING Gating Clock for USBOHCI0 0: Mask 1: Pass

4.1.6.73 0x0AA0 COMBOPHY0 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0AA0			Register Name: COMBOPHY0_CLK_REG
Bit	Read/Write	Default/Hex	Description

Offset: 0x0AA0			Register Name: COMBOPHY0_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	COMBOPHY0_CLK_GATING. Gating Clock. 0: Clock is OFF 1: Clock is ON COMBOPHY0_CLK = Clock Source/M.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL. Clock Source Select 00: VIDEO0PLL4X 01: PERIPLL2X 10: AUDIO1PLL_DIV2
23:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M M= FACTOR_M +1. FACTOR_M is from 0 to 31.

4.1.6.74 0x0AA4 COMBOPHY0 Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x0AA4			Register Name: COMBOPHY0_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	COMBOPHY0_GATING. Gating Clock For COMBOPHY0. 0: Mask 1: Pass

4.1.6.75 0x0ABC DPSS_TOP Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x0ABC			Register Name: DPSS_TOP_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	DPSS_TOP_RST DPSS_TOP Reset 0: Assert 1: De-assert
15:1	/	/	/

Offset: 0x0ABC			Register Name: DPSS_TOP_BGR_REG
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	DPSS_TOP_GATING Gating Clock for DPSS_TOP 0: Mask 1: Pass

4.1.6.76 0x0B24 DSI Clock Register (Default Value: 0x0000_0000)

Offset: 0x0B24			Register Name: DSI_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	DSI_CLK_GATING Gating Clock 0: Clock is OFF 1: Clock is ON DSI_CLK = Clock Source/M.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: HOSC 001: PERIPLL1X 010: VIDEO0PLL2X 011: AUDIO1PLL_DIV2
23:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M. M= FACTOR_M +1 FACTOR_M is from 0 to 15.

4.1.6.77 0x0B4C DSI Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x0B4C			Register Name: DSI_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	DSI_RST DSI Reset 0: Assert 1: DE-assert
15:1	/	/	/

Offset: 0x0B4C			Register Name: DSI_BGR_REG
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	DSI_GATING Gating Clock for DSI 0: Mask 1: Pass

4.1.6.78 0x0B60 TCONLCD Clock Register (Default Value: 0x0000_0000)

Offset: 0x0B60			Register Name: TCONLCD_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	TCONLCD_CLK_GATING Gating Clock 0: Clock is OFF 1: Clock is ON TCONLCD_CLK = Clock Source/M/N.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: VIDEO0PLL1X 001: VIDEO0PLL4X 010: PERIPLL2X 011: AUDIO1PLL_DIV2
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M. M= FACTOR_M +1 FACTOR_M is from 0 to 15.

4.1.6.79 0x0B7C TCONLCD Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x0B7C			Register Name: TCONLCD_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/

Offset: 0x0B7C			Register Name: TCONLCD_BGR_REG
Bit	Read/Write	Default/Hex	Description
16	R/W	0x0	TCONLCD_RST TCON LCD Reset 0: Assert 1: DE-assert
15:1	/	/	/
0	R/W	0x0	TCONLCD_GATING Gating Clock for TCON LCD 0: Mask 1: Pass

4.1.6.80 0x0BAC LVDS Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x0BAC			Register Name: LVDS_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	LVDS0_RST LVDS0 Reset 0: Assert 1: De-assert
15:0	/	/	/

4.1.6.81 0x0C50 TPADC Clock Register (Default Value: 0x0000_0000)

Offset: 0x0C50			Register Name: TPADC_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	TPADC_CLK_GATING Gating Clock 0: Clock is OFF 1: Clock is ON TPADC_CLK = Clock Source.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: HOSC 001: AUDIO1PLL_DIV2
23:0	/	/	/

4.1.6.82 0x0C5C TPADC Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x0C5C			Register Name: TPADC_BGR_REG

Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	TPADC_RST TPADC Reset 0: Assert 1: DE-assert
15:1	/	/	/
0	R/W	0x0	TPADC_GATING Gating Clock for TPADC 0: Mask 1: Pass

4.1.6.83 0x0D00 RISC-V Clock Register (Default Value: 0x0000_0100)

Offset: 0x0D00			Register Name: RISC-V_CLK_REG
Bit	Read/Write	Default/Hex	Description
31:27	/	/	/
26:24	R/W	0x0	RISC-V_CLK_SEL Clock Source Select 000: HOSC 001: CLK32K 010: CLK1M_RC 011: PERIPLL_800M 100: PERIPLL1X 101: CPUPLL 110: AUDIO1PLL_DIV2 RISC-V_CLK = Clock Source/M RISC-V_AXI_CLK = RISC-V_CLK/N The clock select is lack of glitch switching and supports dynamic configuration.
23:10	/	/	/
9:8	R/W	0x1	RISC-V_AXI_DIV_CFG Factor N N = FACTOR_N + 1 FACTOR_N is from 1 to 3. The clock division is lack of glitch switching and supports dynamic configuration.
7:5	/	/	/

Offset: 0x0D00			Register Name: RISC_V_CLK_REG
Bit	Read/Write	Default/Hex	Description
4:0	R/W	0x0	RISC_V_DIV_CFG Factor M. M= FACTOR_M +1 FACTOR_M is from 0 to 31. The clock division is lack of glitch switching and supports dynamic configuration.

4.1.6.84 0x0D0C RISC_V_CFG Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x0D0C			Register Name: RISC_V_CFG_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	RISC_V_CFG_GATING Gating Clock for RISC_V 0: Mask 1: Pass

4.1.6.85 0x0F00 CCU Security Switch Register (Default Value: 0x0000_0000)

Offset: 0x0F00			Register Name: CCU_SEC_SWITCH_REG
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/
2	R/W	0x0	MBUS_SEC MBUS clock register security 0: Secure 1: Non-secure
1	R/W	0x0	BUS_SEC Bus relevant registers security 0: Secure 1: Non-secure
0	R/W	0x0	PLL_SEC PLL relevant registers security 0: Secure 1: Non-secure

4.1.6.86 0x0F04 PLL Lock Debug Control Register (Default Value: 0x0000_0000)

Offset: 0x0F04			Register Name: PLL_LOCK_DBG_CTRL_REG
Bit	Read/Write	Default/Hex	Description

Offset: 0x0F04			Register Name: PLL_LOCK_DBG_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	PLL_LOCK_FLAG_EN Debug Enable 0: Disable 1: Enable
30:23	/	/	/
22:20	R/W	0x0	PLL_LOCK_FLAG_SEL Debug Select 000: PLL_CPUX 001: PLL_DDR 010: PLL_PERI(2X) 011: PLL_VIDEO0(4X) 100: DISPLL 101: / 110: / 111: PLL_AUDIO1 Others: /
19:0	/	/	/

4.1.6.87 0x0F30 CCU Fanout Clock Gate Register (Default Value:0x0000_0000)

Offset: 0x0F30			Register Name: CCU_FAN_GATE_REG
Bit	Read/Write	Default/Hex	Description
31:5	/	/	/
4	R/W	0x0	CLK32K_EN Gating for CLK32K 0: Clock is OFF 1: Clock is ON
3	R/W	0x0	CLK25M_EN Gating for CLK25M 0: Clock is OFF 1: Clock is ON
2	R/W	0x0	CLK16M_EN Gating for CLK16M 0: Clock is OFF 1: Clock is ON
1	R/W	0x0	CLK12M_EN Gating for CLK12M 0: Clock is OFF 1: Clock is ON

Offset: 0x0F30			Register Name: CCU_FAN_GATE_REG
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	CLK24M_EN Gating for CLK24M 0: Clock is OFF 1: Clock is ON

4.1.6.88 0x0F34 CLK27M FANOUT Register (Default Value:0x0000_0000)

Offset: 0x0F34			Register Name: CLK27M_FAN_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	CLK27M_EN Gating for CLK27M 0: Clock is OFF 1: Clock is ON SCLK=Clock Source/M/N.
30:26	/	/	/
25:24	R/W	0x0	CLK27M_SCR_SEL Clock Source Select 000: VIDEO0PLL1X 001: / 010: / 011: /
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:5	/	/	/
4:0	R/W	0x0	CLK27M_DIV0 Factor M M= FACTOR_M +1 FACTOR_M is from 0 to 31.

4.1.6.89 0x0F38 PCLK FANOUT Register (Default Value:0x0000_0000)

Offset: 0x0F38			Register Name: PCLK_FAN_REG
Bit	Read/Write	Default/Hex	Description

Offset: 0x0F38			Register Name: PCLK_FAN_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	PCLK_DIV_EN Gating for PCLK 0: Clock is OFF 1: Clock is ON PCLK = APB0_CLK/M.
30:5	/	/	/
4:0	R/W	0x0	PCLK_DIV Factor M M= FACTOR_M +1 FACTOR_M is from 0 to 31.

4.1.6.90 0x0F3C CCU FANOUT Register (Default Value:0x0000_0000)

Offset: 0x0F3C			Register Name: CCU_FAN_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23	R/W	0x0	CLK_FANOUT2_EN Gating for CLK_FANOUT2 0: Clock is OFF 1: Clock is ON
22	R/W	0x0	CLK_FANOUT1_EN Gating for CLK_FANOUT1 0: Clock is OFF 1: Clock is ON
21	R/W	0x0	CLK_FANOUT0_EN Gating for CLK_FANOUT0 0: Clock is OFF 1: Clock is ON
20:9	/	/	/
8:6	R/W	0x0	CLK_FANOUT2_SEL Clock Fanout2 Select 000: CLK32K (From PERIPLL2X) 001: CLK12M (From DCXO/2) 010: CLK16M (From PERIPLL2X) 011: CLK24M (From DCXO) 100: CLK25M (From PERIPLL1X) 101: CLK27M 110: PCLK CLK_FANOUT2 can be selected to output from the above seven sources.

Offset: 0x0F3C			Register Name: CCU_FAN_REG
Bit	Read/Write	Default/Hex	Description
5:3	R/W	0x0	CLK_FANOUT1_SEL Clock Fanout1 Select 000: CLK32K (From PERIPLL2X) 001: CLK12M (From DCXO/2) 010: CLK16M (From PERIPLL2X) 011: CLK24M (From DCXO) 100: CLK25M (From PERIPLL1X) 101: CLK27M 110: PCLK CLK_FANOUT1 can be selected to output from the above seven sources.
2:0	R/W	0x0	CLK_FANOUT0_SEL Clock Fanout0 Select 000: CLK32K (From PERIPLL2X) 001: CLK12M (From DCXO) 010: CLK16M (From PERIPLL2X) 011: CLK24M (From DCXO) 100: CLK25M (From PERIPLL1X) 101: CLK27M 110: PCLK CLK_FANOUT0 can be selected to output from the above seven sources.

4.1.6.91 0x0F40 PLL Control Register0 (Default Value: 0x0000_0000)

Offset: 0x0F40			Register Name: PLL_CTRL_REG0
Bit	Read/Write	Default/Hex	Description
31:25	/	/	/
24	R/W	0x0	TEST_CLK_SEL. PLL Reference Clock Select. 0: DCXO 1: External Test Clock
23:0	/	/	/

4.1.6.92 0x0F44 PLL Control Register1 (Default Value: 0x0003_0001)

Offset: 0x0F44			Register Name: PLL_CTRL_REG1
Bit	Read/Write	Default/Hex	Description

Offset: 0x0F44			Register Name: PLL_CTRL_REG1
Bit	Read/Write	Default/Hex	Description
31:24	R/W	0x0	KEY_FIELD. Key Field for LDO Enable bit. If the key field value is 0xA7, the bit[23:0] can be modified
23:19	/	/	/
18:16	R/W	0x3	PLLVDD_LDO_OUT_CTRL. PLLVDD LDO Output Control. 000: 0.90v 001: 0.94v 010: 0.98v 011: 1.02v 100: 1.06v 101: 1.10v 110: 1.14v 111: 1.18v
15:5	/	/	/
4	R/W	0x0	MBIAS_EN. Chip Master Bias Enable. 0: From Internal Bias 1: From ADDA Bias
3	R/W	0x0	PLLTTEST_EN. For Verify (Back door clock PLLTEST enable). Enable plltestoutput. 0: Output clock gate is 0 1: Output clock
2:1	/	/	/
0	R/W	0x1	LDO_EN. Ldo Enable. 0: Disable 1: Enable Note: PLL Power enable (power source from VCC_PLL)

4.1.6.93 0x0F48 HOSC And 32k Select Register (Default Value: 0x0000_0000)

Offset: 0x0F48			Register Name: HOSC_32K_SEL_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/

Offset: 0x0F48			Register Name: HOSC_32K_SEL_REG
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	HOSC_32K_SEL Hosc and 32k clock select in standby 0:HOSC 1:CLK32K



5 Processors

5.1 RISC-V System

5.1.1 Overview

The RISC-V system includes RISC-V IP core and related peripheral devices (RISCV_CFG, RISCV_TIMESTAMP, Watchdog, BROM, and so on), which are interconnected by BUS Matrix.

The RISCV_CFG has the following features:

- RISCV-C907 single-core system, I-Cache 32KB, D-CACHE 16KB
- Supports JTAG debugging
- Supports 128-bit GMB Master bus access to the storage bus
- Supports 32-bit AHB Master bus access to peripheral bus
- Supports 32-bit Slave bus access to internal devices of the subsystem
- Built-in 32KB BROM for startup
- Built-in interrupt controller PLIC, supporting up to 192 interrupt inputs
- Built-in Watchdog



NOTE

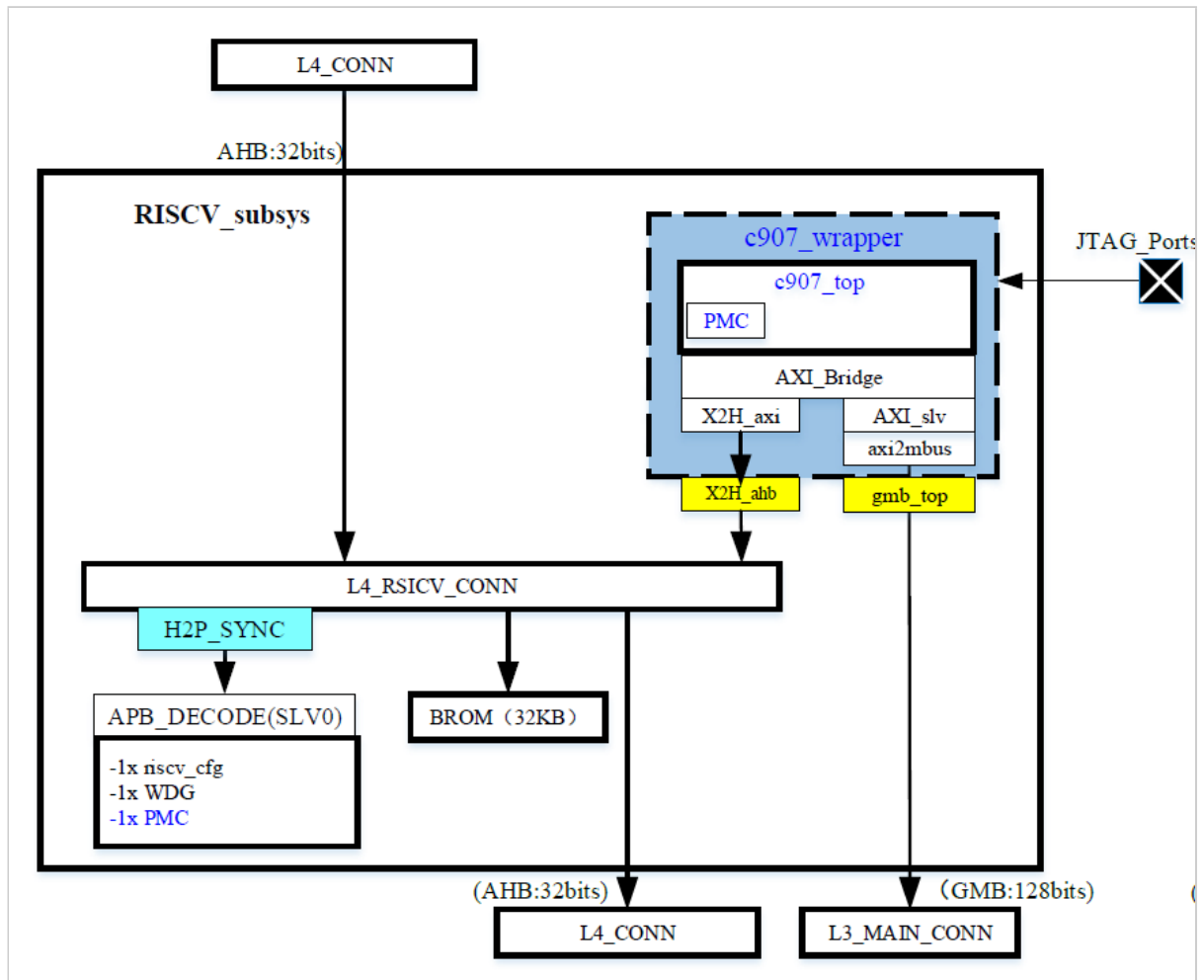
For RISC-V watchdog description, see Section 7.3 RISC-V Watchdog.

For RISC-V PLIC description, see Section 10.1 Platform-Level Interrupt Controller (PLIC).

5.1.2 Block Diagram

The following figure shows the block diagram of RISC-V system.

Figure 5-1 RISC-V System Block Diagram



- **RISC-V_subsystem:** It is the top layer of the entire system. There is an AHB Slave interface for accessing the internal resources of riscv_subsys. There is an AHB Master interface for C907 to access the resources of external systems. There is also a GMB Master interface for accessing the storage space of DRAM.
- **c907_wrapper:** It contains the Pingtoug C907 kernel. AXI Bridge is used to split access into two parts. 0x00000000 to 0x3FFFFFFF is the IO space, and 0x40000000 to 0xFFFFFFFF is the Memory space. The AXI signal is converted into an AHB signal through the AXI2AHB adapter bridge and connected to the top-level bus of the system. The AXI signal is converted into an MBUS signal through the AXI2MBUS adapter bridge (where Power is split), and then transformed into a GMB protocol signal through GMB_TOP, and finally connected to the DRAM.
- **Other modules:** AHB Matrix (used to connect various sub-modules), AHB2APB (used to convert AHB signals into APB signals), BROM (stores solidified read-only RISC-V boot code), RISC_V_CFG(used to configure relevant information of subsystems and read relevant information) TIMESTAMP (the 64-bit reference clock of the system), WDG (the watchdog of the RISC-V subsystem).

6 Thermal Sensor

6.1 Thermal Sensor Controller (THS)

6.1.1 Overview

The thermal sensors are common elements in wide range of modern system on chips (SoCs) platform. The thermal sensors are used to constantly monitor the temperature on the chip.

The SoC embeds one thermal sensor in possible hot spots on the die, sensor0 is located in the RISC-V CPU. This thermal sensor generates interrupt to lower temperature via DVFS, on reaching a certain thermal threshold.

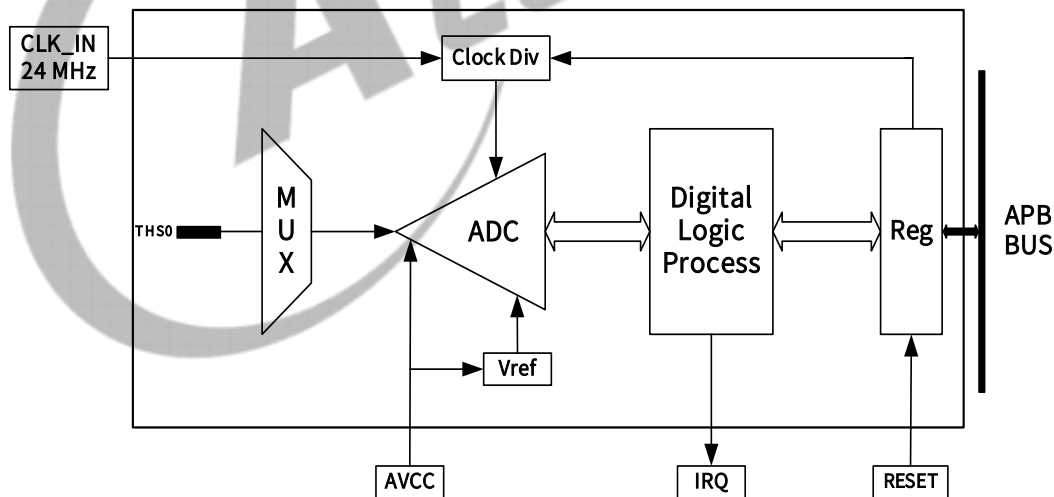
The THS has the following features:

- Averaging filter for thermal sensor reading
- Support over-temperature protection interrupt and over-temperature alarm interrupt
- Support $\pm 3^{\circ}\text{C}$ error limit from 60°C to 125°C , and $\pm 5^{\circ}\text{C}$ from -40°C to 60°C

6.1.2 Block Diagram

The following figures show the block diagrams of the THS.

Figure 6-1 THS Block Diagram

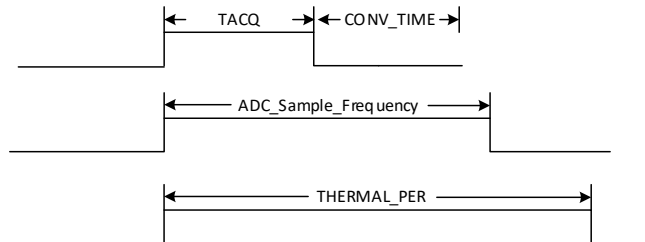


6.1.3 Functional Description

6.1.3.1 Timing Requirements

The following figure shows the timing requirements for the THS module.

Figure 6-2 Thermal Sensor Timing Requirement



CLK_IN = 24 MHz

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

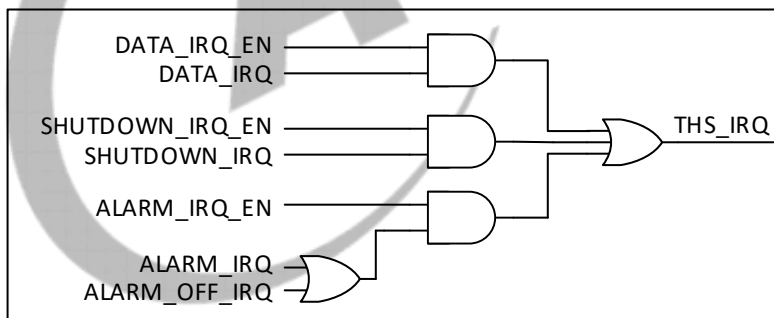
TACQ > $1/24 \text{ MHz} \times 24 \text{ Cycles}$

THERMAL_PER > ADC_Sample_Frequency > TACQ + CONV_TIME

6.1.3.2 Interrupts

The THS module has four interrupt sources: DATA_IRQ, SHUTDOWN_IRQ, ALARM_IRQ, and ALARM_OFF_IRQ. The following figure shows thermal sensor interrupt sources.

Figure 6-3 Thermal Sensor Controller Interrupt Source



DATA_IRQ

The interrupt is generated when the measured sensor_data is updated.

SHUTDOWN_IRQ

The interrupt is generated when the temperature is higher than the shutdown threshold.

ALARM_IRQ

The interrupt is generated when the temperature is higher than the Alarm_Threshold.

ALARM_OFF_IRQ

The interrupt is generated when the temperature is lower than the Alarm_Off_Thershold. It is triggered at the falling edge.

6.1.3.3 THS Temperature Conversion Formula

-40°C to 65°C: $T = (\text{sensor\_data} - 2822) / (-15.1)$

65°C to 125°C: $T = (\text{sensor\_data} - 2835) / (-15.32)$

Unit of T: Celsius degree (°C).

The sensor_data is read from the sensor data register.

6.1.4 Register List

Module Name	Base Address
THS	0x02009400

Register Name	Offset	Description
THS_CTRL	0x0000	THS Control Register
THS_EN	0x0004	THS Enable Register
THS_PER	0x0008	THS Period Control Register
THS_DATA_INTC	0x0010	THS Data Interrupt Control Register
THS_SHUT_INTC	0x0014	THS Shut Interrupt Control Register
THS_ALARM_INTC	0x0018	THS Alarm Interrupt Control Register
THS_DATA_INTS	0x0020	THS Data Interrupt Status Register
THS_SHUT_INTS	0x0024	THS Shut Interrupt Status Register
THS_ALARM_INTS	0x0028	THS Alarm off Interrupt Status Register
THS_ALARM_INTS	0x002C	THS Alarm Interrupt Status Register
THS_FILTER	0x0030	THS Median Filter Control Register
THS_ALARM_CTRL	0x0040	THS Alarm Threshold Control Register
THS_SHUTDOWN_CTRL	0x0080	THS Shutdown Threshold Control Register
THS_DATA	0x00C0	THS Data Register

6.1.5 Register Description

6.1.5.1 0x0000 THS Control Register (Default Value: 0x01DF_002F)

Offset: 0x0000			Register Name: THS_CTRL
Bit	Read/Write	Default/Hex	Description
31:16	R/W	0x1DF	TACQ ADC acquire time, default (2uS) CLK_IN/(n+1)

Offset: 0x0000			Register Name: THS_CTRL
Bit	Read/Write	Default/Hex	Description
15:0	/	0x2F	Reserved

6.1.5.2 0x0004 THS Enable Register (Default Value: 0x0000_0000)

Offset: 0x0004			Register Name: THS_EN
Bit	Read/Write	Default/Hex	Description
31:19	/	/	/
18:17	R/W	0x0	Psensor Control
16	R/W	0x0	PSENSOR_EN Enable Temperature Measurement Psensor 0: Sensor is T sensor; 1: Sensor is Psensor
15:1	/	/	/
0	R/W	0x0	THS0_EN Enable temperature measurement sensor0 0: Disable 1: Enable

6.1.5.3 0x0008 THS Period Control Register (Default Value: 0x0003_A000)

Offset: 0x0008			Register Name: THS_PER
Bit	Read/Write	Default/Hex	Description
31:12	R/W	0x3A	THERMAL_PER Default (10 ms) $4096 * (n+1) / \text{CLK\_IN}$
11:0	/	/	/

6.1.5.4 0x0010 THS Data Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0010			Register Name: THS_DATA_INTC
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	THS0_DATA_IRQ_EN Selects Temperature measurement data of sensor0 0: Disable 1: Enable

6.1.5.5 0x0014 THS Shut Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0014			Register Name: THS_SHUT_INTC
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	SHUT_INT0_EN Selects shutdown interrupt for sensor0 0: Disable 1: Enable

6.1.5.6 0x0018 THS Alarm Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0018			Register Name: THS_ALARM_INTC
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	ALARM_INT0_EN Selects Alarm interrupt for sensor0 0: Disable 1: Enable

6.1.5.7 0x0020 THS Data Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: THS_DATA_INTS
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W1C	0x0	THS0_DATA_IRQ_STS Data interrupt status for sensor0 Write '1' to clear this interrupt.

6.1.5.8 0x0024 THS Shut Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: THS_SHUT_INTS
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W1C	0x0	SHUT_INT0_STS Shutdown interrupt status for sensor0 Write '1' to clear this interrupt.

6.1.5.9 0x0028 THS Alarm off Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0028			Register Name: THS_ALARMO_INTS

Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W1C	0x0	ALARM_OFF0_STS Alarm interrupt off pending for sensor0 Write '1' to clear this interrupt.

6.1.5.10 0x002C THS Alarm Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x002C			Register Name: THS_ALARM_INTS
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W1C	0x0	ALARM_INT0_STS Alarm interrupt pending for sensor0 Write '1' to clear this interrupt.

6.1.5.11 0x0030 THS Median Filter Control Register (Default: 0x0000_0001)

Offset: 0x0030			Register Name: THS_FILTER
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/
2	R/W	0x0	FILTER_EN Filter Enable 0: Disable 1: Enable
1:0	R/W	0x1	FILTER_TYPE Average Filter Type 00: 2 01: 4 10: 8 11: 16

6.1.5.12 0x0040 THS Alarm Threshold Control Register (Default: 0x05A0_0684)

Offset: 0x0040			Register Name: THS_ALARM_CTRL
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0x5A0	ALARM0_T_HOT Thermal sensor0 Alarm Threshold for hot temperature
15:12	/	/	/

Offset: 0x0040			Register Name: THS_ALARM_CTRL
Bit	Read/Write	Default/Hex	Description
11:0	R/W	0x684	ALARM0_T_HYST Thermal sensor0 Alarm threshold for hysteresis temperature

6.1.5.13 0x0080 THS Shutdown Threshold Control Register (Default: 0x0000_04E9)

Offset: 0x0080			Register Name: THS_SHUTDOWN_CTRL
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R/W	0x4E9	SHUT0_T_HOT Thermal sensor0 Shutdown Threshold for hot temperature

6.1.5.14 0x00C0 THS Data Register (Default: 0x0000_0000)

Offset: 0x00C0			Register Name: THS_DATA
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R	0x0	THS0_DATA Temperature measurement data of sensor0

7 Timers

7.1 Timer0

7.1.1 Overview

The timer module implements the timing and counting functions. The timer module includes timer0-7, watchdog and audio video synchronization (AVS).

The main features for timer0-1 are as follows:

- Alternative count clock: CLK32K or HOSC. The CLK32K is the internal low-frequency clock.
- Supports 8 prescale factors
- Programmable 32-bit down timer
- Supports two timing modes: periodic mode and single counting mode
- Generates an interrupt when the count is decreased to 0

The watchdog is used to transmit a reset signal to reset the entire system when an exception occurs in the system. The main features for the watchdog are as follows:

- Single clock source: HOSC/750
- Supports 12 initial values
- Supports the generation of timeout interrupts
- Supports the generation of reset signals
- Supports Watchdog Restart

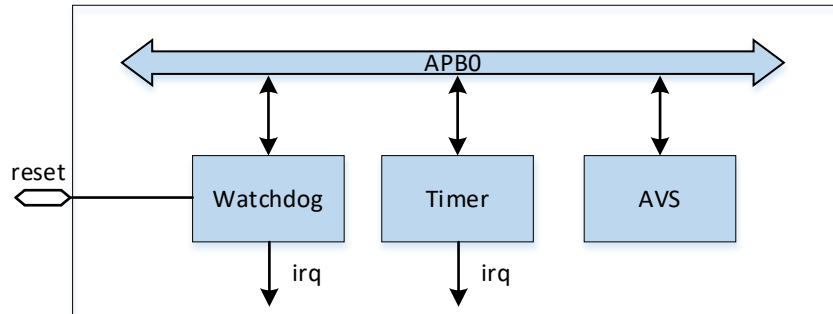
The AVS is used to synchronize the audio and video. The main features for the AVS are as follows:

- Single clock source: HOSC
- Programmable 33-bit up timer
- Supports updating the initial value anytime
- 12-bit frequency divider factor
- Supports Pause/Start function

7.1.2 Block Diagram

The following figure shows the functional block diagram of the timer module.

Figure 7-1 Timer Block Diagram



The watchdog, timer (timer0 and timer1), and AVS are all mounted at the APB0 bus. The system configures the parameters of these configure registers via APB0 bus.

The timer and watchdog are both down counters and support generating interrupts after the counting value reaches 0.

For watchdog, the system is responsible for configuring the interval value. If the system fails to restart the watchdog regularly because of some exceptional situations, such as the bus hang, the watchdog will send out a Watchdog Reset External signal to reset the system. And the signal will be transmitted to the Reset pad to reset the PMIC.

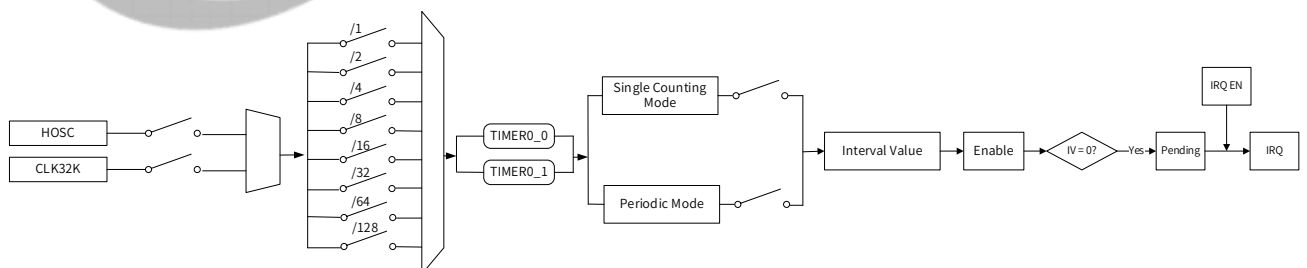
7.1.3 Functional Description

7.1.3.1 Timer

The timer (timer0 or timer1), is a 32-bit down counter. The counter value is decremented by 1 on each rising edge of the timer clock.

The following figure shows the block diagram for the timer.

Figure 7-2 Block Diagram for the Timer



The clock source for the timer can be either HOSC or CLK32K. The CLK32K is the internal low-frequency clock.

Each timer has a prescale that divides the working clock frequency by 1, 2, 4, 8, 16, 32, 64, or 128. And each timer can generate independent interrupts.

Timing Modes

The timer has two timing modes: the single counting mode and periodic mode. You can configure the timing mode via the bit [7] of TMRn_CTRL_REG (n = 0 or 1). The value 0 is for the period mode and value 1 is for the single counting mode.

- Single Counting Mode

In the single counting mode, the timer starts counting from the interval value and generates an interrupt after the counter decreases to 0, and then stops counting. It starts to count again only when this interval value is reloaded.

- Periodic Mode

In the periodic mode, the timer restarts another round of counting after generating the interrupt. It reloads data from the TMRn_INTV_VALUE_REG and then continues to count.

Formula for Calculating the Timer Time

The following formula describes the relationship among timer parameters.

$$T_{\text{timer}} = \frac{\text{TMRn\_INTV\_VALUE\_REG} - \text{TMRn\_CUR\_VALUE\_REG}}{\text{TMRn\_CLK\_SRC}} \times \text{TMRn\_CLK\_PRES}$$

Where,

The parameter n is either 0 or 1;

T_{timer} is the elapsed time of the timer;

TMRn_INTV_VALUE_REG is the interval value of the timer;

TMRn_CUR_VALUE_REG is the current value of the timer;

TMRn_CLK_SRC is the frequency of the timer clock source;

TMRn_CLK_PRES is the prescale ratio of the timer clock.

Initializing the Timer

Follow the steps below to initialize the timer:

Step 1 Configure the timer parameters clock source, prescale factor, and timing mode by writing TMRn_CTRL_REG. There is no sequence requirement of configuring the parameters.

Step 2 Write the interval value.

- Write TMRn_INTV_VALUE_REG to configure the interval value for the timer.
- Write bit [1] of TMRn_CTRL_REG to load the interval value to the timer. The value of the bit will be cleared automatically after loading the interval value.

- c) Write bit [0] of TMRn_CTRL_REG to start the timer. To get the current value of the timer, read TMRn_CUR_VALUE_REG.

Processing the Interrupt

Follow the steps below to process the interrupt:

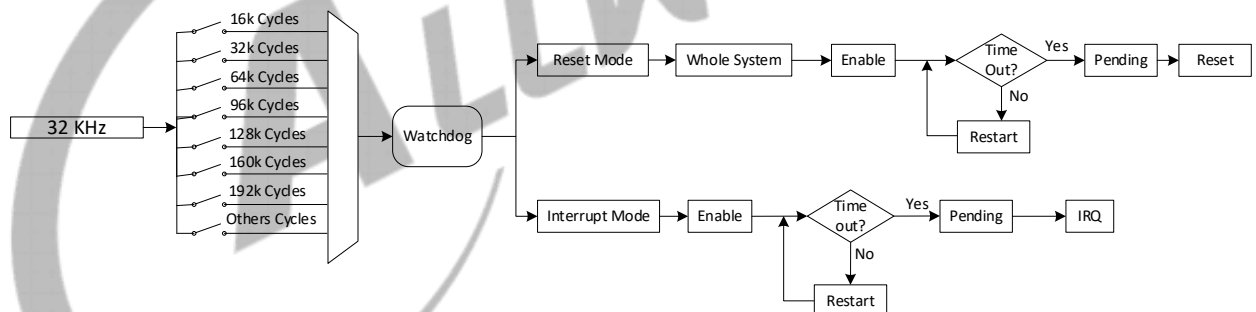
- Step 1** Enable interrupts for the timer: write the enable bit of the corresponding interrupt in [TMR_IRQ_EN_REG](#) for the timer. The timer will generate an interrupt every time the count value reaches 0.
- Step 2** After entering the interrupt process, write the pending bit of the corresponding interrupt in [TMR_IRQ_STA_REG](#) to clear the interrupt pending, and execute the process of waiting for the interrupt.
- Step 3** Resume the interrupt and continue to execute the interrupted process.

7.1.3.2 Watchdog

The watchdog is a 32-bit down counter. The counter value is decremented by 1 on each rising edge of the count clock.

The following figure shows the block diagram for the watchdog.

Figure 7-3 Block Diagram for the Watchdog



The clock source of the watchdog is HOSC/750. There are 12 configurable initial count values.

Operating Modes

The watchdog has two operating modes: the interrupt mode and reset mode.

- In the interrupt mode, when the counter value reaches 0 and [WDOG_IRQ_EN_REG](#) is enabled, the watchdog generates an interrupt.
- In the reset mode, when the counter value reaches 0, the watchdog generates a reset signal to reset the entire system.

You can configure the operating mode for the watchdog via the bit[1:0] of the [WDOG_CFG_REG](#). The value 0x2 is for the interrupt mode and the value 0x1 is for the reset mode.

Both the interrupt mode and reset mode support Watchdog Restart. You can make the watchdog to count from the initial value at any time by configuring the [WDOG_CTRL_REG](#): write 0xA57 to bit[12:1], then write 1 to bit[0].

Initializing the Watchdog

Follow the steps below to initialize the watchdog:

- Step 1** Write the bit[1:0] of [WDOG_CFG_REG](#) to configure the watchdog operating mode so that the watchdog can generate interrupts or output reset signals.
- Step 2** Write the bit[7:4] of [WDOG_MODE_REG](#) to configure the initial count value.
- Step 3** Write the bit[0] of [WDOG_MODE_REG](#) to enable the watchdog.

Processing the Interrupt

In the interrupt mode, the watchdog is used as a counter. It generates an interrupt every time the count value reaches 0.

Follow the steps below to process the interrupt:

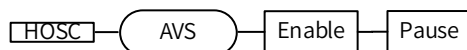
- Step 1** Write the enable bit of [WDOG_IRQ_EN_REG](#) to enable the interrupt.
- Step 2** After entering the interrupt process, write the pending bit of [WDOG_IRQ_STA_REG](#) to clear the interrupt pending and execute the process of waiting for the interrupt.
- Step 3** Resume the interrupt and continue to execute the interrupted process.

7.1.3.3 AVS

The AVS is a 33-bit up counter. The counter value is increased by 1 on each rising edge of the count clock. There is a clock gate in section 4.1 Clock Controller Unit (CCU) to control the input clock of the AVS counter. To operate the AVS, open the clock gate first.

The following figure shows the block diagram for the AVS.

Figure 7-4 Block Diagram for the AVS



The clock source of the AVS is HOSC. There is a 12-bit division factor for each AVS, N0 for AVS0 and N1 for AVS1. When the timer increases from 0 to N1 or N2, the AVS counter adds 1. When the counter reaches 33-bit upper limit, the AVS will start to count from the initial value again.

The AVS supports changing the initial value and division factor at any time. And the AVS supports restarting from the initial value or pausing at any time.

Starting or Pausing the AVS

Follow the steps below:

Step 1 Write [AVS_CNT_DIV_REG](#) to configure the division factor.

Step 2 Write AVS_CNTn_REG (n = 0 or 1) to configure the initial value.

Step 3 Write [AVS_CNT_CTL_REG](#) to enable the AVS. You can pause the AVS at any time.

7.1.4 Programming Guidelines

7.1.4.1 Configuring the Timer

The following example shows how to make a one-millisecond delay with the clock source selected as HOSC, the operating mode sets as single counting mode, and the pre-scale sets as 2.

```
writel (0x2EE0, TMR_0_INTV); //Set the interval value
writel (0x94, TMR_0_CTRL); //Select Single mode, 24 MHz clock source, 2
pre-scale
writel(readl(TMR_0_CTRL) | (1<<1), TMR_0_CTRL); //Set the Reload bit
while((readl(TMR_0_CTRL)>>1) &1); //Waiting the Reload bit turns to 0
writel(readl(TMR_0_CTRL) | (1<<0), TMR_0_CTRL); //Enable Timer0
```

7.1.4.2 Resetting the Watchdog

The following example shows how to make the watchdog to generate a reset signal to the whole system after 1 second. The clock source for the watchdog is HOSC/750.

```
writel (0x1, WDOG_CONFIG); //Set the operating mode as the reset
mode.
writel (0x10, WDOG_MODE); //Set the interval value as 1 s.
writel(readl(WDOG_MODE) | (1<<0), WDOG_MODE); //Enable the Watchdog.
```

7.1.4.3 Restarting the Watchdog

The following example shows how to restart the watchdog. In this example, the clock source is HOSC/750, the interval value is 1 second, and the watchdog operating mode is the reset mode.

If the execution time of “other codes” is shorter than 1 second, the watchdog will restart from the interval value before it counts to zero and generates the reset signal. Otherwise, the watchdog will reset the whole system before the code “writel(readl(WDOG_CTRL) |(0xA57<<1) | (1<<0), WDOG_CTRL)” is executed.

```
writel (0x1, WDOG_CONFIG); //To whole system
writel (0x10, WDOG_MODE); //Interval Value set 1s
```

```
writel(readl(WDOG_MODE) | (1<<0), WDOG_MODE); //Enable Watchdog
```

----other codes---

```
writel(readl(WDOG_CTRL) |(0xA57<<1) | (1<<0), WDOG_CTRL); //Write 0xA57 at Key Field and Restart Watchdog
```

7.1.5 Register List

Module Name	Base Address
TIMER0	0x02050000

Register Name	Offset	Description
TMR_IRQ_EN_REG	0x0000	Timer IRQ Enable Register
TMR_IRQ_STA_REG	0x0004	Timer IRQ Status Register
TMR0_CTRL_REG	0x0010	TIMER0_0 Control Register
TMR0_INTV_VALUE_REG	0x0014	TIMER0_0 Interval Value Register
TMR0_CUR_VALUE_REG	0x0018	TIMER0_0 Current Value Register
TMR1_CTRL_REG	0x0020	TIMER0_1 Control Register
TMR1_INTV_VALUE_REG	0x0024	TIMER0_1 Interval Value Register
TMR1_CUR_VALUE_REG	0x0028	TIMER0_1 Current Value Register
WDOG_IRQ_EN_REG	0x00A0	Watchdog IRQ Enable Register
WDOG_IRQ_STA_REG	0x00A4	Watchdog Status Register
WDOG_SOFT_RST_REG	0x00A8	Watchdog Software Reset Register
WDOG_CTRL_REG	0x00B0	Watchdog Control Register
WDOG_CFG_REG	0x00B4	Watchdog Configuration Register
WDOG_MODE_REG	0x00B8	Watchdog Mode Register
WDOG_OUTPUT_CFG_REG	0x00BC	Watchdog Output Configuration Register
AVS_CNT_CTL_REG	0x00C0	AVS Counter Control Register
AVS_CNT0_REG	0x00C4	AVS Counter 0 Register
AVS_CNT1_REG	0x00C8	AVS Counter 1 Register
AVS_CNT_DIV_REG	0x00CC	AVS Counter Divisor Register

7.1.6 Register Description

7.1.6.1 0x0000 Timer IRQ Enable Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: TMR_IRQ_EN_REG
Bit	Read/Write	Default/Hex	Description
31:2	/	/	/
1	R/W	0x0	TMR1_IRQ_EN TIMER0_1 Interrupt Enable 0: Disable 1: Enable

Offset: 0x0000			Register Name: TMR_IRQ_EN_REG
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	TMR0_IRQ_EN TIMER0_0 Interrupt Enable 0: Disable 1: Enable

7.1.6.2 0x0004 Timer IRQ Status Register (Default Value: 0x0000_0000)

Offset: 0x0004			Register Name: TMR_IRQ_STA_REG
Bit	Read/Write	Default/Hex	Description
31:2	/	/	/
1	R/W1C	0x0	TMR1_IRQ_PEND TIMER0_1 IRQ Pending. Set 1 to the bit will clear it. 0: No effect 1: Pending, TIMER0_1 interval value is reached
0	R/W1C	0x0	TMR0_IRQ_PEND TIMER0_0 IRQ Pending. Set 1 to the bit will clear it. 0: No effect 1: Pending, TIMER0_0 interval value is reached

7.1.6.3 0x0010 TIMER0_0 Control Register (Default Value: 0x0000_0004)

Offset: 0x0010			Register Name: TMR0_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	TMR0_MODE TIMER0_0 mode 0: Continuous mode. When interval value reached, the timer will not disable automatically. 1: Single mode. When interval value reached, the timer will disable automatically.
6:4	R/W	0x0	TMR0_CLK_PRES Select the pre-scale of TIMER0_0 clock source 000: /1 001: /2 010: /4 011: /8 100: /16 101: /32 110: /64 111: /128

Offset: 0x0010			Register Name: TMR0_CTRL_REG
Bit	Read/Write	Default/Hex	Description
3:2	R/W	0x1	TMR0_CLK_SRC 00: CLK32K 01: HOSC 10: / 11: /
1	R/W	0x0	TMR0_RELOAD TIMER0_0 Reload 0: No effect 1: Reload TIMER0_0 Interval value After the bit is set, it cannot be written again before it's cleared automatically.
0	R/W	0x0	TMR0_EN TIMER0_0 Enable 0: Stop/Pause 1: Start If the timer is started, it will reload the interval value to internal register, and the current counter will count from interval value to 0. If the current counter does not reach the zero, the timer enable bit is set to "0", the current value counter will pause. At least wait for 2 cycles, the start bit can be set to 1. In timer pause state, the interval value register can be modified. If the timer is started again, and the Software hope the current value register to down-count from the new interval value, the reload bit and the enable bit should be set to 1 at the same time.

7.1.6.4 0x0014 TIMER0_0 Interval Value Register (Default Value: 0x0000_0000)

Offset: 0x0014			Register Name: TMR0_INTV_VALUE_REG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	TMR0_INTV_VALUE TIMER0_0 Interval Value.



NOTE

Take the system clock and timer clock source into consideration when setting the interval value.

7.1.6.5 0x0018 TIMER0_0 Current Value Register (Default Value: 0x0000_0000)

Offset: 0x0018			Register Name: TMR0_CUR_VALUE_REG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	TMR0_CUR_VALUE TIMER0_0 Current Value Timer0 current value is a 32-bit down-counter (from interval value to 0).

7.1.6.6 0x0020 TIMER0_1 Control Register (Default Value: 0x0000_0004)

Offset: 0x0020			Register Name: TMR1_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	TMR1_MODE TIMER0_1 mode 0: Continuous mode. When interval value reached, the timer will not disable automatically. 1: Single mode. When interval value reached, the timer will disable automatically.
6:4	R/W	0x0	TMR1_CLK_PRES Select the pre-scale of TIMER0_1 clock source 000: /1 001: /2 010: /4 011: /8 100: /16 101: /32 110: /64 111: /128
3:2	R/W	0x1	TMR1_CLK_SRC 00: CLK32K 01: HOSC 10: / 11: /
1	R/W	0x0	TMR1_RELOAD TIMER0_1 Reload 0: No effect 1: Reload TIMER0_1 Interval value After the bit is set, it cannot be written again before it's cleared automatically.

Offset: 0x0020			Register Name: TMR1_CTRL_REG
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	TMR1_EN TIMER0_1 Enable 0: Stop/Pause 1: Start If the timer is started, it will reload the interval value to internal register, and the current counter will count from interval value to 0. If the current counter does not reach the zero, the timer enable bit is set to "0", the current value counter will pause. At least wait for 2 cycles, the start bit can be set to 1. In timer pause state, the interval value register can be modified. If the timer is started again, and the Software hope the current value register to down-count from the new interval value, the reload bit and the enable bit should be set to 1 at the same time.

7.1.6.7 0x0024 TIMER0_1 Interval Value Register (Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: TMR1_INTV_VALUE_REG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	TMR1_INTV_VALUE TIMER0_1 Interval Value.



NOTE

Take the system clock and timer clock source into consideration when setting the interval value.

7.1.6.8 0x0028 TIMER0_1 Current Value Register (Default Value: 0x0000_0000)

Offset: 0x0028			Register Name: TMR1_CUR_VALUE_REG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	TMR1_CUR_VALUE TIMER0_1 Current Value Timer1 current value is a 32-bit down-counter (from interval value to 0).

7.1.6.9 0x00A0 Watchdog IRQ Enable Register (Default Value: 0x0000_0000)

Offset: 0x00A0			Register Name: WDOG_IRQ_EN_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	WDOG_IRQ_EN Watchdog Interrupt Enable 0: Disable 1: Enable.

7.1.6.10 0x00A4 Watchdog Status Register (Default Value: 0x0000_0000)

Offset: 0x00A4			Register Name: WDOG_IRQ_STA_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W1C	0x0	WDOG_IRQ_PEND Watchdog IRQ Pending. Set 1 to the bit will clear it 0: No effect 1: Pending, Watchdog interval value is reached

7.1.6.11 0x00A8 Watchdog Software Reset Register (Default Value: 0x0000_0000)

Offset: 0x00A8			Register Name: WDOG_SOFT_RST_REG
Bit	Read/Write	Default/Hex	Description
31:16	W	0x0	KEY_FIELD Key Field. This field should be filled with 0x16AA, and then the bit 0 can be written with the new value.
15:1	/	/	/
0	R/W1C	0x0	Soft Reset Enable 0: De-assert 1: Reset System To use the bit to reset the system, the watchdog first needs to be disabled.

7.1.6.12 0x00B0 Watchdog Control Register (Default Value: 0x0000_0000)

Offset: 0x00B0			Register Name: WDOG_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:13	/	/	/

Offset: 0x00B0			Register Name: WDOG_CTRL_REG
Bit	Read/Write	Default/Hex	Description
12:1	W	0x0	WDOG_KEY_FIELD Watchdog Key Field Should be written at value 0xA57. Writing any other value in this field aborts the write operation.
0	R/W1S	0x0	WDOG_RESTART Watchdog Restart 0: No effect 1: Restart the Watchdog 0

7.1.6.13 0x00B4 Watchdog Configuration Register (Default Value: 0x0000_0001)

Offset: 0x00B4			Register Name: WDOG_CFG_REG
Bit	Read/Write	Default/Hex	Description
31:16	W	0x0	KEY_FIELD Key Field. This field should be filled with 0x16AA, and then the bit 15:0 can be written with the new value.
15:9	/	/	/
8	R/W	0x0	WDOG_CLK_SRC Select Watchdog clock sources 0: HOSC_32K 1: LOSC_32K
7:2	/	/	/
1:0	R/W	0x1	WDOG_CONFIG 00: / 01: To whole system 10: Only interrupt 11: /

7.1.6.14 0x00B8 Watchdog Mode Register (Default Value: 0x0000_0000)

Offset: 0x00B8			Register Name: WDOG_MODE_REG
Bit	Read/Write	Default/Hex	Description
31:16	W	0x0	KEY_FIELD Key Field. This field should be filled with 0x16AA, and then the bit 15:0 can be written with the new value.
15:8	/	/	/

Offset: 0x00B8			Register Name: WDOG_MODE_REG
Bit	Read/Write	Default/Hex	Description
7:4	R/W	0x0	WDOG_INTV_VALUE Watchdog Interval Value Watchdog Interval Value. When the clock source is HOSC_32K, the time of 16000 cycles is equal to 0.5 sec, and when the clock source is 32K, the time of 16000 cycles is based on the number of 32K. 0000: 16000 cycles (0.5s) 0001: 32000 cycles (1s) 0010: 64000 cycles (2s) 0011: 96000 cycles (3s) 0100: 128000 cycles (4s) 0101: 160000 cycles (5s) 0110: 192000 cycles (6s) 0111: 256000 cycles (8s) 1000: 320000 cycles (10s) 1001: 384000 cycles (12s) 1010: 448000 cycles (14s) 1011: 512000 cycles (16s) 1100: / 1101: / 1110: / 1111: /
3:1	/	/	/
0	R/W	0x0	WDOG_EN Watchdog Enable 0: No effect 1: Enable the Watchdog.

7.1.6.15 0x00BC Watchdog Output Configuration Register (Default Value: 0x0000_001F)

Offset: 0x00BC			Register Name: WDOG_OUTPUT_CFG_REG
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R/W	0x1F	WDOG OUTPUT CONFIG Wdog rst valid time config $T=1/32ms*(N+1)$ Default 1ms

7.1.6.16 0x00C0 AVS Counter Control Register (Default Value: 0x0000_0000)

Offset: 0x00C0	Register Name: AVS_CNT_CTL_REG

Bit	Read/Write	Default/Hex	Description
31:10	/	/	/
9	R/W	0x0	AVS_CNT1_PS Audio/Video Sync Counter 1 Pause Control 0: Not pause 1: Pause Counter 1
8	R/W	0x0	AVS_CNT0_PS Audio/Video Sync Counter 0 Pause Control 0: Not pause 1: Pause Counter 0
7:2	/	/	/
1	R/W	0x0	AVS_CNT1_EN Audio/Video Sync Counter 1 Enable/ Disable. The counter source is HOSC. 0: Disable 1: Enable
0	R/W	0x0	AVS_CNT0_EN Audio/Video Sync Counter 0 Enable/ Disable. The counter source is HOSC. 0: Disable 1: Enable

7.1.6.17 0x00C4 AVS Counter 0 Register (Default Value: 0x0000_0000)

Offset: 0x00C4			Register Name: AVS_CNT0_REG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	AVS_CNT0 Counter 0 for Audio/ Video Sync Application The high 32 bits of the internal 33-bits counter register. The initial value of the internal 33-bits counter register can be set by software. The LSB bit of the 33-bits counter register should be zero when the initial value is updated. It will count from the initial value. The initial value can be updated at any time. It can also be paused by setting AVS_CNT0_PS to '1'. When it is paused, the counter won't increase.

7.1.6.18 0x00C8 AVS Counter 1 Register (Default Value: 0x0000_0000)

Offset: 0x00C8			Register Name: AVS_CNT1_REG
Bit	Read/Write	Default/Hex	Description

Offset: 0x00C8			Register Name: AVS_CNT1_REG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	AVS_CNT1 Counter 1 for Audio/ Video Sync Application The high 32 bits of the internal 33-bits counter register. The initial value of the internal 33-bits counter register can be set by software. The LSB bit of the 33-bits counter register should be zero when the initial value is updated. It will count from the initial value. The initial value can be updated at any time. It can also be paused by setting AVS_CNT1_PS to '1'. When it is paused, the counter won't increase.

7.1.6.19 0x00CC AVS Counter Divisor Register (Default Value: 0x05DB_05DB)

Offset: 0x00CC			Register Name: AVS_CNT_DIV_REG
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0x5DB	AVS_CNT1_D Divisor N for AVS Counter 1 AVS CN1 CLK=24MHz/Divisor_N1 Divisor N1 = Bit [27:16] + 1 The number N is from 1 to 0x7ff. The zero value is reserved. The internal 33-bits counter engine will maintain another 12-bits counter. The 12-bits counter is used for counting the cycle number of one 24Mhz clock. When the 12-bits counter reaches ($\geq N$) the divisor value, the internal 33-bits counter register will increase 1 and the 12-bits counter will reset to zero and restart again. Note: It can be configured by software at any time.
15:12	/	/	/

Offset: 0x00CC			Register Name: AVS_CNT_DIV_REG
Bit	Read/Write	Default/Hex	Description
11:0	R/W	0x5DB	AVS_CNT0_D Divisor N for AVS Counter 0 AVS CN0 CLK=24MHz/Divisor_N0 Divisor N0 = Bit [11:0] + 1 The number N is from 1 to 0x7ff. The zero value is reserved. The internal 33-bits counter engine will maintain another 12-bits counter. The 12-bits counter is used for counting the cycle number of one 24Mhz clock. When the 12-bits counter reaches ($\geq$ N) the divisor value, the internal 33-bits counter register will increase 1 and the 12-bits counter will reset to zero and restart again. Note: It can be configured by software at any time.

7.2 High Speed Timer (HSTimer)

7.2.1 Overview

The high speed timer (HSTimer) module implements more precise timing and counting functions.

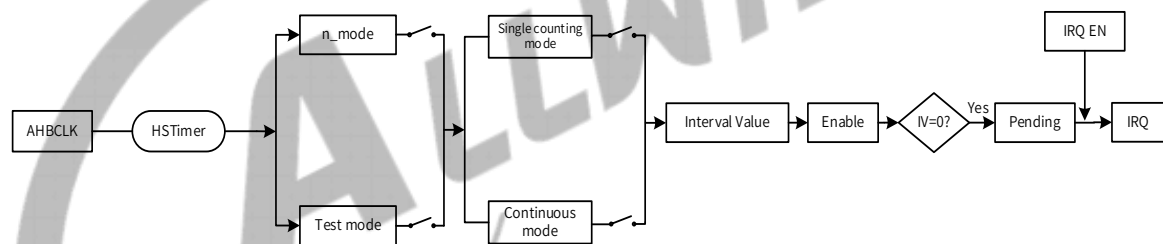
The HSTimer has the following features:

- Single clock source: AHB0
- Supports 5 prescale factors
- Configurable 56-bit down timer
- Supports 2 timing modes: periodic mode and single counting mode
- Supports the test mode
- Generates an interrupt when the count is decreased to 0

7.2.2 Block Diagram

The following figure shows the block diagram of the HSTimer.

Figure 7-5 HSTimer Block Diagram



- 1) n_mode: normal mode. Refer to register [HS_TMR0_CTRL_REG](#).
- 2) IV: Timer current value. Refer to registers of [HS_TMR0_CURNT_LO_REG](#), [HS_TMR0_CURNT_HI_REG](#), [HS_TMR1_CURNT_LO_REG](#), and [HS_TMR1_CURNT_HI_REG](#).

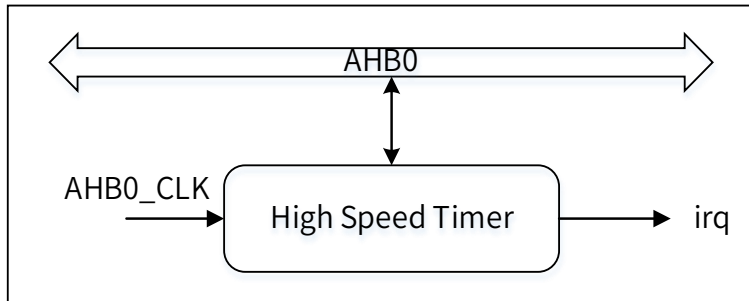
7.2.3 Functional Description

The HSTimers are 56-bit down counters. The counter value is decremented by 1 on each rising edge of the count clock. Each HSTimer has a prescaler that divides the working clock frequency of each working timer by 1, 2, 4, 8, or 16.

7.2.3.1 Typical Application

The following figure shows a typical application of HSTimer module.

Figure 7-6 Typical Application for HSTimer



The HSTimer module is mounted at AHB0, and can control the registers via AHB0. AHB0 is the clock source of the HSTimer. When the count value reaches zero, the HSTimer generates an interrupt.

7.2.3.2 Count Modes

The HSTimer has two count modes: one-shot mode and periodic mode. You can configure the timing mode via the bit [7] of HS_TMRn_CTRL_REG (n = 0 or 1). The value 0 is for the period mode and value 1 is for the one-shot mode.

- Single Counting Mode

When the count value of the HSTimer reaches 0, the HSTimer stops counting. The HSTimer starts to count again only when the initial value is reloaded.

- Periodic Mode

The HSTimer counts continuously. When the count value of the HSTimer reaches 0, the HSTimer reloads an initial value from HS_TMRn_INTV_LO_REG and HS_TMRn_INTV_HI_REG and then continues to count.

7.2.3.3 Operating Modes

The HSTimer has two operating modes: the normal mode and test mode. You can configure the operating mode via the bit [31] of HS_TMRn_CTRL_REG. The value 0 is for the normal mode and value 1 is for the test mode.

- Normal Mode

In the normal mode, the HSTimer is used as a 56-bit down counter, which can finish one-shot counting and periodic counting. The interval value for the HSTimer consists of two parts: HS_TMRn_INTV_LO_REG forms the bit [31:0] and HS_TMRn_INTV_HI_REG forms the bit [55:32]. To read or write the interval value, HS_TMRn_INTV_LO_REG should be done before HS_TMRn_INTV_HI_REG.

- Test Mode

In the test mode, the HSTimer is used as a 24-bit down counter. HS_TMRn_INTV_LO_REG must be set to 0x1, and HS_TMRn_INTV_HI_REG acts as the initial value for the HSTimer.

7.2.3.4 HSTimer Formula

The following formula describes the relationship among HSTimer parameters in the normal mode.

$$T_{HSTimer} = \frac{(HS\_TMRn\_INTV\_HI\_REG \ll 32 + HS\_TMRn\_INTV\_LO\_REG) - (HS\_TMRn\_CURNT\_HI\_REG \ll 32 + HS\_TMRn\_CURNT\_LO\_REG)}{AHB0CLK} \times HS\_TMRn\_CLK$$

Where,

The parameter n is either 0 or 1;

$T_{HSTimer}$ is the elapsed time of the timer;

HS_TMRn_INTV_HI_REG is bit [55:32] of the HSTimer interval value;

HS_TMRn_INTV_LO_REG is bit [31:0] of the HSTimer interval value;

HS_TMRn_CURNT_HI_REG is bit [55:32] of the HSTimer current value;

HS_TMRn_CURNT_LO_REG is bit [31:0] of the HSTimer current value;

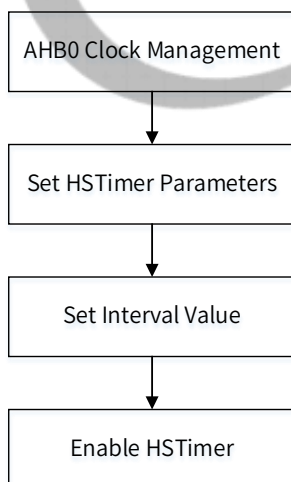
AHB0CLK is the frequency of AHB0 Clock (the HSTimer clock source);

HS_TMRn_CLK is the prescale ratio of the HSTimer clock.

7.2.3.5 Initializing the HSTimer

The following figure shows the process of HSTimer initialization.

Figure 7-7 HSTimer Initialization Process



Step 1 AHB0 clock management: Open the clock gate of AHB0 and de-assert the soft reset of AHB0 in 4.1 Clock Controller Unit (CCU).

- Step 2** Configure the corresponding parameters of the HSTimer: clock source, prescaler factor, operating mode, and timing mode. There is no sequence requirement when writing the above parameters to HS_TMRn_CTRL_REG.
- Step 3** Write the initial value: Write the lower 32 bits of the initial value to HS_TMRn_INTV_LO_REG first, and then the higher 24 bits to HS_TMRn_INTV_HI_REG. Normally, write the bit [1] of HS_TMRn_CTRL_REG to load the initial value. If the HSTimer is in the timing stop stage, write 1 to the bit [1] and bit [0] of HS_TMRn_CTRL_REG at the same time to reload the initial value.
- Step 4** Enable HSTimer: Write the bit [0] of HS_TMRn_CTRL_REG to enable HSTimer to count.
- Step 5** Reading HS_TMRn_CURNT_LO_REG and HS_TMRn_CURNT_HI_REG can get current counting value.

7.2.3.6 Processing the HSTimer Interrupt

Follow the steps below to process the HSTimer interrupt:

- Step 1** Enable interrupt: Write the corresponding interrupt enable bit of [HS_TMR_IRQ_EN_REG](#), when the counting time of HSTimer reaches, the corresponding interrupt generates.
- Step 2** After entering the interrupt process, write [HS_TMR_IRQ_STAS_REG](#) to clear the interrupt pending.
- Step 3** Resume the interrupt and continue to execute the interrupted process.

7.2.4 Programming Guidelines

The following example shows how to make a 1 us delay with HSTimer0. The frequency of the AHB0 clock is 100 MHz, the operating mode is the normal mode, the timing mode is single counting mode, and the pre-scale is 2.

```
writel(0x32, HS_TMR0_INTV_LO);           //Set bit [31:0] of the interval value as 0x32.
writel(0x0, HS_TMR0_INTV_HI);           //Set bit [55:32] of the interval value as 0x0.
writel(0x90, HS_TMR0_CTRL);

//Set the operating mode as Normal Mode, the pre-scale as 2, and the timing mode as Single
Counting Mode.

writel(readl(HS_TMR0_CTRL) | (1<<1), HS_TMR0_CTRL); //Set the reload bit.
writel(readl(HS_TMR0_CTRL) | (1<<0), HS_TMR0_CTRL); //Enable HSTimer0.
while (! (readl(HS_TMR_IRQ_STAS) &1));           //Wait for HSTimer0 to generate
pending.
writel(1, HS_TMR_IRQ_STAS);                 //Clear HSTimer0 pending.
```

7.2.5 Register List

Module Name	Base Address
HSTimer	0x03008000

Register Name	Offset	Description
HS_TMR_IRQ_EN_REG	0x0000	HS Timer IRQ Enable Register
HS_TMR_IRQ_STAS_REG	0x0004	HS Timer IRQ Status Register
HS_TMR0_CTRL_REG	0x0020	HS Timer0 Control Register
HS_TMR0_INTV_LO_REG	0x0024	HS Timer0 Interval Value Low Register
HS_TMR0_INTV_HI_REG	0x0028	HS Timer0 Interval Value High Register
HS_TMR0_CURNT_LO_REG	0x002C	HS Timer0 Current Value Low Register
HS_TMR0_CURNT_HI_REG	0x0030	HS Timer0 Current Value High Register
HS_TMR1_CTRL_REG	0x0040	HS Timer1 Control Register
HS_TMR1_INTV_LO_REG	0x0044	HS Timer1 Interval Value Low Register
HS_TMR1_INTV_HI_REG	0x0048	HS Timer1 Interval Value High Register
HS_TMR1_CURNT_LO_REG	0x004C	HS Timer1 Current Value Low Register
HS_TMR1_CURNT_HI_REG	0x0050	HS Timer1 Current Value High Register

7.2.6 Register Description

7.2.6.1 0x0000 HS Timer IRQ Enable Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: HS_TMR_IRQ_EN_REG
Bit	Read/Write	Default/Hex	Description
31:2	/	/	/
1	R/W	0x0	HS_TMR1_INT_EN HSTimer1 Interrupt Enable 0: Disabled 1: Enabled
0	R/W	0x0	HS_TMR0_INT_EN HSTimer0 Interrupt Enable 0: Disabled 1: Enabled

7.2.6.2 0x0004 HS Timer IRQ Status Register (Default Value: 0x0000_0000)

Offset: 0x0004	Register Name: HS_TMR_IRQ_STAS_REG

Bit	Read/Write	Default/Hex	Description
31:2	/	/	/
1	R/W1C	0x0	HS_TMR1_IRQ_PEND HSTimer1 IRQ Pending The IRQ pending bit for HSTimer1. Write 1 to clear the pending status. 0: No effect 1: Pending, indicates that the initial value of the HSTimer is reached.
0	R/W1C	0x0	HS_TMR0_IRQ_PEND HSTimer0 IRQ Pending The IRQ pending bit for HSTimer0. Write 1 to clear the pending status. 0: No effect 1: Pending, indicates that the initial value of the HSTimer is reached.

7.2.6.3 0x0020 HS Timer0 Control Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: HS_TMR0_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	HS_TMR0_TEST Select the operating mode for HSTimer0 0: Normal mode 1: Test mode In the test mode, the HS_TMR0_INTV_LO_REG must be set to 0x1, and HS_TMR0_INTV_HI_REG acts as the initial value for HSTimer0.
30:8	/	/	/
7	R/W	0x0	HS_TMR0_MODE Select the timing mode for HSTimer0 0: Periodic mode. When the count value is decreased to 0, the timer will restart another round of counting automatically. 1: One-shot mode. When the count value is decreased to 0, the timer will stop counting.

Offset: 0x0020			Register Name: HS_TMR0_CTRL_REG
Bit	Read/Write	Default/Hex	Description
6:4	R/W	0x0	HS_TMR0_CLK Select the pre-scale for the HSTimer0 clock sources 000: /1 001: /2 010: /4 011: /8 100: /16 101: / 110: / 111: /
3:2	/	/	/
1	R/W1S	0x0	HS_TMR0_RELOAD HSTimer0 Reload 0: No effect 1: Reload the interval value of the HSTimer0
0	R/W	0x0	HS_TMR0_EN HSTimer0 Enable 0: Stop/Pause 1: Start By setting the bit to 1, the timer will be started. It reloads the interval value register and then counts from the interval value to 0. By setting the bit to 0 before the timer counts to 0, the timer will be paused. The bit will be locked to 0 for at least 2 cycles. Within the 2 cycles, you cannot set the bit to 1 to restart the timer. The timer supports updating the interval value in the pause state. To start from the updated interval value, set both the reload bit and enable bit to 1. Additionally, in the one-shot mode, after the count value reaches 0, the system will automatically change the bit to 0 to stop the timer.

7.2.6.4 0x0024 HS Timer0 Interval Value Low Register (Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: HS_TMR0_INTV_LO_REG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	HS_TMR0_INTV_VALUE_LO Bit[31:0] of the HSTimer0 interval value.

7.2.6.5 0x0028 HS Timer0 Interval Value High Register (Default Value: 0x0000_0000)

Offset: 0x0028			Register Name: HS_TMR0_INTV_HI_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:0	R/W	0x0	HS_TMR0_INTV_VALUE_HI Bit[55:32] of the HSTimer0 interval value.



NOTE

HSTimer0 is a 56-bit counter. The interval value consists of two parts: HS_TMR0_INTV_VALUE_LO acts as the bit [31:0] and HS_TMR0_INTV_VALUE_HI acts as the bit [55:32]. To read or write the interval value, HS_TMR0_INTV_LO_REG should be done before HS_TMR0_INTV_HI_REG.

7.2.6.6 0x002C HS Timer0 Current Value Low Register (Default Value: 0x0000_0000)

Offset: 0x002C			Register Name: HS_TMR0_CURNT_LO_REG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	HS_TMR0_CUR_VALUE_LO Bit[31:0] of the HSTimer0 current value.

7.2.6.7 0x0030 HS Timer0 Current Value High Register (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: HS_TMR0_CURNT_HI_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:0	R/W	0x0	HS_TMR0_CUR_VALUE_HI Bit[55:32] of the HSTimer0 current value.



NOTE

HSTimer0 is a 56-bit counter. The current value consists of two parts: HS_TMR0_CUR_VALUE_LO acts as the bit [31:0] and HS_TMR0_CUR_VALUE_HI acts as the bit [55:32]. To read or write the current value, HS_TMR0_CUR_VALUE_LO should be done before HS_TMR0_CUR_VALUE_HI.

7.2.6.8 0x0040 HS Timer1 Control Register (Default Value: 0x0000_0000)

Offset: 0x0040			Register Name: HS_TMR1_CTRL_REG
Bit	Read/Write	Default/Hex	Description

Offset: 0x0040			Register Name: HS_TMR1_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	HS_TMR1_TEST Select the operating mode for HSTimer1 0: Normal mode 1: Test mode In the test mode, the HS_TMR1_INTV_LO_REG must be set to 0x1, and HS_TMR1_INTV_HI_REG acts as the interval value for HSTimer1.
30:8	/	/	/
7	R/W	0x0	HS_TMR1_MODE Select the timing mode for HSTimer1 0: Periodic mode. When the count value is decreased to 0, the timer will restart another round of counting automatically. 1: One-shot mode. When the count value is decreased to 0, the timer will stop counting.
6:4	R/W	0x0	HS_TMR1_CLK Select the pre-scale of the HSTimer1 clock sources 000: /1 001: /2 010: /4 011: /8 100: /16 101: / 110: / 111: /
3:2	/	/	/
1	R/W1S	0x0	HS_TMR1_RELOAD HSTimer1 Reload 0: No effect 1: Reload the HSTimer1 interval value.

Offset: 0x0040			Register Name: HS_TMR1_CTRL_REG
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	HS_TMR1_EN HSTimer1 Enable 0: Stop/Pause 1: Start By setting the bit to 1, the timer will be started. It reloads the interval value register and then counts from the interval value to 0. By setting the bit to 0 before the timer counts to 0, the timer will be paused. The bit will be locked to 0 for at least 2 cycles. Within the 2 cycles, you cannot set the bit to 1 to restart the timer. The timer supports updating the interval value in the pause state. To start from the updated interval value, set both the reload bit and enable bit to 1. Additionally, in the one-shot mode, after the count value reaches 0, the system will automatically change the bit to 0 to stop the timer.

7.2.6.9 0x0044 HS Timer1 Interval Value Low Register (Default Value: 0x0000_0000)

Offset: 0x0044			Register Name: HS_TMR1_INTV_LO_REG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	HS_TMR1_INTV_VALUE_LO Bit[31:0] of the HSTimer1 interval value

7.2.6.10 0x0048 HS Timer1 Interval Value High Register (Default Value: 0x0000_0000)

Offset: 0x0048			Register Name: HS_TMR1_INTV_HI_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:0	R/W	0x0	HS_TMR1_INTV_VALUE_HI Bit[55:32] of the HSTimer1 interval value



NOTE

HSTimer1 is a 56-bit counter. The interval value consists of two parts: HS_TMR1_INTV_VALUE_LO acts as the bit [31:0] and HS_TMR1_INTV_VALUE_HI acts as the bit [55:32]. To read or write the interval value, HS_TMR1_INTV_LO_REG should be done before HS_TMR1_INTV_HI_REG.

7.2.6.11 0x004C HS Timer1 Current Value Low Register (Default Value: 0x0000_0000)

Offset: 0x004C			Register Name: HS_TMR1_CURNT_LO_REG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	HS_TMR1_CUR_VALUE_LO Bit[31:0] of the HSTimer1 current value

7.2.6.12 0x0050 HS Timer1 Current Value High Register (Default Value: 0x0000_0000)

Offset: 0x0050			Register Name: HS_TMR1_CURNT_HI_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:0	R/W	0x0	HS_TMR1_CUR_VALUE_HI Bit[55:32] of the HSTimer1 current value



NOTE

HSTimer1 is a 56-bit counter. The current value consists of two parts: HS_TMR1_CUR_VALUE_LO acts as the bit [31:0] and HS_TMR1_CUR_VALUE_HI acts as the bit [55:32]. To read or write the current value, HS_TMR1_CUR_VALUE_LO should be done before HS_TMR1_CUR_VALUE_HI.

7.3 RISC-V Watchdog

7.3.1 Overview

RISC-V Watchdog is used to monitor whether the RISC-V system is abnormal. If yes, a system reset signal will be triggered.

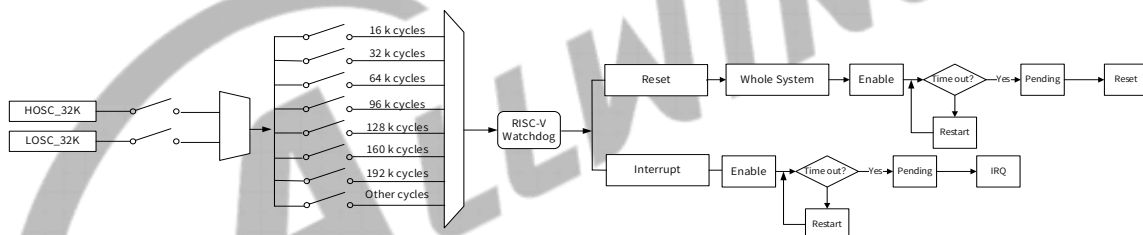
The watchdog has the following features:

- Clock sources: HOSC_32K (32.768 kHz) and LOSC_32K (32 kHz)
- 12 configurable initial count value
- Generates timeout interrupts
- Generates reset signals
- Supports restarting the timing

7.3.2 Block Diagram

The following figure shows the block diagram for the watchdog.

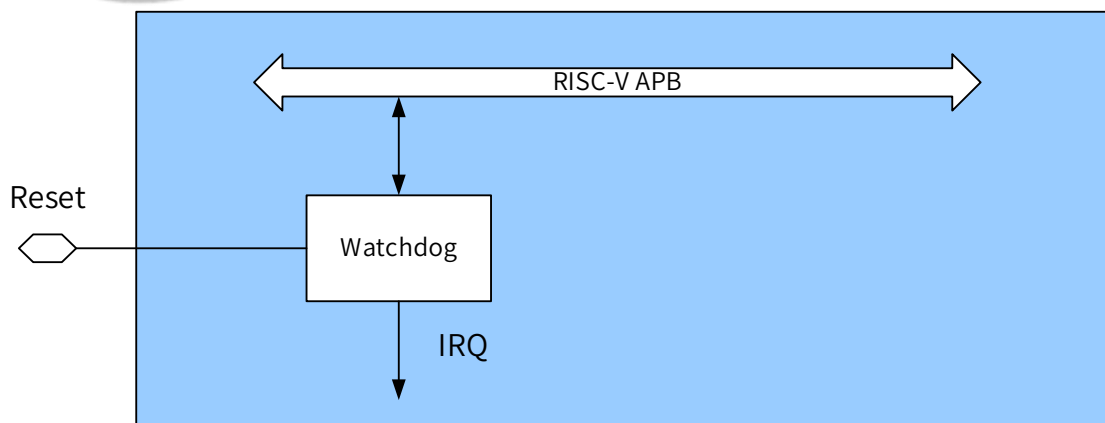
Figure 7-8 Block Diagram for the RISC-V Watchdog



7.3.3 Functional Description

7.3.3.1 Typical Application

Figure 7-9 RISC-V Watchdog Application Diagram



Watchdog configures register by APB bus.

The system configures the time of watchdog, if the system has no timing for restart watchdog (such as bus hang dead), then watchdog sends out watchdog reset external signal to reset system. And the signal will be transmitted to the Reset pad to reset the PMIC.

7.3.3.2 Operating Modes

The watchdog is a 32-bit down counter. The counter value is decreased by 1 on each rising edge of the count clock. The watchdog has two operating modes.

- **Interrupt mode:** The bit [1:0] of the [RWDG_CFG_REG](#) should be set to 0x2 first. When the bit [0] of [RWDG_IRQ_EN_REG](#) is down to 0, the watchdog generates an interrupt and the watchdog enters into interrupt mode. Write “1” can clear the interrupt signal.
- **Reset mode:** The bit [1:0] of the [RWDG_CFG_REG](#) is set to 0x1, when the counter value down to 0, the watchdog generates a reset signal to reset the whole system. There are 12 configurable initial count values.

Both the interrupt mode and reset mode support Watchdog Restart. You can make the watchdog to count from the initial value at any time by configuring the [RWDG_CTRL_REG](#): Write 0xA57 to bit[12:1], then write 1 to bit[0].

7.3.3.3 Initializing the Watchdog

Follow the steps below to initialize the watchdog:

- Step 1** Write the bit[1:0] of [RWDG_CFG_REG](#) to configure the watchdog operating mode so that the watchdog can generate interrupts or output reset signals.
- Step 2** Write the bit[7:4] of [RWDG_MODE_REG](#) to configure the initial count value.
- Step 3** Write the bit[0] of [RWDG_MODE_REG](#) to enable the watchdog.

7.3.3.4 Interrupt Processing

In the interrupt mode, the watchdog is used as a counter.

Follow the steps below to process the interrupt:

- Step 1** Write the enable bit of [RWDG_IRQ_EN_REG](#) to enable the interrupt.
- Step 2** After entering the interrupt process, write the pending bit of [RWDG_IRQ_STAT_REG](#) to clear the interrupt pending and execute the process of waiting for the interrupt.
- Step 3** Resume the interrupt and continue to execute the interrupted process.

7.3.4 Programming Guidelines

7.3.4.1 Watchdog Restart

- Step 1** Configure [RWDOG_CFG_REG](#) [1:0] to select reset mode.
- Step 2** Configure [RWDOG_MODE_REG](#) [7:4] to set watchdog interval value.
- Step 3** Write 1 to [RWDOG_MODE_REG](#) [0] to enable watchdog.
- Step 4** Write 1 to [RWDOG_CTRL_REG](#) [0] to restart watchdog.



NOTE

Before performing a writing operation to the [RWDOG_CTRL_REG](#), [RWDOG_CFG_REG](#), and [RWDOG_MODE_REG](#) register, you need to write a key to the corresponding register:

[RWDOG_CTRL_REG](#) [12:1]=0xA57;

[RWDOG_CFG_REG](#) [31:16]=0x16AA;

[RWDOG_MODE_REG](#) [31:16]=0x16AA.

7.3.4.2 Watchdog Reset

- Step 1** Configure [RWDOG_CFG_REG](#) [1:0] to select reset mode.
- Step 2** Configure [RWDOG_MODE_REG](#) [7:4] to set watchdog interval value.
- Step 3** Write 1 to [RWDOG_MODE_REG](#) [0] to enable watchdog.



NOTE

After the reset signal is triggered, WDG registers will not be reset. If you need to use the watchdog again, it is recommended to write 0 to [RWDOG_CTRL_REG](#) [0] and wait 3 APB clock cycles before writing 1 to it.

7.3.5 Register List

Module Name	Base Address
RISCV_WDG	0x06011000

Register Name	Offset	Description
RWDOG_IRQ_EN_REG	0x0000	RISCV Watchdog IRQ Enable Register
RWDOG_IRQ_STA_REG	0x0004	RISCV Watchdog Status Register
RWDOG_CTRL_REG	0x0010	RISCV Watchdog Control Register
RWDOG_CFG_REG	0x0014	RISCV Watchdog Configuration Register

Register Name	Offset	Description
RWDOG_MODE_REG	0x0018	RISCV Watchdog Mode Register
RWDOG_OUTPUT_CFG_REG	0x001C	RISCV Watchdog Output Configuration Register

7.3.6 Register Description

7.3.6.1 0x0000 RISCV Watchdog IRQ Enable Register (Default Value: 0x0000_0000)

Offset:0x0000			Register Name: RWDOG_IRQ_EN_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	RWDOG_IRQ_EN RISCV Watchdog Interrupt Enable 0: RISCV Watchdog interrupt mask 1: RISCV Watchdog interrupt enable

7.3.6.2 0x0004 RISCV Watchdog Status Register (Default Value: 0x0000_0000)

Offset:0x0004			Register Name: RWDOG_IRQ_STA_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	RWDOG_IRQ_PEND RISCV Watchdog IRQ Pending. Set 1 to the bit will clear it. 0: No effect 1: Pending, RISCV Watchdog interval value is reached

7.3.6.3 0x0010 RISCV Watchdog Control Register (Default Value: 0x0000_0000)

Offset:0x0010			Register Name: RWDOG_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:13	/	/	/
12:1	W	0x0	RWDOG_KEY_FIELD RISCV Watchdog Key Field Should be written at value 0xA57. Writing any other value in this field aborts the write operation.
0	R/W	0x0	RWDOG_RESTART RISCV Watchdog Restart 0: No effect 1: Restart the RISCV Watchdog

7.3.6.4 0x0014 RISC-V Watchdog Configuration Register (Default Value: 0x0000_0001)

Offset:0x0014			Register Name: RWDOG_CFG_REG
Bit	Read/Write	Default/Hex	Description
31:16	W	0x0	KEY_FIELD Key Field. This field should be filled with 0x16AA, and then the bit 0 can be written with the new value.
15:7	/	/	/
8	R/W	0x0	RWDOG_CLK_SRC Select RISC-V Watchdog clock sources. 0: HOSC_32K 1: LOSC_32K
7:2	/	/	/
1:0	R/W	0x1	RWDOG_CONFIG RISC-V Watchdog generates a reset signal 00: only to CPU core 01: to whole system 10: only interrupt 11: /

7.3.6.5 0x0018 RISC-V Watchdog Mode Register (Default Value: 0x0000_0000)

Offset:0x0018			Register Name: RWDOG_MODE_REG
Bit	Read/Write	Default/Hex	Description
31:16	W	0x0	KEY_FIELD Key Field This field should be filled with 0x16AA, and then the bit 0 can be written with the new value.
15:8	/	/	/

Offset: 0x0018			Register Name: RWDOG_MODE_REG
Bit	Read/Write	Default/Hex	Description
7:4	R/W	0x0	RWDOG_INTV_VALUE RISCv Watchdog Interval Value When the clock source is HOSC_32K, the time of 16000 cycles is equal to 0.5 sec, and when the clock source is 32K, the time of 16000 cycles is based on the number of 32K. 0000: 16000 cycles 0001: 32000 cycles 0010: 64000 cycles 0011: 96000 cycles 0100: 128000 cycles 0101: 160000 cycles 0110: 192000 cycles 0111: 256000 cycles 1000: 320000 cycles 1001: 384000 cycles 1010: 448000 cycles 1011: 512000 cycles 1100: / 1101: / 1110: / 1111: /
3:1	/	/	/
0	R/W	0x0	RWDOG_EN RISCv Watchdog Enable 0: No effect 1: Enable the RISCv Watchdog

7.3.6.6 0x001C RISCv Watchdog Output Configuration Register (Default Value: 0x0000_001F)

Offset: 0x001C			Register Name: RWDOG_OUTPUT_CFG_REG
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R/W	0x1F	RWDOG_O_CFG RISCv Watchdog Output Configuration $T = 1/32\text{ms} * (N+1)$ Default 1ms

8 RTC

8.1 RTC

8.1.1 Overview

The Real Time Clock (RTC) is used to implement the time counter and the timing wakeup functions. The RTC can display the year, month, day, week, hour, minute, second in real time. With the independent power, the RTC can still work when system powers off.

The RTC has the following features:

- Embedded with a 16-bit day counter, a 5-bit hour counter, a 6-bit minute counter, and a 6-bit second counter
- 1 kHz counting frequency
- Configurable initial value by software anytime
- Supports generating interrupt and waking up the external devices with the timing alarms function
- Multiple special registers for recording the BROM information



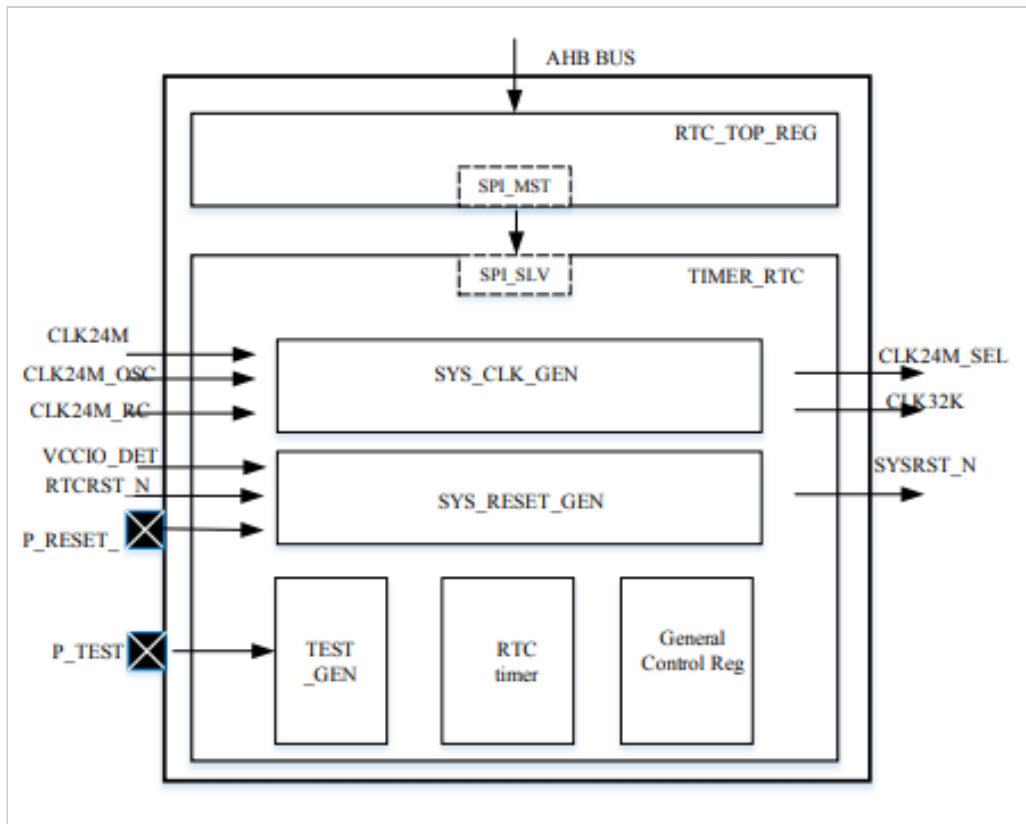
NOTE

The register configuration of RTC is AHB bus. It can only support word operation, not byte operation and half-word operation.

8.1.2 Block Diagram

The following figure shows the block diagram of the RTC.

Figure 8-1 RTC Block Diagram



8.1.3 External Signals

Table 8-1 RTC External Signals

Signal Name	Description	Type
DXIN	Digital Compensated Crystal Oscillator Input	AI
DXOUT	Digital Compensated Crystal Oscillator Output	AO
RESET	Reset Signal (Low Active)	I/O, OD

8.1.4 Functional Descriptions

8.1.4.1 RTC Clock Tree

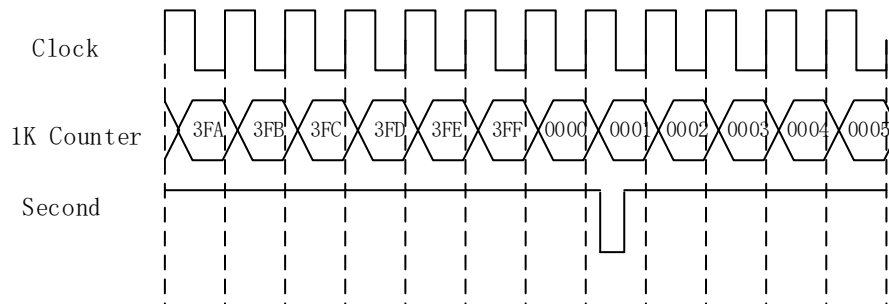
RTC

The clock sources of RTC can be selected by related switches, including 32K divided from internal 16 MHz RC, and high-speed crystal resonator.

The internal RC will be selected by default. After the system starts up, it is recommended to select high-speed crystal resonator as a clock source to provide more accurate clock.

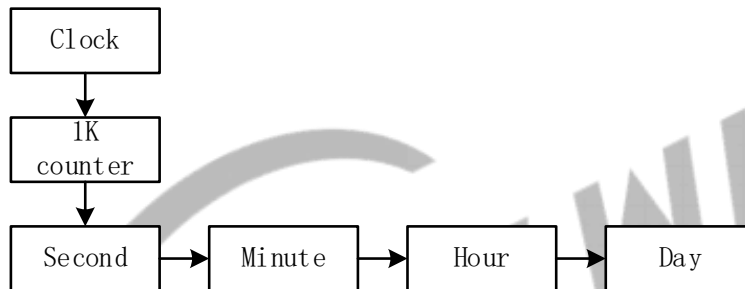
8.1.4.2 Real Time Clock

Figure 8-2 RTC Counter



The 1 kHz counter adds 1 on each rising edge of the clock. When the sampling clock number reaches 0x3FF, 1 kHz counter restarts to count from 0, and the second counter adds 1. The step structure of 1 kHz counter is as follows.

Figure 8-3 RTC 1 kHz Counter Step Structure



According to above implementation, the changing range of each counter is as follows.

Table 8-2 RTC Counter Changing Range

Counter	Range
Second	0 to 59
Minute	0 to 59
Hour	0 to 23
Day	0 to 65535 (The year, month, day need be transformed by software according to day counter)



CAUTION

As there is no error correction mechanism in the hardware, each counter configuration should not exceed a reasonable counting range.

8.1.4.3 Alarm 0

The principle of alarm0 is a comparator. When RTC timer reaches scheduled time, the RTC generates the interrupt, or outputs low level signal by NMI pin to wakeup power management chip.

The RTC only generates one interrupt when RTC timer reached the scheduled day, hour, minute and second counter, then the RTC need set a new scheduled time, the next interrupt can be generated.

8.1.4.4 RC Calibration Usage Scenario

Power-on: Select non-accurate 32K from internal RC.

Normal scenario: Select 32K from high-speed crystal resonator.

Standby scenario: Select 32K from high-speed crystal resonator.

8.1.5 Programming Guidelines

8.1.5.1 RTC Calendar

Step 1 Write time initial value: Write the current time to [RTC_DAY_REG](#) and [RTC_HH_MM_SS_REG](#).

Step 2 After updated time, the RTC restarts to count again. The software can read the current time anytime.



NOTE

- The RTC can only provide day counter, so the current day counter need be converted to year, month, day and week by software.
- Ensure the bit [8:7] of [LOSC_CTRL_REG](#) is 0 before the next time configuration is performed.

Here are the basic code samples.

For example: set time to 21st, 07:08:09 and read it.

```
RTC_DAY_REG = 0x00000015;
```

```
RTC_HH_MM_SS_REG = 0x00070809; //0000 0000 000|0 0000(Hour) 00|00 0000(Minute) 00|00 0000(Second)
```

```
Read (RTC_DAY_REG);
```

```
Read (RTC_HH_MM_SS_REG);
```

8.1.5.2 Alarm0 Setting

- Step 1** Enable alarm0 interrupt by writing 1 to [ALARM0_IRQ_EN](#).
- Step 2** Write the count-down day, hour, minute, second number to [ALARM0_DAY_SET_REG](#), [RTC_HH_MM_SS_REG](#), and [ALARM0_CONFIG_REG](#).
- Step 3** Enable alarm0 function by writing 1 to [ALARM0_ENABLE_REG](#), then the software can query alarm count value in real time by [ALARM0_HH_MM_SS_SET_REG](#). When the setting time is reached, [ALARM0_IRQ_STA_REG](#) is set to 1 to generate an interrupt.
- Step 4** After entering the interrupt process, write 0 to the [ALARM0_IRQ_STA_REG](#) to clear the interrupt pending, and execute the interrupt process.
- Step 5** Resume the interrupt and continue to execute the interrupted process.

8.1.6 Register List

Module Name	Base Address
RTC	0x03090000

Register Name	Offset	Description
RTC		
LOSC_CTRL_REG	0x0000	Low Oscillator Control Register
INTOSC_CLK_PRESCAL_REG	0x0008	Internal OSC Clock Prescaler Register
RTC_DAY_REG	0x0010	RTC Year-Month-Day Register
RTC_HH_MM_SS_SET_REG	0x0014	RTC Hour-Minute-Second Register
ALARM0_DAY_SET_REG	0x0020	Alarm 0 Counter Register
ALARM0_HH_MM_SS_SET_REG	0x0024	Alarm0 Hour-Minute-Second Set Register
ALARM0_ENABLE_REG	0x0028	Alarm 0 Enable Register
ALARM0_IRQ_EN	0x002C	Alarm 0 IRQ Enable Register
ALARM0_IRQ_STA_REG	0x0030	Alarm 0 IRQ Status Register
ALARM0_CONFIG_REG	0x0050	Alarm 0 Configuration Register
CLK32K_FOUT_CTRL_GATING_REG	0x0060	32K Fanout Control Register
GP_DATA_REG	0x0100+N *0x0004	General Purpose Register (N=0~7)
DCXO_CTRL_REG	0x0160	DCXO Control Register
VDD_OFF_GATING_CTRL_REG	0x01F4	VDD Off Gating Control Register
EFUSE_HV_PWRSWT_CTRL_REG	0x0204	Efuse High Voltage Power Switch Control Register
ENC_INFO_KEY_REG	0x0260	Encrypt Information Key Register
ENC_KEY_EN_REG	0x0264	SPIF Encrypt Key Enable Register
SPIF_KEY_REG	0x026C	SPIF Encrypt Key Register

Register Name	Offset	Description
RTC_SPI_CLK_CTRL_REG	0x0310	RTC SPI Clock Control Register

8.1.7 Register Description

8.1.7.1 0x0000 LOSC Control Register (Default Value: 0x0000_4010)

Offset:0x0000			Register Name: LOSC_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:16	W	0x0	KEY_FIELD. Key Field. This field should be filled with 0x16AA, and then the bit 0 and bit 1 can be written with the new value.
15:9	/	/	/
8	R/W	0x0	RTC_HHMMSS_ACCE RTC Hour Minute Second access After writing the RTC HH-MM-SS Register, this bit is set and it will be cleared until the real writing operation is finished. After writing the RTC HH-MM-SS Register, the RTC HH-MM-SS Register will be refreshed for at most one second. Note: Make sure that the bit is 0 for time configuration.
7	R/W	0x0	RTC_DAY_ACCE RTC DAY access After writing the RTC DAY register, this bit is set and it will be cleared until the real writing operation is finished. After writing the RTC DAY register, the DAY register will be refreshed for at most one second. Note: Make sure that the bit is 0 for time configuration.
6:1	/	/	
0	R/W	0x0	LOSC_SRC_SEL. Low Frequency Clock(32.768KHz) source Select. 0: Low Frequency Clock from Internal RC OSC, 1:Low Frequency Clock from 24M HOSC.



NOTE

If the bit [8:7] of LOSC_CTRL_REG is set, the RTC HH-MM-SS, DD and ALARM DD-HH-MM-SS register cannot be written.

8.1.7.2 0x0008 Internal OSC Clock Prescaler Register (Default Value: 0x0000_000F)

Offset:0x0008			Register Name: INTOSC_CLK_PRESCAL_REG
Bit	Read/Write	Default/Hex	Description
31	R	0x1	OSC_ON_DET Whether the external 24M crystal oscillator is an effective marker bit. 1: An external 24M crystal oscillator is detected to be operating. 0: No external 24M crystal oscillator operation was detected.
30:26	/	/	/
25:16	R	0x1	OSC_CNT_DET The counting period of the external 24M crystal oscillator detection stage. When the internal 24M RC operates for 256 cycles, it corresponds to the number of cycles of the 24M crystal oscillator.
15:8	R/W	0x13	INTOSC_TRIM. Internal OSC TRIM Settings Note: This setting does not work when INTOSC_TRIM_SEL is 0 Note: When the software rewrites this register, the high 16 data must be filled with 16AA
7	/	/	/
6	R/W	0x0	INTOSC_TRIM_SEL Internal OSC TRIM setting selection 1: The Internal OSC TRIM setting comes from INTOSC_TRIM 0: The Internal OSC TRIM setting comes from efuse Note: When the software rewrites this register, the high 16 data must be filled with 16AA

Offset:0x0008			Register Name: INTOSC_CLK_PRESCAL_REG
Bit	Read/Write	Default/Hex	Description
5:4	R/W	0x1	HOSC_SEL. Internal 24M clock source selection 0x: The internal 24M clock source is automatically selected. When an external 24M crystal oscillator is detected to be running, the external 24M crystal oscillator is selected; otherwise, the internal OSC is selected 10: The internal 24M clock comes from the internal OSC 11: The internal 24M clock comes from the internal HOSC Note: When the software rewrites this register, the high 16 data must be filled with 16AA
3	R/W	0x1	OSC_DET_EN Enable the external 24M crystal oscillator detection function 1: Enable the external 24M crystal oscillator detection function 0: Disable the external 24M crystal oscillator detection function Note: The external 24M crystal oscillator detection function does not run in real time. Instead, it conducts a round of detection (lasting less than 100us) after being enabled and locks the detection results. If re-detection is required, this bit should be set to 0 and maintained for at least 4 24M cycles. Then set this bit to 1. Note: When the software rewrites this register, the high 16 data must be filled with 16AA
2:0	/	/	/

8.1.7.3 0x0010 RTC Year-Month-DAY Register (Default Value: UDF)

Offset:0x0010			Register Name: RTC_DAY_REG
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x0	DAY Set Day Range from 0 to 65535.

8.1.7.4 0x0014 RTC Hour-Minute-Second Register (Default Value: UDF)

Offset:0x0014			Register Name: RTC_HH_MM_SS_REG
Bit	Read/Write	Default/Hex	Description
31:21	/	/	/
20:16	R/W	0x0	HOUR Set hour Range from 0 to 23.
15:14	/	/	/
13:8	R/W	0x0	MINUTE Set minute Range from 0 to 59.
7:6	/	/	/
5:0	R/W	0x0	SECOND Set second Range from 0 to 59.

8.1.7.5 0x0020 Alarm 0 Day Setting Register (Default Value: 0x0000_0000)

Offset:0x0020			Register Name: ALARM0_DAY_SET_REG
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x0	ALARM0_COUNTER Alarm 0 Counter is based on Day.

8.1.7.6 0x0024 Alarm0 Hour-Minute-Second Set Register (Default Value: UDF)

Offset:0x0024			Register Name: ALARM0_HH_MM_SS_SET_REG
Bit	Read/Write	Default/Hex	Description
31:21	/	/	/
20:16	R/W	0x0	HOUR Current hour Range from 0 to 23.
15:14	/	/	/
13:8	R/W	0x0	MINUTE Current minute Range from 0 to 59.
7:6	/	/	/
5:0	R/W	0x0	SECOND Current second Range from 0 to 59.

8.1.7.7 0x0028 Alarm 0 Enable Register (Default Value: 0x0000_0000)

Offset:0x0028			Register Name: ALARM0_ENABLE_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	ALM_0_EN Alarm 0 Enable 0: Disable 1: Enable

8.1.7.8 0x002C Alarm 0 IRQ Enable Register (Default Value: 0x0000_0000)

Offset:0x002C			Register Name: ALARM0_IRQ_EN
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	ALARM0_IRQ_EN Alarm 0 IRQ Enable 0: Disable 1: Enable

8.1.7.9 0x0030 Alarm 0 IRQ Status Register (Default Value: 0x0000_0000)

Offset:0x0030			Register Name: ALARM0_IRQ_STA_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W1C	0x0	ALARM0_IRQ_PEND Alarm 0 IRQ Pending bit 0: No effect 1: Pending, alarm 0 counter value is reached If alarm 0 IRQ enable is set to 1, the pending bit will be sent to the interrupt controller.

8.1.7.10 0x0050 Alarm 0 Configuration Register (Default Value: 0x0000_0000)

Offset:0x0050			Register Name: ALARM0_CONFIG_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	ALARM_WAKEUP Configuration of alarm wake up output 0: Disable alarm wake up output 1: Enable alarm wake up output

8.1.7.11 0x0060 CLK32K Fanout Control Register (Default Value: 0x0000_0000)

Offset:0x0060			Register Name: CLK32K_FOUT_CTRL_GATING_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	HOSC_TO_32K_DIVIDER_ENABLE 1: Enable the HOSC 24M to 32k divider circuit 0: Disable the HOSC 24M to 32k divider circuit
15:3	/	/	/
2:1	R/W	0x0	LOSC_OUT_SRC_SEL: 10:HOSC divided 32K 01:INT LOSC divided 32K 00:SYS_32K(select by LOSC_SRC_SEL)
0	R/W	0x0	LOSC_OUT_GATING. Configuration of LOSC output, and no LOSC output by default. 0: Mask LOSC output gating 1: Enable LOSC output gating

8.1.7.12 0x0100+N*0x0004 General Purpose Register (Default Value: 0x0000_0000)

Offset:0x0100+N*0x0004 (N=0 to 7)			Register Name: GP_DATA_REGN
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	GP_DATA Data [31:0]



NOTE

General purpose register 0 to 7 value can be stored if the RTC-VIO is larger than 0.7 V.

8.1.7.13 0x0160 DCXO Control Register (Default Value: 0x883F_10F7)

Offset:0x0160			Register Name: DCXO_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x1	CLK_REQ_ENB Clock REQ enable 0: Enable DCXO wake up function 1: Disable DCXO wake up function
30:28	/	/	/
27:24	R/W	0x8	DCXO_ICTRL DCXO current control value
23	/	/	/

Offset:0x0160			Register Name: DCXO_CTRL_REG
Bit	Read/Write	Default/Hex	Description
22:16	R/W	0x3F	DCXO_TRIM DCXO cap array value The capacity cell is 55 fF.
15:13	/	/	/
12:8	R/W	0x10	DCXO_BG DCXO bandgap output voltage
7	R/W	0x1	DCXO_LDO_INRUSHB DCXO LDO driving capacity signal, active high
6	R/W	0x1	XTAL_MODE Xtal mode enable signal, active high 0: For external CLK input mode 1: For normal mode
5:4	R/W	0x3	DCXO_RFCLK_ENHANCE DCXO RFCLK enhance Enhance driving capacity of output OUT_RF_REFCLK, 0x0 for 5 pF, 0x1 for 10 pF, 0x2 for 15 pF, 0x3 for 20 pF.
3	/	/	/
2	R/W	0x1	Reserved
1	R/W	0x1	DCXO_EN DCXO enable 1: Enable 0: Disable
0	R/W	0x1	Reserved

8.1.7.14 0x01F4 VDD Off Gating Control Register (Default Value: 0x0000_0029)

Offset:0x01F4			Register Name: VDD_OFF_GATING_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
3	R/W	0x1	POR_RST_OUT_EN POR Reset Output Enable 0: Disable 1: Enable
2:1	/	/	/
0	R/W	0x1	VCCIO_DET_BYPASS_EN 0: Not bypass 1: Bypass

8.1.7.15 0x0204 Efuse High Voltage Power Switch Control Register (Default Value: 0x0000_0000)

Offset:0x0204			Register Name: EFUSE_HV_PWRSWT_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	EFUSE_1.8V_POWER_SWITCH_CONTROL 1: Open power switch 0: Close power switch

8.1.7.16 0x0260 Encrypt Information Key Register (Default Value: 0x0000_0000)

Offset: 0x0260			Register Name: ENC_INFO_KEY_REG
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x0	KEY_LOCK KEY_LOCK should be set to 0x1689, when read or write SPIF_KEY_REG

8.1.7.17 0x0264 SPIF Encrypt Key Enable Register (Default Value: 0x0000_0000)

Offset: 0x0264			Register Name: ENC_KEY_EN_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	SPIF_KEY_EN Enable SPIF Key Register Info.

8.1.7.18 0x026C SPIF Encrypt Key Register (Default Value: 0x0000_0000)

Offset: 0x026C			Register Name: SPIF_KEY_REG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	SPIF_KEY SPIF Key.

8.1.7.19 0x0310 RTC SPI Clock Control Register (Default Value: 0x0000_0009)

Offset:0x0310			Register Name: RTC_SPI_CLK_CTRL_REG
Bit	Read/Write	Default/Hex	Description

Offset:0x0310			Register Name: RTC_SPI_CLK_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	RTC REG CFG SPI Clock Gating 0: Gating 1: Not Gating Before configuring RTC register, the clock divider of SPI needs be configured firstly, then clock gating needs be enabled. Note: Frequency division and clock gating cannot be set at the same time.
30:5	/	/	/
4:0	R/W	0x9	RTC REG CFG SPI Clock Divider: M Actual SPI Clock = $AHB0/(M+1)$, (0 to 15) The default frequency of AHB0 is 200 MHz, and the default frequency of SPI Clock is 20 MHz. Note: The SPI clock cannot exceed 50 MHz, or else the RTC register may be abnormal.

9 Memory Controllers

9.1 SDRAM Controller (DRAMC)

The DRAMC has the following features:

- 8/16-bit DDR2/DDR3/DDR3L interface, up to 256 MB
- Clock frequency up to 528 MHz for external DDR2/DDR3/DDR3L



9.2 SD/MMC Host Controller (SMHC)

9.2.1 Overview

The SMHC (SD-MMC Host controller) can be configured to support Secure Digital memory (SD Memory), UHS-I Card, Secure Digital I/O (SDIO), Multimedia Cards (MMC), and eMMC.

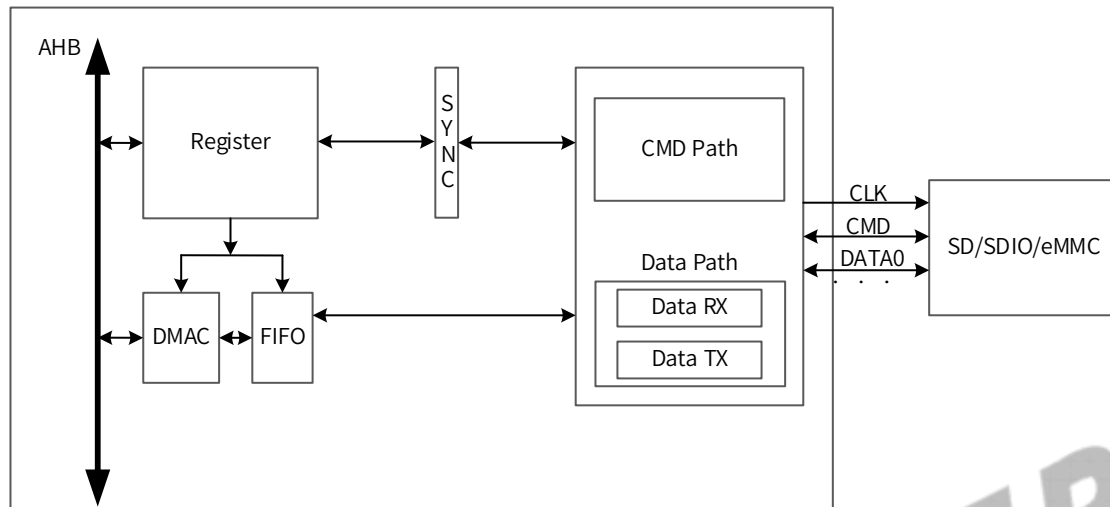
The SMHC has the following features:

- Three SD/MMC Host Controller (SMHC) interfaces
 - SMHC0, compliant with the protocol Secure Digital Memory (up to SD3.0)
 - SMHC2, compliant with the protocol Multimedia Card (up to eMMC4.41)
 - Supports one SD (Version 1.0 to 3.0) or MMC (Version 3.3 to 4.41)
- The SMHC0 support the following:
 - 1-bit or 4-bit data width
 - Maximum performance:
 - SDR mode 100 MHz@1.8 V IO pad
 - DDR mode 50 MHz@1.8 V IO pad
 - SDR mode 50 MHz@3.3 V IO pad
- Supports block size of 1 to 65535 bytes
- Supports hardware CRC generation and error detection
- Supports eMMC boot operation and alternative boot operation
- Supports Command Completion signals and interrupts to host processor, and Command Completion signal disable feature
- Programmable baud rate
- Descriptor-based internal DMA controller
- Internal 1K Bytes FIFO for data transfer

9.2.2 Block Diagram

The following figure shows a block diagram of the SMHC.

Figure 9-1 SMHC Block Diagram



SMHC contains the following sub-blocks:

Table 9-1 SMHC Sub-blocks

Sub-block	Description
Register	It is used to configure the control signal for reading or writing the SD/SDIO/eMMC.
DMAC	The DMA controller that controls the data transfer between the memory and SMHC.
FIFO	A buffer for the data stream between the memory and the SMHC asynchronous clock domain.
SYNC	It synchronizes the signals from the AHB clock domain to the SMHC clock domain.
CMD Path	It sends commands to or receives commands from the SD/SDIO/eMMC.
Data Path	It consists of Data TX and Data RX sub-modules. The Data TX sends data blocks and the CRC codes to the SD/SDIO/eMMC. The Data RX receives data blocks and the CRC codes from the SD/SDIO/eMMC.

9.2.3 External Signals

The following table describes the external signals of SMHC.

Table 9-2 SMHC External Signals

Signal Name	Description	Type
SMHC0		
SDC0-CMD	SD Card Command Output/Response Input	I/O, OD
SDC0-CLK	SD Card Clock Output	O
SDC0-D0	SD Card Data Input/ Output 0	I/O

Signal Name	Description	Type
SDC0-D1	SD Card Data Input/ Output 1	I/O
SDC0-D2	SD Card Data Input/ Output 2	I/O
SDC0-D3	SD Card Data Input/ Output 3	I/O
SMHC2		
SDC2-CMD	eMMC Command Output/Response Input	I/O, OD
SDC2-CLK	eMMC Clock Output	O
SDC2-D0	eMMC Data Input/ Output 0	I/O
SDC2-D1	eMMC Data Input/ Output 1	I/O
SDC2-D2	eMMC Data Input/ Output 2	I/O
SDC2-D3	eMMC Data Input/ Output 3	I/O

9.2.4 Functional Description

9.2.4.1 Data Path

The SMHC and SD/SDIO/eMMC contains the following interface buses: CLK, CMD, and DATA 1/4. During one clock cycle, the SMHC can transmit one bit command with one or two bits data in 1-ch DATA mode, or four or eight bits data in 4-ch DATA mode. The CMD is a bidirection channel for initializing the SD/SDIO/eMMC and transmitting commands. It can work in both the open-drain mode and push-pull mode. The DATA is also a bidirection channel. It works in the push-pull mode.

- Reading Data from the SD/SDIO/eMMC

The register configures the signals for the read operation, and synchronizes the signals to the SMHC clock domain. Then the Data RX reads data from the SD/SDIO/eMMC via the CLK/CMD/DATA interface buses and writes the data in the FIFO. After that, the DMAC transfers the data from the FIFO to the memory.

- Writing Data to the SD/SDIO/eMMC

The register configures the signals for the write operation, and synchronize the signals to the SMHC clock domain. Then the DMAC reads data from the memory and writes the data to the FIFO. After that, the Data TX reads the data from the FIFO and writes the data to the SD/SDIO/eMMC via the CLK/CMD/DATA interface buses.

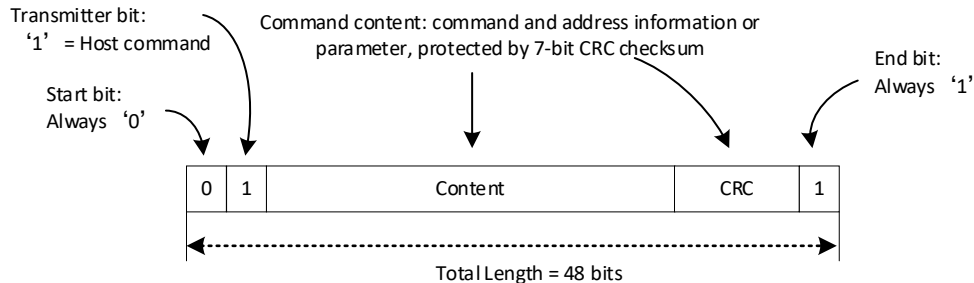
9.2.4.2 Package Format

Data transfer over the SD/SDIO/eMMC bus is based on command and data bitstreams that are initiated by a start bit and terminated by a stop bit. There are three types of SD/SDIO/eMMC packets: command token, response token, and data packet.

Command Tokens

The command token starts an operation. A command is sent from the host to a device. It is transferred serially on the CMD line. Command tokens have the following coding scheme:

Figure 9-2 Command Token Format



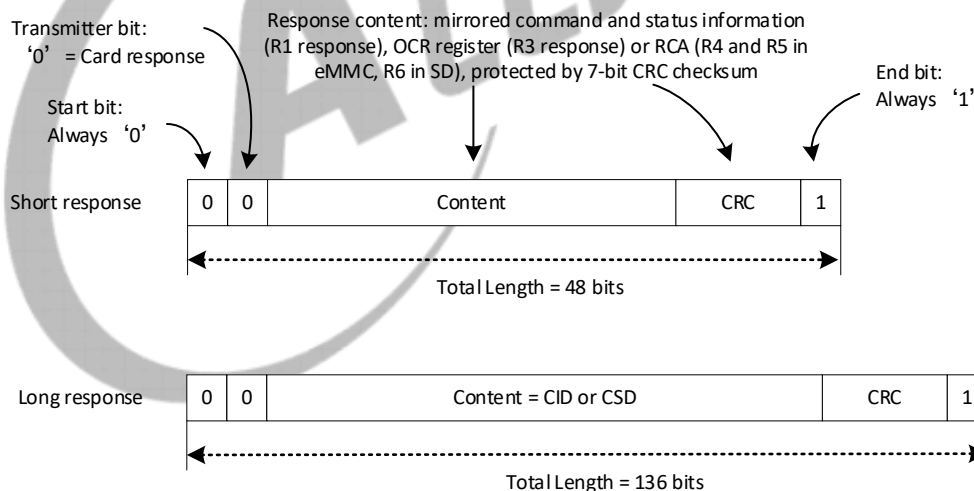
Each command token has 48 bits, preceded by a start bit ('0') and succeeded by an end bit ('1'). To detect transmission errors, each token is protected by CRC bits.

Response Tokens

After receiving a command, the card returns a 48-bit or 136-bit response based on the command type.

A response token is sent from the device to the host as an answer to a previously received command. It is transferred serially on the CMD line.

Figure 9-3 Response Token Format



Data Packet



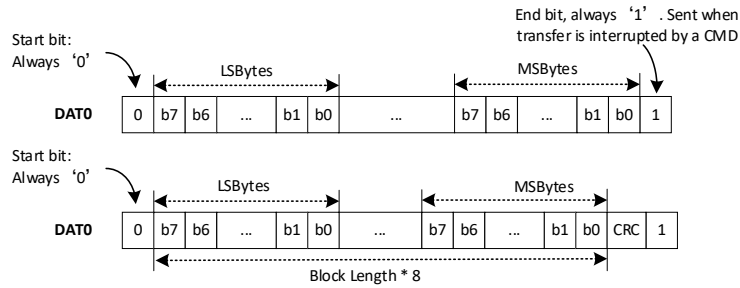
NOTE

This device does not support the bus width of 8 bits.

Data can be transferred from the device to the host or vice versa. Data are transferred via the data lines.

Figure 9-4 Data Packet Format for SDR

1 Bit Bus (only DAT0 used)



4 Bits Bus (DAT3 – DAT0 used)

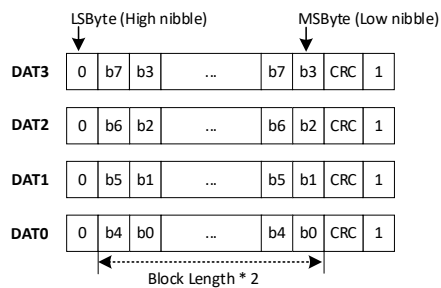
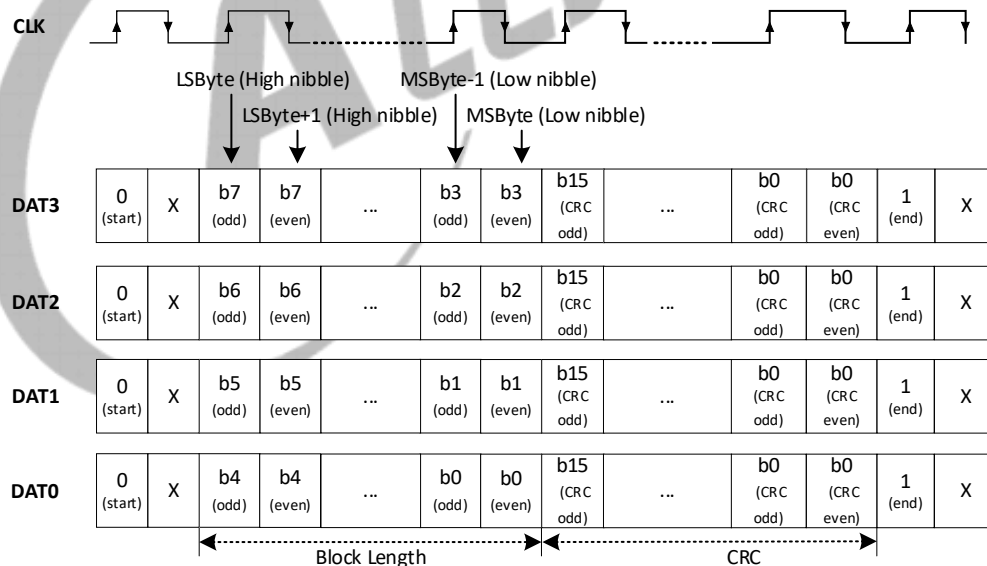


Figure 9-5 Data Packet Format for DDR

4 Bits Bus DDR (DAT3 – DAT0 used)



NOTE

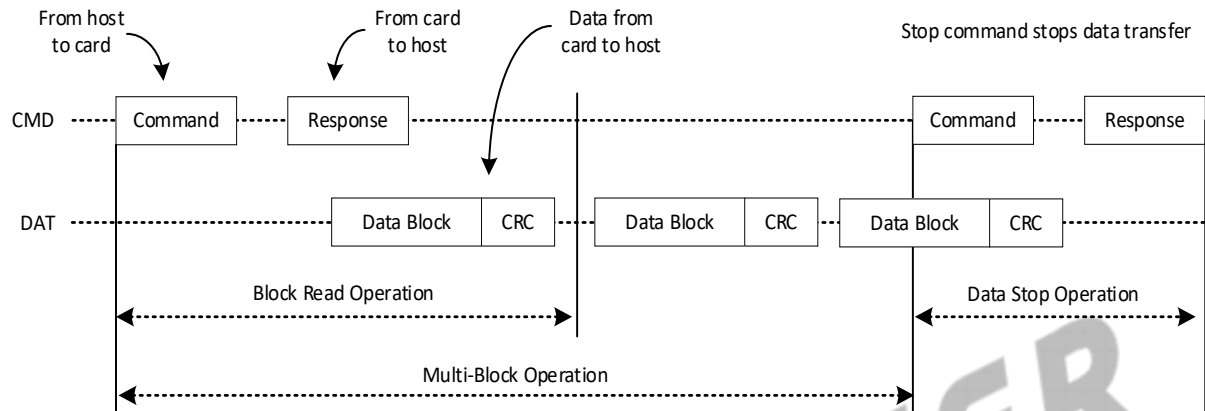
- Bytes data are not interleaved but CRCs are interleaved.
- Start and end bits are only valid on the rising edge ("X" indicates "undefined").

9.2.4.3 Data Transfer

Data transfers to or from the SD/eMMC card are done in blocks. Single and multiple block operations are widely used during data transfer.

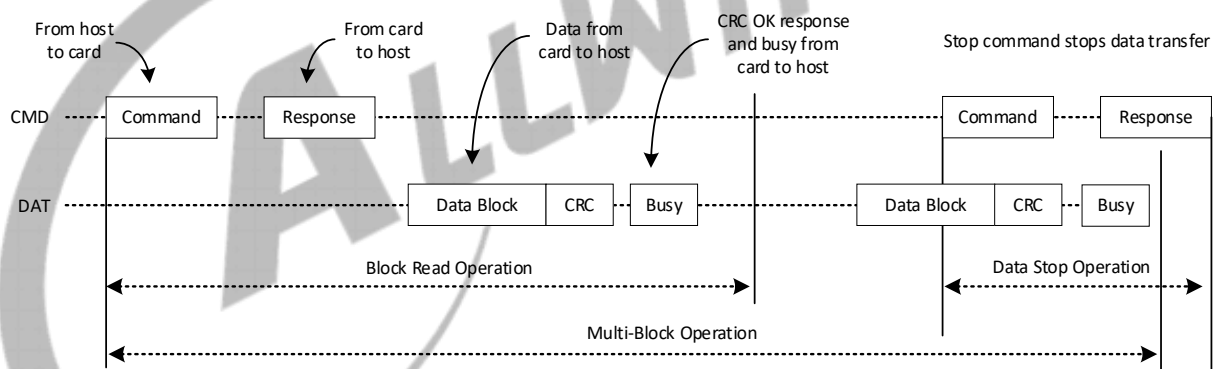
The following figure shows the single-block and multi-block read operation.

Figure 9-6 Single-Block and Multi-Block Read Operation



The following figure shows the single-block and multi-block write operation.

Figure 9-7 Single-Block and Multi-Block Read Operation



9.2.4.4 Internal DMA Controller

The SMHC has an internal DMA controller (IDMAC) to transfer data between the host memory and SMHC port. With a descriptor, the IDMAC can efficiently move data from the source to destination by automatically loading the next DMA transfer arguments, which needs less CPU intervention. Before transferring data in the IDMAC, the host driver should construct a descriptor list, configure arguments of every DMA transfer, and then launch the descriptor and start the DMA.

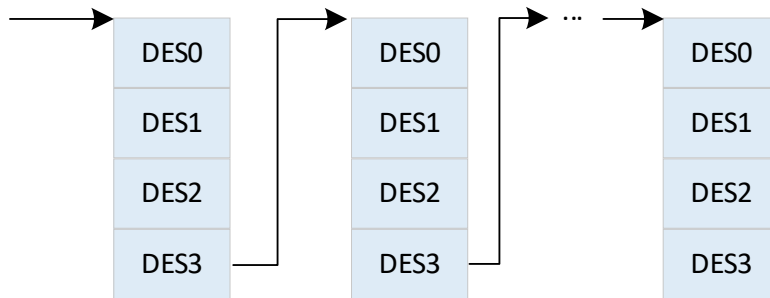
The IDMAC has an interrupt controller. When enabled, it generates an interrupt to the HOST CPU in situations such as data transmission is completed or some error happens.

IDMAC Descriptor Structure

The IDMAC uses a descriptor with a chain structure, and each descriptor points to a unique buffer and the next descriptor.

The following figure shows the internal formats of a descriptor.

Figure 9-8 IDMAC Descriptor Structure Diagram



This figure illustrates the internal formats of a descriptor. The descriptor address must be aligned to the bus width used for 32-bit buses. Each descriptor contains 16 bytes of control and status information.

DES0 corresponds to the [31:0] bits, DES1 corresponds to the [63:32] bits, DES2 corresponds to the [95:64] bits, and DES3 corresponds the [127:96] bits in a descriptor.

The following table shows the bit definition of DES0.

Table 9-3 DES0 Definition

Bits	Name	Description
31	HOLD	DES_OWN_FLAG When set, this bit indicates that the descriptor is owned by the IDMAC. When this bit is reset, it indicates that the descriptor is owned by the host. This bit is cleared when the transfer is over.
30	ERROR	ERR_FLAG When some errors happen in transfer, this bit will be set to 1.
29:5	/	/
4	Chain Flag	CHAIN_MOD When set to 1, this bit indicates that the second address in the descriptor is the next descriptor address. It must be set to 1.
3	First DES Flag	FIRST_FLAG When set to 1, this bit indicates that this descriptor contains the first buffer of data. It must be set to 1 in the first DES.
2	Last DES Flag	LAST_FLAG When set to 1, this bit indicates that the buffers this descriptor points to are the last data buffer.
1	Disable Interrupt on completion	CUR_TXRX_OVER_INT_DIS When set to 1, this bit will prevent the setting of the TX/RX interrupt bit of the IDMAC status register for data that ends in the buffer the descriptor points to.

Bits	Name	Description
0	/	/

The following table shows the bit definition of DES1.

Table 9-4 DES1 Definition

Bits	Name	Description
31:13	/	/
12:0	Buffer size	BUFF_SIZE The bits indicate the data buffer byte size, which must be a multiple of 4 bytes. If this field is 0, the DMA ignores this buffer and proceeds to the next descriptor.

The following table shows the bit definition of DES2.

Table 9-5 DES2 Definition

Bits	Name	Descriptor
31:0	Buffer address pointer	BUFF_ADDR The bits indicate the physical address of the data buffer. It is a word (4 bytes) address.

The following table shows the bit definition of DES3.

Table 9-6 DES3 Definition

Bits	Name	Descriptor
31:0	Next descriptor address	NEXT_DESP_ADDR The bits indicate the pointer to the physical memory where the next descriptor is present. It is a word (4 bytes) address.

9.2.4.5 Timing Specifications

Refer to the following relative specifications:

- Physical Layer Specification Ver3.00 Final
- SDIO Specification Ver2.00
- Multimedia Cards (MMC – version 4.2)
- JEDEC Standard – JESD84-44, Embedded Multimedia Card (eMMC) Card Product Standard
- JEDEC Standard – JESD84-B45, Embedded Multimedia Card (eMMC) Electrical Standard (4.5 Device)
- JEDEC Standard – JESD84-B50, Embedded Multimedia Card (eMMC) Electrical Standard (5.0)

9.2.4.6 Phase Offset of the Command and Data

You can configure the phase offset of the command and data by the [SMHC_DRV_DL](#) register.

SDR Mode

The following figure shows the phase offset of SDR command and data.

Figure 9-9 Phase Offset of Command and Data in SDR Mode

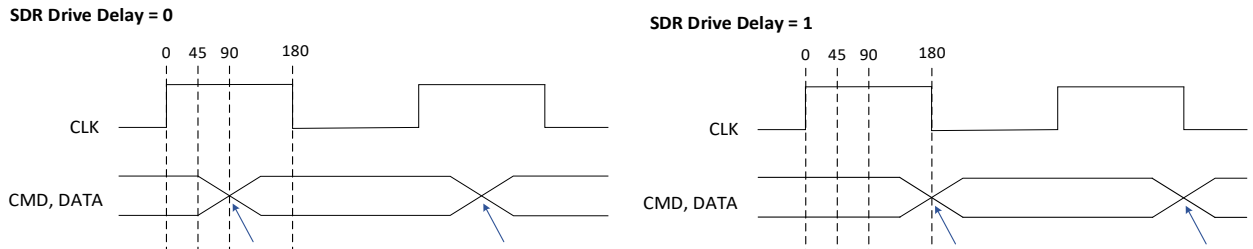


Table 9-7 Phase Offset of Command and Data in SDR Mode

Drive Delay	Command	Data
0	The command is updated in 90° clock position	The data is updated in 90° clock position
1	The command is updated in 180° clock position	The data is updated in 180° clock position

DDR4 Mode

The following figure shows the phase offset of DDR4 command and data.

Figure 9-10 Phase Offset of Command and Data in DDR4 Mode ([SMHC_NTSR](#) [31] = 0)

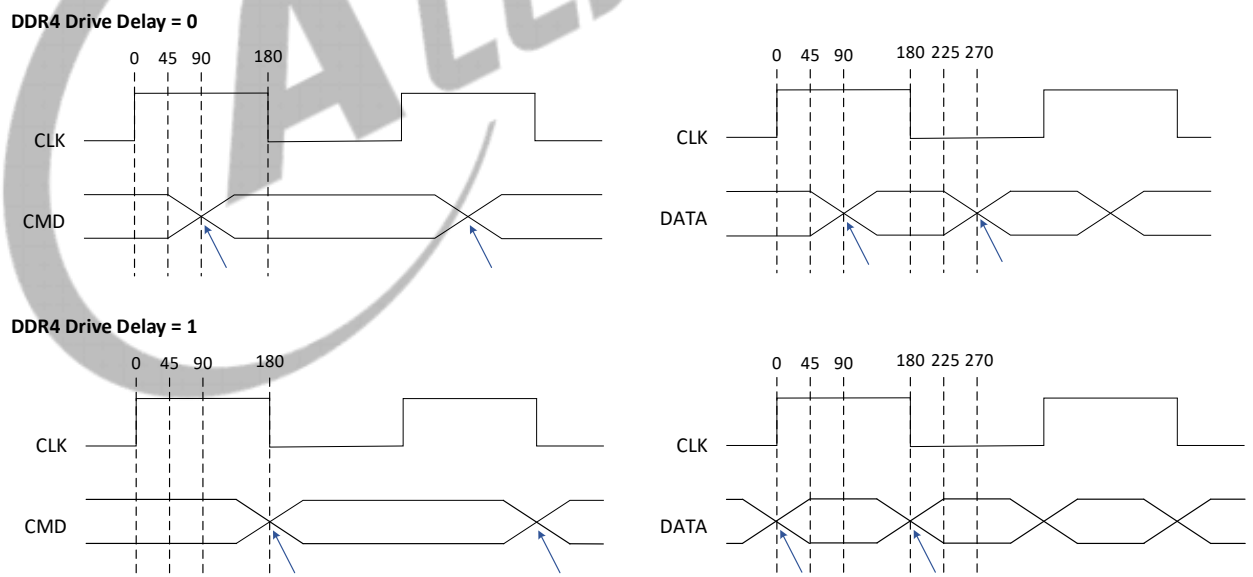


Table 9-8 Phase Offset of Command and Data in DDR4 Mode

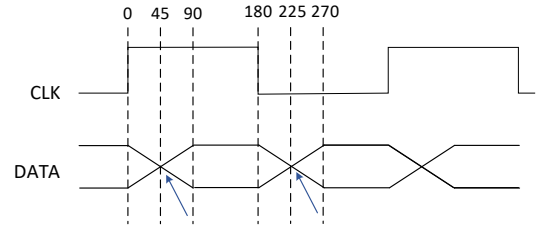
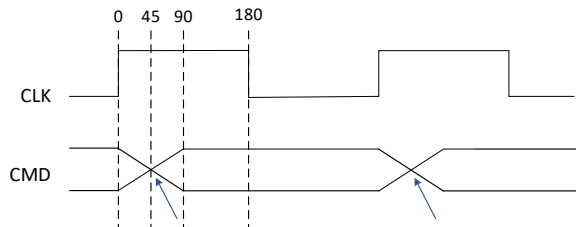
Drive Delay	Command	Data
0	The command is updated in 90° clock position	The data is updated in 90° clock position
1	The command is updated in 180° clock position	The data is updated in 0° or 180° clock position

DDR4 (2x) Mode

The following figure shows the phase offset of DDR4 (2x mode) command and data.

Figure 9-11 Phase Offset of Command and Data in DDR4 (2x Mode) ([SMHC_NTSR](#) [31] = 1)

DDR4 (2x mode) Drive Delay = 0



DDR4 (2x mode) Drive Delay = 1

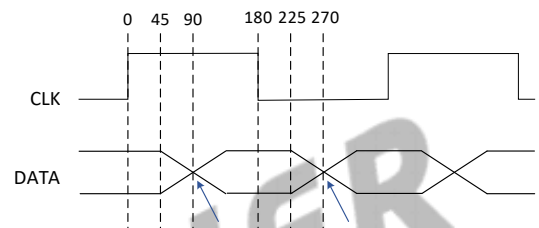
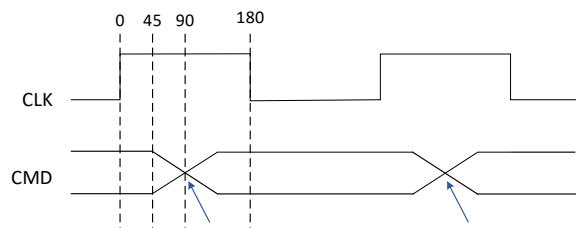


Table 9-9 Phase Offset of Command and Data in DDR4 (2x) Mode

Drive Delay	Command	Data
0	The command is updated in 45° clock position	The data is updated in 45° clock position
1	The command is updated in 90° clock position	The data is updated in 90° clock position

9.2.5 Programming Guidelines

9.2.5.1 Initialization

Before data and commands are exchanged between a card and the SMHC, the SMHC needs to be initialized. Follow the steps below to initialize the SMHC:

Configure the corresponding GPIO register as an SMHC in the section 15.7 GPIO.

Step 1 reset clock by writing 1 to SMHCx_BGR_REG [SMHCx_RST](x=0, 1, or 2), and open clock gating by writing 1 to SMHCx_BGR_REG [SMHCx_GATING] (x=0, 1, or 2); select clock sources and set the division factor by configuring the SMHCx_CLK_REG (x = 0, 1, or 2) register.

Step 2 Configure [SMHC_CTRL](#) to reset the FIFO and controller, and enable the global interrupt; configure [SMHC_INTMASK](#) to 0xFFCE to enable the common interrupts and the interrupts caused by error/anomaly, and then register the interrupt function.

- Step 3** Configure [SMHC_CLKDIV](#) to open clock for devices. Configure [SMHC_CMD](#) as the change clock command (for example 0x80202000). Send the update clock command to deliver clock to devices.
- Step 4** Configure [SMHC_CMD](#) as a normal command. Configure [SMHC_CMDARG](#) to set command parameters. Configure [SMHC_CMD](#) to set parameters like the command sequence, the response receive, the response type, the response length, the response types to send the commands. According to the initialization process in the protocol, you can finish SMHC initialization by sending the corresponding command one by one.

9.2.5.2 Single-block Write (CMD24)

To write a single block of data, follow the steps below:

- Step 1** Write 0x1 to [SMHC_CTRL](#) [DMA_RST] to reset the internal DMA controller; write 0x82 to [SMHC_IDMAC](#) to enable the IDMAC interrupt and configure AHB master burst transfers; configure [SMHC_IDIE](#) to enable the transfer interrupt, receive interrupt, and abnormal interrupt.
- Step 2** Configure [SMHC_FIFOTH](#) to determine the burst size and TX/RX trigger level. For example, if [SMHC_FIFOTH](#) is configured as 0x300F00F0, it indicates the burst size is 16, TX_TL is 15, and RX_TL is 240. Configure [SMHC_DLBA](#) to determine the start address of the DMA descriptor.
- Step 3** To write one block data to sector1, configure [SMHC_BYTCNT](#)[BYTE_CNT] to 0x200 and configure the descriptor according to the data size; set the data sector address of CMD24 (Single Data Block Write) to 0x1, write 0x80002758 to [SMHC_CMD](#), and send CMD24 command to write data to the device.
- Step 4** Check whether [SMHC_RINTSTS](#)[CC] is 1. If yes, the command is sent successfully; otherwise, continue to wait until timeout, and then exit the process.
- Step 5** Check whether [SMHC_IDST](#)[TX_INT] is 1. If yes, the data transfer for writing DMA is completed. Write 0x337 to [SMHC_IDST](#) to clear the interrupt flag. Otherwise, continue to wait until timeout, and then exit the process.
- Step 6** Check whether [SMHC_RINTSTS](#)[DTC] is 1. If yes, the data transfer and CMD24 writing operations are completed. Otherwise, anomaly occurs. Read [SMHC_RINTSTS](#) and [SMHC_STATUS](#) to query the existing anomaly.
- Step 7** Send CMD13 command to query whether the device writing operation is completed and returns to the idle status. For example, device RCA is 0x1234, first set [SMHC_CMDARG](#) to 0x12340000, write 0x8000014D to [SMHC_CMD](#), go to step4 to ensure command transfer is completed, and then check whether the bit8 of [SMHC_RESPO](#) (CMD13 response) is 1. If yes, the device is in the idle status, and the next command can be sent. Otherwise, the

device is in the busy status. Continue to send CMD13 to wait for the device to enter the idle status until timeout.

9.2.5.3 Single-block Read (CMD17)

To read a single block of data, follow the steps below:

- Step 1** Write 0x1 to [SMHC_CTRL](#)[DMA_RST] to reset the internal DMA controller; write 0x82 to [SMHC_IDMAC](#) to enable the IDMAC interrupt and configure AHB master burst transfers; configure [SMHC_IDIE](#) to enable the transfer interrupt, receive interrupt, and abnormal interrupt.
- Step 2** Configure [SMHC_FIFOTH](#) to determine the burst size and TX/RX trigger level. For example, if [SMHC_FIFOTH](#) is configured as 0x300F00F0, it indicates the burst size is 16, TX_TL is 15, and RX_TL is 240. Configure [SMHC_DLBA](#) to determine the start address of the DMA descriptor.
- Step 3** To read one block data from sector1, configure [SMHC_BYTCNT](#)[BYTE_CNT] to 0x200 and configure the descriptor according to the data size; set the data sector address of CMD17 command (Single Data Block Read) to 0x1, write 0x80002351 to [SMHC_CMD](#), and send CMD17 command to read data from the device to DRAM/SRAM.
- Step 4** Check whether [SMHC_RINTSTS](#)[CC] is 1. If yes, the command is sent successfully; otherwise, continue to wait until timeout, and then exit the process.
- Step 5** Check whether [SMHC_IDST](#)[RX_INT] is 1. If yes, the data transfer for writing DMA is completed. Write 0x337 to [SMHC_IDST](#) to clear the interrupt flag. Otherwise, continue to wait until timeout, and then exit the process.
- Step 6** Check whether [SMHC_RINTSTS](#)[DTC] is 1. If yes, data transfer and CMD17 reading operation are completed. Otherwise, anomaly occurs. Read [SMHC_RINTSTS](#) and [SMHC_STATUS](#) to query the existing abnormality.

9.2.5.4 Open-ended Multiple-Block Write (CMD25+Auto CMD12)

To write open-ended multiple data blocks, follow the steps below:

- Step 1** Write 0x1 to [SMHC_CTRL](#)[DMA_RST] to reset the internal DMA controller; write 0x82 to [SMHC_IDMAC](#) to enable the IDMAC interrupt and configure AHB master burst transfers; configure [SMHC_IDIE](#) to enable the transfer interrupt, receive interrupt, and abnormal interrupt.
- Step 2** Configure [SMHC_FIFOTH](#) to determine the burst size and TX/RX trigger level. For example, if [SMHC_FIFOTH](#) is configured as 0x300F00F0, it indicates the burst size is 16, TX_TL is 15, and RX_TL is 240. Configure [SMHC_DLBA](#) to determine the start address of the DMA descriptor.

- Step 3** To write three blocks of data to sectors begin with sector0, configure [SMHC_BYTCNT](#)[BYTE_CNT] to 0x600 and configure the descriptor according to the data size; set the data sector address of CMD25 command (Multiple Data Blocks Write) to 0x0, write 0x80003759 to [SMHC_CMD](#), and send CMD25 command to read data from the device to DRAM/SRAM.
- Step 4** Check whether [SMHC_RINTSTS](#)[CC] is 1. If yes, the command is sent successfully; otherwise, continue to wait until timeout, and then exit the process.
- Step 5** Check whether [SMHC_IDST](#)[TX_INT] is 1. If yes, the data transfer for writing DMA is completed. Write 0x337 to [SMHC_IDST](#) to clear the interrupt flag. Otherwise, continue to wait until timeout, and then exit the process.
- Step 6** Check whether [SMHC_RINTSTS](#)[ACD] and [SMHC_RINTSTS](#)[DTC] are both 1. If yes, the data transfer, CMD12 transfer, and CMD25 writing operations are completed. Otherwise, abnormality exists. Read [SMHC_RINTSTS](#) and [SMHC_STATUS](#) to query the existing abnormality.
- Step 7** Send CMD13 command to query whether the device writing operation is completed and returns to the idle status. For example, device RCA is 0x1234, first set [SMHC_CMDARG](#) to 0x12340000, write 0x8000014D to [SMHC_CMD](#), go to step4 to ensure command transfer completed, and then check whether the bit8 of [SMHC_RESP0](#) (CMD13 response) is 1. If yes, the device is in the idle status, and the next command can be sent. Otherwise, the device is in the busy status. Continue to send CMD13 to wait for the device to enter the idle status until timeout.

9.2.5.5 Open-ended Multiple-block Read (CMD18+Auto CMD12)

To read open-ended multiple data blocks, follow the steps below:

- Step 1** Write 0x1 to [SMHC_CTRL](#)[DMA_RST] to reset the internal DMA controller; write 0x82 to [SMHC_IDMAC](#) to enable the IDMAC interrupt and configure AHB master burst transfers; configure [SMHC_IDIE](#) to enable the transfer interrupt, receive interrupt, and abnormal interrupt.
- Step 2** Configure [SMHC_FIFOTH](#) to determine the burst size and TX/RX trigger level. For example, if [SMHC_FIFOTH](#) is configured as 0x300F00F0, it indicates the burst size is 16, TX_TL is 15, and RX_TL is 240. Configure [SMHC_DLBA](#) to determine the start address of the DMA descriptor.
- Step 3** To read three blocks of data from sectors begin with sector0, configure [SMHC_BYTCNT](#)[BYTE_CNT] to 0x600 and configure the descriptor according to the data size; set the data sector address of CMD18 command (Multiple Data Blocks Read) to 0x0, write 0x80003352 to [SMHC_CMD](#), and send CMD18 command to read data to the device. When the data transfer is completed, CMD12 will be sent automatically.

- Step 4** Check whether [SMHC_RINTSTS](#)[CC] is 1. If yes, the command is sent successfully; otherwise, continue to wait until timeout, and then exit the process.
- Step 5** Check whether [SMHC_IDST](#)[RX_INT] is 1. If yes, the data transfer for writing DMA is completed. Write 0x337 to [SMHC_IDST](#) to clear the interrupt flag. Otherwise, continue to wait until timeout, and then exit the process.
- Step 6** Check whether [SMHC_RINTSTS](#)[ACD] and [SMHC_RINTSTS](#)[DTC] are both 1. If yes, data transfer, CMD12 transfer, and CMD18 reading operation are completed. Otherwise, abnormality exists. Read [SMHC_RINTSTS](#) and [SMHC_STATUS](#) to query the existing abnormality.

9.2.5.6 Pre-Defined Multiple-Block Write (CMD23+CMD25)

To write pre-defined multiple data blocks, follow the steps below:

- Step 1** Write 0x1 to [SMHC_CTRL](#)[DMA_RST] to reset the internal DMA controller; write 0x82 to [SMHC_IDMAC](#) to enable the IDMAC interrupt and configure AHB master burst transfers; configure [SMHC_IDIE](#) to enable the transfer interrupt, receive interrupt, and abnormal interrupt.
- Step 2** Configure [SMHC_FIFOTH](#) to determine the burst size and TX/RX trigger level. For example, if [SMHC_FIFOTH](#) is configured as 0x300F00F0, it indicates the burst size is 16, TX_TL is 15, and RX_TL is 240. Configure [SMHC_DLBA](#) to determine the start address of the DMA descriptor.
- Step 3** To write three blocks of data, configure [SMHC_CMDARG](#) to 0x3 to specify the number of data blocks as three. Then write 0x80000157 to [SMHC_CMD](#) to send the CMD23 command. Check whether [SMHC_RINTSTS](#)[CC] is 1. If yes, the command is sent successful; otherwise, continue to wait until timeout, and then exit the process.
- Step 4** Configure [SMHC_BYTCNT](#)[BYTE_CNT] to 0x600 and configure the descriptor according to the data size; set the data sector address of CMD25 command (Multiple Data Blocks Write) to 0x0, write 0x80002759 to [SMHC_CMD](#), and send CMD25 command to write data to the device.
- Step 5** Check whether [SMHC_RINTSTS](#)[CC] is 1. If yes, the command is sent successful; otherwise, continue to wait until timeout, and then exit the process.
- Step 6** Check whether [SMHC_IDST](#)[TX_INT] is 1. If yes, the data transfer for writing DMA is completed. Write 0x337 to [SMHC_IDST](#) to clear the interrupt flag. Otherwise, continue to wait until timeout, and then exit the process.
- Step 7** Check whether [SMHC_RINTSTS](#)[DTC] is 1. If yes, the data transfer and CMD25 writing operations are completed. Otherwise, abnormality exists. Read [SMHC_RINTSTS](#) and [SMHC_STATUS](#) to query the existing abnormality.

Step 8 Send CMD13 command to query whether the device writing operation is completed and returns to the idle status. For example, device RCA is 0x1234, first set [SMHC_CMDARG](#) to 0x12340000, write 0x8000014D to [SMHC_CMD](#), go to step5 to ensure command transfer completed, and then check whether the bit8 of [SMHC_RESP0](#) (CMD13 response) is 1. If yes, the device is in the idle status, and the next command can be sent. Otherwise, the device is in the busy status. Continue to send CMD13 to wait for the device to enter the idle status until timeout

9.2.5.7 Pre-Defined Multiple-Block Read (CMD23+CMD18)

To read pre-defined multiple data blocks, follow the steps below:

- Step 1** Write 0x1 to [SMHC_CTRL](#) [DMA_RST] to reset the internal DMA controller; write 0x82 to [SMHC_IDMAC](#) to enable the IDMAC interrupt and configure AHB master burst transfers; configure [SMHC_IDIE](#) to enable the transfer interrupt, receive interrupt, and abnormal interrupt.
- Step 2** Configure [SMHC_FIFOTH](#) to determine the burst size and TX/RX trigger level. For example, if [SMHC_FIFOTH](#) is configured as 0x300F00F0, it indicates the burst size is 16, TX_TL is 15, and RX_TL is 240. Configure [SMHC_DLBA](#) to determine the start address of the DMA descriptor.
- Step 3** To read three blocks of data, configure [SMHC_CMDARG](#) to 0x3 to specify the number of data blocks as three. Then write 0x80000157 to [SMHC_CMD](#) to send the CMD23 command. Check whether [SMHC_RINTSTS](#)[CC] is 1. If yes, the command is sent successful; otherwise, continue to wait until timeout, and then exit the process.
- Step 4** Configure [SMHC_BYTCNT](#) [BYTE_CNT] to 0x600 and configure the descriptor according to the data size; set the data sector address of CMD18 (Multiple Data Blocks Read) to 0x0, write 0x80002352 to [SMHC_CMD](#), and send CMD18 command to read data from device to DRAM/SRAM.
- Step 5** Check whether [SMHC_RINTSTS](#)[CC] is 1. If yes, the command is sent successful; otherwise, continue to wait until timeout, and then exit the process.
- Step 6** Check whether [SMHC_IDST](#)[RX_INT] is 1. If yes, the data transfer for writing DMA is completed. Write 0x337 to [SMHC_IDST](#) to clear the interrupt flag. Otherwise, continue to wait until timeout, and then exit the process.
- Step 7** Check whether [SMHC_RINTSTS](#)[DTC] is 1. If yes, the data transfer and CMD18 writing operations are completed. Otherwise, abnormality exists. Read [SMHC_RINTSTS](#) and [SMHC_STATUS](#) to query the existing abnormality.

9.2.6 Register List

Module Name	Base Address
SMHC0	0x04020000
SMHC2	0x04022000

Register Name	Offset	Description
SMHC_CTRL	0x0000	SMHC Control Register
SMHC_CLKDIV	0x0004	SMHC Clock Control Register
SMHC_TMOUT	0x0008	SMHC Timeout Register
SMHC_CTYPE	0x000C	SMHC Bus Width Register
SMHC_BLKSIZE	0x0010	SMHC Block Size Register
SMHC_BYTCNT	0x0014	SMHC Byte Count Register
SMHC_CMD	0x0018	SMHC Command Register
SMHC_CMDARG	0x001C	SMHC Command Argument Register
SMHC_RESP0	0x0020	SMHC Response 0 Register
SMHC_RESP1	0x0024	SMHC Response 1 Register
SMHC_RESP2	0x0028	SMHC Response 2 Register
SMHC_RESP3	0x002C	SMHC Response 3 Register
SMHC_INTMASK	0x0030	SMHC Interrupt Mask Register
SMHC_MINTSTS	0x0034	SMHC Masked Interrupt Status Register
SMHC_RINTSTS	0x0038	SMHC Raw Interrupt Status Register
SMHC_STATUS	0x003C	SMHC Status Register
SMHC_FIFOTH	0x0040	SMHC FIFO Water Level Register
SMHC_FUNS	0x0044	SMHC Function Select Register
SMHC_TBC0	0x0048	SMHC Transferred Byte Count Register 0
SMHC_TBC1	0x004C	SMHC Transferred Byte Count Register 1
SMHC_A12A	0x0058	SMHC Auto Command 12 Argument Register
SMHC_NTSR	0x005C	SMHC New Timing Set Register
SMHC_HWRST	0x0078	SMHC Hardware Reset Register
SMHC_IDMAC	0x0080	SMHC IDMAC Control Register
SMHC_DLBA	0x0084	SMHC Descriptor List Base Address Register
SMHC_IDST	0x0088	SMHC IDMAC Status Register
SMHC_IDIE	0x008C	SMHC IDMAC Interrupt Enable Register
SMHC_THLD	0x0100	SMHC Card Threshold Control Register
SMHC_SFC	0x0104	SMHC Sample FIFO Control Register
SMHC_A23A	0x0108	SMHC Auto Command 23 Argument Register
EMMC_DDR_SBIT_DET	0x010C	SMHC eMMC4.5 DDR Start Bit Detection Control Register
SMHC_EXT_CMD	0x0138	SMHC Extended Command Register
SMHC_EXT_RESP	0x013C	SMHC Extended Response Register
SMHC_DRV_DL	0x0140	SMHC Drive Delay Control Register
SMHC_SAMP_DL	0x0144	SMHC Sample Delay Control Register

Register Name	Offset	Description
SMHC_FIFO	0x0200	SMHC FIFO Register

9.2.7 Register Description

9.2.7.1 0x0000 SMHC Control Register (Default Value: 0x0000_0100)

Offset: 0x0000			Register Name: SMHC_CTRL
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	FIFO_AC_MOD FIFO Access Mode 0: DMA bus 1: AHB bus
30:13	/	/	/
12	R/W	0x0	TIME_UNIT_CMD Time unit for command line The time unit is used to calculate the command line time out value defined in RTO_LMT. 0: 1 card clock period 1: 256 card clock period
11	R/W	0x0	TIME_UNIT_DAT Time unit for data line Time unit is used to calculate the data line time out value defined in DTO_LMT. 0: 1 card clock period 1: 256 card clock period
10	R/W	0x0	DDR_MOD_SEL DDR Mode Select 0: SDR mode 1: DDR mode
9	/	/	/
8	R/W	0x1	CD_DBC_ENB Card Detect (Data[3] status) De-bounce Enable 0: Disable de-bounce 1: Enable de-bounce
7:6	/	/	/
5	R/W	0x0	DMA_ENB DMA Global Enable 0: Disable DMA to transfer data via AHB bus 1: Enable DMA to transfer data

Offset: 0x0000			Register Name: SMHC_CTRL
Bit	Read/Write	Default/Hex	Description
4	R/W	0x0	INT_ENB Global Interrupt Enable 0: Disable interrupts 1: Enable interrupts
3	/	/	/
2	R/W	0x0	DMA_RST DMA Reset
1	R/W	0x0	FIFO_RST FIFO Reset 0: No effect 1: Reset the FIFO This bit is auto-cleared after the completion of the reset operation.
0	R/W	0x0	SOFT_RST Software Reset 0: No effect 1: Reset SD/MMC controller This bit is auto-cleared after the completion of reset operation.

9.2.7.2 0x0004 SMHC Clock Control Register (Default Value: 0x0000_0000)

Offset: 0x0004			Register Name: SMHC_CLKDIV
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	MASK_DATA0 0: Do not mask data0 when update clock 1: Mask data0 when update clock
30:18	/	/	/
17	R/W	0x0	CCLK_CTRL Card Clock Output Control 0: Card clock is always on. 1: Turn off card clock when FSM is in IDLE state.
16	R/W	0x0	CCLK_ENB Card Clock Enable 0: Card Clock is off. 1: Card Clock is on.
15:8	/	/	/
7:0	R/W	0x0	CCLK_DIV Card Clock Divider n: Source clock is divided by 2*n. (n = 0 to 255)

9.2.7.3 0x0008 SMHC Timeout Register (Default Value: 0xFFFF_FF40)

Offset: 0x0008			Register Name: SMHC_TMOUT
Bit	Read/Write	Default/Hex	Description
31:8	R/W	0xffffffff	DTO_LMT Data Timeout Limit This field can set the time that the host waits for the data from the device. Ensure to communicate with the device, this field must be set to the maximum that is greater than the time N_{AC}. About the N_{AC}, the explanation is as follows: When Host read data, data transmission from the Device starts after the access time delay N_{AC} beginning from the end bit of the read command (ACMD51, CMD8, CMD17, and CMD18). When the host reads multiple block (CMD18), a next block's data transmission from the Device starts after the access time delay N_{AC} beginning from the end bit of the previous block. When the host writes data, the value is no effect.
7:0	R/W	0x40	RTO_LMT Response Timeout Limit

9.2.7.4 0x000C SMHC Bus Width Register (Default Value: 0x0000_0000)

Offset: 0x000C			Register Name: SMHC_CTYPE
Bit	Read/Write	Default/Hex	Description
31:2	/	/	/
1:0	R/W	0x0	CARD_WID Card Width 00: 1-bit width 01: 4-bit width 1x: Reserved

9.2.7.5 0x0010 SMHC Block Size Register (Default Value: 0x0000_0200)

Offset: 0x0010			Register Name: SMHC_BLKSIZE
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x200	BLK_SZ Block Size

9.2.7.6 0x0014 SMHC Byte Count Register (Default Value: 0x0000_0200)

Offset: 0x0014			Register Name: SMHC_BYTCNT
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x200	BYTE_CNT Byte counter The number of bytes to be transferred. It must be integer multiple of Block Size (BLK_SZ) for block transfers.

9.2.7.7 0x0018 SMHC Command Register (Default Value: 0x0000_0000)

Offset: 0x0018			Register Name: SMHC_CMD
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	CMD_LOAD Start Command This bit is automatically cleared when the current command is sent. If there is no response error happens, a command complete interrupt bit (CMD_OVER) will be set in the interrupt register. Do not write any other commands until this bit is cleared.
30:29	/	/	/
28	R/W	0x0	VOL_SW Voltage Switch 0: Normal command 1: Voltage switch command, set for CMD11 only.
27	R/W	0x0	BOOT_ABT Boot Abort Setting this bit will terminate the boot operation.
26	R/W	0x0	EXP_BOOT_ACK Expect Boot Acknowledge When the software sets this bit along in mandatory boot operation, the controller expects a boot acknowledge start pattern of 0-1-0 from the selected card.
25:24	R/W	0x0	BOOT_MOD Boot Mode 00: Normal command 01: Mandatory Boot operation 10: Alternate Boot operation 11: Reserved
23:22	/	/	/

Offset: 0x0018			Register Name: SMHC_CMD
Bit	Read/Write	Default/Hex	Description
21	R/W	0x0	PRG_CLK Change Clock 0: Normal command 1: Change Card Clock When this bit is set, the controller will change the clock domain and clock output. No commands will be sent.
20:16	/	/	/
15	R/W	0x0	SEND_INIT_SEQ Send Initialization 0: Normal command sending 1: Send initialization sequence before sending this command.
14	R/W	0x0	STOP_ABT_CMD Stop Abort Command 0: Normal command sending 1: Send <i>Stop</i> or <i>Abort</i> command to stop the current data transfer in progress. (CMD12, CMD52 for writing “I/O Abort” in SDIO CCCR)
13	R/W	0x0	WAIT_PRE_OVER Wait for Data Transfer Over 0: Send command at once, does not care about data transferring. 1: Wait for data transfer completion before sending the current command.
12	R/W	0x0	STOP_CMD_FLAG Send Stop CMD Automatically (CMD12) 0: Do not send stop command at the end of the data transfer. 1: Send stop command automatically at the end of the data transfer. If set, the SMHC_RESP1 will record the response of auto CMD12.
11	R/W	0x0	TRANS_MODE Transfer Mode 0: Block data transfer command 1: Stream data transfer command
10	R/W	0x0	TRANS_DIR Transfer Direction 0: Read operation 1: Write operation

Offset: 0x0018			Register Name: SMHC_CMD
Bit	Read/Write	Default/Hex	Description
9	R/W	0x0	DATA_TRANS Data Transfer 0: Without data transfer 1: With data transfer
8	R/W	0x0	CHK_RESP_CRC Check Response CRC 0: Do not check response CRC 1: Check response CRC
7	R/W	0x0	LONG_RESP Response Type 0: Short Response (48 bits) 1: Long Response (136 bits)
6	R/W	0x0	RESP_RCV Response Receive 0: Command without response 1: Command with response
5:0	R/W	0x0	CMD_IDX CMD Index Command index value

9.2.7.8 0x001C SMHC Command Argument Register (Default Value: 0x0000_0000)

Offset: 0x001C			Register Name: SMHC_CMDARG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	CMD_ARG Command argument

9.2.7.9 0x0020 SMHC Response 0 Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: SMHC_RESP0
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	CMD_RESP0 Response 0 Bit[31:0] of response

9.2.7.10 0x0024 SMHC Response 1 Register (Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: SMHC_RESP1
Bit	Read/Write	Default/Hex	Description

Offset: 0x0024			Register Name: SMHC_RESP1
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	CMD_RESP1 Response 1 Bit[63:31] of response

9.2.7.11 0x0028 SMHC Response 2 Register (Default Value: 0x0000_0000)

Offset: 0x0028			Register Name: SMHC_RESP2
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	CMD_RESP2 Response 2 Bit[95:64] of response

9.2.7.12 0x002C SMHC Response 3 Register (Default Value: 0x0000_0000)

Offset: 0x002C			Register Name: SMHC_RESP3
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	CMD_RESP3 Response 3 Bit[127:96] of response

9.2.7.13 0x0030 SMHC Interrupt Mask Register (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: SMHC_INTMASK
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	CARD_REMOVAL_INT_EN Card Removed Interrupt Enable
30	R/W	0x0	CARD_INSERT_INT_EN Card Inserted Interrupt Enable
29:17	/	/	/
16	R/W	0x0	SDIO_INT_EN SDIO Interrupt Enable
15	R/W	0x0	DEE_INT_EN Data End-bit Error Interrupt Enable
14	R/W	0x0	ACD_INT_EN Auto Command Done Interrupt Enable
13	R/W	0x0	DSE_BC_INT_EN Data Start Error Interrupt Enable
12	R/W	0x0	CB_IW_INT_EN Command Busy and Illegal Write Interrupt Enable

Offset: 0x0030			Register Name: SMHC_INTMASK
Bit	Read/Write	Default/Hex	Description
11	R/W	0x0	FU_FO_INT_EN FIFO Underrun/Overflow Interrupt Enable
10	R/W	0x0	DSTO_VSD_INT_EN Data Starvation Timeout/V1.8 Switch Done Interrupt Enable
9	R/W	0x0	DTO_BDS_INT_EN Data Timeout/Boot Data Start Interrupt Enable
8	R/W	0x0	RTO_BACK_INT_EN Response Timeout/Boot ACK Received Interrupt Enable
7	R/W	0x0	DCE_INT_EN Data CRC Error Interrupt Enable
6	R/W	0x0	RCE_INT_EN Response CRC Error Interrupt Enable
5	R/W	0x0	DRR_INT_EN Data Receive Request Interrupt Enable
4	R/W	0x0	DTR_INT_EN Data Transmit Request Interrupt Enable
3	R/W	0x0	DTC_INT_EN Data Transfer Complete Interrupt Enable
2	R/W	0x0	CC_INT_EN Command Complete Interrupt Enable
1	R/W	0x0	RE_INT_EN Response Error Interrupt Enable
0	/	/	/

9.2.7.14 0x0034 SMHC Masked Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0034			Register Name: SMHC_MINTSTS
Bit	Read/Write	Default/Hex	Description
31	R	0x0	M_CARD_REMOVAL_INT Card Removed
30	R	0x0	M_CARD_INSERT Card Inserted
29:17	/	/	/
16	R	0x0	M_SDIO_INT SDIO Interrupt

Offset: 0x0034			Register Name: SMHC_MINTSTS
Bit	Read/Write	Default/Hex	Description
15	R	0x0	M_DEE_INT Data End-bit Error When the bit is set during receiving data, it means that the host controller does not receive the valid data end bit. When the bit is set during transmitting data, it means that the host controller does not receive the CRC status token or received CRC status token is negative.
14	R	0x0	M_ACD_INT Auto Command Done When set, it means auto-stop command (CMD12) completed.
13	R	0x0	M_DSE_BC_INT Data Start Error/Busy Clear When set during receiving data, it means that the host controller found an error start bit. When the bit is set during transmitting data, it means that the busy signal is cleared after the last block.
12	R	0x0	M_CB_IW_INT Command Busy and Illegal Write
11	R	0x0	M_FU_FO_INT FIFO Underrun/Overflow
10	R	0x0	M_DSTO_VSD_INT Data Starvation Timeout/V1.8 Switch Done
9	R	0x0	M_DTO_BDS_INT Data Timeout/Boot Data Start
8	R	0x0	M_RTO_BACK_INT Response Timeout/Boot ACK Received
7	R	0x0	M_DCE_INT Data CRC Error When the bit is set during receiving data, it means that the received data have data CRC error. When the bit is set during transmitting data, it means that the received CRC status taken is negative.
6	R	0x0	M_RCE_INT Response CRC Error

Offset: 0x0034			Register Name: SMHC_MINTSTS
Bit	Read/Write	Default/Hex	Description
5	R	0x0	M_DRR_INT Data Receive Request When set, it means that there are enough data in FIFO during receiving data.
4	R	0x0	M_DTR_INT Data Transmit Request When set, it means that there is enough space in FIFO during transmitting data.
3	R	0x0	M_DTC_INT Data Transfer Complete
2	R	0x0	M_CC_INT Command Complete
1	R	0x0	M_RE_INT Response Error When set, Transmit Bit error or End Bit error or CMD Index error may occurs.
0	/	/	/

9.2.7.15 0x0038 SMHC Raw Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0038			Register Name: SMHC_RINTSTS
Bit	Read/Write	Default/Hex	Description
31	R/W1C	0x0	CARD_REMOVAL Card Removed Write 1 to clear this bit.
30	R/W1C	0x0	CARD_INSERT Card Inserted Write 1 to clear this bit.
29:17	/	/	/
16	R/W1C	0x0	SDIOI_INT SDIO Interrupt Write 1 to clear this bit.
15	R/W1C	0x0	DEE Data End-bit Error When the bit is set during receiving data, it means that the host controller does not receive the valid data end bit. When the bit is set during transmitting data, it means that the host controller does not receive the CRC status token. Write 1 to clear this bit.

Offset: 0x0038			Register Name: SMHC_RINTSTS
Bit	Read/Write	Default/Hex	Description
14	R/W1C	0x0	ACD Auto Command Done When set, it means that the auto-stop command (CMD12) is completed. Write 1 to clear this bit.
13	R/W1C	0x0	DSE_BC Data Start Error/Busy Clear When the bit is set during receiving data, it means that the host controller found an error start bit. When the bit is set during transmitting data, it means that the busy signal is cleared after the last block. Write 1 to clear this bit.
12	R/W1C	0x0	CB_IW Command Busy and Illegal Write Write 1 to clear this bit.
11	R/W1C	0x0	FU_FO FIFO Underrun/Overflow Write 1 to clear this bit.
10	R/W1C	0x0	DSTO_VSD Data Starvation Timeout/V1.8 Switch Done Write 1 to clear this bit.
9	R/W1C	0x0	DTO_BDS Data Timeout/Boot Data Start When the bit is set during receiving data, it means that some of the channel of DATA[3:0] lack of the start bit. Write 1 to clear this bit.
8	R/W1C	0x0	RTO_BACK Response Timeout/Boot ACK Received Write 1 to clear this bit.
7	R/W1C	0x0	DCE Data CRC Error When the bit is set during receiving data, it means that the received data have data CRC error. When the bit is set during transmitting data, it means that the received CRC status taken is negative. Write 1 to clear this bit.
6	R/W1C	0x0	RCE Response CRC Error Write 1 to clear this bit.

Offset: 0x0038			Register Name: SMHC_RINTSTS
Bit	Read/Write	Default/Hex	Description
5	R/W1C	0x0	DRR Data Receive Request When set, it means that there are enough data in FIFO during receiving data. Write 1 to clear this bit.
4	R/W1C	0x0	DTR Data Transmit Request When set, it means that there is enough space in FIFO during transmitting data. Write 1 to clear this bit.
3	R/W1C	0x0	DTC Data Transfer Complete When set, it means that the current command completes even through error occurs. Write 1 to clear this bit.
2	R/W1C	0x0	CC Command Complete When set, it means that the current command completes even through error occurs. Write 1 to clear this bit.
1	R/W1C	0x0	RE Response Error When set, it means that the transmit bit error, end bit error, or CMD index error may occur. Write 1 to clear this bit.
0	/	/	/

9.2.7.16 0x003C SMHC Status Register (Default Value: 0x0000_0006)

Offset: 0x003C			Register Name: SMHC_STATUS
Bit	Read/Write	Default/Hex	Description
31	R	0x0	DMA_REQ DMA Request DMA request signal state
30:26	/	/	/
25:17	R	0x0	FIFO_LEVEL FIFO Level Number of filled locations in FIFO

Offset: 0x003C			Register Name: SMHC_STATUS
Bit	Read/Write	Default/Hex	Description
16:11	R	0x0	RESP_IDX Response Index Index of previous response, including any auto-stop sent by the controller.
10	R	0x0	FSM_BUSY Data FSM Busy Data transmit or receive state-machine is busy.
9	R	0x0	CARD_BUSY Card Data Busy Inverted version of DATA[0] 0: Card data is not busy. 1: Card data is busy.
8	R	0x0	CARD_PRESENT Data[3] Status The level of DATA[3], checks whether the card is present. 0: The card is not present. 1: The card is present.
7:4	R	0x0	FSM_STA Command FSM States 0000: Idle 0001: Send init sequence 0010: TX CMD start bit 0011: TX CMD TX bit 0100: TX CMD index + argument 0101: TX CMD CRC7 0110: TX CMD end bit 0111: RX response start bit 1000: RX response IRQ response 1001: RX response TX bit 1010: RX response CMD index 1011: RX response data 1100: RX response CRC7 1101: RX response end bit 1110: CMD path wait NCC 1111: Wait; CMD-to-response turn around
3	R	0x0	FIFO_FULL FIFO Full 0: FIFO is not full 1: FIFO is full

Offset: 0x003C			Register Name: SMHC_STATUS
Bit	Read/Write	Default/Hex	Description
2	R	0x1	FIFO_EMPTY FIFO Empty 0: FIFO is not empty 1: FIFO is empty
1	R	0x1	FIFO_TX_LEVEL FIFO TX Water Level Flag 0: FIFO does not reach the transmit trigger level 1: FIFO reaches the transmit trigger level
0	R	0x0	FIFO_RX_LEVEL FIFO RX Water Level Flag 0: FIFO does not reach the receive trigger level. 1: FIFO reaches the receive trigger level.

9.2.7.17 0x0040 SMHC FIFO Water Level Register (Default Value: 0x000F_0000)

Offset: 0x0040			Register Name: SMHC_FIFOTH
Bit	Read/Write	Default/Hex	Description
31	/	/	/
30:28	R/W	0x0	BSIZE_OF_TRANS Burst Size of Multiple Transaction 000: 1 transfers 001: 4 010: 8 011: 16 Others: Reserved It should be programmed the same as the DMA controller multiple transaction size. The units for the transfer are the DWORD. A single transfer would be signaled based on this value. The value should be sub-multiple of (RX_TL + 1) and (FIFO_DEPTH - TX_TL) Recommended: FIFO_DEPTH = 256, FIFO_SIZE = 256 * 32 = 1K MSize = 16, TX_TL = 240, RX_TL = 15
27:24	/	/	/

Offset: 0x0040			Register Name: SMHC_FIFOTH
Bit	Read/Write	Default/Hex	Description
23:16	R/W	0xF	RX_TL RX Trigger Level 0x0 to 0xFE: The RX trigger level is from 0 to 254. 0xFF: Reserved Indicates the FIFO threshold for the FIFO request host to receive data from the FIFO. When the FIFO data level is greater than this value, the DMA request is raised if DMA enabled, or the RX interrupt bit is set if interrupt enabled. At the end of the packet, if the last transfer is less than this level, the value is ignored and the relative request will be raised as usual. Recommended: 15 (means greater than 15)
15:8	/	/	/
7:0	R/W	0x0	TX_TL TX Trigger Level 0x1 to 0xFF: The TX trigger level is 1 to 255. 0x0: No trigger Indicates the FIFO threshold for the FIFO request host to transmit data to the FIFO. When the FIFO data level is less than or equal to this value, the DMA TX request is raised if DMA enabled, or TX request interrupt bit is set if interrupt enabled. At the end of the packet, if the last transfer is less than this level, the value is ignored and the relative request will be raised as usual. Recommended: 240 (means less than or equal to 240)

9.2.7.18 0x0044 SMHC Function Select Register (Default Value: 0x0000_0000)

Offset: 0x0044			Register Name: SMHC_FUNS
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/

Offset: 0x0044			Register Name: SMHC_FUNS
Bit	Read/Write	Default/Hex	Description
2	R/W	0x0	ABT_RDATA Abort Read Data 0: Ignored 1: After the suspend command is issued during the read-transfer, the software polls card to find when the suspend happens. Once the suspend occurs, the software sets the bit to reset the data state-machine, which is waiting for the next block of data. This bit is used in the SDIO card suspends sequence and is auto-cleared once the controller resets to the idle state.
1	R/W	0x0	READ_WAIT Read Wait 0: Clear SDIO read wait 1: Assert SDIO read wait
0	R/W	0x0	HOST_SEND_MMC_IRQRESQ Host Send MMC IRQ Response 0: Ignored 1: Send auto IRQ response When the host is waiting for the MMC card interrupt response, setting this bit will make the controller cancel the waiting state and return to the idle state, at which time, the controller will receive the IRQ response sent by itself. This bit is auto-cleared after the response is sent.

9.2.7.19 0x0048 SMHC Transferred Byte Count Register 0 (Default Value: 0x0000_0000)

Offset: 0x0048			Register Name: SMHC_TBC0
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	TBC0 Transferred Count 0 The number of bytes transferred between the card and internal FIFO. The register should be accessed in full to avoid read-coherency problems and read only after the data transfer completes.

9.2.7.20 0x004C SMHC Transferred Byte Count Register 1 (Default Value: 0x0000_0000)

Offset: 0x004C			Register Name: SMHC_TBC1

Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	TBC1 Transferred Count 1 The number of bytes transferred between the Host/DMA memory and internal FIFO. The register should be accessed in full to avoid read-coherency problems and read only after the data transfer completes.

9.2.7.21 0x0058 SMHC Auto Command 12 Argument Register (Default Value: 0x0000_FFFF)

Offset: 0x0058			Register Name: SMHC_A12A
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0xffff	SD_A12A Auto CMD12 Argument The argument of command 12 automatically sent by the controller.

9.2.7.22 0x005C SMHC New Timing Set Register (Default Value: 0x8171_0000)

Offset: 0x005C			Register Name: SMHC_NTZR
Bit	Read/Write	Default/Hex	Description
31	R/W	0x1	MODE_SELECT 0: Old mode of Sample/Output Timing 1: New mode of Sample/Output Timing
30:26	/	/	/
25	R/W	0x0	BOOT_DAT_RX_PHASE_CLR After boot ack, before receiving boot data, clear data lines' input phase. 0: Disable 1: Enable
24	R/W	0x1	CMD_DAT_RX_PHASE_CLR Clear the input phase of command lines and data lines during the update clock operation. 0: Disabled 1: Enabled
23	/	/	/
22	R/W	0x1	DAT_CRC_STATUS_RX_PHASE_CLR Clear the input phase of data lines before receiving the CRC status. 0: Disabled 1: Enabled

Offset: 0x005C			Register Name: SMHC_NTSR
Bit	Read/Write	Default/Hex	Description
21	R/W	0x1	DAT_TRANS_RX_PHASE_CLR Clear the input phase of data lines before transferring the data. 0: Disabled 1: Enabled
20	R/W	0x1	DAT_RECV_RX_PHASE_CLR Clear the input phase of data lines before receiving the data. 0: Disabled 1: Enabled
19:17	/	/	/
16	R/W	0x1	CMD_SEND_RX_PHASE_CLR Clear command rx phase before sending the command. 0: Disabled 1: Enabled
15:10	/	/	/
9:8	R/W	0x0	DAT_SAMPLE_TIMING_PHASE 00: Sample timing phase offset 90° 01: Sample timing phase offset 180° 10: Sample timing phase offset 270° 11: Reserved
7:6	/	/	/
5:4	R/W	0x0	CMD_SAMPLE_TIMING_PHASE 00: Sample timing phase offset 90° 01: Sample timing phase offset 180° 10: Sample timing phase offset 270° 11: Ignore
3:0	/	/	/

9.2.7.23 0x0078 SMHC Hardware Reset Register (Default Value: 0x0000_0001)

Offset: 0x0078			Register Name: SMHC_HWRST
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x1	HW_RST 1: Active mode 0: Reset These bits cause the cards to enter the pre-idle state, which requires them to be re-initialized.

9.2.7.24 0x0080 SMHC IDMAC Control Register (Default Value: 0x0000_0000)

Offset: 0x0080			Register Name: SMHC_IDMAC
Bit	Read/Write	Default/Hex	Description
31	W	0x0	DES_LOAD_CTRL When IDMAC fetches a descriptor, if the valid bit of a descriptor is not set, IDMAC FSM will go to the suspend state. Setting this bit will make the IDMAC refetch descriptor again and do the transfer normally.
30:8	/	/	/
7	R/W	0x0	IDMAC_ENB IDMAC Enable When set, the IDMAC is enabled.
6:2	/	/	/
1	R/W	0x0	FIX_BUST_CTRL Fixed Burst Controls whether the AHB Master interface performs fixed burst transfers or not. When set, the AHB will use only SINGLE, INCR4, and INCR8 during the start of normal burst transfers. When reset, the AHB will use SINGLE and INCR burst transfer operations.
0	R/W	0x0	IDMAC_RST DMA Reset When set, the DMA Controller resets all its internal registers. It is automatically cleared after 1 clock cycle.

9.2.7.25 0x0084 SMHC Descriptor List Base Address Register (Default Value: 0x0000_0000)

Offset: 0x0084			Register Name: SMHC_DLBA_REG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	DES_BASE_ADDR Start of Descriptor List Contains the base address of the First Descriptor. It is a word (4 Byte) address.

9.2.7.26 0x0088 SMHC IDMAC Status Register (Default Value: 0x0000_0000)

Offset: 0x0088			Register Name: SMHC_IDST_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16:13	R	0x0	Reserved

Offset: 0x0088			Register Name: SMHC_IDST_REG
Bit	Read/Write	Default/Hex	Description
12:10	R	0x0	IDMAC_ERR_STA Error Bits Indicates the type of error that caused a Bus Error. Valid only with Fatal Bus Error bit (IDST[2]) set. This field does not generate an interrupt. 001: Host Abort received during the transmission. 010: Host Abort received during the reception. Others: Reserved
9	R/W1C	0x0	ABN_INT_SUM (AIS) Abnormal Interrupt Summary Logical OR of the following: IDST[2]: Fatal Bus Interrupt IDST[4]: Descriptor Unavailable Bit Interrupt IDST[5]: Card Error Summary Interrupt Only unmasked bits affect this bit. This is a sticky bit and must be cleared each time a corresponding bit that causes AIS to be set is cleared. Write "1" to clear this bit.
8	R/W1C	0x0	NOR_INT_SUM (NIS) Normal Interrupt Summary Logical OR of the following: IDST[0]: Transmit Interrupt IDST[1]: Receive Interrupt Only unmasked bits affect this bit. This is a sticky bit and must be cleared each time a corresponding bit that causes NIS to be set is cleared. Write "1" to clear this bit.
7:6	/	/	/
5	R/W1C	0x0	ERR_FLAG_SUM Card Error Summary Indicates the status of the transaction to/from the card; also present in RINTSTS. Indicates the logical OR of the following bits: EBE: End Bit Error RTO: Response Timeout RCRC: Response CRC SBE: Start Bit Error DRTO: Data Read Timeout DCRC: Data CRC for Receive RE: Response Error Writing 1 clears this bit.

Offset: 0x0088			Register Name: SMHC_IDST_REG
Bit	Read/Write	Default/Hex	Description
4	R/W1C	0x0	DES_UNAVL_INT Descriptor Unavailable Interrupt This bit is set when the descriptor is unavailable due to OWN bit = 0 (DES0[31] =0). Writing 1 clears this bit.
3	/	/	/
2	R/W1C	0x0	FATAL_BERR_INT Fatal Bus Error Interrupt Indicates that a Bus Error occurred (IDST[12:10]). When this bit is set, the DMA disables all its bus accesses. Writing 1 clears this bit.
1	R/W1C	0x0	RX_INT Receive Interrupt Indicates the completion of data reception for a descriptor. Writing 1 clears this bit.
0	R/W1C	0x0	TX_INT Transmit Interrupt Indicates that data transmission is finished for a descriptor. Writing 1 clears this bit.

9.2.7.27 0x008C SMHC IDMAC Interrupt Enable Register (Default Value: 0x0000_0000)

Offset: 0x008C			Register Name: SMHC_IDIE_REG
Bit	Read/Write	Default/Hex	Description
31:6	/	/	/
5	R/W	0x0	ERR_SUM_INT_ENB Card Error Summary Interrupt Enable. When set, it enables the Card Interrupt Summary.
4	R/W	0x0	DES_UNAVL_INT_ENB Descriptor Unavailable Interrupt. When set along with Abnormal Interrupt Summary Enable, the Descriptor Unavailable Interrupt is enabled.
3	/	/	/

Offset: 0x008C			Register Name: SMHC_IDIE_REG
Bit	Read/Write	Default/Hex	Description
2	R/W	0x0	FERR_INT_ENB Fatal Bus Error Enable When set with Abnormal Interrupt Summary Enable, the Fatal Bus Error Interrupt is enabled. When reset, the Fatal Bus Error Enable Interrupt is disabled.
1	R/W	0x0	RX_INT_ENB Receive Interrupt Enable. When set with Normal Interrupt Summary Enable, Receive Interrupt is enabled. When reset, the Receive Interrupt is disabled.
0	R/W	0x0	TX_INT_ENB Transmit Interrupt Enable. When set with Normal Interrupt Summary Enable, Transmit Interrupt is enabled. When reset, the Transmit Interrupt is disabled.

9.2.7.28 0x0100 SMHC Card Threshold Control Register (Default Value: 0x0000_0000)

Offset: 0x0100			Register Name: SMHC_THLD
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0x0	CARD_WR_THLD Card Read/Write Threshold Size
15:3	/	/	/
2	R/W	0x0	CARD_WR_THLD_ENB Card Write Threshold Enable 0: Card write threshold disabled 1: Card write threshold enabled Host controller initiates write transfer only if the card threshold amount of data is available in transmit FIFO.
1	R/W	0x0	BCIG Busy Clear Interrupt Generation 0: Busy clear interrupt disabled 1: Busy clear interrupt enabled The application can disable this feature if it does not want to wait for a Busy Clear Interrupt.

Offset: 0x0100			Register Name: SMHC_THLD
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	CARD_RD_THLD_ENB Card Read Threshold Enable 0: Card read threshold disabled 1: Card read threshold enabled Host controller initiates Read Transfer only if the CARD_RD_THLD amount of space is available in receive FIFO.

9.2.7.29 0x0104 SMHC Sample FIFO Control Register (Default Value: 0x0000_0006)

Offset: 0x0104			Register Name: SMHC_SFC
Bit	Read/Write	Default/Hex	Description
31:5	/	/	/
4:1	R/W	0x3	STOP_CLK_CTRL Stop Clock Control When receiving data, if CARD_RD_THLD_ENB is set and CARD_RD_THLD is set the same with BLK_SZ, the device clock may stop at the block gap during data receiving. This field is used to control the position of the stopping clock. The value can be changed between 0x0 and 0xF, but actually, the available value and the position of the stopping clock must be decided by the actual situation. The value increases one in this field is linked to one cycle (two cycles in DDR mode) that the position of the stopping clock moved up.
0	R/W	0x0	BYPASS_EN Bypass enable When set, sample FIFO will be bypassed.

9.2.7.30 0x0108 SMHC Auto Command 23 Argument Register (Default Value: 0x0000_0000)

Offset: 0x0108			Register Name: SMHC_A23A
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	A23A Auto CMD23 Argument The argument of command 23 is automatically sent by controller with this field.

9.2.7.31 0x010C SMHC eMMC4.5 DDR Start Bit Detection Control Register (Default Value: 0x0000_0000)

Offset: 0x010C			Register Name: EMMC_DDR_SBIT_DET
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	HALF_START_BIT Control for start bit detection mechanism inside mstorage based on duration of start bit. For eMMC 4.5, start bit can be: 0: Full cycle 1: Less than one full cycle Set HALF_START_BIT = 1 for eMMC 4.5 and above; set to 0 for SD applications.

9.2.7.32 0x0138 SMHC Extended Command Register (Default Value: 0x0000_0000)

Offset: 0x0138			Register Name: SMHC_EXT_CMD
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	AUTO_CMD23_EN Send CMD23 Automatically When setting this bit, send CMD23 automatically before sending the command specified in the SMHC_CMD register. When SOFT_RST is set, this field will be cleared.

9.2.7.33 0x013C SMHC Extended Response Register (Default Value: 0x0000_0000)

Offset: 0x013C			Register Name: SMHC_EXT_RESP
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	SMHC_EXT_RESP When AUTO_CMD23_EN is set, this register stores the response of CMD23.

9.2.7.34 0x0140 SMHC Drive Delay Control Register (Default Value: 0x0001_0000)

Offset: 0x0140			Register Name: SMHC_DRV_DL
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/

Offset: 0x0140			Register Name: SMHC_DRV_DL
Bit	Read/Write	Default/Hex	Description
17	R/W	0x0	DAT_DRV_PH_SEL Data Drive Phase Select When 0x005C[31]=0: 0: Data drive phase offset is: 90° at SDR mode 90° at DDR4 mode. 1: Data drive phase offset is: 180° at SDR mode 0° at DDR4 mode. When 0x005C[31]=1: 0: Data drive phase offset is 90° at SDR mode 45° at DDR4 (2x) mode. 1: Data drive phase offset is 180° at SDR mode 90° at DDR4 (2x) mode.
16	R/W	0x1	CMD_DRV_PH_SEL Command Drive Phase Select When 0x005C[31]=0: 0: Command drive phase offset is 90° at SDR mode 90° at DDR4 mode. 1: Command drive phase offset is 180° at SDR mode 180° at DDR4 mode. When 0x005C[31]=1: 0: Command drive phase offset is 90° at SDR mode 45° at DDR4 (2x) mode. 1: Command drive phase offset is 180° at SDR mode 90° at DDR4 (2x) mode.
15:0	/	/	/

9.2.7.35 0x0144 SMHC Sample Delay Control Register (Default Value: 0x0000_2000)

Offset: 0x0144			Register Name: SMHC_SAMP_DL
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/

Offset: 0x0144			Register Name: SMHC_SAMP_DL
Bit	Read/Write	Default/Hex	Description
15	R/W	0x0	SAMP_DL_CAL_START Sample Delay Calibration Start When set, it means that start sample delay chain calibration.
14	R	0x0	SAMP_DL_CAL_DONE Sample Delay Calibration Done When set, it means that sample delay chain calibration is done and the result of calibration is shown in SAMP_DL.
13:8	R	0x20	SAMP_DL Sample Delay 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. Generally, it is necessary to do drive delay calibration when the card clock is changed. This bit is valid only when SAMP_DL_CAL_DONE is set.
7	R/W	0x0	SAMP_DL_SW_EN Sample Delay Software Enable When set, it means that enable the sample delay specified at SAMP_DL_SW.
6	/	/	/
5:0	R/W	0x0	SAMP_DL_SW Sample Delay Software The relative delay between the clock line and command line, data line. 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.

9.2.7.36 0x0200 SMHC FIFO Register (Default Value: 0x0000_0000)

Offset: 0x0200			Register Name: SMHC_FIFO
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	TX/RX_FIFO Data FIFO

9.3 SPI Flash Controller (SPIF)

9.3.1 Overview

The SPI Flash Controller (SPIF) is a synchronous, serial communication interface which allows rapid data communication with fewer software interrupts. Different from SPI, this IP is typically designed for higher speed Flash devices and it only works at Master mode.

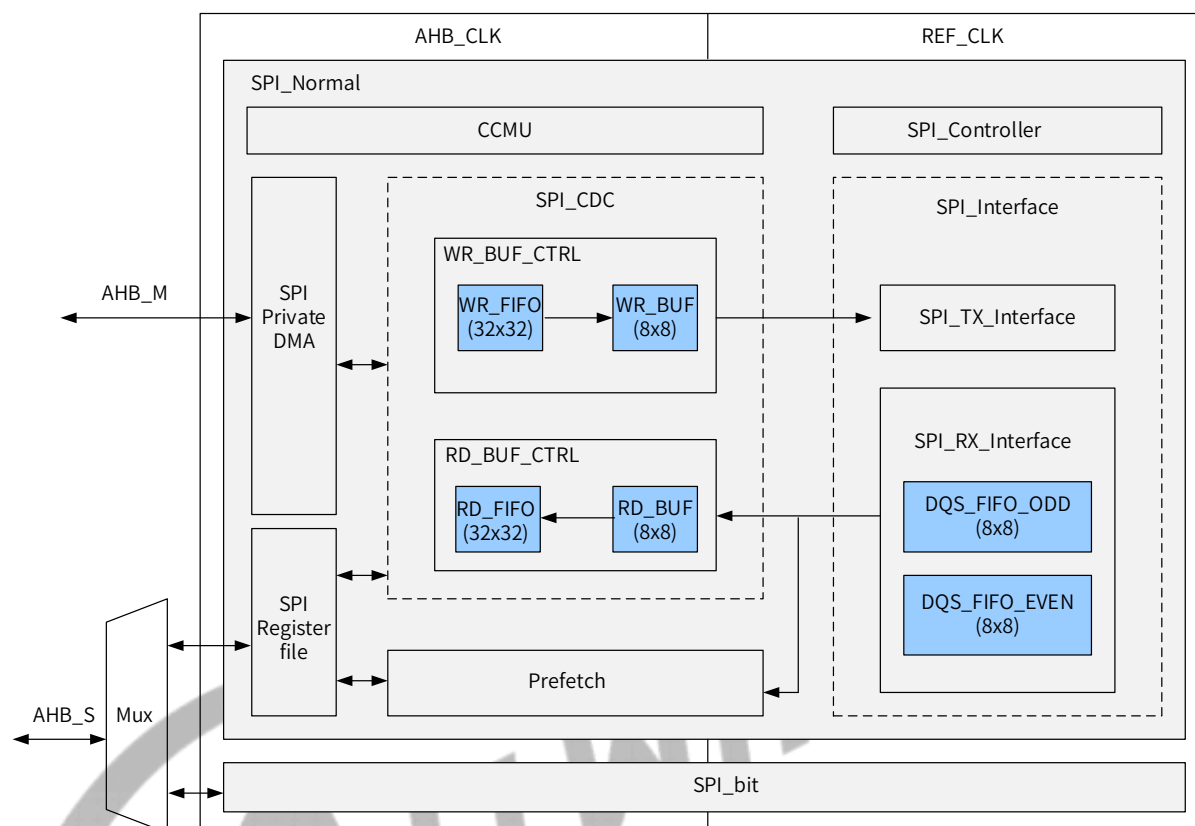
The SPI_Flash has the following features:

- Supports multiple SPI mode
 - Standard SPI
 - Dual-Output/Dual-Input SPI and Dual I/O SPI
 - Quad-Output/Quad-Input SPI, Quad I/O, and QPI
 - Programmable serial data frame length: 1 bit to 32 bits
- Supports XIP mode of operation for executing code from FLASH memory
- Supports Single Transfer Rate (STR) mode and Dual Transfer Rate (DTR) mode, and DTR mode supports DQS signal
- High Speed Clock Frequency
 - 100MHz for STR Mode
 - 100MHz for DTR Mode
- Software Write Protection
 - Write protection for all/portion of memory via software
 - Top/Bottom Block protection
- Programmable delay between transactions
- Support Mode0, Mode1, Mode2 and Mode3
- Support control signal configuration
 - Four chip select to support multiple peripherals
 - Polarity and phase of the Chip Select (SPIF_CS) and SPI Clock (SPIF_CLK) are configurable
- Supports data scramble/descramble
- Supports private DMA
 - Supports chain transmission
 - Supports descriptor in 8-word length
- Supports byte transfer

9.3.2 Block Diagram

The following figure shows a block diagram of the SPI_Flash.

Figure 9-12 SPI_Flash Block Diagram



SPI_Flash contains the following sub-blocks:

Table 9-10 SPI_Flash Sub-blocks

Sub-block	Description
SPI_Register_File	It is responsible for implementing registers configuration through AHB1 bus and CPU data prefetching.
SPI_Private_DMA	It is the private DMA for SPI, which contains AHB master and supports TXFIFO(WR_FIFO) and RXFIFO(RD_FIFO).
CMU	It is responsible for generating internal clock and communication clock. In addition, it Implements communication clock delay, clock sources selection, clock gating, and scan. It also supports synchronous release of asynchronous reset and scan.
SPI_CDC	SPI clock-domain-cross module, in which bit width transition finishes. (TX: 32bit->8bit; RX: 8bit->32bit) WR_BUF: Cache the write SPI data of AHB, and write them to WR_FIFO. RD_BUF: Cache the read SPI data of RD_FIFO to AHB. WR_FIFO: SPI write data FIFO (SRAM 32x32) RD_FIFO: SPI read data FIFO (SRAM 32x32) in the normal mode.
Prefetch	When prefetching instructions, the CPU can directly access SPI through this module.

Sub-block	Description
SPI_Controller	SPI controlling center. It generates TX/RX controlling signal in the course of SPI communication.
SPI_Interface	Standard SPI interface. It is responsible for receiving and transmitting data with Devices.
SPI_bit	the processing module in SPI 3-wire mode.

9.3.3 External Signals

The following table describes the external signals of SPI_Flash. When using SPI_Flash, the corresponding PADS are selected as SPI_Flash function via 15.7 GPIO.

Table 9-11 SPI_Flash External Signals

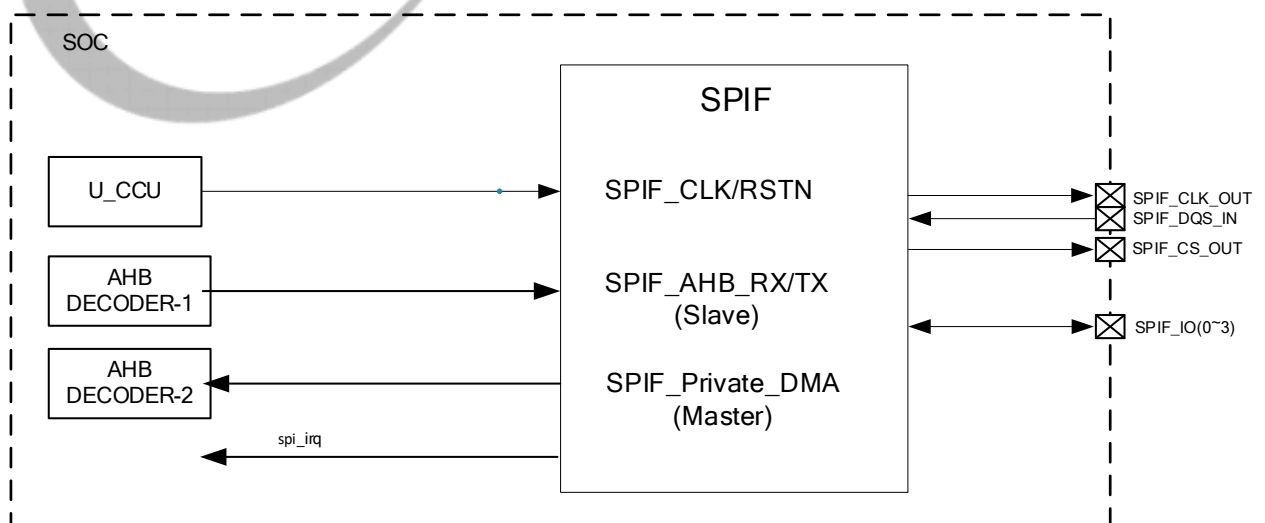
Signal Name	Description	Type
SPIF0-CS0	SPIF0 Peripheral Chip Select Signal 0, Low Active	O
SPIF0-CS1	SPIF0 Peripheral Chip Select Signal 1, Low Active	O
SPIF0-CLK	SPIF0 Master Mode Clock Output	O
SPIF0-MOSI	SPIF0 Master Data Out, Slave Data In	I/O
SPIF0-MISO	SPIF0 Master Data In, Slave Data Out	I/O
SPIF0-WP	SPIF0 Write Protection (Active Low)	I/O
SPIF0-HOLD	SPIF0 Hold Signal	I/O

9.3.4 Functional Description

9.3.4.1 Typical Application

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

Figure 9-13 Typical Application



The SPI_Flash is running in Master device. SPIF_CLK is generated and transmitted to external device. The data from the TX FIFO is routed to the MOSI pin. The data from slave is received on the MISO pin and sent to RX FIFO. Chip Select(SPIF_CS) signal is active in low level, which should be set to low before data are transmitted or received.

9.3.4.2 SPI_Flash Support List

Table 9-12 SPI_Flash Support List

SPI mode		Specification Items					
		STR	DTR	DTR-RX-DQS	SCK_MODE	Address Size	Remark
Bit Mode	3-wire	√	×	×	mode0	×	/
	4-wire						
Standard SPI	1-1-1-1	√	√	√	- mode 0/1/2/3 (STR) - mode0 (DTR)	24bit/32 bit	1-1-1-1: cmd-addr-m ode-data. - mode is optional and can be turned off.
Dual SPI	1-1-1-2	√	√	√			/
	1-1-2-2						
	1-2-2-2						
	2-2-2-2						
Quad SPI	1-1-1-4	√	√	√			/
	1-1-4-4						
	1-4-4-4						
	4-4-4-4						

9.3.4.3 SPI Clock

The SPI includes two clock domains: ahb_clk and spi_clk:

- ahb_clk supports parameter analysis (ahb_register) and DMA.
- spi_clk supports the communication between cross-domain clock, main control unit, and SPI-TX/RX and external devices.

The SPI_CMU divides the external SPI reference clock to get the SPI internal clock, which will be handled to get the sckt/sckr in accordance with the clock properties configured by the CPU. Then, the sckt/sckr will be used by the SPI_TX_INTERFACE/SPI_RX_INTERFACE.

The clock properties include:

- SPI Clock Mode
- DQS EN
- STR or DTR

SPI Clock Mode

The SPI Flash supports 4 different modes for data transfer. Software can select one of the four modes in which the SPI works by setting the bit5(SPI_CPOL) and bit4(SPI_CPHA) of [SPI Global Control Register](#)[0x0004].

The SPI_CPOL defines the signal polarity when SPIF_CLK is in the idle state. The SPIF_CLK is high level when POL is '1' and it is low level when POL is '0'. The SPI_CPHA decides whether the leading edge of SPIF_CLK is used for setup or sample data. The leading edge is used for setup data when PHA is '1' and for sample data when PHA is '0'. The four kind of modes are listed Table:

Table 9-13 SPI Flash Modes with Clock Polarity and Phase

SPI Mode	POL	PHA	Leading Edge	Trailing Edge
0	0	0	Rising, Sample	Falling, Setup
1	0	1	Rising, Setup	Falling, Sample
2	1	0	Falling, Sample	Rising, Setup
3	1	1	Falling, Setup	Rising, Sample

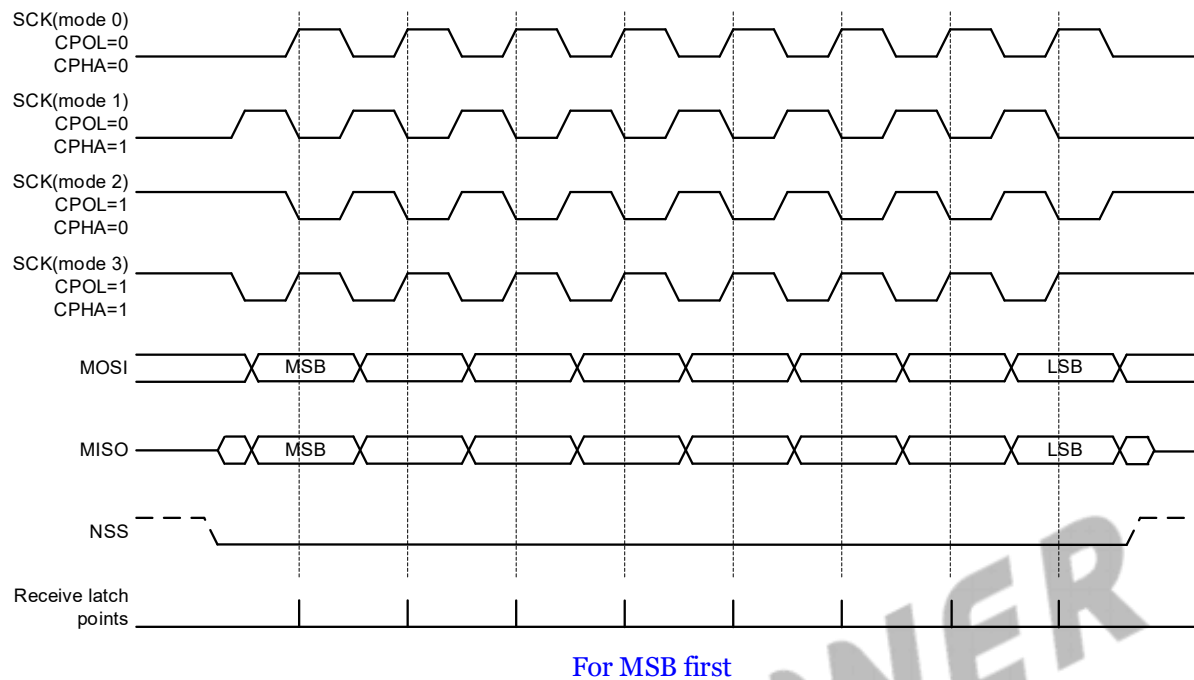
During Phase 0, Polarity 0 and Phase 1, Polarity 1 operations, output data changes on the falling edge and input data is shifted in on the rising edge.

During Phase 1, Polarity 0 and Phase 0, Polarity 1 operations, output data changes on the rising edges and is shifted in on falling edges.

The Bit Mode only supports Phase 0, Polarity 0 operation; the most significant bit (MSB) of data is transmitted first which is not configurable.

The following figure describe four waveforms for SPIF_CLK.

Figure 9-14 SPI Transfer Mode



STR & DTR

• Single Transfer Rate (STR)

In the SPI mode 0 and mode 3, the input data of devices will be latched at the rising edge of SCK, and the output data can be used at the falling edge of SCK.

Figure 9-15 SPI STR Transfer



• Double Transfer Rate (DTR)

As with the STR command, the instruction bit is latched at the rising edge of the clock in the DTR command. However, after that, the address and input data are latched at the dual edge. After the instruction bit is latched at the falling edge of SCK, the first address bit will be latched at the next rising edge of SCK. The first output data bit will be sent at the falling edge of the last access latency period.

As with the STR command, the SCK period is obtained by measuring the cycle between two adjacent SCK falling edges. In the mode 0, the SCK is already at a low level when some command starts to be executed, thus the first SCK period during the command execution indicates the cycle from the falling edge of CS# to the first falling edge of SCK.

Figure 9-16 SPI DTR Transfer Example, 1-4-4

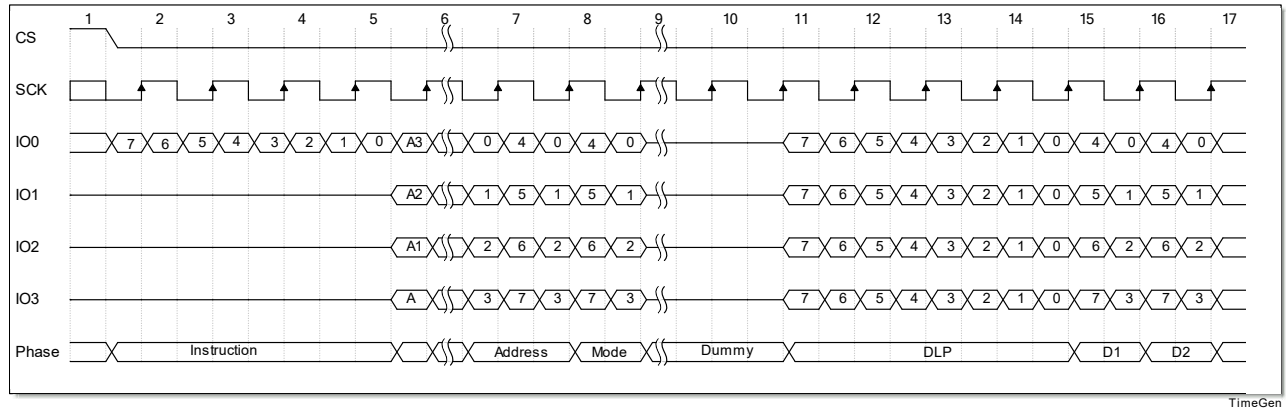
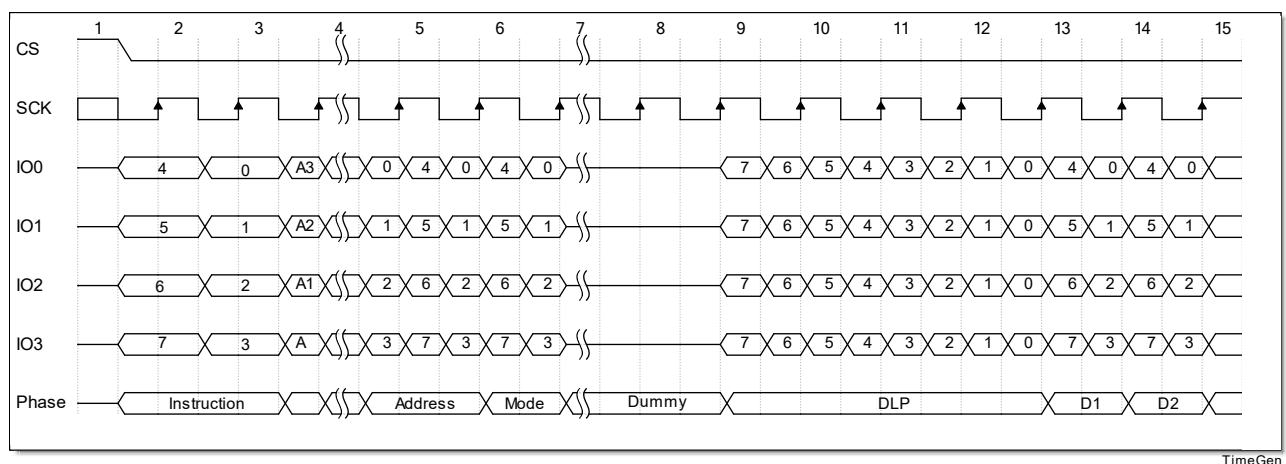


Figure 9-17 SPI DTR Transfer Example, 4-4-4



9.3.4.4 SPI Chip Select Signal

The SPI supports one chip select signal, which can be activated by the SPI_CS_SEL (bit[7:6]) of [SPI_GLOBAL_CTRL](#). The polarities of CS signals can be configured by the SPI_CS_POL (bit[5]) of [SPI_GLOBAL_CTRL](#).

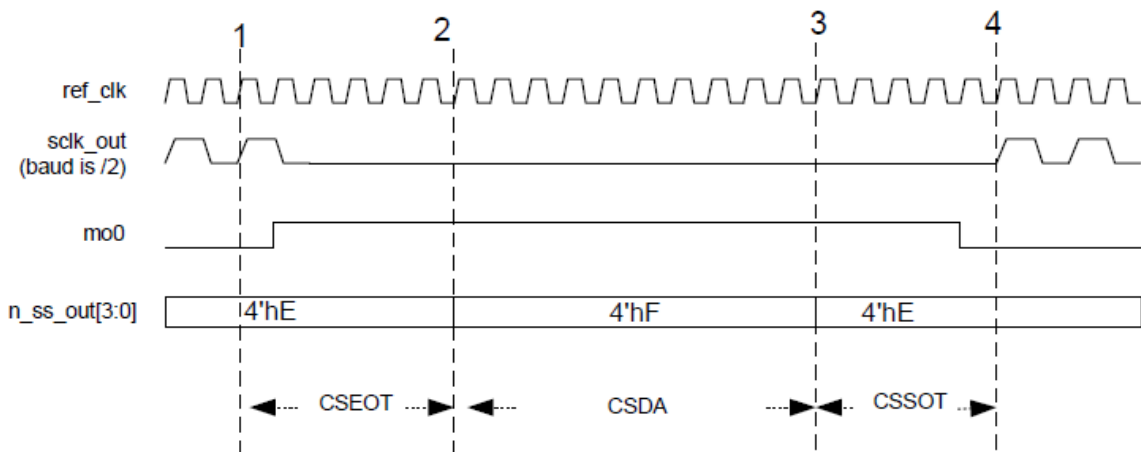
The various intervals of CS signals can be configured by [SPI_CS_DELAY](#). The specific configuration steps are as follows.

Step 1 Delay CSSOT and output the effective data after CS is enabled.

Step 2 Delay [CSEOT](#) at the rising edge of last valid data, and disable CS.

Step 3 After the CS is disabled, the interval between enabling CS and disabling CS is [CSDA](#).

Figure 9-18 SPI CS Select Timing Relation



9.3.4.5 SPI Transfer Mode

SPI supports multiple transfer modes such as SPI Bit Mode, SPI Standard Mode, SPI Dual Mode, and SPI Quad Mode. Their main differences lie in the number of wires. Even in the same transfer mode, the subdivision modes will be further derived based on the number of wires used to data transfer. The number of data lines used by Command-Address-Data is specified on the subdivision pattern heading in the manner of X-X-X. For example, one Data line for Command, two for Address, and two for data, are identified as (1-2-2).

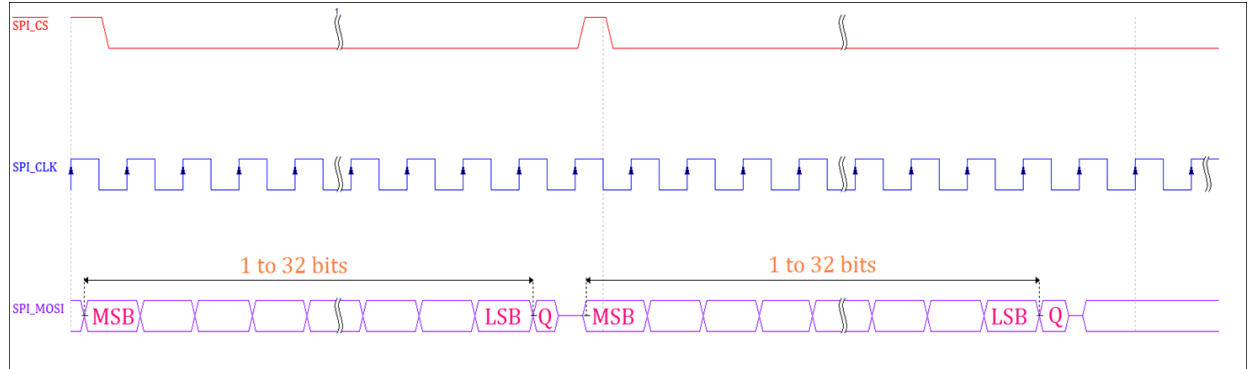
SPI Bit Mode (3-Wire/4-Wire)

For some specific scenarios, the devices such as the sensor take the SPI interfaces as the communication protocol. Generally, their data volume is relatively small, and some devices support three wires, so the SPI Bit Mode is added, and 3-wire mode and 4-wire mode are subdivided, which includes SPIF_CS, SPIF_CLK, and 1/2 data line. The transmission length ranges from 1-32bit.

The 4-Wire Mode is selected when the Work Mode Select(bit[1:0]) is equal to 0x3 in the [SPI Bit-Aligned Transfer Configure Register](#). In SPI 4-Wire Mode, the input data and output data use the independent two data lines. The MISO is used for inputting data, and the MOSI is used for outputting data.

The SPI 3-Wire Mode is only valid when the SPI controller works as Master Device, and selected when the Work Mode Select(bit[1:0]) is equal to 0x2 in the [SPI Bit-Aligned Transfer Configure Register](#). In the 3-Wire mode, the input data and the output data use the same single data line. The following figure describe this mode.

Figure 9-19 SPI 3-Wire Mode



SPI Standard Mode (4-Wire)

Signal Wire: CS#, SCK, IO0, and IO1.

Figure 9-20 SPI CMD with Single IO

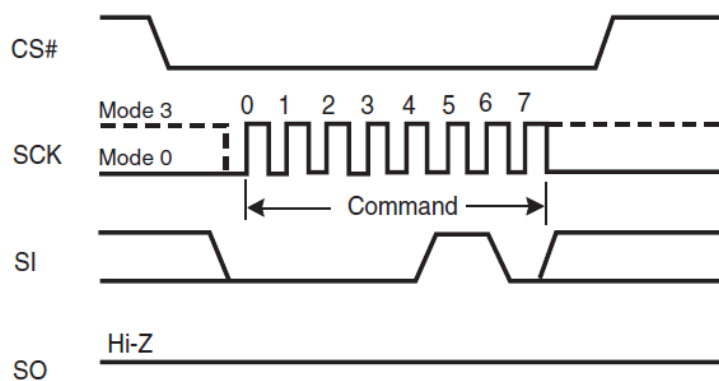


Figure 9-21 SPI Write CMD and DATA with One Wire

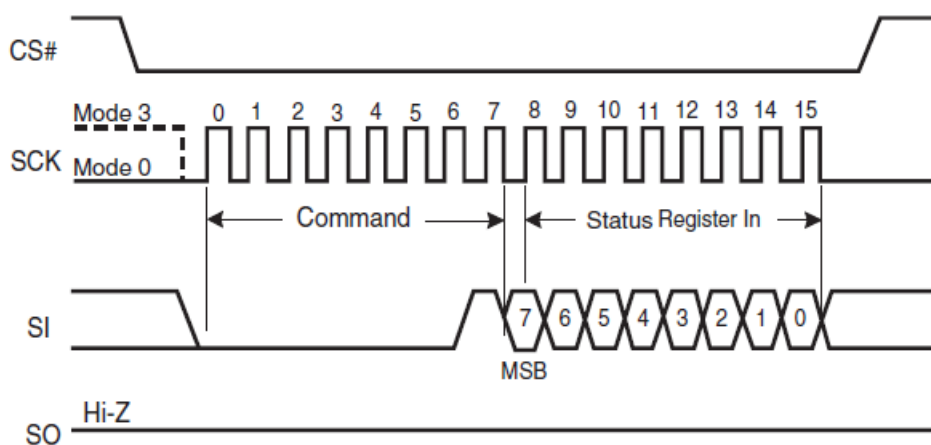


Figure 9-22 SPI Write CMD and Read DATA with One Wire (No Dummy)

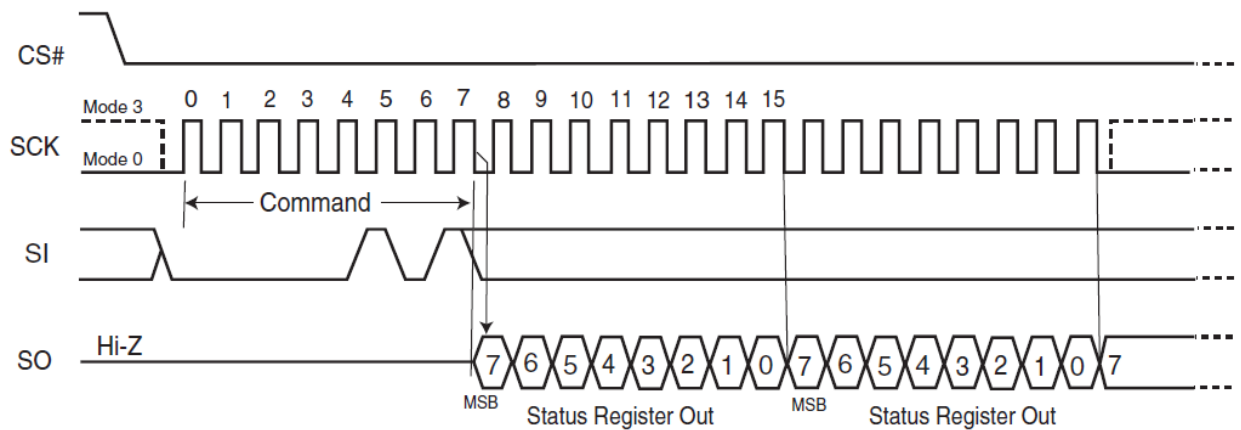
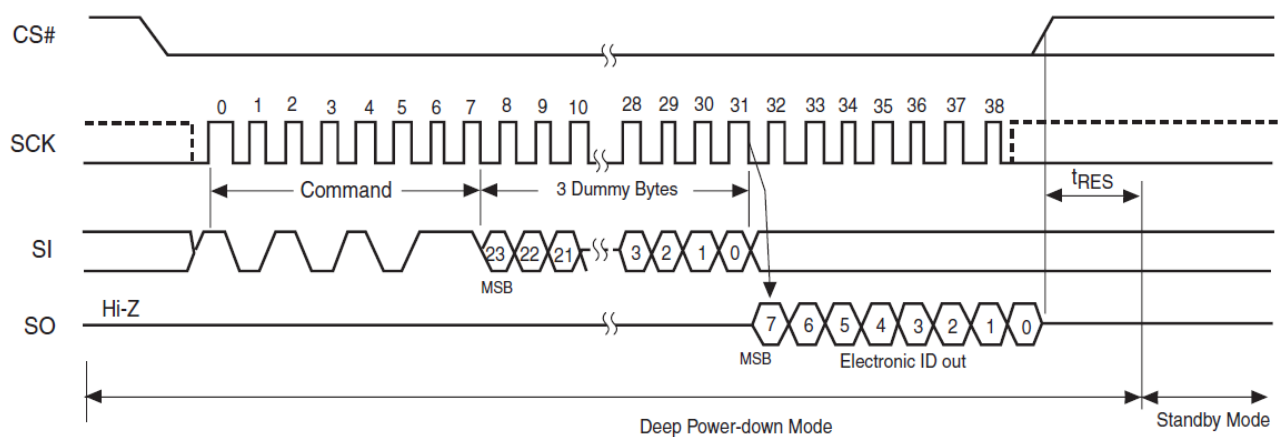


Figure 9-23 SPI Write CMD and Read DATA with One Wire (with Dummy)



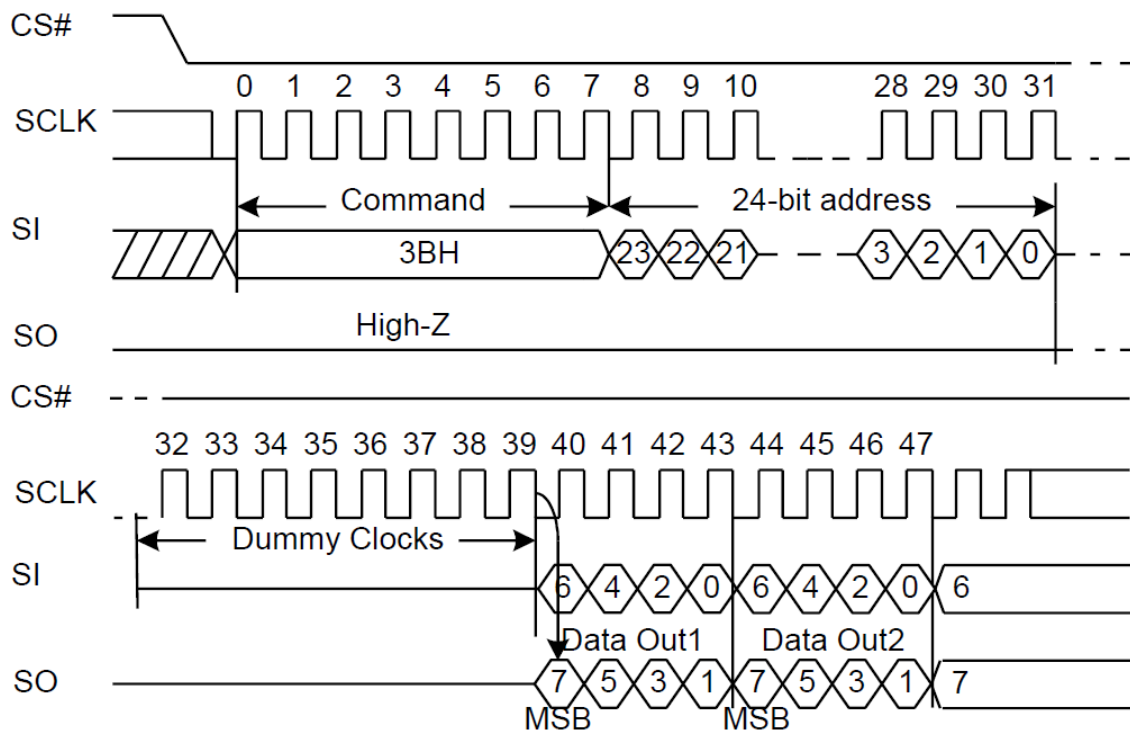
SPI Dual Mode

Using the dual mode allows data to be transferred to or from the device at two times the rate of standard single mode SPI devices. Data can be read at a faster speed using two data bits (MOSI and MISO) at a time. The following describe the Dual Input/Dual Output SPI (1-1-2), the Dual IO SPI (1-2-2), and the (2-2-2) SPI Mode.

- SPI Dual Input/Dual Output Mode (1-1-2)**

In the dual Input/dual Output SPI, the command, address, and the dummy bytes are output in the unit of a single bit in serial mode through SPIF_MOSI line. Only the data bytes are output (write) and input (read) in unit of dual bits through the SPIF_MOSI and SPIF_MISO.

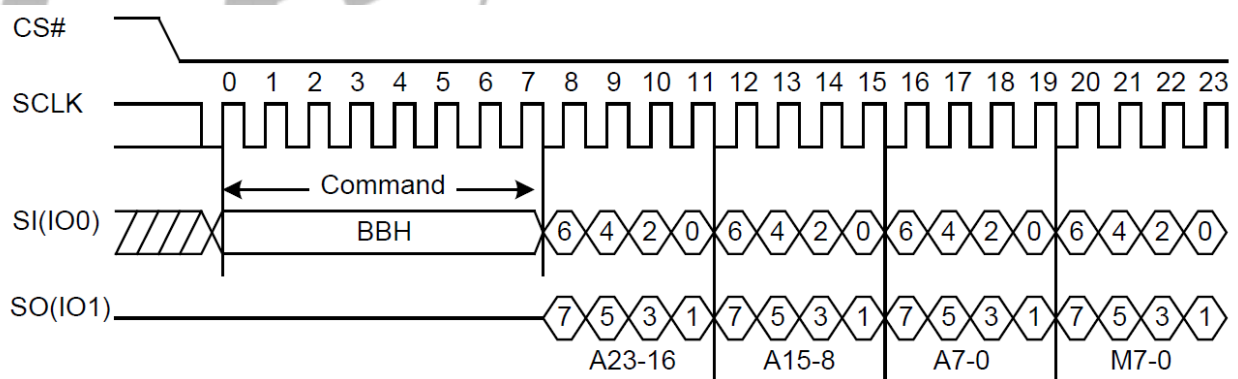
Figure 9-24 SPI Dual Input/Dual Output Mode (1-1-2)



- **SPI Dual IO Mode (1-2-2)**

In the Dual IO SPI, only the command bytes are output in the unit of a single bit in serial mode through SPIF_MOSI line. The address bytes and the dummy bytes are output in the unit of dual bits through the SPIF_MOSI and SPIF_MISO. And the data bytes are output (write) and input (read) in the unit of dual bits through the SPIF_MOSI and SPIF_MISO.

Figure 9-25 SPI Dual Input/Dual Output Mode (1-2-2)



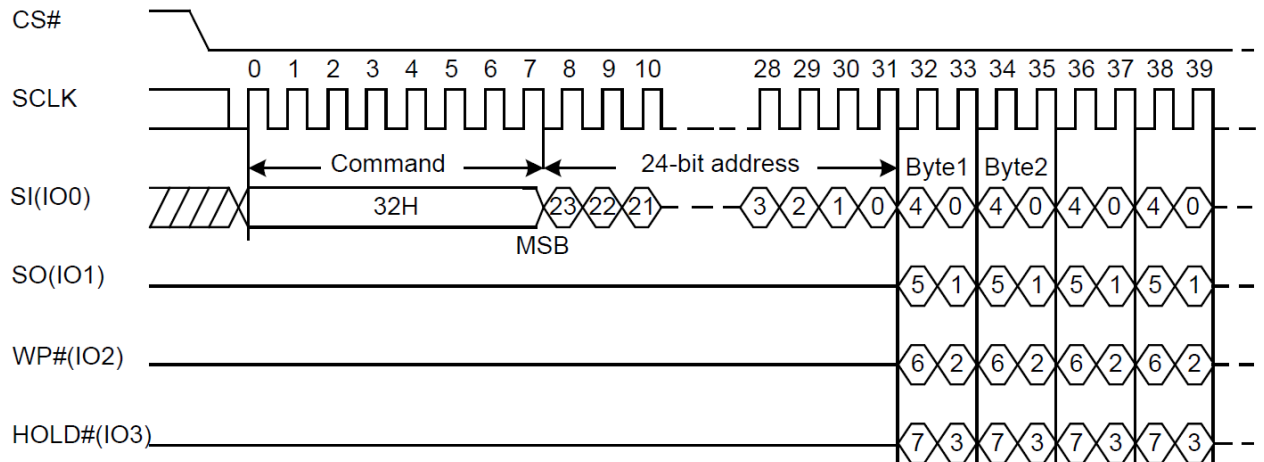
SPI Quad Mode

Using the quad mode allows data to be transferred to or from the device at 4 times the rate of standard single mode SPI devices, data can be read at fast speed using four data bits (MOSI, MISO, IO2(WP#) and IO3(HOLD#)) at the same time. The following describe the Quad Input/Quad Output SPI (1-1-4), 1-4-4 mode, and 4-4-4 mode.

- Quad Input/Quad Output SPI (1-1-4)

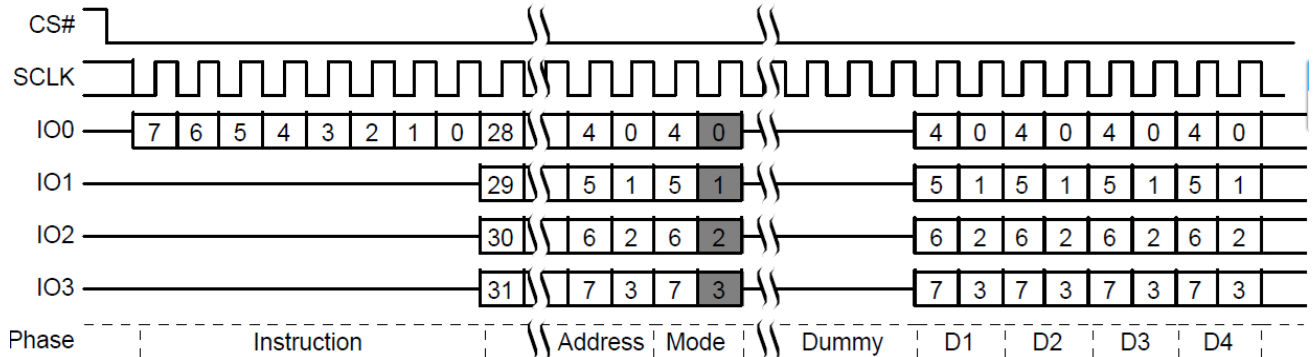
In the Quad Input/Quad Output SPI, the command, address, and the dummy bytes are output in unit of a single bit in serial mode through SPIF_MOSI line. Only the data bytes are output (write) and input(read) in unit of quad bits through the SPIF_MOSI, SPIF_MISO, SPIF_WP# and SPIF_HOLD#.

Figure 9-26 SPI Quad Input/Quad Output Mode (1-1-4)



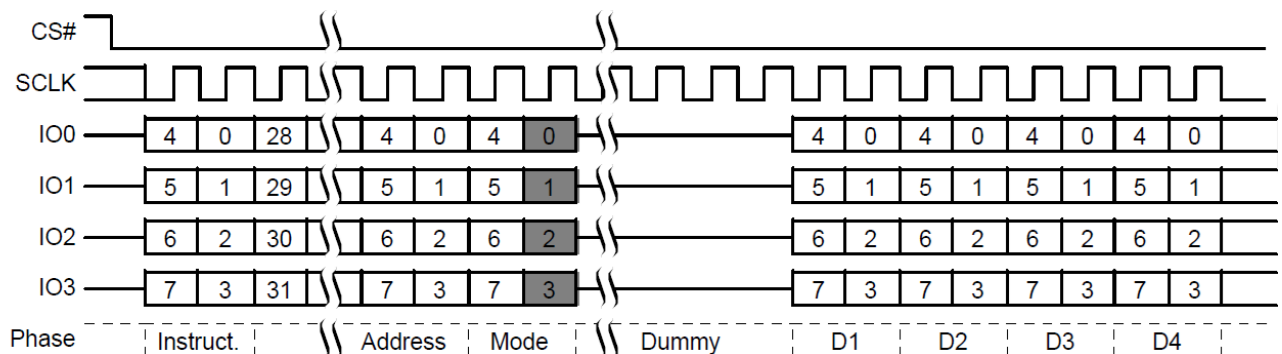
- 1-4-4 Mode

Figure 9-27 SPI Quad Input/ Quad Output Mode (1-4-4)



- 4-4-4 Mode

Figure 9-28 SPI Quad Input/ Quad Output Mode (4-4-4)



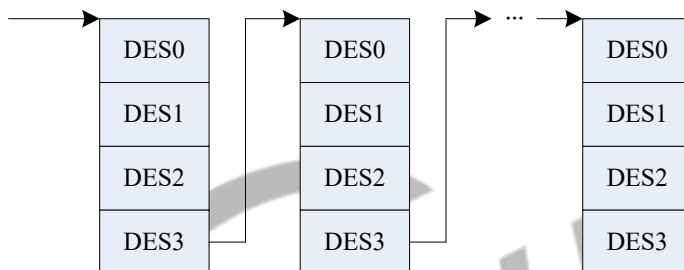
9.3.4.6 XIP Mode

If the CPU wants to directly access Flash, the prefetch mode is needed to improve efficiency. In the XIP mode, the CPU continues fetching instructions from flash and cache them (The maximum cache size is 128 Bytes.). Then HREADY_I should be controlled by handshake to send back the instructions to CPU through AHB.

9.3.4.7 SPI Flash Private DMA

The private DMA adopts the chained descriptor structure. The address and size of the first descriptor are configured by registers. After getting the first descriptor, the hardware will obtain the data from the address recorded in the descriptors. When the current transfer finishes, the hardware will continue to obtain descriptors based on the address of the next descriptor given by the previous descriptor until the terminal character is found. Then, a DMA transfer ends.

Figure 9-29 Descriptor Structure Diagram



The private DMA also supports SPI parameter configuration. The SPI transfer can be configured by the parameters of the descriptors 4-7.

Descriptor0 Definition

Bits	Descriptor
31:7	Reserved
6:4	HBURST_TYPE indicate hburst len 000: SINGLE, hburst_len = 1 011: INCR4, hburst_len = 4 101: INCR8, hburst_len = 8 111: INCR16, hburst_len = 16
3:2	/
1	DMA_DIR DMA Write Process or Read Process 0: Read 1: Write
0	DMA_FINISH_FLAG DMA Finish Flag

Descriptor1 Definition

Bits	Descriptor
31:24	DMA_BLK_LEN DMA Block Len Mode 0: 8Byte 1: 16Byte 2: 32Byte 3: 64Byte Recommended Configuration: The data volume of DMA_BLK_LEN is greater than or equal to that of HBURST_TYPE.
23:17	/
16:0	DMA_DATA_LEN Indicate the data byte number of current DMA operation.

Descriptor2 Definition

Bits	Descriptor
31:0	DMA_Buffer_SADDR The real address is as below The word address is needed, namely, the byte address abandons the low 2 bits.

Descriptor3 Definition

Bits	Descriptor
31:0	NEXT_DESCRIPTOR_ADDR These bits indicate the pointer to the physical memory where the next descriptor is present, which are word (4byte) address (The lower two bits are deleted from the byte address).

Descriptor4 Definition

Bit	Description
31:30	/
29	CMD_DTR CMD DTR Control 1: CMD DTR 0: CMD not DTR CAUTION: When CMD DTR, DTR must be on and CMD2 is required.
28	COMMAND_TRANS_EN Set to '1' if command data need trans to device SPI Controller FSM Phase Enable 1: Enable 0: Disable

Bit	Description
27:25	/
24	ADDRESS_TRANS_EN Set to '1' if address need trans to device SPI Controller FSM Phase Enable 1: Enable 0: Disable
23:21	/
20	MODE_BIT_TRANS_EN Set to '1' if Mode bit need trans after address SPI Controller FSM Phase Enable 1: Enable 0: Disable
19:17	/
16	DUMMY_BIT_TRANS_EN Dummy Bit State Enable SPI Controller FSM Phase Enable 1: Enable 0: Disable
15:13	/
12	TX_DATA_EN Set to '1' if Data Need Trans SPI Controller FSM Phase Enable 1: Enable 0: Disable
11:9	/
8	RX_DATA_EN Set to '1' if Data Need Receive SPI Controller FSM Phase Enable 1: Enable 0: Disable
7:0	/

Descriptor5 Definition

Bit	Description
31:0	ADDR_OPCODE Address Content Trans Through SPI

Descriptor6 Definition

This register should be setup while the controller is idle.

Bit	Description

Bit	Description
31:24	CMD_OPCODE Command Content Trans Through SPI
23:16	MODE_OPCODE Mode Content Trans Through SPI
15:8	CMD_OPCODE2 Command2 Content Trans Through SPI
7:6	CMD_TRANS_TYPE Command Transfer Type 00: Command can be Shifted to the device on DQ0 01: Command can be Shifted to the device on DQ0 and DQ1 10: Command can be Shifted to the device on DQ0- DQ3 11 : Command can be Shifted to the device on DQ0-DQ7
5:4	ADDR_TRANS_TYPE Address Transfer Type 00: Address can be Shifted to the device on DQ0 01: Address can be Shifted to the device on DQ0 and DQ1 10: Address can be Shifted to the device on DQ0- DQ3 11: Address can be Shifted to the device on DQ0-DQ7
3:2	MODE_BIT_TRANS_TYPE Mode Bit Transfer Type 00: Mode Bit can be Shifted to the device on DQ0 01: Mode Bit can be Shifted to the device on DQ0 and DQ1 10: Mode Bit can be Shifted to the device on DQ0- DQ3 11: Mode Bit can be Shifted to the device on DQ0-DQ7
1:0	DATA_TRANS_TYPE Data Transfer Type 00: Opcode can be Shifted to the device on DQ0 only 01: Opcode can be Shifted to the device on DQ0 and DQ1 only 10: Opcode can be Shifted to the device on DQ0- DQ3 11: Opcode can be Shifted to the device on DQ0-DQ7

Descriptor7 Definition

Bit	Description
31	DATA_TRANS_NUM[16]
30:29	
28	SPI_NORMAL_EN If DMA configures SPI, this bit start SPI FSM.
27	/
26	CMD_LEN 0: 1byte 1: 2byte

Bit	Description
25:24	ADDR_SIZE_MODE Address Size Mode 00: Address Size 8bit 01: Address Size 16bit 10: Address Size 24bit 11: Address Size 32bit
23:16	DUMMY_TRANS_NUM Number of Dummy Cycles A value of 0 = 1 Cycle A value of 1 = 1 Cycle ... A value of N = N Cycle
15:0	DATA_TRANS_NUM Num of Data Trans Through SPI(Byte) 0: Non-Write. 1: Write 1 Byte. 2: Write 2 Bytes. 3: Write 3 Bytes. 65535: Write 65535 Bytes. Note: These Bits indicate the number of data bytes in a CHIP SELECT period. Notice the difference between DATA_TRANS_NUM here and DMA_DATA_LEN in Descriptor1.

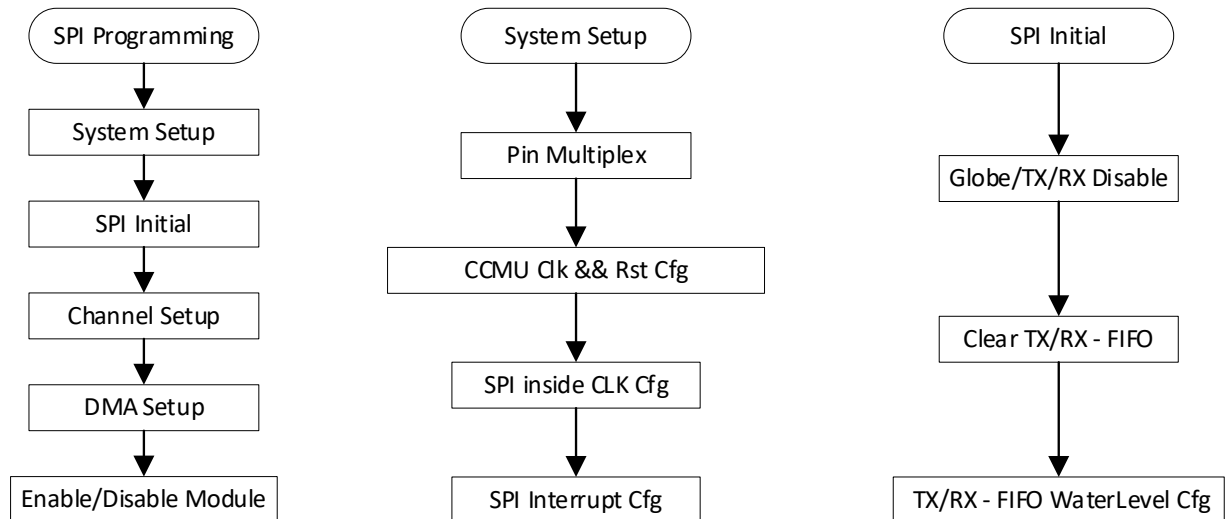
9.3.5 Programming Guidelines

9.3.5.1 DMA Transfer

The software operation of the SPI DMA transfer is divided into 5 steps as follows.

- System Setup
- SPI Initialization
- Channel Setup
- DMA Setup
- Enable SPI

Figure 9-30 SPI Programming flow



System Setup

Step 1 Configure SPI Pin by programming the 15.7 GPIO.

Step 2 Configure SPI ref clock, AHB Clock, De-assert SPI ref Reset, and AHB reset in Section 4.1 Clock Controller Unit (CCU).

Step 3 Configure SPI Internal Clock

a) Configure clock source ([SPI_TIMING_CFG](#))

➤ STR Mode

Write 0 to the CLK_SPI_SRC_SEL bit (bit [24]), the CLK_SCK_SRC_SEL bit (bit [25]) bit, and the CLK_SCKOUT_SRC_SEL bit (bit [26]) bit.

➤ DTR Mode

Write 0 to the CLK_SPI_SRC_SEL bit (bit [24]) and the CLK_SCK_SRC_SEL bit (bit [25]) bit.

Write 1 to the CLK_SCKOUT_SRC_SEL bit (bit [26]) bit.



NOTE

When the value of the CLK_SCKOUT_SRC_SEL bit (bit [26]) bit is 1, the real output clock frequency of SPIFC-CLK signal is the SPIFC clock frequency divided by 2.

b) Configure the sckr clock of SPI data receiving.

➤ Set the SCKR_DLY_MODE_SEL (bit[20]) in the [SPI_TIMING_CFG](#).

➤ Set the DIGITAL_SCKR_DELAY_CFG (bit[18:16]) in the [SPI_TIMING_CFG](#).

➤ Set the ANALOG_SAMP_DL_SW_VALUE (bit[5:0]) in the [SPI_TIMING_CFG](#).

c) Generate SPI output clock

- Select SPI working mode (SPI_MODE) in the SPI_CPOL (bit [5]) and SPI_CPHA (bit [4]) in the register [SPI_GLOBAL_CTRL](#) by referring to the section SPI Clock Mode.
- DTR_EN (bit [16]) in the ([SPI_GLOBAL_CTRL](#) [0x0004]) is closed by default. When it is closed, the frequency of output clock is equal to that of SCKR and SCKT. When it is open, the frequency of output clock is lowered to half that of SCKR and SCKT.

Step 4 Set [SPI_INT_EN](#) (0x0014[31:0]) as 0.

SPI Initialization

Step 1 Disable the [SPI_GLOBAL_CTRL](#) by writing 0 to SPI_DMA_MODE_EN (bit[2]), SPI_PMODE_EN (bit[3]), SPI_CPU_MODE_EN (bit[20]), and SPI_AUTO_READ_STATUS_MODE_EN (bit[21]).

Step 2 Reset TX/RX FIFO ([SPI_GLOBAL_CTRL_ADD](#) [0x0008])

Reset the CDC-BUF/FIFO of TX channel: Write 1 to CDC_WF_SRST to reset the WR_BUFF and WR_FIFO in SPI_WR_BUF_CTRL and the SWF in SPI_TX_INTERFACE.

Reset the CDC-BUF/FIFO of RX channel: Write 1 to CDC_RF_SRST.

Step 3 Configure the water level of FIFO in [SPI_CDC_FIFO_TRIG_LEVEL](#).

Channel Setup

Step 1 Select the SPI channel parameter resource by setting the SPI_CFG_MODE(bit[0]) of [SPI_GLOBAL_CTRL](#).



NOTE

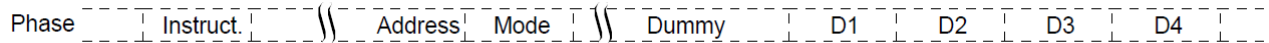
If the parameters source from CPU, finish the configuration in reference to all the guidelines in the section Channel Setup. If the parameters source from DMA, skip the steps after CS Delay Configuration in this section to initiate the DMA. For the descriptions of DMA descriptors, refer to the section SPI Flash Private DMA.

Step 2 SPI Interface Configuration—Public Configuration

- a) CS Delay Configuration: [SPI_CS_DELAY](#)[0x001C]
 - CSDA (bit [23:16]): The intervals of two adjacent CS enable. The minimum interval is recommended to be 6sclk. The actual value is CSDA+1.
 - CSEOT (bit [15:8]): After the SCLK_OUT is invalid, the CS signal will be de-asserted after CSEOT SCLK. The minimum interval is 1sclk. The actual value is CSEOT+1.
 - CSSOT (bit [7:0]): After the CS is valid, the SCLK_OUT signal will be asserted after CSSOT SCLK. The minimum interval is 1sclk. The actual value is CSSOT+1.
- b) Activate SPI Trans Phase ([SPI_TRANS_PHA_CFG](#))

In a transmission of the SPI interface, the transferring content of I/O line is as follows. Enable the corresponding Trans Phase based on the content to be transferred by writing 1 to the COMMAND_TRANS_EN (bit [28]), ADDRESS_TRANS_EN (bit [24]), MODE_BIT_TRANS_EN (bit [20]), DUMMY_BIT_TRANS_EN (bit [16]), TX_DATA_EN (bit [12]), and RX_DATA_EN (bit [8]).

Figure 9-31 SPI Transfer Phase Flow Diagram



RX_DATA_EN and TX_DATA_EN cannot be activated at the same time.

- c) Configure the number of I/O used by TransPhase by setting the CMD_TRANS_TYPE (bit [13:12]), ADDR_TRANS_TYPE (bit [9:8]), MODE_BIT_TRANS_TYPE (bit [5:4]), and DATA_BIT_TRANS_TYPE (bit [1:0]) in the [\(SPI_TRANS_CFG2\)](#).
- d) Configure the transferring volume of TransPhase in the [SPI_TRANS_NUM](#)[0x002C].

Step 3 SPI interface configuration—normal read and write

Configure the contents of all TransPhase by setting the following bits.

- ADDR: [SPI_TRANS_CFG1](#) [0x0024], ADDR_OPCODE (bit [31:0])
- CMD: [SPI_TRANS_CFG2](#) [0x0028], CMD_OPCODE (bit[31:24])
- CMD2: [SPI_TRANS_CFG2](#) [0x0028], CMD_OPCODE2 (bit[15:8])
- MODE: [SPI_TRANS_CFG2](#) [0x0028], CMD_OPCODE (bit[23:16])

Step 4 SPI Interface Configuration—XIP Mode

- a) Configure whether to use address mapping mode by setting the XIP_ADDR_MAP_MODE (bit[1]) in the [SPI_GLOBAL_CTRL](#).
- b) Configure the contents of all TransPhase by setting the following bits.
 - CMD: [SPI_TRANS_CFG2](#)[0x0028], CMD_OPCODE (bit[31:24])
 - MODE: [SPI_TRANS_CFG2](#)[0x0028], CMD_OPCODE (bit[23:16])
- c) Configure the effective range of XIP by setting the [XIP_START_ADDR](#) and the [XIP_END_ADDR](#).

Step 5 SPI Interface Configuration—BIT Mode

Refer to the functions of SPI Bit Mode (3-Wire/4-Wire).

DMA Setup

- Step 1** Configure the length and starting address of the first descriptor in the [SPI_DMA_CTRL](#) and [SPI_DESCRIPTOR_SADDR](#) to initiate DMA.

Step 2 Configure the length and the starting address of storage location for the first descriptor. Then, finish the following operations to complete DMA Configuration.

➤ Descriptor0

■ AHB Master Burst Type Configuration

HBURST_TYPE (bit [6:4]): Support SINGLE/INCR4/INCR8/INCR16 mode.

■ DMA Transfer Direction Configuration

DMA_DIR (bit [1]): DMA read or DMA write.

■ DMA Finish Flag

DMA_FINISH_FLAG (bit [0]): Currently the memory space allocated by the CPU to DMA may be a part of the total data volume to be transferred. For instance, the 256 Byte memory space is allocated to transfer 1 MB data. In this situation, a DMA descriptor merely points to the allocated 256 Byte data location, and it will also point to the next descriptor to fetch data. When the data of the last descriptor is fetched, the stop bit should be pulled up to terminate a DMA handling.

➤ Descriptor1

■ DMA_BLK_LEN ([31:16])

DMA BLK length. It indicates the size of each DMA packet, with multiples of 8 Byte aligned to achieve the optimized rate when accessing storage.

■ DMA_DATA_LEN (bit [15:0])

It indicates the data volume indicated by the current descriptor of DMA

➤ Descriptor2

It indicates the data storage location of current descriptor.

➤ Descriptor3

It indicates the storage location of the next descriptor.

➤ Descriptor4-7

The SPI configuration parameters. In the course of setting parameters with DMA for SPI, the content of descriptors is configured to SPI.

Enable SPI

Only one of DMA mode, CPU mode, AUTO_READ_STATUS_MODE mode, and XIP mode can be enabled at a time. To enable DMA mode, write 1 to the SPI_NMODE_EN in the [SPI_GLOBAL_CTRL](#) to, then it will be auto pulled down.

9.3.5.2 CPU Transfer

The software operation of CPU transfer is almost the same as DMA transfer. However, there are still two main differences.

One difference is that when the CPU transfers, channel parameter cannot come from DMA descriptors. [SPI_GLOBAL_CTRL](#)[0x0004], SPI_CFG_MODE(bit[0]) must be set to 0. All parameters should come from REGISTER FILE. [SPI_GLOBAL_CTRL](#)[0x0004], SPI_CPU_MODE_EN(bit[20]) is used to start CPU transfer (Different from descriptor and [SPI_GLOBAL_CTRL](#)[0x0004], SPI_NMODE_EN(bit[2])).

The other difference is that when reading data from Register [0x0210], it is recommended to read [SPI_CDC_FIFO_STA](#)[0x0050], RF_CNT([bit5-0]), especially when total read number is not integer multiple of trigger level. When writing data to Register [0x0220], it is always ready as long as trigger level is not reached.

NOTE

Register [0x0040] and [0x0044] are not used in CPU transfer.

9.3.5.3 Auto Status Reading

The software operation of STATUS READING is almost the same as CPU transfer. CMD_TRANS_EN and RX_DATA_EN must be set high because RX data path is used for status reading. MODE_BIT_TRANS_EN and DUMMY_BIT_TRANS_EN should be set accordingly. But STATUS OPCODE will not go through READ_FIFO and READ_BUFFER. Meanwhile, I/O Pins used should be set accordingly. [SPI_STATUS_READ](#)[0x0068] and [SPI_STATUS_READ_2](#)[0x006C] are used for STATUS READING. And to avoid endless reading and dead lock, the CPU should tell the maximum reading times. Then, to start STATUS READING, SPI_READ_STATUS_MODE_EN should be set. In this mode, DATA_TRANS_NUM (002C, [15:0]) is suggested to be 1 and DTR_EN [0x0004, bit 16] is suggested to be 0.

9.3.5.4 XIP

The software operation of XIP is almost the same as CPU transfer.

The System Setup part, SPI Initialization part and Channel Setup part is the same as DMA transfer.

After the basic initialization and setup, enable [SPI_GLOBAL_CTRL](#) [0x0004], SPI_PMODE_EN(bit 3) to enter XIP mode. Then directly read in XIP area (see Section 2 Memory Map) to get data.

9.3.6 Register List

Module Name	Base Address
SPIF	0x0C000000

Register Name	Offset	Description
SPI_GLOBAL_CTRL	0x0004	SPI Global Control Register
SPI_GLOBAL_CTRL_ADD	0x0008	SPI Global Control Additional Register

Register Name	Offset	Description
SPI_TIMING_CFG	0x000C	SPI Timing Configure Register
SPI_TIMING_DLY_STA	0x0010	SPI Timing Delay State Register
SPI_INT_EN	0x0014	SPI Interrupt Enable Register
SPI_INT_STA	0x0018	SPI Interrupt Status Register
SPI_CS_DELAY	0x001C	SPI Chipselect Delay Register
SPI_TRANS_PHA_CFG	0x0020	SPI Transfer Phase Configure Register
SPI_TRANS_CFG1	0x0024	SPI Trans Configure1 Register
SPI_TRANS_CFG2	0x0028	SPI Trans Configure2 Register
SPI_TRANS_NUM	0x002C	SPI Trans Number Register
SPI_XIP_STA	0x0030	SPI XIP State Register
SPI_XIP_START_ADDR	0x0034	SPI XIP Start Address Register
SPI_XIP_END_ADDR	0x0038	SPI XIP End Address Register
SPI_XIP_MAP_ADDR	0x003C	SPI XIP Map Address Register
SPI_DMA_CTRL	0x0040	SPI DMA Control Register
SPI_DESCRIPTOR_SADDR	0x0044	SPI DMA Descriptor Start Address Register
SPI_CDC_FIFO_TRIG_LEVEL	0x004C	SPI CDC FIFO Trigger Level Register
SPI_CDC_FIFO_STA	0x0050	SPI CDC FIFO Status Register
SPI_BATC	0x0054	SPI Bit-Aligned Transfer Configure Register
SPI_BA_CCR	0x0058	SPI Bit-Aligned CLOCK Configuration Register
SPI_TBR	0x005C	SPI TX Bit Register
SPI_RBR	0x0060	SPI RX Bit Register
SPI_STATUS	0x0064	SPI Status Register
SPI_STATUS_READ	0x0068	SPI Status Read Register
SPI_STATUS_READ2	0x006C	SPI Status Read Register_2
SPI_XIP_RX_DATA	0x0200	SPI XIP RX Data Register
SPI_RX_DATA	0x0210	SPI RX Data Register
SPI_TX_DATA	0x0220	SPI TX Data Register

9.3.7 Register Description

9.3.7.1 0x0004 SPI Global Control Register (Default Value: 0x0000_0100)

Offset: 0x0004			Register Name: SPI_GLOBAL_CTRL
Bit	Read/Write	Default/Hex	Description
31:23	/	/	/
22	R/W	0x0	START_MODE 0: Old start mode 1: New start mode

Offset: 0x0004			Register Name: SPI_GLOBAL_CTRL
Bit	Read/Write	Default/Hex	Description
21	R/WAC	0x0	SPI_READ_STATUS_MODE_EN SPI READ STATUS Mode Enable 0: Disable 1: Enable This bit is not used for data transfer. It is used for status confirm. And when this operation ends, the last returned byte value is sent to 0x0064[31:24].
20	R/WAC	0x0	SPI_CPU_MODE_EN SPI CPU Mode Enable 0: Disable 1: Enable CPU transfer mode.
19	R/W	0x0	SPI_CLK_OUTPUT 0: Disable 1: Enable When the SPI_CLK_OUTPUT is enabled, the SPI_FLASH will convey the sclk from CCU to the spi_sck_out pin. This bit cannot be enabled in the process of data transmission to avoid anomaly.
18	R/W	0x0	SPI_TX_CFG_FBS Transmit Interface First Transmit Bit Select. 0: MSB First 1: LSB First
17	R/W	0x0	SPI_RX_CFG_FBS Receive Interface First Transmit Bit Select. 0: MSB First 1: LSB First
16	R/W	0x0	DTR_EN Double Trans Rate Enable 0: Disable 1: Enable
15	R/W	0x0	SPI_WP_EN 0: Disable 1: Enable In Standard SPI or Dual SPI, SPI PIN Write Protect Function can be open In the open status, SPI_WP_IO2_OUT is enabled. The exact polarity is decided by the configuration of SPI_WP_POL. In the closed status, SPI_WP_IO2_OUT is controlled by hardware. The exact polarity is decided by the controller.

Offset: 0x0004			Register Name: SPI_GLOBAL_CTRL
Bit	Read/Write	Default/Hex	Description
14	R/W	0x0	SPI_WP_POL SPI PIN Write Protect Polar 0: Active high polarity (0 = Idle) 1: Active low polarity (1 = Idle)
13	R/W	0x0	SPI_HOLD_EN 0: Disable 1: Enable In Standard SPI or Dual SPI, SPI PIN HOLD Function can be open. In the open status, SPI_HOLD_IO3_OUT is enabled. The exact polarity is decided by the configuration of SPI_HOLD_POL. In the closed status, SPI_HOLD_IO3_OUT is controlled by hardware. The exact polarity is decided by the controller.
12	R/W	0x0	SPI_HOLD_POL SPI PIN HOLD Polar 0: Active high polarity (0 = Idle) 1: Active low polarity (1 = Idle)
11	/	/	/
10	R/W	0x0	DQS_RX_EN Receive DQS Signal Enable 0: Disable 1: Enable
9	R/W	0x0	DUMMY_BIT_POL Dummy Bit Trans Value 0: Dummy bit Trans 0 1: Dummy bit Trans 1 In the dummy period, the output data of Hold and Write Protection will be controlled by the DUMMY_BIT_POPL.
8	R/W	0x1	SPI_CS_POL SPI Chip Select Signal Polarity Control 0: Active high polarity (0 = idle) 1: Active low polarity (1 = idle)
7:6	R/W	0x0	SPI_CS_SEL SPI Chip Select Select one of four external SPI Master/Slave Devices 00: SPIF_CS0 will be asserted 01: SPIF_CS1 will be asserted (not uesed) 10: SPIF_CS2 will be asserted (not uesed) 11: SPIF_CS3 will be asserted (not uesed)

Offset: 0x0004			Register Name: SPI_GLOBAL_CTRL
Bit	Read/Write	Default/Hex	Description
5	R/W	0x0	SPI_CPOL SPI Clock Polarity Control 0: Active High Polarity. 1: Active Low Polarity.
4	R/W	0x0	SPI_CPHA SPI Clock/Data Phase Control 0: Phase 0(Leading Edge for Sample Data) 1: Phase 1(Leading Edge For Setup Data)
3	R/W	0x0	SPI_PMODE_EN SPI XIP Mode Enable 0: Configuring AHB0-bus XIP 1: Configure AHB0-bus XIP Done When CPU needs to configure AHB0-bus XIP operation, Clear 0,then send configuration parameters. When CPU Configure AHB0-bus XIP Done , Set 1. When SPI Controller detect 'Enable XIP =1', parameters will be latched.
2	R/WAC	0x0	SPI_DMA_MODE_EN SPI DMA Mode Enable 0: Disable 1: Enable DMA transfer mode.
1	R/W	0x0	XIP_ADDR_MAP_MODE XIP Address Map Mode When the CPU accesses Flash through SPI, the address map mode can be selected by adding N to the address and then mapping the fetching address to the ADDR+N.
0	R/W	0x0	SPI_CFG_MODE SPI Parameter Config Source in normal(DMA) mode. Set to 0 When CPU Transfer. 0: From register 1: From DMA descriptor

9.3.7.2 0x0008 SPI Global Control Additional Register (Default Value: 0x0000_0000)

Offset: 0x0008			Register Name: SPI_GLOBAL_CTRL_ADD
Bit	Read/Write	Default/Hex	Description
31:5	/	/	/

Offset: 0x0008			Register Name: SPI_GLOBAL_CTRL_ADD
Bit	Read/Write	Default/Hex	Description
4	R/WAC	0x0	SPI_FLASH_DMA_END Terminate the DMA Chain after the present BLK is done. After the DMA chain terminates, set 1 to this bit to stop DMA transmission after the DMA module finishes the present BLK whether the current chain completes or the finish_flag is set to 1. DMA_END and SPI_SRST should be used together: Enable the DMA_END first, and then enable the SPI_FLASH_SRST after DMA_END is cleared.
3	R/WAC	0x0	SPI_FLASH_SRST Reset All the Logic in SPI Flash (DMA chain is not included) SPI is globally reset, and all the clock domain logic is reset
2	/	/	/
1	R/WAC	0x0	CDC_WF_SRST Transfer FIFO Software Reset, In SPI_WR_BUF_CTRL Module 0: Auto Clear to 0. 1: Reset Transfer FIFO. Writing 1 to this bit will reset the control portion of the transfer FIFO, and auto clear to 0 when completing reset operation. Writing 0 to this bit has no effect.
0	R/WAC	0x0	CDC_RF_SRST Receiver FIFO Software Reset, In SPI_RD_BUF_CTRL Module 0: Auto Clear to 0. 1: Reset Receiver FIFO. Writing 1 to this bit will reset the control portion of the receiver FIFO, and auto clear to 0 when completing reset operation, write 0 to this bit has no effect.

9.3.7.3 0x000C SPI Timing Configure Register (Default Value: 0x0000_0000)

Offset: 0x000C			Register Name: SPI_TIMING_CFG
Bit	Read/Write	Default/Hex	Description
31:27	/	/	/

Offset: 0x000C			Register Name: SPI_TIMING_CFG
Bit	Read/Write	Default/Hex	Description
26	R/W	0x0	CLK_SCKOUT_SRC_SEL SPI Output Clk Select Select the rate of SPI output clock (SCK_OUT). 0: Original Clock 1: Two-frequency Clock
25	R/W	0x0	CLK_SCK_SRC_SEL SPI Interface Clock Select 0: Original Clock 1: Two-frequency Clock
24	R/W	0x0	CLK_SPI_SRC_SEL SPI Inter Process Clock Select 0: Original Clock 1: Two-frequency
23:22	/	/	/
21	R/W	0x0	SCKT_DELAY_MODE_SEL SCKT Delay Mode Select 0: Select the SCKT without delay 1: Select the SCKT after the analog delay chain adjustment
20	R/W	0x0	SCKR_DELAY_MODE_SEL SCKR Delay Mode Select 0: Digital delay 1: Digital delay + analog delay
19	/	/	/
18:16	R/W	0x0	DIGITAL_SCKR_DELAY_CFG Digital SCKR Delay Value Config, Step 0.5 spi_clk 000: Delay 0 clk 001: Delay 0.5 clk 010: Delay 1 clk 011: Delay 1.5 clk 100: Delay 2 clk 101: Delay 2.5 clk 110: Delay 3 clk 111: Delay 3.5 clk
15	R/W	0x0	ANALOG_SAMP_DL_CAL_START_TX Sample Delay Calibration Start When set, start sample delay chain calibration.
14	R/W	0x0	ANALOG_SAMP_DL_SW_EN_TX Sample Delay Software Enable When set, enable sample delay specified at SAMP_DL_SW

Offset: 0x000C			Register Name: SPI_TIMING_CFG
Bit	Read/Write	Default/Hex	Description
13:8	R/W	0x0	ANALOG_SAMP_DL_SW_VALUE_TX Sample Delay Software Value The relative delay between clock line and command line, data lines. It can be determined according to the value of SAMP_DL, the cycle of card clock and device' s input timing requirement.
7	R/W	0x0	ANALOG_SAMP_DL_CAL_START_RX Sample Delay Calibration Start When set, start sample delay chain calibration.
6	R/W	0x0	ANALOG_SAMP_DL_SW_EN_RX Sample Delay Software Enable When set, enable sample delay specified at SAMP_DL_SW
5:0	R/W	0x0	ANALOG_SAMP_DL_SW_VALUE_RX Sample Delay Software Value The relative delay between clock line and command line, data lines. It can be determined according to the value of SAMP_DL, the cycle of card clock and device' s input timing requirement.

9.3.7.4 0x0010 SPI Timing Delay State Register (Default Value: 0x0000_2020)

Offset: 0x0010			Register Name: SPI_TIMING_DLY_STA
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15	R	0x0	ANALOG_SAMP_DL_CAL_DONE_TX Sample Delay Calibration Done When set, it means that sample delay chain calibration is done and the result of calibration is shown in SAMP_DL.
14	/	/	/

Offset: 0x0010			Register Name: SPI_TIMING_DLY_STA
Bit	Read/Write	Default/Hex	Description
13:8	R	0x20	ANALOG_SAMP_DL_TX Sample Delay Value It indicates the number of delay cells corresponding to current card clock. The delay time generated by these delay cells is equal to the cycle of card clock nearly. Generally, it is necessary to do drive delay calibration when card clock is changed. This bit is valid only when SAMP_DL_CAL_DONE is set.
7	R	0x0	ANALOG_SAMP_DL_CAL_DONE_RX Sample Delay Calibration Done When set, it means that sample delay chain calibration is done and the result of calibration is shown in SAMP_DL.
6	/	/	/
5:0	R	0x20	ANALOG_SAMP_DL_RX Sample Delay Value It indicates the number of delay cells corresponding to current card clock. The delay time generated by these delay cells is equal to the cycle of card clock nearly. Generally, it is necessary to do drive delay calibration when card clock is changed. This bit is valid only when SAMP_DL_CAL_DONE is set.

9.3.7.5 0x0014 SPI Interrupt Enable Register (Default Value: 0x0000_0000)

Offset: 0x0014			Register Name: SPI_INT_EN
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28	R/W	0x0	READ_STATUS_DONE_INT_EN 0: Disable. 1: Enable.
27	R/W	0x0	READ_STATUS_TIME_OUT_INT_EN 0: Disable. 1: Enable.

Offset: 0x0014			Register Name: SPI_INT_EN
Bit	Read/Write	Default/Hex	Description
26	R/W	0x0	SPI_TRANS_DONE_INT_EN SPI Transmit Finish Interrupt Enable 0: Disable. 1: Enable.
25	R/W	0x0	XIP_READ_WORD_ACK_INT_EN XIP Mode Read back request word Interrupt Enable 0: Disable. 1: Enable.
24	R/W	0x0	DMA_TRANS_DONE_INT_EN DMA Chain Transmit Finish Interrupt Enable 0: Disable. 1: Enable.
23	R/W	0x0	DQS_RF_ODD_UDF_INT_EN DQS Receive FIFO Underflow Interrupt Enable 0: Disable. 1: Enable.
22	R/W	0x0	DQS_RF_ODD_OVF_INT_EN DQS Receive FIFO Overflow Interrupt Enable 0: Disable. 1: Enable.
21	R/W	0x0	DQS_RF_EVEN_UDF_INT_EN DQS Receive FIFO Underflow Interrupt Enable 0: Disable. 1: Enable.
20	R/W	0x0	DQS_RF_EVEN_OVF_INT_EN DQS Receive FIFO Overflow Interrupt Enable 0: Disable. 1: Enable.
19:12	/	/	/
11	R/W	0x0	WF_UDF_INT_EN Write FIFO Underflow Interrupt Enable 0: Disable. 1: Enable.
10	R/W	0x0	WF_OVF_INT_EN Write FIFO Overflow Interrupt Enable 0: Disable. 1: Enable.
9	R/W	0x0	RF_UDF_INT_EN Read FIFO Underflow Interrupt Enable 0: Disable. 1: Enable.

Offset: 0x0014			Register Name: SPI_INT_EN
Bit	Read/Write	Default/Hex	Description
8	R/W	0x0	RF_OVF_INT_EN Read FIFO Overflow Interrupt Enable 0: Disable. 1: Enable.
7:5	/	/	/
4	R/W	0x0	WF_RDY_INT_EN Write FIFO Ready Request Interrupt Enable 0: Disable. 1: Enable.
3:1	/	/	/
0	R/W	0x0	RF_RDY_INT_EN Read FIFO Ready Request Interrupt Enable 0: Disable. 1: Enable.

9.3.7.6 0x0018 SPI Interrupt Status Register (Default Value: 0x0000_0010)

Offset: 0x0018			Register Name: SPI_INT_STA
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28	R/W1C	0x0	READ_STATUS_DONE_INT READ_STATUS MODE Done Interrupt. When set, this bit indicates that at least one wanted bit has got expected value.
27	R/W1C	0x0	READ_STATUS_TIME_OUT_INT READ_STATUS MODE TIME OUT Interrupt. When set, this bit indicates that READ_STATUS MODE ends and in the written period, no wanted bit gets expected value.
26	R/W1C	0x0	SPI_TRANS_DONE_INT SPI Transmit Finish Interrupt
25	R/W1C	0x0	XIP_READ_WORD_ACK_INT XIP Mode Read Back Request Word Interrupt When set, this bit indicates that Write FIFO has underflow. Writing "1" to this bit clears it.
24	R/W1C	0x0	DMA_TRANS_DONE_INT DMA Chain Transmit Finish Interrupt When set, this bit indicates that Write FIFO has underflow. Writing "1" to this bit clears it.

Offset: 0x0018			Register Name: SPI_INT_STA
Bit	Read/Write	Default/Hex	Description
23	R/W1C	0x0	DQS_RF_ODD_UDF_INT DQS Receive FIFO underflow Interrupt Flag When set, this bit indicates that Write FIFO has underflow. Writing "1" to this bit clears it.
22	R/W1C	0x0	DQS_RF_ODD_OVF_INT DQS Receive FIFO Overflow Interrupt Flag When set, this bit indicates that Write FIFO has overflowed. Writing "1" to this bit clears it.
21	R/W1C	0x0	DQS_RF_EVEN_UDF_INT DQS Receive FIFO Underflow Interrupt Flag Note: When set, this bit indicates that Read FIFO has underflow. Writing "1" to this bit clears it.
20	R/W1C	0x0	DQS_RF_EVEN_OVF_INT DQS Receive FIFO Overflow Interrupt Flag When set, this bit indicates that Read FIFO has overflowed. Writing "1" to this bit clears it.
19:12	/	/	/
11	R/W1C	0x0	WF_UDF_INT Write FIFO underflow Interrupt Flag When set, this bit indicates that Write FIFO has underflow. Writing "1" to this bit clears it.
10	R/W1C	0x0	WF_OVF_INT Write FIFO Overflow Interrupt Flag When set, this bit indicates that Write FIFO has overflowed. Writing "1" to this bit clears it.
9	R/W1C	0x0	RF_UDF_INT Read FIFO Underflow Interrupt Flag When set, this bit indicates that Read FIFO has underflow. Writing "1" to this bit clears it.
8	R/W1C	0x0	RF_OVF_INT Read FIFO Overflow Interrupt Flag When set, this bit indicates that Read FIFO has overflowed. Writing "1" to this bit clears it.
7:5	/	/	/
4	R/W1C	0x1	WF_RDY_INT Write FIFO Ready Request Interrupt Flag 0: WF_WL > WF_EMPTY_TRIG_LEVEL. 1: WF_WL <= WF_EMPTY_TRIG_LEVEL. This bit is set any time if WF_WL <= WF_EMPTY_TRIG_LEVEL. Writing "1" to this bit clears it.
3:1	/	/	/

Offset: 0x0018			Register Name: SPI_INT_STA
Bit	Read/Write	Default/Hex	Description
0	R/W1C	0x0	RF_RDY_INT Read FIFO Ready Request Interrupt Flag 0: RF_WL < RX_EMPTY_TRIG_LEVEL and rest total RX data byte number is larger than RX_EMPTY_TRIG_LEVEL. 1: RF_WL >= RX_EMPTY_TRIG_LEVEL, or rest total RX data byte number is smaller than RX_EMPTY_TRIG_LEVEL meanwhile RF_WL >= 1. This bit is set any time if condition is reached. Writing "1" to this bit clears it.

9.3.7.7 0x001C SPI Chipselect Delay Register (Default Value: 0x0000_0606)

Offset: 0x001C			Register Name: SPI_CS_DELAY
Bit	Read/Write	Default/Hex	Description
31:24	R/W	0x0	DBG_SEL 0: The high 8 bits of debug signal are io*_oen, and the low 8 bits are 0. 1: The high 8 bits are WF_CNT, and the low 8 bits are RF_CNT. 2: The high 8 bits are WB_CNT, and the low 8 bits are RB_CNT. 3: The high 8 bits are DMA state machine, and the low 8 bits are SPI state machine. 4: Debug signal are read DRQ and write DRQ. 2'b0: dqs_rfifo_even water level; 4'b0: dqs_rfifo_odd water level.
23:16	R/W	0x0	CSDA Chip Select De-Assert The interval of two adjacent CS signal enable, namely, the interval of two SPI scheduling. The unit is the i_clk_spi_ref. The minimum interval A (CSDA=0) is 4 clk (used for ending SPI functions) The maximum A (CSDA=FB) is FF clk (used for ending SPI functions) When CSDA is X, the actual interval is X+A.

Offset: 0x001C			Register Name: SPI_CS_DELAY
Bit	Read/Write	Default/Hex	Description
15:8	R/W	0x6	CSEOT Chip Select End of Transfer The interval between the rising edge of last output clock signal (SPI_CLK_OUT) and the CS signal disable. The unit is the reference clock (i_clk_spi_ref). The minimum interval B (CSEOT=0) is: DTR-TX: 3 clk DTR-RX: 5 clk STR-TX, CPOL=0: 3 clk STR-RX, CPOL=0: 4 clk STR-TX, CPOL=1: 2.5 clk STR-RX, CPOL=1: 3.5 clk When CSEOT is X, the actual interval is X+B. When using delay, the length of CSEOT should be larger than that of delay, or the data will be lost.
7:0	R/W	0x6	CSSOT Chip Select Start of Transfer The interval between the CS signal enable and the rising edge of the first output clock signal (SPI_CLK_OUT). The unit is the reference clock (i_clk_spi_ref). The minimum interval C (CSSOT=0) is: CPOL=0: 1 clk CPOL=1: 1.5 clk When CSSOT is X, the actual interval is X+C. The minimum interval is recommended to be 6, or the anomaly will occur in some special conditions. The minimum interval should be larger than 6 in NEW_START_MODE.

9.3.7.8 0x0020 SPI Transfer Phase Configure Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: SPI_TRANS_PHA_CFG
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29	R/W	0x0	CMD_DTR CMD DTR Control 1: CMD DTR 0: CMD not DTR CAUTION: When CMD DTR, DTR must be on and CMD_LEN must be 2 bytes.

Offset: 0x0020			Register Name: SPI_TRANS_PHA_CFG
Bit	Read/Write	Default/Hex	Description
28	R/W	0x0	COMMAND_TRANS_EN Set to 1 if command data need trans to device SPI Controller FSM Phase Enable 1: Enable 0: Disable
27:25	/	/	/
24	R/W	0x0	ADDRESS_TRANS_EN Set to 1 if address need trans to device SPI Controller FSM Phase Enable 1: Enable 0: Disable
23:21	/	/	/
20	R/W	0x0	MODE_BIT_TRANS_EN Set to 1 if Mode bit need trans after address SPI Controller FSM Phase Enable 1: Enable 0: Disable
19:17	/	/	/
16	R/W	0x0	DUMMY_BIT_TRANS_EN Dummy Bit State Enable SPI Controller FSM Phase Enable 1: Enable 0: Disable
15:13	/	/	/
12	R/W	0x0	TX_DATA_EN Set to 1 if Data Need Trans SPI Controller FSM Phase Enable 1: Enable 0: Disable
11:9	/	/	/
8	R/W	0x0	RX_DATA_EN Set to '1' if Data Need Receive SPI Controller FSM Phase Enable 1: Enable 0: Disable
7:0	/	/	/

9.3.7.9 0x0024 SPI Trans Configure1 Register (Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: SPI_TRANS_CFG1
Bit	Read/Write	Default/Hex	Description

Offset: 0x0024			Register Name: SPI_TRANS_CFG1
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	ADDR_OPCODE Address Content Trans Through SPI When the length of ADDR is configured as 24bit. a) If the MSB first is configured, the low 24 bits of ADDR_OPCODE is used, namely, ADDR_OPCODE [23:0]. The sending sequence is from bit23 to bit0. b) If the LSB first is configured, the high 24 bits of ADDR_OPCODE is used, namely, ADDR_OPCODE [31:8]. The sending sequence is from bit8 to bit31. When the length of ADDR is configured as 16bit. c) If the MSB first is configured, the low 16 bits of ADDR_OPCODE is used, namely, ADDR_OPCODE [15:0]. The sending sequence is from bit15 to bit0. d) If the LSB first is configured, the high 16 bits of ADDR_OPCODE is used, namely, ADDR_OPCODE [31:16]. The sending sequence is from bit16 to bit31. When the length of ADDR is configured as 8bit. e) If the MSB first is configured, the low 8 bits of ADDR_OPCODE is used, namely, ADDR_OPCODE [7:0]. The sending sequence is from bit7 to bit0. f) If the LSB first is configured, the high 8 bits of ADDR_OPCODE is used, namely, ADDR_OPCODE [31:24]. The sending sequence is from bit24 to bit31.

9.3.7.10 0x0028 SPI Trans Configure2 Register (Default Value: 0x0000_0000)



NOTE

This register should be setup while the controller is idle.

Offset: 0x0028			Register Name: SPI_TRANS_CFG2
Bit	Read/Write	Default/Hex	Description
31:24	R/W	0x0	CMD_OPCODE Command Content Trans Through SPI

Offset: 0x0028			Register Name: SPI_TRANS_CFG2
Bit	Read/Write	Default/Hex	Description
23:16	R/W	0x0	MODE_OPCODE Mode Content Trans Through SPI
15:8	R/W	0x0	CMD_OPCODE2 Command2 Content Trans Through SPI
7:6	R/W	0x0	CMD_TRANS_TYPE Command Transfer Type 00: Command can be Shifted to the device on DQ0 01: Command can be Shifted to the device on DQ0 and DQ1 10: Command can be Shifted to the device on DQ0-DQ3 11: Command can be Shifted to the device on DQ0-DQ7
5:4	R/W	0x0	ADDR_TRANS_TYPE Address Transfer Type 00: Address can be Shifted to the device on DQ0 01: Address can be Shifted to the device on DQ0 and DQ1 10: Address can be Shifted to the device on DQ0- DQ3 11 : Address can be Shifted to the device on DQ0-DQ7
3:2	R/W	0x0	MODE_BIT_TRANS_TYPE Mode Bit Transfer Type 00: Mode Bit can be Shifted to the device on DQ0 01: Mode Bit can be Shifted to the device on DQ0 and DQ1 10: Mode Bit can be Shifted to the device on DQ0- DQ3 11 : Mode Bit can be Shifted to the device on DQ0-DQ7
1:0	R/W	0x0	DATA_TRANS_TYPE Data Transfer Type 00: Opcode can be Shifted to the device on DQ0 only 01: Opcode can be Shifted to the device on DQ0 and DQ1 only 10: Opcode can be Shifted to the device on DQ0- DQ3 11 : Opcode can be Shifted to the device on DQ0-DQ7

9.3.7.11 0x002C SPI Trans Number Register (Default Value: 0x0000_0000)

Offset: 0x002C			Register Name: SPI_TRANS_NUM
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	DATA_TRANS_NUM_HIGH Bit [16] of DATA_TRANS_NUM

Offset: 0x002C			Register Name: SPI_TRANS_NUM
Bit	Read/Write	Default/Hex	Description
30:27	/	/	/
26	R/W	0x0	CMD_LEN 0: 1byte 1: 2byte
25:24	R/W	0x0	ADDR_SIZE_MODE Address Size Mode 00: Address Size 8bit. 01: Address Size 16bit. 10: Address Size 24bit. 11: Address Size 32bit.
23:16	R/W	0x0	DUMMY_TRANS_NUM Number of Dummy Cycles A value of 0 = 1 Cycle A value of 1 = 1 Cycle ... A value of N = N Cycle
15:0	R/W	0x0	DATA_TRANS_NUM_LOW Bit [15:0] of DATA_TRANS_NUM. Num of Data Trans Through SPI(Byte) 0: Non-Write. 1: Write 1 Byte. 2: Write 2 Bytes. 3: Write 3 Bytes. 65535: Write 65535 Bytes. Note: These Bits Indicate Current Operation is Write Data To Flash.

9.3.7.12 0x0030 SPI XIP State Register (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: SPI_XIP_STA
Bit	Read/Write	Default/Hex	Description
31:7	/	/	/
6:0	R	0x0	XIP_HIT_POSITION XIP hit position

9.3.7.13 0x0034 SPI XIP Start Address Register (Default Value: 0x0000_0000)

Offset: 0x0034			Register Name: SPI_XIP_START_ADDR
Bit	Read/Write	Default/Hex	Description

Offset: 0x0034			Register Name: SPI_XIP_START_ADDR
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	XIP_START_ADDR XIP Work Start Address

9.3.7.14 0x0038 SPI XIP End Address Register (Default Value: 0x0000_0000)

Offset: 0x0038			Register Name: SPI_XIP_END_ADDR
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	XIP_END_ADDR XIP Work End Address

9.3.7.15 0x003C SPI XIP Map Address Register (Default Value: 0x0000_0000)

Offset: 0x003C			Register Name: SPI_XIP_MAP_ADDR
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	XIP_MAP_ADDR XIP Map Address, N When the address mapping mode (0x0004, bit[1]) is activated, the actual access address is ADDR+N.

9.3.7.16 0x0040 SPI DMA Control Register (Default Value: 0x0000_0000)

Offset: 0x0040			Register Name: SPI_DMA_CTRL
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:4	R/W	0x0	DMA_DESCRIPTOR_LEN DMA Descriptor Length In the initial design, there are eight DMA descriptors and the length of each descriptor is 4 Byte. In this bit, the length of descriptor is set to 32=8*4Byte.
3:1	/	/	/
0	R/WAC	0x0	CFG_DMA_START Config DMA Start Signal When all the DMA parameters are configured well, pull up a pulse of this signal to enable DMA chain.

9.3.7.17 0x0044 SPI DMA Descriptor Start Address Register (Default Value: 0x0000_0000)

Offset: 0x0044			Register Name: SPI_DESCRIPTOR_SADDR
Bit	Read/Write	Default/Hex	Description

Offset: 0x0044			Register Name: SPI_DESCRIPTOR_SADDR
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	FIRST_DESCRIPTOR_SADDR First Descriptor Start Address The WORD address is needed. Namely, the low 2 bit of byte address being removed.

9.3.7.18 0x004C SPI CDC FIFO Trigger Level Register (Default Value: 0x1001_1a10)

Offset: 0x004C			Name: SPI_CDC_FIFO_TRIG_LEVEL
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:24	R/W	0x10	WF_DMA_TRIG_LEVEL Write FIFO DMA Request Trigger Level (WORD) In the case of DMA transfer, this bit should be set to be less than or equal to 16 to avoid the efficiency reduction.
23:22	/	/	/
21:16	R/W	0x01	WF_CPU_TRIG_LEVEL Write FIFO CPU Request Trigger Level (WORD) In the case of DMA transfer, when WF_CNT is less than WF_CPU_TRIG_LEVEL, WF_RDY interrupts will be asserted.
15:14	/	/	/
13:8	R/W	0x1a	RF_FULL_TRIG_LEVEL Read FIFO Full Request Trigger Level (WORD) It is recommended to be less than or equal to 26 to avoid FIFO anomaly. It only determines the volume of RX_FIFO and has no effect on the RX_RDY_INT.
7:6	/	/	/
5:0	R/W	0x10	RF_RDY_TRIG_LEVEL Read FIFO RDY Request Trigger Level (WORD). In the case of CPU reading data, when the rest data volume is larger than RF_RDY_TRIG_LEVEL, and RF_CN is greater than RF_RDY_TRIG_LEVEL, or the rest data volume is less than RF_RDY_TRIG_LEVEL, and RF_CNT is larger than 1, the RF_RDY_INT will be sent out.

9.3.7.19 0x0050 SPI CDC FIFO Status Register (Default Value: 0x0000_0000)

Offset: 0x0050			Name: SPI_CDC_FIFO_STA
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:26	R	0x0	Flash Controller State Debug Reg.
25	R	0x0	WB_WR Write Buffer Could or Not Write.
24:21	R	0x0	WB_CNT Write Buffer Counter(Byte) These Bits Indicate Number of Bytes in Write Buffer.
20	R	0x0	RB_RD_RDY_STATE Read Buffer Could or Not Read
19:16	R	0x0	RB_CNT Read Buffer Counter(Byte) These Bits Indicate Number of Bytes in Read Buffer.
15:14	/	/	/
13:8	R	0x0	WF_CNT Write FIFO Counter (Level) These Bits Indicate Water Level in Write FIFO. 0: 0 level in Write FIFO. 1: 1 level in Write FIFO. ... 32: 32 level in Write FIFO.
7:6	/	/	/
5:0	R	0x0	RF_CNT Read FIFO Counter (Level) These Bits Indicate Water Level in Read FIFO. 0: 0 level in Read FIFO. 1: 1 level in Read FIFO. ... 32: 32 level in Read FIFO.

9.3.7.20 0x0054 SPI Bit-Aligned Transfer Configure Register (Default Value: 0x0000_00A0)

Offset: 0x0054			Register Name: SPI_BATC
Bit	Read/Write	Default/Hex	Description

Offset: 0x0054			Register Name: SPI_BATC
Bit	Read/Write	Default/Hex	Description
31	R/WAC	0x0	TCE Transfer Control Enable In master mode, it is used to start t1o transfer the serial bit's frame, it is only valid when Work Mode Select==0x10/0x11. 0: Idle 1: Initiates transfer. Writing 1 to this bit will start to transfer serial bits' frame (the value comes from the SPI TX Bit Register or SPI RX Bit Register), and will auto clear after the bursts transfer completely. Write 0 to this bit has no effect.
30	R/W	0x0	MSMS Master Sample Standard 0: Standard Sample Mode 1: Delay Sample Mode In the Standard Sample Mode, SPI master samples the data at the standard rising edge of SCLK for each SPI mode. In the Delay Sample Mode, SPI master samples data at the edge that is half cycle delayed by the standard rising edge of SCLK defined in respective SPI mode.
29:26	/	/	
25	R/W1C	0x0	TBC Transfer Bits Completed 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. 0: Busy 1: Transfer Completed Note: It is only valid when Work Mode Select=0x10/0x11.
24	R/W	0x0	TBC_INT_EN Transfer Bits Completed Interrupt Enable 0: Disable 1: Enable Note: It is only valid when Work Mode Select=0x10/0x11.
23:22	/	/	/

Offset: 0x0054			Register Name: SPI_BATC
Bit	Read/Write	Default/Hex	Description
21:16	R/W	0x00	RX_FEM_LEN Configure the length of serial data frame(burst) of RX 000000: 0bit 000001: 1bit ... 100000: 32bits Other values: Reserved Note: It is only valid when Work Mode Select==0x10/0x11, and can't be written when TCE=1.
15:14	/	/	/
13:8	R/W	0x00	TX_FEM_LEN Configure the length of serial data frame(burst) of TX 000000: 0bit 000001: 1bit ... 100000: 32bits Other values: Reserved Note: It is only valid when Work Mode Select==0x10/0x11, and can't be written when TCE=1.
7	R/W	0x1	SS_LEVEL When control CS signal manually, set this bit to 1 or 0 to control the level of CS signal. 0: Set CS to low 1: Set CS to high Note: It is only valid when Work Mode Select==0x10/0x11, and only work in Mode0, can't be written when TCE=1.
6	R/W	0x0	SS_OWNER CS Output Owner Select Usually, controller sends CS signal automatically with data together. When this bit is set to 1, software must manually write SPI_CTL_REG.SS_LEVEL to 1 or 0 to control the level of CS signal. 0: SPI controller 1: Software Note: It is only valid when Work Mode Select==0x10/0x11, and only work in Mode0, can't be written when TCE=1.

Offset: 0x0054			Register Name: SPI_BATC
Bit	Read/Write	Default/Hex	Description
5	R/W	0x1	SPOL SPI Chip Select Signal Polarity Control 0: Active high polarity (0 = Idle) 1: Active low polarity (1 = Idle) Note: It is only valid when Work Mode Select==0x10/0x11, and only work in Mode0, can't be written when TCE=1.
4	/	/	/
3:2	R/W	0x0	SS_SEL SPI Chip Select Select one of four external SPI Master/Slave Devices 00: SPIF_CS0 will be asserted 01: SPIF_CS1 will be asserted (not used) 10: SPIF_CS2 will be asserted (not used) 11: SPIF_CS3 will be asserted (not used) Note: It is only valid when Work Mode Select=0x10/0x11, and only work in Mode0, can't be written when TCE=1.
1:0	R/W	0x0	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: Reserve 10: Data frame is bit aligned in 3-Wire SPI 11: Data frame is bit aligned in Standard SPI

9.3.7.21 0x0058 SPI Bit-Aligned CLOCK Configuration Register (Default Value: 0x0000_0000)



NOTE

This register is only valid when Work Mode Select=0x10/0x11.

Offset: 0x0058			Register Name: SPI_BA_CCR
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0x0	CDR_N Clock Divide Rate (Master Mode Only) The SPI_SCLK is determined according to the following equation: $SPI\_CLK = Source\_CLK / (2 * (CDR\_N + 1))$.

9.3.7.22 0x005C SPI TX Bit Register (Default Value: 0x0000_0000)



NOTE

This register is only valid when Work Mode Select=0x10/0x11.

Offset: 0x005C			Register Name: SPI_TBR
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	VTB The Value of the Transmit Bits This register is used to store the value of the transmitted serial data frame. Note: In the process of transmission, the LSB is transmitted first.

9.3.7.23 0x0060 SPI RX Bit Register (Default Value: 0x0000_0000)



NOTE

This register is only valid when Work Mode Select=0x10/0x11.

Offset: 0x0060			Register Name: SPI_RBR
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	VRB The Value of the Receive Bits This register is used to store the value of the received serial data frame. Note: In the process of transmission, the LSB is transmitted first.

9.3.7.24 0x0064 SPI Status Register (Default Value: 0x0000_0885)

Offset: 0x0064			Register Name: SPI_STATUS
Bit	Read/Write	Default/Hex	Description
31:24	R	0x0	READ_STATUS_RESULT
23:12	/	/	/
11:8	R	0x8	DQS_RF_ODD_WL Rest of space.
7: 4	R	0x8	DQS_RF_EVEN_WL Rest of space.

Offset: 0x0064			Register Name: SPI_STATUS
Bit	Read/Write	Default/Hex	Description
3	R	0x0	WF_FULL Write FIFO Full Interrupt Flag 0: Write FIFO is Not Full. 1: Write FIFO is Full. Note: This bit is set when the Write FIFO is full.
2	R	0x1	WF_EMPTY Write FIFO Empty Request Interrupt Flag 0: Write FIFO always contains one or more bytes. 1: Write FIFO has been empty. Note: This bit is set if the Write FIFO is empty.
1	R	0x0	RF_FULL Read FIFO Full Interrupt Flag 0: Not Full. 1: Read FIFO has been Full. Note: This bit is set when Read FIFO is full.
0	R	0x1	RF_EMPTY Read FIFO Empty Interrupt Flag 0: Not Empty. 1: Read FIFO has been Empty. Note: This bit is set if the Read FIFO is empty.

9.3.7.25 0x0068 SPI Status Read Register (Default Value: 0x0000_0000)

Offset: 0x0068			Register Name: SPI_STATUS_READ
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:16	R/W	0x0	SPI_READ_STATUS_DETECTED_BIT_POSITION These 8 bits are used to distinguish the wanted bit in a STATUS BYTE. For every bit that CPU cares, put a high level in the corresponding bit position.
15:8	/	/	/
7:0	R/W	0x0	SPI_READ_STATUS_DETECTED_BIT_VALUE These 8 bits are used to distinguish the wanted bit in a STATUS BYTE. For every bit that CPU cares, put the level that CPU wants in the corresponding bit position. And when one bit is detected the same value, READ_STATUS_MODE shall stop and return a DONE interrupt. Then CPU shall check the exact result in 0x0064[31:24].

9.3.7.26 0x006C SPI STATUS READ Register_2 (Default Value: 0x0000_0000)

Offset: 0x006C			Register Name: SPI_STATUS_READ2
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:16	R/W	0x0	MAXIMUM_STATUS_READ_TIME These 14 bits are used to tell the module that when reading FLASH status, how many times it should read before it stops reading and instead return a TIME_OUT INTERRUPT to avoid dead lock. Note: The suggested minimum value is 1.
15:14	/	/	/
13:0	R/W	0x0	READ_STATUS_INTERVAL The INTERVAL is counted with 32KHz clock. These 14 bits are used to tell that when READING STATUS, after how many rising edge of the 32K clock should the module start a READ STATUS visit to the connected FLASH. Note: The suggested minimum value is 1.

9.3.7.27 0x0200 SPI XIP RX Data Register (Default Value: 0x0000_0000)

Offset: 0x0200			Register Name: SPI_XIP_RX_DATA
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	SPI_XIP_RX_DATA SPI XIP data.

9.3.7.28 0x0210 SPI RX Data Register (Default Value: 0x0000_0000)

Offset: 0x0210			Register Name: SPI_RX_DATA
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	SPI_RX_DATA SPI RX Data. It cannot be read when DMA transfer.

9.3.7.29 0x0220 SPI TX Data Register (Default Value: 0x0000_0000)

Offset: 0x0220			Register Name: SPI_TX_DATA
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	SPI_TX_DATA SPI TX Data. It cannot be written when DMA transfer.

10 Interrupts

10.1 Platform-Level Interrupt Controller (PLIC)

10.1.1 Overview

The PLIC is only used for sampling, priority arbitration and distribution for external interrupt sources.

- The interrupt can be configured as machine mode and super user mode
- Up to 192 interrupt source sampling, supporting level interrupt and pulse interrupt
- 32 levels of interrupt priority
- Maintains independently the interrupt enable for each interrupt mode (machine/super user)
- Maintains independently the interrupt threshold for each interrupt mode (machine/super user)
- Configurable access permission for PLIC registers

10.1.2 Functional Description

The following table describes the details of interrupt sources.

Table 10-1 Interrupt Sources

Interrupt Number	Interrupt Source	Interrupt Vector	Description
0		0x0000	
1		0x0004	
2	UART0	0x0008	
3	UART1	0x000C	
4	UART2	0x0010	
5	UART3	0x0014	
6	UART4	0x0018	
7	UART5	0x001C	
8		0x0020	
9	TWI0	0x0024	
10	TWI1	0x0028	
11	TWI2	0x002C	
12		0x0030	
13		0x0034	
14		0x0038	
15	SPI0	0x003C	
16	SPI1	0x0040	

Interrupt Number	Interrupt Source	Interrupt Vector	Description
17	SPI_FLASH0	0x0044	
18	PWM	0x0048	
19		0x004C	
20		0x0050	
21		0x0054	
22		0x0058	
23	SPDIF	0x005C	
24		0x0060	
25	AUDIO_CODEC	0x0064	
26	I2S_PCM0	0x0068	
27		0x006C	
28		0x0070	
29	USB0_DEVICE	0x0074	
30	USB0_EHCI	0x0078	
31	USB0_OHCI	0x007C	
32		0x0080	
33		0x0084	
34		0x0088	
35		0x008C	
36	PWM_BL0	0x0090	
37	PWM_BL1	0x0094	
38	IR_RX	0x0098	IR_RX_IRQ[9:0] OR to 1bit
39		0x009C	
40	SD0	0x00A0	
41		0x00A4	
42	SD2	0x00A8	
43	MSI	0x00AC	
44		0x00B0	
45		0x00B4	
46		0x00B8	
47		0x00BC	
48	CCU_FERR	0x00C0	
49		0x00C4	
50	DMA_NS	0x00C8	DMA 0~11 channel non-secure irq
51		0x00CC	
52	CE_NS	0x00D0	
53		0x00D4	
54	SPINLOCK	0x00D8	
55	HSTIMER0	0x00DC	
56	HSTIMER1	0x00E0	

Interrupt Number	Interrupt Source	Interrupt Vector	Description
57	GPA	0x00E4	GPADC_IRQ
58	THS	0x00E8	THS_IRQ
59	TIMER0	0x00EC	
60	TIMER1	0x00F0	
61		0x00F4	
62	TPA	0x00F8	TPADC_IRQ
63	WATCHDOG	0x00FC	
64		0x0100	
65		0x0104	
66	VE	0x0108	
67	GPIOA_NS	0x010C	
68		0x0110	
69	GPIOB_NS	0x0114	
70		0x0118	
71	GPIOC_NS	0x011C	
72		0x0120	
73	GPIOD_NS	0x0124	
74		0x0128	
75	GPIOE_NS	0x012C	
76		0x0130	
77	GPIOF_NS	0x0134	
78		0x0138	
79		0x013C	
80		0x0140	
81		0x0144	
82		0x0148	
83		0x014C	
84		0x0150	
85		0x0154	
86		0x0158	
87	DE	0x015C	
88	DI	0x0160	
89		0x0164	
90	LCD	0x0168	
91		0x016C	
92	DSI	0x0170	
93		0x0174	
94		0x0178	
95		0x017C	
96		0x0180	
97		0x0184	
98		0x0188	

Interrupt Number	Interrupt Source	Interrupt Vector	Description
99		0x018C	
100		0x0190	
101		0x0194	
102		0x0198	
103		0x019C	
104		0x01A0	
105		0x01A4	
106		0x01A8	
107		0x01AC	
108		0x01B0	
109		0x01B4	
110		0x01B8	
111	G2D0	0x01BC	
112	G2D1	0x01C0	
113	G2D2	0x01C4	
114	G2D3	0x01C8	
115		0x01CC	
116		0x01D0	
117		0x01D4	
118		0x01D8	
119		0x01DC	
120		0x01E0	
121		0x01E4	
122		0x01E8	
123		0x01EC	
124		0x01F0	
125		0x01F4	
126		0x01F8	
127		0x01FC	
128		0x0200	
129		0x0204	
130		0x0208	
131	RISCV_WDG	0x020C	
132		0x0210	
133		0x0214	
134		0x0218	
135		0x021C	
136	NMI	0x0220	
137		0x0224	
138		0x0228	
139		0x022C	
140		0x0230	

Interrupt Number	Interrupt Source	Interrupt Vector	Description
141		0x0234	
142		0x0238	
143		0x023C	
144	ALARM0	0x0240	
145		0x0244	
146		0x0248	
147		0x024C	
148		0x0250	
149		0x0254	
150		0x0258	
151		0x025C	
152		0x0260	
153		0x0264	
154		0x0268	
155		0x026C	
156		0x0270	
157		0x0274	
158		0x0278	
159		0x027C	
160		0x0280	
161~191		0x0284~	

10.1.3 Register List

Module Name	Base Address
RISCV_PLIC	0x10000000

Register Name	Offset	Description
PLIC_PRIO_REGn	0x0000+n*0x0004 (0<n<192)	PLIC Priority Register n
PLIC_IP_REGn	0x1000+n*0x0004 (0≤n<7)	PLIC Interrupt Pending Register n
PLIC_MIE_REGn	0x2000+n*0x0004 (0≤n<7)	PLIC Machine Mode Interrupt Enable Register n (0≤n<7)
PLIC_SIE_REGn	0x2080+n*0x0004 (0≤n<7)	PLIC Superuser Mode Interrupt Enable Register n
PLIC_CTRL_REG	0x1FFFFC	PLIC Control Register
PLIC_MTH_REG	0x200000	PLIC Machine Threshold Register
PLIC_MCLAIM_REG	0x200004	PLIC Machine Claim Register
PLIC_STH_REG	0x201000	PLIC Superuser Threshold Register
PLIC_SCLAIM_REG	0x201004	PLIC Superuser Claim Register

10.1.4 Register Description

10.1.4.1 $0x0000+n*0x0004$ ($0<n<192$) PLIC Priority Register n (Default Value: $0x0000\_0000$)

Offset: $0x0000+n*0x0004$ ($0<n<192$)			Register Name: PLIC_PRIO_REGn
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	PLIC_PRIO PLIC Priority Support for 32 different levels of priority. Where, a priority sets to 0 indicates that the interrupt is invalid. Machine mode interrupts have unconditionally higher priority than super-user mode interrupts. When the interrupt target mode is the same, priority 1 is the lowest priority, priority 31 is the highest priority. When multiple interrupts of the same priority are waiting arbitration, the interrupt source ID is compared. The smaller ID has the higher priority.

10.1.4.2 $0x1000+n*0x0004$ ($0\leq n<7$) PLIC Interrupt Pending Register n (Default Value: $0x0000\_0000$)

Offset: $0x1000+n*0x0004$ ($0\leq n<7$)			Register Name: PLIC_IP_REGn
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	PLIC_IP PLIC Interrupt Pending.

10.1.4.3 $0x2000+n*0x0004$ ($0\leq n<7$) PLIC Machine Mode Interrupt Enable Register n (Default Value: $0x0000\_0000$)

Offset: $0x2000+n*0x0004$ ($0\leq n<7$)			Register Name: PLIC_MIE_REGn
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	PLIC_MIE PLIC Machine Mode Interrupt Enable.

10.1.4.4 $0x2080+n*0x0004$ PLIC Superuser Mode Interrupt Enable Register n ($0\leq n<7$) (Default Value: $0x0000\_0000$)

Offset: $0x2080+n*0x0004$ ($0\leq n<7$)			Register Name: PLIC_SIE_REGn
Bit	Read/Write	Default/Hex	Description

Offset: 0x2080+n*0x0004 (0≤n<7)			Register Name: PLIC_SIE_REGn
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	PLIC_SIE PLIC Superuser Mode Interrupt Enable.

10.1.4.5 0x1FFFC PLIC Control Register (Default Value: 0x0000_0000)

Offset: 0x1FFFC			Register Name: PLIC_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	PLIC_CTRL PLIC Control 0: Only the machine mode can access to all registers in PLIC. The super-user mode cannot access PLIC_CTRL, PLIC_PRIO, PLIC_IP, and PLIC_IE registers, only access the interrupt threshold register and the interrupt response/completion register. The normal-user mode cannot access any registers in PLIC. 1: The machine mode can access to all registers in PLIC. The super-user mode can access all registers except PLL_CTRL in PLIC. The normal-user mode cannot access any registers in PLIC.

10.1.4.6 0x200000 PLIC Machine Threshold Register (Default Value: 0x0000_0000)

Offset: 0x200000			Register Name: PLIC_MTH_REG
Bit	Read/Write	Default/Hex	Description
31:5	/	/	/
4:0	R/W	0x0	PLIC_MTH PLIC Machine Threshold Indicate the interrupt threshold of the current interrupt mode. If the threshold is configured to 0, it indicates that all interrupts are permitted.

10.1.4.7 0x200004 PLIC Machine Claim Register (Default Value: 0x0000_0000)

Offset: 0x200004			Register Name: PLIC_MCLAIM_REG
Bit	Read/Write	Default/Hex	Description
31:10	/	/	/

Offset: 0x200004			Register Name: PLIC_MCLAIM_REG
Bit	Read/Write	Default/Hex	Description
9:0	R/W	0x0	PLIC_MCLAIM PLIC Machine Claim Read register: Return the current stored ID value of the register. The read operation indicates that the interrupt of the corresponding ID starts to perform. The PLIC starts to process the interrupt response. Write register: Indicate that the interrupt of the corresponding ID is complete. The writing operation cannot update the response/completion register. The PLIC starts to process the interrupt completion.

10.1.4.8 0x201000 PLIC Superuser Threshold Register (Default Value: 0x0000_0000)

Offset: 0x201000			Register Name: PLIC_STH_REG
Bit	Read/Write	Default/Hex	Description
31:5	/	/	/
4:0	R/W	0x0	PLIC_STH. PLIC Superuser Threshold. Indicate the interrupt threshold of the current interrupt mode. If the threshold is configured to 0, it indicates that all interrupts are permitted.

10.1.4.9 0x201004 PLIC Superuser Claim Register (Default Value: 0x0000_0000)

Offset: 0x201004			Register Name: PLIC_SCLAIM_REG
Bit	Read/Write	Default/Hex	Description
31:10	/	/	/
9:0	R/W	0x0	PLIC_SCLAIM. PLIC Superuser Claim. Read register: Return the current stored ID value of the register. The read operation indicates that the interrupt of the corresponding ID starts to perform. The PLIC starts to process the interrupt response. Write register: Indicate that the interrupt of the corresponding ID is complete. The writing operation cannot update the response/completion register. The PLIC starts to process the interrupt completion.

11 DMA

11.1 DMA Controller (DMAC)

11.1.1 Overview

The direct memory access (DMA) is a method of transferring data between peripherals and memories (including the SRAM and DRAM) without using the CPU. The data can be transferred between a peripheral and a memory, between peripherals, or between memories through DMA. It is an efficient way to offload data transfer duties from the CPU.

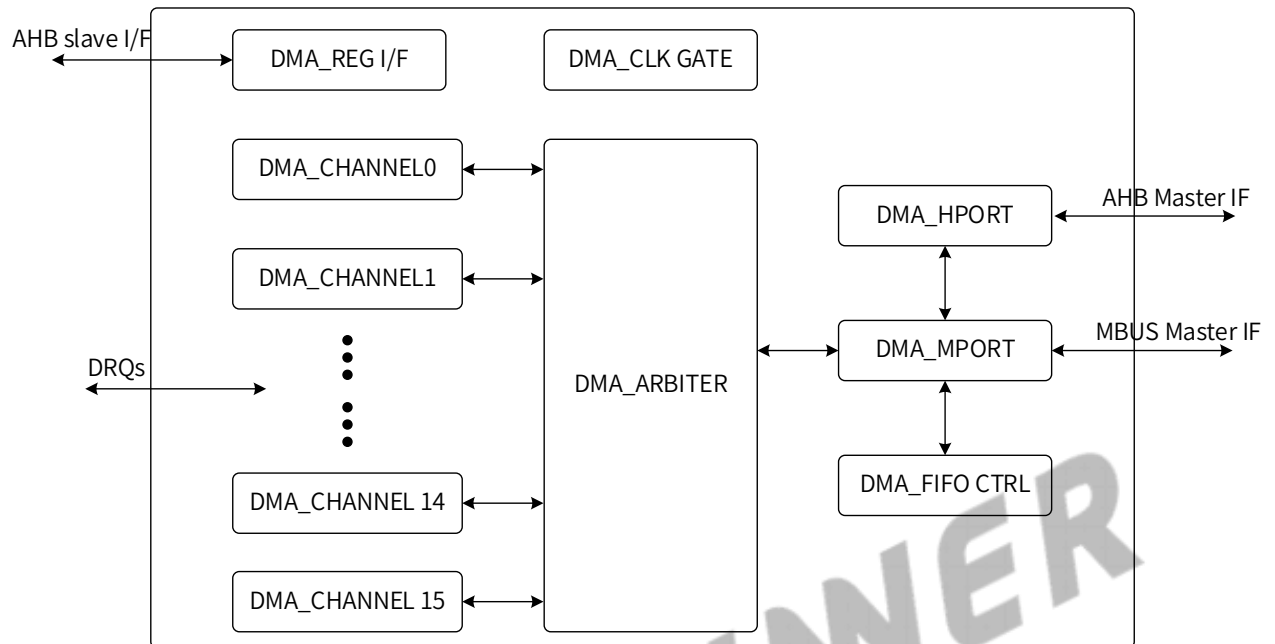
The DMAC has the following features:

- Up to 12 DMA channels
- Provides 64 peripheral DMA requests for data reading and 64 peripheral DMA requests for data writing
- Supports transferring data with a linked list
- Supports programmable 8-bit, 16-bit, 32-bit, and 64-bit data width
- Supports programmable DMA burst length
- DRQ response includes the waiting mode and handshake mode
- DMA channel supports pause function
- Memory devices support non-aligned transform

11.1.2 Block Diagram

The following figure shows a block diagram of DMAC.

Figure 11-1 DMAC Block Diagram



DMAC contains the following sub-blocks:

Table 11-1 DMAC Sub-blocks

Sub-block	Description
DMA_ARBITER	The port arbitrates the DMA read/write requests from all channels, and converts the requests to the read/write requests of ports.
DMA_CHANNELS	DMA transfer engine. Each channel is independent. The system uses the polling mechanism to decide the priorities of DMA channels. When the DMA requests from multiple peripherals are valid simultaneously, the channel with the higher priority starts data transfer first. When DMA_ARBITER is idle, channel 0 has the highest priority, whereas channel 15 has the lowest priority. When DMA_ARBITER is busy processing the request from channel n, channel (n+1) has the highest priority. For n = 15, the channel (n + 1) should be channel 0.
DRQs	DMA requests. Peripherals use the DMA request signals to request a data transfer.
DMA_MPORT	The port receives the read/write requests from DMA_ARBITER, and converts them to the corresponding MBUS access requests. It is mainly used for accessing the DRAM.
DMA_HPORT	The port for accessing the AHB Master. It is mainly used for accessing the SRAM and IO devices.

Sub-block	Description
DMA_FIFO_CTRL	Internal FIFO cell control module.
DMA_REG Interface	DMA_REG is the common register module that is mainly used to resolve AHB1 demands.
DMA_CLKGATE	The control module for hardware auto clock gating.

The DMAC integrates 16 independent DMA channels and each channel has an independent FIFO controller. When the DMA channel starts, the DMAC gets a DMA descriptor from the DMA_DESC_ADDR_REG and uses it as the configuration information for the data transfer of the current DMA package. Then the DMAC can transfer data between the specified devices. After transferring a DMA package, the DMAC judges whether the current channel transfer is finished via the linked address in the descriptor. If the linked address shows all the packages are transferred, the DMAC will end the chain transmission and close the channel.

11.1.3 Functional Description

11.1.3.1 DRQ Port Types

The following table shows the source DRQ types and destination DRQ types of different ports.

Table 11-2 DMA DRQ Types

Source DRQ Type		Destination DRQ Type	
port0	SRAM	port0	SRAM
port1	DRAM	port1	DRAM
port2	OWA	port2	OWA
port3	I2S0_RX	port3	I2S0_TX
port4		port4	
port5		port5	
port6		port6	
port7	AUDIO_CODEC	port7	AUDIO_CODEC
port8		port8	
port9		port9	
port10		port10	
port11		port11	
port12	GPADC	Port12	
port13	TPADC	port13	
port14	UART0-RX	port14	UART0-TX
port15	UART1-RX	port15	UART1-TX
port16	UART2-RX	port16	UART2-TX
port17	UART3-RX	port17	UART3-TX
port18	UART4-RX	port18	UART4-TX
port19	UART5-RX	port19	UART5-TX
port20		Port20	
port21		port21	

Source DRQ Type		Destination DRQ Type	
port22	SPI0-RX	port22	SPI0-TX
port23	SPI1-RX	port23	SPI1-TX
port24		port24	
port25		port25	
port26		port26	
port27		port27	
port28		port28	
port29		port29	
Port30	OTG0_EP1	Port30	OTG0_EP1
Port31	OTG0_EP2	Port31	OTG0_EP2
Port32	OTG0_EP3	Port32	OTG0_EP3
Port33	OTG0_EP4	Port33	OTG0_EP4
Port34	OTG0_EP5	Port34	OTG0_EP5
Port35		Port35	
Port36		Port36	
Port37		Port37	
Port38		Port38	
Port39		Port39	
Port40	CE	Port40	CE
Port41		Port41	
Port42		Port42	
Port43	TWI0	Port43	TWI0
Port44	TWI1	Port44	TWI1
Port45	TWI2	Port45	TWI2
Port46		Port46	
Port47		Port47	
Port48-63		Port48-63	

11.1.3.2 DMAC Descriptor

The DMAC descriptor is the configuration information of DMA transfer and decides the DMA working mode. Each descriptor includes 6 words: Configuration, Source Address, Destination Address, Byte Counter, Parameter, and Link.

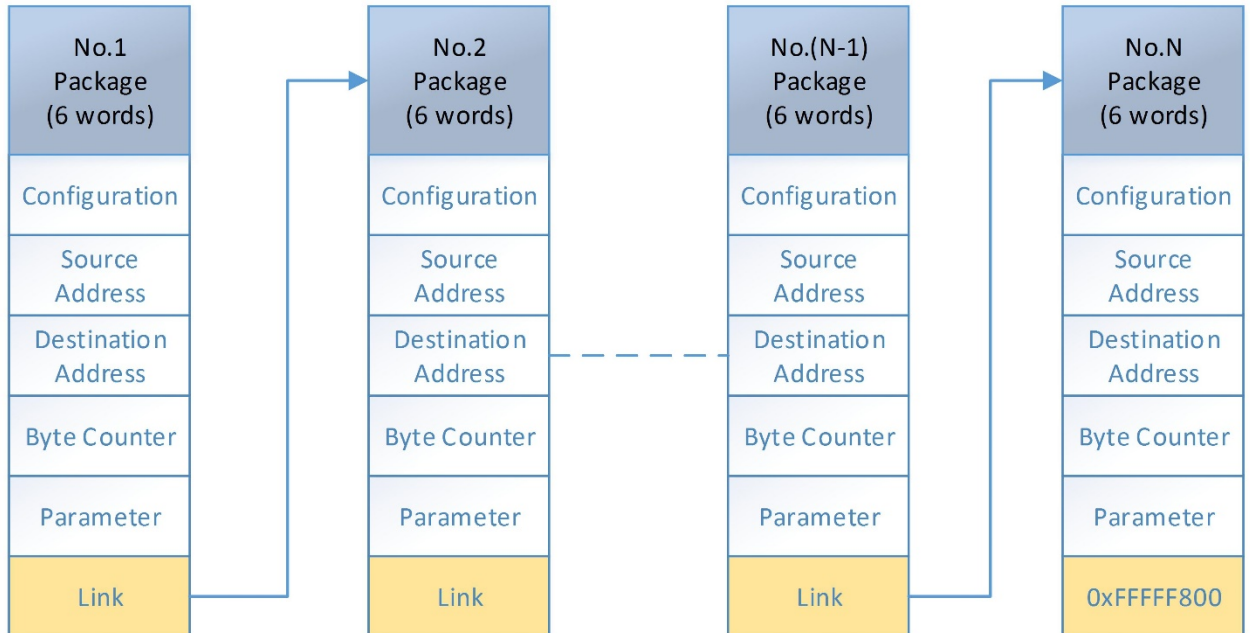
Figure 11-2 DMA Descriptor

Configuration
Source Address
Destination Address
Byte Counter
Parameter
Link

- **Configuration:** It is mainly used for configuring the following information by [DMAC_CFG_REG](#).
 - **DRQ type:** DRQ type of the source and destination devices.
 - **Address counting mode:** For both the source and destination devices, there are two address counting modes: the IO mode and linear mode. The IO mode is for IO devices whose address is fixed during the data transfer and the linear mode is for the memory whose address is increasing during the data transfer.
 - **Transferred block length:** The amount of data that non-memory peripherals can transfer in a valid DRQ. The block length supports 1 beat, 4 beats, 8 beats, and 16 beats.
 - **Transferred data width:** The data width of operating the non-memory peripherals. The data width supports 8 bits, 16 bits, 32 bits, and 64 bits.
 - **BMODE mode:** It is used in the following scenario: the source is an IO device, and the destination is a memory device. In this mode, the data amount of DMA block transmission can be limited to that when the DRQ threshold of source IO device is 1.
- **Source Address:** It configures the address of the source device.
- **Destination Address:** It configures the address of the destination device. DMA reads data from the source address and then writes data to the destination address.
- **Byte counter:** It configures the data amount of a package. The maximum value is $(2^{25}-1)$ bytes. If the data amount of the package reaches the maximum value, even if DRQ is valid, the DMA will stop the current transfer.
- **Parameter:** It configures the interval of each data block. When detecting that the DRQ holds high, the DMA will start block transmission regardless of the DRQ change. When transmission ends, DMA will wait the set period and continue to the next DRQ detection.

- **Link:** If the value of the link is 0xFFFFF800, the current package is at the end of the linked list. The DMAC will stop the data transfer after transferring the package; otherwise, the value of the link is considered as the descriptor address of the next package.

Figure 11-3 DMA Linked List Transfer



DMA reads data from the source address and then writes data to the destination address.

Both the DMA source and destination addresses have 34 bits. In the descriptor, because there are only 32 bits in the Source/Destination Address field, another 2 bits are stored in the Parameter field.

The following table shows the details of the related fields in the descriptor.

Table 11-3 DMA Address Descriptor

Descriptor Group	Bit	Description
Source Address	31:0	DMA transfers the lower 32 bits of the 34-bit source address
Destination Address	31:0	DMA transfers the lower 32 bits of the 34-bit destination address
Parameter	31:20	Reserved
	19:18	DMA transfers the higher 2 bits of the 34-bit destination address
	17:16	DMA transfers the high 2 bits of the 34-bit source address
	15:8	Reserved
	7:0	Wait Clock Cycles Set the waiting time in DRQ mode
Link	31:2	The address of the next group descriptor, the lower 30 bits of the word address
	1:0	The address of the next group descriptor, the higher 2 bits of the word address

From the above table, you can get:

Real DMA source address (in byte mode) = {Parameter [17:16], Source Address [31:0]};

Real DMA destination address (in byte mode) = {Parameter [19:18], Destination Address [31:0]};

Real link address (in byte mode) = {Link [1:0], Link [31:2], 2'b00}.

11.1.3.3 Clock Gating

The DMA_CLK_GATE module can auto control clock gating by hardware, which is mainly used for generating clocks for DMAC sub-modules and some of their circuits.

The DMA_CLK_GATE module consists of two parts: the channel clock gate and the common clock gate.

Channel Clock Gate: When the system accesses the registers of the enabled DMA channels, the channel clock gate will be automatically open. When the system finishes accessing or the DMA data transfer is completed, the channel clock gate will be automatically closed after a 16-HCLK-cycle delay; also, the related circuit clocks, such as channel control and FIFO control, will be closed.

Common Clock Gate: When all the DMA channels are closed, the clock gates of common circuits will be automatically closed. The common circuits include the common circuits of FIFO control modules, MPORT modules, and MBUS.

Whether DMA clock gating can support all the functions stated above or not is decided by software.

11.1.3.4 Interrupts

There are three kinds of DMA interrupts: the half package interrupt, package end interrupt, and queue end interrupt.

- Half package interrupt

When enabled, the DMAC sends out a half package interrupt after transferring half of a package.

- Package end interrupt

When enabled, the DMAC sends out a package end interrupt after transferring a complete package.

- Queue end interrupt

When enabled, the DMAC sends out a queue end interrupt after transferring a complete queue.

Notice that when CPU does not respond to the interrupts timely, or two DMA interrupts are generated very closely, the later interrupt may override the former one. As for CPU, the DMAC has only a system interrupt source.

11.1.3.5 Transfer Mode

The peripherals initiate data transfer by transmitting DMA request signals to the DMAC. After receiving the request signal, the DMAC converts it to the internal DRQ signal and controls the DMA data transfer.

The DMAC supports two data transfer modes: the waiting mode and handshake mode.

Waiting Mode

- a) When the DMAC detects a valid external request signal, the DMAC starts to operate the peripheral device. The internal DRQ always holds high before the transferred data amount reaches the transferred block length.
- b) When the transferred data amount reaches the transferred block length, the internal DRQ will pull low automatically.
- c) The internal DRQ holds low for certain clock cycles (WAIT_CYC), and then the DMAC restarts to detect the external requests. If the external request signal is valid, then the next transfer starts.

Handshake Mode

- a) When the DMAC detects a valid external request signal, the DMAC starts to operate the peripheral device. The internal DRQ always holds high before the transferred data amount reaches the transferred block length.
- b) When the transferred data amount reaches the transferred block length, the internal DRQ will be pulled down automatically. For the last data transfer of the block, the DMAC sends a DMA Last signal with the DMA commands to the peripheral device. The DMA Last signal will be packed as part of the DMA commands and transmitted on the bus. It is used to inform the peripheral device that it is the end of the data transfer for the current DRQ.
- c) When the peripheral device receives the DMA Last signal, it can judge that the data transfer for the current DRQ is finished. To continue the data transfer, it sends a DMA Active signal to the DMAC.



NOTE

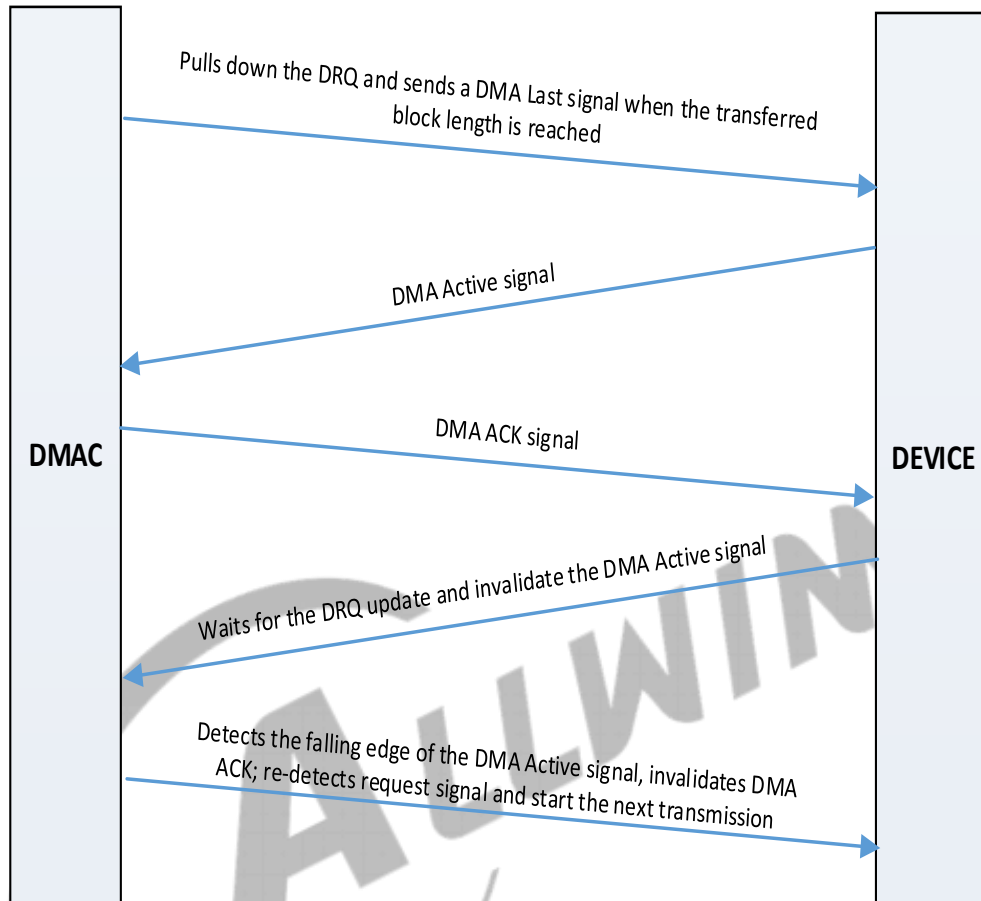
A DMA Active signal will be converted to one DRQ signal in the DMA module. To generate multiple DRQs, the peripheral devices need to send out multiple DMA Active signals via the bus protocol.

- d) When the DMAC receives the DMA Active signal, it will send back a DMA ACK signal to the peripheral device.
- e) When the peripheral device receives the DMA ACK signal, it will wait for all the operations on the local device to be completed, and both the FIFO and DRQ status are refreshed. Then it invalidates the DMA Active signal.

- f) When the DMAC detects the falling edge of the DMA Active signal, it invalidates the corresponding DMA ACK signal, and restarts to detect the external request signals. If a valid request signal is detected, the next data transfer starts.

The following figure shows the workflow of the handshake mode.

Figure 11-4 Workflow of the DMAC Handshake Mode



11.1.3.6 Address Auto-Alignment

For the non-IO devices whose start address is not 32-byte-aligned, the DMAC will align the address to the 32-byte through the burst transfer within 32 bytes, which will improve the DRAM access efficiency.

The following example shows how the DMAC adjusts the address: when the peripheral device of a DMA channel is a non-IO device whose start address is 0x86 (not 32-byte-aligned), the DMAC firstly uses a 26-byte burst transfer to align the address to 0xA0 (32-byte-aligned), and then transfers data by 64-byte burst (the maximum transfer amount that MBUS allows).

The IO devices do not support address alignment, so the bit width of IO devices must match the address offset; otherwise, the DMAC will ignore the inconsistency and directly transmit data of the corresponding bit width to the address.

The address of the DMA descriptor does not support the address auto-alignment. Make sure the address is word-aligned; otherwise the DMAC cannot identify the descriptor.

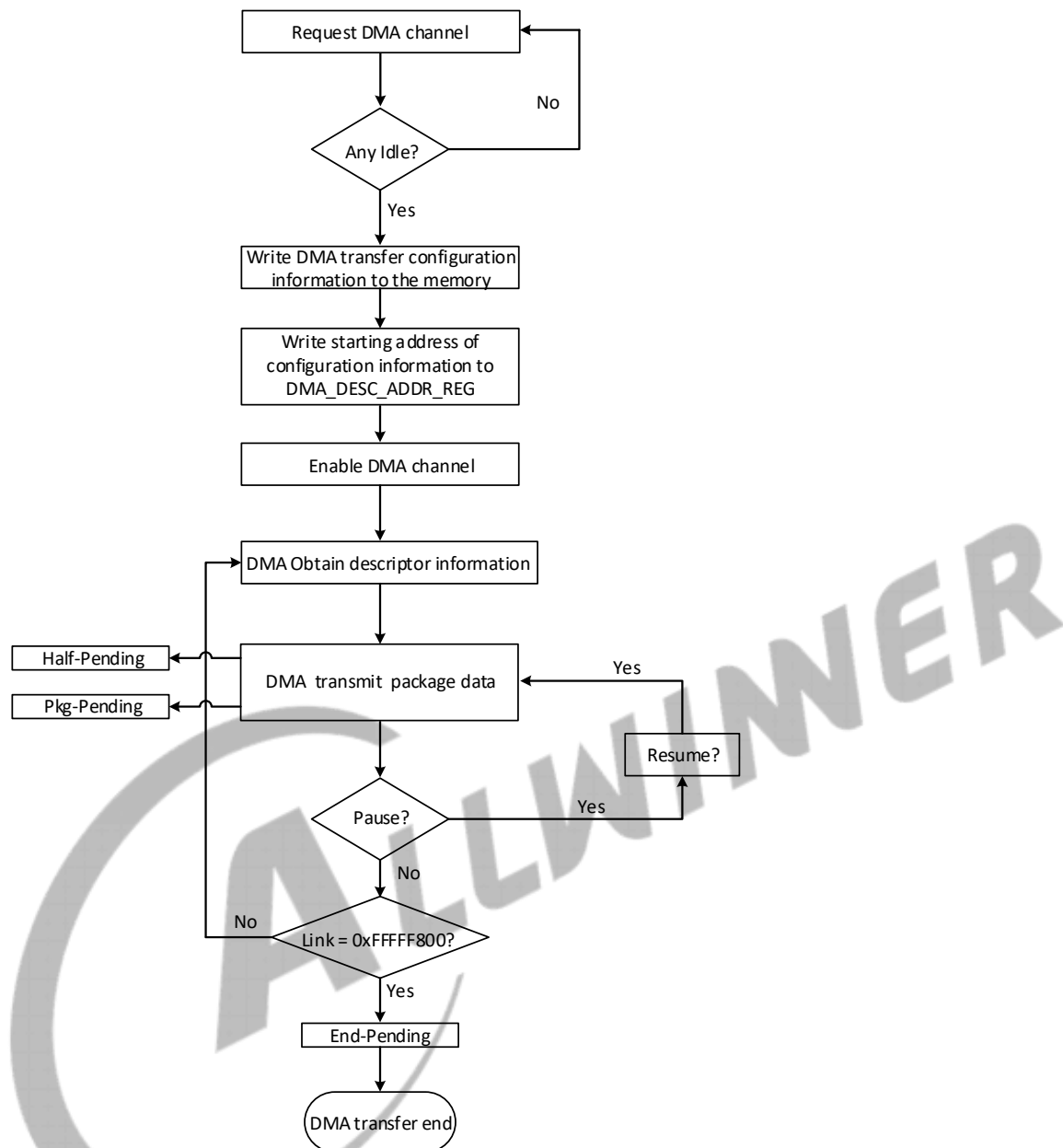
11.1.4 Programming Guidelines

11.1.4.1 DMAC Transfer Procedure

The DMAC transfer procedure is as follows.

- Step 1** Request DMA channel, and check if the DMA channel is idle by checking if it is enabled. A disabled channel indicates it is idle, while an enabled channel indicates it is busy.
- Step 2** Write the descriptor with 6 words into the memory. The descriptor must be word-aligned. For more details, refer to section 11.1.3.2 DMAC Descriptor.
- Step 3** Write the start address of the descriptor to [DMAC_DESC_ADDR_REG](#).
- Step 4** Enable the DMA channel, and write the corresponding channel to [DMAC_EN_REG](#).
- Step 5** The DMA obtains the descriptor information.
- Step 6** Start to transmit a package. When half of the package is completed, the DMA sends a Half Package Transfer Interrupt; when a total package is completed, the DMA sends a Package End Transfer Interrupt. This interrupt status can be read by [DMAC_IRQ_PEND_REG](#).
- Step 7** Set [DMAC_PAU_REG](#) to pause or resume the data transfer.
- Step 8** After completing a total package transfer, the DMA decides to start the next package transfer or end the transfer by the link of the descriptor. If the link is 0xFFFFF800, the transfer ends; otherwise, the next package starts to transmit. When the transfer ends, the DMA sends a Queue End Transfer Interrupt.
- Step 9** Disable the DMA channel.

Figure 11-5 DMAC Transfer Process



11.1.4.2 DMAC Clock Control

If the DMAC clock is synchronous with the AHB0 clock, enable the DMAC gating bit of AHB0 clock before accessing the DMAC register.

If the reset input signal of the DMAC is asynchronous with AHB0 and is low valid by default, de-assert the reset signal of the DMAC before accessing the DMA register.

To avoid the indefinite state within registers, de-assert the reset signal first, and then enable the gating bit of AHB0.

As the DMAC supports Clock Auto Gating function, it will automatically disable the DMAC clock in the idle state to reduce power consumption. The Clock Auto Gating is enabled by default.

11.1.4.3 Interrupt Processing

Follow the steps below to process the DMAC interrupt:

- Step 1** Enable interrupt: write the corresponding interrupt enable bit of [DMAC_IRQ_EN_REG](#). The system generates an interrupt when the corresponding condition is satisfied.
- Step 2** After entering the interrupt process, write to clear the interrupt pending and execute the process of waiting for the interrupt.
- Step 3** Resume the interrupt and continue to execute the interrupted process.

11.1.4.4 Configuring DMAC

To configure the DMAC, follow the guidelines below:

- Make sure the transfer bit width of IO devices is consistent with the offset of the start address.
- The MBUS protocol does not support the read operation of non-integer words. For the devices whose bit width is not word-aligned, after receiving the read command, they should resolve the read command according to their FIFO bit width instead of the command bit width, and ignore the redundant data caused by the inconsistency of the bit width.
- When the DMA transfer is paused, this is equivalent to invalid DRQ. Because there is a certain time delay between DMA transfer commands, the DMAC will not stop data transfer until the DMAC finishes processing the current command and the commands in Arbiter (at most 32 bytes' data).

DMAC application example:

```
writel (0x00000000, mem_address + 0x00); //Set configurations. The mem_address must be
word-aligned.
writel (0x00001000, mem_address + 0x04); // Set the start address for the source device.
writel (0x20000000, mem_address + 0x08); //Set the start address for the destination device.
writel (0x00000020, mem_address + 0x0C); // Set the data package size.
writel (0x00000000, mem_address + 0x10); //Set the parameters.
writel (0xFFFFF800, mem_address + 0x14); //Set the start address for the next descriptor.
writel (mem_address, 0x03002000 + 0x100 + 0x08); //Set the start address for the DMA channel0
descriptor.
do {
If (mem_address == readl (0x03002000 + 0x100 + 0x08));
break;
} while (1); //Make sure that the writing operation is valid.
writel (0x000000001, 0x03002000 + 0x100 + 0x00); // Enable DMA channel0 transfer.
```

The DMAC supports increasing data package in transfer, pay attention to the following points:

- The 0xFFFFF800 value of [DMAC_FDESC_ADDR_REGN](#) indicates that the DMA channel has got back the descriptor of the last package. The DMA channel will automatically stop the data transfer after transferring the current package.
- To add a package during the data transfer, check if the DMA channel has got back the descriptor of the last package. If yes, you cannot add any package in the current queue. Request another DMA channel with a new DRQ to transfer the package. Otherwise, you can add the package by modifying the [DMAC_FDESC_ADDR_REGN](#) of the last package from 0xFFFFF800 to the start address of the to-be-added package.

To ensure that the modification is valid, read the value of [DMAC_FDESC_ADDR_REGN](#) after the modification. The value 0xFFFFF800 indicates the modification fails and the other values indicate you have successfully added packages to the queue.

Another problem is, the system needs some time to process the modification, during which the DMA channel may get back the descriptor of the last package. You can read [DMAC_CUR_SRC_REGN](#) and [DMAC_CUR_DEST_REGN](#) and check if the increasing memory address accords with the information of the added package. If yes, the package is added successfully; otherwise, the modification failed.

To ensure a higher rate of success, it is suggested that you add the package before the half package interrupt of the penultimate package.

11.1.5 Register List

Module Name	Base Address
DMAC	0x03002000

Register Name	Offset	Description
DMAC_IRQ_EN_REG0	0x0000	DMAC IRQ Enable Register 0
DMAC_IRQ_EN_REG1	0x0004	DMAC IRQ Enable Register 1
DMAC_IRQ_PEND_REG0	0x0010	DMAC IRQ Pending Status Register 0
DMAC_IRQ_PEND_REG1	0x0014	DMAC IRQ Pending Status Register 1
DMAC_AUTO_GATE_REG	0x0028	DMAC Auto Gating Register
DMAC_STA_REG	0x0030	DMAC Status Register
DMAC_EN_REGN	0x0100 + N*0x0040	DMAC Channel Enable Register N (N = 0 to 11)
DMAC_PAU_REGN	0x0104 + N*0x0040	DMAC Channel Pause Register N (N = 0 to 11)
DMAC_DESC_ADDR_REGN	0x0108 + N*0x0040	DMAC Channel Start Address Register N (N = 0 to 11)
DMAC_CFG_REGN	0x010C + N*0x0040	DMAC Channel Configuration Register N (N = 0 to 11)
DMAC_CUR_SRC_REGN	0x0110 + N*0x0040	DMAC Channel Current Source Register N (N = 0 to 11)

Register Name	Offset	Description
DMAC_CUR_DEST_REGN	0x0114 + N*0x0040	DMAC Channel Current Destination Register N (N = 0 to 11)
DMAC_BCNT_LEFT_REGN	0x0118 + N*0x0040	DMAC Channel Byte Counter Left Register N (N = 0 to 11)
DMAC_PARA_REGN	0x011C + N*0x0040	DMAC Channel Parameter Register N (N = 0 to 11)
DMAC_MODE_REGN	0x0128 + N*0x0040	DMAC Mode Register N (N = 0 to 11)
DMAC_FDESC_ADDR_REGN	0x012C + N*0x0040	DMAC Former Descriptor Address Register N (N = 0 to 11)
DMAC_PKG_NUM_REGN	0x0130 + N*0x0040	DMAC Package Number Register N (N = 0 to 11)

11.1.6 Register Description

11.1.6.1 0x0000 DMAC IRQ Enable Register 0 (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: DMAC_IRQ_EN_REG0
Bit	Read/Write	Default/Hex	Description
31	/	/	/
30	R/W	0x0	DMA7_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 7 0: Disabled 1: Enabled
29	R/W	0x0	DMA7_PKG_IRQ_EN Enable the package end interrupt of DMA channel 7 0: Disabled 1: Enabled
28	R/W	0x0	DMA7_HLAF_IRQ_EN Enable the half package interrupt of DMA channel 7 0: Disabled 1: Enabled
27	/	/	/
26	R/W	0x0	DMA6_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 6 0: Disabled 1: Enabled
25	R/W	0x0	DMA6_PKG_IRQ_EN Enable the package end interrupt of DMA channel 6 0: Disabled 1: Enabled

Offset: 0x0000			Register Name: DMAC_IRQ_EN_REG0
Bit	Read/Write	Default/Hex	Description
24	R/W	0x0	DMA6_HLAF_IRQ_EN Enable the half package interrupt of DMA channel 6 0: Disabled 1: Enabled
23	/	/	/
22	R/W	0x0	DMA5_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 5 0: Disabled 1: Enabled
21	R/W	0x0	DMA5_PKG_IRQ_EN Enable the package end interrupt of DMA channel 5 0: Disabled 1: Enabled
20	R/W	0x0	DMA5_HLAF_IRQ_EN Enable the half package interrupt of DMA channel 5 0: Disabled 1: Enabled
19	/	/	/
18	R/W	0x0	DMA4_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 4 0: Disabled 1: Enabled
17	R/W	0x0	DMA4_PKG_IRQ_EN Enable the package end interrupt of DMA channel 4 0: Disabled 1: Enabled
16	R/W	0x0	DMA4_HLAF_IRQ_EN Enable the half package interrupt of DMA channel 4 0: Disabled 1: Enabled
15	/	/	/
14	R/W	0x0	DMA3_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 3 0: Disabled 1: Enabled
13	R/W	0x0	DMA3_PKG_IRQ_EN Enable the package end interrupt of DMA channel 3 0: Disabled 1: Enabled

Offset: 0x0000			Register Name: DMAC_IRQ_EN_REG0
Bit	Read/Write	Default/Hex	Description
12	R/W	0x0	DMA3_HLAF_IRQ_EN Enable the half package interrupt of DMA channel 3 0: Disabled 1: Enabled
11	/	/	/
10	R/W	0x0	DMA2_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 2 0: Disabled 1: Enabled
9	R/W	0x0	DMA2_PKG_IRQ_EN Enable the package end interrupt of DMA channel 2 0: Disabled 1: Enabled
8	R/W	0x0	DMA2_HLAF_IRQ_EN Enable the half package interrupt of DMA channel 2 0: Disabled 1: Enabled
7	/	/	/
6	R/W	0x0	DMA1_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 1 0: Disabled 1: Enabled
5	R/W	0x0	DMA1_PKG_IRQ_EN Enable the package end interrupt of DMA channel 1 0: Disabled 1: Enabled
4	R/W	0x0	DMA1_HLAF_IRQ_EN Enable the half package interrupt of DMA channel 1 0: Disabled 1: Enabled
3	/	/	/
2	R/W	0x0	DMA0_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 0 0: Disabled 1: Enabled
1	R/W	0x0	DMA0_PKG_IRQ_EN Enable the package end interrupt of DMA channel 0 0: Disabled 1: Enabled

Offset: 0x0000			Register Name: DMAC_IRQ_EN_REG0
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	DMA0_HLAF_IRQ_EN Enable the half package interrupt of DMA channel 0 0: Disabled 1: Enabled

11.1.6.2 0x0004 DMAC IRQ Enable Register 1 (Default Value: 0x0000_0000)

Offset: 0x0004			Register Name: DMAC_IRQ_EN_REG1
Bit	Read/Write	Default/Hex	Description
31:15	/	/	/
14	R/W	0x0	DMA11_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 11 0: Disabled 1: Enabled
13	R/W	0x0	DMA11_PKG_IRQ_EN Enable the package end interrupt of DMA channel 11 0: Disabled 1: Enabled
12	R/W	0x0	DMA11_HALF_IRQ_EN Enable the half package interrupt of DMA channel 11 0: Disabled 1: Enabled
11	/	/	/
10	R/W	0x0	DMA10_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 10 0: Disabled 1: Enabled
9	R/W	0x0	DMA10_PKG_IRQ_EN Enable the package end interrupt of DMA channel 10 0: Disabled 1: Enabled
8	R/W	0x0	DMA10_HALF_IRQ_EN Enable the half package interrupt of DMA channel 10 0: Disabled 1: Enabled
7	/	/	/
6	R/W	0x0	DMA9_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 9 0: Disabled 1: Enabled

Offset: 0x0004			Register Name: DMAC_IRQ_EN_REG1
Bit	Read/Write	Default/Hex	Description
5	R/W	0x0	DMA9_PKG_IRQ_EN Enable the package end interrupt of DMA channel 9 0: Disabled 1: Enabled
4	R/W	0x0	DMA9_HALF_IRQ_EN Enable the half package interrupt of DMA channel 9 0: Disabled 1: Enabled
3	/	/	/
2	R/W	0x0	DMA8_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 8 0: Disabled 1: Enabled
1	R/W	0x0	DMA8_PKG_IRQ_EN Enable the package end interrupt of DMA channel 8 0: Disabled 1: Enabled
0	R/W	0x0	DMA8_HALF_IRQ_EN Enable the half package interrupt of DMA channel 8 0: Disabled 1: Enabled

11.1.6.3 0x0010 DMAC IRQ Pending Status Register 0 (Default Value: 0x0000_0000)

Offset: 0x0010			Register Name: DMAC_IRQ_PEND_REG0
Bit	Read/Write	Default/Hex	Description
31	/	/	/
30	R/W1C	0x0	DMA7_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 7. Write 1 to clear the pending status. 0: No effect 1: Pending
29	R/W1C	0x0	DMA7_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 7. Write 1 to clear the pending status. 0: No effect 1: Pending

Offset: 0x0010			Register Name: DMAC_IRQ_PEND_REG0
Bit	Read/Write	Default/Hex	Description
28	R/W1C	0x0	DMA7_HALF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 7. Write 1 to clear the pending status. 0: No effect 1: Pending
27	/	/	/
26	R/W1C	0x0	DMA6_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 6. Write 1 to clear the pending status. 0: No effect 1: Pending
25	R/W1C	0x0	DMA6_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 6. Write 1 to clear the pending status. 0: No effect 1: Pending
24	R/W1C	0x0	DMA6_HALF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 6. Write 1 to clear the pending status. 0: No effect 1: Pending
23	/	/	/
22	R/W1C	0x0	DMA5_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 5. Write 1 to clear the pending status. 0: No effect 1: Pending
21	R/W1C	0x0	DMA5_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 5. Write 1 to clear the pending status. 0: No effect 1: Pending

Offset: 0x0010			Register Name: DMAC_IRQ_PEND_REG0
Bit	Read/Write	Default/Hex	Description
20	R/W1C	0x0	DMA5_HALF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 5. Write 1 to clear the pending status. 0: No effect 1: Pending
19	/	/	/
18	R/W1C	0x0	DMA4_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 4. Write 1 to clear the pending status. 0: No effect 1: Pending
17	R/W1C	0x0	DMA4_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 4. Write 1 to clear the pending status. 0: No effect 1: Pending
16	R/W1C	0x0	DMA4_HALF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 4. Write 1 to clear the pending status. 0: No effect 1: Pending
15	/	/	/
14	R/W1C	0x0	DMA3_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 3. Write 1 to clear the pending status. 0: No effect 1: Pending
13	R/W1C	0x0	DMA3_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 3. Write 1 to clear the pending status. 0: No effect 1: Pending

Offset: 0x0010			Register Name: DMAC_IRQ_PEND_REG0
Bit	Read/Write	Default/Hex	Description
12	R/W1C	0x0	DMA3_HALF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 3. Write 1 to clear the pending status. 0: No effect 1: Pending
11	/	/	/
10	R/W1C	0x0	DMA2_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 2. Write 1 to clear the pending status. 0: No effect 1: Pending
9	R/W1C	0x0	DMA2_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 2. Write 1 to clear the pending status. 0: No effect 1: Pending
8	R/W1C	0x0	DMA2_HALF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 2. Write 1 to clear the pending status. 0: No effect 1: Pending
7	/	/	/
6	R/W1C	0x0	DMA1_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 1. Write 1 to clear the pending status. 0: No effect 1: Pending
5	R/W1C	0x0	DMA1_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 1. Write 1 to clear the pending status. 0: No effect 1: Pending

Offset: 0x0010			Register Name: DMAC_IRQ_PEND_REG0
Bit	Read/Write	Default/Hex	Description
4	R/W1C	0x0	DMA1_HALF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 1. Write 1 to clear the pending status. 0: No effect 1: Pending
3	/	/	/
2	R/W1C	0x0	DMA0_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 0. Write 1 to clear the pending status. 0: No effect 1: Pending
1	R/W1C	0x0	DMA0_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 0. Write 1 to clear the pending status. 0: No effect 1: Pending
0	R/W1C	0x0	DMA0_HALF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 0. Write 1 to clear the pending status. 0: No effect 1: Pending

11.1.6.4 0x0014 DMAC IRQ Pending Status Register 1 (Default Value: 0x0000_0000)

Offset: 0x0014			Register Name: DMAC_IRQ_PEND_REG1
Bit	Read/Write	Default/Hex	Description
31:15	/	/	/
14	R/W1C	0x0	DMA11_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 11. Write 1 to clear the pending status. 0: No effect 1: Pending

Offset: 0x0014			Register Name: DMAC_IRQ_PEND_REG1
Bit	Read/Write	Default/Hex	Description
13	R/W1C	0x0	DMA11_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 11. Write 1 to clear the pending status. 0: No effect 1: Pending
12	R/W1C	0x0	DMA11_HALF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 11. Write 1 to clear the pending status. 0: No effect 1: Pending
11	/	/	/
10	R/W1C	0x0	DMA10_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 10. Write 1 to clear the pending status. 0: No effect 1: Pending
9	R/W1C	0x0	DMA10_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 10. Write 1 to clear the pending status. 0: No effect 1: Pending
8	R/W1C	0x0	DMA10_HALF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 10. Write 1 to clear the pending status. 0: No effect 1: Pending
7	/	/	/
6	R/W1C	0x0	DMA9_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 9. Write 1 to clear the pending status. 0: No effect 1: Pending

Offset: 0x0014			Register Name: DMAC_IRQ_PEND_REG1
Bit	Read/Write	Default/Hex	Description
5	R/W1C	0x0	DMA9_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 9. Write 1 to clear the pending status. 0: No effect 1: Pending
4	R/W1C	0x0	DMA9_HALF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 9. Write 1 to clear the pending status. 0: No effect 1: Pending
3	/	/	/
2	R/W1C	0x0	DMA8_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 8. Write 1 to clear the pending status. 0: No effect 1: Pending
1	R/W1C	0x0	DMA8_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 8. Write 1 to clear the pending status. 0: No effect 1: Pending
0	R/W1C	0x0	DMA8_HALF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 8. Write 1 to clear the pending status. 0: No effect 1: Pending

11.1.6.5 0x0028 DMAC Auto Gating Register (Default Value: 0x0000_0000)

Offset: 0x0028			Register Name: DMAC_AUTO_GATE_REG
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/
2	R/W	0x0	DMA_MCLK_CIRCUIT Auto gating bit of DMA MCLK interface circuit 0: Auto gating enabled 1: Auto gating disabled

1	R/W	0x0	DMA_COMMON_CIRCUIT Auto gating bit of DMA common circuit 0: Auto gating enabled 1: Auto gating disabled
0	R/W	0x0	DMA_CHAN_CIRCUIT Auto gating bit of DMA channel circuit 0: Auto gating enabled 1: Auto gating disabled



NOTE

When initializing the DMA Controller, the bit [2] should be set up.

11.1.6.6 0x0030 DMAC Status Register (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: DMAC_STA_REG
Bit	Read/Write	Default/Hex	Description
31	R	0x0	MBUS FIFO STATUS 0: Empty 1: Not Empty
30:12	/	/	/
11:0	R	0x0	DMA_STATUS DMA Channel [11:0] Status The meaning of each bit: 0: Idle 1: Busy

11.1.6.7 0x0100 + N*0x0040 DMAC Channel Enable Register N (Default Value: 0x0000_0000)

Offset: 0x0100 + N*0x0040 (N = 0 to 11)			Register Name: DMAC_EN_REGN
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	DMA_EN DMA Channel Enable 0: Disabled 1: Enabled

11.1.6.8 0x0104 + N*0x0040 DMAC Channel Pause Register N (Default Value: 0x0000_0000)

Offset: 0x0104 + N*0x0040 (N = 0 to 11)			Register Name: DMAC_PAU_REGN

Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	DMA_PAUSE Pause the DMA Channel Transfer Data 0: Resume Transferring 1: Pause Transferring

11.1.6.9 0x0108 + N*0x0040 DMAC Channel Descriptor Address Register N (Default Value: 0x0000_0000)

Offset: 0x0108 + N*0x0040 (N = 0 to 11)			Register Name: DMAC_DESC_ADDR_REGN
Bit	Read/Write	Default/Hex	Description
31:2	R/W	0x0	DMA_DESC_ADDR Lower 30 bits of DMA channel descriptor address The descriptor address must be word-aligned.
1:0	R/W	0x0	DMA_DESC_HIGH_ADDR Higher 2 bits of DMA channel descriptor high address The real address is as follows. DMA Channel Descriptor Address = {bit[1:0], bit[31:2], 2'b00}

11.1.6.10 0x010C + N*0x0040 DMAC Channel Configuration Register N (Default Value: 0x0000_0000)

Offset: 0x010C + N*0x0040 (N = 0 to 11)			Register Name: DMAC_CFG_REGN
Bit	Read/Write	Default/Hex	Description
31	/	/	/
30	R	0x0	BMODE_SEL 0: Normal Mode 1: BMODE
29:27	/	/	/
26:25	R	0x0	DMA_DEST_DATA_WIDTH DMA Destination Data Width 00: 8-bit 01: 16-bit 10: 32-bit 11: 64-bit

Offset: 0x010C + N*0x0040 (N = 0 to 11)			Register Name: DMAC_CFG_REGN
Bit	Read/Write	Default/Hex	Description
24	R	0x0	DMA_ADDR_MODE DMA Destination Address Mode 0: Linear Mode 1: IO Mode
23:22	R	0x0	DMA_DEST_BLOCK_SIZE DMA Destination Block Size 00: 1 01: 4 10: 8 11: 16
21:16	R	0x0	DMA_DEST_DRQ_TYPE DMA Destination DRQ Type The details in DRQ Type and Port Corresponding Relation.
15:11	/	/	/
10:9	R	0x0	DMA_SRC_DATA_WIDTH DMA Source Data Width 00: 8-bit 01: 16-bit 10: 32-bit 11: 64-bit
8	R	0x0	DMA_SRC_ADDR_MODE DMA Source Address Mode 0: Linear Mode 1: IO Mode
7:6	R	0x0	DMA_SRC_BLOCK_SIZE DMA Source Block Size 00: 1 01: 4 10: 8 11: 16
5:0	R	0x0	DMA_SRC_DRQ_TYPE DMA Source DRQ Type The details in DRQ Type and Port Corresponding Relation.

11.1.6.11 0x0110 + N*0x0040 DMAC Channel Current Source Address Register N (Default Value: 0x0000_0000)

Offset: 0x0110 + N*0x0040 (N = 0 to 11)			Register Name: DMAC_CUR_SRC_REGN
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	DMA_CUR_SRC DMA Channel Current Source Address.

11.1.6.12 0x0114 + N*0x0040 DMAC Channel Current Destination Address Register N (Default Value: 0x0000_0000)

Offset: 0x0114 + N*0x0040 (N = 0 to 11)			Register Name: DMAC_CUR_DEST_REGN
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	DMA_CUR_DEST DMA Channel Current Destination Address.

11.1.6.13 0x0118 + N*0x0040 DMAC Channel Byte Counter Left Register N (Default Value: 0x0000_0000)

Offset: 0x0118 + N*0x0040 (N = 0 to 11)			Register Name: DMAC_BCNT_LEFT_REGN
Bit	Read/Write	Default/Hex	Description
31:25	/	/	/
24:0	R	0x0	DMA_BCNT_LEFT DMA Channel Byte Counter Left.

11.1.6.14 0x011C + N*0x0040 DMAC Channel Parameter Register N (Default Value: 0x0000_0000)

Offset: 0x011C + N*0x0040 (N = 0 to 11)			Register Name: DMAC_PARA_REGN
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R	0x0	WAIT_CYC Wait Clock Cycles

11.1.6.15 0x0128 + N*0x0040 DMAC Mode Register N (Default Value: 0x0000_0000)

Offset: 0x0128 + N*0x0040 (N = 0 to 11)			Register Name: DMAC_MODE_REGN
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
3	R/W	0x0	DMA_DST_MODE Destination Communication Mode Select 0: Waiting mode 1: Handshake mode
2	R/W	0x0	DMA_SRC_MODE Source Communication Mode Select 0: Waiting mode 1: Handshake mode
1:0	/	/	/

11.1.6.16 0x012C + N*0x0040 DMAC Former Descriptor Address Register N (Default Value: 0x0000_0000)

Offset: 0x012C + N*0x0040 (N = 0 to 11)			Register Name: DMAC_FDESC_ADDR_REGN
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	DMA_FDESC_ADDR This register is used to store the former value of DMA Channel Descriptor Address Register.

11.1.6.17 0x0130 + N*0x0040 DMAC Package Number Register N (Default Value: 0x0000_0000)

Offset: 0x0130 + N*0x0040 (N = 0 to 11)			Register Name: DMAC_PKG_NUM_REGN
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	DMA_PKG_NUM This register records the number of packages which has been completed in one transmission.

12 Video&Graphic Engine

12.1 Video Encoder

The Video Encoding supports JPEG/MJPEG encoding (JPGE).The JPGE is a high-performance JPEG encoder implemented by using hardware. It supports 64-megapixel snapshot or HD MJPEG encoding.

The JPGE has the following features:

- Supports ISO/IEC 10918-1 (CCITT T.81) baseline process (DCT sequential) encoding
- Encodes the pictures in the chrominance sampling format of YCbCr4:2:0 and YCbCr4:2:2
- Supports multiple input picture formats:
 - NV12/NV21/YUV420SP/Tile
 - NU12/NU21/YVU420SP
 - YU12/YUV420P
 - YV12/YVU420P
 - NV16/YUV422SP
 - NV61/YVU422SP
 - YU16/YUV422P
 - YV16/YVU422P
 - ARGB8888/RGBA8888/ABGR8888/BGRA8888
- Supports JPEG encoding with the performance of 1280x720@30fps
- Supports configurable picture resolutions
 - Minimum picture resolution: 16 x 16
 - Maximum picture resolution: 8192x8192
- Supports the picture width or height step of 16
- Supports configurable quantization tables for the Y component, Cb component and Cr component respectively
- Supports the MJPEG output bit rate ranging from 2 kbit/s to 60 Mbit/s

12.2 Video Decoder

The Video Decoder has the following features:

- Supports ITU-T H.264 BP/MP/HP@ level 4.2
 - Maximum video resolution: 2032 x 2032
 - Maximum decoding rate: 1920x1080@30fps@8bit
- Supports MPEG1
 - Maximum video resolution: 1920 x 1088
 - Maximum decoding rate: 1280x720@30fps
- Supports MPEG2 SP@ML, MP@HL
 - Maximum video resolution: 2048x2048
 - Maximum decoding rate: 1280x720@30fps
- Supports MPEG4 SP@L0-3,ASP@L0-5
 - Maximum video resolution: 2048x2048
 - Maximum decoding rate: 1920x1080@30fps
- Supports ITU-T H.263 Base Profile
 - Maximum video resolution: 1920 x 1088
 - Maximum decoding rate: 1280x720@30fps
- Supports JPEG JFIF file format, Not Support Non-interleaved Scan
 - Maximum video resolution: 16384 x 16384
 - Maximum decoding rate: 1280x720@30fps
- Supports PNG
 - Maximum video resolution: 2048x2048
 - Maximum decoding rate: 1024x768<100ms

12.3 Graphics Processing

12.3.1 Display Engine (DE)

The Display Engine (DE) is a hardware composer to transfer image layers from a local bus or a video buffer to the LCD interface. The DE supports many overlay windows to blend, and supports video post-processing in the video channel.

The DE has the following features:

- One display output, up to 1366x800@60fps
- Supports 2x UI channel and 1x video channel. Each channel supports 4 overlay layer

- Video input format
 - YUV422/YUV420/YUV411
 - ARGB8888/XRGB8888/RGB888/ RGB565/ARGB4444/ ARGB1555
- UI input format: ARGB8888/ARGB8565/RGB888/RGB565/ARGB4444/ARGB1555. UI0 channel supports palette format
- Supports output RGB@8bit, and dither to 6bit
- Supports gamma and csc
- One Video channel and one UI channel support zooming function
- Supports transparent overlay between VI and UI channels
- Supports updating registers in rcq mode
- Supports simple RGB write-back
- VI layer and UI layer, with a maximum output support of 1366*800
- The Video channel supports input up to 2048*1280

12.3.2 De-interlacer (DI)

The De-interlacer (DI) converts the interlaced input video frame to progressive video frame.

The DI has the following features:

- Supports video resolution from 32 x 32 to 2048 x 1280 pixel
- Supports up to 720p@60fps
- Supports Inter-field interpolation/motion adaptive de-interlace method
- Supports 8 bit NV12/NV21/YV12/planar YUV422/YUV422UV combine

12.3.3 Graphic 2D (G2D)

The Graphic 2D (G2D) engine is hardware accelerator for 2D graphic.

The G2D has the following features:

- Supports mixer layer size up to 2048 x 2048 pixels, and rotation size up to 4096x4096 pixels
- Supports gradient color fill
- Supports pre-multiply alpha image data
- Supports color key
- Supports two pipes Porter-Duff alpha blending
- Supports multiple video formats 4:2:0, 4:2:2, 4:1:1 and multiple pixel formats (8/16/24/32 bits graphics layer)

- Supports memory scan order option
- Supports any format convert function above
- Supports 1/16x to 32x resize ratio
- Supports 32-phase 8-tap horizontal anti-alias filter, 32-phase 4-tap vertical anti-alias filter.
- Supports window clip
- Supports FillRectangle, FillGradient, BitBlt, StretchBlt and MaskBlt
- Supports horizontal and vertical flip, clockwise 0/90/180/270 degree rotate
- Supports 4 Thread RCQ function for mixer and rotate
- Supports dither
- Supports ARGB8565 input format



13 Video Output

13.1 LVDS

13.1.1 Overview

LVDS implements the conversion of HV timing data from TCON_LCD into LVDS protocol format and output them to panels with LVDS interfaces.

The LVDS module has the following features:

LVDS Host

- Compatible with TIA/EIA 644-A standard
- Compatible with VESA/JEIDA format
- 8/6-bit format output
- Single LVDS driving one panel, up to 1366x768@60fps
- LVDS Lane remap

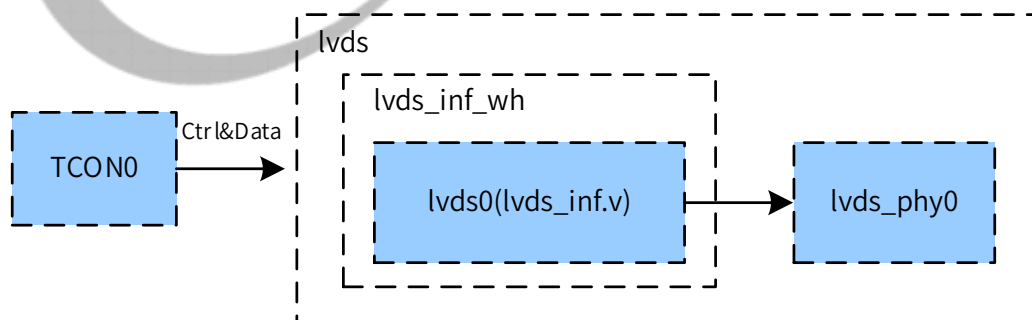
LVDS PHY

- Up to 600 MHz's bandwidth for LVDS data lane transmission
- Polarity reversing for output signal
- LVDS PHY, DSI PHY, and RGB PHY combo

13.1.2 Block Diagram

The following figure shows a block diagram of the LVDS.

Figure 13-1 LVDS Block Diagram



13.1.3 External Signals

The following table describes the external signals of LVDS.

Table 13-1 LVDS External Signals

Signal Name	Description	Type
LVDS0-CKP	LVDS0 Differential Clock (Positive)	AO
LVDS0-CKN	LVDS0 Differential Clock (Negative)	AO
LVDS0-D0P	LVDS0 Differential Data 0 (Positive)	AO
LVDS0-D0N	LVDS0 Differential Data 0 (Negative)	AO
LVDS0-D1P	LVDS0 Differential Data 1 (Positive)	AO
LVDS0-D1N	LVDS0 Differential Data 1 (Negative)	AO
LVDS0-D2P	LVDS0 Differential Data 2 (Positive)	AO
LVDS0-D2N	LVDS0 Differential Data 2 (Negative)	AO
LVDS0-D3P	LVDS0 Differential Data 3 (Positive)	AO
LVDS0-D3N	LVDS0 Differential Data 3 (Negative)	AO

13.1.4 Functional Description

13.1.4.1 LVDS Interface Timing

8/6-bit Mode

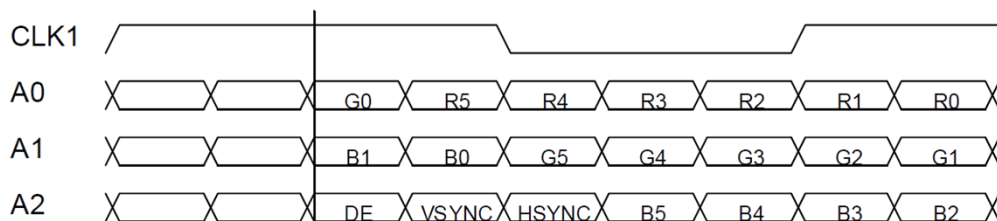
LVDS protocol requires that one set of RGB data transmission is implemented in 7 clock cycles. There is 7-bit video data transmitted in one clock cycle and the duty cycle of clock high/low logic level is 4:3. With more differential pairs, the volume of data that is able to be sent at a time can become larger.

The IP includes one clock lane and four data lanes for one LVDS interface, supporting 6-bit (RGB: 18-bit) mode and 8-bit (RGB: 24-bit) mode. Data lanes transmit both video data and video synchronization signals {DE, VSYNC, HSYNC} (see Figure 13-2).

In 6-bit mode, only three data lanes are available. In this case, RGB [7:2] in 8-bit mode are transmitted taken as RGB [5:0] in 6-bit mode.

In 8-bit mode, four data lanes are used to send 28-bit (4*7 bits) data totally, which contains 24-bit pixel information, 3-bit timing information, and 1-bit invalid bit.

Figure 13-2 VESA 6-Bit format



JEDIA/VSEA Format

LVDS protocol supports two video timing formats: JEDIA mode and VSEA mode. As Figure 13-3 and Figure 13-4 shown below, the two video data format modes have different data sending bits with a similar sending method. For VSEA mode transmission, RGB data and video

synchronization signals should be remapped to perform serial output in 7 clock cycles. See Figure 13-3 for output sequence. For JEDIA mode transmission, you can get the same transmission sequence as that of VSEA mode by setting the lower 2 bits of video data to the higher 2 bits.

Figure 13-3 Single LVDS VSEA Mode

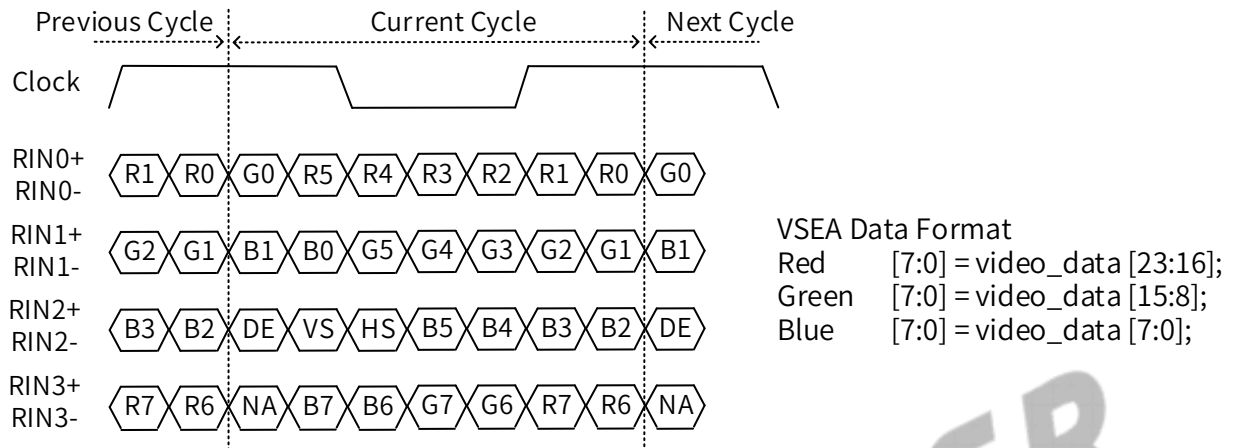
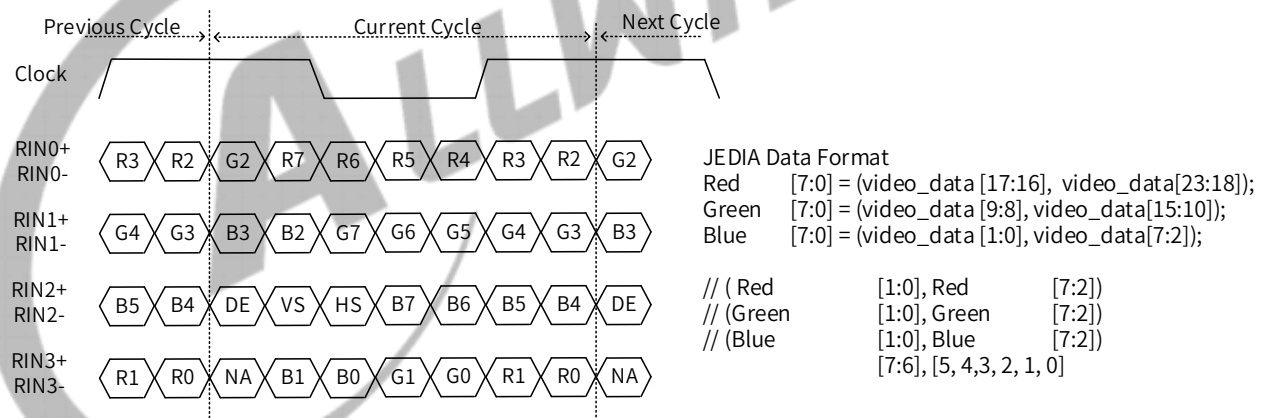


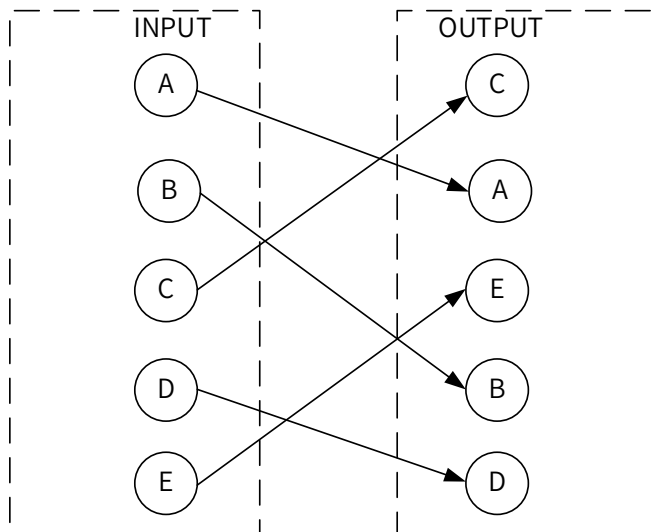
Figure 13-4 Single LVDS JEDIA Mode



13.1.4.2 Lane Remap Function

LVDS module supports 5-lane random mapping between digital input LVDS_IN [4:0] and analog PHY output, and supports mapping relationship changing through register configuration to achieve more flexibility of PAD output sequence. The following figure shows an example to explain lane mapping. The input sequence is {A, B, C, D, E} while the output sequence is remapped as {C, A, E, B, D}.

Figure 13-5 Interface Mapping



13.1.5 Programming Guidelines



NOTE

LVDS controller implements data conversion from RGB timing to LVDS protocol. TCON_LCD should be configured as HV timing and LVDS controller configuration should be finished before TCON_LCD is enabled.

Step 1 Configure LVDS controller.

- Configure [LCD_LVDS_IF_REG](#).LCD_LVDS_DATA_POL to set data/clock polarity of LVDS.
- Configure [LCD_LVDS_IF_REG](#).LCD_LVDS_BITWIDTH to set the bit width of pixel data.
- Configure [LCD_LVDS_IF_REG](#).LCD_LVDS_MODE to set the LVDS data format.
- Configure [LCD_LVDS_IF_REG](#).LCD_LVDS_DIR to set the LVDS output direction.
- Configure [LCD_LVDS_IF_REG](#).LCD_LVDS_LINK to select LVDS work mode as single link.
- Configure [LCD_LVDS_IF_REG](#).LCD_LVDS_EN t as 0x1 to enable LVDS controller.

Step 2 Take PHY0 as an example to configure analog LVDS PHY. The procedure for PHY1 (LCD_LVDS_ANA1_REG) is consistent with that of PHY0.

- Configure [LCD_LVDS_ANA0_REG](#).LVDS_REG_PLR to select whether the polarity of LVDS data channel [3:0] is reversed.
- Configure [LCD_LVDS_ANA0_REG](#).LVDS_REG_PLRC to select whether the polarity of LVDS clock channel is reversed.
- Configure [LCD_LVDS_ANA0_REG](#).LVDS_REG_R to set the amplitude of the common signals.
- Configure [LCD_LVDS_ANA0_REG](#).LVDS_REG_C to set the amplitude of the differential signals.

- e) Configure [LCD_LVDS_ANA0_REG](#).LVDS_HPREN_DRV as 0xf to enable data channel [3:0] drive.
- f) Configure [LCD_LVDS_ANA0_REG](#).LVDS_HPREN_DRVC as 0x1 to enable clock channel drive.
- g) Configure [LCD_LVDS_ANA0_REG](#).LVDS_EN_MB as 0x1 to enable analog circuit of LVDS PHY.

13.1.6 Register List

Module Name	Base Address	Comments
LVDS	0x05461000	Share the same address space with TCON_LCD.

Register Name	Offset	Description
LCD_LVDS_IF_REG	0x0084	LCD LVDS Interface Register
LCD_LVDS_ANA0_REG	0x0220	LCD LVDS Analog Register0
LCD_LVDS_LMAP0_REG	0x0248	LCD LVDS Lane Map Register0

13.1.7 Register Description

13.1.7.1 0x0084 LCD LVDS Interface Register (Default Value: 0x0000_0000)

Offset: 0x0084			Register Name: LCD_LVDS_IF_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	LCD_LVDS_EN Enable LVDS Interface. 0: Disable 1: Enable
30	R/W	0x0	LCD_LVDS_LINK 0: Single link 1: Reserved
29	/	/	/
28	R/W	0x0	LCD_LVDS_DIR Set the LVDS direction 0: Normal 1: Reverse
27	R/W	0x0	LCD_LVDS_MODE Set the LVDS data mode 0: NS mode(VESA) 1: JEIDA mode

Offset: 0x0084			Register Name: LCD_LVDS_IF_REG
Bit	Read/Write	Default/Hex	Description
26	R/W	0x0	LCD_LVDS_BITWIDTH Set the bit width of data 0: 24bit 1: 18bit
25	R/W	0x0	LCD_LVDS_DEBUG_EN Enable LVDS debug function 0: Disable 1: Enable
24	R/W	0x0	LCD_LVDS_DEBUG_MODE Set the output signal in debug mode. 0: Mode0 Random data (generated by PRBS10) 1: Mode1 Output CLK period=7/2 LVDS CLK period
23	R/W	0x0	LCD_LVDS_CORRECT_MODE Set the LVDS correct mode. 0: Mode0. Phase calibration can be performed in any region. 1: Mode1. Phase calibration can be performed in the blank region of the image.
22:21	/	/	/
20	R/W	0x0	LCD_LVDS_CLK_SEL Select the clock source of LVDS. 0: Revered 1: LCD CLK
19:5	/	/	/
4:0	R/W	0x0	LCD_LVDS_DATA_POL Set the data/clock polarity of LVDS. Bit4: LVDS data [3] channel polarity Bit3: LVDS clock channel polarity Bit2: LVDS data [2] channel polarity Bit1: LVDS data [1] channel polarity Bit0: LVDS data [0] channel polarity 0: Reverse 1: Normal Note: When LCD_LVDS_DIR=1, the channel order corresponding to these bits will be reversed.

13.1.7.2 0x0220 LCD LVDS Analog Register0 (Default Value: 0x0000_0000)

Offset: 0x0220			Register Name: LCD_LVDS_ANA0_REG
Bit	Read/Write	Default/Hex	Description

Offset: 0x0220			Register Name: LCD_LVDS_ANA0_REG
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	LVDS_TEST_EN Test Enable 0: Disable 1: Enable. After enabled, LVDS output depends on the configuration of LVDS_TDATA_SEL. Note: LCD_LVDS_DIR should be set to 0x0.
6:5	R/W	0x0	LVDS_TDATA_SEL 00: LVDS data channel output is 0. 01: LVDS data channel output is 1. 10: LVDS data channel output is Vsync. 11: LVDS data channel output is Hsync.
4:0	/	/	/

13.1.7.3 0x0248 LCD LVDS Lane Map Register0 (Default Value:0x0000_0000)

Offset: 0x0248			Register Name: LCD_LVDS_LMAP0_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	LVDS_LMAP_EN 0: Disable, disable channel mapping logic. 1: Enable, enable channel mapping logic.
30:19	/	/	/
18:16	R/W	0x0	LANE4_MAP_SOURCE Physical channel lane4 mapping data source. 000: Data comes from address channel 0. 001: Data comes from address channel 1. 010: Data comes from address channel 2. 011: Data comes from address channel 3. 100: Data comes from address channel 4. Others: Reserved
15	/	/	/
14:12	R/W	0x0	LANE3_MAP_SOURCE Physical channel lane3 mapping data source. 000: Data comes from address channel 0. 001: Data comes from address channel 1. 010: Data comes from address channel 2. 011: Data comes from address channel 3. 100: Data comes from address channel 4. Others: Reserved
11	/	/	/

Offset: 0x0248			Register Name: LCD_LVDS_LMAP0_REG
Bit	Read/Write	Default/Hex	Description
10:8	R/W	0x0	LANE2_MAP_SOURCE Physical channel lane2 mapping data source. 000: Data comes from address channel 0. 001: Data comes from address channel 1. 010: Data comes from address channel 2. 011: Data comes from address channel 3. 100: Data comes from address channel 4. Others: Reserved
7	/	/	/
6:4	R/W	0x0	LANE1_MAP_SOURCE Physical channel lane1 mapping data source. 000: Data comes from address channel 0. 001: Data comes from address channel 1. 010: Data comes from address channel 2. 011: Data comes from address channel 3. 100: Data comes from address channel 4. Others: Reserved
3	/	/	/
2:0	R/W	0x0	LANE0_MAP_SOURCE Physical channel lane0 mapping data source. 000: Data comes from address channel 0. 001: Data comes from address channel 1. 010: Data comes from address channel 2. 011: Data comes from address channel 3. 100: Data comes from address channel 4. Others: Reserved

13.2 MIPI DSI

13.2.1 Overview

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 V1.0.

The MIPI DSI includes the following features:

- Compliance with MIPI DSI V1.02
- Supports 4-lane MIPI DSI, up to 1280x800@60fps
- Supports non-burst mode with sync pulse/sync event and burst mode
- Pixel format: RGB888, RGB666, RGB666 loosely packed and RGB565
- Continuous and non-continuous lane clock modes
- Generic commands support bidirectional communication in LP through data lane 0
- Supports low power data transmission
- Supports ULPS and escape modes
- Supports hardware checksum

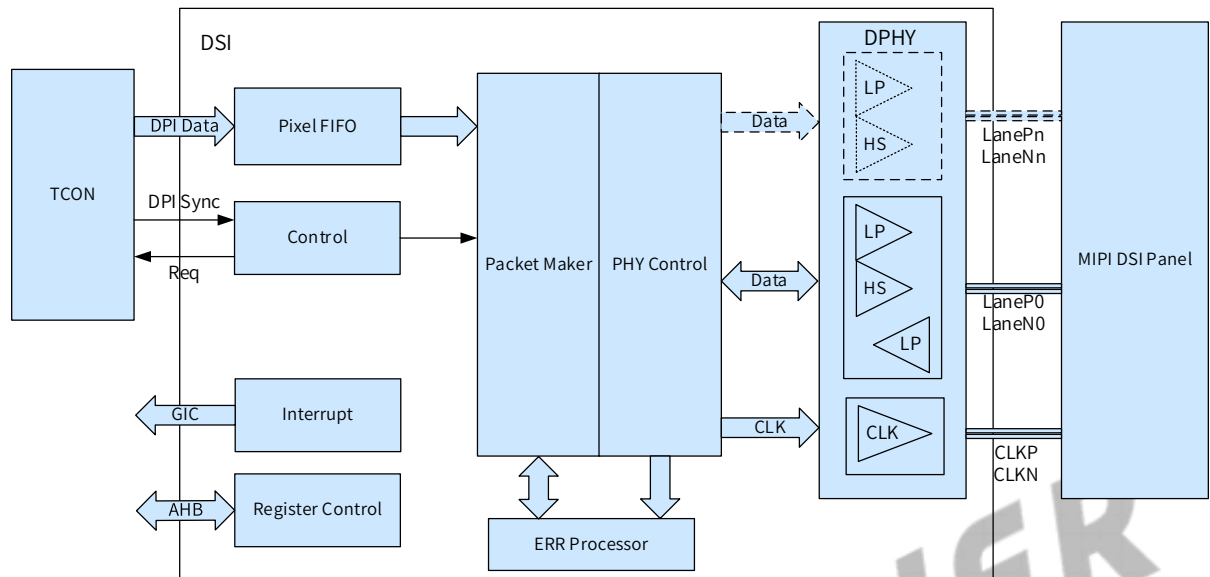
The MIPI DSI includes the following features:

- Compliance with D-PHY v1.2
- Support up to 4 lanes and lane speed up to 1.0 Gbps per lane in forward direction
- Aggregate throughput up to 4.0 Gbps with 4 data lanes
- Bidirectional communication in Low Power mode through data lane 0
- Support Low Power Escape modes and Ultra Low Power state
- Support multiplex with GPIO, GPIO can work in 3.3V or 1.8V power supply
- 1.8V±5% analog supply operation

13.2.2 Block Diagram

The following figure shows a block diagram of the MIPI DSI.

Figure 13-6 MIPI DSI Block Diagram



13.2.3 External Signals

The following table describes the external signals of MIPI DSI.

Table 13-2 MIPI DSI External Signals

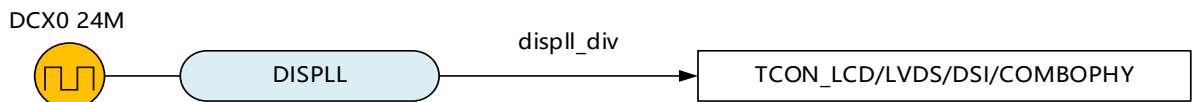
Signal Name	Description	Type
DSI-CKP	DSI Differential Clock (Positive)	AO
DSI-CKN	DSI Differential Clock (Negative)	AO
DSI-D0P	DSI Differential Data Line 0 (Positive)	A I/O
DSI-D0N	DSI Differential Data Line 0 (Negative)	A I/O
DSI-D1P	DSI Differential Data Line (Positive)	AO
DSI-D1N	DSI Differential Data Line (Negative)	AO
DSI-D2P	DSI Differential Data Line (Positive)	AO
DSI-D2N	DSI Differential Data Line (Negative)	AO
DSI-D3P	DSI Differential Data Line (Positive)	AO
DSI-D3N	DSI Differential Data Line (Negative)	AO

13.2.4 Functional Description

13.2.4.1 DISPLL Distribution

The following figure shows the DISPLL distribution for modules.

Figure 13-7 DISPLL Distribution



DISPLL is located inside the VIDEO_OUT_SYS system and mainly provides the function clock for the TCON_LCD/LVDS/DSI/COMBOPHY module.

DISPLL can output two clocks:

- **clk_hs**: 1 GHz at max. It is mainly used as DPHY serial clock (only used for output to the DSI interface).
- **clk_ls**: 1.2 GHz at max. It serves as the function clock for the TCON_LCD module (used to output RGB/LVDS/DSI interface timing). This clock is divided internally in TCON_LCD to obtain the DSI/LVDS module function clock.

13.2.4.2 DISPLL Specifications

The specifications of DISPLL are shown in the following table.

Table 13-3 DISPLL Specifications

PLL	Stable Operating Frequency	Actual Operating Frequency	Spread Spectrum	Linear FM	Pk-Pk	Lock Time
DISPLL	1.26 GHz-2.52 GHz	≤ 2.4 GHz	Yes	Yes	70ps@1GHz z	0.1 ms

13.3 TCON LCD

13.3.1 Overview

The Timing Controller_LCD (TCON_LCD) is a module that processes video signals received from system through a complicated arithmetic and then generates control signals and transmits them to the LCD panel driver IC.

The TCON_LCD includes the following features:

- One TCON_LCD controller, supporting LCD/DSI/LVDS display
- Supports parrallel RGB interface, up to 1280x720@60fps

- Supports serial RGB/DUMMY RGB interface, up to 800 x 480@60fps
- Supports i8080 interface, up to 800x480@60fps
- Supports BT.656 interface
- Supports RGB888, RGB666 and RGB565 pixel formats
- Supports Fsync function and outputting pulses to the sensor for synchronization
- Single-lane RGB maximum rate is no more than 168 MHz

13.3.2 Block Diagram

Figure 13-8 TCON_LCD Block Diagram

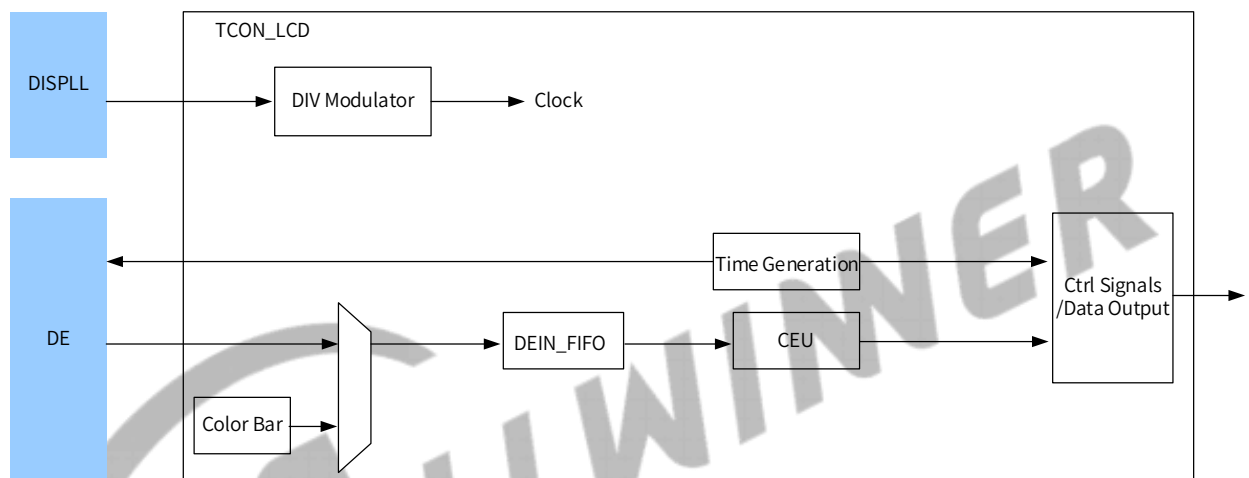


Table 13-4 TCON_LCD Sub-blocks

Sub-blocks	Description
ColorBar (Vdbist)	The circuit of embedded test data source in TCON. It provides RGB colorbar pattern, grey pattern, lattice pattern, and grid pattern.
Div Modulator	It is responsible for clock division and phase adjustment.
DEIN_FIFO	It supports temporary storage and backpressure on the DE data stream. It can implement data interaction between TCON_LCD and DE.
CEU	It supports strengthening or adjusting the RGB color, and converting from RGB to YUV.
Timing Generation	It generates the corresponding timing based on the configured parameters and the row state machine and column state machine in the module.

13.3.3 External Signals

The following table describes the external I/O signals of LCD.

Table 13-5 TCON_LCD External Signals

Signal Name	Description	Type

Signal Name	Description	Type
LCD0-CLK	LCD0 Clock	O
LCD0-VSYNC	LCD0 Vertical Synchronization	O
LCD0-HSYNC	LCD0 Horizontal Synchronization	O
LCD0-DE	LCD0 Data Enable	O
LCD0-D0	LCD0 Data Input/Output 0	I/O
LCD0-D1	LCD0 Data Input/Output 1	I/O
LCD0-D2	LCD0 Data Input/Output 2	I/O
LCD0-D3	LCD0 Data Input/Output 3	I/O
LCD0-D4	LCD0 Data Input/Output 4	I/O
LCD0-D5	LCD0 Data Input/Output 5	I/O
LCD0-D6	LCD0 Data Input/Output 6	I/O
LCD0-D7	LCD0 Data Input/Output 7	I/O
LCD0-D8	LCD0 Data Input/Output 8	I/O
LCD0-D9	LCD0 Data Input/Output 9	I/O
LCD0-D10	LCD0 Data Input/Output 10	I/O
LCD0-D11	LCD0 Data Input/Output 11	I/O
LCD0-D12	LCD0 Data Input/Output 12	I/O
LCD0-D13	LCD0 Data Input/Output 13	I/O
LCD0-D14	LCD0 Data Input/Output 14	I/O
LCD0-D15	LCD0 Data Input/Output 15	I/O
LCD0-D16	LCD0 Data Input/Output 16	I/O
LCD0-D17	LCD0 Data Input/Output 17	I/O
LCD0-D18	LCD0 Data Input/Output 18	I/O
LCD0-D19	LCD0 Data Input/Output 19	I/O
LCD0-D20	LCD0 Data Input/Output 20	I/O
LCD0-D21	LCD0 Data Input/Output 21	I/O
LCD0-D22	LCD0 Data Input/Output 22	I/O
LCD0-D23	LCD0 Data Input/Output 23	I/O

For parallel RGB, the data of LCD is high-aligned. The correspondence is as follows.

Table 13-6 The Correspondence between LCD and RGB

LCD I/O	Parallel RGB I/O		
	RGB565	RGB666	RGB888
LCD-D23	R4	R5	R7
LCD-D22	R3	R4	R6
LCD-D21	R2	R3	R5
LCD-D20	R1	R2	R4
LCD-D19	R0	R1	R3
LCD-D18	-	R0	R2
LCD-D17	-	-	R1
LCD-D16	-	-	R0
LCD-D15	G5	G5	G7

LCD I/O	Parallel RGB I/O		
	RGB565	RGB666	RGB888
LCD-D14	G4	G4	G6
LCD-D13	G3	G3	G5
LCD-D12	G2	G2	G4
LCD-D11	G1	G1	G3
LCD-D10	G0	G0	G2
LCD-D9	-	-	G1
LCD-D8	-	-	G0
LCD-D7	B4	B5	B7
LCD-D6	B3	B4	B6
LCD-D5	B2	B3	B5
LCD-D4	B1	B2	B4
LCD-D3	B0	B1	B3
LCD-D2	-	B0	B2
LCD-D1	-	-	B1
LCD-D0	-	-	B0

The multiplex relationship among LCD, LVDS, and DSI is shown as follows.

Table 13-7 The Correspondence among LCD, LVDS, and DSI.

LCD I/O	LVDS I/O	DSI I/O
LCD-D0	/	/
LCD-D1	/	/
LCD-D2	LVDS0-D0P	DSI-D0P
LCD-D3	LVDS0-D0N	DSI-D0N
LCD-D4	LVDS0-D1P	DSI-D1P
LCD-D5	LVDS0-D1N	DSI-D1N
LCD-D6	LVDS0-D2P	DSI-CKP
LCD-D7	LVDS0-D2N	DSI-CKN
LCD-D8	/	/
LCD-D9	/	/
LCD-D10	LVDS0-CKP	DSI-D2P
LCD-D11	LVDS0-CKN	DSI-D2N
LCD-D12	LVDS0-D3P	DSI-D3P
LCD-D13	LVDS0-D3N	DSI-D3N
LCD-D14	/	/
LCD-D15	/	/
LCD-D16	/	/
LCD-D17	/	/
LCD-D18	/	/
LCD-D19	/	/
LCD-D20	/	/
LCD-D21	/	/

LCD I/O	LVDS I/O	DSI I/O
LCD-D22	/	/
LCD-D23	/	/
LCD-CLK	/	/
LCD-DE	/	/
LCD-HSYNC	/	/
LCD-VSYNC	/	/



13.3.4 Functional Description

13.3.4.1 Control signal and data port mapping



NOTE

The meaning of 256K in CPU mode is $2^{18}/1024=256K$.

In CPU TRIG mode, TCON_TRIG signal is multiplexed with VSYNC signal.

		SYNC RGB				YUV		DC	CPU Cmd	CPU 18-bit	CPU 18bit								CPU1 6 bit	CPU 18bit			CPU 16bit		CPU 18bit		LVDS	DSI
External I/O	Internal Pin	Para RGB	Serial RGB			BT656	BT601	YUV422		256K	256K								64K	256K			64K		256K		Single Link	
			1 st	2 nd	3 rd						1 st	2 nd	3 rd	1 st	2 nd	1 st	2 nd	1 st		2 nd	3 rd	1 st	2 nd	1 st	2 nd	LVDS0		
LCD0_VSYNC	IO0	VSYNC					VSYNC	VSYNC	CS																			
LCD0_HSYNC	IO1	HSYNC					HSYNC		RD																			
LCD0_CLK	IO2	DCLK				DCLK	DCLK	DCLK	WR																			
LCD0_DE	IO3	DE					DE	HSYNC	RS																			
LCD0_D23	D23	R7							D23	R5	R5	B5	G5	R5		R5	B5	R4										
LCD0_D22	D22	R6							D22	R4	R4	B4	G4	R4		R4	B4	R3										
LCD0_D21	D21	R5							D21	R3	R3	B3	G3	R3		R3	B3	R2										
LCD0_D20	D20	R4							D20	R2	R2	B2	G2	R2		R2	B2	R1										
LCD0_D19	D19	R3							D19	R1	R1	B1	G1	R1		R1	B1	R0										
LCD0_D18	D18	R2							D18	R0	R0	B0	G0	R0		R0	B0	G5										
LCD0_D17	D17	R1							D17																			
LCD0_D16	D16	R0							D16																			
LCD0_D15	D15	G7							D15	G5								G4										
LCD0_D14	D14	G6							D14	G4								G3										
LCD0_D13	D13	G5							D13	G3																D3N	D3N	
LCD0_D12	D12	G4	D71	D72	D73	D7	D7	D7	D12	G2	G5	R5	B5	G5	B5	G5		G2	R5	G5	B5	R4	G2	R5	G2	D3P	D3P	
LCD0_D11	D11	G3	D61	D62	D63	D6	D6	D6	D11	G1	G4	R4	B4	G4	B4	G4		G1	R4	G4	B4	R3	G1	R4	G1	CKN	D2N	
LCD0_D10	D10	G2	D51	D52	D53	D5	D5	D5	D10	G0	G3	R3	B3	G3	B3	G3		G0	R3	G3	B3	R2	G0	R3	G0	CKP	D2P	
LCD0_D9	D9	G1							D9																			
LCD0_D8	D8	G0							D8																			
LCD0_D7	D7	B7	D41	D42	D43	D4	D4	D4	D7	B5	G2	R2	B2	G2	B2	G2		B4	R2	G2	B2	R1	B4	R2	B5	D2N	CKN	
LCD0_D6	D6	B6	D31	D32	D33	D3	D3	D3	D6	B4	G1	R1	B1	G1	B1	G1		B3	R1	G1	B1	R0	B3	R1	B4	D2P	CKP	
LCD0_D5	D5	B5	D21	D22	D23	D2	D2	D2	D5	B3	G0	R0	B0	G0	B0	G0		B2	R0	G0	B0	G5	B2	R0	B3	D1N	D1N	
LCD0_D4	D4	B4	D11	D12	D13	D1	D1	D1	D4	B2								B1				G4	B1	G5	B2	D1P	D1P	

		SYNC RGB				YUV		DC	CPU Cmd	CPU 18-bit	CPU 18bit							CPU1 6 bit	CPU 18bit			CPU 16bit		CPU 18bit		LVDS	DSI
External I/O	Internal Pin	Para RGB	Serial RGB			BT656	BT601	YUV422		256K	256K							64K	256K			64K		256K		Single Link	
			1 st	2 nd	3 rd						1 st	2 nd	3 rd	1 st	2 nd	1 st	2 nd		1 st	2 nd	3 rd	1 st	2 nd	1 st	2 nd	LVDS0	
LCD0_D3	D3	B3	D01	D02	D03	D0	D0	D0	D3	B1								B0				G3	B0	G4	B1	D0N	D0N
LCD0_D2	D2	B2							D2	B0														G3	B0	D0P	D0P
LCD0_D1	D1	B1							D1																		
LCD0_D0	D0	B0							D0																		



13.3.4.2 HV interface (Sync+DE mode)

HV I/F is also known as Sync + DE mode, which is widely used in TFT LCD module for PMP/MP4 applications.

Its signals are defined as:

Table 13-8 HV Panel Signals

Signal Name	Description	Type
Vsync	Vertical sync, indicates one new frame	O
Hsync	Horizontal sync, indicates one new scan line	O
DCLK	Dot clock, pixel data are sync by this clock	O
DE	LCD data enable	O
D[23..0]	24-bit RGB output from input FIFO for panel	O

The timing diagram of HV interface is as follows.

Figure 13-9 HV Interface Vertical Timing

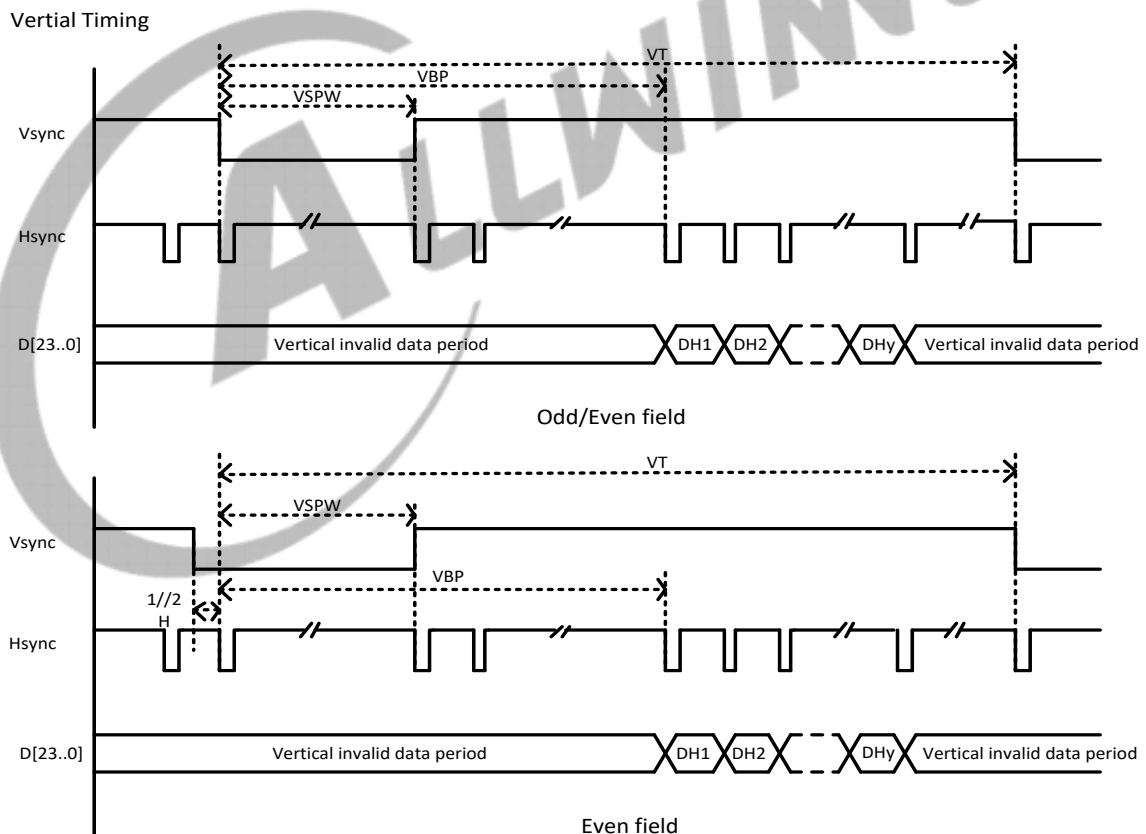
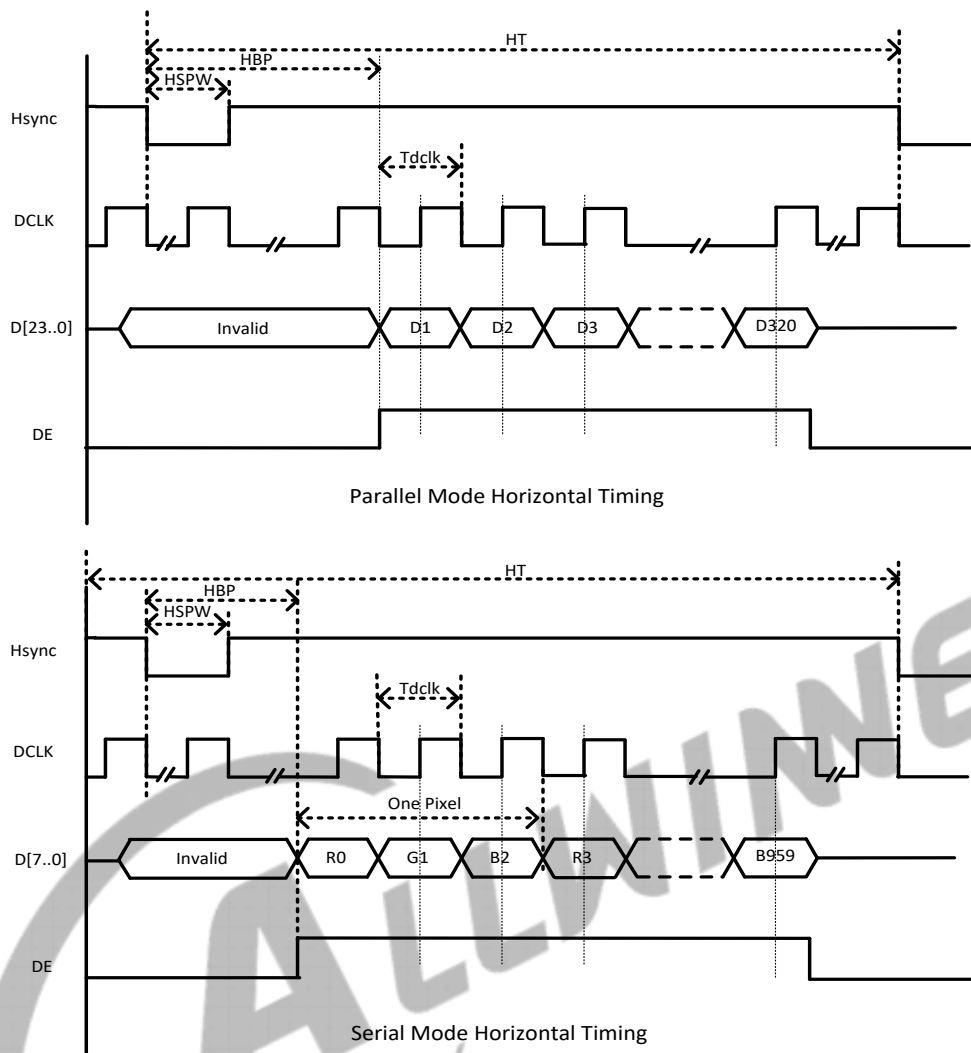


Figure 13-10 HV Interface Horizontal Timing



13.3.4.3 BT.656 Interface

The BT.656 transmission protocol is needed when outputting the data in YUV format in the HV mode. Compared with the standard CSI interface, BT.656 has no SYNC signals like Field, VSYN, and HSYN. Instead, it embeds the field synchronization information of image transmission in the data stream transmitted by Data Bus. The Data Bus bit width of BT.656 is 8bit. And the transmitting image data type is the interlaced YUV422. The encoding format for each row is as follows:

Row=End of Active Video (EAV) + horizontal blanking data (80H/10H) + Start of Active Video (SAV) + valid data (UYVY)

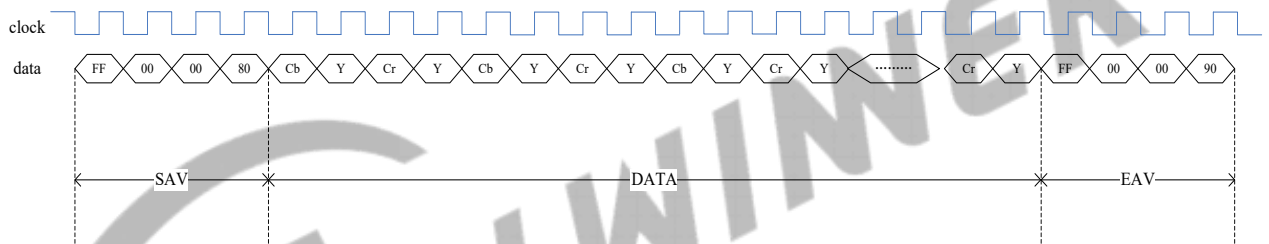
Both of the embedded EAV and SAV are four bytes in the format of FF 00 00 XY. The first three types are fixed to 8'hFF 8'h00 8'h00 in hexadecimal. The last bytes XY are used as the control word. The specific image row and field synchronization information is as follows.

Table 13-9 Analysis of XY Signal

BIT	7	6	5	4	3	2	1	0
XY	1'b1	F	V	H	P3	P2	P1	P0
F	0: During field 1 1: During field 2							
V	1: During field blanking 0: Field active							
H	0: In the start of active video (SAV) 1: In the end of active video (EAV)							
P3-P0	Protection Bits. Single channel: $P3=V^{\wedge}H$, $P2=F^{\wedge}H$, $P1=F^{\wedge}V$, $P0=F^{\wedge}V^{\wedge}H$ Multi-Channel: Channel ID							

During single-channel data transmission, P3-P0 is used for check. The transmission timing is shown in the following figure.

Figure 13-11 Single-channel Transmission Timing



In the programming, the final output format is successfully set in the RGB2YUV. Specifically, the 8bit data output by the RGB2YUV is shown in the above figure. The ctrl_data_out module performs pin mapping for the data.

13.3.4.4 i8080 Interface

The i8080 I/F LCD panel is most common interface for small size, low resolution LCD panels. The CPU control signals are active low.

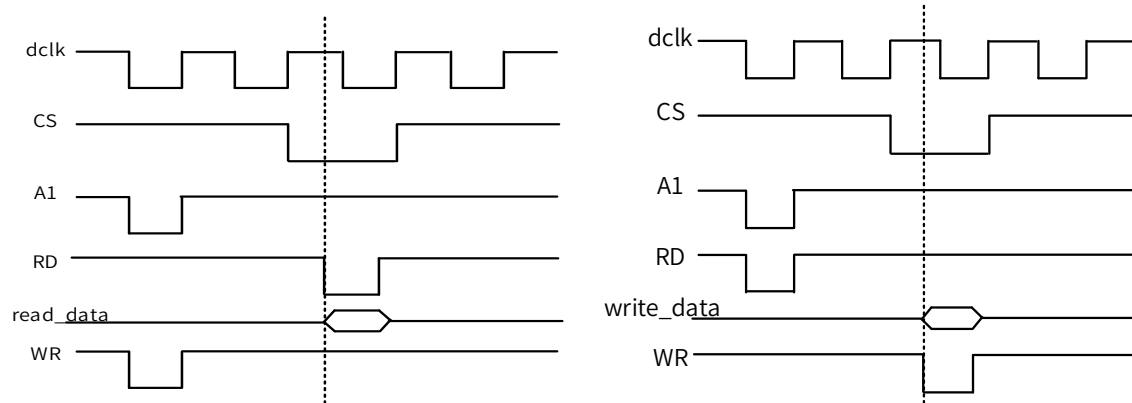
Table 13-10 CPU Panel Signals

Signal Name	Description	Type
CS	Chip select, active low	O
WR	Write strobe, active low	O
RD	Read strobe, active low	O
A1	Address bit, controlled by "LCD_CPUI/F" BIT25, ctrl_out[3]	O
D[23..0]	Digital RGB output signal	I/O

The following figure shows the relationship between basic timing and CPU timing. WR is fixed to 1 when reading. And it is the reverse phase of dclk1 when writing, in which it is low active and only has 1 period of low level. CS is active when pixel data is valid. RD is always set to 1 when

writing. And it is the reverse phase of dclk1 when writing, in which it is low active and only has 1 period of low level. 1 is set by “LCD_CPUI/F”. Set to 0 or 1 when reading, and set to 1 when reading.

Figure 13-12 CPU interface timing (left read, right write)



When CPU I/F is in the idle state, it can generate WR/RD timing by setting “LCD_CPUI/F”. The CS strobe is one DCLK width, and the WR/RD strobe is half DCLK width.

13.3.4.5 CEU Module

This module enhances color data from DE.

$$R' = (((Rr \cdot R + Rg \cdot G + Rb \cdot B + 8) / 16) + Rc + 8) / 16$$

$$G' = (((Gr \cdot R + Gg \cdot G + Gb \cdot B + 8) / 16) + Gc + 8) / 16$$

$$B' = (((Br \cdot R + Bg \cdot G + Bb \cdot B + 8) / 16) + Bc + 8) / 16$$

$$Rr, Rg, Rb, Gr, Gg, Gb, Br, Bg, Bb \quad s13 \quad (-16, 16)$$

$$Rc, Gc, Bc \quad s19 \quad (-16384, 16384)$$

$$R, G, B \quad u8 \quad [0-255]$$

R' has the range of [Rmin, Rmax]

G' has the range of [Rmin, Rmax]

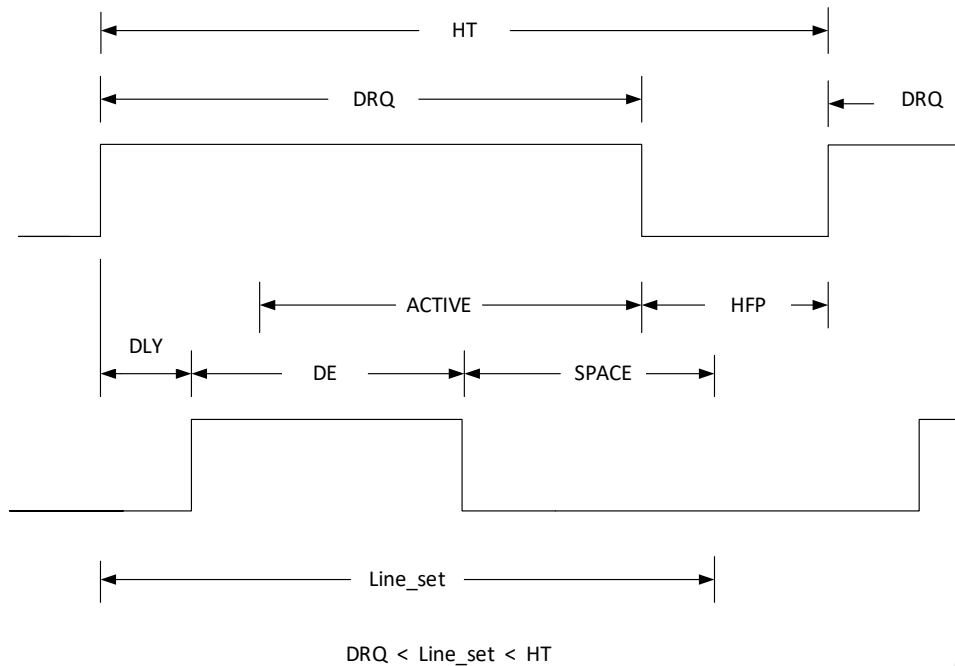
B' has the range of [Rmin, Rmax]

13.3.4.6 MIPI DSI

The requirements on MIPI DSI mode are as follows.

- (1). When using MIPI DSI as display interface, the data clk of TCON needs be started firstly.
- (2). When it is used with DSI video mode, the setting of block space needs to meet the following relationship.

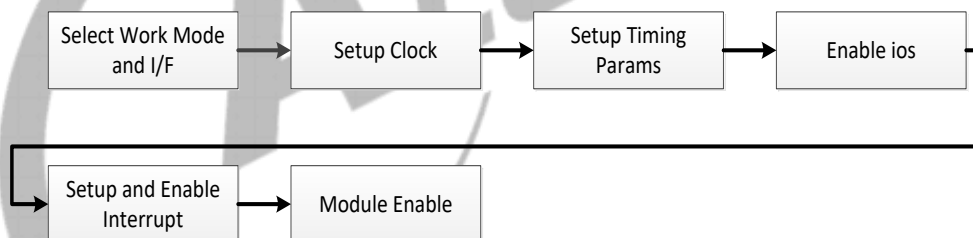
Figure 13-13 The Data Timing of MIPI DSI Video Mode



13.3.5 Programming Guidelines

13.3.5.1 HV Mode Configuration

Figure 13-14 HV Mode Initial Process



Step 1 Select HV interface type

Step 2 Set the clock, if you need to use the phase adjustment function, please set bit [31:28] of LCD_DCLK_REG to 0xf.

Step 3 Set the timing parameters x, ht, hbp, hspw, y, vt, vbp, vspw. Note that hbp contains hspw and vbp contains vspw. vt needs to be set to 2 times the actual value.

Step 4 Enable IO output.

Step 5 Set and enable the interrupt function. Note that when using line interrupt, you need to set LCD_LINE_INT_NUM in [LCD_GINT1_REG](#) first, and then enable LCD_LINE_INT_EN in [LCD_GINT0_REG](#) for line interrupt.

Step 6 Module enable

Table 13-11 HV Mode Configuration Example

Step	Register	Typical Value of Simulation	Description
1	LCD_HV_IF_REG (0x0058)	32'h0000_0000	Select parallel RGB
	LCD_CTL_REG (0x0040)	32'h0000_0041	Select HV (Sync+DE) mode
	/	/	Enable some functions such as CEU enable.
2	LCD_DCLK_REG (0x0044)	32'hf000_0002	/
3	LCD_BASIC0_REG (0x0048)	32'h0063_000f	Set x and y.
	LCD_BASIC1_REG (0x004C)	32'h0095_0013	Set ht and hbp.
	LCD_BASIC2_REG (0x0050)	32'h0030_0001	Set vt and vbp.
	LCD_BASIC3_REG (0x0054)	32'h0009_0000	Set HSPW and VSPW to get the configured value. The actual value is equal to the configured value plus one.
4	LCD_IO_POL_REG (0x0088)	32'h0000_0000	/
	LCD_IO_TRI_REG (0x008C)	32'he000_0000	1: Disable 0: Enable
5	LCD_GINT1_REG (0x0008)	32'h0010_0000	/
	LCD_GINT0_REG (0x0004)	32'h2000_0000	/
6	LCD_CTL_REG (0x0040)	32'h8000_0041	Open module enable.
	LCD_GCTL_REG (0x0000)	32'h8000_0000	Enable vs and hs to count.

13.3.5.2 BT.656 Mode Configuration

The BT.656 mode configuration process is similar to the HV mode. The following table is an example of typical parameter configuration.

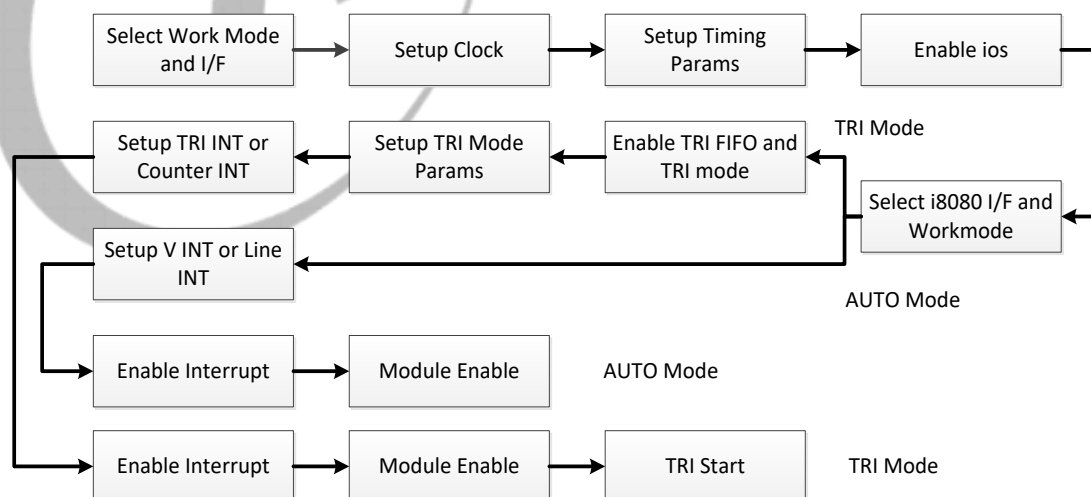
Table 13-12 BT.656 Mode Configuration Example

Step	Register	Typical Value of Simulation	Description
1	LCD_HV_IF_REG (0x0058)	32'hC080_0000	Select parallel RGB
	LCD_CTL_REG (0x0040)	32'h0000_0021	Select data source
	/		Enable some functions such as 3DFIFO and CEU.

Step	Register	Typical Value of Simulation	Description
2	LCD_DCLK_REG (0x0044)	32'hf000_0004	/
3	LCD_BASIC0_REG (0x0048)	32'h0117_0013	Set x and y.
	LCD_BASIC1_REG (0x004C)	32'h0513_0077	Set ht and hbp.
	LCD_BASIC2_REG (0x0050)	32'h0064_0018	Set vt and vbp.
	LCD_BASIC3_REG (0x0054)	32'h004f_0004	Set HSPW and VSPW
4	LCD_IO_POL_REG (0x0088)	32'h0400_0000	/
	LCD_IO_TRI_REG (0x008C)	32'he000_0000	1: Disable 0: Enable
5	LCD_GINT1_REG (0x0008)	32'h0000_0000	/
	LCD_GINT0_REG (0x0004)	32'h8300_0003	Bit15 asserted
6	LCD_CTL_REG (0x0040)	32'h8000_0041	Open module enable.
	LCD_GCTL_REG (0x0000)	32'h8000_0000	Enable vs and hs to count.

13.3.5.3 i8080 Mode Configuration

Figure 13-15 i8080 Mode Initial Process



Step 1 Select i8080 interface type.

Step 2 The step is the same as HV mode, but pulse adjustment function is invalid.

Step 3 The step is the same as HV mode. When using TRI mode, it is best to configure LCD timing parameters in HV mode, or a handful of functions such as CMAP will not be able to apply.

Step 4 The step is the same as HV mode.

Step 5 Select type and operating mode of i8080, the operating mode includes TRI mode and AUTO mode, and the two operating modes are different.

-----If for TRI mode-----

Step 1 Open TRI FIFO switch, and TRI mode function.

Step 2 Set parameters of TRI mode, including block size, block space and block number.

NOTE

- When output interface is parallel mode, then the setting value of block space parameter is not less than 20.
- When output interface is 2 cycle serial mode, then the setting value of block space parameter is not less than 40.
- When output interface is 3 cycle serial mode, then the setting value of block space parameter is not less than 60.
- When output interface is 4 cycle serial mode, then the setting value of block space parameter is not less than 80.

Step 3 Set the tri interrupt or counter interrupt. When using the two interrupts, mainly in the interrupt service function the tri start operation need be operated (the bit1 of LCD_CPU_IF_REG is set to "1"). If using TE trigger interrupt, you select the external input pin as a trigger signal, the 24-bit for offset 0x8C register is set to "1", to open up input of pad.

Step 4 Open the total switch of interrupt.

Step 5 Open the total enable of interrupt.

Step 6 Operate "tri start" operation (the bit1 of [LCD_CPU_IF_REG](#) is set to "1").

-----If for Auto mode-----

Step 1 Set and open V interrupt or Line interrupt, the step is the same as HV mode.

Step 2 Open module total enable.

The following table is an example of typical parameter configuration.

Table 13-13 i8080 Mode Configuration Example

Step	Register	Typical Value of Simulation	Description
1	LCD_HV_IF_REG (0x0058)	32'h0000_0000	Select parallel RGB
	LCD_CTL_REG (0x0040)	32'h8100_0001	Select data source.
	/	/	Enable some functions such as 3DFIFO and CEU enable.
2	LCD_DCLK_REG (0x0044)	32'hf000_0002	/
3	LCD_BASIC0_REG (0x0048)	32'h0063_000f	Set x and y.
	LCD_BASIC1_REG (0x004C)	32'h0095_0013	Set ht and hbp.
	LCD_BASIC2_REG (0x0050)	32'h0030_0001	Set vt and vbp.
	LCD_BASIC3_REG (0x0054)	32'h0009_0000	Set HSPW and VSPW.
4	LCD_IO_POL_REG (0x0088)	32'h0000_0000	/
	LCD_IO_TRI_REG (0x008C)	32'he000_0000	1: Disable 0: Enable
5	LCD_CPU_IF_REG (0x0060)	32'h0401_0005	Select the type and working mode of the i8080 interface.
TRI mode			
6	LCD_CPU_IF_REG (0x0060)	32'h0401_0005	Enable the trigger FIFO and trigger mode.
7	LCD_CPU_TRI0_REG (0x0160)	32'h00ff_031f	Configure parameters in TRI mode.
	LCD_CPU_TRI1_REG (0x0164)	32'h0000_01df	
8	LCD_GINT0_REG (0x0004)	Write 1 to bit[26] or bit[27]	Select trig interrupt or counter interrupt.
9	LCD_CTL_REG (0x0040)	32'h8000_0000	Open module enable.
	LCD_GCTL_REG (0x0000)	32'h8000_0000	
10	LCD_CPU_IF_REG (0x0060)	32'h0401_0007	Tri start.
AUTO mode			
6	LCD_GINT1_REG (0x0008)	32'h0010_0000	/
	LCD_GINT0_REG (0x0004)	32'h2000_0000	/

Step	Register	Typical Value of Simulation	Description
7	LCD_CTL_REG (0x0040)	32'h8100_0040	Open module enable.
	LCD_GCTL_REG (0x0000)	32'h8000_0000	Enable vs and hs to start counting

13.3.6 Register List

Module Name	Base Address
TCON_LCD0	0x05461000

Register Name	Offset	Description
LCD_GCTL_REG	0x0000	LCD Global Control Register
LCD_GINT0_REG	0x0004	LCD Global Interrupt Register0
LCD_GINT1_REG	0x0008	LCD Global Interrupt Register1
LCD_CTL_REG	0x0040	LCD Control Register
LCD_DCLK_REG	0x0044	LCD Data Clock Register
LCD_BASIC0_REG	0x0048	LCD Basic Timing Register0
LCD_BASIC1_REG	0x004C	LCD Basic Timing Register1
LCD_BASIC2_REG	0x0050	LCD Basic Timing Register2
LCD_BASIC3_REG	0x0054	LCD Basic Timing Register3
LCD_HV_IF_REG	0x0058	LCD HV Panel Interface Register
LCD_CPU_IF_REG	0x0060	LCD CPU Panel Interface Register
LCD_CPU_WR_REG	0x0064	LCD CPU Panel Write Data Register
LCD_CPU_RD0_REG	0x0068	LCD CPU Panel Read Data Register0
LCD_CPU_RD1_REG	0x006C	LCD CPU Panel Read Data Register1
LCD_IO_POL_REG	0x0088	LCD IO Polarity Register
LCD_IO_TRI_REG	0x008C	LCD IO Control Register
LCD_IO_ADJ_REG	0x0090	LCD IO Adjust Register
LCD_DEBUG_REG	0x00FC	LCD Debug Register
LCD_CEU_CTL_REG	0x0100	LCD CEU Control Register
LCD_CEU_COEF_MUL_REG0	0x0110+N*0x04 (N=0-2)	LCD CEU Coefficient Register0
LCD_CEU_COEF_MUL_REG1	0x0120+N*0x04 (N=0-2)	LCD CEU Coefficient Register1
LCD_CEU_COEF_MUL_REG2	0x0130+N*0x04 (N=0-2)	LCD CEU Coefficient Register2
LCD_CEU_COEF_ADD_REG	0x011C+N*0x10 (N=0-2)	LCD CEU Coefficient Add Register
LCD_CEU_COEF_RANG_REG	0x0140+N*0x04 (N=0-2)	LCD CEU Coefficient Rang Register
LCD_CPU_TRI0_REG	0x0160	LCD CPU Panel Trigger Register0
LCD_CPU_TRI1_REG	0x0164	LCD CPU Panel Trigger Register1
LCD_CPU_TRI2_REG	0x0168	LCD CPU Panel Trigger Register2
LCD_CPU_TRI3_REG	0x016C	LCD CPU Panel Trigger Register3
LCD_CPU_TRI4_REG	0x0170	LCD CPU Panel Trigger Register4
LCD_CPU_TRI5_REG	0x0174	LCD CPU Panel Trigger Register5
LCD_SAFE_PERIOD_REG	0x01F0	LCD Safe Period Register

Register Name	Offset	Description
FSYNC_GEN_CTRL_REG	0x0228	LCD FSYNC Generate Control Register
FSYNC_GEN_DLY_REG	0x022C	LCD FSYNC Generate Delay Register
LCD_SYNC_CTL_REG	0x0230	LCD Sync Control Register

13.3.7 Register Description

13.3.7.1 0x0000 LCD Global Control Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: LCD_GCTL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	LCD_EN When it's disabled, the module will be reset to idle state (except configuration logic). 0: Disable 1: Enable
30:0	/	/	/

13.3.7.2 0x0004 LCD Global Interrupt Register0 (Default Value: 0x0000_0000)

Offset: 0x0004			Register Name: LCD_GINT0_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	LCD_VB_INT_EN Enable the Vb interrupt. 0: Disable 1: Enable
30	/	/	/
29	R/W	0x0	LCD_LINE_INT_EN Enable the line interrupt. 0: Disable 1: Enable
28	/	/	/
27	R/W	0x0	LCD_TRI_FINISH_INT_EN Enable the trigger finish interrupt (TRI mode). 0: Disable 1: Enable

Offset: 0x0004			Register Name: LCD_GINT0_REG
Bit	Read/Write	Default/Hex	Description
26	R/W	0x0	LCD_TRI_COUNTER_INT_EN Enable the trigger counter interrupt 0: Disable 1: Enable Before enabling, configure the TRI_INT_MODE bit of the LCD_CPU_TRI3_REG (offset: 0x016C) register first.
25:16	/	/	/
15	R/WOC	0x0	LCD_VB_INT_FLAG Asserted during vertical no-display period every frame. Write 0 to clear it.
14	/	/	/
13	R/WOC	0x0	LCD_LINE_INT_FLAG Trigger when SY0 match the current LCD scan line. Write 0 to clear it.
12	/	/	/
11	R/WOC	0x0	LCD_TRI_FINISH_INT_FLAG Trigger when CPU trigger mode finish. Write 0 to clear it.
10	R/WOC	0x0	LCD_TRI_COUNTER_INT_FLAG Trigger when tri counter reaches this value Write 0 to clear it. 00: Disable 01: Interrupt trigger flag in counter mode 10: Flag for Interrupt trigger in logic high level of GPIO TRIGGLE signal 11: Flag for Interrupt trigger in logic low level of GPIO TRIGGLE signal
9	R/WOC	0x0	LCD_TRI_UNDERFLOW_FLAG Only used in DSI video mode, tri when SYNC by DSI but not finish Write 0 to clear it.
8:3	/	/	/
2	R/W	0x0	FSYNC_INT_INV Enable the FSYNC interrupt set signal inverse polarity. When FSYNC is positive, this bit must be 1. And vice versa.

Offset: 0x0004			Register Name: LCD_GINT0_REG
Bit	Read/Write	Default/Hex	Description
1	R/W	0x0	DE_INT_FLAG Asserted at the first valid line in every frame. Write 0 to clear it.
0	R/W	0x0	FSYNC_INT_FLAG Asserted at the FSYNC signal in every frame. Write 0 to clear it.

13.3.7.3 0x0008 LCD Global Interrupt Register1 (Default Value: 0x0000_0000)

Offset: 0x0008			Register Name: LCD_GINT1_REG
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0x0	LCD_LINE_INT_NUM Scan line for LCD line trigger (including inactive lines). Setting it for the specified line for trigger0. Configure this bit before writing 1 to the LCD_LINE_INT_EN in the LCD_GINT0_REG register Note: SY0 is writable only when LINE_TRG0 disable.
15:0	/	/	/

13.3.7.4 0x0040 LCD Control Register (Default Value: 0x0000_0000)

Offset: 0x0040			Register Name: LCD_CTL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	LCD_EN It executes at the beginning of the first blank line of LCD timing. 0: Disable 1: Enable
30:26	/	/	/
25:24	R/W	0x0	LCD_IF Set the interface type of LCD controller. 00: HV(Sync+DE) 01: 8080 I/F 1x: Reserved
23:22	/	/	/

Offset: 0x0040			Register Name: LCD_CTL_REG
Bit	Read/Write	Default/Hex	Description
21	R/WAC	0x0	LCD_FIFO1_RST LCD FIFO1 Reset Signal (High Active) Write 1 and then 0 at this bit will reset FIFO 1 Note: 1 holding time must more than 1 DCLK
20:19	/	/	/
18:16	R/W	0x0	RGB_ORDER_SWAP If the input is DATA[23:0]={R, G, B}, adsjut the sequence of R, G, and B. The adjusted data is the input of lcd_fifo. 000: RGB 001: RBG 010: GRB 011: GBR 100: BRG 101: BGR Others: Reserved
15:9	/	/	/
8:4	R/W	0x0	LCD_START_DLY LCD_START_DLY determines the period of de_sync staying high. When the active zone finishes, the de_sync signal is pulled up. During this period, counting starts. When counting to LCD_STRT_DLY, the de_sync signal is pulled down. The unit of delay is T_{line} . The bit is valid only when LCD_EN == 1 (This LCD_EN is for LCD timing enable) Note: de_sync signal is the field synchronization signal sent from TCON to DE.
3	/	/	/
2:0	R/W	0x0	LCD_SRC_SEL LCD Source Select 000: DE 001: Color Check 010: Grayscale Check 011: Black by White Check 100: Test Data all 0 101: Test Data all 1 110: Reversed 111: Gridding Check

13.3.7.5 0x0044 LCD Data Clock Register (Default Value: 0x0000_0000)

Offset: 0x0044			Register Name: LCD_DCLK_REG
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0x0	LCD_DCLK_EN LCD clock enable. 0000: dclk_en = 0; dclk1_en = 0; dclk2_en = 0; 0001/0010: dclk_en = 1; dclk1_en = 0; dclk2_en = 0; 0011: dclk_en = 1; dclk1_en = 1; dclk2_en = 0; 0101: dclk_en = 1; dclk1_en = 0; dclk2_en = 1; 1111: dclk_en = 1; dclk1_en = 1; dclk2_en = 1; Others: Reversed dclk_en is used to configure dclk (dclk is the pixel clock with phase 0). dclk1_en is used to configure dclk1 (dclk1 is pixel clock with phase 1/3). dclk2_en is used to configure dclk2 (dclk2 is pixel clock with phase 2/3).
27:7	/	/	/
6:0	R/W	0x0	LCD_DCLK_DIV LCD DCLK Diving Value ranging from 1 to 32 $fdclk = fcd\_clk / LCD\_DCLK\_DIV$ If dclk1 and dclk2 are used, LCD_DCLK_DIV configured value should be no less than 6.

13.3.7.6 0x0048 LCD Basic Timing Register0 (Default Value: 0x0000_0000)

Offset: 0x0048			Register Name: LCD_BASIC0_REG
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0x0	WIDTH_X Panel width is X+1
15:12	/	/	/
11:0	R/W	0x0	HEIGHT_Y Panel height is Y+1

13.3.7.7 0x004C LCD Basic Timing Register1 (Default Value: 0x0000_0000)

Offset: 0x004C			Register Name: LCD_BASIC1_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/

Offset: 0x004C			Register Name: LCD_BASIC1_REG
Bit	Read/Write	Default/Hex	Description
28:16	R/W	0x0	HT $T_{hcycle} = (HT+1) * T_{dclk}$ Computation: 1) parallel: $HT = X + BLANK$ Limitation: 1) parallel: $HT \geq (HBP + 1) + (X+1) + 2$ 2) serial 1: $HT \geq (HBP + 1) + (X+1) * 3 + 2$ 3) serial 2: $HT \geq (HBP + 1) + (X+1) * 3/2 + 2$
15:12	/	/	/
11:0	R/W	0x0	HBP Horizontal back porch (in dclk) $T_{hbp} = (HBP + 1) * T_{dclk}$

13.3.7.8 0x0050 LCD Basic Timing Register2 (Default Value: 0x0000_0000)

Offset: 0x0050			Register Name: LCD_BASIC2_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	0x0	VT $T_{VT} = (VT)/2 * T_{hsync}$ $VT/2 \geq (VBP+1) + (Y+1) + 2$ VT [0] is used to identify the used scanning mode. If VT [0] ==0, the current transmission uses progressive scanning. if VT[0]==1, the current transmission uses interlaced scanning.
15:12	/	/	/
11:0	R/W	0x0	VBP $T_{vbp} = (VBP + 1) * T_{hsync}$

13.3.7.9 0x0054 LCD Basic Timing Register3 (Default Value: 0x0000_0000)

Offset: 0x0054			Register Name: LCD_BASIC3_REG
Bit	Read/Write	Default/Hex	Description
31:26	/	/	/
25:16	R/W	0x0	HSPW $T_{hspw} = (HSPW+1) * T_{dclk}$ Note: $HT > (HSPW+1)$

Offset: 0x0054			Register Name: LCD_BASIC3_REG
Bit	Read/Write	Default/Hex	Description
15:10	/	/	/
9:0	R/W	0x0	VSPW $Tv_{spw} = (VSPW+1) * Th_{sync}$ Note: $VT/2 > (VSPW+1)$

13.3.7.10 0x0058 LCD HV Panel Interface Register (Default Value: 0x0000_0000)

Offset: 0x0058			Register Name: LCD_HV_IF_REG
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0x0	HV_MODE Set the HV mode of LCD controller. 0000: 24bit/1cycle parallel mode 1000: 8bit/3cycle RGB serial mode (RGB888) 1010: 8bit/4cycle Dummy RGB (DRGB) 1011: 8bit/4cycle RGB Dummy (RGBD) 1100: 8bit/2cycle YUV serial mode (BT656) Others: Reversed
27:26	R/W	0x0	RGB888_ODD_ORDER Serial RGB888 mode Output sequence at odd lines of the panel (line 1, 3, 5, 7...). 00: R→G→B 01: B→R→G 10: G→B→R 11: R→G→B
25:24	R/W	0x0	RGB888_EVEN_ORDER Serial RGB888 mode Output sequence at even lines of the panel (line 2, 4, 6, 8...). 00: R→G→B 01: B→R→G 10: G→B→R 11: R→G→B
23:22	R/W	0x0	YUV_SM Serial YUV mode Output sequence 2-pixel-pair of every scan line. 00: UYVY 01: VYUY 10: YVYU 11: YUYV

Offset: 0x0058			Register Name: LCD_HV_IF_REG
Bit	Read/Write	Default/Hex	Description
21:20	R/W	0x0	YUV_EAV_SAV_F_LINE_DLY When the BT656_F_MASK bit of LCD_CEU_CTL_REG (offset: 0x0100) register is configured as 0, the F signal timing in BT.656 transmission is configured in this bit, and the value could be configured as maximum 3 lines. (The F signal is the odd and even field flag in the current transmission) 00: When a field counting ends, the F signal flips immediately. 01: When both the field counting value and the line counting value are 2, the F signal flips. 10: When both the field counting value and the line counting value are 3, the F signal flips. 11: Reserved.
19	R/W	0x0	CCIR_CSC_DIS Select '0' LCD convert source from RGB to YUV. 0: Enable 1: Disable Only valid when HV mode is "1100".
18:0	/	/	/

13.3.7.11 0x0060 LCD CPU Panel Interface Register (Default Value: 0x0000_0000)

Offset: 0x0060			Register Name: LCD_CPU_IF_REG
Bit	Read/Write	Default/Hex	Description

Offset: 0x0060			Register Name: LCD_CPU_IF_REG
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0x0	CPU_MODE Set the CPU interface work mode. 0000: 18bit/256K mode (output 1 pixel in 1 cycle) 0010: 18bit/256K mode (1st, 2nd, 3rd) 0100: 18bit/256K mode (1st and 2nd use D[12:5]) 0110: 18bit/256K mode (1st and 2nd use D[23:18]) 1000: 16bit 64K mode (output 1 pixel in 1 cycle) 1010: 18bit 256K mode (output 1 pixel in 2 cycles) 1100: 18bit 256K mode (output 1 pixel in 3 cycles) 1110: 16bit 64K mode (output 1 pixel in 2 cycles) xxx1: 24bit for DSI
27	/	/	/
26	R/W	0x0	DA CPU Mode AUTO Transmission In the active region, the IO [3] port outputs the DA value. in the non-active region, if the WR/RD operation is not executed, the IO [3] port outputs the DA value. FLUSH/TRI Transmission When LCD_FIFO is not empty, the IO [3] port outputs the DA value. when LCD_FIFO is empty, if the WR/RD operation is not executed, the IO[3] outputs the DA value.
25	R/W	0x0	CA CPU Mode AUTO Transmission In the non-active region, if WR/RD operation is executed, the IO [3] port outputs the CA value; FLUSH/TRI Transmission When LCD_FIFO is empty and WR/RD operation is executed, the IO[3] port outputs the CA value.
24	/	/	/

Offset: 0x0060			Register Name: LCD_CPU_IF_REG
Bit	Read/Write	Default/Hex	Description
23	R	0x0	WR_FLAG The status of write operation. 0: Write operation is finishing 1: Write operation is pending
22	R	0x0	RD_FLAG The status of read operation. 0: Read operation is finishing 1: Read operation is pending
21:18	/	/	/
17	R/W	0x0	AUTO Auto Transfer Mode: Auto transmission: If configured as 1, the current frame data is transmitted in the active region, and the transmission is controlled by HV timing (only valid in 8080 mode). This bit is sampled at the end of the active region.
16	R/W	0x0	FLUSH Direct transfer mode: Flush transmission: When AUTO is 0 and FLUSH is configured as 1, if LCD_FIFO is not empty, pixel data can be output to the subsequent module without being controlled by HV timing (only valid in 8080 mode). Data output rate control by DCLK.
15:3	/	/	/
2	R/W	0x0	TRI_FIFO_EN Enable the trigger FIFO. 0: Disable 1: Enable
1	R/W1S	0x0	TRI_START Software must make sure write '1' only when this flag is '0'. Write '1' to start a frame flush, write '0' has no effect. This flag indicated frame flush is running.
0	R/W	0x0	TRI_EN Enable trigger mode. 0: Trigger mode disable 1: Trigger mode enable

13.3.7.12 0x0064 LCD CPU Panel Write Data Register (Default Value: 0x0000_0000)

Offset: 0x0064			Register Name: LCD_CPU_WR_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:0	W	0x0	DATA_WR Under Auto transmission, when in the VBlank region and WR is valid, the DATA_WR value is output through D [23:0]; Under Flush/TRIG transmission, when lcd_fifo is empty and WR is valid, the DATA_WR value is output through D[23:0].

13.3.7.13 0x0068 LCD CPU Panel Read Data Register0 (Default Value: 0x0000_0000)

Offset: 0x0068			Register Name: LCD_CPU_RD0_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:0	R	0x0	DATA_RD0 Data read on 8080 bus. Launch a new read operation on 8080 bus.

13.3.7.14 0x006C LCD CPU Panel Read Data Register1 (Default Value: 0x0000_0000)

Offset: 0x006C			Register Name: LCD_CPU_RD1_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:0	R	0x0	DATA_RD1 Data read on 8080 bus, without a new read operation on 8080 bus.

13.3.7.15 0x0088 LCD IO Polarity Register (Default Value: 0x0000_0000)

Offset: 0x0088			Register Name: LCD_IO_POL_REG
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/

Offset: 0x0088			Register Name: LCD_IO_POL_REG
Bit	Read/Write	Default/Hex	Description
29:28	R/W	0x0	DCLK_SEL Set the phase offset of clock and data in HV mode. 00: Use DCLK0 (normal phase offset) 01: Use DCLK1 (1/3 phase offset) 10: Use DCLK2 (2/3 phase offset) Others: Reserved
27	R/W	0x0	IO3_INV Enable invert function of IO3. 0: Not invert 1: Invert
26	R/W	0x0	IO2_INV Enable invert function of IO2. 0: Not invert 1: Invert
25	R/W	0x0	IO1_INV Enable invert function of IO1. 0: Not invert 1: Invert
24	R/W	0x0	IO0_INV Enable invert function of IO0. 0: Not invert 1: Invert
23:0	R/W	0x0	Data_INV LCD output port D [23:0] polarity control, with independent bit control. 0: Normal polarity 1: Invert the specify output

13.3.7.16 0x008C LCD IO Control Register (Default Value: 0x0FFF_FFFF)

Offset: 0x008C			Register Name: LCD_IO_TRI_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28	R/W	0x0	RGB_ENDIAN Set the endian of data bits. 0: Normal 1: Bits_invert (The RGB byte order remains unchanged, and the bits within the bytes are inverted.)

Offset: 0x008C			Register Name: LCD_IO_TRI_REG
Bit	Read/Write	Default/Hex	Description
27	R/W	0x1	IO3_OUTPUT_TRI_EN Enable the output of IO3. 1: Disable 0: Enable
26	R/W	0x1	IO2_OUTPUT_TRI_EN Enable the output of IO2. 1: Disable 0: Enable
25	R/W	0x1	IO1_OUTPUR_TRI_EN Enable the output of IO1. 1: Disable 0: Enable
24	R/W	0x1	IO0_OUTPUT_TRI_EN Enable the output of IO0. 1: Disable 0: Enable
23:0	R/W	0xFFFFFFFF	DATA_OUTPUT_TRI_EN LCD output port D [23:0] output enable, with independent bit control. 1: Disable 0: Enable

13.3.7.17 0x0090 LCD IO Adjust Register (Default Value: 0x0000_0000)

Offset: 0x0090			Register Name: LCD_IO_ADJ_REG
Bit	Read/Write	Default/Hex	Description
31:10	/	/	/
9:8	R/W	0x0	HIGH_DATA_ADJ DATA[23:16] reverse valid bit 00: No adjustment 01: 5-bit reverse order 10: 6-bit reverse order 11: 8-bit reverse order
7:6	/	/	/
5:4	R/W	0x0	MID_DATA_ADJ DATA[15:8] reverse valid bit 00: No adjustment 01: 5-bit reverse order 10: 6-bit reverse order 11: 8-bit reverse order
3:2	/	/	/

Offset: 0x0090			Register Name: LCD_IO_ADJ_REG
Bit	Read/Write	Default/Hex	Description
1:0	R/W	0x0	LOW_DATA_ADJ DATA[7:0] reverse valid bit 00: No adjustment 01: 5-bit reverse order 10: 6-bit reverse order 11: 8-bit reverse order For example, In the case of D7-D0 6bit reverse order, the adjusted order is {DATA[2:7], DATA[1:0]}

13.3.7.18 0x00FC LCD Debug Register (Default Value: 0x2000_0000)

Offset: 0x00FC			Register Name: LCD_DEBUG_REG
Bit	Read/Write	Default/Hex	Description
31	R/W0C	0x0	LCD_FIFO_UNDERFLOW The flag shows whether the FIFOs in underflow status. 0: Not underflow 1: Underflow
30	/	/	/
29	R	0x1	LCD_FIELD_POL LCD field indication signal, initial value is 1. The flag indicates the current field polarity. 0: Second field 1: First field
28	/	/	/
27:16	R	0x0	LCD_CURRENT_LINE Scan lines of the current field (line count, starting from the vfp region with an initial value of 1).
15	R/W0C	0x0	WR Consistent with 0x060 WR_FLAG register, write screen register flag bit.
14	R/W0C	0x0	RD Consistent with 0x060 RD_FLAG register, read screen register flag bit.
13:0	/	/	/

13.3.7.19 0x0100 LCD CEU Control Register (Default Value: 0x0000_0000)

Offset: 0x0100	Register Name: LCD_CEU_CTL_REG

Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	CEU_EN Enable CEU function. 0: Bypass 1: Enable
30	R/W	0x0	BT656_F_MASK The F signal generation mode selection enable during BT656 transmission. BT656 F Mask 0: Disable (F signal timing is configured in the YUV_EAV_SAV_F_LINE_DLY bit of LCD_HV_IF_REG register) 1: Enable (F signal timing is configured in the BT656_F_MASK_VALUE bit of LCD_CEU_CTL_REG register)
29	R/W	0x0	BT656_F_MASK_VALUE BT656 F Mask Value 0/1
28:0	/	/	/

13.3.7.20 0x0110+N*0x04(N=0-2) LCD CEU Coefficient Register0 (Default Value: 0x0000_0000)

Offset: 0x0110+N*0x04(N=0-2)			Register Name: LCD_CEU_COEF_MUL_REG0
Bit	Read/Write	Default/Hex	Description
31:13	/	/	/
12:0	R/W	0x0	CEU_COEF_MUL_VALUE Signed 13bit value, range of (-16,16). N=0: Rr N=1: Rg N=2: Rb

13.3.7.21 0x0120+N*0x04(N=0-2) LCD CEU Coefficient Register1 (Default Value: 0x0000_0000)

Offset: 0x0120+N*0x04(N=0-2)			Register Name: LCD_CEU_COEF_MUL_REG1
Bit	Read/Write	Default/Hex	Description
31:13	/	/	/
12:0	R/W	0x0	CEU_COEF_MUL_VALUE Signed 13bit value, range of (-16,16). N=0: Gr N=1: Gg N=2: Gb

13.3.7.22 0x0130+N*0x04(N=0-2) LCD CEU Coefficient Register2 (Default Value: 0x0000_0000)

Offset: 0x0130+N*0x04(N=0-2)			Register Name: LCD_CEU_COEF_MUL_REG2
Bit	Read/Write	Default/Hex	Description
31:13	/	/	/
12:0	R/W	0x0	CEU_COEF_MUL_VALUE Signed 13bit value, range of (-16,16). N=0: Br N=1: Bg N=2: Bb

13.3.7.23 0x011C+N*0x10(N=0-2) LCD CEU Add Register (Default Value: 0x0000_0000)

Offset: 0x011C+N*0x10(N=0-2)			Register Name: LCD_CEU_COEF_ADD_REG
Bit	Read/Write	Default/Hex	Description
31:19	/	/	/
18:0	R/W	0x0	CEU_COEF_ADD_VALUE Signed 19bit value, range of (-16384, 16384). N=0: Rc N=1: Gc N=2: Bc

13.3.7.24 0x0140+N*0x04(N=0-2) LCD CEU Coefficient Rang Register (Default Value: 0x0000_0000)

Offset: 0x0140+N*0x04(N=0-2)			Register Name: LCD_CEU_COEF_RANG_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:16	R/W	0x0	CEU_COEF_RANGE_MIN Unsigned 8bit value, range of [0,255].
15:8	/	/	/
7:0	R/W	0x0	CEU_COEF_RANGE_MAX Unsigned 8bit value, range of [0,255].

13.3.7.25 0x0160 LCD CPU Panel Trigger Register0 (Default Value: 0x0000_0000)

Offset: 0x0160			Register Name: LCD_CPU_TRI0_REG
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/

Offset: 0x0160			Register Name: LCD_CPU_TRI0_REG
Bit	Read/Write	Default/Hex	Description
27:16	R/W	0x0	BLOCK_SPACE Trigger mode: The delay after reading a line from TRIGFIFO. Counting starts when a line of transmission ends. When the count value is equal to BLOCK_SPACE, TRIGFIFO can initiate the next line of reading. The spaces between data blocks. It should be set >20*pixel.
15:12	/	/	/
11:0	R/W	0x0	BLOCK_SIZE Trigger mode: The configuration value is the size of the effective display region X. The size of data block. It is usually set as X.

13.3.7.26 0x0164 LCD CPU Panel Trigger Register1 (Default Value: 0x0000_0000)

Offset: 0x0164			Register Name: LCD_CPU_TRI1_REG
Bit	Read/Write	Default/Hex	Description
31:16	R	0x0	BLOCK_CURRENT_NUM Shows the current data block transmitting to panel.
15:0	R/W	0x0	BLOCK_NUM The number of data blocks. It is usually set as Y.

13.3.7.27 0x0168 LCD CPU Panel Trigger Register2 (Default Value: 0x0020_0000)

Offset: 0x0168			Register Name: LCD_CPU_TRI2_REG
Bit	Read/Write	Default/Hex	Description
31:16	R/W	0x20	START_DLY The period of de_sync staying high in the Trigger mode $T_{dly} = (Start\_Delay + 1) * dclk * 8.$
15	R/W	0x0	TRANS_START_MODE Select the FIFOs used in CPU mode. 0: 3D_FIFO+TRI_FIFO 1: TRI_FIFO

Offset: 0x0168			Register Name: LCD_CPU_TRI2_REG
Bit	Read/Write	Default/Hex	Description
14:13	R/W	0x0	SYNC_MODE Set the sync mode in CPU interface. 0x: Auto 10: 0 (de_sync = 0) 11: 1 (de_sunc = 1) Note: de_sync signal is the field synchronization signal sent from TCON to DE.
12:0	R/W	0x0	TRANS_START_SET Transmission Start Trigger Value. When the amount of data in the buffer (selected in the TRANS_START_MODE bit) is not less than the configured value, the pixel data in TRIGFIFO can be transferred to LCD_FIFO. Usual set as the length of a line.

13.3.7.28 0x016C LCD CPU Panel Trigger Register3 (Default Value: 0x0000_0000)

Offset: 0x016C			Register Name: LCD_CPU_TRI3_REG
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:28	R/W	0x0	TRI_INT_MODE Configure the CPU to trigger interrupt mode. When configured as 01, it is Counter mode. The counting range is configured by CONTER_N and COUNTER_M, and the interrupt period is (Count_N+1) *(Count_M+1) *4dclk. 00: Disable 01: Counter mode 10: GPIO TRIGGLE signal high level trigger mode 11: GPIO TRIGGLE signal low level trigger mode
27:24	/	/	/
23:8	R/W	0x0	COUNTER_N Count value in Counter mode, the counting period is COUNTER_N * (2*COUNTER_M*dclk).
7:0	R/W	0x0	COUNTER_M Count value in Counter mode, the counting period is COUNTER_M*dclk.

13.3.7.29 0x0170 LCD CPU Panel Trigger Register4 (Default Value: 0x0000_0000)

Offset: 0x0170	Register Name: LCD_CPU_TRI4_REG

Bit	Read/Write	Default/Hex	Description
31:25	/	/	/
24	R/W	0x0	A1_First_Valid Valid in first Block (non-active region)
23:0	R/W	0x0	D23_TO_D0_First_Valid Valid in first Block

13.3.7.30 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 (active region)
23:0	R/W	0x0	D23_TO_D0_NON_First_Valid Valid in Block except first.

13.3.7.31 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	SAFE_PERIOD_FIFO_NUM 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.
15:4	R/W	0x0	SAFE_PERIOD_LINE 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.
3	/	/	/
2:0	R/W	0x0	SAFE_PERIOD_MODE Select the save mode 000: Unsafe 001: Safe 010: Safe at FIFO_CURR_NUM > SAFE_PERIOD_FIFO_NUM 011: Safe at 2 and safe at sync active 100: Safe at line

13.3.7.32 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	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.
7	/	/	/
6	R/W	0x0	SENSOR_ACT1_VALUE Sensor Active1 Value 0: Fsync active_1 period output 0 1: Fsync active_1 period output 1
5	R/W	0x0	SENSOR_ACT0_VALUE Sensor Active0 Value 0: Fsync active_0 period output 0 1: Fsync active_0 period output 1
4	R/W	0x0	SENSOR_DIS_VALUE Sensor Disable Value 0: Fsync disable period output 0 1: Fsync disable period output 1
3	/	/	/
2	R/W	0x0	HSYNC_POL_SEL Hsync Polarity Select 0: Normal 1: Opposite hsync to hsync counter
1	R/W	0x0	SEL_VSYNC_EN Select Vsync Enable 0: Select vsync falling edge to start state machine 1: Select vsync rising edge to start state machine
0	R/W	0x0	FSYNC_GEN_EN Fsync Generate Enable 0: Disable 1: Enable

13.3.7.33 0x022C LCD FSYNC Generate Delay Register (Default Value: 0x0000_0000)

Offset: 0x022C			Register Name: FSYNC_GEN_DLY_REG
Bit	Read/Write	Default/Hex	Description

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.

13.3.7.34 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
7:0	/	/	/

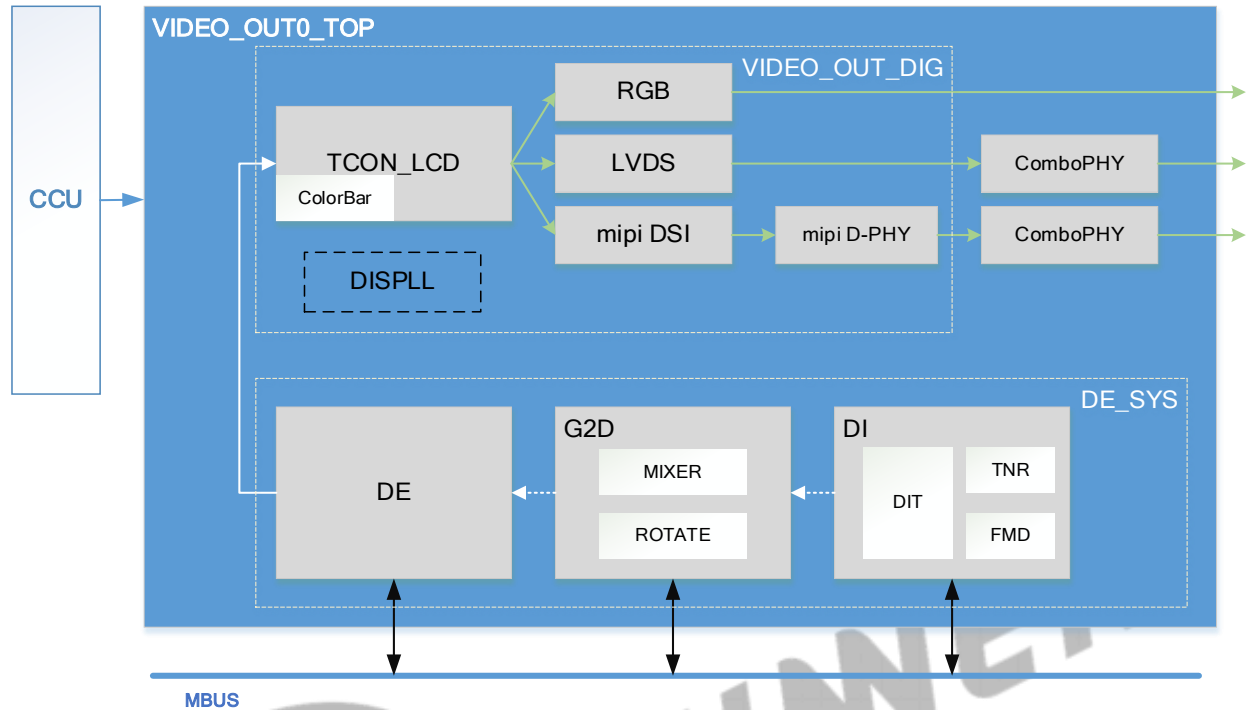
13.4 VIDEO_OUT SYS

13.4.1 Overview

The VIDEO_OUT SYS is responsible for display output timing control and data output.

13.4.2 Block Diagram

Figure 13-16 VIDEO_OUT SYS Block Diagram



13.4.3 Register List

Module Name	Base Address	Comments
VIDEO_OUT0_SYS	0x0546 0000	VIDEO_OUT0_SYS Register

Register Name	Offset	Description
LCD_CLK_SRC_COMB_CLK_SRC	0x000C	LCD Clock And Combophy Clock Source Select Register
VOO_MODULE_GATING	0x0020	TCON LCD Output To LVDS/DSI Gating Register

13.4.4 Register Description

13.4.4.1 0x000C LCD Clock And Combophy Clock Source Select Register (Default Value: 0x0000_0000)

Offset: 0x000C			Register Name: LCD_CLK_SRC_COMB_CLK_SRC
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/

Offset: 0x000C			Register Name: LCD_CLK_SRC_COMB_CLK_SRC
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	LCD0_CLK_SRC_SEL LCD0_CLK Source Select 0: CCU 1: Displl

13.4.4.2 0x0020 TCON LCD Output To LVDS/DSI Gating Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: VO0_MODULE_GATING
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	LCD0_DSI_GATE LCD0 DSI Gate 0: Gating 1: Release
15:0	/	/	/

14 Audio Interfaces

14.1 Audio Codec

14.1.1 Overview

The Audio Codec has the following features: The Audio Codec is high-performance audio encoder and decoder module which supports DAC/ADC, Dynamic Range Controller (DRC) and Digital Volume Controller (DVC) functions.

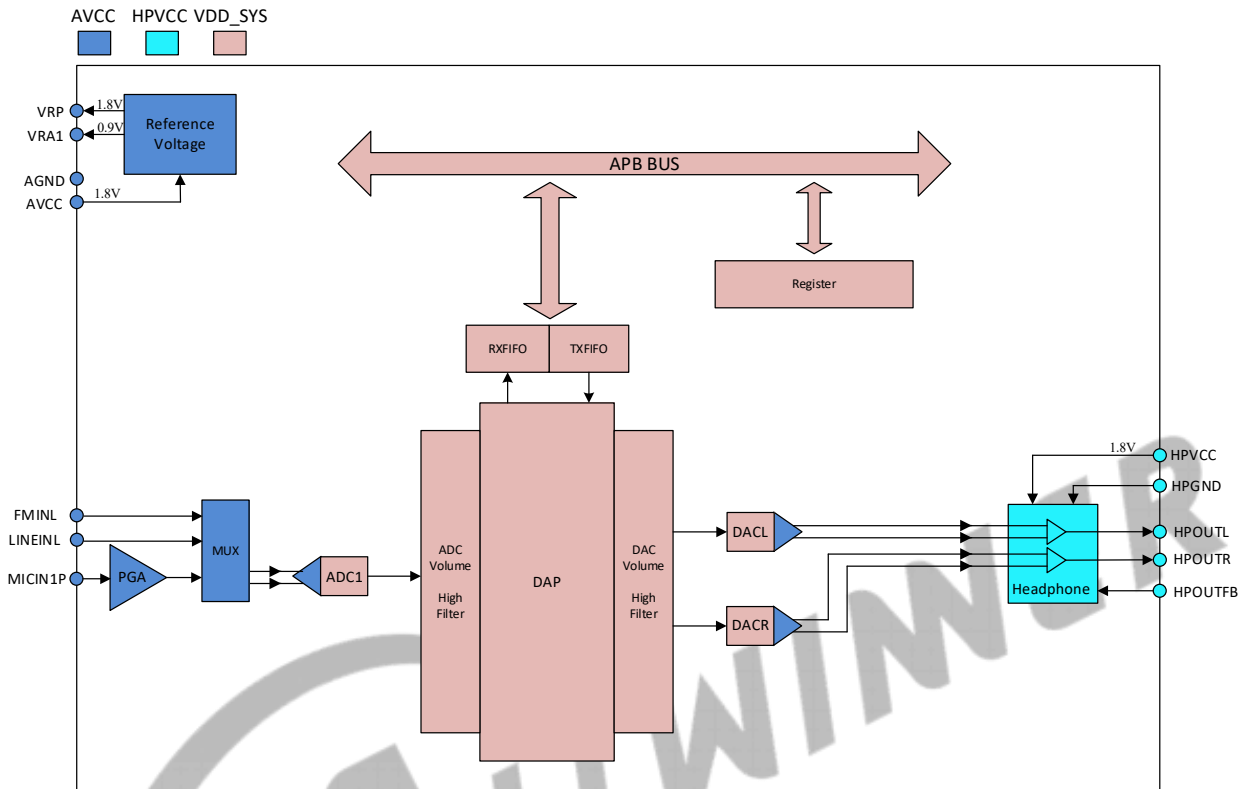
The Audio Codec has the following features:

- Two audio digital-to-analog(DAC) channel
 - SNR 100dB $\pm$ 2dB, THD+N -85 $\pm$ 3dB
 - Output level $\geq$ 0.54Vrms, Crosstalk $\leq$ -75 $\pm$ 3dB
 - Supports 16-bit and 20-bit Sample Resolution
 - Supports DAC Sample Rates from 8KHz to 192KHz
- One audio analog-to-digital(ADC) channels
 - SNR 95dB $\pm$ 3dB, THD+N -80 $\pm$ 3dB
 - Input level $\geq$ 800mFFS@3.2Vpp input
 - Supports 16-bit and 20-bit Sample Resolution
 - Sample Rates from 8KHz to 48KHz
- Audio inputs:
 - One single microphone inputs(MICIN1P)
 - One line-in input (LINEINL)
 - One FM-in input (FMINL)
- One analog audio output:
 - One stereo headphone output (HPOUTL and HPOUTR) .
- Stereo headphone driver
 - 95 $\pm$ 3dB SNR@A-weight
 - Output Level 0.54 Vrms@10k Ohm/THD+N -77 $\pm$ 3dB, 0.37Vrms@16 Ohm/THD+N -40dB
- Support Dynamic Range Controller adjusting the DAC playback and ADC capture
- One 128x20-bits FIFO for DAC data transmit, one 128x20-bits FIFO for ADC data receive
- Programmable FIFO thresholds
- Supports DMA and Interrupt

14.1.2 Block Diagram

The following figure shows the block diagram of Audio Codec.

Figure 14-1 Audio Codec Block Diagram



14.1.3 External Signals

Table 14-1 Audio Codec External Signals

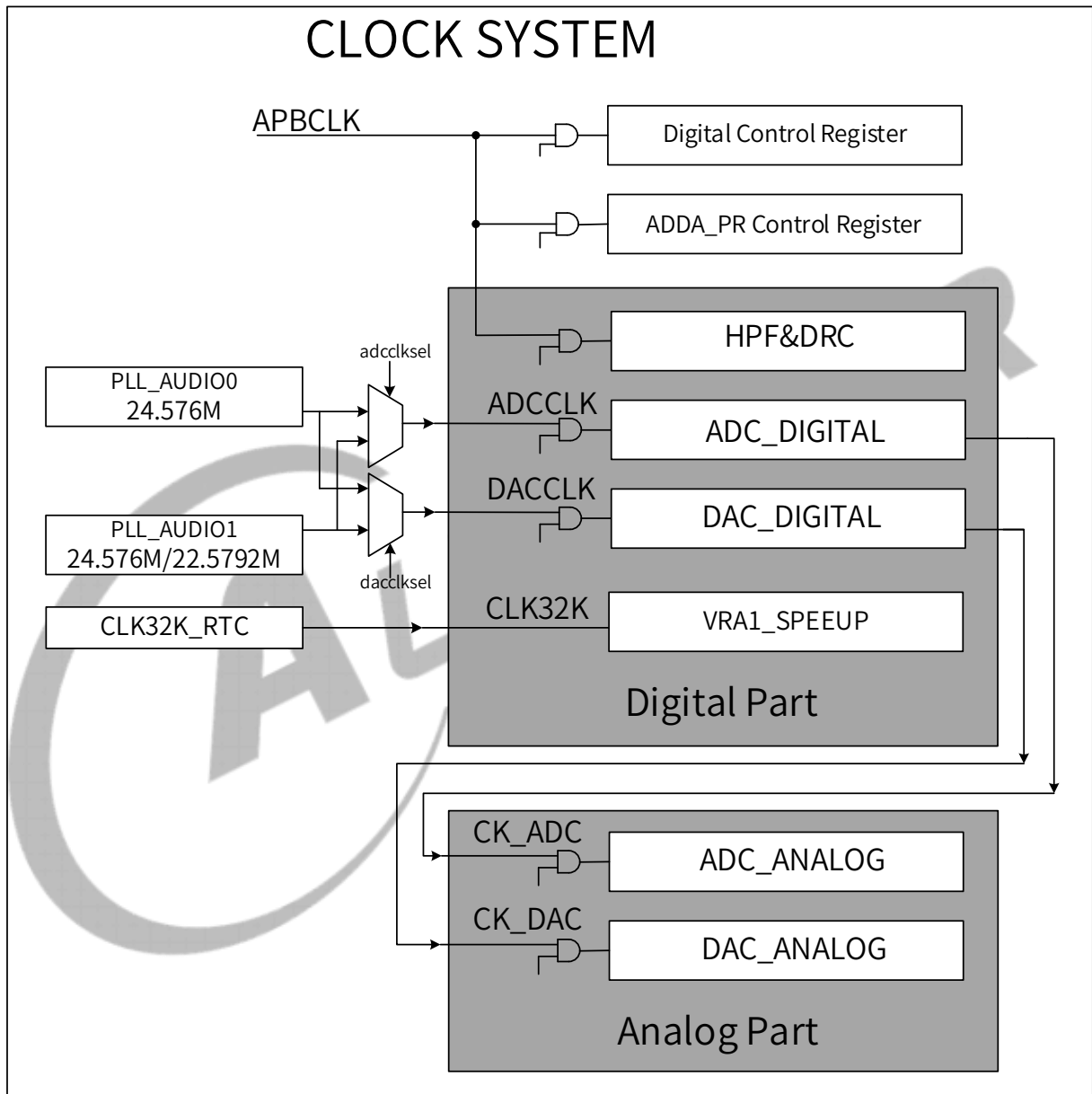
Signal Name	Description	Type
HPOUTL	Headphone Left Output	AO
HPOUTR	Headphone Right Output	AO
HPOUTFB	Pseudo Differential Headphone Ground Reference	AI
FMIN	FMIN Left Input	AI
LINEIN	LINEIN Left Single-End Input	AI
MICIN	Microphone Differential Positive Input	AI
VRA	Internal Reference Voltage	AO
VCC18-PLL-HPVCC	Headphone Power	P
AVCC	Power Supply for Analog Part	P
AGND	Analog Ground	G

14.1.4 Functional Description

14.1.4.1 Clock Sources

The following figure describes the clock source of Audio Codec. For clock setting, configuration, and gating information, refer to Section 4.1 Clock Controller Unit (CCU).

Figure 14-2 Audio Codec Clock Diagram



- Digital Part

The clock sources for the digital part are the PLL_AUDIO1. For the ADC clock, configure [AUDIO_CODEC_ADC_CLK_REG](#) [26:24] to select the clock source. For the DAC clock, configure [AUDIO_CODEC_DAC_CLK_REG](#) [26:24] to select the clock source. The PK-PK jitter of PLL_AUDIO should be less than 200 ps. PLL_AUDIO0 and PLL_AUDIO1 need use decimal division.

The clock source for VRA1_SPEEDUP is CLK32K from RTC.

- Analog Part

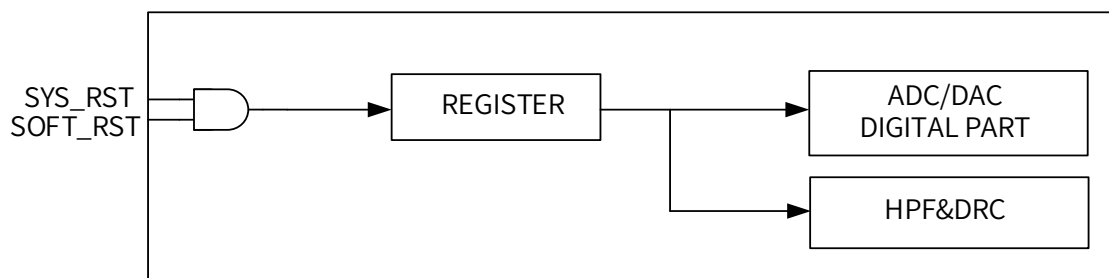
The clock source for the analog part is the CK_ADC and CK_DAC, both of which are divided from the digital part

14.1.4.2 Reset System

Digital Part Reset System

The following figure shows the reset system of the audio codec digital part.

Figure 14-3 Audio Codec Digital Part Reset System

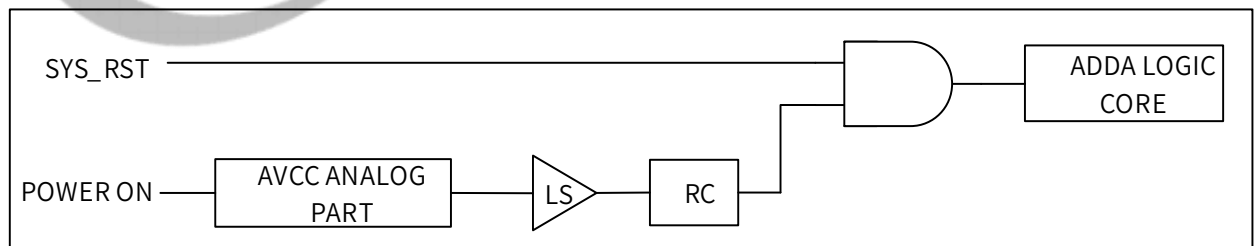


The SYS_RST reset comes from the VDD-SYS domain and is produced by the RTC domain which is controlled by CCU. Each domain has the de-bounce to confirm the reset system is strong. For the codec register part, MIX can be reset by the SYS_RST when being powered on or the system soft is writing the reset control logic. The other parts can be reset by the soft configuration through writing the register.

Analog Part Reset System

The following figure shows the reset system of the audio codec analog part.

Figure 14-4 Audio Codec Analog Part Reset System

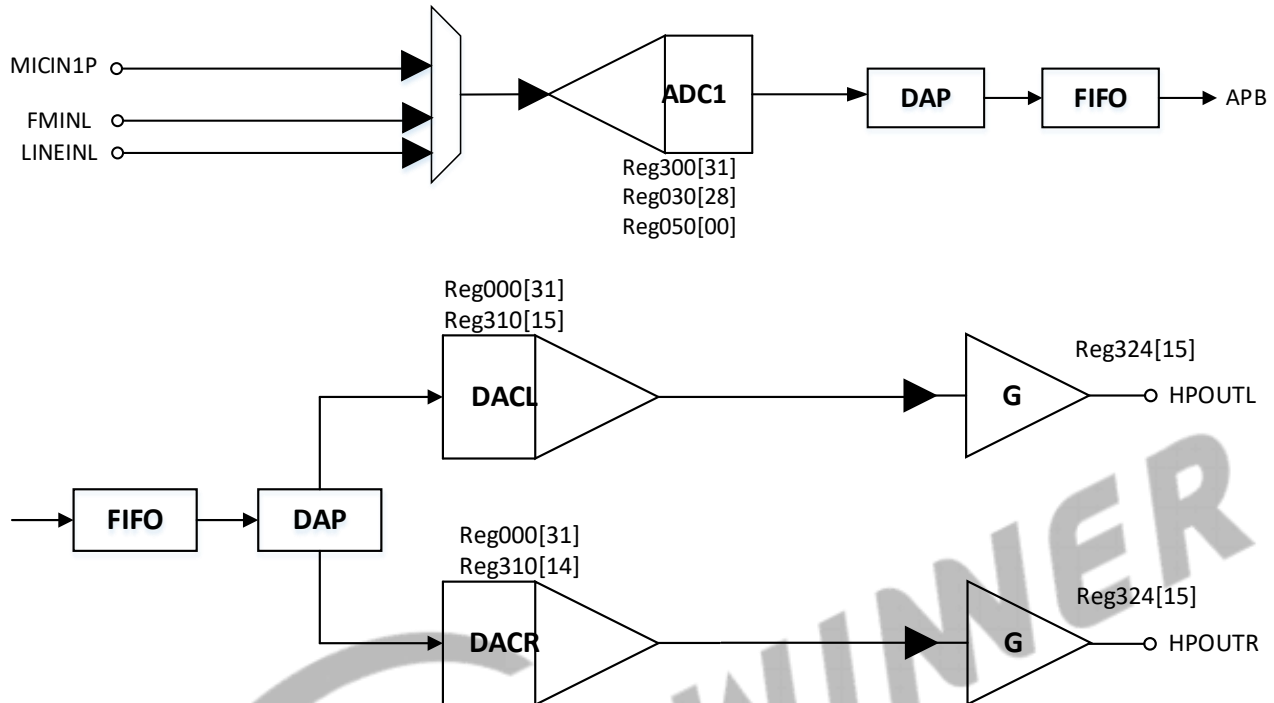


When AVCC is powered on, it sends the AVCC_POR signal. The AVCC_POR signal passes the level shift and RC filter part to the Audio Codec logic core.

14.1.4.3 Data Path Diagram

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

Figure 14-5 Audio Codec Data Path Diagram



14.1.4.4 One ADC

The ADC are used for recording sound and a reference signal. The sample rates of the ADC is independent of DAC sample rate. The digital ADC part can be enabled/ disabled by the bit[28] of the AC_ADC_FIFOC register.

14.1.4.5 Stereo DAC

The stereo DAC sample rate can be configured by setting the register. To save power, the analog DACL can be enabled or disabled by setting the bit[15] of the [DAC_REG](#) register, and the analog DACR can be enabled or disabled by setting the bit[14] of the [DAC_REG](#) register. The digital DAC part can be enabled or disabled by the bit[31] of the [AC_DAC_DPC](#) register.

14.1.4.6 Analog Audio Input Path

The Audio Codec supports 3 analog audio input paths:

- MICIN1P
- LINEINL
- FMINL

MICIN1P, LINEINL, FMINL provide single input that can be mixed into the ADC1 record mixer. The MICIN is a high impedance, low capacitance input suitable for connecting to various differential microphones of different dynamics and sensitivity. The gain for each pre-amplifier can be set independently.

14.1.4.7 Analog Audio Output Path

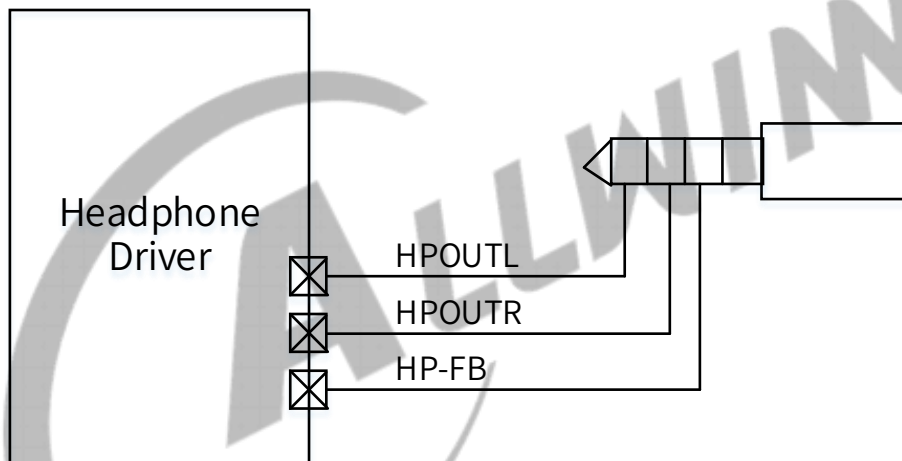
The Audio Codec has two types of analog output ports:

- HPOUTL
- HPOUTR

Headphone Output

The headphone PA is powered up or down by HP2_REG [bit21] (HP_DRVEN). HPOUTL/R can drive a 16R or 32R headphone load without DC capacitors by using Charge Pump to generate the negative rails. HP-FB is the ground loop noise rejection feedback.

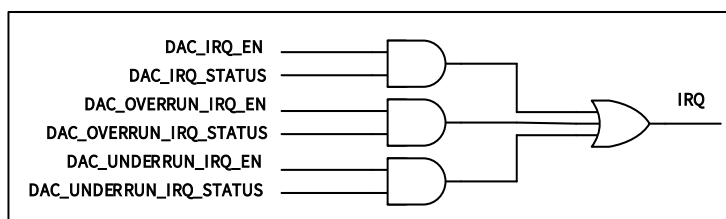
Figure 14-6 Headphone Output Application



14.1.4.8 Interrupt

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

Figure 14-7 Audio Codec Interrupt System

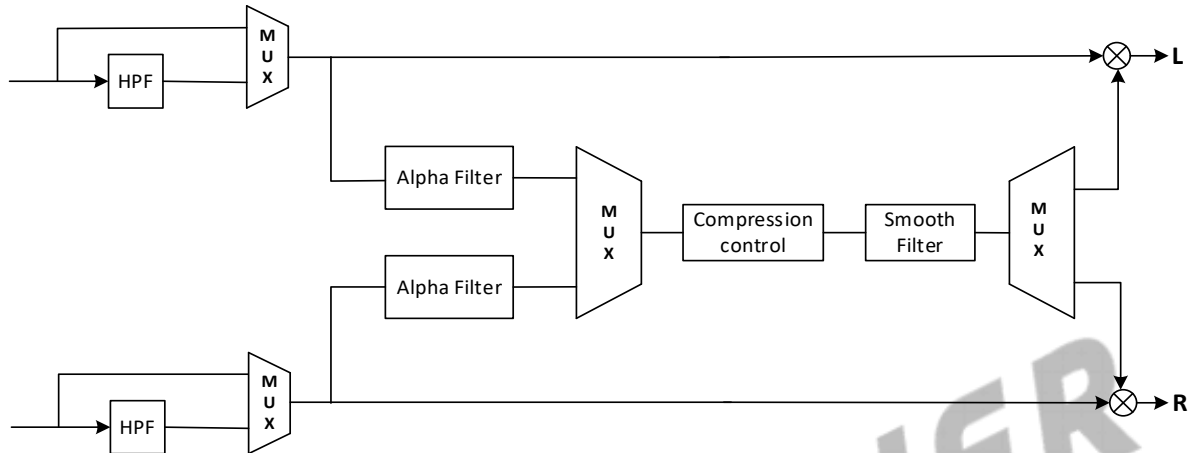


14.1.4.9 Digital Audio Processor (DAP)

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.

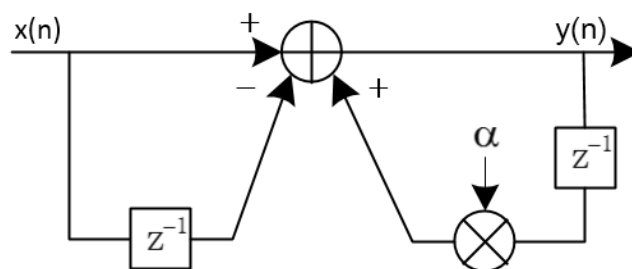
Figure 14-8 DAP Data Flow



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 14-9 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 14-10 DRC static Curve Parameters

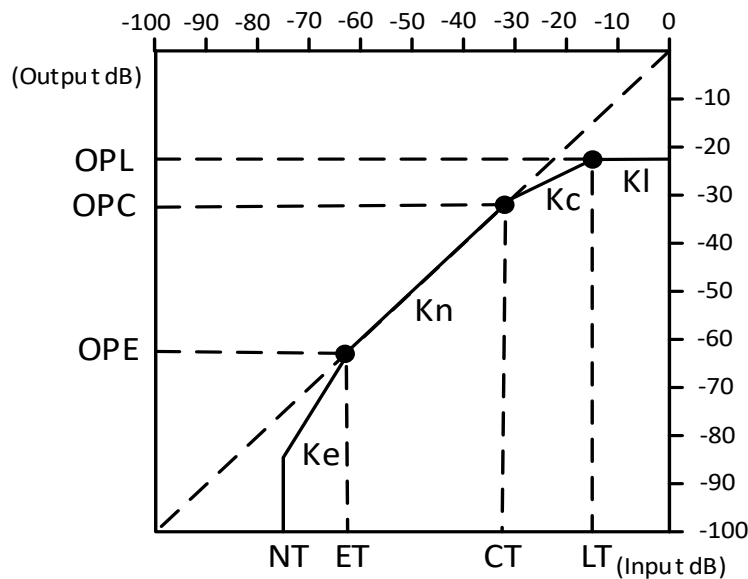
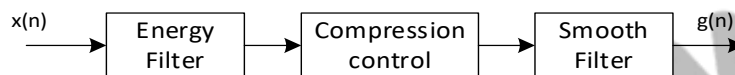


Figure 14-11 DRC Block Diagram



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:

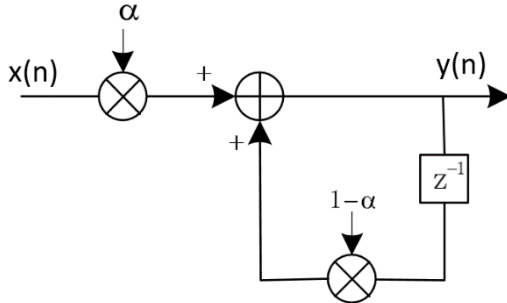
- **Number format**

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 14-12 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 $\alpha = 1 - e^{-2.2T_s/ta}$.

- **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 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 0110.1010 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

$$K = \frac{1}{n}$$

computed by

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

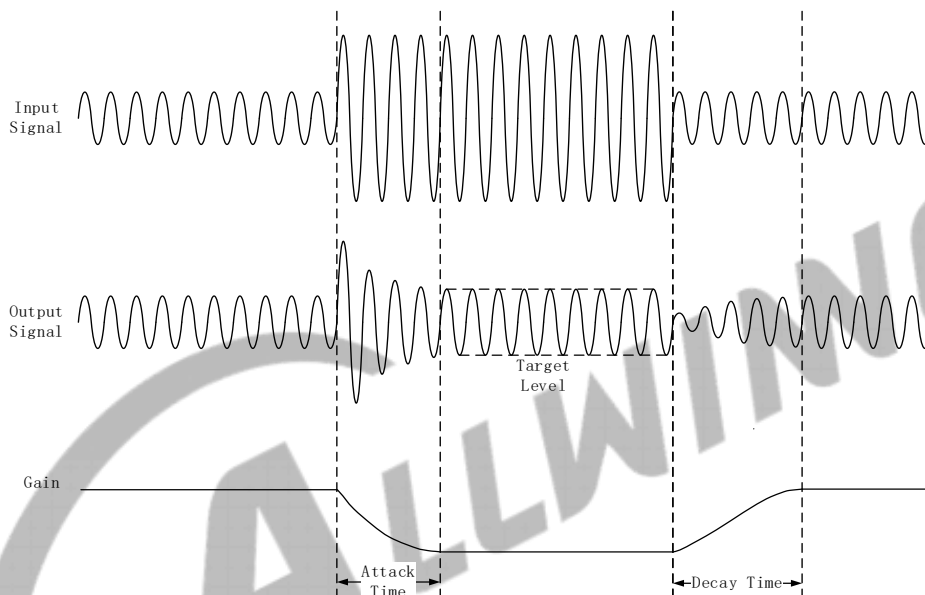
For example, it is desired to set 2:1, then the Kc require to set to 2:1 is $K_c = 1/2 = 0.5$, Kc is entered as a 32-bit number in 8.24 format.

Therefore, $K_c = 0.5 = 0000\ 0000.1000\ 0000\ 0000\ 0000\ 0000\ 0000 = 0x0080\ 0000$ 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 the following table. 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 $\alpha = 1 - e^{-2.2T_s / \tau_a}$.

Figure 14-13 Gain Smooth Filter



14.1.5 Programming Guidelines

14.1.5.1 Recording

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.

Step 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_AUDIO1_CTRL_REG](#) to configure PLL_AUDIO1 frequency and enable PLL_AUDIO1. For details, refer to Section 4.1 Clock Controller Unit (CCU).

Step 2 Configure the sample rate and data transfer format, then open the ADC.

Step 3 Configure the DMA and DMA request.

Step 4 Enable the ADC DRQ and DMA.

14.1.5.2 Playing

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

Step 1 Codec initialization: configure [AUDIO_CODEC_BGR_REG](#) and [AUDIO_CODEC_DAC_CLK_REG](#) to open the audio codec bus clock gating, release the bus reset, and open the PLL_AUDIO DAC clock gating; set up [PLL_AUDIO1_CTRL_REG](#) to configure PLL_AUDIO0 frequency and enable PLL_AUDIO0. For details of PLL_AUDIO0, refer to Section 4.1 Clock Controller Unit (CCU).

Step 2 Set up the sample rate and data transfer format, then open the DAC.

Step 3 Configure the DMA and DMA request.

Step 4 Enable the DAC DRQ and DMA.

14.1.6 Register List

Module Name	Base Address
AUDIOCODEC	0x02030000

Register Name	Offset	Description
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
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
AC_DAC_DRC_LPFHAT	0x010C	DAC DRC Left Peak Filter High Attack Time Coef Register

Register Name	Offset	Description
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

Register Name	Offset	Description
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
AC_DAC_DRC_SFHRT	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_LPFHAT	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_LRMSHAT	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_RRMSHAT	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

Register Name	Offset	Description
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_LLT	0x0258	ADC DRC Limiter Threshold Low Setting Register
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_SFHRT	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
DAC_REG	0x0310	DAC Analog Control Register
RAMP_REG	0x031C	RAMP Analog Control Register
HP2_REG	0x0340	HEADPHONE2 Analog Control Register
POWER_REG	0x0348	POWER Analog Control Register

Register Name	Offset	Description
ADC_CUR_REG	0x034C	ADC Current Analog Control Register

14.1.7 Register Description

14.1.7.1 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: Disable 1: Enable
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: Enable 1: Disable
23:19	/	/	/
18	R/W	0x0	HPF_EN High Pass Filter Enable 0: Disable 1: Enable
17:12	R/W	0x0	DVOL. Digital volume control: DVC, ATT=DVC[5:0]*(-1.16dB) 64 steps, -1.16dB/step
11	/	/	/
10:8	R/W	0x0	DITHER_SGM Dither Sigma 000: Reserved 001: 1/2 010: 1/2 ² 011: 1/2 ³ 100: 1/2 ⁴ 101: 1/2 ⁵ 110: 1/2 ⁶ 111: Reserved

Offset: 0x0000			Register Name: AC_DAC_DPC
Bit	Read/Write	Default/Hex	Description
7:4	R/W	0x0	DITHER_SFT Dither Shift Can increase the output amplitude of dither. 0000: $x2^0$ 0001: $x2^1$ 0010: $x2^2$ 0011: $x2^3$... 1011: $x2^{11}$ 1100: $x2^{12}$ 1101—1111: Reserved
3:2	/	/	/
1	R/W	0x0	DITHER_EN DSM Dither Enable 0: Disable 1: Enable
0	R/W	0x0	HUB_EN Audio Hub Enable Only if EN_DA set 1, HUB_EN can take effect. SYS Domin: ADDA/ I2S0/ OWA0 TXFIFO Hub Enable. 0: Disable 1: Enable

14.1.7.2 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: Disable 1: Enable

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

14.1.7.3 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	DAC_FS. Sample Rate of DAC 000: 48KHz 010: 24KHz 100: 12KHz 110: 192KHz 001: 32KHz 011: 16KHz 101: 8KHz 111: 96KHz 44.1KHz/22.05KHz/11.025KHz can be supported by Audio PLL Configure Bit

Offset: 0x0010			Register Name: AC_DAC_FIFOC
Bit	Read/Write	Default/Hex	Description
28	R/W	0	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 under run 0: Sending zero 1: Sending last audio sample
25:24	R/W	0x0	FIFO_MODE. For 20-bits transmitted audio sample: 00/10: FIFO_I[19:0] = {TXDATA[31:12]} 01/11: FIFO_I[19:0] = {TXDATA[19:0]} For 16-bits 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 less than or equal N, 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 Notes: 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 same data

Offset: 0x0010			Register Name: AC_DAC_FIFOC
Bit	Read/Write	Default/Hex	Description
5	R/W	0x0	TX_SAMPLE_BITS. Transmitting Audio Sample Resolution 0: 16 bits 1: 20 bits
4	R/W	0x0	DAC_DRQ_EN. DAC FIFO Empty DRQ Enable 0: Disable 1: Enable
3	R/W	0x0	DAC_IRQ_EN. DAC FIFO Empty IRQ Enable 0: Disable 1: Enable
2	R/W	0x0	FIFO_UNDERRUN_IRQ_EN. DAC FIFO Under Run IRQ Enable 0: Disable 1: Enable
1	R/W	0x0	FIFO_OVERRUN_IRQ_EN. DAC FIFO Over Run IRQ Enable 0: Disable 1: Enable
0	R/WC	0x0	FIFO_FLUSH. DAC FIFO Flush Write '1' to flush TX FIFO, self clear to '0'

14.1.7.4 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	/	/	/

Offset: 0x0014			Register Name: AC_DAC_FIFOS
Bit	Read/Write	Default/Hex	Description
3	R/WC	0x1	TXE_INT. TX FIFO Empty Pending Interrupt 0: No Pending IRQ 1: FIFO Empty Pending Interrupt Write '1' to clear this interrupt or automatic clear if interrupt condition fails.
2	R/WC	0x0	TXU_INT. TX FIFO Under run Pending Interrupt 0: No Pending Interrupt 1: FIFO Under run Pending Interrupt Write '1' to clear this interrupt
1	R/WC	0x0	TXO_INT. TX FIFO Overrun Pending Interrupt 0: No Pending Interrupt 1: FIFO Overrun Pending Interrupt Write '1' to clear this interrupt
0	/	/	/

14.1.7.5 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. Transmitting left, right channel sample data should be written this register one by one. The left channel sample data is first and then the right channel sample.

14.1.7.6 0x0024 DAC TX FIFO Counter Register (Default Value : 0x0000_0000)

Offset: 0x0024			Register Name: AC_DAC_CNT
Bit	Read/Write	Default/Hex	Description

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 been updated by the initial value, the counter register should count on base of this initial value. Notes: It is used for Audio/ Video Synchronization

14.1.7.7 0x0028 DAC Debug Register (Default Value : 0x0000_0000)

Offset: 0x0028			Register Name: AC_DAC_DG
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
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: cksel 0: CODEC Clock from PLL 1: CODEC Clock from OSC (for Debug)
7	/	/	/
6	R/W	0x0	DA_SWP. DAC output channel swap enable, da_swp 0: Disable 1: Enable
5:3	/	/	/
2:0	R/W	0x0	ADDA_LOOP_MODE ADDA Loop MODE SELECT 000: Disable 001: ADDA LOOP MODE DACL connect to ADC1 1xx: Reserved

14.1.7.8 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	ADFS. Sample Rate of ADC 000: 48KHz 010: 24KHz 100: 12KHz 110: Reserved 001: 32KHz 011: 16KHz 101: 8KHz 111: Reserved 44.1KHz/22.05KHz/11.025KHz can be supported by PLL_Audio Configure Bit
28	R/W	0x0	EN_AD. ADC Digital Part Enable 0: Disable 1: Enable
27:26	R/W	0x0	ADCFDT ADC FIFO Delay Time For writing Data after EN_AD 00:5ms 01:10ms 10:20ms 11:30ms
25	R/W	0x0	ADCDFEN ADC FIFO Delay Function For writing Data after EN_AD 0: Disable 1: Enable
24	R/W	0x0	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-bits 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-bits 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]}

Offset: 0x0030			Register Name: AC_ADC_FIFOC
Bit	Read/Write	Default/Hex	Description
23:22	/	/	/
21	R/W	0x0	RX_SYNC_EN_START Only if RX_SYNC_EN set 1, RX_SYNC_EN_START can take effect. SYS Domin: ADDA/ I2S0 RX Synchronize Enable Start. 0: Disable 1: Enable
20	R/W	0x0	RX_SYNC_EN Audiocodec RX Synchronize Enable 0: Disable 1: Enable
19:17	/	/	/
16	R/W	0x0	RX_SAMPLE_BITS. Receiving Audio Sample Resolution 0: 16 bits 1: 20 bits
15:12	/	/	/
11:4	R/W	0x40	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] Notes: WLEVEL represents the number of valid samples in the RX FIFO
3	R/W	0x0	ADC_DRQ_EN. ADC FIFO Data Available DRQ Enable. 0: Disable 1: Enable
2	R/W	0x0	ADC_IRQ_EN. ADC FIFO Data Available IRQ Enable. 0: Disable 1: Enable
1	R/W	0x0	ADC_OVERRUN_IRQ_EN. ADC FIFO Over Run IRQ Enable 0: Disable 1: Enable
0	R/WC	0x0	ADC_FIFO_FLUSH. ADC FIFO Flush. Write '1' to flush TX FIFO, self clear to '0'.

14.1.7.9 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	0xA0	PLACE HOLDER No meaning
23:16	R/W	0xA0	PLACE HOLDER1 No meaning
15:8	/	/	/
7:0	R/W	0xA0	ADC1_VOL ADC1 channel volume (-119.25dB To 71.25dB, 0.75dB/Step) 0x00: Mute 0x01: -119.25dB 0x9F = -0.75dB 0xA0 = 0dB 0xA1 = 0.75dB 0xFF = 71.25dB

14.1.7.10 0x0038 ADC FIFO Status Register (Default Value : 0x0000_0001)

Offset: 0x0038			Register Name: AC_ADC_FIFOS
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
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/WC	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 interrupt condition fails.
2	/	/	/

Offset: 0x0038			Register Name: AC_ADC_FIFOS
Bit	Read/Write	Default/Hex	Description
1	R/WC	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	PLACE HOLDER No meaning

14.1.7.11 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 Host can get one sample by reading this register. The left channel sample data is first and then the right channel sample.

14.1.7.12 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	RX_CNT. RX Sample Counter 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 been updated by the initial value, the counter register should count on base of this initial value. Notes: It is used for Audio/ Video Synchronization

14.1.7.13 0x0050 ADC Digital Control Register (Default Value: 0x0000_0000)

Offset: 0x0050			Register Name: ADC_DIG_CTRL
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/

Offset: 0x0050			Register Name: ADC_DIG_CTRL
Bit	Read/Write	Default/Hex	Description
16	R/W	0x0	ADC1_VOL_EN ADC1 Volume Control Enable 0: Disable 1: Enable
15:1	/	/	/
0	R/W	0x0	ADC_CHANNEL_EN Bit 0: ADC1 enable

14.1.7.14 0x0054 VRA1Speedup Down Control Register (Default Value: 0x0000_0000)

Offset: 0x0054			Register Name: VRA1SPEEDUP_DOWN_CTRL
Bit	Read/Write	Default/Hex	Description
31:05	/	/	/
4	R	0x0	VRA1SPEEDUP_DOWN_STATE Only if VAR1SPEEDUP_DOWN_Further_CTRL (0x348[29]) set 0, VAR1Speedup Down State is valid. 0: VAR1Speedup_Down not work 1: VAR1Speedup_Down work
3:2	/	/	/
1	R/W	0x0	VRA1SPEEDUP_DOWN_CTRL VAR1Speedup Down Manual Control Enable 0: Disable,VAR1Speedup Down convert 1 after bus rst release 32ms. 1: Enable,VAR1Speedup Down convert 1 immediately.
0	R/W	0x0	VRA1SPEEDUP_DOWN_RST_CTRL VAR1Speedup Down RST Manual Control Enable 0: Disable,VAR1Speedup Down convert 1 after bus rst release 32ms. 1: Enable,VAR1Speedup Down reset 0 immediately.

14.1.7.15 0x00F0 DAC DAP Control Register (Default Value : 0x0000_0000)

Offset: 0x00F0			Register Name: AC_DAC_DAP_CTRL
Bit	Read/Write	Default/Hex	Description

Offset: 0x00F0			Register Name: AC_DAC_DAP_CTRL
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	DDAP_EN DAP for DRC Enable 0 : bypass 1 : enable
30	/	/	/
29	R/W	0x0	DDAP_DRC_EN DRC enable control 0: Disable 1: Enable
28	R/W	0x0	DDAP_HPF_EN HPF enable control 0: Disable 1: Enable
27:0	/	/	/

14.1.7.16 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_DAP0_EN DAP for ADC enable 0: Bypass 1: Enable
30	/	/	/
29	R/W	0x0	ADC_DRC0_EN ADC DRC0 enable control 0: Disable 1: Enable
28	R/W	0x0	ADC_HPF0_EN ADC HPF0 enable control 0: Disable 1: Enable
27:0	/	/	/

14.1.7.17 0x0100 DAC DRC High HPF Coef Register (Default Value: 0x0000_00FF)

Offset: 0x0100			Register Name: AC_DAC_DRC_HHPFC
Bit	Read/Write	Default/Hex	Description
31:11	/	/	/

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

14.1.7.18 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	DAC_DRC_LHPF_COE HPF coefficient setting and the data is 3.24 format.

14.1.7.19 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	/	/	/
15	R	0	DAC_DRC_DELAY_BUF_STATE DRC delay buffer data output state when drc delay function is enable and the drc function disable. After disable drc function and this bit go to 0, the user should write the drc delay function bit to 0; 0 : not complete 1 : is complete
14:10	/	/	/
13:8	R/W	0	DAC_DRC_SIGNAL_DELAY_TIME_SET 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 disable, the signal delay time is unused.
7	R/W	0x1	DAC_DRC_DELAY_BUF_EN The delay buffer use or not when the drc disable and the drc buffer data output completely 0 : don't use the buffer 1 : use the buffer

Offset: 0x0108			Register Name: AC_DAC_DRC_CTRL
Bit	Read/Write	Default/Hex	Description
6	R/W	0x0	DAC_DRC_GAIN_MAX_LIMIT_EN DRC gain max limit enable 0 : disable 1 : enable
5	R/W	0x0	DAC_DRC_GAIN_MIN_LIMIT_EN DRC gain min limit enable. when this fuction enable, it will overwrite the noise detect fuction. 0 : disable 1 : enable
4	R/W	0x0	DAC_DRC_DETECT_NOISE_EN Control the drc to detect noise when ET enable 0 : disable 1 : enable
3	R/W	0x0	DAC_DRC_SIGNAL_FUNC_SEL Signal function Select 0 : RMS filter 1 : Peak filter When Signal function Select Peak filter, the RMS parameter is unused. (AC_DRC_LRMSHAT / AC_DRC_LRMSLAT / AC_DRC_LRMSHAT / AC_DRC_LRMSLAT) When Signal function Select 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 / AC_DRC_RPFLRT)
2	R/W	0x0	DAC_DRC_DELAY_FUNC_EN Delay function enable 0 : disable 1 : enable When the Delay function enable is disable, the Signal delay time is unused.
1	R/W	0x0	DAC_DRC_LT_EN DRC LT enable 0 : disable 1 : enable When the DRC LT is disable the LT, KI and OPL parameter is unused.

Offset: 0x0108			Register Name: AC_DAC_DRC_CTRL
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	DAC_DRC_ET_EN DRC ET enable 0 : disable 1 : enable When the DRC ET is disable the ET, Ke and OPE parameter is unused.

14.1.7.20 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	DAC_DRC_LPFHAT The left peak filter attack time parameter setting, which determine by the equation that $AT = 1 - \exp(-2.2T_s/ta)$. The format is 3.24. (The default value is 1ms)

14.1.7.21 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_DRC_LPFLAT The left peak filter attack time parameter setting, which determine by the equation that $AT = 1 - \exp(-2.2T_s/ta)$. The format is 3.24. (The default value is 1ms)

14.1.7.22 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	/	/	/

Offset: 0x0114			Register Name: AC_DAC_DRC_RPFHAT
Bit	Read/Write	Default/Hex	Description
15:0	R/W	0x000B	DAC_DRC_RPFHAT The right peak filter attack time parameter setting, which determine by the equation that $AT = 1 - \exp(-2.2Ts/ta)$. The format is 3.24. (The default value is 1ms)

14.1.7.23 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_DRC_RPFLAT The left peak filter attack time parameter setting, which determine by the equation that $AT = 1 - \exp(-2.2Ts/ta)$. The format is 3.24. (The default value is 1ms)

14.1.7.24 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_DRC_LPFHRT The left peak filter release time parameter setting, which determine by the equation that $RT = \exp(-2.2Ts/tr)$. The format is 3.24. (The default value is 100ms)

14.1.7.25 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	/	/	/

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

14.1.7.26 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	0x00FF	DAC_DRC_RPFHRT The right peak filter attack time parameter setting, which determine by the equation that $RT = \exp(-2.2Ts/tr)$. The format is 3.24. (The default value is 100ms)

14.1.7.27 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_DRC_RPFLRT The right peak filter release time parameter setting, which determine by the equation that $AT = \exp(-2.2Ts/tr)$. The format is 3.24. (The default value is 100ms)

14.1.7.28 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_DRC_LRMSHAT The left RMS filter average time parameter setting, which determine by the equation that $AT = 1 - \exp(-2.2Ts/tav)$. The format is 3.24. (The default value is 10ms)

14.1.7.29 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_DRC_LRMSLAT The left RMS filter average time parameter setting, which determine by the equation that $AT = 1 - \exp(-2.2T_s/\tau_{av})$. The format is 3.24. (The default value is 10ms)

14.1.7.30 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	0x0001	DAC_DRC_RRMSHAT The right RMS filter average time parameter setting, which determine by the equation that $AT = 1 - \exp(-2.2T_s/\tau_{av})$. The format is 3.24.(The default value is 10ms)

14.1.7.31 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_DRC_RRMSLAT The right RMS filter average time parameter setting, which determine by the equation that $AT = 1 - \exp(-2.2T_s/\tau_{av})$. The format is 3.24.(10ms)

14.1.7.32 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_DRC_HCT The compressor threshold setting, which set by the equation that $CT_{in} = -CT/6.0206$. The format is 8.24. (The default value is -40dB)

14.1.7.33 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 set by the equation that $CT_{in} = -CT/6.0206$. The format is 8.24. (The default value is -40dB)

14.1.7.34 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 determine by the equation that $K_c = 1/R$, there, R is the ratio of the compressor, which always is integer. The format is 8.24. (The default value is <2 : 1>)

14.1.7.35 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 determine by the equation that $K_c = 1/R$, there, R is the ratio of the compressor, which always is interger. The format is 8.24. (The default value is <2 : 1>)

14.1.7.36 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 determine by the equation $OPC_{in} = OPC/6.0206$. The format is 8.24. (The default value is -40dB)

14.1.7.37 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 determine by the equation $OPC_{in} = OPC / 6.0206$. The format is 8.24. (The default value is -40dB)

14.1.7.38 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 set by the equation that $LT_{in} = -LT / 6.0206$. The format is 8.24. (The default value is -10dB)

14.1.7.39 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 set by the equation that $LT_{in} = -LT / 6.0206$. The format is 8.24. (The default value is -10dB)

14.1.7.40 0x015C DAC DRC Limiter Slope High Setting Register (Default Value: 0x0000_0005)

Offset: 0x015C			Register Name: AC_DAC_DRC_HKL
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x0005	DAC_DRC_HKL The slope of the limiter which determine by the equation that $Kl = 1/R$, there, R is the ratio of the limiter, which always is integer. The format is 8.24. (The default value is <50:1>)

14.1.7.41 0x0160 DAC DRC Limiter Slope Low Setting Register (Default Value: 0x0000_1EB8)

Offset: 0x0160			Register Name: AC_DAC_DRC_LKL
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x1EB8	DAC_DRC_LKL The slope of the limiter which determine by the equation that $Kl = 1/R$, there, R is the ratio of the limiter, which always is integer. The format is 8.24. (The default value is <50 :1>)

14.1.7.42 0x0164 DAC DRC Limiter High Output at Limiter Threshold(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 determine by equation $OPTin=OPT/6.0206$. The format is 8.24. (The default value is -25dB)

14.1.7.43 0x0168 DAC DRC Limiter Low Output at Limiter Threshold(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 determine by equation $OPTin=OPT/6.0206$. The format is 8.24. (The default value is -25dB)

14.1.7.44 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 set by the equation that $ETin = -ET/6.0206$, The format is 8.24. (The default value is -70dB)

14.1.7.45 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	/	/	/
15:0	R/W	0x7291	DAC_DRC_LET The expander threshold setting, which set by the equation that $ET_{in} = -ET/6.0206$, The format is 8.24. (The default value is -70dB)

14.1.7.46 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
31:16	/	/	/
15:0	R/W	0x0500	DAC_DRC_HKE The slope of the expander which determine by the equation that $Ke = 1/R$, there, R is the ratio of the expander, which always is integer and the ke must larger than 50. The format is 8.24. (The default value is <1:5>)

14.1.7.47 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 determine by the equation that $Ke = 1/R$, there, R is the ratio of the expander, which always is integer and the ke must larger than 50. The format is 8.24. (The default value is <1:5>)

14.1.7.48 0x017C DAC DRC Expander High Output at Expander Threshold(Default Value: 0x0000_F45F)

Offset: 0x017C			Register Name: AC_DAC_DRC_HOPE
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/

Offset: 0x017C			Register Name: AC_DAC_DRC_HOPE
Bit	Read/Write	Default/Hex	Description
15:0	R/W	0xF45F	DAC_DRC_HOPE The output of the expander which determine by equation $OPE_{in} = OPE / 6.0206$. The format is 8.24. (The default value is -70dB)

14.1.7.49 0x0180 DAC DRC Expander Low Output at Expander Threshold(Default Value: 0x0000_8D6E)

Offset: 0x0180			Register Name: AC_DAC_DRC_LOPE
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x8D6E	DAC_DRC_LOPE The output of the expander which determine by equation $OPE_{in} = OPE / 6.0206$. The format is 8.24. (The default value is -70dB)

14.1.7.50 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 determine by the equation that $K_n = 1/R$, there, R is the ratio of the linear, which always is integer . The format is 8.24. (The default value is <1:1>)

14.1.7.51 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 determine by the equation that $K_n = 1/R$, there, R is the ratio of the linear, which always is integer . The format is 8.24. (The default value is <1:1>)

14.1.7.52 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 determine by the equation that $AT = 1 - \exp(-2.2T_s/tr)$. The format is 3.24. (The default value is 5ms)

14.1.7.53 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	/	/	/
15:0	R/W	0x5600	DAC_DRC_SFLAT The smooth filter attack time parameter setting, which determine by the equation that $AT = 1 - \exp(-2.2T_s/tr)$. The format is 3.24. (The default value is 5ms)

14.1.7.54 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 determine by the equation that $RT = 1 - \exp(-2.2T_s/tr)$. The format is 3.24. (The default value is 200ms)

14.1.7.55 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

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 determine by the equation that $RT = 1 - \exp(-2.2T_s/tr)$. The format is 3.24. (The default value is 200ms)

14.1.7.56 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 determine by equation $MXG_{in} = MXG/6.0206$. The format is 8.24 and must $-20dB < MXG < 30dB$. (The default value is -10dB)

14.1.7.57 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	/	/	/
15:0	R/W	0xCB0F	DAC_DRC_MXGLS The max gain setting which determine by equation $MXG_{in} = MXG/6.0206$. The format is 8.24 and must $-20dB < MXG < 30dB$. (The default value is -10dB)

14.1.7.58 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 determine by equation $MNG_{in} = MNG/6.0206$. The format is 8.24 and must $-60dB \leq MNG \leq -40dB$. (The default value is -40dB)

14.1.7.59 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 determine by equation $MNG_{in} = MNG / 6.0206$. The format is 8.24 and must $-60dB \leq MNG \leq -40dB$. (The default value is -40dB)

14.1.7.60 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 determine by the equation that $RT = 1 - \exp(-2.2T_s/tr)$. The format is 3.24. (The default value is 30ms)

14.1.7.61 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	/	/	/
15:0	R/W	0x640C	DAC_DRC_EPSLC The gain smooth filter release and attack time parameter setting in expander region, which determine by the equation that $RT = 1 - \exp(-2.2T_s/tr)$. The format is 3.24. (The default value is 30ms)

14.1.7.62 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 the hpf coefficient setting which format is 3.24.(gain = 1)

14.1.7.63 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 the hpf coefficient setting which format is 3.24.(gain = 1)

14.1.7.64 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.

14.1.7.65 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	ADC_DRC_LHPFC HPF coefficient setting and the data is 3.24 format.

14.1.7.66 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	ADC_DRC_DELAY_BUF_OUTPUT_STATE DRC delay buffer data output state when drc delay function is enable and the drc function disable. After disable drc function and this bit go to 0, the user should write the drc delay function bit to 0; 0 : not complete 1 : is complete
14:10	/	/	/

Offset: 0x0208			Register Name: AC_ADC_DRC_CTRL
Bit	Read/Write	Default/Hex	Description
13:8	R/W	0x0	ADC_DRC_SIGNAL_DELAY_TIME_SET 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 disable, the signal delay time is unused.
7	R/W	0x1	ADC_DRC_DELAY_BUF_EN The delay buffer use or not when the drc disable and the drc buffer data output completely 0 : don't use the buffer 1 : use the buffer
6	R/W	0x0	ADC_DRC_GAIN_MAX_LIMIT_EN DRC gain max limit enable 0 : disable 1 : enable
5	R/W	0x0	ADC_DRC_GAIN_MIN_LIMIT_EN DRC gain min limit enable. when this fuction enable, it will overwrite the noise detect funciton. 0 : disable 1 : enable
4	R/W	0x0	ADC_DRC_DETECT_NOISE_EN Control the drc to detect noise when ET enable 0 : disable 1 : enable

Offset: 0x0208			Register Name: AC_ADC_DRC_CTRL
Bit	Read/Write	Default/Hex	Description
3	R/W	0x0	ADC_DRC_SIGNAL_FUNC_SEL Signal function Select 0 : RMS filter 1 : Peak filter When Signal function Select Peak filter, the RMS parameter is unused. (AC_DRC_LRMSLAT / AC_DRC_LRMSLAT / AC_DRC_LRMSLAT / AC_DRC_LRMSLAT) When Signal function Select 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 / AC_DRC_RPFLRT)
2	R/W	0x0	ADC_DRC_DELAY_FUNC_EN Delay function enable 0 : disable 1 : enable When the Delay function enable is disable, the Signal delay time is unused.
1	R/W	0x0	ADC_DRC_LT_EN DRC LT enable 0 : disable 1 : enable DRC LT enable 0: Disable 1: Enable When the DRC LT is disable the LT, KI and OPL parameter is unused.
0	R/W	0x0	ADC_DRC_ET_EN DRC ET enable 0 : disable 1 : enable When the DRC ET is disable the ET, Ke and OPE parameter is unused.

14.1.7.67 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

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

14.1.7.68 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 determine by the equation that $AT = 1 - \exp(-2.2T_s/ta)$. The format is 3.24. (The default value is 1ms)

14.1.7.69 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 determine by the equation that $AT = 1 - \exp(-2.2T_s/ta)$. The format is 3.24. (The default value is 1ms)

14.1.7.70 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	/	/	/

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

14.1.7.71 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 determine by the equation that $RT = \exp(-2.2Ts/tr)$. The format is 3.24. (The default value is 100ms)

14.1.7.72 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 determine by the equation that $RT = \exp(-2.2Ts/tr)$. The format is 3.24. (The default value is 100ms)

14.1.7.73 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	/	/	/

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

14.1.7.74 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 determine by the equation that $AT = \exp(-2.2T_s/tr)$. The format is 3.24. (The default value is 100ms)

14.1.7.75 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 determine by the equation that $AT = 1 - \exp(-2.2T_s/tav)$. The format is 3.24. (The default value is 10ms)

14.1.7.76 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 determine by the equation that $AT = 1 - \exp(-2.2T_s/tav)$. The format is 3.24. (The default value is 10ms)

14.1.7.77 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
31:16	/	/	/
15:0	R/W	0x0001	ADC_DRC_RRMSHAT The right RMS filter average time parameter setting, which determine by the equation that $AT = 1 - \exp(-2.2T_s/t_{av})$. The format is 3.24. (The default value is 10ms)

14.1.7.78 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 determine by the equation that $AT = 1 - \exp(-2.2T_s/t_{av})$. The format is 3.24. (The default value is 10ms)

14.1.7.79 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 set by the equation that $CT_{in} = -CT/6.0206$. The format is 8.24. (The default value is -40dB)

14.1.7.80 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 set by the equation that $CT_{in} = -CT/6.0206$. The format is 8.24. (The default value is -40dB)

14.1.7.81 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 determine by the equation that $K_c = 1/R$, there, R is the ratio of the compressor, which always is integer. The format is 8.24. (The default value is <2 : 1>)

14.1.7.82 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 determine by the equation that $K_c = 1/R$, there, R is the ratio of the compressor, which always is integer. The format is 8.24. (The default value is <2 : 1>)

14.1.7.83 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 determine by the equation $OPC_{in} = OPC / 6.0206$. The format is 8.24. (The default value is -40dB)

14.1.7.84 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	/	/	/

Offset: 0x0250			Register Name: AC_ADC_DRC_LOPC
Bit	Read/Write	Default/Hex	Description
15:0	R/W	0x2C3F	ADC_DRC_LOPC The output of the compressor which determine by the equation $OPC_{in} = OPC / 6.0206$. The format is 8.24. (The default value is -40dB)

14.1.7.85 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 set by the equation that $LT_{in} = -LT / 6.0206$. The format is 8.24. (The default value is -10dB)

14.1.7.86 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 set by the equation that $LT_{in} = -LT / 6.0206$. The format is 8.24. (The default value is -10dB)

14.1.7.87 0x025C ADC DRC Limiter Slope High Setting Register (Default Value: 0x0000_0005)

Offset: 0x025C			Register Name: AC_ADC_DRC_HKL
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x0005	ADC_DRC_HKL The slope of the limiter which determine by the equation that $K_l = 1/R$, there, R is the ratio of the limiter, which always is integer. The format is 8.24. (The default value is <50:1>)

14.1.7.88 0x0260 ADC DRC Limiter Slope Low Setting Register (Default Value: 0x0000_1EB8)

Offset: 0x0260			Register Name: AC_ADC_DRC_LKL

Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x1EB8	ADC_DRC_LKL The slope of the limiter which determine by the equation that $K_l = 1/R$, there, R is the ratio of the limiter, which always is integer. The format is 8.24. (The default value is <50:1>)

14.1.7.89 0x0264 ADC DRC Limiter High Output at Limiter Threshold(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 determine by equation $OPT_{in} = OPT/6.0206$. The format is 8.24. (The default value is -25dB)

14.1.7.90 0x0268 ADC DRC Limiter Low Output at Limiter Threshold(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 determine by equation $OPT_{in} = OPT/6.0206$. The format is 8.24. (The default value is -25dB)

14.1.7.91 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 set by the equation that $ET_{in} = -ET/6.0206$. The format is 8.24. (The default value is -70dB)

14.1.7.92 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	/	/	/
15:0	R/W	0x7291	ADC_DRC_LET The expander threshold setting, which set by the equation that $ET_{in} = -ET/6.0206$. The format is 8.24. (The default value is -70dB)

14.1.7.93 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 determine by the equation that $Ke = 1/R$, there, R is the ratio of the expander, which always is integer and the ke must larger than 50. The format is 8.24. (The default value is <1:5>)

14.1.7.94 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 determine by the equation that $Ke = 1/R$, there, R is the ratio of the expander, which always is integer and the ke must larger than 50. The format is 8.24. (The default value is <1:5>)

14.1.7.95 0x027C ADC DRC Expander High Output at Expander Threshold(Default Value: 0x0000F45F)

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 determine by equation $OPE_{in} = OPE/6.0206$. The format is 8.24. (The default value is -70dB)

14.1.7.96 0x0280 ADC DRC Expander Low Output at Expander Threshold(Default Value: 0x0000_8D6E)

Offset: 0x0280			Register Name: AC_ADC_DRC_LOPE
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x8D6E	ADC_DRC_LOPE The output of the expander which determine by equation $OPE_{in} = OPE / 6.0206$. The format is 8.24. (The default value is -70dB)

14.1.7.97 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 determine by the equation that $K_n = 1/R$, there, R is the ratio of the linear, which always is integer. The format is 8.24. (The default value is <1:1>)

14.1.7.98 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 determine by the equation that $K_n = 1/R$, there, R is the ratio of the linear, which always is integer. The format is 8.24. (The default value is <1:1>)

14.1.7.99 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	/	/	/

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

14.1.7.100 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	/	/	/
15:0	R/W	0x5600	ADC_DRC_SFLAT The smooth filter attack time parameter setting, which determine by the equation that $AT = 1 - \exp(-2.2T_s/tr)$. The format is 3.24. (The default value is 5ms)

14.1.7.101 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 determine by the equation that $RT = 1 - \exp(-2.2T_s/tr)$. The format is 3.24. (The default value is 200ms)

14.1.7.102 0x0298 ADC DRC Smooth filter Gain Low Release Time Coef Register (Default Value: 0x0000_0F04)

Offset: 0x0298			Register Name: AC_ADC_DRC_SFLRT
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/

Offset: 0x0298			Register Name: AC_ADC_DRC_SFLRT
Bit	Read/Write	Default/Hex	Description
15:0	R/W	0x0F04	ADC_DRC_SFLRT The gain smooth filter release time parameter setting, which determine by the equation that $RT = 1 - \exp(-2.2T_s/tr)$. The format is 3.24. (The default value is 200ms)

14.1.7.103 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 determine by equation $MXG_{in} = MXG/6.0206$. The format is 8.24 and must $-20dB < MXG < 30dB$ (The default value is -10dB)

14.1.7.104 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	/	/	/
15:0	R/W	0xCB0F	ADC_DRC_MXGLS The max gain setting which determine by equation $MXG_{in} = MXG/6.0206$. The format is 8.24 and must $-20dB < MXG < 30dB$ (The default value is -10dB)

14.1.7.105 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 determine by equation $MNG_{in} = MNG/6.0206$. The format is 8.24 and must $-60dB \leq MNG \leq -40dB$. (The default value is -40dB)

14.1.7.106 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 determine by equation $MNG_{in} = MNG / 6.0206$. The format is 8.24 and must $-60dB \leq MNG \leq -40dB$. (The default value is -40dB)

14.1.7.107 0x02AC ADC DRC 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 determine by the equation that $RT = 1 - \exp(-2.2T_s/tr)$. The format is 3.24. (The default value is 30ms)

14.1.7.108 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	/	/	/
15:0	R/W	0x640C	ADC_DRC_EPSLC The gain smooth filter release and attack time parameter setting in expander region, which determine by the equation that $RT = 1 - \exp(-2.2T_s/tr)$. The format is 3.24. (The default value is 30ms)

14.1.7.109 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 the hpf coefficient setting which format is 3.24.(gain = 1)

14.1.7.110 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 the hpf coefficient setting which format is 3.24.(gain = 1)

14.1.7.111 0x0300 ADC1 Analog Control Register (Default Value:0x0010_0054)

Offset:0x0300			Register Name: ADC1_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	ADC1_EN ADC1 Channel Enable 0: Disable 1: Enable
30	R/W	0x0	MIC1_PGA_EN MIC1 PGA Enable 0: Disable 1: Enable
29	R/W	0x0	ADC1_DITHER_CTRL ADC1 Dither Control 0: New Dither Off 1: New Dither On
28	/	/	/
27	R/W	0x0	FMINLEN FMINL Enable 0: Disable 1: Enable
26	R/W	0x0	CROSSTALK_AVOID_EN CROSSTALK AVOID Enable 0: Disable 1: Enable
25:24	R/W	0x0	DSM_DITHER_LVL Dither Level Control (Dither level is positive related to the ctrl bits) 0: No Level 1: Min Level 2: Middle Level 3: Max Level

Offset:0x0300			Register Name: ADC1_REG
Bit	Read/Write	Default/Hex	Description
23	R/W	0x0	LINEINLEN LINEINL Enable 0: Disable 1: Enable
22	/	/	/
21:20	R/W	0x1	IOPAAF2S PGA OPAAF2 Bias Current Select 00: 6uA 01: 7uA 10: 8uA 11: 9uA
19:18	R/W	0x0	LINEINL_GAIN CTRL LINEINL gain ctrl 00: 0dB 11: 6dB
17:16	R/W	0x0	FML_GAIN CTRL FML gain ctrl 00: 0dB 11: 6dB
15:14	/	/	/
13	R/W	0x0	DSM RESET CTRL DSM reset ctrl 0: reset ctrl by micen lineen fmen 1: reset disable ignore micen lineen fmen
12	R/W	0x0	MIC_START_CTRL MIC start ctrl 0: NORMAL 1: MIC_CM STARUP SPEEDUP

Offset:0x0300			Register Name: ADC1_REG
Bit	Read/Write	Default/Hex	Description
11:8	R/W	0x0	ADC1_PGA_GAIN_CTRL ADC1 PGA gain settings: 0x0: 0 dB 0x1: 6 dB 0x2: 6 dB 0x3: 6 dB 0x4: 9 dB 0x5: 12 dB 0x6: 15 dB 0x7: 18 dB 0x8: 21 dB 0x9: 24 dB 0xA: 27 dB 0xB: 30 dB 0xC: 33 dB 0xD: 36 dB 0xE: 36 dB 0xF: 36 dB
7:6	R/W	0x1	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 BIAS_REG<11:10> from 1uA to 4uA For example: BIAS_REG<11:10> =11, IOPADC=4uA 00: 1.50*4uA=6uA 01: 1.75*4uA=7uA 10: 2.00*4uA=8uA 11: 2.25*4uA=9uA
5:4	R/W	0x1	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 BIAS_REG<11:10> from 1uA to 4uA

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 BIAS_REG<11:10> from 1uA to 4uA
1	R/W	0x0	PGA1_OP1_CHOPPER_EN PGA1 OP1 CHOPPER EN 0: PGA1 OP1 disable 1: PGA1 OP1 en
0	R/W	0x0	PGA1_OP2_CHOPPER_EN PGA1 OP2 CHOPPER EN 0: PGA1 OP2 disable 1: PGA1 OP2 en

14.1.7.112 0x0310 DAC Analog Control Register (Default Value:0x0000_0000)

Offset: 0x0310			Register Name: DAC_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	DAC_CH_EN DAC Chopper Control 0: Disable 1: Enable
30	R/W	0x0	DAC_CH_NOL_EN DAC Chopper Non-overlap Control 0: Disable 1: Enable
29:28	R/W	0x0	DAC_CH_CKSET DAC Chopper Frequency Settings: 00: 96kHz 01: 192kHz 10: 384kHz 11: 768kHz

Offset: 0x0310			Register Name: DAC_REG
Bit	Read/Write	Default/Hex	Description
27:26	R/W	0x0	DAC_CH_DLYSET DAC Chopper Ck Delay Level Control (positive related to the ctrl bits) 00: 5ns 01: 10ns 10: 12ns 11: 15ns
25:24	R/W	0x0	DAC_CH_NOLDLY_SET DACPF Chopper Ck Non-overlap Level Control (positive related to the ctrl bits)
23:16	/	/	/
15	R/W	0x0	DACL_EN DACL Enable 0: Disable 1: Enable
14	R/W	0x0	DACR_EN DACR Enable 0: Disable 1: Enable
13	/	/	/
12	R/W	0x0	LMUTE DACL to Left Channel LINEOUT Mute Control 0: Mute 1: Not mute
11	/	/	/
10	R/W	0x0	RMUTE DACR to Right Channel LINEOUT Mute Control 0: Mute 1: Not mute
9:8	R/W	0x0	CKDAC_DLYSET DAC Clock Delay Level Control (positive related to the ctrl bits) 00: 5ns 01: 10ns 10: 12ns 11: 15ns
7:0	/	/	/

14.1.7.113 0x031C RAMP Analog Control Register (Default Value:0x0018_0000)

Offset:0x031C	Register Name: RAMP_REG

Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	RAMP_RISE_INT_EN RAMP Rise Interrupt Enable 0: Enable 1: Disable
30	R/W1C	0x0	RAMP_RISE_INT RK Increase upward Finish and Rampen Pull Down Instruction 0: No Pending IRQ 1: Ramp Rise Finish Pending Interrupt Write '1' to clear this interrupt
29	R/W	0x0	RAMP_FALL_INT_EN RAMP Fall Int Enable 0: Enable 1: Disable
28	R/W1C	0x0	RAMP_FALL_INT RK Downward Decrease Finish and Rampen Pull Down Instruction 0: No Pending IRQ 1: Ramp Fall Finish Pending Interrupt Write '1' to clear this interrupt
27:25	/	/	/
24	R/W	0x0	RAMP_SRST Ramp Soft Reset 0: Disable 1: Enable
23:22	/	/	/
21	R/W	0x0	RK_OPT_EN RK Optimize Enable 0: Disable 1: Enable
20:16	R/W	0x18	RAMP_CLK_DIV_M Analog Ramp Clk Div Freq Value : M(from 0 to 31,Default:24). $Ana\_Ramp\_Clk = 24MHz / (M+1)$ Default Ramp Clk Freq: $24MHz / (24+1) = 960KHz$
15	R/W	0x0	HP_PULL_OUT_EN Heanphone Pullout Enable 0: Disable 1: Enable

14:12	R/W	0x0	RAMP_HOLD_STEP RAMP HOLD STEP 000: 9600 001: 19200 010: 38400 011: 76800 100: 96000 101: 115200 110: 153600 111: 192000 Ramp Hold Time = Ramp Hold Step/Ramp Clk Freq When Ramp Clk Freq equal to 960KHz, Corresponding Ramp Hold time of each gear: 000: 9600/960KHz=10ms 001: 19200/960KHz=20ms 010: 38400/960KHz=40ms 011: 76800/960KHz=80ms 100: 96000/960KHz=100ms 101: 115200/960KHz=120ms 110: 153600/960KHz=160ms 111: 192000/960KHz=200ms
11:10	/	/	/
9:8	R/W	0x0	GAP_STEP Gap Step 00: ramp step 01: ramp step*2 10: ramp step*3 11: ramp step*4
7	/	/	/

6:4	R/W	0x0	RAMP_STEP RK Frequency Gear, Control Ramp Rise/Fall Total Time 000: 20 001: 30 010: 40 011: 60 100: 80 101: 120 110: 160 111: 240 Ramp Rise/Fall Total Time =(Ramp Step/Ramp Clk Freq)*4*1024 When Default Ramp Clk Freq equal to 960KHz, Corresponding time of each gear: 000: (20/960KHz)*4*1024=85.3ms 001: (30/960KHz)*4*1024=128ms 010: (40/960KHz)*4*1024=170.6ms 011: (60/960KHz)*4*1024=256ms 100: (80/960KHz)*4*1024=341.3ms 101: (120/960KHz)*4*1024=512ms 110: (160/960KHz)*4*1024=682.6ms 111: (240/960KHz)*4*1024=1024ms
3	R/W	0x0	RMD_EN Ramp Manual Down Enable 0: Disable 1: Enable
2	R/W	0x0	RMU_EN Ramp Manual Up Enable 0: Disable 1: Enable
1	R/W	0x0	RMC_EN Ramp Manual Control Enable 0: Disable 1: Enable
0	R/W	0x0	RD_EN Ramp Digital Enable 0: Disable 1: Enable

14.1.7.114 0x0340 HEADPHONE2 Analog Control Register (Default Value:0x0600_4000)

Offset: 0x0340	Register Name: HP2_REG

Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	HPFB_BUF_EN Headphone Feedback Buffer OP Enable 0: Disable 1: Enable
30:28	R/W	0x0	HEADPHONE_GAIN HeadPhone Gain 000:0dB 001:-6dB 010:-12dB 011:-18dB 100:-24dB 101:-30dB 110:-36dB 111:-42dB
27:26	R/W	0x1	HPFB_RES Headphone Feedback Big Resistor Control 00: 0.28 MΩ 01: 0.6 MΩ 10: 0.88 MΩ 11: 1.20 MΩ
25:24	R/W	0x2	OPDRV_CUR Headphone OP Output Stage Current Setting 00: Min 11: Max
23:22	R/W	0x0	HP_OFFSET_ADJ Headphone Drive offset adjust 00: min 11: max
21	R/W	0x0	HP_DRVEN Headphone Driver Enable 0: Disable 1: Enable
20	R/W	0x0	HP_VO_OFFSET_EN Headphone Driver offset control Enable 0: Disable 1: Enable
19	R/W	0x0	RSWITCH RSwitch 0: HPOUT OUTPUT VCM of RAMP_DAC 1: VRA1

18	R/W	0x0	RAMPEN Ramp DAC Enable 0: Disable 1: Enable
17	R/W	0x0	HPFB_IN_EN Headphone Feedback PAD IN Switch Enable 0: Disable 1: Enable
16	R/W	0x0	RAMP_FINAL_CONTROL Headphone Ramp Final Step Control 0: Ramp Output Select Ramp 1: Ramp Output Select HPFB buffer Output
15	R/W	0x0	RAMP_OUT_EN Ramp Output Switch Enable 0: Disable 1: Enable
14:10	/	/	/
9:8	R/W	0x0	RAMP_FILTER Ramp filter resistor select 00: 0R 01: 1.57k 10: 3.14k 11: 9.43k
7	R/W	0x0	HP_CHOPPER_EN Headphone Chopper enable 0: Disable 1: Enable
6	R/W	0x0	HP_CH_NOL_EN Headphone Chopper Nov-overlapping Enable 0: Disable 1: Enable
5:4	R/W	0x0	HP_CHOPPER_CKS Headphone Chopper Clock frequency Select 00: 768k 01: 384k 10: 192k 11: 96k
3:2	R/W	0x0	HP_CH_DLYSET Headphone Chopper Delay Select 00: min 11: max

1:0	R/W	0x0	HP_CH_NOLDLY_SET Headphone Chpper Non-overlapping Delay Select 00: min 11: max

14.1.7.115 0x0348 POWER Analog Control Register (Default Value:0x0000_0000)

Offset:0x0348			Register Name: POWER_REG
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29	R/W	0x0	VAR1SPEEDUP_DOWN_FURTHER_CTRL VRA1Speedup Down Further Control In Adda Analog 0: Digital Signal Interface Pin l_vra1speedup (vra1_speedup_down) Normally Control VRA1 Speedup down 1: Manual Control Finish VRA1 Speedup down, Ignore Digital Signal Interface Pin l_vra1speedup (vra1_speedup_down) Control
28:17	/	/	/
16	R	0x0	AVCCPOR Avccpor Monitor
15	R/W	0x0	I_BG_BUF_DISABLE Bg buffer disable 0: Enable 1: Disable Note: gating is 1,default is 0
14:8	/	/	/
7:4	R/W	0x0	ALDO_TRMMING Aldo trim value 0000:1.8V During trimming, first perform coarse adjustment of the BG output, and then fine-tune the LDO output.
3:0	R/W	0x0	BG_TRMMING Trimming BG output 0000: BG output is 0.9 0000-1111-1110-1101-1100-1011-1010-1001-1000 output increases from 0.9 0000-0001-0010-0011-0100-0101-0110-0111 output decreases from 0.9

14.1.7.116 0x034C ADC Current Analog Control Register (Default Value:0x0000_0000)

Offset:0x034C			Register Name: ADC_CUR_REG
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:6	R/W	0x0	ADC1_PGA_CH_DLYSET PGA Chopper Ck Delay Level Control (positive related to the ctrl bits) 00: 5ns 01: 10ns 10: 12ns 11: 15ns
5	R/W	0x0	ADC1_PGA_CH_EN PGA Chopper Control 0: Disable 1: Enable
4	R/W	0x0	ADC1_PGA_CHOPPER_NOL_EN PGA Chopper Non-overlap Control 0: Disable 1: Enable
3:2	R/W	0x0	ADC1_PGA_CH_NOLDLYSET PGA Chopper Ck Non-overlap Level Control (positive related to the ctrl bits)
1:0	R/W	0x0	ADC1_PGA_CHOPPERCKS PGA Chopper Frequency Settings: 00: 96kHz 01: 192kHz 10: 384kHz 11: 768kHz

14.2 I2S/PCM

14.2.1 Overview

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

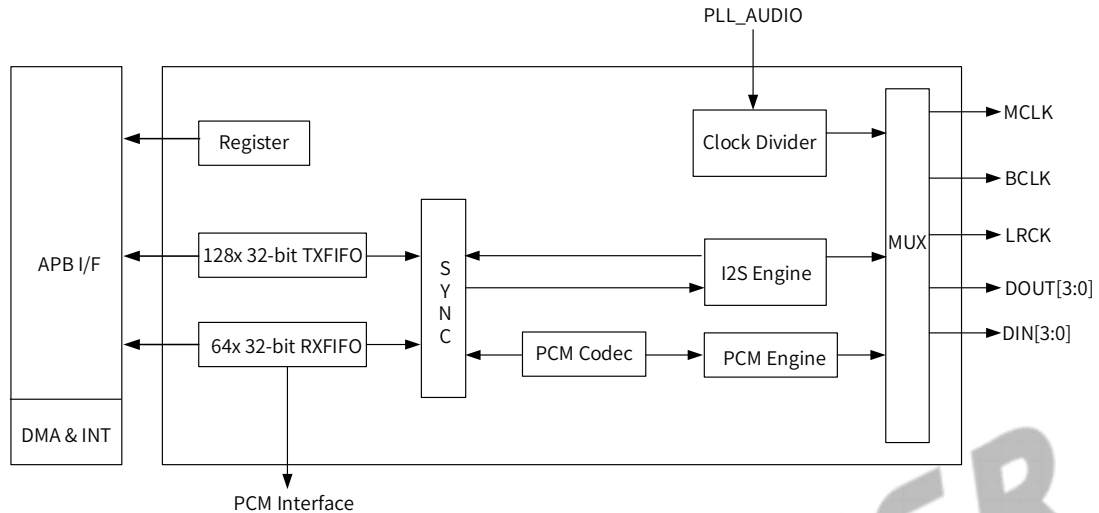
The I2S/PCM controller includes the following features:

- 1x I2S/PCM (I2S0)
- 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)
- FIFOs for transmitting and receiving data
 - Programmable FIFO thresholds
 - 128 depth x 32-bit width TXFIFO and 64 depth x 32-bit width RXFIFO
- Supports multiple function clocks
 - 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 (Only if the IO PAD and Peripheral I2S/PCM satisfy Timing Parameters)
- 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 ($\text{sample rate} \times \text{channels} \times \text{slot width} \leq 24.576 \text{ MHz}$)
 - 8-bit u-law and 8-bit A-law companded sample
- Supports master/slave mode

14.2.2 Block Diagram

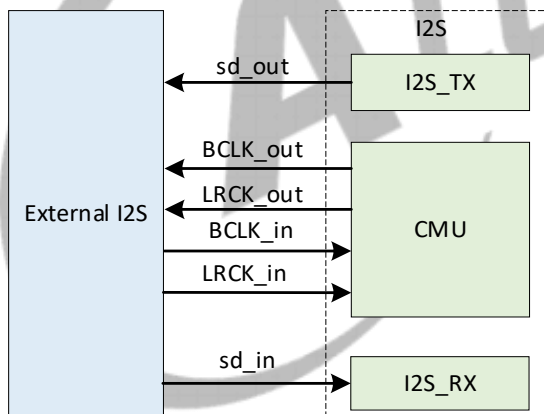
The following figure shows the functional block diagram of the I2S/PCM interface.

Figure 14-14 I2S/PCM Interface System Block Diagram



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

Figure 14-15 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 master 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 slave 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.

14.2.3 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 the serial data output pin, and DIN is the serial data input pin. For details about General Purpose I/O port, refer to the Section 15.7 GPIO.

Table 14-2 I2S/PCM External Signals

Signal Name	Description	Type
I2S0-MCLK	I2S0 Master Clock	O
I2S0-LRCK	I2S0/PCM0 Sample Rate Clock/Sync	I/O
I2S0-BCLK	I2S0/PCM0 Bit Rate Clock	I/O
I2S0-DIN0	I2S0/PCM0 Serial Data Input 0	I
I2S0-DOUT0	I2S0/PCM0 Serial Data Output 0	O

14.2.4 Functional Description

14.2.4.1 I2S/PCM Transmit Modes

The I2S/PCM supports standard I2S mode, Left-justified I2S mode, Right-justified I2S mode, PCM mode, and TDM mode. The software can select the modes by setting [I2S/PCM_CTL](#). The following figures describe the waveforms for SYNC, BCLK, DOUT, and DIN in different modes.

Figure 14-16 I2S Standard Mode Timing

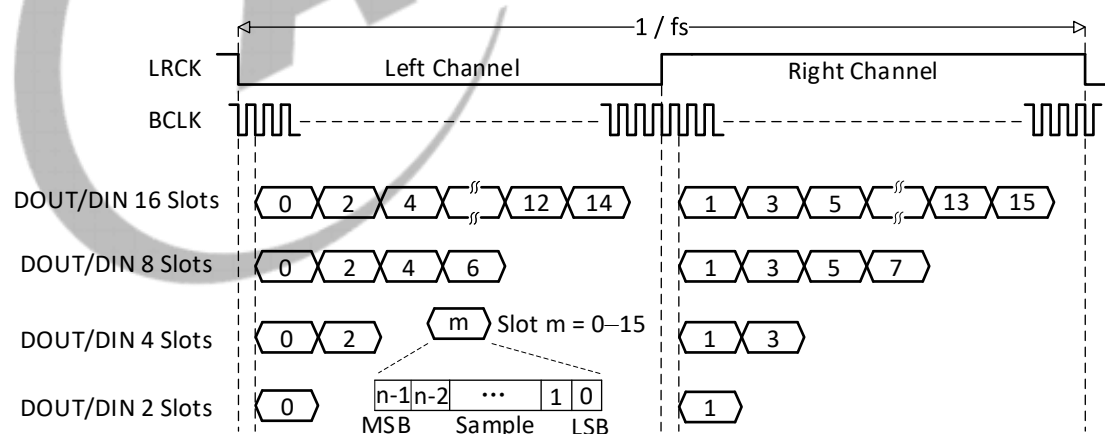


Figure 14-17 Left-Justified Mode Timing

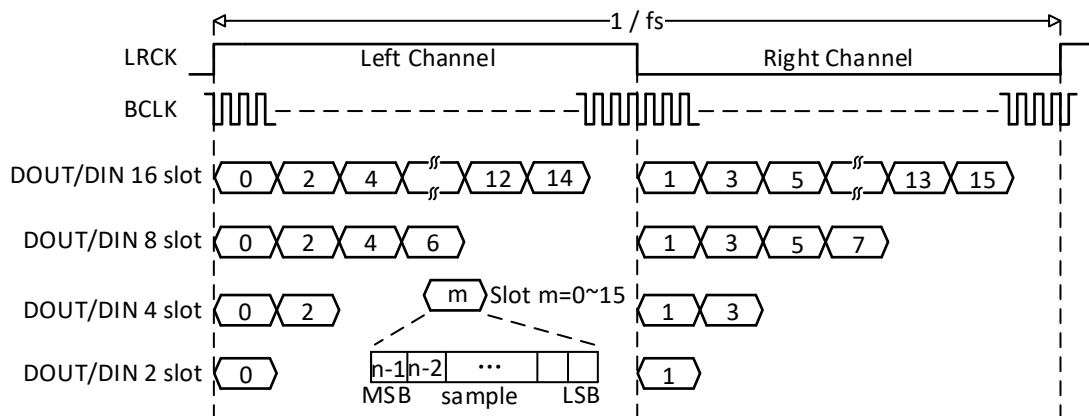


Figure 14-18 Right-Justified Mode Timing

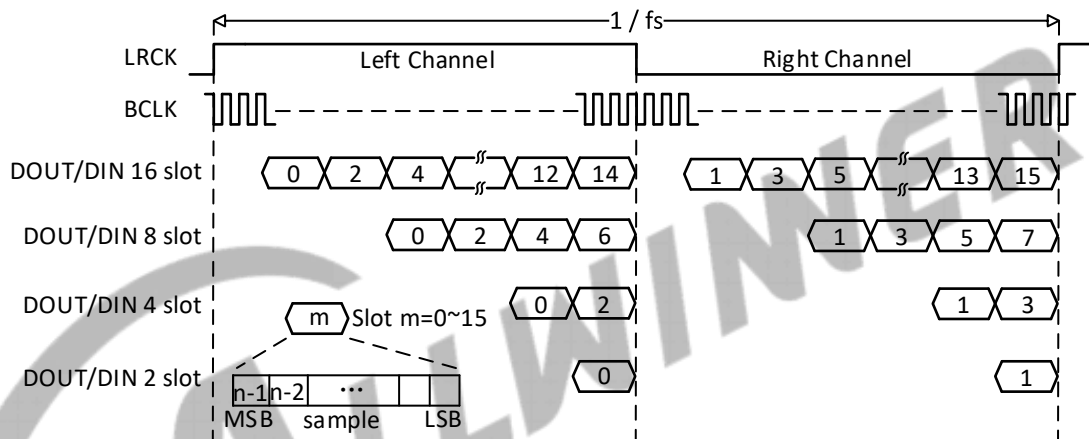


Figure 14-19 PCM Long Frame Mode Timing

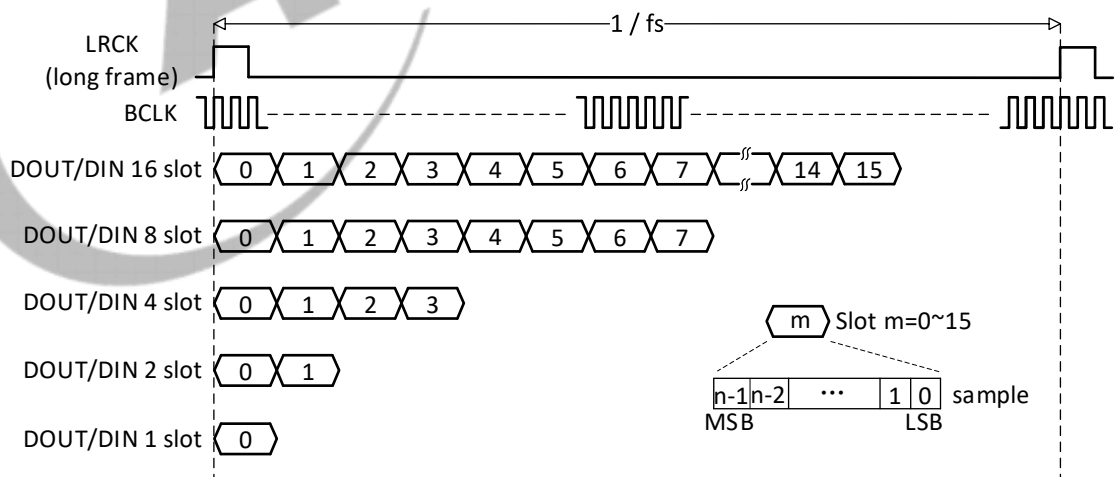
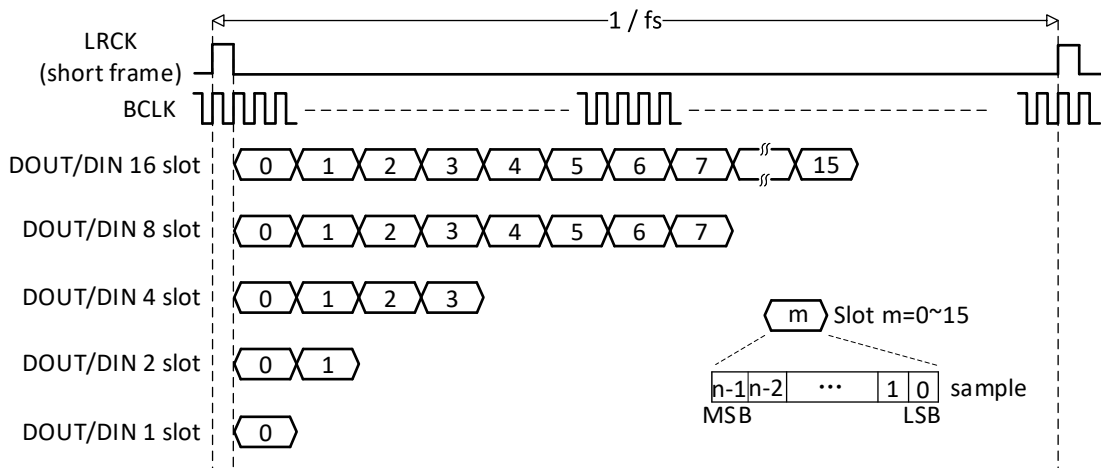


Figure 14-20 PCM Short Frame Mode Timing

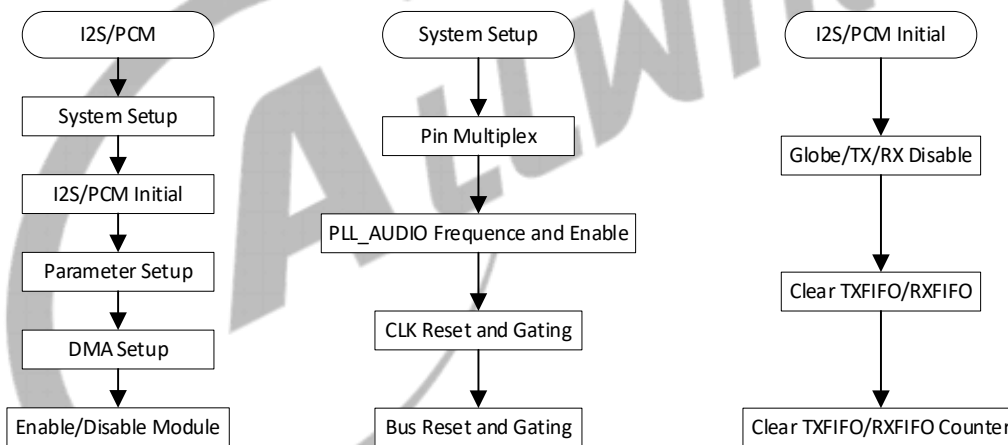


14.2.5 Programming Guidelines

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 14-21 I2S/PCM Operation Flow



14.2.5.1 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. Firstly, disable the PLL_AUDIO through PLL_AUDIOx Control Register[PLL_EN] in the CCU. Secondly, set up the frequency of the PLL_AUDIO in the PLL_AUDIOx Control Register. After that, enable the I2S/PCM gating through the I2Sx_CLK_REG when you check out that the PLL_AUDIOx Control Register[LOCK] becomes to 1. At last, reset and enable the I2S/PCM bus gating by setting [I2S_BGR_REG](#).

After the system setup, the register of I2S/PCM can be setup. Firstly, initialize the I2S/PCM. You should close the Globe Enable bit ([I2S/PCM_CTL\[0\]](#)), Transmitter Block Enable bit ([I2S/PCM_CTL\[2\]](#)), and Receiver Block Enable bit ([I2S/PCM_CTL\[1\]](#)) by writing 0. After that, clear

the TX/RX FIFO by writing 0 to the bit[25:24] of [I2S/PCM_FCTL](#). At last, you can clear the TX FIFO and RX FIFO counter by writing 0 to [I2S/PCM_TXCNT](#) and [I2S/PCM_RXCNT](#).

14.2.5.2 Channel Setup and DMA Setup

First, you can set up the I2S/PCM of master and slave. The configuration can be referred to the protocol of I2S/PCM. Then, you can set up the translation mode, the sample resolution, the wide of the slot, the channel slot number, and the trigger level, and so on. The setup of the 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 Section 11 DMA. In this module, you just enable the DRQ.

14.2.5.3 Enabling and Disabling I2S/PCM

Step 1 Enable TX/RX by writing [I2S/PCM_CTL\[TXEN\]](#)/[I2S/PCM_CTL\[RXEN\]](#).

Step 2 Enable I2S/PCM by writing 1 to [I2S/PCM_CTL\[Globe Enable\]](#).

Step 3 Write 0 to the Globe Enable bit to disable I2S/PCM.

14.2.6 Register List

Module Name	Base Address
I2S0	0x02032000

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/PCM_ISTA	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_TX0CHMAP0	0x0044	I2S/PCM TX0 Channel Mapping Register0
I2S/PCM_TX0CHMAP1	0x0048	I2S/PCM TX0 Channel Mapping Register1
I2S/PCM_RXCHSEL	0x0064	I2S/PCM RX Channel Select Register
I2S/PCM_RXCHMAP0	0x0068	I2S/PCM RX Channel Mapping Register0

Register Name	Offset	Description
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

14.2.7 Register Description

14.2.7.1 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 Only if RX_SYNC_EN set 1,RX_SYNC_EN_START can take effect. Audiocodect /I2S0 RX Synchronize Enable Start. 0: Disable 1: Enable
20	R/W	0x0	RX_SYNC_EN I2S RX Synchronize Enable. 0: Disable 1: Enable
19	/	/	/
18	R/W	0x1	BCLK_OUT Bit Clock Direction Select 0: Input 1: Output
17	R/W	0x1	LRCK_OUT LR Clock Direction Select 0: Input 1: Output
16:12	/	/	/
11	R/W	0x0	DOUT3_EN Data3 Output Enable 0: Disable, Hi-Z State 1: Enable
10	R/W	0x0	DOUT2_EN Data2 Output Enable 0: Disable, Hi-Z State 1: Enable

Offset: 0x0000			Register Name: I2S/PCM_CTL
Bit	Read/Write	Default/Hex	Description
9	R/W	0x0	DOUT1_EN Data1 Output Enable 0: Disable, Hi-Z State 1: Enable
8	R/W	0x0	DOUT0_EN Data0 Output Enable 0: Disable, Hi-Z State 1: Enable
7	/	/	/
6	R/W	0x0	DOUT_MUTE_EN Data Output Mute Enable 0: Normal Transfer 1: Force DOUT to Output 0
5:4	R/W	0x0	MODE_SEL Mode Selection 0: PCM Mode (offset 0: Long Frame; offset 1: Short Frame) 1: Left Mode (offset 0: LJ Mode; offset 1: I2S Mode) 2: Right-Justified Mode 3: Reserved
3	R/W	0x0	LOOP Loop Back Test 0: Normal Mode 1: Loop Back Test When Set '1', Connecting the DOUT with the DIN.
2	R/W	0x0	TXEN Transmitter Block Enable 0: Disable 1: Enable
1	R/W	0x0	RXEN Receiver Block Enable 0: Disable 1: Enable
0	R/W	0x0	GEN Globe Enable A disable on this bit overrides any other block or channel enables. 0: Disable 1: Enable

14.2.7.2 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 (Only Apply in PCM Mode) LRCK Width 0: LRCK = 1 BCLK Width (Short Frame) 1: LRCK = 2 BCLK Width (Long Frame)
29:20	/	/	/
19	R/W	0x0	LRCK_POLARITY When Apply in I2S / Left-Justified / Right-Justified Mode: 0: Left Channel When LRCK is Low 1: Left Channel When LRCK is High When Apply 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 sample frame. This value is interpreted as follow: PCM Mode: Number of BCLKs within (Left + Right) channel width. I2S / Left-Justified / Right-Justified Mode: Number of BCLKs within each individual channel width (Left or Right) . N+1 For example: N = 7: 8 BCLKs width ... N = 1023: 1024 BCLKs width
7	R/W	0x0	BCLK_POLARITY BCLK Polarity Select 0: Normal Mode, DOUT Drive Data at Negative Edge 1: Invert Mode, DOUT Drive Data at Positive Edge

Offset: 0x0004			Register Name: I2S/PCM_FMT0
Bit	Read/Write	Default/Hex	Description
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
3	R/W	0x0	EDGE_TRANSFER Edge Transfer 0: DOUT Drive Data and DIN Sample Data at the Different BCLK Edge 1: DOUT Drive Data and DIN Sample Data at the Same BCLK Edge 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.
2:0	R/W	0x3	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

14.2.7.3 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	/	/	/

Offset: 0x0008			Register Name: I2S/PCM_FMT1
Bit	Read/Write	Default/Hex	Description
7	R/W	0x0	RX_MLS MSB / LSB First Select 0: MSB First 1: LSB First
6	R/W	0x0	TX_MLS MSB / LSB First Select 0: MSB First 1: LSB First
5:4	R/W	0x3	SEXT Sign Extend in Slot [Sample Resolution < Slot Width] 00: Zeros or Audio Gain Padding at LSB Position 01: Sign Extension at MSB Position 10: Reserved 11: Transfer 0 after each Sample in each Slot
3:2	R/W	0x0	RX_PDM PCM Data Mode 00: Linear PCM 01: Reserved 10: 8-bits u-law 11: 8-bits A-law
1:0	R/W	0x0	TX_PDM PCM Data Mode 00: Linear PCM 01: Reserved 10: 8-bits u-law 11: 8-bits A-law

14.2.7.4 0x000C I2S/PCM Interrupt Status Register (Default Value: 0x0000_0010)

Offset: 0x000C			Register Name: I2S/PCM_ISTA
Bit	Read/Write	Default/Hex	Description
31:7	/	/	/
6	R/W1C	0x0	TXU_INT TXFIFO Underrun Pending Interrupt 0: No Pending Interrupt 1: TXFIFO Underrun Pending Interrupt Write '1' to clear this interrupt.

Offset: 0x000C			Register Name: I2S/PCM_ISTA
Bit	Read/Write	Default/Hex	Description
5	R/W1C	0x0	TXO_INT TXFIFO Overrun Pending Interrupt 0: No Pending Interrupt 1: TXFIFO Overrun Pending Interrupt Write '1' to clear this interrupt.
4	R	0x1	TXE_INT TXFIFO Empty Pending Interrupt 0: No Pending IRQ 1: TXFIFO Empty Pending Interrupt When Data in TXFIFO are Less than TX Trigger Level
3	/	/	/
2	R/W1C	0x0	R XU_INT RXFIFO Underrun Pending Interrupt 0: No Pending Interrupt 1: RXFIFO Underrun Pending Interrupt Write '1' to clear this interrupt.
1	R/W1C	0x0	R XO_INT RXFIFO Overrun Pending Interrupt 0: No Pending IRQ 1: RXFIFO Overrun Pending IRQ Write '1' to clear this interrupt.
0	R	0x0	R XA_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

14.2.7.5 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 Host can get one sample by reading this register. The left channel sample data is first and then the right channel sample.

14.2.7.6 0x0014 I2S/PCM FIFO Control Register (Default Value: 0x0004_00F0)

Offset: 0x0014			Register Name: I2S/PCM_FCTL
Bit	Read/Write	Default/Hex	Description

Offset: 0x0014			Register Name: I2S/PCM_FCTL
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	HUB_EN Audio Hub Enable Only if TXEN set 1,HUB_EN can take effect. Audiocodec / I2S0 OWA TXFIFO Hub Enable. 0 : Disable 1 : Enable
30:26	/	/	/
25	R/WAC	0x0	FTX Flush TX FIFO Write '1' to flush TXFIFO, self clear to '0'.
24	R/WAC	0x0	FRX Flush RX FIFO Write '1' to flush RXFIFO, self clear to '0'.
23:19	/	/	/
18:12	R/W	0x40	TXTL TXFIFO Empty Trigger Level Interrupt and DMA request trigger level for TXFIFO normal condition. Trigger Level = TXTL
11:10	/	/	/
9:4	R/W	0xF	RXTL RXFIFO Trigger Level Interrupt and DMA request trigger level for RXFIFO normal condition. Trigger Level = RXTL + 1
3	/	/	/
2	R/W	0x0	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-bits 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}

Offset: 0x0014			Register Name: I2S/PCM_FCTL
Bit	Read/Write	Default/Hex	Description
1:0	R/W	0x0	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-bits 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]}

14.2.7.7 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	/	/	/
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

Offset: 0x0018			Register Name: I2S/PCM_FSTA
Bit	Read/Write	Default/Hex	Description
6:0	R	0x0	RXA_CNT RXFIFO Available Sample Word Counter

14.2.7.8 0x001C I2S/PCM DMA And 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: Disable 1: Enable
6	R/W	0x0	TXUI_EN TXFIFO Underrun Interrupt Enable 0: Disable 1: Enable
5	R/W	0x0	TXOI_EN TXFIFO Overrun Interrupt Enable 0: Disable 1: Enable 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: Disable 1: Enable
3	R/W	0x0	RX_DRQ RXFIFO Data Available DRQ Enable 0: Disable 1: Enable When set to '1', RXFIFO DMA request line is asserted if data is available in RXFIFO.
2	R/W	0x0	RXUI_EN RXFIFO Underrun Interrupt Enable 0: Disable 1: Enable
1	R/W	0x0	RXOI_EN RXFIFO Overrun Interrupt Enable 0: Disable 1: Enable

Offset: 0x001C			Register Name: I2S/PCM_INT
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	RXAI_EN RXFIFO Data Available Interrupt Enable 0: Disable 1: Enable

14.2.7.9 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	TX_DATA TX Sample Transmitting left, right channel sample data should be written this register one by one. The left channel sample data is first and then the right channel sample.

14.2.7.10 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	MCLKO_EN MCLK Out Enable 0: Disable MCLK Output 1: Enable MCLK Output Notes: Whether in Slave or Master mode, when this bit is set to '1', MCLK should be output.

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

14.2.7.11 0x0028 I2S/PCM TX Sample Counter Register(Default Value: 0x0000_0000)

Offset: 0x0028			Register Name: I2S/PCM_TXCNT
Bit	Read/Write	Default/Hex	Description

Offset: 0x0028			Register Name: I2S/PCM_TXCNT
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 been updated by the initial value, the counter register should count on base of this initial value.

14.2.7.12 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	RX_CNT RX Sample Counter 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 been updated by the initial value, the counter register should count on base of this initial value.

14.2.7.13 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	TX_SLOT_HIZ TX Slot Value in Half Cycle of BCLK 0: Normal Mode for the Last Half Cycle of BCLK in the Slot 1: Turn to Hi-Z State for the Last Half Cycle of BCLK in the Slot

Offset: 0x0030			Register Name: I2S/PCM_CHCFG
Bit	Read/Write	Default/Hex	Description
8	R/W	0x0	TX_STATE The state of transmission line When there is No Transmission 0: Transfer Level 0 When Not Transferring Slot 1: Turn to Hi-Z State (TDM) When Not Transferring Slot
7:4	R/W	0x0	RX_SLOT_NUM RX Channel/Slot Number Which between CPU/DMA and RXFIFO 0: 1 Channel or Slot ... 7: 8 Channels or Slots 8: 9 Channels or Slots ... 15:16 Channels or Slots
3:0	R/W	0x0	TX_SLOT_NUM TX Channel/Slot Number Which between CPU/DMA and TXFIFO 0: 1 Channel or Slot ... 7: 8 Channels or Slots 8: 9 Channels or Slots ... 15:16 Channels or Slots

14.2.7.14 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	TX0_OFFSET TX0 offset Tune, TX0 Data offset to LRCK 0: No offset n: Data is offset by n BCLKs to LRCK

Offset: 0x0034			Register Name: I2S/PCM_TX0CHSEL
Bit	Read/Write	Default/Hex	Description
19:16	R/W	0x0	TX0_CHSEL TX0 Channel (Slot) Number Select for each Output 0: 1 Channel / Slot ... 7: 8 Channels / Slots 8: 9 Channels / Slots ... 15: 16 Channels / Slots
15:0	R/W	0x0	TX0_CHEN TX0 Channel (Slot) Enable, bit[15:0] Refer to Slot [15:0]. When One or More Slot(s) is(are) Disabled, the Affected Slot(s) is(are) Set to Disable State. 0: Disable 1: Enable

14.2.7.15 0x0044 I2S/PCM TX0 Channel Mapping Register0(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 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
27:24	R/W	0x0	TX0_CH14_MAP TX0 Channel 14 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th sample

Offset: 0x0044			Register Name: I2S/PCM_TX0CHMAP0
Bit	Read/Write	Default/Hex	Description
23:20	R/W	0x0	TX0_CH13_MAP TX0 Channel 13 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
19:16	R/W	0x0	TX0_CH12_MAP TX0 Channel 12 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
15:12	R/W	0x0	TX0_CH11_MAP TX0 Channel 11 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th sample
11:8	R/W	0x0	TX0_CH10_MAP TX0 Channel 10 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
7:4	R/W	0x0	TX0_CH9_MAP TX0 Channel 9 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample

Offset: 0x0044			Register Name: I2S/PCM_TX0CHMAP0
Bit	Read/Write	Default/Hex	Description
3:0	R/W	0x0	TX0_CH8_MAP TX0 Channel 8 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample

14.2.7.16 0x0048 I2S/PCM TX0 Channel Mapping Register1(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 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
27:24	R/W	0x0	TX0_CH6_MAP TX0 Channel 6 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
23:20	R/W	0x0	TX0_CH5_MAP TX0 Channel 5 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample

Offset: 0x0048			Register Name: I2S/PCM_TX0CHMAP1
Bit	Read/Write	Default/Hex	Description
19:16	R/W	0x0	TX0_CH4_MAP TX0 Channel 4 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
15:12	R/W	0x0	TX0_CH3_MAP TX0 Channel 3 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
11:8	R/W	0x0	TX0_CH2_MAP TX0 Channel 2 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
7:4	R/W	0x0	TX0_CH1_MAP TX0 Channel 1 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
3:0	R/W	0x0	TX0_CH0_MAP TX0 Channel 0 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample

14.2.7.17 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 0: 1 Channel / Slot ... 7: 8 Channels / Slots 8: 9 Channels / Slots ... 15: 16 Channels / Slots
15:0	/	/	/

14.2.7.18 0x0068 I2S/PCM RX Channel Mapping Register0(Default Value: 0x0000_0000)

Offset: 0x0068			Register Name: I2S/PCM_RXCHMAP0
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:24	R/W	0x0	RX_CH15_MAP RX Channel 15 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
23:20	/	/	/
19:16	R/W	0x0	RX_CH14_MAP RX Channel 14 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
15:12	/	/	/

Offset: 0x0068			Register Name: I2S/PCM_RXCHMAP0
Bit	Read/Write	Default/Hex	Description
11:8	R/W	0x0	RX_CH13_MAP RX Channel 13 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
7:4	/	/	/
3:0	R/W	0x0	RX_CH12_MAP RX Channel 12 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample

14.2.7.19 0x006C I2S/PCM RX Channel Mapping Register1(Default Value: 0x0000_0000)

Offset: 0x006C			Register Name: I2S/PCM_RXCHMAP1
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:24	R/W	0x0	RX_CH11_MAP RX Channel 11 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
23:20	/	/	/
19:16	R/W	0x0	RX_CH10_MAP RX Channel 10 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
15:12	/	/	/

Offset: 0x006C			Register Name: I2S/PCM_RXCHMAP1
Bit	Read/Write	Default/Hex	Description
11:8	R/W	0x0	RX_CH9_MAP RX Channel 9 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
7:4	/	/	/
3:0	R/W	0x0	RX_CH8_MAP RX Channel 8 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample

14.2.7.20 0x0070 I2S/PCM RX Channel Mapping Register2(Default Value: 0x0000_0000)

Offset: 0x0070			Register Name: I2S/PCM_RXCHMAP2
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:24	R/W	0x0	RX_CH7_MAP RX Channel 7 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
23:20	/	/	/
19:16	R/W	0x0	RX_CH6_MAP RX Channel 6 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
15:12	/	/	/

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 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
7:4	/	/	/
3:0	R/W	0x0	RX_CH4_MAP RX Channel 4 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample

14.2.7.21 0x0074 I2S/PCM RX Channel Mapping Register3(Default Value: 0x0000_0000)

Offset: 0x0074			Register Name: I2S/PCM_RXCHMAP3
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:24	R/W	0x0	RX_CH3_MAP RX Channel 3 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
23:20	/	/	/
19:16	R/W	0x0	RX_CH2_MAP RX Channel 2 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
15:12	/	/	/

Offset: 0x0074			Register Name: I2S/PCM_RXCHMAP3
Bit	Read/Write	Default/Hex	Description
11:8	R/W	0x0	RX_CH1_MAP RX Channel 1 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample
7:4	/	/	/
3:0	R/W	0x0	RX_CH0_MAP RX Channel 0 Mapping 0: 1st Sample ... 7: 8th Sample 8: 9th Sample ... 15: 16th Sample

14.3 One Wire Audio (OWA)

14.3.1 Overview

The One Wire Audio (OWA) provides a serial bus interface for audio data. This interface is widely used for consumer audio connection.

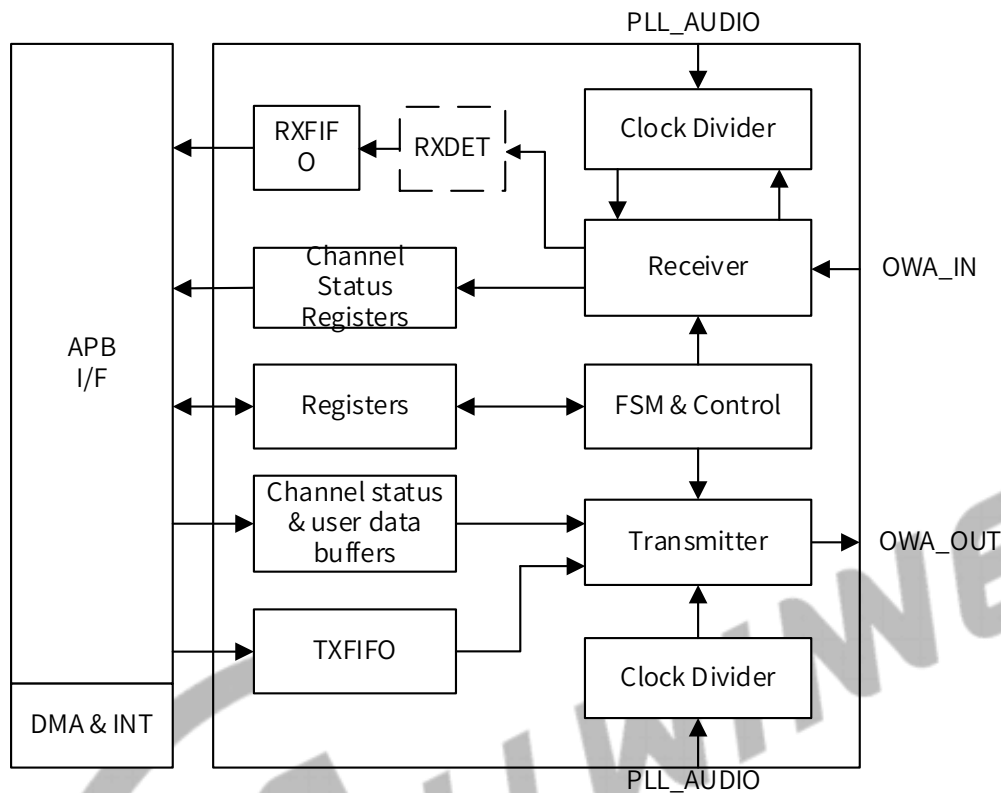
The OWA includes the following features:

- One OWA TX
- Compatible with standard IEC-60958 and IEC-61937
 - IEC-60958 supports data formats: 16 bits, 20 bits, and 24 bits
- TXFIFO and RXFIFO
 - One 128×24bits TXFIFO for audio data transfer
 - Programmable FIFO thresholds
- Supports TX DMA slave interface
- Multiple function clock
 - Separate clock for OWA TX
 - OWA TX function clock includes a series of 24.576MHz/49.152MHz and 22.579MHz frequency (TX function clock 49.152MHz support ample rate 384KHz)
- Supports hardware parity on TX
 - Hardware parity generation on the transmitter
- Supports channel status insertion for the transmitter

14.3.2 Block Diagram

The following figure shows the OWA block diagram.

Figure 14-22 OWA Block Diagram



14.3.3 External Signals

The OWA is a Biphasic-Mark Encoding Digital Audio Transfer protocol. In this protocol, the CLK signal and data signal are transferred in the same line. The following table describes the external signals of OWA. OWA-OUT is the output pin for the output CLK and DATA.

Table 14-3 OWA External Signals

Signal Name	Description	Type
OWA-OUT	One Wire Audio Output	O

14.3.4 Functional Description

14.3.4.1 Biphasic-Mark Code (BMC)

In the OWA format, the digital signal is coded using the biphasic-mark code (BMC). The clock, frame, and data are embedded in only one signal—the data pin. In the BMC system, each data bit is encoded into two logical states (00, 01, 10, or 11) at the pin. The following figure and table show how data is encoded to the BMC format.

The frequency of the clock is twice the data bit rate, as shown in the following figure. Also, the clock is always programmed to $128 \times f_s$, where f_s is the sample rate. The device receiving in the OWA format can recover the clock and frame information from the BMC signal.

Figure 14-23 OWA Biphase-Mark Code

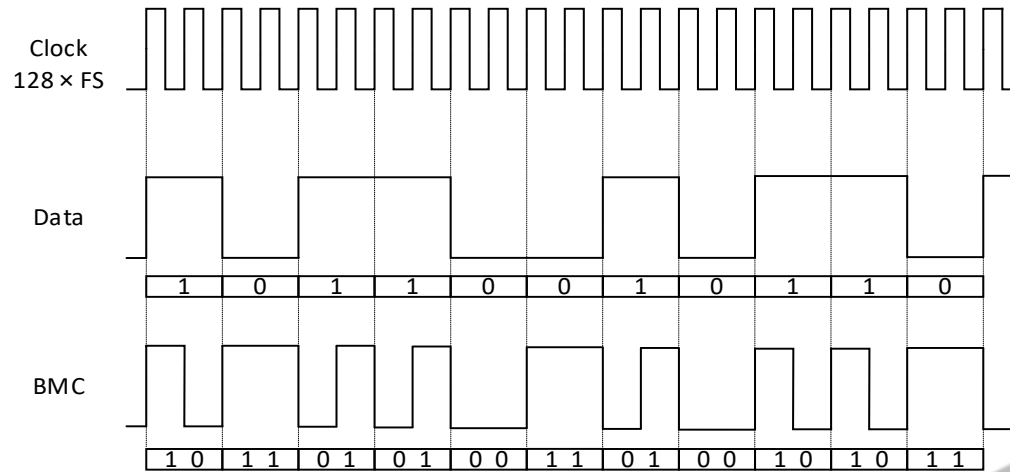


Table 14-4 Biphase-Mark Encoder

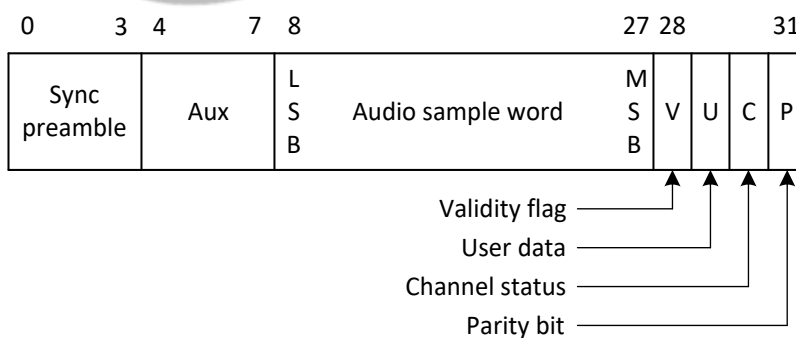
Data	Previous State	BMC
0	0	11
0	1	00
1	0	10
1	1	01

14.3.4.2 IEC60958 Transmit Format

The OWA supports digital audio data transfer and receive. It also supports full-duplex synchronous work mode. The software can set the work mode by the OWA Control Register.

Every audio sample transmitted in a sub-frame consists of 32-bit, numbered from 0 to 31. The following figure shows a sub-frame.

Figure 14-24 OWA Sub-Frame Format



Bits 0-3 carry one of the four permitted preambles to signify the type of audio sample in the current sub-frame. The preamble is not encoded in BMC format, and therefore the preamble code can contain more than two consecutive 0 or 1 logical states in a row.

Bits 4-27 carry the audio sample word in linear 2s-complement representation. The most-significant bit (MSB) is carried by bit 27. When a 24-bit coding range is used, the least-significant bit (LSB) is in bit 4. When a 20-bit coding range is used, bit [8:27] carry the audio sample word with the LSB in bit 8. Bit [4:7] may be used for other applications and are designated auxiliary sample bits.

If the source provides fewer bits than the interface allows (either 20 or 24), the unused LSBs are set to logical 0. For a nonlinear PCM audio application or a data application, the main data field may carry any other information.

Bit 28 carries the validity bit (V) associated with the main data field in the sub-frame.

Bit 29 carries the user data channel (U) associated with the main data field in the sub-frame.

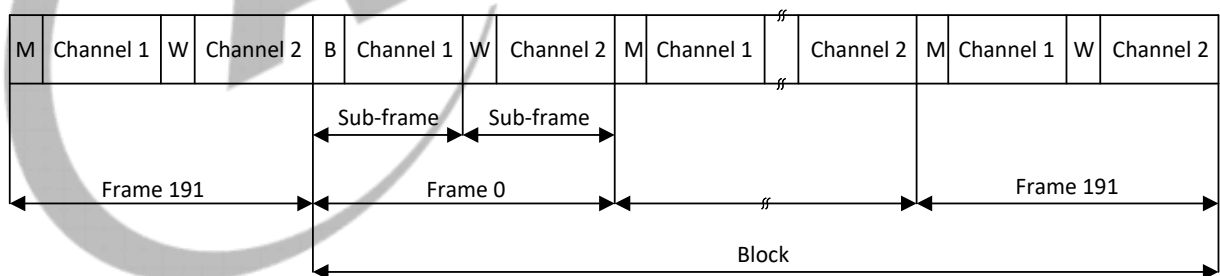
Bit 30 carries the channel status information (C) associated with the main data field in the sub-frame. The channel status indicates if the data in the sub-frame is a digital audio or some other type of data.

Bit 31 carries a parity bit (P) such that bit 4-31 carry an even number of 1s and an even number of 0s (even parity). As shown in the following table, the preambles (bit 0-3) are also defined with even parity.

Table 14-5 Preamble Codes

Preamble Code	Previous Logical State	Logical State	Description
B (or Z)	0	1110 1000	Start of a block and sub-frame 1
M (or X)	0	1110 0010	Sub-frame 1
W (or Y)	0	1110 0100	Sub-frame 2

Figure 14-25 OWA Frame/Block Format



14.3.4.3 IEC61937 Transmit Format

IEC 61937 applies to the digital audio interface by using the IEC 60958 series for the conveying of non-linear PCM encoded audio bitstreams. The non-linear PCM encoded audio bitstream is transferred by using the basic 16-bit data area of the IEC 60958 subframes, i.e. in time-slots 12 to 27. Because the non-linear PCM encoded audio bitstream to be transported is at a lower data rate than that supported by the IEC 60958 interface, the audio bitstream is broken into a sequence of discrete data-bursts, and stuffing between the data-bursts is necessary.

IEC 60958 Data Burst

The method of placing the data into the IEC 60958 bitstream is to format the data to be transmitted into data-bursts and to send each data-burst in a continuous sequence of IEC 60958 frames.

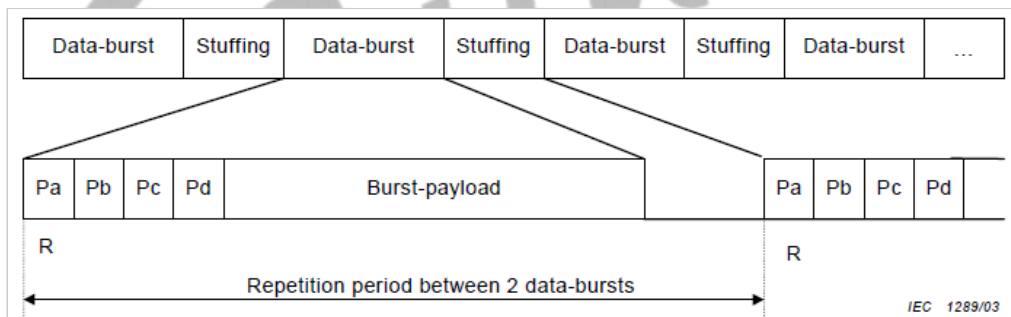
Table 14-6 Bit Allocation of Data-Burst in IEC 60958 Subframes

Subframe	Bit of subframes				
	MSB b27	b26	b25 b14	b13	LSB b12
Frame 0; subframe B or M	0	1		14	15
Frame 0; subframe W	16	17		30	31
Frame 1; subframe B or M	32	33		46	47
Frame 1; subframe W	48	49		62	63
Frame 2; subframe B or M	64	65		78	79
-----			-----		
Last subframe B or M of data-burst	n-32	n-31		n-18	n-17
Last subframe W of data-burst	n-16	n-15		n-2	n-1

Data Burst Format

Each data-burst contains a burst-preamble consisting of four 16-bit words (Pa, Pb, Pc and Pd) followed by the burst-payload which contains data of an encoded audio frame.

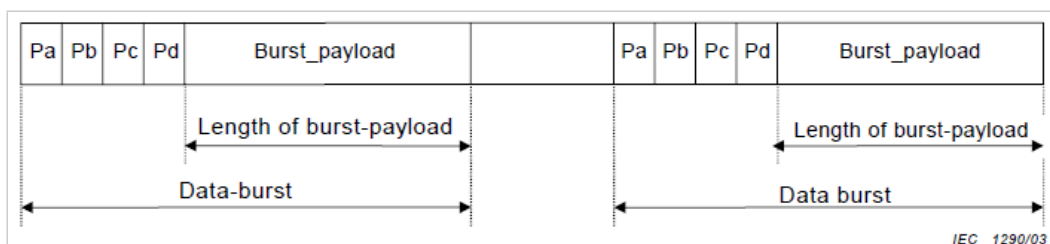
Figure 14-26 Data-Burst Format



- Burst-preamble

The burst-preamble consists of four mandatory fields. Pa and Pb represent a synchronization word. Pc gives information about the type of data, and some information/control for the receiver. Pd gives the length of the burst-payload, and is limited to 65535 bits in the case of Pd represent bits' length, or is limited to 65535 bytes in the case of Pd represent bytes' length.

Figure 14-27 Data-burst Preamble



The four preamble words are contained in two sequential IEC 60958 frames. The frame beginning the data-burst contains preamble word Pa in subframe 1, and Pb in subframe 2. The next frame contains Pc in subframe 1 and Pd in subframe 2. When placed into an IEC 60958 subframe, the MSB of a 16-bit burst-preamble word is placed into time-slot 27 and the LSB is placed into time-slot 12.

Figure 14-28 Data-burst Preamble words

Preamble word	Length of field	Contents	Value MSB..LSB
Pa	16-bit	Sync word 1	F872h
Pb	16-bit	Sync word 2	4E1Fh
Pc	16-bit	Burst-info	Table 5
Pd	16-bit	Length-code	Number of bits or number of bytes according to data-type

- Burst-information

The 16-bit burst-information contains information about the data which will be found in the data-burst.

Figure 14-29 Fields of Burst-information

Bits of Pc	Value	Contents	Remark
0 – 6		Data-type	See IEC 61937-2
7	0	Error-flag indicating a valid burst-payload	
	1	Error-flag indicating that the burst-payload may contain errors	
8 – 12		Data-type-dependent info	
13 – 15	0	Bitstream-number	

NOTE The repetition period of pause data-bursts depends on the application in which IEC 60958 is used to convey encoded audio bitstreams.

The 7-bit data-type is defined in bits 0-6 of the burst-preamble Pc, the bit 6 is the MSB. This data-type field indicates the format of the burst-payload, which will be conveyed in the data-burst. Typical properties of a data-type are the reference point and repetition period of the burst, which is the number of sampling periods of the audio between the reference point of the current data-burst and the reference point of the next data-burst. The reference point is inherently defined for each data-type.

The error-flag bit is available to indicate if the contents of the data-burst contain data errors. If a data-burst is thought to be error-free, or if the data source does not know if the data contains errors, then the value of this bit is set to a '0'. If the data source does know that a particular data-burst contains some errors this bit may be set to a '1'. The usage of this bit by receiver is optional.

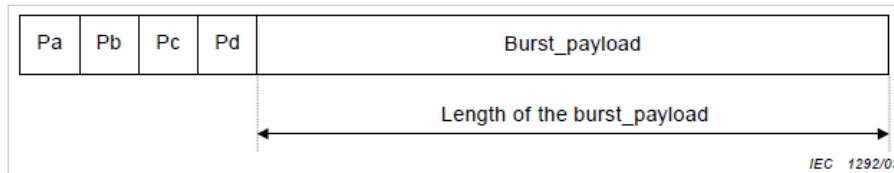
The meaning of the 5-bit data-type-dependent info depends on the value of the data-type.

The 3-bit bitstream-number indicates to which bitstream the data-burst belongs. Eight codes (0-7) are available so that up to eight independent bitstreams may be multiplexed in one bitstream in a time multiplex. Each independent bitstream shall use a unique bit-streamnumber.

- Length-code

The length-code indicates the number of bits or bytes according to data-type within the databurst, from 0 to 65535. The size of the Pa, Pb, Pc and Pd is not counted in the value of the length-code. In other words, the length-code indicates the number of bits of the burst-payload in bits, plus the conditional length of Pe and Pf, or the number of bytes of the burst-payload in bytes, plus the conditional length of Pe and Pf if exists.

Figure 14-30 Length of the Burst-Payload Specified by Pd



14.3.4.4 Audio Sample Ratio Detection

The sampling rate is calculated according to the data pulse back-stepping method. In the first phase lock of the CDR, find the period of 1 Frame. Then count by using the high-speed sampling clock and read the counting value of the pulse width of Period Max, then the sampling rate can be calculated.

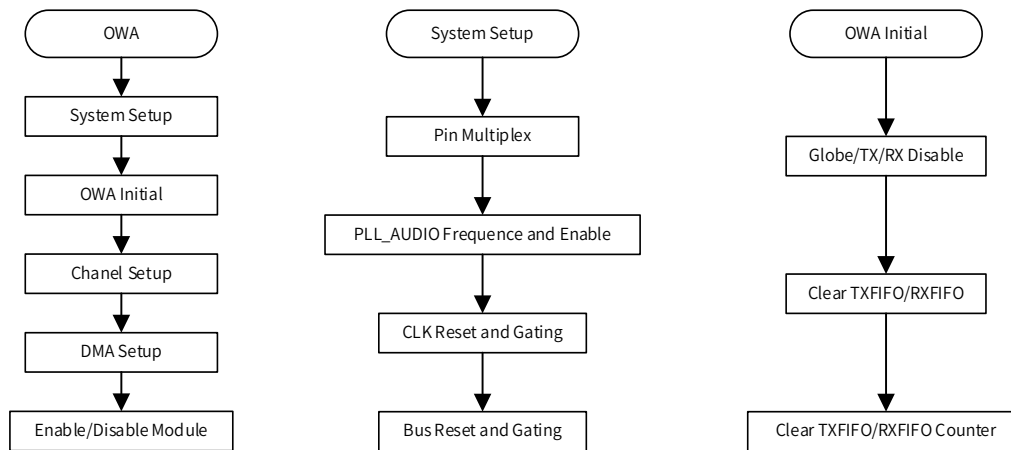
Table 14-7 The Relation between Different System Clock and Sample Ratio

TX Sample Rate(kHz)	Sample Clock Cycles	
	196.608 MHz-SysClk	200 MHz-SysClk
22.05	8916(±5)	9070(±5)
24	8192(±5)	8333(±5)
32	6144(±5)	6250(±5)
44.1	4458(±5)	4535(±5)
48	4096(±5)	4166(±5)
96	2048(±5)	2083(±5)
176.4	1114(±5)	1133(±5)
192	1024(±5)	1041(±5)

14.3.5 Programming Guidelines

The software operation of the OWA is divided into five steps: system setup, OWA initialization, channel setup, DMA setup and enable/disable module. The following sections describe these five steps.

Figure 14-31 OWA Operation Flow



14.3.5.1 System Setup and OWA Initialization

The first step in the OWA initialization is properly programming the GPIO because the OWA port is a multiplex pin. You can find the function in Section 15.7 GPIO.

The clock source for the OWA should be followed. Firstly, reset the audio PLL in PLL_AUDIOx Control Register. Secondly, set up the frequency of the Audio PLL in the PLL_AUDIOx Control Register. After that, enable the OWA gating. Lastly, enable the OWA bus gating.

After the system setup, the register of OWA can be set up. Firstly, reset the OWA by writing 1 to [OWA_GENCTL\[0\]](#) and clear the TX FIFO by writing 1 to [OWA_FCTL\[17:16\]](#). After that, enable the globe enable bit by writing 1 to [d\[1\]](#) and clear the interrupt and TX counter by setting [OWA_ISTA](#) and [OWA_TX_CNT](#).

14.3.5.2 Parameter Setup and DMA Setup

You can set up the audio type, clock divider ratio, the sample format, and the trigger level, and so on. The setup of the register can be found in the specification.

The OWA supports two methods to transfer the data. The most common way is DMA; the configuration of DMA can be found in Section 11 DMA. In this module, you just enable the DRQ in [OWA_INT\[7\]](#).

14.3.5.3 Enabling and Disabling OWA

Step 1 Enable TX by writing [OWA_TXCFG\[31\]](#).

Step 2 Enable OWA by writing 1 to [OWA_GENCTL\[1\]](#).

Step 3 Write 0 to [OWA_GENCTL\[1\]](#) to disable process.

14.3.6 Register List

Module Name	Base Address

Module Name	Base Address
OWA	0x02036000

Register Name	Offset	Description
OWA_GENCTL	0x0000	OWA General Control Register
OWA_TXCFG	0x0004	OWA TX Configuration Register
OWA_ISTA	0x000C	OWA Interrupt Status Register
OWA_FCTL	0x0014	OWA FIFO Control Register
OWA_FSTA	0x0018	OWA FIFO Status Register
OWA_INT	0x001C	OWA Interrupt Control Register
OWA_TXFIFO	0x0020	OWA TX FIFO Register
OWA_TX_CNT	0x0024	OWA TX Counter Register
OWA_TX_CHSTA0	0x002C	OWA TX Channel Status Register0
OWA_TX_CHSTA1	0x0030	OWA TX Channel Status Register1

14.3.7 Register Description

14.3.7.1 0x0000 OWA General Control Register (Default Value: 0x0000_0080)

Offset: 0x0000			Register Name: OWA_GENCTL
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R	0x1	Reserved
6:2	/	/	/
1	R/W	0x0	GEN Global Enable A disable on this bit overrides any other block or channel enables and flushes all FIFOs. 0: Disable 1: Enable
0	R/WAC	0x0	RST_TX Reset TX 0: Normal 1: Reset Self-clear to 0.

14.3.7.2 0x0004 OWA TX Configuration Register (Default Value: 0x0000_00F0)

Offset: 0x0004			Register Name: OWA_TXCFG
Bit	Read/Write	Default/Hex	Description

Offset: 0x0004			Register Name: OWA_TXCFG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	TX_SINGLE_MODE TX Single Channel Mode 0: Disable 1: Enable
30:18	/	/	/
17	R/W	0x0	ASS Audio Sample Select when TX FIFO Underrun 0: Sending 0 1: Sending the last audio Note: This bit is only valid in PCM mode.
16	R/W	0x0	TX_AUDIO TX Data Type 0: Linear PCM (Valid bit of both sub-frame set to 0) 1: Non-audio (Valid bit of both sub-frame set to 1)
15:9	/	/	/
8:4	R/W	0xF	TX_RATIO TX Clock Divide Ratio Clock divide ratio = TX_RATIO + 1 $F_s = PLL\_AUDIO / [(TX\_RATIO + 1) * 64 * 2]$
3:2	R/W	0x0	TX_SF TX Sample Format 00: 16 bits 01: 20 bits 10: 24 bits 11: Reserved
1	R/W	0x0	TX_CHM CHSTMODE 0: Channel status A and B set to 0 1: Channel status A and B generated from TX_CHSTA
0	R/W	0x0	TXEN TX Enable 0: Disable 1: Enable

14.3.7.3 0x000C OWA Interrupt Status Register (Default Value: 0x0000_0010)

Offset: 0x000C			Register Name: OWA_ISTA
Bit	Read/Write	Default/Hex	Description
31:7	/	/	/

Offset: 0x000C			Register Name: OWA_ISTA
Bit	Read/Write	Default/Hex	Description
6	R/W1C	0x0	TXU_INT TX FIFO Underrun Pending Interrupt 0: No Pending IRQ 1: TXFIFO Underrun Pending Interrupt Writing “1” to clear this interrupt.
5	R/W1C	0x0	TXO_INT TX FIFO Overrun Pending Interrupt 0: No Pending IRQ 1: TXFIFO Overrun Pending Interrupt Writing “1” to clear this interrupt.
4	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.
3:0	/	/	/

14.3.7.4 0x0014 OWA FIFO Control Register (Default Value: 0x0004_0200)

Offset: 0x0014			Register Name: OWA_FCTL
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	HUB_EN Audio Hub Enable The bit will take effect only when the TXEN is set to 1. Audio Codec/I2S0 OWA TXFIFO Hub Enable. 0: Disable 1: Enable
30	R/W1C	0x0	FTX Write ‘1’ to flush TXFIFO, self-clear to ‘0’.
29:20	/	/	/
19:12	R/W	0x40	TXTL TX FIFO Empty Trigger Level Interrupt and DMA request trigger level for TX FIFO normal condition. Trigger Level = TXTL
11:3	/	/	/

Offset: 0x0014			Register Name: OWA_FCTL
Bit	Read/Write	Default/Hex	Description
2	R/W	0x0	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 [23:0] = {APB_WDATA [31:12], 4' h0} Mode 1: TXFIFO[23:0] = {APB_WDATA[19:0], 4' h0}
1:0	/	/	/

14.3.7.5 0x0018 OWA FIFO Status Register (Default Value: 0x8080_0000)

Offset: 0x0018			Register Name: OWA_FSTA
Bit	Read/Write	Default/Hex	Description
31	R	0x1	TXE TXFIFO Empty (indicate the TXFIFO is not full) 0: No room for new sample in TXFIFO 1: More than one room for new sample in TXFIFO (>= 1 Word)
30:24	/	/	/
23:16	R	0x80	TXE_CNT TXFIFO Empty Space Word Counter
15:0	/	/	/

14.3.7.6 0x001C OWA Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x001C			Register Name: OWA_INT
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	TX_DRQ TXFIFO Empty DRQ Enable 0: Disable 1: Enable
6	R/W	0x0	TXUI_EN TXFIFO Underrun Interrupt Enable 0: Disable 1: Enable

Offset: 0x001C			Register Name: OWA_INT
Bit	Read/Write	Default/Hex	Description
5	R/W	0x0	TXOI_EN TXFIFO Overrun Interrupt Enable 0: Disable 1: Enable
4	R/W	0x0	TXEI_EN TXFIFO Empty Interrupt Enable 0: Disable 1: Enable
3:0	/	/	/

14.3.7.7 0x0020 OWA TX FIFO Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: OWA_TXFIFO
Bit	Read/Write	Default/Hex	Description
31:0	W	0x0	TX_DATA Transmitting A, B channel data should be written by this register one by one. The first is A channel data, and the latter is B channel data.

14.3.7.8 0x0024 OWA TX Counter Register (Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: OWA_TX_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 on the base of this initial value.

14.3.7.9 0x002C OWA TX Channel Status Register0 (Default Value: 0x0000_0000)

Offset: 0x002C			Register Name: OWA_TX_CHSTA0
Bit	Read/Write	Default/Hex	Description
31: 30	/	/	/

Offset: 0x002C			Register Name: OWA_TX_CHSTA0
Bit	Read/Write	Default/Hex	Description
29:28	R/W	0x0	CA Clock Accuracy 00: Level 2 01: Level 1 10: Level 3 11: Not matched
27:24	R/W	0x0	FREQ Sampling Frequency 0000: 44.1 kHz 0001: Not indicated 0010: 48 kHz 0011: 32 kHz 0100: 22.05 kHz 0101: Reserved 0110: 24 kHz 0111: Reserved 1000: Reserved 1001: 768 kHz 1010: 96 kHz 1011: Reserved 1100: 176.4 kHz 1101: Reserved 1110: 192 kHz 1111: Reserved
23:20	R/W	0x0	CN Channel Number
19:16	R/W	0x0	SN Source Number
15:8	R/W	0x0	CC Category Code Indicates the kind of equipment that generates the digital audio interface signal.
7:6	R/W	0x0	MODE Mode 00: Default Mode 01 to 11: Reserved

Offset: 0x002C			Register Name: OWA_TX_CHSTA0
Bit	Read/Write	Default/Hex	Description
5:3	R/W	0x0	EMP Emphasis Additional format information For bit 1 = "0", Linear PCM audio mode: 000: 2 audio channels without pre-emphasis 001: 2 audio channels with 50 μ s/15 μ s pre-emphasis 010: Reserved (for 2 audio channels with pre-emphasis) 011: Reserved (for 2 audio channels with pre-emphasis) 100 to 111: Reserved For bit 1 = "1", other than Linear PCM applications: 000: Default state 001 to 111: Reserved
2	R/W	0x0	CP Copyright 0: Copyright is asserted 1: No copyright is asserted
1	R/W	0x0	TYPE Audio Data Type 0: Linear PCM samples 1: Non-linear PCM audio
0	R/W	0x0	PRO Application Type 0: Consumer application 1: Professional application This bit must be fixed to "0".

14.3.7.10 0x0030 OWA TX Channel Status Register 1 (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: OWA_TX_CHSTA1
Bit	Read/Write	Default/Hex	Description
31:10	/	/	/
9:8	R/W	0x0	CGMS_A 00: Copying is permitted without restriction 01: One generation of copies may be made 10: Condition not be used 11: No copying is permitted

Offset: 0x0030			Register Name: OWA_TX_CHSTA1
Bit	Read/Write	Default/Hex	Description
7:4	R/W	0x0	ORIG_FREQ Original Sampling Frequency 0000: Not indicated 0001: 192 kHz 0010: 12 kHz 0011: 176.4 kHz 0100: Reserved 0101: 96 kHz 0110: 8 kHz 0111: 88.2 kHz 1000: 16 kHz 1001: 24 kHz 1010: 11.025 kHz 1011: 22.05 kHz 1100: 32 kHz 1101: 48 kHz 1110: Reserved 1111: 44.1 kHz
3:1	R/W	0x0	WL Sample Word Length For bit 0 = “0” : 000: Not indicated 001: 16 bits 010: 18 bits 100: 19 bits 101: 20 bits 110: 17 bits 111: Reserved For bit 0 = “1” : 000: Not indicated 001: 20 bits 010: 22 bits 100: 23 bits 101: 24 bits 110: 21 bits 111: Reserved
0	R/W	0x0	MWL Max Word Length 0: Maximum audio sample word length is 20 bits 1: Maximum audio sample word length is 24 bits

15 Peripherals

15.1 USB2.0 DRD (USB0)

15.1.1 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.

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:

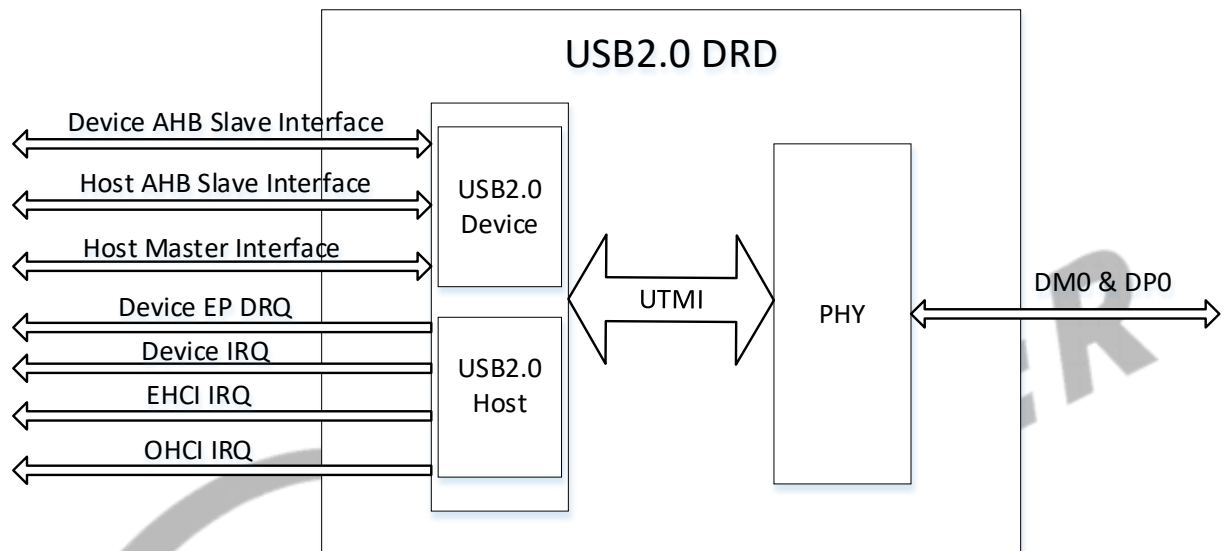
- Complies with USB2.0 Specification
- Supports USB Host function
 - Compatible with Enhanced Host Controller Interface (EHCI) Specification, Version 1.0
 - Compatible with Open Host Controller Interface (OHCI) Specification, Version 1.0a
 - Supports High-Speed (HS, 480 Mbit/s), Full-Speed (FS, 12 Mbit/s), 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
- Support BC1.2 protocol(only as CDP Host)

15.1.2 Block Diagram

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

Figure 15-1 USB2.0 DRD Controller Block Diagram



15.1.3 External Signals

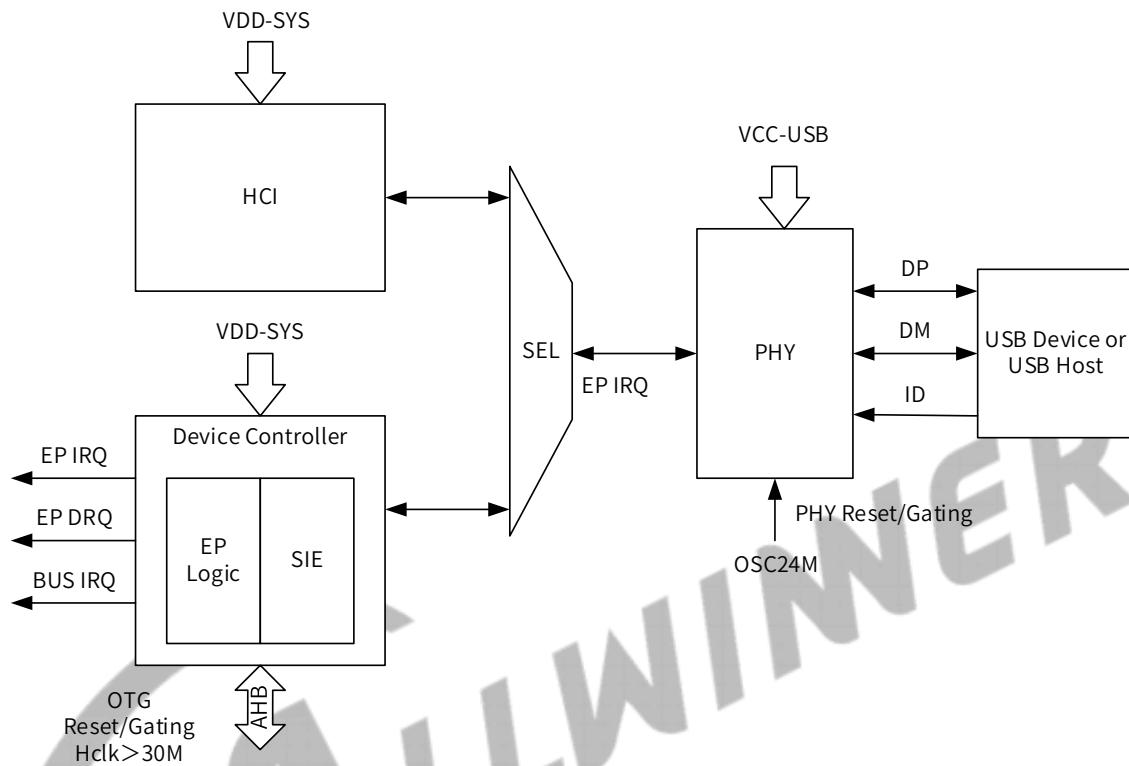
Table 15-1 USB2.0 DRD External Signals

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

15.1.4 Functional Description

15.1.4.1 Controller and PHY Connection Diagram

Figure 15-2 USB2.0 DRD Controller and PHY Connection Diagram



15.2 PWM

15.2.1 Overview

The Pulse Width Modulation (PWM) module can output the configurable PWM waveforms and measure the external input waveforms.

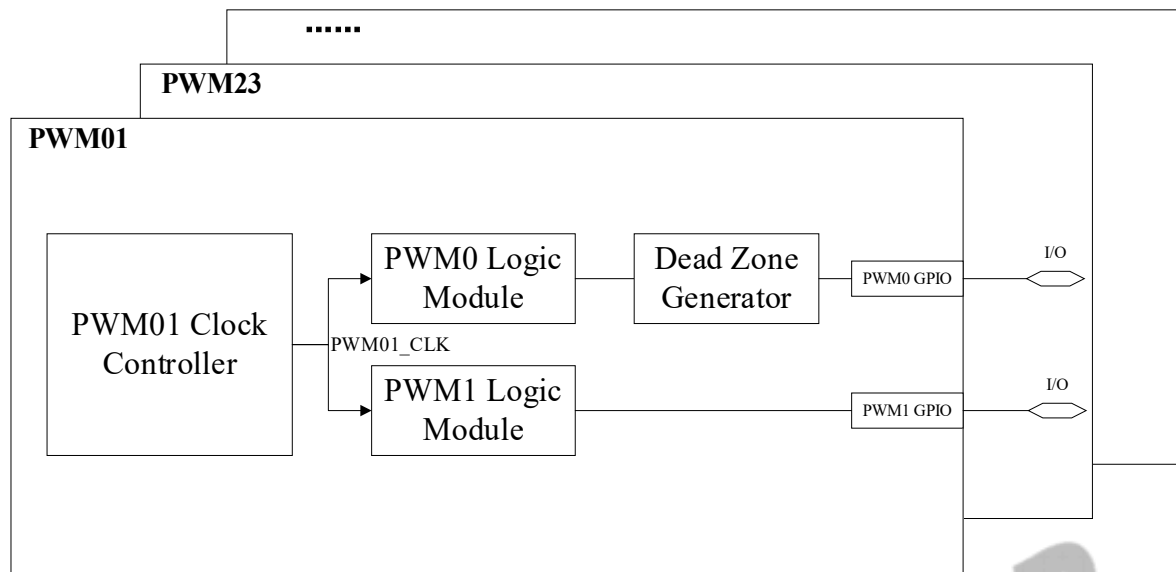
The PWM has the following features:

- Up to 4 PWM channels
 - Supports PWM continuous mode output
 - Supports PWM pulse mode output, and the pulse number is configurable
 - Output frequency range:
 - 0 to 24 MHz (when the clock source is DCXO)
 - 0 to 100 MHz (when the clock source is APB1 clock)
 - Various duty-cycle: 0% -100%
 - Minimum resolution: 1/65536
- Maximum 2 complementary pairs output
 - PWM01 pair (PWM-0 + PWM-1), PWM23 pair (PWM-2 + PWM-3)
 - Supports dead-zone generator, and the dead-zone time is configurable
- Maximum 2 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
- Maximum 4 channels capture input
 - Supports rising edge detection and falling edge detection for input waveform pulse
 - Supports pulse-width measurement for input waveform pulse

15.2.2 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 15-3 PWM Block Diagram



Each PWM pair consists of 1 clock module, 2 timer logic module, and 1 programmable dead-zone generator.

15.2.3 External Signals

The following table describes the external signals of the PWM.

Table 15-2 PWM External Signals

Signal	Description	Type
PWM-0	PWM Wave Output /Capture Wave Input 0	I/O
PWM-1	PWM Wave Output /Capture Wave Input 1	I/O
PWM-2	PWM Wave Output /Capture Wave Input 2	I/O
PWM-3	PWM Wave Output /Capture Wave Input 3	I/O

15.2.4 Functional Description

15.2.4.1 PWM Clock Controller

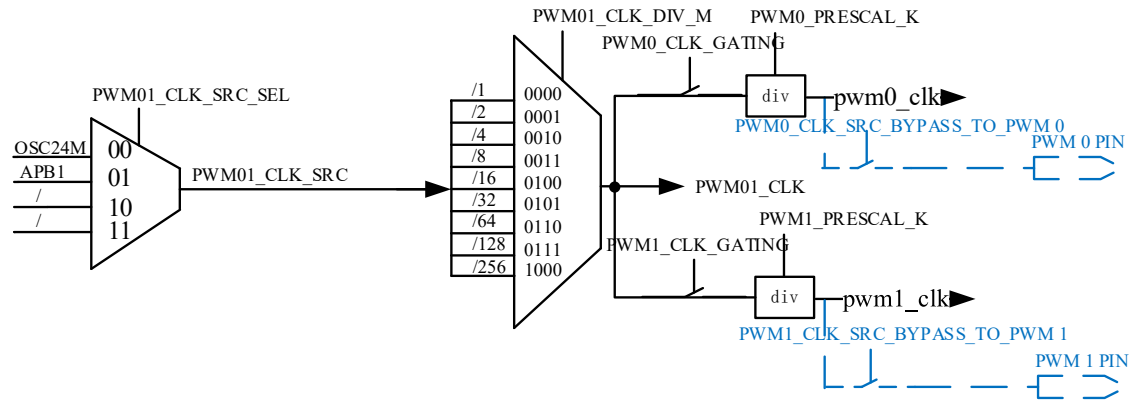
The clock controller of each PWM pair, for instance PWM01, includes clock source select (PWM01_CLK_SRC), 1-256 scaler (PWM01_CLK_DIV_M). Each PWM channel has the secondary frequency division (PWM_PRESCAL_K), clock source bypass (PWMx_CLK_BYPASS) and clock switch (PWMx_CLK_GATING).

The clock sources of PWM include OSC24M (HOSC) and APB1. The OSC comes from the external high-frequency oscillator; the APB1 is usually the 100MHz APB1 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 following figure indicate the bypass function of the clock source, see Figure 15-5 for the

details about implement. At last, the output clock of the clock controller is sent to the PWM logic module.

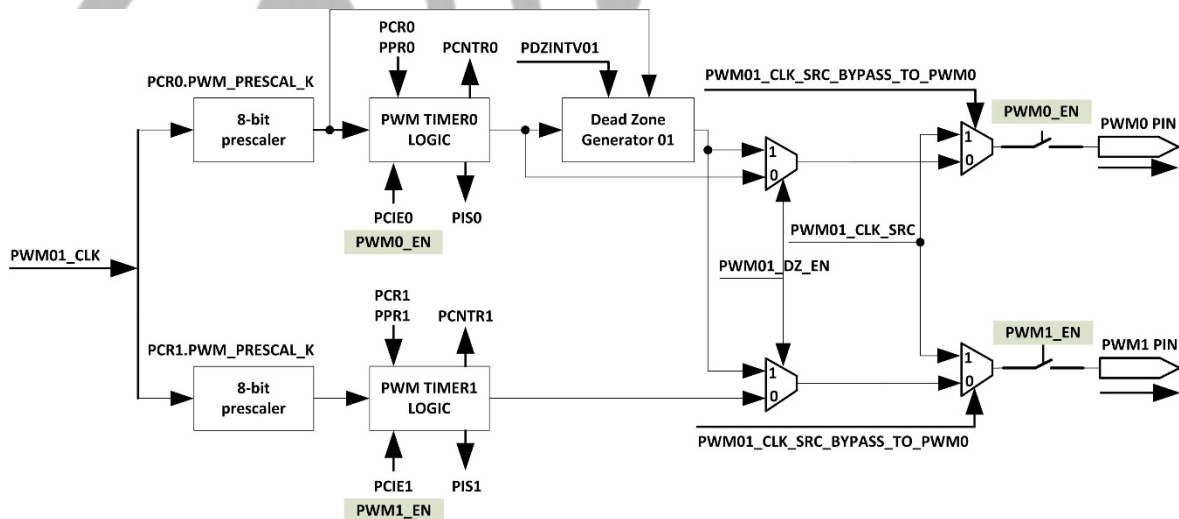
Figure 15-4 PWM01 Clock Controller Diagram



15.2.4.2 PWM Output

Take PWM01 as an example. Figure 15-5 indicates the PWM01 output logic diagram. The logic diagrams of other PWM pairs are the same as that of PWM01.

Figure 15-5 PWM01 Output Logic Module Diagram



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, and 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 glitches when updating the values of the PWM_ENTIRE_CYCLE and PWM_ACT_CYCLE. The PWM_CNTSTART doesn't support cache load, and only can be configured before enabling PWM output.

The PWM supports waveform output in cycle mode and pulse mode.

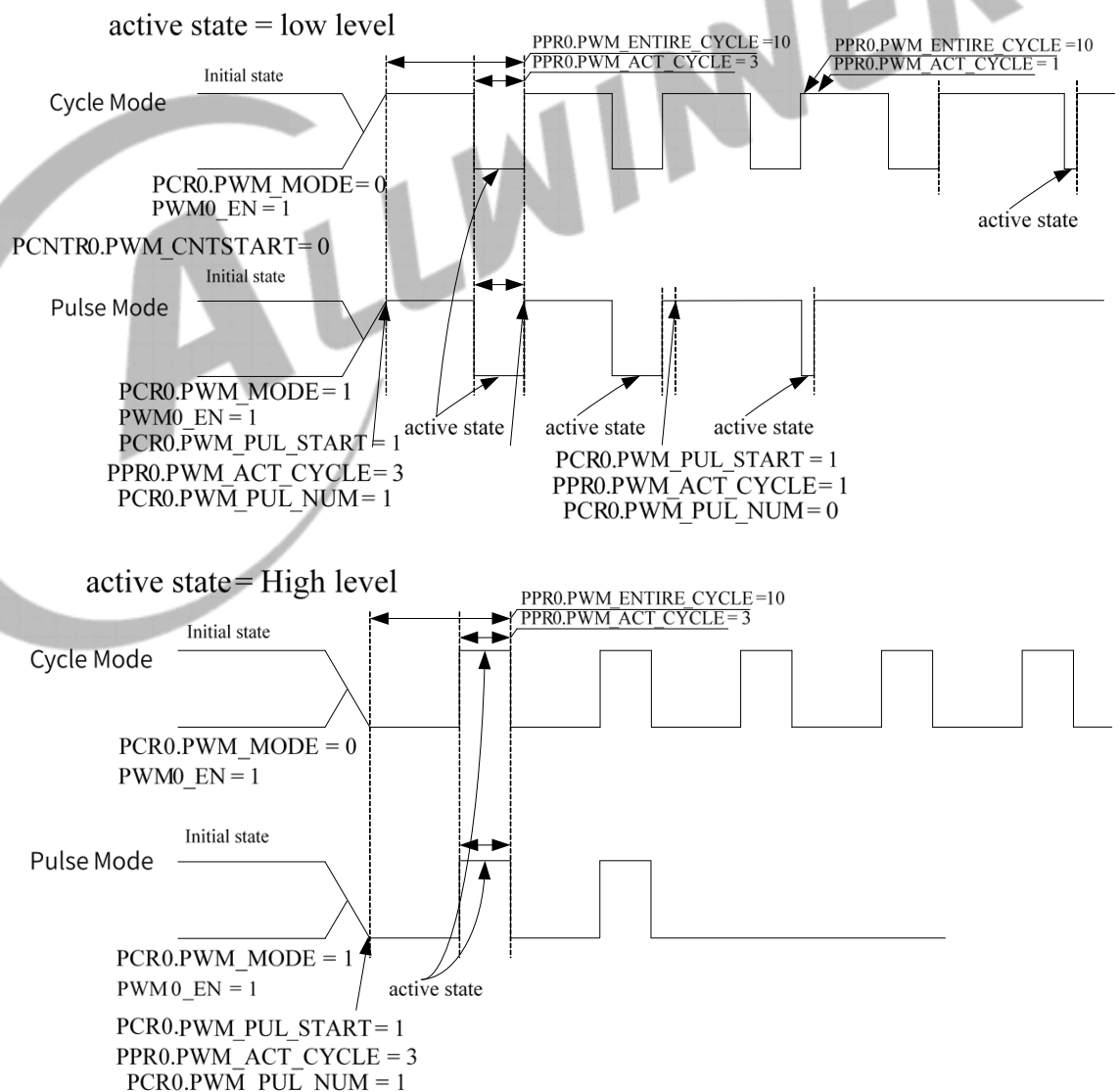
Cycle mode: The PWM outputs the set PWM waveform continually; that is, the output waveform is a continuous PWM square wave.

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.

NOTE

As the data width of PWM_PUL_CNT is 16 bits, it output maximum 65536 PWM waveforms.

The following figure shows active state in cycle mode and pulse mode.

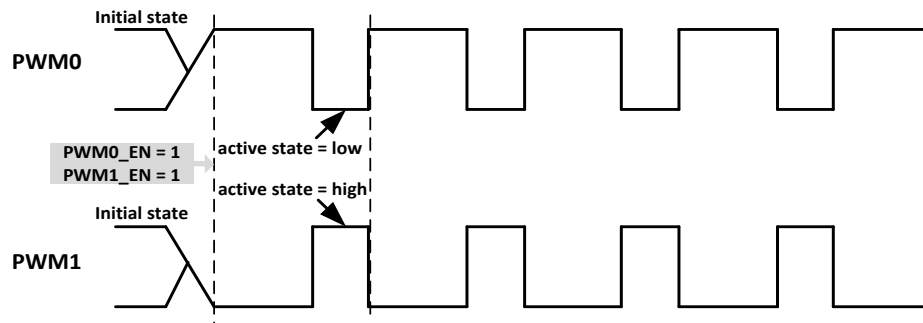


Every PWM pair supports two kinds of waveform output, including complementary pair Waveform output and PWM pair output with dead-time.

Complementary Pair Waveform Output

The following figure shows the complementary pair waveform output of PWM01.

Figure 15-6 PWM01 Complementary Pair Output



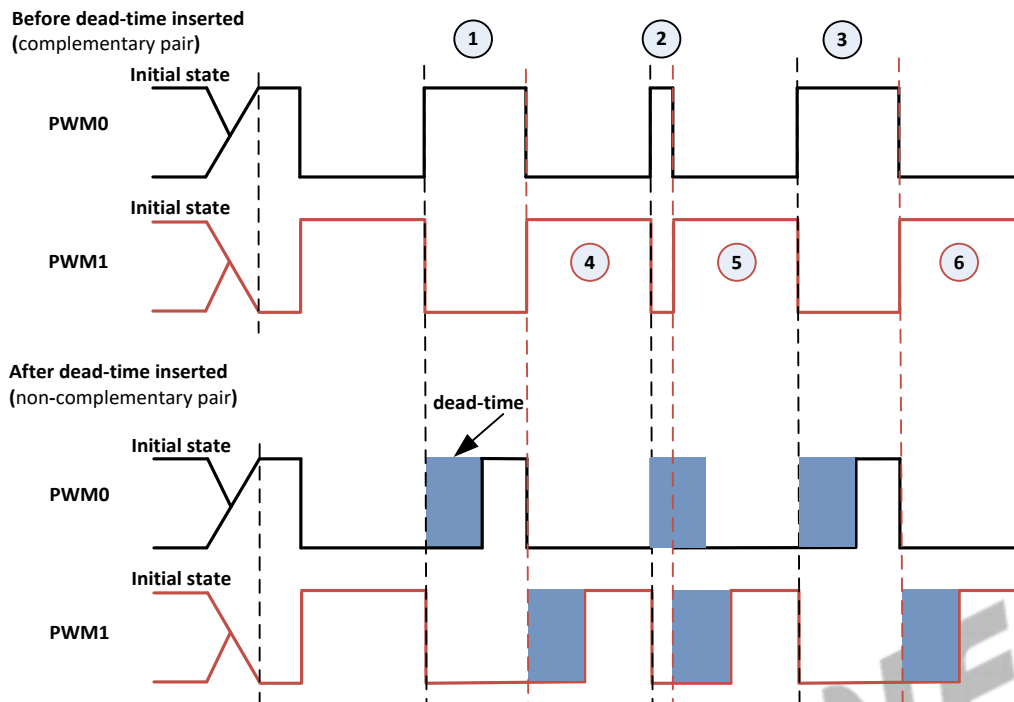
The complementary pair output needs to satisfy the following conditions:

- PWM0 and PWM1 share 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

Waveform Output with Dead-time Insertion

Every PWM pair has a programmable dead-time generator. When the dead-time function of the PWM pair is enabled, the PWM01 waveform output is decided by PWM timer logic and DeadZone Generator. The following figure shows the waveform output with dead-time insertion.

Figure 15-7 Output Waveform with Dead-time Insertion



The principle of inserting dead-time is to insert dead-time at every rising edge. If the high level period in mark ② is less than dead-time, then dead-time will override the high level period. Therefore, 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:

$$\text{Dead time} = (\text{PWM01\_CLK} / \text{PWM0\_PRESCALE\_K})^{-1} * \text{PDZINTV01}$$

15.2.4.3 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 (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 $\text{PCNTR0} > (\text{PPR0.PWM\_ENTIRE\_CYCLE} - \text{PPR0.PWM\_ACT\_CYCLE})$, then PWM0 outputs 1 (high level).

When $\text{PCNTR0} \leq (\text{PPR0.PWM\_ENTIRE\_CYCLE} - \text{PPR0.PWM\_ACT\_CYCLE})$, then PWM0 outputs 0 (low level).

The formula of the output period and the duty-cycle for PWM are as follows.

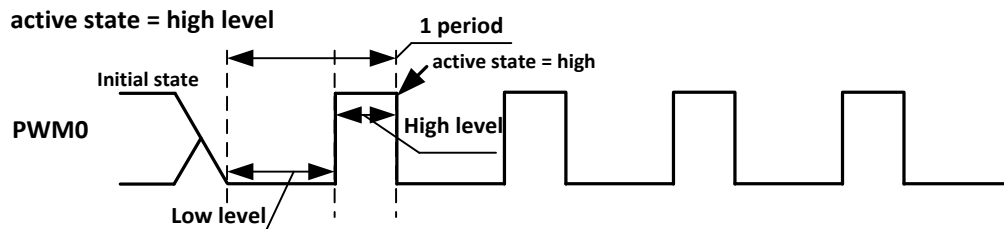
$$T_{\text{period}} = (\text{PWM01\_CLK} / \text{PWM0\_PRESCALE\_K})^{-1} * (\text{PPR0.PWM\_ENTIRE\_CYCLE} + 1)$$

$$T_{\text{high-level}} = (\text{PWM01\_CLK} / \text{PWM0\_PRESCALE\_K})^{-1} * \text{PPR0.PWM\_ACT\_CYCLE}$$

$$T_{\text{low-level}} = (\text{PWM01\_CLK} / \text{PWM0\_PRESCALE\_K})^{-1} * (\text{PPR0.PWM\_ENTIRE\_CYCLE} + 1 - \text{PPR0.PWM\_ACT\_CYCLE})$$

$$\text{Duty-cycle} = (\text{high level time}) / (1 \text{ period time}) = T_{\text{high-level}} / T_{\text{period}}$$

Figure 15-8 PWM0 High Level Active State



Active state of PWM0 channel is low level (PCR0.PWM_ACT_STA = 0)

When PCNTR0 > (PPR0.PWM_ENTIRE_CYCLE - PPR0.PWM_ACT_CYCLE), then PWM0 outputs 0.

When PCNTR0 ≤ (PPR0.PWM_ENTIRE_CYCLE - PPR0.PWM_ACT_CYCLE), then PWM0 outputs 1.

The formula of the output period and the duty-cycle for PWM are as follows.

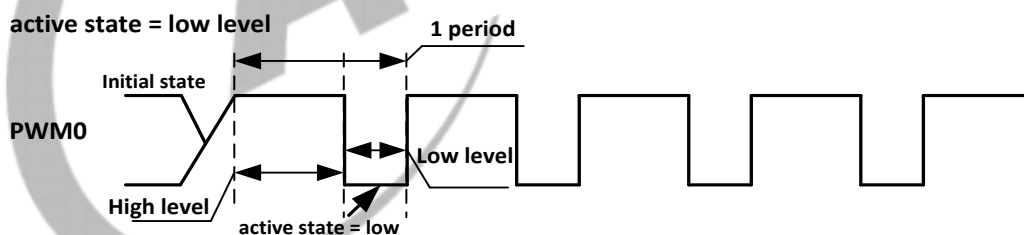
$$T_{\text{period}} = (\text{PWM01\_CLK} / \text{PWM0\_PRESCALE\_K})^{-1} * (\text{PPR0.PWM\_ENTIRE\_CYCLE} + 1)$$

$$T_{\text{high-level}} = (\text{PWM01\_CLK} / \text{PWM0\_PRESCALE\_K})^{-1} * (\text{PPR0.PWM\_ENTIRE\_CYCLE} + 1 - \text{PPR0.PWM\_ACT\_CYCLE})$$

$$T_{\text{low-level}} = (\text{PWM01\_CLK} / \text{PWM0\_PRESCALE\_K})^{-1} * \text{PPR0.PWM\_ACT\_CYCLE}$$

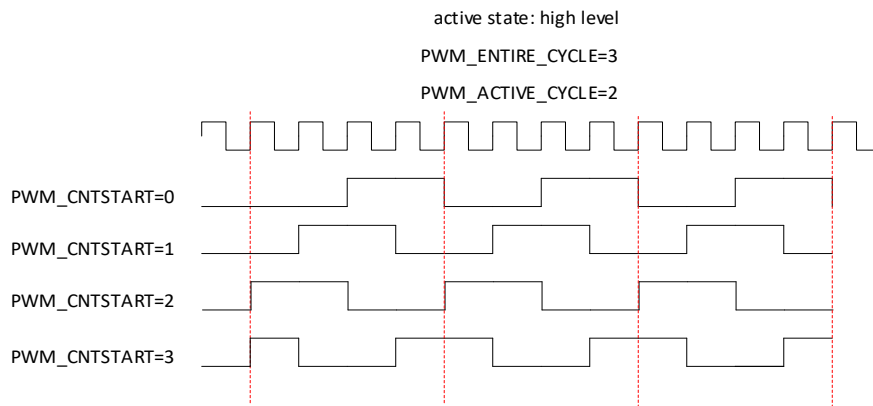
$$\text{Duty-cycle} = (\text{low level time}) / (1 \text{ period time}) = T_{\text{low-level}} / T_{\text{period}}$$

Figure 15-9 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.

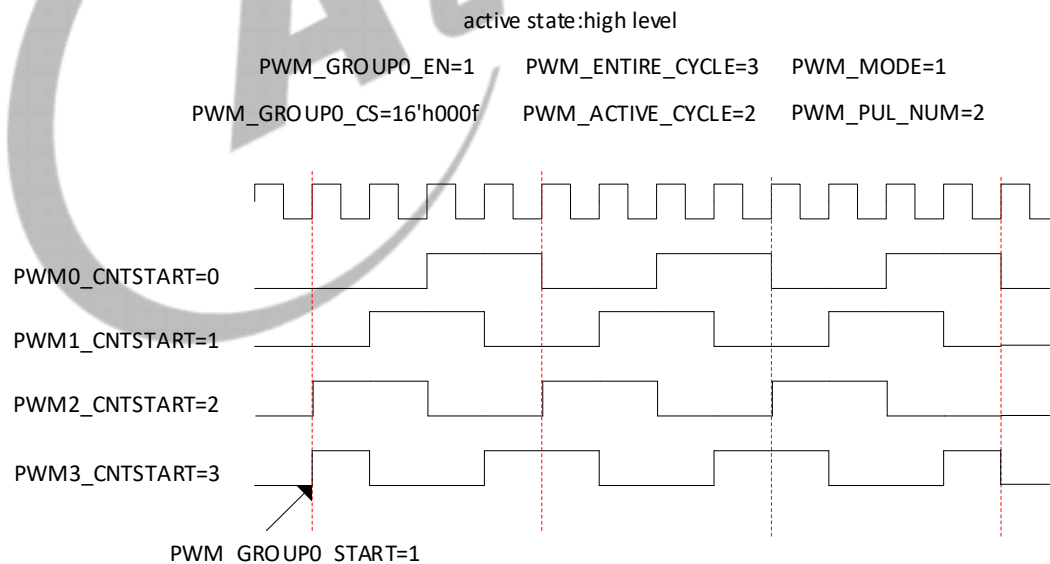
Figure 15-10 Phase of PWM0 High Level Active State



15.2.4.4 PWM Group Mode

The PWM group mode is used to satisfy the PWM group signals of stepper motors in different phases. Take PWM Group0 as an example. The PGR0.CS selects the PWM channel working in the same group; the same parameters of PWM_ENTIRE_CYCLE, PWM_ACT_CYCLE are set through the same clock configuration; the different PWM_COUNTER_START can output PWM group signals with the same duty-cycle and the different phase. PCNTR counting starts from 0 by default, and the phase of output waveform can be controlled through PWM_CNTSTART configuration. If the stepper motor uses the PWM group mode, it is recommended to divide 0, 1, 2, 3 into a group, and 4, 5, 6, 7 another group.

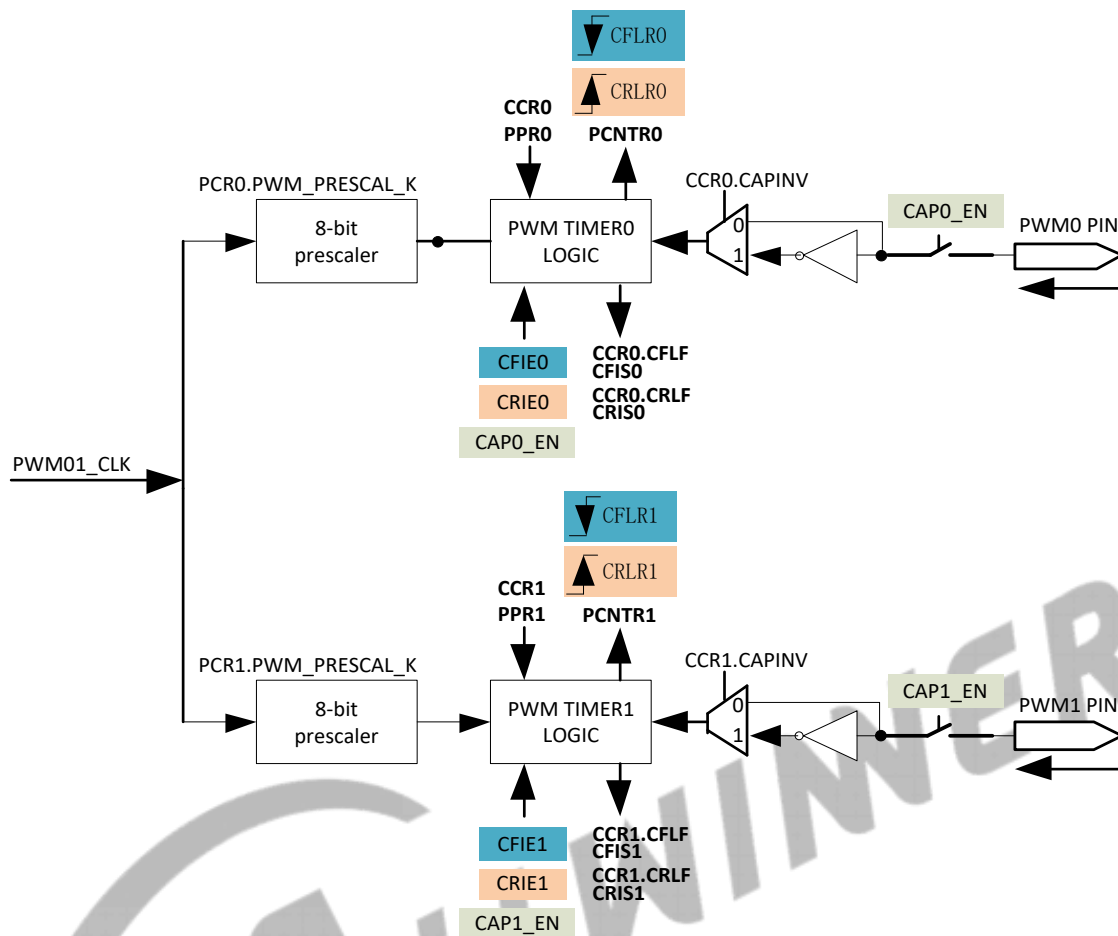
Figure 15-11 Group 0-3 PWM Signal Output



15.2.4.5 Waveform Capture

Each PWM pair is also used for capturing external clock. The following figure shows the PWM01 capture logic module diagram.

Figure 15-12 PWM01 Capture Logic Module Diagram



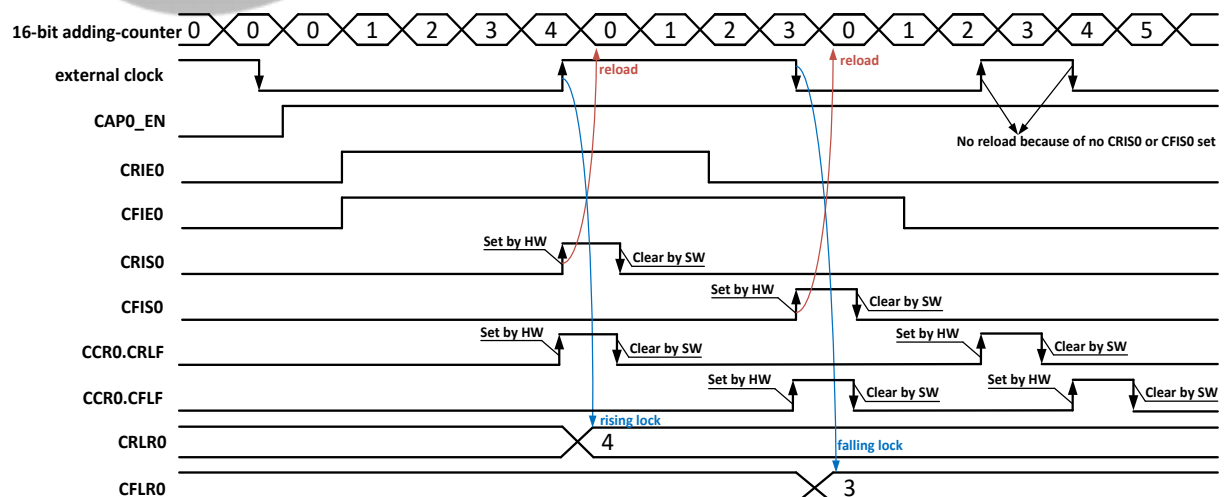
Take the PWM0 channel as an example. The PWM0 channel has one CFLR₀ and one CRLR₀ for capturing the values of up-counters on the falling edge and rising edge. You can calculate the period of the external clock by CFLR₀ and CRLR₀.

$$T_{low-level} = (PWM01\_CLK / PWM0\_PRESCALE\_K)^{-1} * CRLR0$$

$$T_{high-level} = (PWM01\_CLK / PWM0\_PRESCALE\_K)^{-1} * CFLR0$$

$$T_{period} = T_{high-level} + T_{low-level}$$

Figure 15-13 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 PCNTR is locked to CRLR0, and CCR0[CRLF] is set to 1. If CRIE0 is 1, and CRIS0 is set to 1, the PWM0 channel will send interrupt requests, and the up-counter will be reloaded to 0 and continue to count. If CRIE0 is 0, CRIS0 will not be set to 1 when the timer logic module of PWM0 captures a rising edge; thus, the up-counter will not be reloaded to 0.

When the timer logic module of PWM0 captures a falling edge, the current value of PCNTR is locked to CFLR0, and CCR0[CFLF] is set to 1. If CFIE0 is 1, and CFIS0 is set to 1, the PWM0 channel will send interrupt requests, and the up-counter will be reloaded to 0 and continue to count. If CFIE0 is 0, CFIS0 will not be set to 1 when the timer logic module of PWM0 captures a falling edge; thus, the up-counter will not be reloaded to 0.

15.2.5 Programming Guidelines

The following working mode takes PWM01 as an example, other PWM pairs and PWM01 are consistent.

15.2.5.1 Configuring Clock

- Step 1** PWM gating: When using PWM, write 1 to [PCGR](#) [PWMx_CLK_GATING].
- Step 2** PWM clock source select: Set [PCCR01](#) [PWM01_CLK_SRC] to select HOSC or APB1 clock.
- Step 3** PWM clock divider: Set [PCCR01](#) [PWM01_CLK_DIV_M] to select different frequency division coefficient (1/2/4/8/16/32/64/128/256).
- Step 4** PWM clock bypass: Set [PCGR](#) [PWM_CLK_SRC_BYPASS_TO_PWM] to 1, output the PWM clock after the secondary frequency division to the corresponding PWM output pin.
- Step 5** PWM internal clock configuration: Set [PCR](#) [PWM_PRESCAL_K] to select any frequency division coefficient from 1 to 256.



NOTE

The channels in the complementary output and group mode needs the same clock configuration, simultaneously opened clock gating, PWM parameter configuration, and enabled PWM output at the same time to ensure the synchronization among channels. Besides, as a PWM pair shares the same first-level clock division and gating, their two channels cannot be allocated to different groups in the group mode. If the allocation still exists, the same first-level clock division and the simultaneously opened gating is required. The total module needs to be reset when the group mode regroup.

15.2.5.2 Configuring PWM

- Step 1** PWM mode: Set [PCR](#) [PWM_MODE] to select cycle mode or pulse mode, if pulse mode, PCR[PWM_PUL_NUM] needs to be configured.
- Step 2** PWM active level: Set [PCR](#) [PWM_ACT_STA] to select a low level or high level.
- Step 3** PWM duty-cycle: Configure [PPR](#) [PWM_ENTIRE_CYCLE] and [PPR](#) [PWM_ACT_CYCLE] after clock gating is opened.
- Step 4** PWM starting/stopping phase: Configure [PCNTR](#) [PWM_COUNTER_START] after the clock gating is enabled and before the PWM is enabled. You can verify whether the configuration was successful by reading back [PCNTR](#) [PWM_COUNTER_STATUS].
- Step 5** Enable PWM: Configure [PER](#) to select the corresponding PWM enable bit; when selecting pulse mode, [PCR](#) [PWM_PUL_START] needs to be enabled.

15.2.5.3 Configuring Deadzone

- Step 1** Set initial value: set [PDZCR01](#) [PDZINTV01].
- Step 2** Enable Deadzone: set [PDZCR01](#) [PWM01_DZ_EN].

15.2.5.4 Configuring Capture Input

- Step 1** Enable capture: Configure CER to enable the corresponding channel.
- Step 2** Capture mode: Configure [CCR](#) [CRLF] and [CCR](#) [CFLF] to select rising edge capture or falling edge capture, configure [CCR](#) [CAPINV] to select whether the input signal does reverse processing.

15.2.6 Register List

Module Name	Base Address
PWM	0x0200 0C00

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
PCGR	0x0040	PWM Clock Gating Register
PDZCR01	0x0060	PWM01 Dead Zone Control Register

Register Name	Offset	Description
PDZCR23	0x0064	PWM23 Dead Zone Control Register
PER	0x0080	PWM Enable Register
PGR0	0x0090	PWM Group0 Register
PGR1	0x0094	PWM Group1 Register
CER	0x00C0	Capture Enable Register
PCR	0x0100+0x0000+N*0x0020 (N= 0–3)	PWM Control Register
PPR	0x0100+0x0004+N*0x0020 (N= 0–3)	PWM Period Register
PCNTR	0x0100+0x0008+N*0x0020 (N= 0–3)	PWM Count Register
PPCNTR	0x0100+0x000C+N*0x0020 (N= 0–3)	PWM Pulse Count Register
CCR	0x0100+0x0010+N*0x0020 (N= 0–3)	Capture Control Register
CRLR	0x0100+0x0014+N*0x0020 (N= 0–3)	Capture Rise Lock Register
CFLR	0x0100+0x0018+N*0x0020 (N= 0–3)	Capture Fall Lock Register

15.2.7 Register Description

15.2.7.1 0x0000 PWM IRQ Enable Register (Default Value: 0x0000_0000)

Offset:0x0000			Register Name: PIER
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
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
15:4	/	/	/
3	R/W	0x0	PCIE3 PWM Channel 3 Interrupt Enable 0: PWM Channel 3 Interrupt Disable 1: PWM Channel 3 Interrupt Enable

Offset:0x0000			Register Name: PIER
Bit	Read/Write	Default/Hex	Description
2	R/W	0x0	PCIE2 PWM Channel 2 Interrupt Enable 0: PWM Channel 2 Interrupt Disable 1: PWM Channel 2 Interrupt Enable
1	R/W	0x0	PCIE1 PWM Channel 1 Interrupt Enable 0: PWM Channel 1 Interrupt Disable 1: PWM Channel 1 Interrupt Enable
0	R/W	0x0	PCIE0 PWM Channel 0 Interrupt Enable 0: PWM Channel 0 Interrupt Disable 1: PWM Channel 0 Interrupt Enable

15.2.7.2 0x0004 PWM IRQ Status Register (Default Value: 0x0000_0000)

Offset:0x0004			Register Name: PISR
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17	R/W1C	0x0	PGIS1 PWM Group 1 Interrupt Status
16	R/W1C	0x0	PGIS0 PWM Group 0 Interrupt Status
15:4	/	/	/
3	R/W1C	0x0	PIS3 PWM Channel 3 Interrupt Status When the PWM channel 3 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit. Reads 0: PWM channel 3 interrupt is not pending. Reads 1: PWM channel 3 interrupt is pending. Writes 0: No effect. Writes 1: Clear PWM channel 3 interrupt status.
2	R/W1C	0x0	PIS2 PWM Channel 2 Interrupt Status When the PWM channel 2 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit. Reads 0: PWM channel 2 interrupt is not pending. Reads 1: PWM channel 2 interrupt is pending. Writes 0: No effect. Writes 1: Clear PWM channel 2 interrupt status.

Offset:0x0004			Register Name: PISR
Bit	Read/Write	Default/Hex	Description
1	R/W1C	0x0	PIS1 PWM Channel 1 Interrupt Status When the PWM channel 1 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit. Reads 0: PWM channel 1 interrupt is not pending. Reads 1: PWM channel 1 interrupt is pending. Writes 0: No effect. Writes 1: Clear PWM channel 1 interrupt status.
0	R/W1C	0x0	PIS0 PWM Channel 0 Interrupt Status When the PWM channel 0 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit. Reads 0: PWM channel 0 interrupt is not pending. Reads 1: PWM channel 0 interrupt is pending. Writes 0: No effect. Writes 1: Clear PWM channel 0 interrupt status.

15.2.7.3 0x0010 PWM Capture IRQ Enable Register (Default Value: 0x0000_0000)

Offset:0x0010			Register Name: CIER
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15	R/W	0x0	CFIE7 If the enable bit is set to 1, when the 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 the enable bit is set to 1, when the 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

Offset:0x0010			Register Name: CIER
Bit	Read/Write	Default/Hex	Description
13	R/W	0x0	CFIE6 If the enable bit is set to 1, when the 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 the enable bit is set to 1, when the 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 the enable bit is set to 1, when the 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 the enable bit is set to 1, when the 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 the enable bit is set to 1, when the 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
8	R/W	0x0	CRIE4 If the enable bit is set to 1, when the 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 the enable bit is set to 1, when the 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

Offset:0x0010			Register Name: CIER
Bit	Read/Write	Default/Hex	Description
6	R/W	0x0	CRIE3 If the enable bit is set to 1, when the 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 the enable bit is set to 1, when the 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 the enable bit is set to 1, when the 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 the enable bit is set to 1, when the 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 the enable bit is set to 1, when the 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 the enable bit is set to 1, when the 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
0	R/W	0x0	CRIE0 If the enable bit is set to 1, when the capture channel 0 captures rising edge, it generates a capture channel 0 pending. 0: Capture channel 0 rise lock interrupt disable 1: Capture channel 0 rise lock interrupt enable

15.2.7.4 0x0014 PWM Capture IRQ Status Register (Default Value: 0x0000_0000)

Offset:0x0014			Register Name: CISR
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
15	R/W1C	0x0	CFIS7 Status of the capture channel 7 falling lock interrupt When the capture channel 7 captures falling edge, if the fall lock interrupt (CFIE7) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 7 interrupt is not pending. Reads 1: The capture channel 7 interrupt is pending. Writes 0: No effect. Writes 1: Clear the status of the capture channel 7 interrupt.
14	R/W1C	0x0	CRIS7 Status of the capture channel 7 rising lock interrupt When the capture channel 7 captures rising edge, if the rise lock interrupt (CRIE7) is enabled, this bit is set to 1 by hardware. Write 1 to clear this bit. Reads 0: The capture channel 7 interrupt is not pending. Reads 1: The capture channel 7 interrupt is pending. Writes 0: No effect. Writes 1: Clear the status of the capture channel 7 interrupt.
13	R/W1C	0x0	CFIS6 Status of the capture channel 6 falling lock interrupt When the capture channel 6 captures falling edge, if the fall lock interrupt (CFIE6) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 6 interrupt is not pending. Reads 1: The capture channel 6 interrupt is pending. Writes 0: No effect. Writes 1: Clear the status of the capture channel 6 interrupt.

Offset:0x0014			Register Name: CISR
Bit	Read/Write	Default/Hex	Description
12	R/W1C	0x0	CRIS6 Status of the capture channel 6 rising lock interrupt. When the capture channel 6 captures rising edge, if the rise lock interrupt (CRIE6) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 6 interrupt is not pending. Reads 1: The capture channel 6 interrupt is pending. Writes 0: No effect. Writes 1: Clear the status of the capture channel 6 interrupt.
11	R/W1C	0x0	CFIS5 Status of the capture channel 5 falling lock interrupt. When the capturing channel 5 captures falling edge, if the fall lock interrupt (CFIE5) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 5 interrupt is not pending. Reads 1: The capture channel 5 interrupt is pending. Writes 0: No effect. Writes 1: Clear the status of the capture channel 5 interrupt.
10	R/W1C	0x0	CRIS5 Status of the capture channel 5 rising lock interrupt. When the capture channel 5 captures rising edge, if the rise lock interrupt (CRIE5) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 5 interrupt is not pending. Reads 1: The capture channel 5 interrupt is pending. Writes 0: No effect. Writes 1: Clear the status of the capture channel 5 interrupt.

Offset:0x0014			Register Name: CISR
Bit	Read/Write	Default/Hex	Description
9	R/W1C	0x0	CFIS4 Status of the capture channel 4 falling lock interrupt When the capture channel 4 captures falling edge, if the fall lock interrupt (CFIE4) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 4 interrupt is not pending. Reads 1: The capture channel 4 interrupt is pending. Writes 0: No effect. Writes 1: Clear the status of the capture channel 4 interrupt.
8	R/W1C	0x0	CRIS4 Status of the capture channel 4 rising lock interrupt. When the capture channel 4 captures rising edge, if the rise lock interrupt (CRIE4) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 4 interrupt is not pending. Reads 1: The capture channel 4 interrupt is pending. Writes 0: No effect. Writes 1: Clear the status of the capture channel 4 interrupt status.
7	R/W1C	0x0	CFIS3 Status of the capture channel 3 falling lock interrupt. When the capture channel 3 captures falling edge, if the fall lock interrupt (CFIE3) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 3 interrupt is not pending. Reads 1: The capture channel 3 interrupt is pending. Writes 0: no effect. Writes 1: Clear the status of the capture channel 3 interrupt.

Offset:0x0014			Register Name: CISR
Bit	Read/Write	Default/Hex	Description
6	R/W1C	0x0	CRIS3 Status of the capture channel 3 rising lock interrupt When the capture channel 3 captures rising edge, if the rise lock interrupt (CRIE3) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 3 interrupt is not pending. Reads 1: The capture channel 3 interrupt is pending. Writes 0: no effect. Writes 1: Clear the status of the capture channel 3 interrupt.
5	R/W1C	0x0	CFIS2 Status of the capture channel 2 falling lock interrupt When the capture channel 2 captures falling edge, if the fall lock interrupt (CFIE2) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 2 interrupt is not pending. Reads 1: The capture channel 2 interrupt is pending. Writes 0: no effect. Writes 1: Clear the status of the capture channel 2 interrupt.
4	R/W1C	0x0	CRIS2 Status of the capture channel 2 rising lock interrupt. When the capture channel 2 captures rising edge, if the rise lock interrupt (CRIE2) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 2 interrupt is not pending. Reads 1: The capture channel 2 interrupt is pending. Writes 0: no effect. Writes 1: Clear the status of the capture channel 2 interrupt.

Offset:0x0014			Register Name: CISR
Bit	Read/Write	Default/Hex	Description
3	R/W1C	0x0	CFIS1 Status of the capture channel 1 falling lock interrupt When the capture channel 1 captures falling edge, if the fall lock interrupt (CFIE1) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 1 interrupt is not pending. Reads 1: The capture channel 1 interrupt is pending. Writes 0: no effect. Writes 1: Clear the status of the capture channel 1 interrupt.
2	R/W1C	0x0	CRIS1 Status of the capture channel 1 rising lock interrupt. When the capture channel 1 captures rising edge, if the rise lock interrupt (CRIE1) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 1 interrupt is not pending. Reads 1: The capture channel 1 interrupt is pending. Writes 0: no effect. Writes 1: Clear the status of the capture channel 1 interrupt.
1	R/W1C	0x0	CFIS0 Status of the capture channel 0 falling lock interrupt When the capture channel 0 captures falling edge, if the fall lock interrupt (CFIE0) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 0 interrupt is not pending. Reads 1: The capture channel 0 interrupt is pending. Writes 0: no effect. Writes 1: Clear the status of the capture channel 0 interrupt.

Offset:0x0014			Register Name: CISR
Bit	Read/Write	Default/Hex	Description
0	R/W1C	0x0	CRIS0 Status of the capture channel 0 rising lock interrupt When the capture channel 0 captures rising edge, if the rise lock interrupt (CRIE0) is enabled, this bit is set 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 0 interrupt is not pending. Reads 1: The capture channel 0 interrupt is pending. Writes 0: no effect. Writes 1: Clear the status of the capture channel 0 interrupt.

15.2.7.5 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	0x0	PWM01_CLK_SRC Select PWM01 Clock Source 00: HOSC 01: APB1 Others: Reserved
6:4	/	/	/
3:0	R/W	0x0	PWM01_CLK_DIV_M PWM01 Clock Divide M 0000: /1 0001: /2 0010: /4 0011: /8 0100: /16 0101: /32 0110: /64 0111: /128 1000: /256 Others: Reserved

15.2.7.6 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	0x0	PWM23_CLK_SRC_SEL Select PWM23 Clock Source 00: HOSC 01: APB1 Others: Reserved
6:4	/	/	/
3:0	R/W	0x0	PWM23_CLK_DIV_M PWM23 Clock Divide M 0000: /1 0001: /2 0010: /4 0011: /8 0100: /16 0101: /32 0110: /64 0111: /128 1000: /256 Others: Reserved

15.2.7.7 0x0040 PWM Clock Gating Register (Default Value: 0x0000_0000)

Offset: 0x0040			Register Name: PCGR
Bit	Read/Write	Default/Hex	Description
31:20	/	/	/
19	R/W	0x0	PWM3_CLK_BYPASS Bypass clock source (after pre-scale) to PWM3 output 0: not bypass 1: bypass
18	R/W	0x0	PWM2_CLK_BYPASS Bypass clock source (after pre-scale) to PWM2 output 0: not bypass 1: bypass
17	R/W	0x0	PWM1_CLK_BYPASS Bypass clock source (after pre-scale) to PWM1 output 0: not bypass 1: bypass

Offset: 0x0040			Register Name: PCGR
Bit	Read/Write	Default/Hex	Description
16	R/W	0x0	PWM0_CLK_BYPASS Bypass clock source (after pre-scale) to PWM0 output 0: not bypass 1: bypass
15:4	/	/	/
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

15.2.7.8 0x0060 PWM01 Dead Zone Control Register (Default Value: 0x0000_0000)

Offset: 0x0060			Register Name: PDZCR01
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:8	R/W	0x0	PWM01_DZ_INTV PWM01 Dead Zone Interval Value
7:1	/	/	/
0	R/W	0x0	PWM01_DZ_EN PWM01 Dead Zone Enable 0: Dead Zone disable 1: Dead Zone enable

15.2.7.9 0x0064 PWM23 Dead Zone Control Register (Default Value: 0x0000_0000)

Offset: 0x0064			Register Name: PDZCR23
Bit	Read/Write	Default/Hex	Description

Offset:0x0064			Register Name: PDZCR23
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:8	R/W	0x0	PWM23_DZ_INTV PWM23 Dead Zone Interval Value
7:1	/	/	/
0	R/W	0x0	PWM23_DZ_EN PWM23 Dead Zone Enable 0: Dead Zone disable 1: Dead Zone enable

15.2.7.10 0x0080 PWM Enable Register (Default Value: 0x0000_0000)

Offset:0x0080			Register Name: PER
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
3	R/W	0x0	PWM3_EN When PWM is enabled, the 16-bit up-counter starts working and PWM channel3 is permitted to output PWM waveform. 0: PWM disable 1: PWM enable
2	R/W	0x0	PWM2_EN When PWM is enabled, the 16-bit up-counter starts working and PWM channel2 is permitted to output PWM waveform. 0: PWM disable 1: PWM enable
1	R/W	0x0	PWM1_EN When PWM is enabled, the 16-bit up-counter starts working and PWM channel1 is permitted to output PWM waveform. 0: PWM disable 1: PWM enable
0	R/W	0x0	PWM0_EN When PWM is enabled, the 16-bit up-counter starts working and PWM channel0 is permitted to output PWM waveform. 0: PWM disable 1: PWM enable

15.2.7.11 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 The PWM channels selected in PWMG0_CS start to output PWM waveform at the same time.
16	R/W	0x0	PWMG0_EN PWM Group0 Enable.
15:0	R/W	0x0	PWMG0_CS If bit[i] is set, the PWM i is selected as one channel of PWM Group0.

15.2.7.12 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 The 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, the PWM i is selected as one channel of PWM Group1.

15.2.7.13 0x00C0 Capture Enable Register (Default Value: 0x0000_0000)

Offset: 0x00C0			Register Name: CER
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
3	R/W	0x0	CAP3_EN When enabling the capture function, the 16-bit up-counter starts working, and the capture channel3 is permitted to capture external falling edge or rising edge. 0: Capture disable 1: Capture enable

Offset: 0x00C0			Register Name: CER
Bit	Read/Write	Default/Hex	Description
2	R/W	0x0	CAP2_EN When enabling the capture function, the 16-bit up-counter starts working, and the capture channel2 is permitted to capture external falling edge or rising edge. 0: Capture disable 1: Capture enable
1	R/W	0x0	CAP1_EN When enabling the capture function, the 16-bit up-counter starts working, and the capture channel1 is permitted to capture external falling edge or rising edge. 0: Capture disable 1: Capture enable
0	R/W	0x0	CAP0_EN When enabling the capture function, the 16-bit up-counter starts working, and the capture channel is permitted to capture external falling edge or rising edge. 0: Capture disable 1: Capture enable

15.2.7.14 0x0100 + N*0x20 PWM Control Register (Default Value: 0x0000_0000)

Offset: 0x0100 + 0x0 + N*0x20 (N=0-3)			Register Name: PCR
Bit	Read/Write	Default/Hex	Description
31:16	R/W	0x0	PWM_PUL_NUM In pulse mode, the PWM outputs pulse for PWM_PUL_NUM+1 times and then stops.
15:12	/	/	/
11	R	0x0	PWM_PERIOD_RDY PWM Period Register Ready 0: PWM period register is ready to write 1: PWM period register is busy

Offset:0x0100+0x0+N*0x20 (N=0-3)			Register Name: PCR
Bit	Read/Write	Default/Hex	Description
10	R/WAC	0x0	PWM_PUL_START PWM Pulse Output Start 0: No effect 1: Output pulse for PWM_PUL_NUM+1. After finishing configuration for the output pulse, set this bit once , then PWM would output waveform. After the waveform is finished, the bit will be cleared automatically.
9	R/W	0x0	PWM_MODE PWM Output Mode Select 0: Cycle mode 1: Pulse mode
8	R/W	0x0	PWM_ACT_STA PWM Active State 0: Low Level 1: High Level
7:0	R/W	0x0	PWM_PRESCAL_K PWM pre-scale K, actual pre-scale is (K+1). K = 0, actual pre-scale: 1 K = 1, actual pre-scale: 2 K = 2, actual pre-scale: 3 K = 3, actual pre-scale: 4 ... K = 255, actual pre-scale: 256

15.2.7.15 0x0104 + N*0x20 PWM Period Register (Default Value: 0x0000_0000)

Offset:0x0100+0x04+N*0x20 (N=0-3)			Register Name: PPR
Bit	Read/Write	Default/Hex	Description
31:16	R/W	0x0	PWM_ENTIRE_CYCLE Number of the entire cycles in the PWM clock. 0: 1 cycle 1: 2 cycles ... N: N+1 cycles If the register needs to be modified dynamically, the PCLK should be faster than the PWM CLK.

Offset: 0x0100+0x04+N*0x20 (N=0-3)			Register Name: PPR
Bit	Read/Write	Default/Hex	Description
15:0	R/W	0x0	PWM_ACT_CYCLE Number of the active cycles in the PWM clock. 0: 0 cycle 1: 1 cycle ... N: N cycles

15.2.7.16 0x0108 + N*0x20 PWM Counter Register (Default Value: 0x0000_0000)

Offset: 0x0100+0x08+N*0x20 (N=0-3)			Register Name: PCNTR
Bit	Read/Write	Default/Hex	Description
31:16	R/W	0x0	PWM_COUNTER_START PWM counter value is set for phase control.
15:0	R	0x0	PWM_COUNTER_STATUS On PWM output or capture input, reading this register could get the current value of the PWM 16-bit up-counter.

15.2.7.17 0x010C + N*0x20 PWM Pulse Counter Register (Default Value: 0x0000_0000)

Offset: 0x0100+0x0C+N*0x20 (N=0-3)			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.

15.2.7.18 0x0110 + N*0x20 PWM Capture Control Register (Default Value: 0x0000_0000)

Offset: 0x0100+0x10+N*0x20 (N=0-3)			Register Name: CCR
Bit	Read/Write	Default/Hex	Description
31:5	/	/	/
4	R/W1C	0x0	CRLF When the capture channel captures a rising edge, the current value of the 16-bit up-counter is latched to CRLR, and then this bit is set 1 by hardware. Write 1 to clear this bit.

Offset:0x0100+0x10+N*0x20 (N=0-3)			Register Name: CCR
Bit	Read/Write	Default/Hex	Description
3	R/W1C	0x0	CFLF When the capture channel captures a falling edge, the current value of the 16-bit up-counter is latched to CFLR, and then this bit is set 1 by hardware. Write 1 to clear this bit.
2	R/W	0x0	CRTE Rising edge capture trigger enable
1	R/W	0x0	CFTE Falling edge capture trigger enable
0	R/W	0x0	CAPINV Inverse the signal input from capture channel before 16-bit counter of capture channel. 0: not inverse 1: inverse

15.2.7.19 0x0114 + N*0x20 PWM Capture Rise Lock Register (Default Value: 0x0000_0000)

Offset:0x0100+0x14+N*0x20 (N=0-3)			Register Name: CRLR
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R	0x0	CRLR When the capture channel captures a rising edge, the current value of the 16-bit up-counter is latched to the register.

15.2.7.20 0x0118 + N*0x20 PWM Capture Fall Lock Register (Default Value: 0x0000_0000)

Offset:0x0100+0x18+N*0x20 (N=0-3)			Register Name: CFLR
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R	0x0	CFLR When the capture channel captures a falling edge, the current value of the 16-bit up-counter is latched to the register.

15.3 PWM_BL

15.3.1 Overview

The PWM_BL supports backlight BOOST control and adjusts PWM_BL output duty cycle by detecting feedback voltage level.

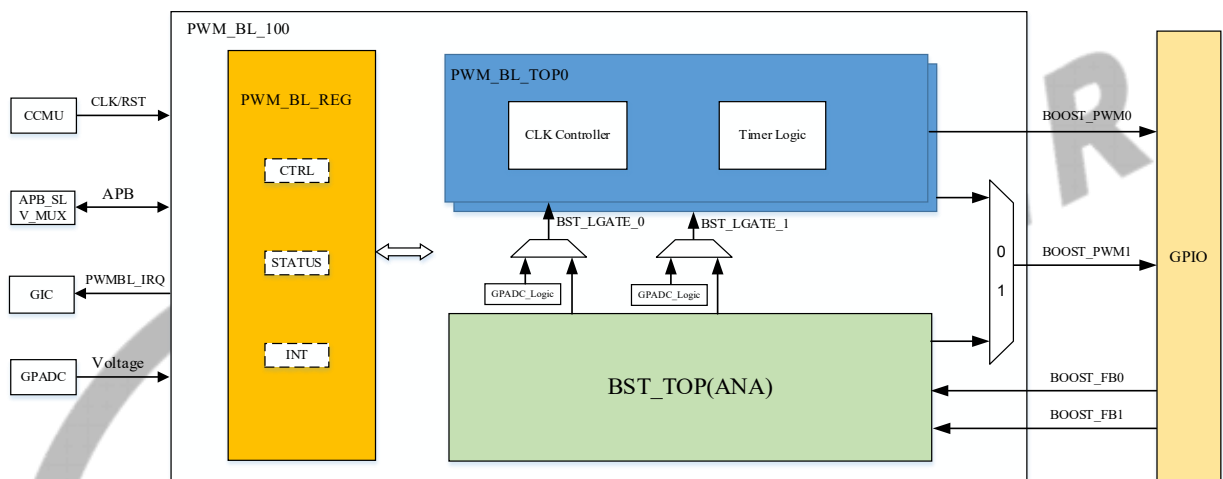
The PWM_BL has the following features

- Supports two PWM_BL outputs: PWM_BL0 outputs BOOST_PWM0 from the digital circuit, and PWM_BL1 outputs BOOST_PWM1 from either the hybrid analog-digital circuit or the purely analog circuit.
- Supports backlight BOOST control and adjusting PWM_BL output duty cycle by feedback voltage level detection.
- Supports GPADC voltage quantization mode, where the difference between the quantized voltage and the reference voltage is divided into four levels. The intervals are configurable, and different duty cycle adjustment steps can be selected according to the intervals.
- Analog circuit:
 - Supports configurable feedback reference voltage v_{ref_max} (reference voltage for maximum brightness), ranging from 0.2V to 0.6V with 0.1V steps. Backlight brightness is adjustable in 256 levels, with the N-th level brightness according to a reference voltage of $v_{ref\_max} * N / 256$.
 - Boost function can be controlled via software.
 - Screen analog circuit: 10uH external inductor, 10uF load capacitor, ensuring flicker-free dimming with no abnormalities. For example, output less than 0.7% voltage ripple at 10.4V/20mA (actual screen requirement)
 - Screen backlight circuit: 10uH external inductor, 10uF load capacitor, ensuring flicker-free dimming with no abnormalities. For example, output less than 2% voltage ripple at 12.5V/40mA (actual screen requirement)
 - Overcurrent protection (OCP) function: turn off the N-MOS transistor when triggered.
- PWM_BL0/1 (digital circuit characteristics):
 - 0~1.2 MHz output frequency range
 - Configurable step of duty cycle adjustment. Dynamic configuration is supported.
 - Configurable interval of duty cycle adjustment, ranging from 0-255 PWM clock cycles
 - Configurable starting duty cycle
 - Configurable output polarity
 - After triggering OCP, two behaviors are configurable: constant low-level output; or after OCP recovery, the BOOST_PWM duty cycle resumes adjustment from the initial value.
- PWM_BL1 (purely analog circuit characteristics):

- Supports 0.25M/0.5M/0.75M/1M working frequencies
- Supports ibias trim and test output
- Supports OCP (triggered when exceeding 120% of maximum load current) and disable function
- Supports ramp trim function
- Supports GPADC voltage quantization mode with FB signal pre-amplification by 2 times and disable function
- Supports gating and disable function

15.3.2 Block Diagram

Figure 15-14 PWM_BL Block Diagram



The PWM_BL_100 mainly consists of digital module PWM_BL_TOP, analog module BST_TOP, and register modules. PWM_BL0 outputs BOOST_PWM0 from the digital circuit, and PWM_BL1 outputs BOOST_PWM1 from either the digital circuit or the purely analog circuit.

- PWM_BL_TOP: Backlight BOOST control is implemented through a hybrid analog-digital approach.
 - CLK Control: Clock configuration module, including clock source selection, two-level frequency division selection and clock gate.
 - Timer Logic: It can adjust and output the duty cycle of BOOST_PWM0 according to the feedback voltage result from analog circuit detection.
- BST_TOP: Includes two analog circuits
 - BST0: It implements feedback voltage detection, and then sends detection results to digital logic.
 - BST1: It implements backlight BOOST control through purely analog circuit and then adjusts and outputs the duty cycle of signal BOOST_PWM1.
- GPADC_Loic: It compares the quantized voltage input by GPADC and the reference voltage configured by registers and then generates duty cycle to adjust indication signals

- PWM_BL_REG: PWM_BL_100 register modules.

15.3.3 External Signals

Table 15-3 PWM_BL External Signals

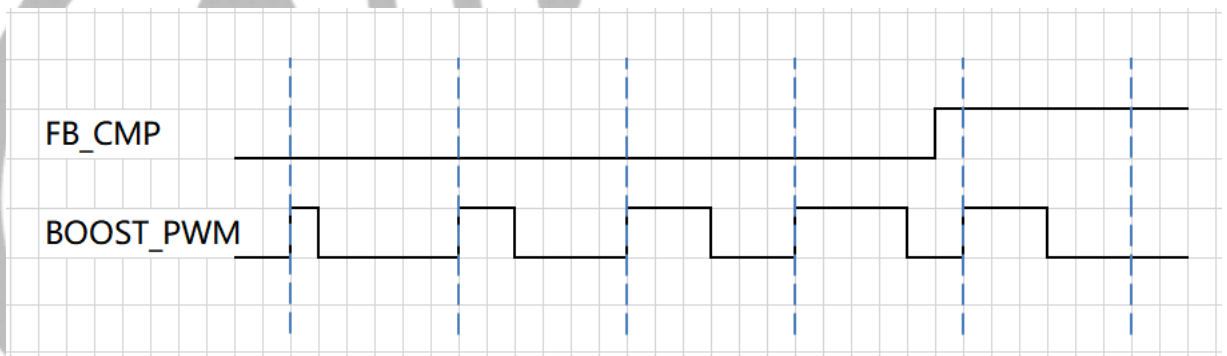
Signal Name	Description	Type
BOOST0-PWM	PWM Backlight Output 0	O
BOOST0-FB	PWM Backlight Input 0	I
BOOST1-PWM	PWM Backlight Output 1	O
BOOST1-FB	PWM Backlight Input 1	I

15.3.4 Typical application

15.3.4.1 PWM_BL0 outputs PWM Waveform

The PWM_BL0 working diagram is shown as below. The output BOOST_PWM signal adjusts output duty cycle based on the feedback voltage result FB_CMP through analog detection. When FB_CMP is 0, it indicates the feedback voltage is lower than the reference voltage; and the output BOOST_PWM duty cycle increases progressively. When FB_CMP is 1, it indicates the feedback voltage is higher than the reference voltage; and the output BOOST_PWM duty cycle decreases progressively.

Figure 15-15 PWM_BL0 Working Diagram

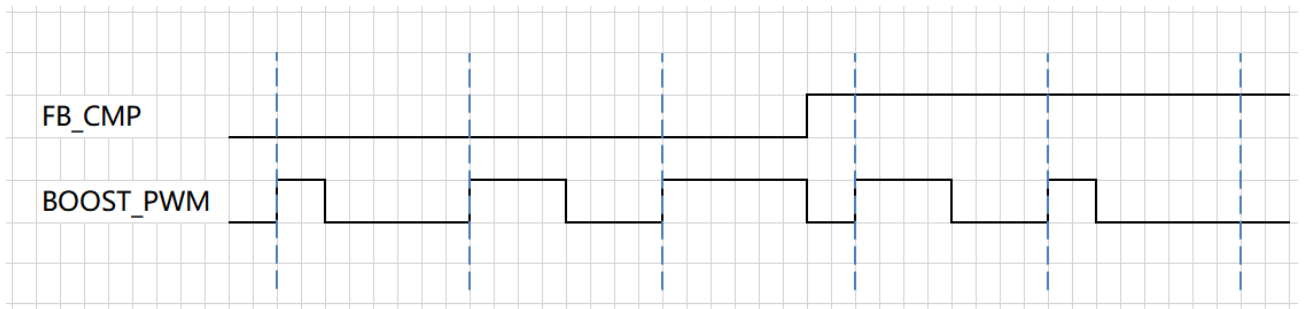


15.3.5 Functional Description

15.3.5.1 PWM_BL Duty Cycle Adjustment Step

As shown in the figure below, the duty cycle adjustment step is configured during the PWM_BL0 duty cycle adjustment. The effective level time increment and decrement value of output BOOST_PWM signals can be configured as required.

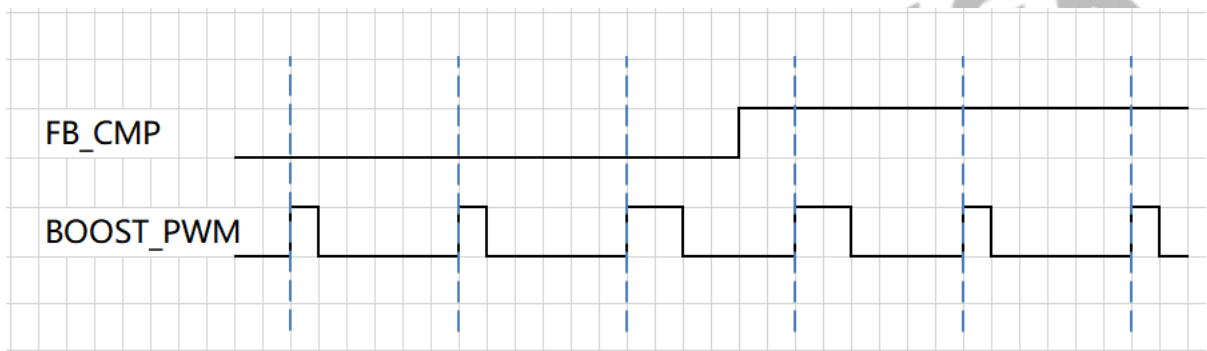
Figure 15-16 PWM_BL0 Duty Cycle Adjustment Step



15.3.5.2 PWM_BL Duty Cycle Adjustment Step Interval

As shown in the figure below, the duty cycle adjustment interval is configured during the PWM_BL0 duty cycle adjustment. The effective level time increment and decrement intervals of output BOOST_PWM signals can be configured as required.

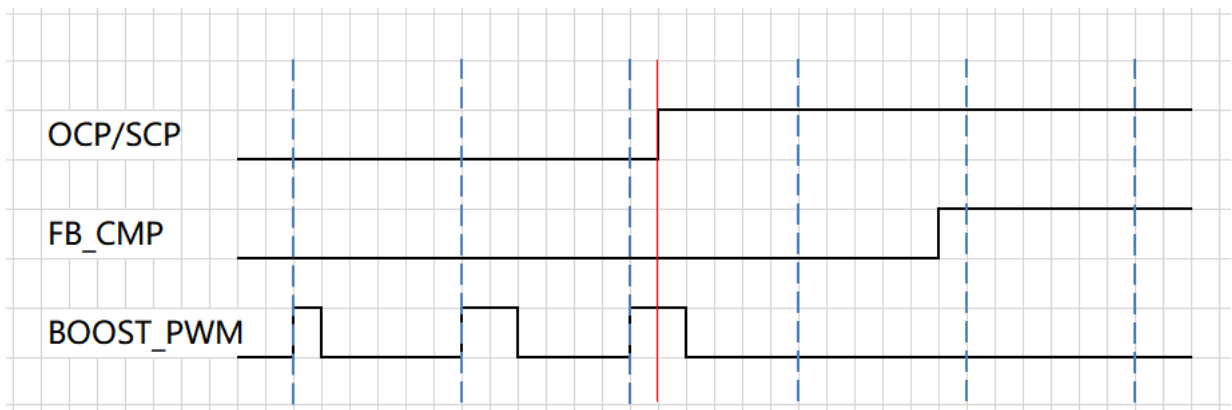
Figure 15-17 PWM_BL0 Duty Cycle Adjustment Step Interval



15.3.5.3 PWM_BL Abnormal Protection

The figure below shows PWM_BL0 overvoltage and overcurrent protection. When the overvoltage and overcurrent protection signals generated by analog circuits are set to 1, the output BOOST_PWM signal level will be low at the next clock cycle.

Figure 15-18 PWM_BL0 Abnormal Protection



15.3.5.4 PWM_BL Uses the Quantized Voltage Output by GPADC

PWM_BL uses the quantized voltage output by GPADC, and the output BOOST_PWM duty cycle adjustment working procedure is shown as below.

1. Input BOOST feedback voltage through GPIO and send it to the analog component BST_TOP inside the PWM_BL, and then send processed feedback voltage to GPADC.
2. The GPADC quantizes the two analog circuits of BST_TOP into 12-bit voltage values, and then sends the effective voltage indication signals to PWM_BL. The 12-bit quantized voltage value (V) and the real voltage value have a linear relationship. If the input voltage range is 0-1.8V, the quantized voltage value (V) is 0-4095; if the input voltage range is 0-0.9V, the quantized voltage value (V) is 0-2047.
3. The 12-bit reference voltage (PWM_GPADC_VF) is configured in the register PWMBL_GPADC_VF_REG, and its quantized standard should have a same linear relationship with that of GPADC. In the comparison between quantized voltage value (V) and PWM_GPADC_VF, the BOOST_PWM duty cycle output by PWM_BL increases progressively on the condition of $V < \text{PWM\_GPADC\_VF}$, and the BOOST_PWM duty cycle output by PWM_BL decreases progressively on the condition of $V > \text{PWM\_GPADC\_VF}$. Whether to use its comparison result or that of original BST analog circuits inside the PWM_BL can be set through the PWMBL_GPADC_EN in the register PWMBL_GPADC_CTRL_REG.
4. As shown below, the voltage difference $\Delta = |V - \text{PWM\_GPADC\_VF}|$ has four levels.
 - 1) In the case of $0 < \text{voltage difference } \Delta < V_0$, the duty adjustment step is PWM_GPADC_ACT_CYCLE_STEP0 configured by register PWMBL_GPADC_AC_STEP0_REG.
 - 2) In the case of $V_0 < \text{voltage difference } \Delta < V_1$, the duty adjustment step is PWM_GPADC_ACT_CYCLE_STEP1 configured by register PWMBL_GPADC_AC_STEP1_REG.
 - 3) In the case of $V_1 < \text{voltage difference } \Delta < V_2$, the duty adjustment step is PWM_GPADC_ACT_CYCLE_STEP2 configured by register PWMBL_GPADC_AC_STEP2_REG.
 - 4) In the case of $V_2 < \text{voltage difference } \Delta$, the duty adjustment step is PWM_GPADC_ACT_CYCLE_STEP3 configured by register PWMBL_GPADC_AC_STEP3_REG.

15.3.6 Programming Guidelines

15.3.6.1 Configure system source of PWM_BL_100

The system source registers are shown in the following table. For details, refer to the CCU section.

Table 15-4 PWM_BL_100 System Source Configuration

Procedure	Register	Offset	Configuration Description

Procedure	Register	Offset	Configuration Description
PWM_BL_100			
Step1	PWM_BL_RST	0x0794[16]	PWM_BL_100 reset, which should be released before clock configuration.
Step2	CLK_SRC_SEL	0x0790[26:24]	PWM_BL_100 clock sources select
Step3	FACTOR_M	0x0790[4:0]	PWM_BL_100 clock sources post divider configuration
Step4	PWM_BL_CLK_GATING	0x0790[31]	PWM_BL_100 400M clock enable
Step5	PWM_BL_GATING	0x0794[0]	PWM_BL_100 APB clock enable

15.3.6.2 Configure PWM_BL0 Output without GPADC Voltage Quantization Mode

Table 15-5 PWM_BL0 Output Configuration without GPADC Voltage Quantization Mode

Procedure	Register	Offset	Configuration Description
Step1	PWMBL_USE_ANA	0x0000[2]	The implementation method configuration is shown as below. PWM_BL0 is implemented through hybrid digital-analog circuits PWM_BL1 is implemented through purely analog circuits or hybrid digital-analog circuits
Step2	PWMBL_CLK_SRC_SEL	0x0004[31:30]	pwmbL_clk clock sources can be selected from 24Mhz, 100MHz, and 400MHz
Step3	PWMBL_CLK_DIV_M	0x0004[3:0]	The primary frequency division factor of working clock
Step4	PWMBL_PRESCAL_K	0x0004[15:8]	The secondary frequency division factor of working clock
Step5	PWMBL_CLK_GATING	0x0000[1]	Working clock gate enable
Step6	PWM_ACT_STA	0x0000[4]	Output polarity configuration
Step7	PWMBL_PRD_REG	0x0008[15:0]	Output period length configuration
Step8	PWM_MAX_ACT_CYCLE	0x000C[15:0]	Maximum cycle configuration of effective output level
Step9	PWM_INIT_ACT_CYCLE	0x000C[31:16]	Initial value configuration of effective output level
Step10	PWM_ACT_CYCLE_INCREASE_STEP	0x0010[15:0]	Step increase and decrease after duty cycle update
Step11	PWM_ACT_UPDATE_CYCLE	0x0014[7:0]	PWM_BL duty cycle update time interval configuration

Procedure	Register	Offset	Configuration Description
Step12	PWMBL_DIMM	0x0054[23:16]	PWM_BL BST load current configuration
Step13	PWMBL_VREF_MAX_SEL	0x0054[2:0]	The reference voltage configuration required for the maximum screen brightness
Step14	PWMBL_BST_EN	0x0050[0]	PWMBL BST Enable
Step16	PWMBL_EN	0x0000[0]	PWM Channel 0 Enable

15.3.6.3 Configure PWM_BL0 Output with GPADC Voltage Quantization Mode

Table 15-6 PWM_BL0 Output Configuration with GPADC Voltage Quantization Mode

Procedure	Register	Offset	Configuration
Step1	PWMBL_USE_ANA	0x0000[2]	The implementation method configuration is shown as below. PWM_BL0 is implemented through hybrid digital-analog circuits PWM_BL1 is implemented through purely analog circuit or hybrid digital-analog circuits
Step2	PWMBL_CLK_SRC_SEL	0x0004[31:30]	pwmb_clk clock sources can be selected from 24Mhz, 100MHz, and 400MHz
Step3	PWMBL_CLK_DIV_M	0x0004[3:0]	The primary frequency division factor of working clock
Step4	PWMBL_PRESCAL_K	0x0004[15:8]	The secondary frequency division factor of working clock
Step5	PWMBL_CLK_GATING	0x0000[1]	Working clock gate enable
Step6	PWM_ACT_STA	0x0000[4]	Output polarity configuration
Step7	PWMBL_PRD_REG	0x0008[15:0]	Output period length configuration
Step8	PWM_MAX_ACT_CYCLE	0x000C[15:0]	Maximum cycle configuration of effective output level
Step9	PWM_INIT_ACT_CYCLE	0x000C[31:16]	Initial value configuration of effective output level

Procedure	Register	Offset	Configuration
Step10	PWM_GPADC_ACT_CYCLE_STEP0, PWM_GPADC_ACT_CYCLE_STEP1, PWM_GPADC_ACT_CYCLE_STEP2, PWM_GPADC_ACT_CYCLE_STEP3	0x0120[15:0], 0x0124[15:0], 0x0128[15:0], 0x012C[15:0]	Step increase and decrease after duty cycle update
Step11	PWM_GPADC_VF	0x0104[11:0]	Reference Voltage Configuration
Step12	PWM_GPADC_V0, PWM_GPADC_V1, PWM_GPADC_V2	0x0110[11:0], 0x0114[11:0], 0x0118[11:0]	Voltage Range Configuration
Step15	PWMBL_GPADC_EN	0x0100[0]	Enable GPADC voltage quantization mode
Step15	PWMBL_BST_EN	0x0050[0]	PWMBL BST Enable
Step16	PWMBL_EN	0x0000[0]	PWM Channel 0 Enable

15.3.6.4 Cautions and notes



NOTE

1. PWM_BL0 and PWM_BL1 are implemented using hybrid digital-analog circuits, with no need for registers PWMBL_ANA2_REG and PWMBL_ANA3_REG configuration.
2. Registers PWM_ACT_CYCLE_INCREASE_STEP and PWM_GPADC_VF support dynamic configuration.

15.3.7 Register list

Module Name	Base Address	Comments
PWM_BL	0x0200A000---0x0200A3FF	PWM BOOST0

Register Name	Offset	Description
PWM_BL		
PWMBL_CTRL_REG	0x0000+N*0x0200(N=0~1)	PWMBL Control Register
PWMBL_CLK_CFG_REG	0x0004+N*0x0200(N=0~1)	PWMBL Clock Configuration Register
PWMBL_PRD_REG	0x0008+N*0x0200(N=0~1)	PWMBL Period Register
PWMBL_ACT_CYCLE_REG	0x000C+N*0x0200(N=0~1)	PWMBL Active Cycle Register
PWMBL_ACT_CYCLE_STEP_REG	0x0010+N*0x0200(N=0~1)	PWMBL Active Cycle Step Register
PWMBL_ACT_UP_CYCLE_REG	0x0014+N*0x0200(N=0~1)	PWMBL Active Update Cycle Register
PWMBL_OCP_FLT_V_REG	0x0018+N*0x0200(N=0~1)	PWMBL OCP Filter Value Register
PWMBL_INT_EN_REG	0x0040+N*0x0200(N=0~1)	PWMBL Interrupt Enable Register
PWMBL_INT_PENDING_REG	0x0044+N*0x0200(N=0~1)	PWMBL Interrupt Pending Register
PWMBL_ANA0_REG	0x0050+N*0x0200(N=0~1)	PWMBL Analog 0 Register
PWMBL_ANA1_REG	0x0054+N*0x0200(N=0~1)	PWMBL Analog 1 Register
PWMBL_ANA2_REG	0x0058+N*0x0200(N=0~1)	PWMBL Analog 2 Register
PWMBL_ANA3_REG	0x005C+N*0x0200(N=0~1)	PWMBL Analog 3 Register
PWMBL_RSD_REG	0x0060+N*0x0200(N=0~1)	PWMBL Reserved Register
PWMBL_GPADC_CTRL_REG	0x0100+N*0x0200(N=0~1)	PWMBL GPADC Control Register
PWMBL_GPADC_VF_REG	0x0104+N*0x0200(N=0~1)	PWMBL GPADC Voltage Reference Register
PWMBL_GPADC_V0_REG	0x0110+N*0x0200(N=0~1)	PWMBL GPADC Voltage0 Register
PWMBL_GPADC_V1_REG	0x0114+N*0x0200(N=0~1)	PWMBL GPADC Voltage1 Register
PWMBL_GPADC_V2_REG	0x0118+N*0x0200(N=0~1)	PWMBL GPADC Voltage2 Register
PWMBL_GPADC_AC_STEP0_REG	0x0120+N*0x0200(N=0~1)	PWMBL GPADC Active Cycle Step0 Register

Register Name	Offset	Description
PWMBL_GPADC_AC_STEP1_REG	0x0124+N*0x0200(N=0~1)	PWMBL GPADC Active Cycle Step1 Register
PWMBL_GPADC_AC_STEP2_REG	0x0128+N*0x0200(N=0~1)	PWMBL GPADC Active Cycle Step2 Register
PWMBL_GPADC_AC_STEP3_REG	0x012C+N*0x0200(N=0~1)	PWMBL GPADC Active Cycle Step3 Register

15.3.8 Register Description

15.3.8.1 0x0000+N*0x0200 (N=0~1) PWMBL Control Register (Default Value: 0x0000_0000)

Offset: 0x0000+N*0x0200 (N=0~1)			Register Name: PWMBL_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:6	/	/	/
5	R/W	0x0	PWMBL_OCP_INIT_EN. 0: Once OCP overcurrent occurs, the output BOOST_PWM waveform is fixed to 0, which can only be adjusted based on the initial value if the PWM_BL is turned off and then turned on. After OCP overcurrent resumes, the duty cycle of the output BOOST_PWM waveform is adjusted again from PWM_INIT_ACT_CYCLE.
4:3	/	/	/
2	R/W	0x0	PWMBL_USE_ANA. 0: PWM_BL1 is implemented by purely analog circuit 1: PWM_BL1 is implemented by hybrid analog-digital circuit. Note: PMW_BL0 is implemented by hybrid analog-digital circuit.
1	R/W	0x0	PWMBL_CLK_GATING. Gating clock for PWM_BL. 0: Mask 1: Pass
0	R/W	0x0	PWMBL_EN. PWM Channel 0 Enable 0: PWM disable 1: PWM enable.

15.3.8.2 0x0004+N*0x0200 (N=0~1) PWMBL Clock Configuration Register (Default Value: 0x0000_0000)

Offset: 0x0004+N*0x0200 (N=0~1)			Register Name: PWMBL_CLK_CFG_REG
Bit	Read/Write	Default/Hex	Description
31:30	R/W	0x00	PWMBL_CLK_SRC_SEL. pwmbL_clk Clock Source Select 00: OSC24M 01: APB_CLK (100MHz) 10: 400M 11: /
29:16	/	/	/
15:8	R/W	0x0000	PWMBL_PRESCAL_K. PWMBL 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.
6:4	/	/	/
3:0	R/W	0x0000	PWMBL_CLK_DIV_M. PWMBL clock divide M 0000: /1 0001: /2 0010: /4 0011: /8 0100: /16 0101: /32 0110: /64 0111: /128 1000: /256 others: /

15.3.8.3 0x0008+N*0x0200 (N=0~1) PWMBL Period Register (Default Value: 0x0000_0000)

Offset:0x0008+N*0x0200 (N=0~1)			Register Name: PWMBL_PRD_REG
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/

Offset:0x0008+N*0x0200 (N=0~1)			Register Name: PWMBL_PRD_REG
Bit	Read/Write	Default/Hex	Description
15:0	R/W	0x0	PWM_ENTIRE_CYCLE. PWM_BL cycle values at the unit of pwmb_clk clock cycle 0 = 1 cycle 1 = 2 cycles N = N+1 cycles

15.3.8.4 0x000C+N*0x0200 (N=0~1) PWMBL Active Cycle Register (Default Value: 0x0000_0000)

Offset:0x000C+N*0x0200 (N=0~1)			Register Name: PWMBL_ACT_CYCLE_REG
Bit	Read/Write	Default/Hex	Description
31:16	R/W	0x0	PWM_INIT_ACT_CYCLE PWM_BL initial effective value at the unit of pwmb_clk clock cycle 0 = 0 cycle 1 = 1 cycles N = N cycles
15:0	R/W	0x0	PWM_MAX_ACT_CYCLE The maximum effective PWMBL value at the unit of pwmb_clk clock cycle PWM_MAX_ACT_CYCLE should be smaller than PWM_ENTIRE_CYCLE, indicating the output PWM_BL waveform duty cycle rate cannot be 100%. 0= 0 cycle 1= 1 cycles N= N cycles

15.3.8.5 0x0010+N*0x0200 (N=0~1) PWMBL Active Cycle Step Register (Default Value: 0x0000_0000)

Offset:0x0010+N*0x0200 (N=0~1)			Register Name: PWMBL_ACT_CYCLE_STEP_REG
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/

Offset:0x0010+N*0x0200 (N=0~1)			Register Name: PWMBL_ACT_CYCLE_STEP_REG
Bit	Read/Write	Default/Hex	Description
15:0	R/W	0x0	PWM_ACT_CYCLE_INCREASE_STEP The increased and decreased step during the PWMBL duty cycle update at the unit of pwmbld_clk clock cycle 0= 0 cycle 1= 1 cycles N= N cycles

15.3.8.6 0x0014+N*0x0200 (N=0~1) PWMBL Active Update Cycle Register (Default Value: 0x0000_0000)

Offset:0x0014+N*0x0200 (N=0~1)			Register Name: PWMBL_ACT_UP_CYCLE_REG
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0x0	PWM_ACT_UPDATE_CYCLE. PWM_BL duty cycle update time interval at the unit of BOOST_PWM cycle 0= 1 cycle 1= 2 cycles N= N+1 cycles

15.3.8.7 0x0018+N*0x0200 (N=0~1) PWMBL OCP Filter Value Register (Default Value: 0x0000_0000)

Offset:0x0018+N*0x0200 (N=0~1)			Register Name: PWMBL_OCP_FLT_V_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	PWM_OCP_FLT_EN Enable OCP overvoltage signal filtering 0: Disable 1: Enable
30:8	/	/	/
7:0	R/W	0x0	PWM_OCP_FLT_V If the period of the filtered OCP overvoltage signal by analog output is longer than PWM_OCP_FLT_V, the OCP overvoltage signal will be considered as effective. The unit is pwmbld_clk clock cycle.

15.3.8.8 0x0040+N*0x0200 (N=0~1) PWMBL Interrupt Enable Register (Default Value: 0x0000_0000)

Offset: 0x0040+N*0x0200 (N=0~1)			Register Name: PWMBL_INT_EN_REG
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
3	R/W	0x0	PWMBL_ACT_CYCLE_MAX_INT_EN PWM_BL duty cycle is equal to PWM_MAX_ACT_CYCLE 0: Disable 1: Enable
2	R/W	0x0	PWMBL_ACT_CYCLE_ERROR_INT_EN PWM_BL duty cycle interrupt error enable. When the duty cycle is 0, the analog output indicator still should be duty cycle decrease 0: Disable 1: Enable
1	/	/	/
0	R/W	0x0	PWMBL_OCP_TRIG_INT_EN Output overcurrent interrupt enable 0: Disable 1: Enable

15.3.8.9 0x0044+N*0x0200 (N=0~1) PWMBL Interrupt Pending Register (Default Value: 0x0000_0000)

Offset: 0x0044+N*0x0200 (N=0~1)			Register Name: PWMBL_INT_PENDING_REG
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/
3	R/W1C	0x0	PWMBL_ACT_CYCLE_MAX_INT_PENDING When PWM_BL duty cycle is equal to PWM_MAX_ACT_CYCLE, enable this bit. 0: Disable 1: Enable
2	R/W1C	0x0	PWMBL_ACT_CYCLE_ERROR_INT_PENDING PWM_BL duty cycle error interrupt pending. When the duty cycle is 0, the analog output indicator still should be duty cycle decrease. 0: Disable 1: Enable
1	/	/	/

Offset: 0x0044+N*0x0200 (N=0~1)			Register Name: PWMBL_INT_PENDING_REG
Bit	Read/Write	Default/Hex	Description
0	R/W1C	0x0	PWMBL_OCP_TRIG_INT_PENDING Output overcurrent interrupt pending 0: No pending. 1: Pending.

15.3.8.10 0x0050+N*0x0200 (N=0~1) PWMBL Analog 0 Register (Default Value: 0x0000_0000)

Offset: 0x0050+N*0x0200 (N=0~1)			Register Name: PWMBL_ANA0_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	PWMBL_BST_EN PWM_BL BST Enable 0: PWM disable 1: PWM enable.

15.3.8.11 0x0054+N*0x0200 (N=0~1) PWMBL Analog 1 Register (Default Value: 0x0000_0000)

Offset: 0x0054+N*0x0200 (N=0~1)			Register Name: PWMBL_ANA1_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	PWMBL_OCP_DISABLE OCP Disable 0: OCP enable 1: OCP disable.
30:24	/	/	/
23:16	R/W	0x0	PWMBL_DIMM. BST load current configuration is used for adjusting screen brightness based on vref_max value. The screen brightness has 256 levels. 0x0: Iload_min_1(The current value required by minimum screen brightness) 0x1: (Iload_max_1-Iload_min_1)*1/128 ... 0xFF: Iload_max_1
15:3	/	/	/

Offset: 0x0054+N*0x0200 (N=0~1)			Register Name: PWMBL_ANA1_REG
Bit	Read/Write	Default/Hex	Description
2:0	R/W	0x0	PWMBL_VREF_MAX_SEL The reference voltage required by maximum screen brightness. It is 0.2V by default, which is implemented through an external 10-ohm resistor with a maximum 20mA load current. 000: 0.2V 001: 0.3V 010: 0.4V 011: 0.5V 100: 0.6V 101: 0.7V 110: 0.8V

15.3.8.12 0x0058+N*0x0200 (N=0~1) PWMBL Analog 2 Register (Default Value: 0x0000_0000)

Offset: 0x0058+N*0x0200 (N=0~1)			Register Name: PWMBL_ANA2_REG
Bit	Read/Write	Default/Hex	Description
31:2	/	/	/
1:0	R/W	0x0	PWMBL_FREQ_CFG BST working frequency configuration 00: 1 MHz 01: 0.75 MHz 10: 0.5 MHz 11: 0.25 MHz

15.3.8.13 0x005C+N*0x0200 (N=0~1) PWMBL Analog 3 Register (Default Value: 0x0000_0000)

Offset: 0x005C+N*0x0200 (N=0~1)			Register Name: PWMBL_ANA3_REG
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/
2:0	R/W	0x0	PWMBL_IBIAS Ibias Configuration 000: 1uA 001: 0.9uA 010: 0.82uA 011: 0.75uA 100: 1.7uA 101: 1.45uA 110: 1.26uA 111: 1.11uA

15.3.8.14 0x0060+N*0x0200 (N=1) PWMBL Reserved Register (Default Value: 0x0000_0000)

Offset: 0x0060+N*0x0200 (N=1)			Register Name: PWMBL_RSD_REG
Bit	Read/Write	Default/Hex	Description
31:10	/	/	/
9	R/W	0x0	DIS_GATING_1 0: Disable analog circuit gating. It is 0 by default. 1: Enable PWM_BL1 digital circuit.
8:7	R/W	0x0	SW_CLK_1 Adjust switch capacitor clock frequency with the default equivalent resistance set to 125M. 00: 8k 01: 16k 10: 31k 11: 4k
6	R/W	0x0	BST_CONFIG_1 Adjust trim_ibias through ibias value detection. It is 0 by default. When it is 1, the detected ideal voltage value is 12uA*R by connecting a resistor to the FB terminal. Then, ibias_trim_1<2:0> is adjusted based on the voltage value.
5:3	R/W	0x0	BST_CONFIG_1 trim_ramp. It is 0 by default. 000:1.17V 001:1.31V 010:1.46V 011:1.6V 100:0.59V 101:0.74V 110:0.88V 111:1.02V
2	R/W	0x0	BST_CONFIG_1 Whether to amplify the vfb_1 by 2 times and then output as fb_bst_1 to GPADC. 0: Amplify the vfb_1 by 2 times. The bit is 0 by default. 1: Not amplify the vfb_1 by 2 times.
1	R/W	0x0	BST_CONFIG_1 Whether to amplify the vfb_0 by 2 times and then output as fb_bst_1 to GPADC. 0: Amplify the vfb_0 by 2 times. The bit is 0 by default. 1: Not amplify the vfb_0 by 2 times.
0	/	/	/

15.3.8.15 0x0100+N*0x0200 (N=0~1) PWMBL GPADC Control Register (Default Value: 0x0000_0000)

Offset: 0x0100+N*0x0200 (N=0~1)			Register Name: PWMBL_GPADC_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	PWMBL_GPADC_EN PWM_BL is implemented by comparing between feedback voltage and reference voltage 0: Implemented by the BST circuit inside the PWM_BL. 1: Implemented by the GPADC quantized voltage and GPADC reference voltage configured by registers.

15.3.8.16 0x0104+N*0x0200 (N=0~1) PWMBL GPADC Voltage Reference Register (Default Value: 0x0000_0000)

Offset:0x0104+N*0x0200 (N=0~1)			Register Name: PWMBL_GPADC_VF_REG
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R/W	0x0	PWM_GPADC_VF Compare the GPADC quantized voltage with GPADC reference voltage and compute difference value The analog voltage is quantized to a 12-bit digital value through register configuration, and the configuration method should be the same with the GPADC quantized method.

15.3.8.17 0x0110+N*0x0200 (N=0~1) PWMBL GPADC Voltage0 Register (Default Value: 0x0000_0000)

Offset:0x0110+N*0x0200 (N=0~1)			Register Name: PWMBL_GPADC_V0_REG
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R/W	0x0	PWM_GPADC_V0 The difference between the quantized voltage and the reference voltage is categorized into levels, and the register holds the value for level 0.

15.3.8.18 0x0114+N*0x0200 (N=0~1) PWMBL GPADC Voltage1 Register (Default Value: 0x0000_0000)

Offset:0x0114+N*0x0200 (N=0~1)			Register Name: PWMBL_GPADC_V1_REG

Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R/W	0x0	PWM_GPADC_V1 The difference between the quantized voltage and the reference voltage is categorized into levels, and the register holds the value for level 1.

15.3.8.19 0x0118+N*0x0200 (N=0~1) PWMBL GPADC Voltage2 Register (Default Value: 0x0000_0000)

Offset:0x0118+N*0x0200 (N=0~1)			Register Name: PWMBL_GPADC_V2_REG
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R/W	0x0	PWM_GPADC_V2 The difference between the quantized voltage and the reference voltage is categorized into levels, and the register holds the value for level 2.

15.3.8.20 0x0120+N*0x0200 (N=0~1) PWMBL GPADC Active Cycle Step0 Register (Default Value: 0x0000_0000)

Offset:0x0120+N*0x0200 (N=0~1)			Register Name: PWMBL_GPADC_AC_STEP0_REG
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x0	PWM_GPADC_ACT_CYCLE_STEP0 The difference between the quantized voltage and the reference voltage is categorized into levels. The following is the duty cycle adjustment step when the difference is within the level 0. 0 = 0 cycle 1 = 1 cycles N = N cycles

15.3.8.21 0x0124+N*0x0200 (N=0~1) PWMBL GPADC Active Cycle Step1 Register (Default Value: 0x0000_0000)

Offset:0x0124+N*0x0200 (N=0~1)			Register Name: PWMBL_GPADC_AC_STEP1_REG
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/

Offset:0x0124+N*0x0200 (N=0~1)			Register Name: PWMBL_GPADC_AC_STEP1_REG
Bit	Read/Write	Default/Hex	Description
15:0	R/W	0x0	PWM_GPADC_ACT_CYCLE_STEP1 The difference between the quantized voltage and the reference voltage is categorized into levels. The following is the duty cycle adjustment step when the difference is within the level 1. 0 = 0 cycle 1 = 1 cycles N = N cycles

15.3.8.22 0x0128+N*0x0200 (N=0~1) PWMBL GPADC Active Cycle Step2 Register (Default Value: 0x0000_0000)

Offset:0x0128+N*0x0200 (N=0~1)			Register Name: PWMBL_GPADC_AC_STEP2_REG
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x0	PWM_GPADC_ACT_CYCLE_STEP2 The difference between the quantized voltage and the reference voltage is categorized into bins. The following is the duty cycle adjustment level when the difference is within the level 2. 0 = 0 cycle 1 = 1 cycles N = N cycles

15.3.8.23 0x012C+N*0x0200 (N=0~1) PWMBL GPADC Active Cycle Step3 Register (Default Value: 0x0000_0000)

Offset:0x012C+N*0x0200 (N=0~1)			Register Name: PWMBL_GPADC_AC_STEP3_REG
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/

Offset:0x012C+N*0x0200 (N=0~1)			Register Name: PWMBL_GPADC_AC_STEP3_REG
Bit	Read/Write	Default/Hex	Description
15:0	R/W	0x0	PWM_GPADC_ACT_CYCLE_STEP3. The difference between the quantized voltage and the reference voltage is categorized into levels. The following is the duty cycle adjustment step when the difference is within the level 3. 0 = 0 cycle 1 = 1 cycles N = N cycles



15.4 Two Wire Interface (TWI)

15.4.1 Overview

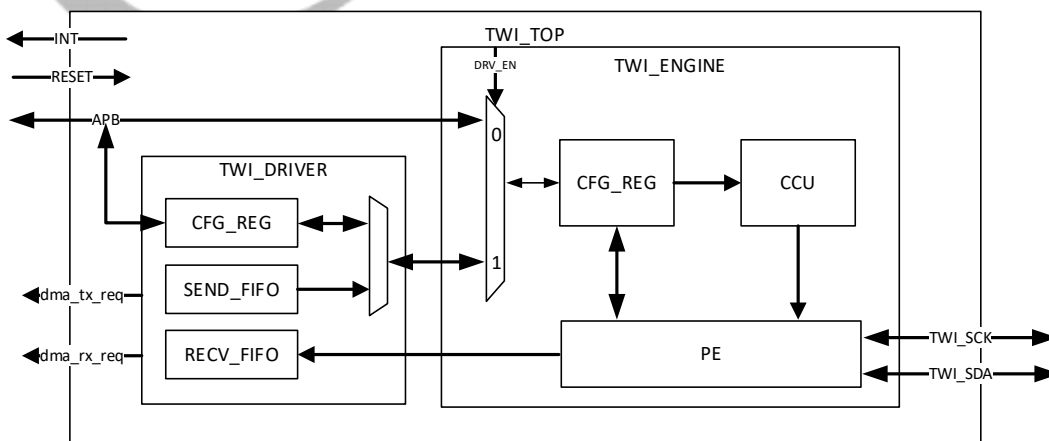
This Two Wire Interface (TWI) Controller is used to connect SoC on the 2-wire bus. It can support all the standard 2-wire bus transfer. Master mode and Slave mode are all supported. The TWI has the following features:

- Up to 3TWI controllers (TWI0, TWI1, TWI2)
- Compliant with I2C bus standard
- Support master mode or slave mode
- Supports 7-bit standard address and 10-bit extended address.
- Supports standard mode (100 kbit/s) and high-speed mode (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
- Salve mode features:
 - Interrupt on address detection
- The TWI controller includes one TWI engine and one TWI driver. And the TWI driver supports packet transmission and DMA mode when TWI works in master mode

15.4.2 Block Diagram

The following figure shows the block diagram of TWI.

Figure 15-19 TWI Block Diagram



TWI contains the following sub-blocks:

Table 15-7 TWI Sub-blocks

Sub-block	Description
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
SEND_FIFO	The register address bytes and the written data bytes are buffered in SEND_FIFO
RECV_FIFO	The read data bytes are buffered in RECV_FIFO

The controller includes TWI engine and TWI driver. Each time the TWI engine sends a START signal, a STOP signal, or a BYTE data, or a corresponding ACK, the TWI engine will generate an interrupt, and wait for the CPU to process and clear the interrupt before the next START, STOP, or BYTE, ACK transmission can be performed. Therefore, when a device communication is completed, many interrupts will be generated, and the CPU needs to wait for the previous interrupt before it can configure the next one. The TWI driver defines each communication with the device as a packet transmission. The CPU can directly configure the slave address, register address and data transmission for one or more package transmissions without waiting for interruption, then start the TWI driver, and the TWI driver can control the TWI engine to complete a pre-configured communication, and report an interrupt to the CPU after completion.

15.4.3 External Signals

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 configured as a slave device, the TWIn-SCK is an input pin.

Table 15-8 TWI External Signals

Signal Name	Description	Type
TWI0-SCK	TWI0 Serial Clock Signal	I/O
TWI0-SDA	TWI0 Serial Data Signal	I/O
TWI1-SCK	TWI1 Serial Clock Signal	I/O
TWI1-SDA	TWI1 Serial Data Signal	I/O
TWI2-SCK	TWI2 Serial Clock Signal	I/O
TWI2-SDA	TWI2 Serial Data Signal	I/O
TWI3-SCK	TWI3 Serial Clock Signal	I/O
TWI3-SDA	TWI3 Serial Data Signal	I/O

15.4.4 Functional Description

15.4.4.1 TWI Engine Master and Slave Mode in 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.

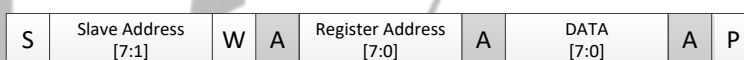
When the CPU host wants to start a bus transfer, it initiates a bus START to enter the master mode by setting IM_STA bit in the 2WIRE_CNTR register to high (before it must be low). The TWI Engine will assert INT line and INT_FLAG to indicate a completion for the START condition and each consequent byte transfer. At each interrupt, the micro-processor needs to check the 2WIRE_STAT register 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 2WIRE_DATA data register, and set the 2WIRE_CNTR control register. 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 byte transfer or START condition.

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

Figure 15-20 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

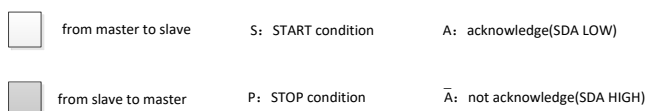
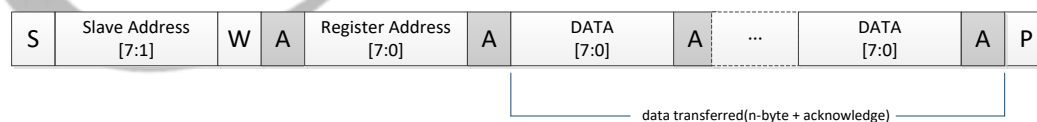
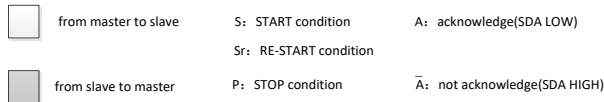
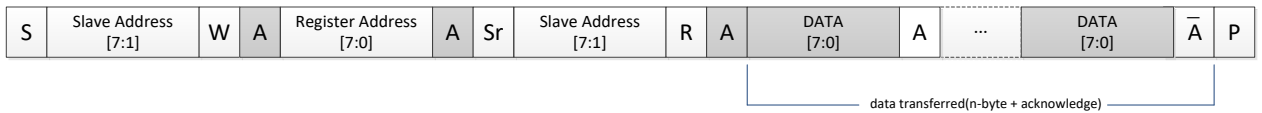


Figure 15-21 Read Timing in 7-bit Standard Addressing Mode

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



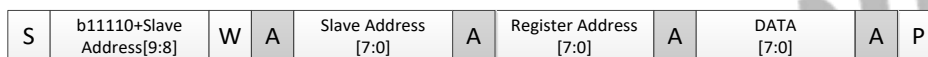
Slave addr = 7-bit , register addr = 8-bit, data = n-byte



The following figures describe the write/read timing in 10-bit extended address mode.

Figure 15-22 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

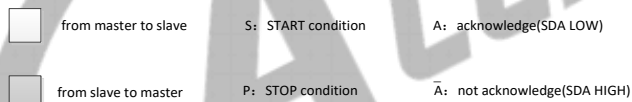
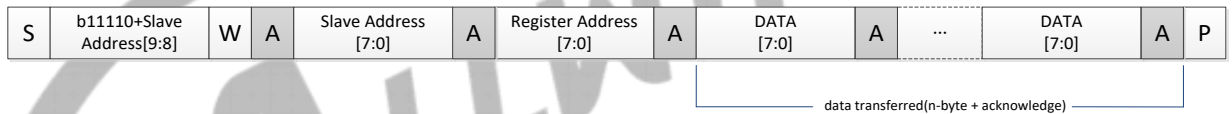
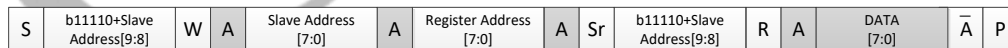
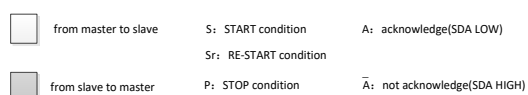


Figure 15-23 Read Timing in 10-bit Extended Addressing Mode

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



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



15.4.4.2 Write/Read Packet Transmission in 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 15-24 TWI Driver Write Packet Transmission

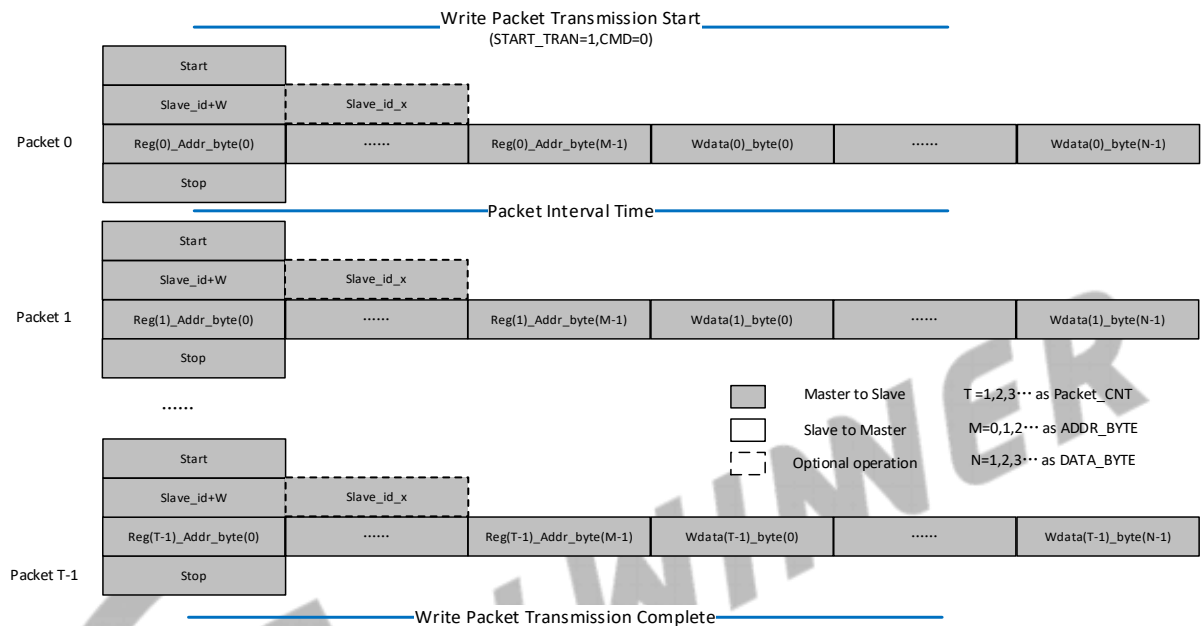
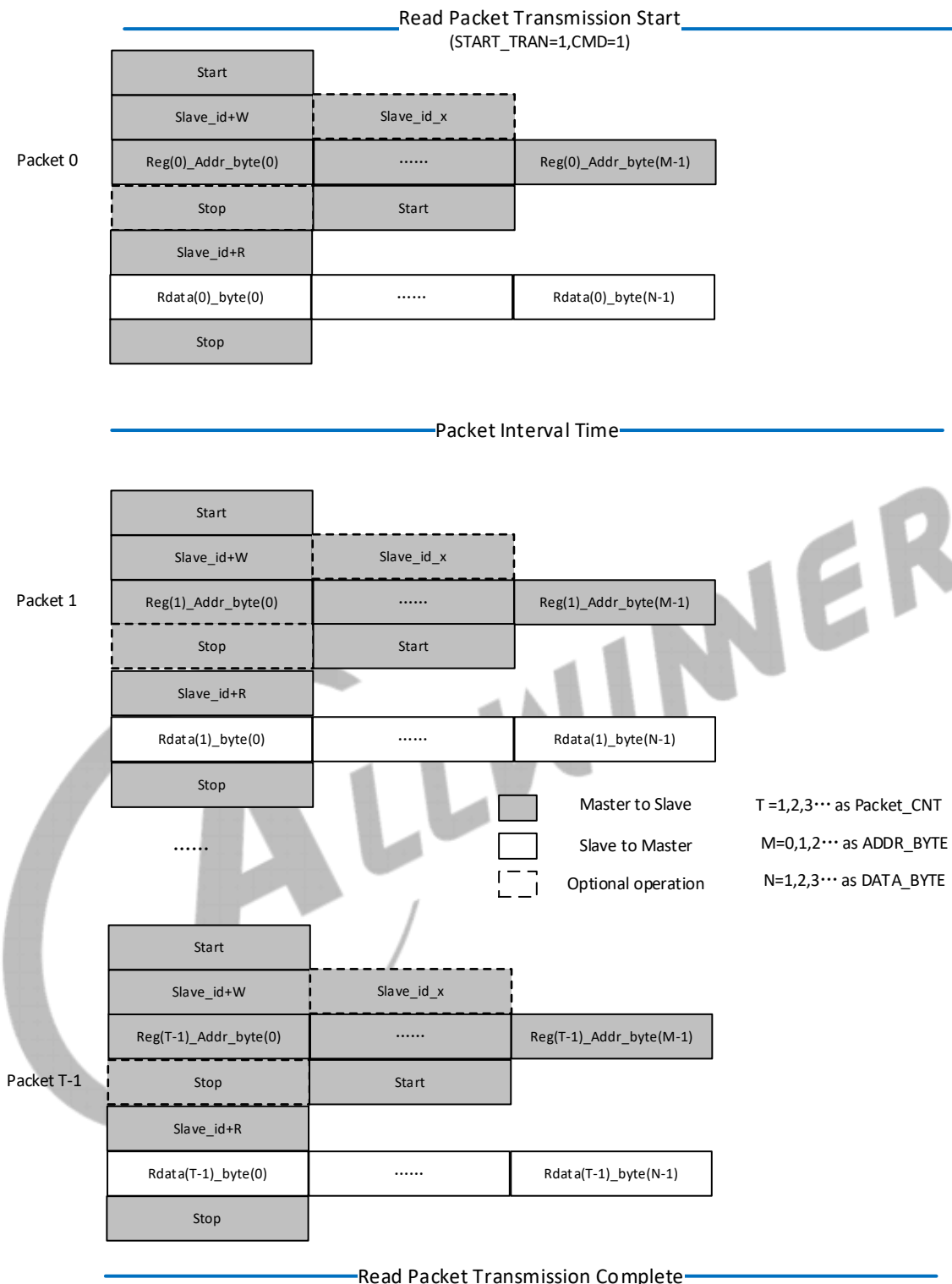


Figure 15-25 TWI Driver Read Packet Transmission



15.4.4.3 Programming State Diagram

Figure 15-26 shows the TWI programming state diagram. For the value between two states, see the [TWI_STAT](#) register.

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;

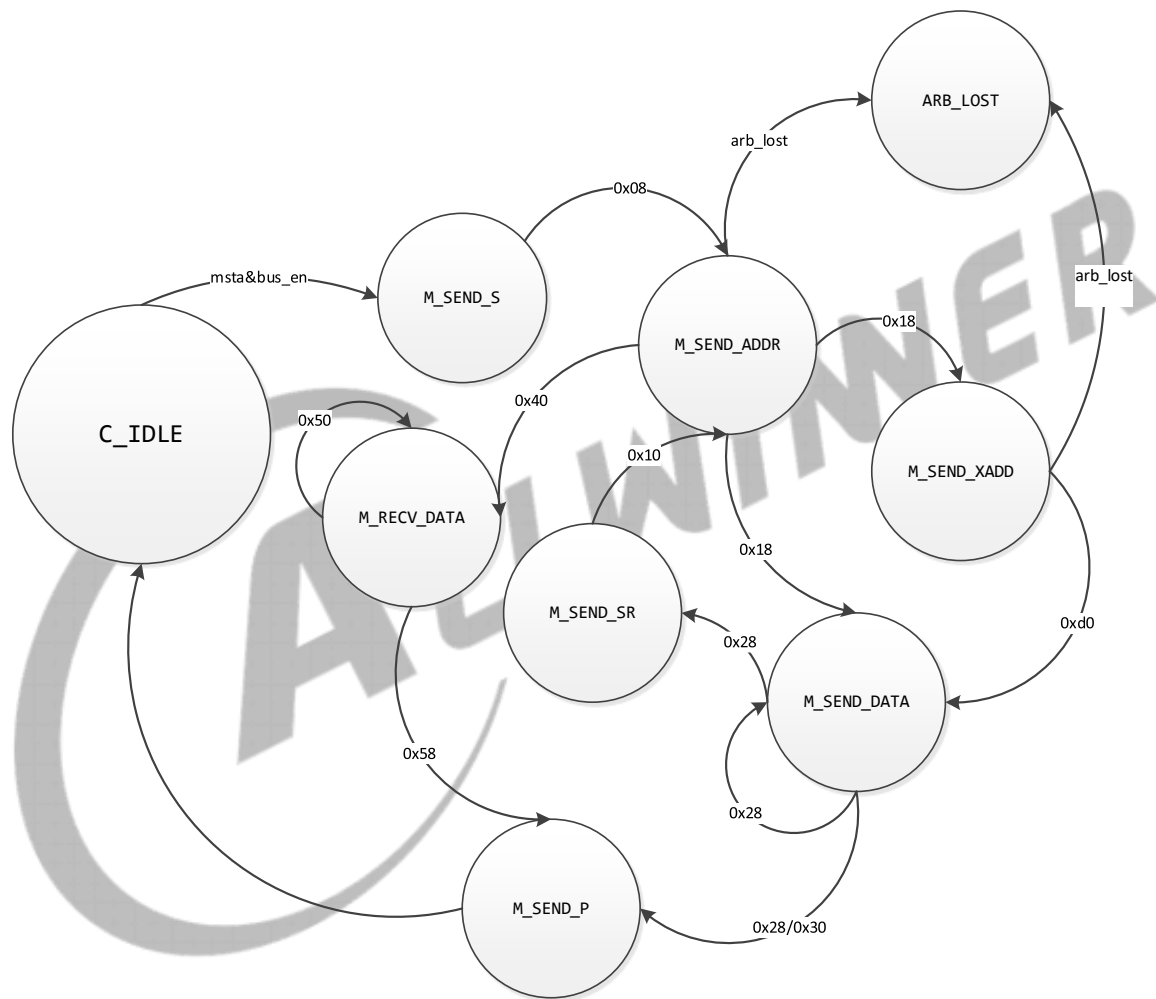
M_SEND_P: master sends STOP signal;

M_RECV_DATA: master receives data;

ARB_LOST: Arbitration lost;

C_IDLE: Idle.

Figure 15-26 TWI Programming State Diagram



15.4.5 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 15.4.7.1 and 15.4.7.2.

15.4.5.1 TWI Engine Initialization

Follow the steps below to initialize the TWI engine.

- Step 1** Configure the corresponding GPIO multiplex function as TWI mode.
- Step 2** For TWIn, set `TWI_BGR_REG[TWIn_GATING]` in CCU module to 0 to close TWIn clock.
- Step 3** For TWIn, set `TWI_BGR_REG[TWIn_RST]` in CCU module to 0, then set to 1 to reset TWIn.
- Step 4** For TWIn, set `TWI_BGR_REG[TWIn_GATING]` in CCU module to 1 to open TWIn clock.
- Step 5** Configure [TWI_CCR\[CLK_M\]](#) and [TWI_CCR\[CLK_N\]](#) to get the needed rate (The clock source of TWI is from APB1).
- Step 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 GIC module. In slave mode, configure [TWI_ADDR](#) and [TWI_XADDR](#) registers to finish TWI initialization configuration.

15.4.5.2 Writing Data Operation in TWI Engine

Follow the steps below to write data to the device.

- Step 1** Clear [TWI_EFR](#) register, and configure [TWI_CNTR\[M_STA\]](#) to 1 to transmit the START signal.
- Step 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).
- Step 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).
- Step 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](#)).
- Step 5** After transmission completes, write [TWI_CNTR\[M_STP\]](#) to 1 to transmit the STOP signal and end this write-operation.

15.4.5.3 Reading Data Operation in TWI Engine

Follow the steps below to read data from the device.

- Step 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.

- Step 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).
- Step 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).
- Step 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.
- Step 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.
- Step 6** Write [TWI_CNTR\[M_STP\]](#) to 1 to transmit the STOP signal and end this read-operation.

15.4.5.4 TWI Driver Initialization

Follow the steps below to initialize the TWI driver.

- Step 1** Configure corresponding GPIO multiplex function as TWI mode.
- Step 2** For TWIn, set TWIn_BGR_REG [TWIn_GATING] in CCU module to 0 to close TWIn clock.
- Step 3** For TWIn, set TWIn_BGR_REG [TWIn_RST] in CCU module to 0, then set to 1 to reset TWIn.
- Step 4** For TWIn, set TWIn_BGR_REG [TWIn_GATING] in CCU module to 1 to open TWIn clock.
- Step 5** Set [TWI_DRV_CTRL](#) [TWI_DRV_EN] to 1 to enable the TWI driver.
- Step 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).
- Step 7** Set [TWI_DRV_CTRL](#)[RESTART_MODE] to 0 and [READ_TRAN_MODE] to 1, and set [TWI_DRV_INT_CTRL](#)[TRAN_COM_INT_EN] to 1.
- Step 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.

15.4.5.5 Writing Packet Transmission for TWI Driver

Follow the steps below to write package to the device.

- Step 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.

- Step 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.
- Step 3** Configure [TWI_DRV_CFG](#)[PACKET_CNT] to set the written packet number.
- Step 4** Configure DMA channel, including TWI TXFIFO, device register address, and the written data.
- Step 5** Set [START_TRAN] to 1 to start TWI Driver transmission.
- Step 6** When TWI driver transmission completes, the interrupt is triggered, it indicates that the write packet transmission ends.

15.4.5.6 Reading Packet Transmission for TWI Driver

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

- Step 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.
- Step 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.
- Step 3** Configure [TWI_DRV_CFG](#)[PACKET_CNT] to set the read packet number.
- Step 4** Configure DMA channel, including TWI TXFIFO, TWI RXFIFO, device register address and the read data.
- Step 5** Set TWI_DRV_CTRL [START_TRAN] to 1 to start TWI Driver transmission.
- Step 6** When TWI driver transmission completes, the interrupt is triggered, it indicates that the read packet transmission ends.

15.4.6 Register List

Module Name	Base Address	Comments
TWI0	0x02502000	
TWI1	0x02502400	TWI1 register is the same with TWI0.
TWI2	0x02502800	TWI2 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

Register Name	Offset	Description
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

15.4.7 Register Description

15.4.7.1 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

NOTE

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.

15.4.7.2 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]

15.4.7.3 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

15.4.7.4 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.

Offset: 0x000C			Register Name: TWI_CNTR
Bit	Read/Write	Default/Hex	Description
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.
4	R/WAC	0x0	M_STP Master Mode Stop 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. The M_STP bit is cleared automatically. Writing a '0' to this bit has no effect.

Offset: 0x000C			Register Name: TWI_CNTR
Bit	Read/Write	Default/Hex	Description
3	R/W1C	0x0	INT_FLAG Interrupt Flag 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.
2	R/W	0x0	A_ACK Assert Acknowledge 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: (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. 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. 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. The TWI will not respond as a slave unless A_ACK is set.
1	/	/	/
0	R/W	0x0	CLK_COUNT_MODE 0: scl clock high period count on oscl 1: scl clock high period count on iscl

15.4.7.5 0x0010 TWI Status Register (Default Value: 0x0000_00F8)

Offset: 0x0010	Register Name: TWI_STAT

Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R	0xF8	STA Status Information Byte Code 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 transmitted 0x58: Data byte received in master mode, not ACK transmitted 0x60: Slave address + Write bit received, ACK transmitted 0x68: Arbitration lost in the address as master, slave address + Write bit received, ACK transmitted 0x70: General Call address received, ACK transmitted 0x78: Arbitration lost in the address as master, General Call address received, ACK transmitted 0x80: Data byte received after slave address received, ACK transmitted 0x88: Data byte received after slave address received, not ACK transmitted 0x90: Data byte received after General Call received, ACK transmitted 0x98: Data byte received after General Call received, not ACK transmitted 0xA0: STOP or repeated START condition received in slave mode 0xA8: Slave address + Read bit received, ACK transmitted 0xB0: Arbitration lost in the address as master, slave address + Read bit received, ACK transmitted

			0xB8: Data byte transmitted in slave mode, ACK received 0xC0: Data byte transmitted in slave mode, ACK not received 0xC8: The Last byte transmitted in slave mode, ACK received 0xD0: Second Address byte + Write bit transmitted, ACK received 0xD8: Second Address byte + Write bit transmitted, ACK not received 0xF8: No relevant status information, INT_FLAG=0 Others: Reserved

15.4.7.6 0x0014 TWI Clock Register (Default Value: 0x0000_0080)

Offset: 0x0014			Register Name: TWI_CCR
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8	R/W	0x0	CLK_DUTY_30_EN Enabling duty cycle 30% of Clock as Master 0: Bit [7] takes effect 1: 30% Configure the duty cycle as 30% when the rate is 400 kbit/s
7	R/W	0x1	CLK_DUTY Setting duty cycle of clock as master 0: 50% 1: 40% Configure the duty cycle as 50% when the rate is 100 kbit/s
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.

Offset: 0x0014			Register Name: TWI_CCR
Bit	Read/Write	Default/Hex	Description
2:0	R/W	0x0	CLK_N The TWI bus is sampled by the TWI at the frequency defined by F0: $F_{\text{samp}} = F_0 = F_{\text{in}} / 2^{\text{CLK\_N}}$ The TWI SCL output frequency, in master mode, is F1/10: $F_1 = F_0 / (\text{CLK\_M} + 1)$ $F_{\text{scl}} = F_1 / 10 = F_{\text{in}} / (2^{\text{CLK\_N}} * (\text{CLK\_M} + 1) * 10)$ Specially, $F_{\text{scl}} = F_1 / 11$ when CLK_M=0 and CLK_DUTY=40% due to the delay of SCL sample debounce. For Example: $F_{\text{in}} = 24 \text{ MHz}$ (APB clock input) For 400 kHz full speed 2-wire, CLK_N = 1, CLK_M = 2 $F_0 = 24 \text{ MHz} / 2^1 = 12 \text{ MHz}, F_1 = F_0 / (10 * (2 + 1)) = 0.4 \text{ MHz}$ For 100 kHz standard speed 2-wire, CLK_N = 1, CLK_M = 11 $F_0 = 24 \text{ MHz} / 2^1 = 12 \text{ MHz}, F_1 = F_0 / (10 * (11 + 1)) = 0.1 \text{ MHz}$ In SLAVE mode: When the APB clock is 100 MHz, and the TWI clock is 100 KHz, It is recommended to configure CLK_N to 4. When the APB clock is 100 MHz, and the TWI clock is 400 KHz, It is recommended to configure CLK_N to 2. When the APB clock is 24 MHz, and the TWI clock is 100 KHz, It is recommended to configure CLK_N to 2. When the APB clock is 24 MHz, and the TWI clock is 400 KHz, It is recommended to configure CLK_N to 1.

15.4.7.7 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.

15.4.7.8 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	DBN Data Byte Number Follow Read Command Control 00: No data byte can be written after the read command 01: Only 1-byte data can be written after the read command 10: 2-bytes data can be written after the read command 11: 3-bytes data can be written after the read command

15.4.7.9 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	SCL_STATE Current State of TWI_SCL 0: Low 1: High
4	R	0x1	SDA_STATE Current State of TWI_SDA 0: Low 1: High
3	R/W	0x1	SCL_CTL TWI_SCL Line State Control Bit When the line control mode is enabled (bit[2] is set), this bit decides the output level of TWI_SCL. 0: Output low level 1: Output high level
2	R/W	0x0	SCL_CTL_EN TWI_SCL Line State Control Enable When this bit is set, the state of TWI_SCL is controlled by the value of bit[3]. 0: Disable TWI_SCL line control mode 1: Enable TWI_SCL line control mode

Offset: 0x0020			Register Name: TWI_LCR
Bit	Read/Write	Default/Hex	Description
1	R/W	0x1	SDA_CTL TWI_SDA Line State Control Bit When the line control mode is enabled (bit[0] is set), this bit decides the output level of TWI_SDA. 0: Output low level 1: Output high level
0	R/W	0x0	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

15.4.7.10 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	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.
30	/	/	/
29	R/W	0x0	RESTART_MODE Restart mode 0: RESTART 1: STOP+START Define the TWI_DRV action after sending the register address.
28	R/W	0x0	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.

Offset: 0x0200			Register Name: TWI_DRV_CTRL
Bit	Read/Write	Default/Hex	Description
27:24	R	0x0	TRAN_RESULT Transition result 000: OK 001: FAIL Other: Reserved
23:16	R	0xf8	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 9 th SCL clock Other: Reserved
15:8	R/W	0x10	TIMEOUT_N Timeout number When sending the 9 th clock, assert fail signal when the slave device does not respond after $N \cdot F_{SCL}$ cycles. And the software must do a reset to the TWI_DRV module and send a stop condition to slave.
7:2	/	/	/
1	R/W	0x0	SOFT_RESET Software reset 0: Normal 1: Reset
0	R/W	0x0	TWI_DRV_EN TWI driver enable 0: Module disable 1: Module enable (only use in TWI Master Mode)

15.4.7.11 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	PKT_INTERVAL Define the interval between each packet for PKT_INTERVAL F_{SCL} cycles.
15:0	R/W	0x1	PACKET_CNT The FIFO data is transmitted as PACKET_CNT packets in current format.

15.4.7.12 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.

15.4.7.13 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_BYTE How many bytes be sent as slave device reg address 0-255
15:0	R/W	0x1	DATA_BYTE How many bytes be sent/received as data 1-65535

15.4.7.14 0x0210 TWI_DRV Bus Control Register (Default Value: 0x0000_80C0)

Offset: 0x0210			Register Name: TWI_DRV_BUS_CTRL
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17	R/W	0x0	CLK_DUTY_30_EN Enabling duty cycle 30% of Clock as Master 0: Bit [15] takes effect 1: 30% The duty cycle is configured as 30% when the rate is 400 kbit/s.
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
15	R/W	0x1	CLK_DUTY Setting duty cycle of clock as Master 0: 50% 1: 40% The duty cycle is configured as 50% when the rate is 100 kbit/s.
14:12	R/W	0x0	CLK_N TWI_DRV bus sampling clock $F_0 = 24\text{MHz} / 2^{\text{CLK\_N}}$
11:8	R/W	0x0	CLK_M TWI_DRV output SCL frequency is $F_{\text{SCL}} = F_1 / 10 = (F_0 / (\text{CLK\_M} + 1)) / 10$ Specially, $F_{\text{oscl}} = F_1 / 11$ 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

15.4.7.15 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.
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

15.4.7.16 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

Offset: 0x0218			Register Name: TWI_DRV_DMA_CFG
Bit	Read/Write	Default/Hex	Description
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.

15.4.7.17 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
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

15.4.7.18 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.

15.4.7.19 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	/	/	/

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

ALLWINNER

15.5 General Purpose ADC (GPADC)

15.5.1 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.

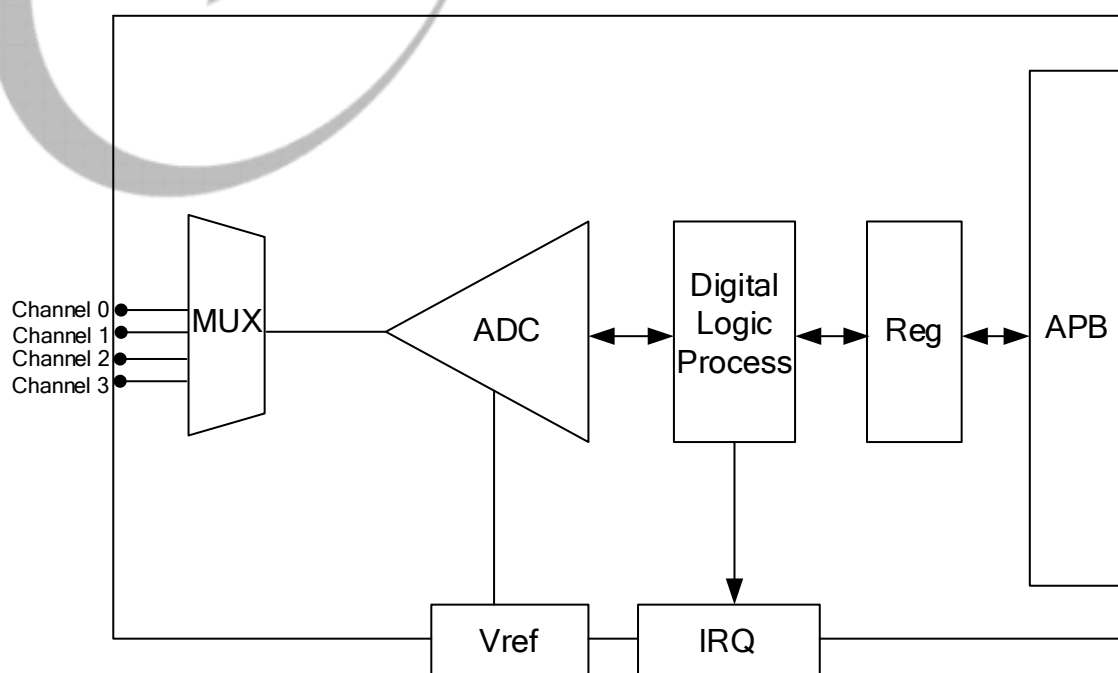
The GPADC has the following features:

- Successive-Approximation Register ADC (SAR ADC)
- Includes 1 GPADC: GPADC0 has 1 pads, multiplexing GPIOA(0~3)
- 64 FIFO depth of data register
- 12-bit sampling resolution
- Up to 1 MHz sampling frequency
- Input voltage: 0 V to 1.8 V.
- Supports data comparison and interrupts
- Two operation modes
 - Single conversion mode
 - Continuous conversion mode

15.5.2 Block Diagram

The following table shows the block diagram of the GPADC.

Figure 15-27 GPADC Block Diagram



15.5.3 External Signals

The following table describes the external signals of the GPADC.

Table 15-9 GPADC External Signals

Signal Name	Description	Type
GPADC	General Purpose ADC Input	AI

15.5.4 Functional Description

15.5.4.1 Operation Mode

- Single conversion mode

The GPADC samples and converts once in a specified channel, and then the converted data is updated at the data registers of the corresponding channels.

- Continuous conversion mode

The GPADC keeps sampling and converting in a specified channel until the ADC is disabled, and then the converted data is updated at the data registers of the corresponding channels.

- Burst conversion mode

The GPADC keeps sampling and converting in a specified channel until the ADC is disabled, and then the converted data is sequentially stored at the FIFO.

15.5.4.2 Clock and Timing Requirements

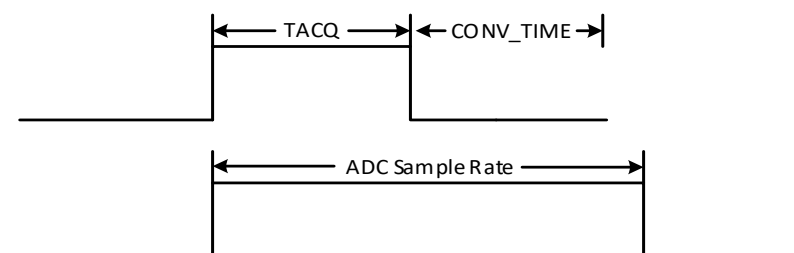
CLK_IN = 24 MHz

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

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

ADC Sample Frequency > TACQ+CONV_TIME

Figure 15-28 GPADC Timing Requirement



15.5.4.3 GPADC Calculation Formula

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

Where:

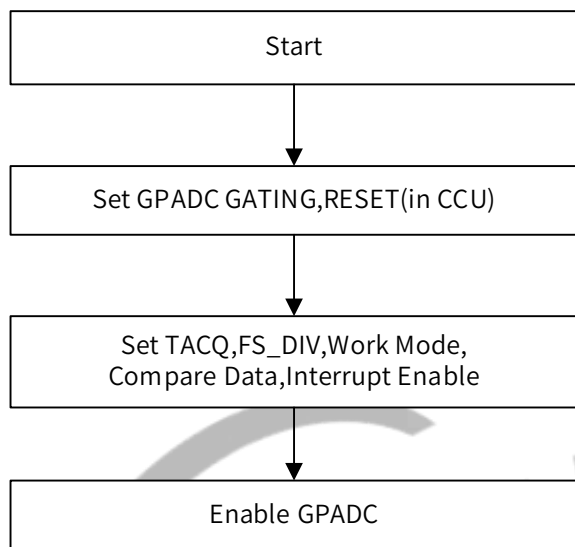
$V_{REF} = 1.8V$

15.5.5 Programming Guidelines

15.5.5.1 Initialization

The GPADC initial process is as follows.

Figure 15-29 GPADC Initial Process



Query Mode

Take channel0 for example:

- Step 1** Write 0x1 to the bit [16] of [GPADC_BGR_REG](#) to deassert reset.
- Step 2** Write 0x1 to the bit [0] of [GPADC_BGR_REG](#) to enable the GPADC clock.
- Step 3** Write 0x1 to the bit [0] of [GP_CTRL](#) to enable the GPADC VCM-buffer enable.
- Step 4** Write 0x2F to the bit [15:0] of [GP_SR_CON](#) to set the acquiring time of ADC.
- Step 5** Write 0x1DF to the bit [31:16] of [GP_SR_CON](#) to set the ADC sample frequency divider.
- Step 6** Write 0x2 to the bit [19:18] of [GP_CTRL](#) to set the continuous conversion mode.
- Step 7** Write 0x1 to the bit [0] of [GP_CS_EN](#) to enable the analog input channel.
- Step 8** Write 0x1 to the bit [16] of [GP_CTRL](#) to enable the ADC function.
- Step 9** Read the bit [0] of [GP_DATA_INTS](#). If the bit is 1, then data conversion is complete.
- Step 10** Read the bit [11:0] of [GP_CH0_DATA](#), and calculate voltage value based on GPADC formula.

Interrupt Mode

Take channel0 for example:

- Step 1** Write 0x1 to the bit [16] of [GPADC_BGR_REG](#) to deassert reset.
- Step 2** Write 0x1 to the bit [0] of [GPADC_BGR_REG](#) to enable the GPADC clock.
- Step 3** Write 0x1 to the bit[0] of [GP_CTRL](#) to enable the GPADC VCM-buffer enable.
- Step 4** Write 0x2F to the bit [15:0] of [GP_SR_CON](#) to set the acquiring time of ADC.
- Step 5** Write 0x1DF to the bit [31:16] of [GP_SR_CON](#) to set the ADC sample frequency divider.
- Step 6** Write 0x2 to the bit [19:18] of [GP_CTRL](#) to set the continuous conversion mode.
- Step 7** Write 0x1 to the bit [0] of [GP_CS_EN](#) to enable the analog input channel.
- Step 8** Write 0x1 to the bit [0] of [GP_DATA_INTC](#) to enable the GPADC data interrupt.
- Step 9** Set interrupt based on PLIC module.
- Step 10** Put interrupt handler address into interrupt vector table based on the GPADC IRQ number.
- Step 11** Write 0x1 to the bit [16] of [GP_CTRL](#) to enable the ADC function.
- Step 12** Read the bit [11:0] of [GP_CH0_DATA](#) from the interrupt handler, calculate voltage value based on GPADC formula.

15.5.6 Register List

Module Name	Base Address
GPADC	0x02009000

Register Name	Offset	Description
GP_SR_CON	0x0000	GPADC Sample Rate Configure Register
GP_CTRL	0x0004	GPADC Control Register
GP_CS_EN	0x0008	GPADC Compare and Select Enable Register
GP_FIFO_INTC	0x000C	GPADC FIFO Interrupt Control Register
GP_FIFO_INTS	0x0010	GPADC FIFO Interrupt Status Register
GP_FIFO_DATA	0x0014	GPADC FIFO Data Register
GP_CDATA	0x0018	GPADC Calibration Data Register
GP_DATAH_INTC	0x0020	GPADC Data Low Interrupt Configure Register
GP_DATAH_INTC	0x0024	GPADC Data High Interrupt Configure Register
GP_DATA_INTC	0x0028	GPADC Data Interrupt Configure Register
GP_DATAH_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_CH0_CMP_DATA	0x0040	GPADC CH0 Compare Data Register

Register Name	Offset	Description
GP_CH1_CMP_DATA	0x0044	GPADC CH1 Compare Data Register
GP_CH2_CMP_DATA	0x0048	GPADC CH3 Compare Data Register
GP_CH3_CMP_DATA	0x004C	GPADC CH4 Compare Data Register
GP_CH0_DATA	0x0080	GPADC CH0 Data Register
GP_CH1_DATA	0x0084	GPADC CH1 Data Register
GP_CH3_DATA	0x0088	GPADC CH3 Data Register
GP_CH4_DATA	0x008C	GPADC CH4 Data Register

15.5.7 Register Description

15.5.7.1 0x0000 GPADC Sample Rate Configure Register (Default Value: 0x01DF_002F)

Offset: 0x0000			Register Name: GP_SR_CON
Bit	Read/Write	Default/Hex	Description
31:16	R/W	0x1DF	FS_DIV ADC Sample Frequency Divider $CLK\_IN/(n+3)$ Default value: 50K
15:0	R/W	0x2F	TACQ ADC Acquiring Time $(n+1)/CLK\_IN$ Default value: 2 us Note: The maximum of TACQ must not exceed 0x2F to ensure the accuracy of sampling.

15.5.7.2 0x0004 GPADC Control Register (Default Value: 0x0080_0000)

Offset: 0x0004			Register Name: GP_CTRL
Bit	Read/Write	Default/Hex	Description
31:24	R/W	0x0	ADC_FIRST_DLY ADC First Convert Delay setting ADC conversion of each channel is delayed by N samples.
23	R/W	0x1	ADC_AUTOCALI_EN ADC Auto Calibration
22	/	/	/
21:20	R/W	0x0	ADC_OP_BIAS ADC OP Bias Adjust the bandwidth of the ADC amplifier.

Offset: 0x0004			Register Name: GP_CTRL
Bit	Read/Write	Default/Hex	Description
19:18	R/W	0x0	GP_MODE GPADC Work Mode 00: Single conversion mode 01: Reserved 10: Continuous conversion mode 11: Burst conversion mode
17	R/WAC	0x0	ADC_CALI_EN ADC Calibration 1: Start Calibration, it is cleared to 0 after calibration
16	R/WAC	0x0	ADC_EN ADC Function Enable Before the bit is enabled, configure ADC parameters including the work mode and channel number, etc. 0: Disable 1: Enable When selecting a single conversion mode, the bit can be cleared automatically after the switch is completed.
15:1	/	/	/
0	R/W	0x0	GPADC_VCM_BUF_EN Gpadc built in VCM-buffer enable signal 0:Disable 1:Enable

15.5.7.3 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

Offset: 0x0008			Register Name: GP_CS_EN
Bit	Read/Write	Default/Hex	Description
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	/	/	/
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 0 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

15.5.7.4 0x000C GPADC FIFO Interrupt Control Register (Default Value: 0x0000_1F00)

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

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

15.5.7.5 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	/	/	/
14:8	R	0x0	RXA_CNT. ADC FIFO available Sample Word Counter
7:0	/	/	/

15.5.7.6 0x0014 GPADC FIFO Data Register (Default Value: UDF)

Offset: 0x0014			Register Name: GP_FIFO_DATA
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/

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

15.5.7.7 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

15.5.7.8 0x0020 GPADC Data Low Interrupt Configure Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: GP_DATAH_INTC
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
3	R/W	0x0	CH3_LOW_IRQ_EN Channel 3 Voltage Low Available Interrupt Enable 0: Disable 1: Enable
2	R/W	0x0	CH2_LOW_IRQ_EN Channel 2 Voltage Low Available Interrupt Enable 0: Disable 1: Enable
1	R/W	0x0	CH1_LOW_IRQ_EN Channel 1 Voltage Low Available Interrupt Enable 0: Disable 1: Enable
0	R/W	0x0	CH0_LOW_IRQ_EN Channel 0 Voltage Low Available Interrupt Enable 0: Disable 1: Enable

15.5.7.9 0x0024 GPADC Data High Interrupt Configure 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 Channel 3 Voltage High Available Interrupt Enable 0: Disable 1: Enable
2	R/W	0x0	CH2_HIG_IRQ_EN Channel 2 Voltage High Available Interrupt Enable 0: Disable 1: Enable
1	R/W	0x0	CH1_HIG_IRQ_EN Channel 1 Voltage High Available Interrupt Enable 0: Disable 1: Enable
0	R/W	0x0	CH0_HIG_IRQ_EN Channel 0 Voltage High Available Interrupt Enable 0: Disable 1: Enable

15.5.7.10 0x0028 GPADC Data Interrupt Configure Register (Default Value: 0x0000_0000)

Offset: 0x0028			Register Name: GP_DATA_INTC
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
3	R/W	0x0	CH3_DATA_IRQ_EN Channel 3 Data Available Interrupt Enable 0: Disable 1: Enable
2	R/W	0x0	CH2_DATA_IRQ_EN Channel 2 Data Available Interrupt Enable 0: Disable 1: Enable
1	R/W	0x0	CH1_DATA_IRQ_EN Channel 1 Data Available Interrupt Enable 0: Disable 1: Enable

Offset: 0x0028			Register Name: GP_DATA_INTC
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	CH0_DATA_IRQ_EN Channel 0 Data Available Interrupt Enable 0: Disable 1: Enable

15.5.7.11 0x0030 GPADC Data Low Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: GP_DATA1_INTS
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
3	R/W1C	0x0	CH3_LOW_PENDGING Channel 3 Voltage Low Available Interrupt Status 0: NO Pending IRQ 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_PENDGING Channel 2 Voltage Low Available Interrupt Status 0: NO Pending IRQ 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_PENDGING Channel 1 Voltage Low Available Interrupt Status 0: NO Pending IRQ 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	CH0_LOW_PENDGING Channel 0 Voltage Low Available Interrupt Status 0: NO Pending IRQ 1: Channel 0 Voltage Low Available Pending IRQ Write '1' to clear this interrupt or automatic clear if interrupt condition fails

15.5.7.12 0x0034 GPADC Data High Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0034			Register Name: GP_DATAH_INTS
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
3	R/W1C	0x0	CH3_HIG_PENGDING Channel 3 Voltage High Available Interrupt Status 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 Channel 2 Voltage High Available Interrupt Status 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 Channel 1 Voltage High Available Interrupt Status 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 Channel 0 Voltage High Available Interrupt Status 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

15.5.7.13 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	/	/	/

Offset: 0x0038			Register Name: GP_DATA_INTS
Bit	Read/Write	Default/Hex	Description
3	R/W1C	0x0	CH3_DATA_PENDING Channel 3 Data Available Interrupt Status 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_PENDING Channel 2 Data Available Interrupt Status 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_PENDING Channel 1 Data Available Interrupt Status 0: No Pending IRQ 1: Channel 1 Data Available Pending IRQ Write '1' to clear this interrupt or automatic clear if interrupt condition fails
0	R/W1C	0x0	CH0_DATA_PENDING Channel 0 Data Available Interrupt Status 0: No Pending IRQ 1: Channel 0 Data Available Pending IRQ Write '1' to clear this interrupt or automatic clear if interrupt condition fails

15.5.7.14 0x0040 GPADC CH0 Compare Data Register (Default Value: 0x0BFF_0400)

Offset: 0x0040			Register Name: GP_CH0_CMP_DATA
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0xBFF	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

15.5.7.15 0x0044 GPADC CH1 Compare Data Register (Default Value: 0x0BFF_0400)

Offset: 0x0044			Register Name: GP_CH1_CMP_DATA
Bit	Read/Write	Default/Hex	Description

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

15.5.7.16 0x0048 GPADC CH2 Compare Data Register (Default Value: 0x0BFF_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

15.5.7.17 0x004C GPADC CH3 Compare Data Register (Default Value: 0x0BFF_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

15.5.7.18 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

15.5.7.19 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

15.5.7.20 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

15.5.7.21 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

15.6 Touch Panel ADC (TPADC)

15.6.1 Overview

The Touch Panel ADC (TPADC) is a 4-wire resistive touch screen controller, including a 12-bit SAR type A/D converter.

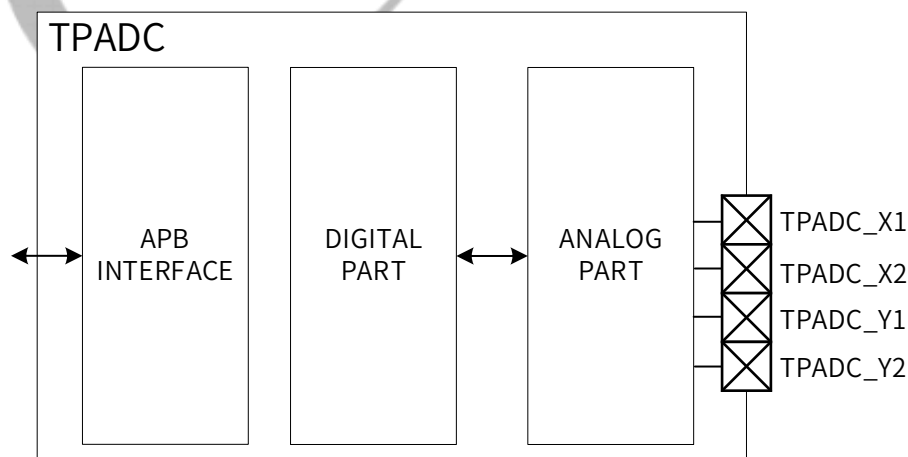
The TPADC has the following features:

- 12-bit SAR type A/D converter
- Configurable sample frequency up to 750 kHz
- One 32x12 FIFO for storing A/D conversion result
- Supports DMA slave interface
- Supports 4-wire resistive touch panel input detection
 - Supports pen down detection with programmable sensitivity
 - Supports single touch coordinate measurement
 - Supports dual touch detection
 - Supports touch pressure measurement with programmable threshold
 - Supports median and averaging filter for noise reduction
 - Supports X and Y coordinate exchange function
- Supports Aux ADC with up to 4 channels

15.6.2 Block Diagram

The following figure shows the block diagram of the TPADC.

Figure 15-30 TPADC Block Diagram



15.6.3 External Signals

The following table describes the external signals of the TPADC. The TPADC pin is the analog input signal.

Table 15-10 TPADC External Signals

Signal Name	Description	Type
TP-X1	Touch Panel X1 Input	AI
TP-X2	Touch Panel X2 Input	AI
TP-Y1	Touch Panel Y1 Input	AI
TP-Y2	Touch Panel Y2 Input	AI

15.6.4 Functional Description

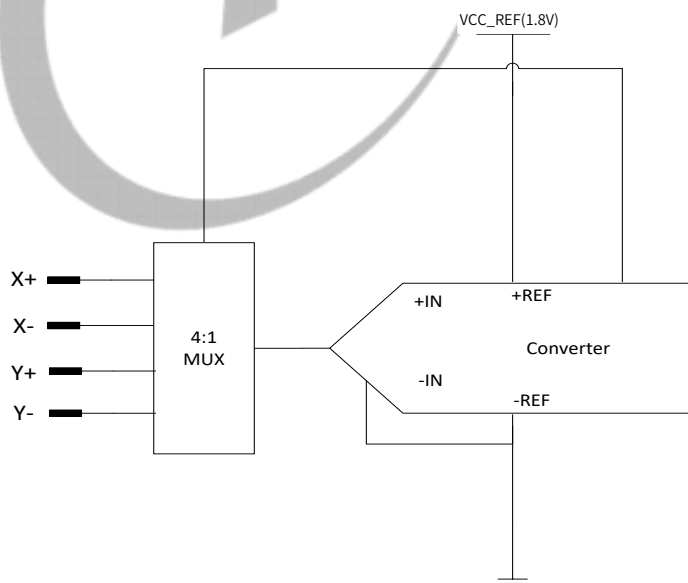
15.6.4.1 Single-ended Mode and Differential Mode

The controller is a typical type of successive approximation ADC (SAR ADC) which contains a sample/hold, analog-to-digital conversion, serial data output functions.

The analog inputs (X+, X-, Y+, Y-) enter the ADC through the control register, the ADC can work in single-ended or differential mode. Selecting Aux ADC should work in single-ended mode; for a touch screen application, it works in a differential mode, which can effectively eliminate the impact on conversion accuracy caused by the parasitic resistance of the driver switch and external interference.

The following figure shows TPADC Single-Ended Mode for the measurement of Aux, using the 1.8V reference source as the ADC reference voltage.

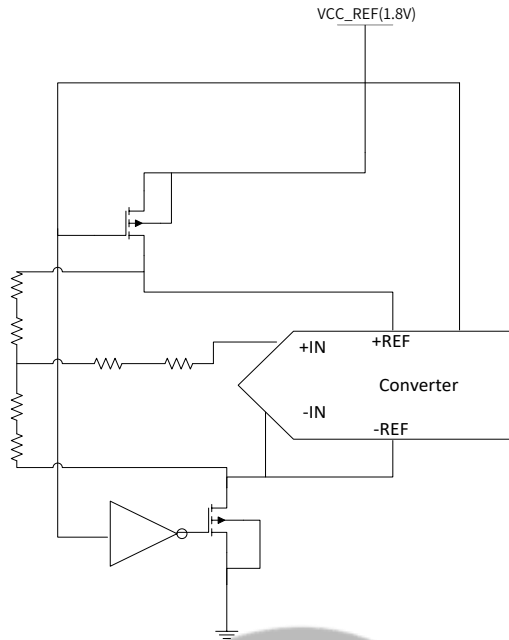
Figure 15-31 TPADC Single-Ended Mode for AUX ADC



The following figure shows TPADC differential mode for the measurement of X/Y/Z coordinate of Touch Panel. The advantage of differential mode: +REF and -REF can directly input to the Y+ and

Y-(or X+ and X-), which can eliminate the measurement error of X+/X-(or Y+/Y-) because of the switch on resistance. The disadvantage is that: both the sample or conversion process, the driver will need to be enabled. Compared with single-ended mode, the power consumption increases.

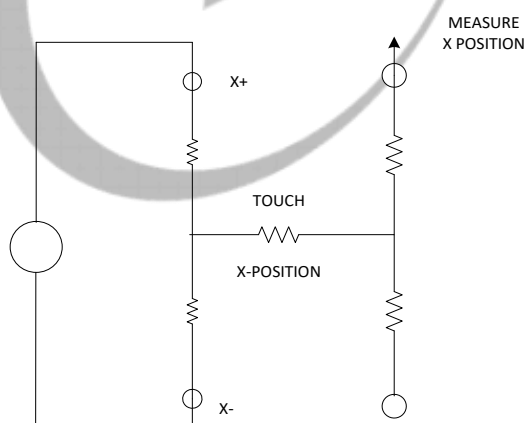
Figure 15-32 TPADC Differential Mode for Touch Panel



15.6.4.2 Single Touch Measurement

The following figure shows the operation principle of the single touch X-Coordinate measurement.

Figure 15-33 Single Touch X-Coordinate Measurement for Touch Panel



For an X coordinate measurement, the X+ pin is internally switched to VCC_REF and X- to GND. The X plate becomes a potential divider, and the voltage at the point of contact is proportional to its X co-ordinate. This voltage is measured on the Y+, which carries no current (hence there is no voltage drop in RY+ or RY-). Due to the ratio metric measurement method, the supply voltage does not affect measurement accuracy. The voltage references VREF+ and VREF- are taken from

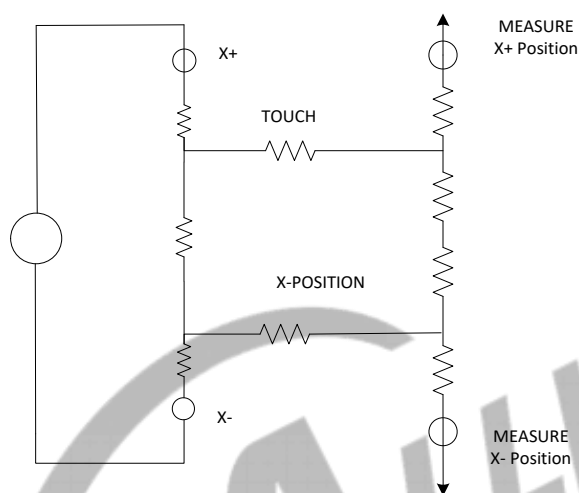
after the matrix switches, so that any voltage drop in these switches has no effect on the ADC measurement.

Y coordinate measurements are similar to X coordinate measurements, with the X and Y plates interchanged.

15.6.4.3 Dual Touch Measurement

In single touch mode, it only needs to test X+ and Y+ signals. But in dual touch mode, it needs to test X+, X-, Y+, and Y- signals. The following figure shows the operation principle of dual touch detection for touch panel.

Figure 15-34 Dual Touch Detection for Touch Panel

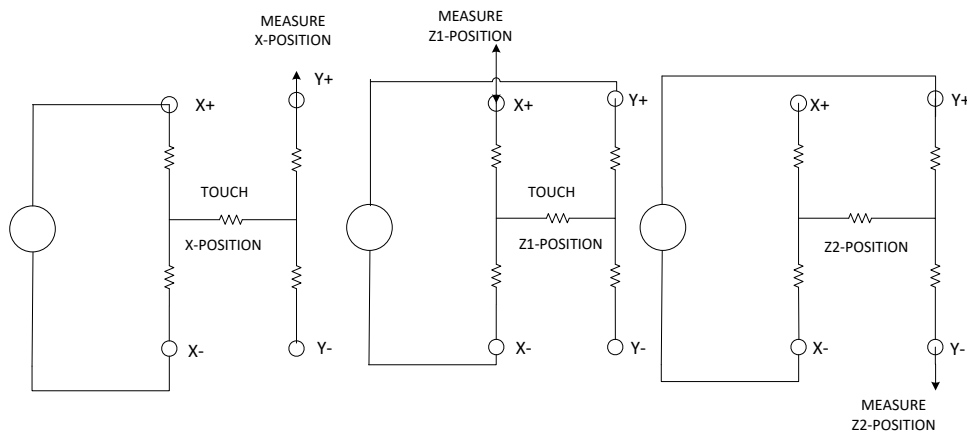


For X coordinates measurement, the X+ pin is internally switched to 1.8 V and X- to GND. The controller needs to test Y+ and Y-, Y coordinates measurement is similar. And record $\Delta X = |X+ - X-|$, $\Delta Y = |Y+ - Y-|$. In practice, we can set a threshold. If ΔX or ΔY greater than the threshold, we consider it as a dual touch, otherwise as a single touch.

15.6.4.4 Touch Pressure Measurement

The pressure applied to the touch screen by a pen or finger to filter unavailable can also be measured by the controller using some simple calculations. The contact resistance between the X and Y plates is measured, which provides a good indication of the size of the depressed area and the applied pressure. The area of the touch spot t is proportional to the size of the object touching it. And the value of this resistance (R_{touch}) can be calculated using two different methods.

Figure 15-35 Touch Pressure Measurement for Touch Panel



• First Method

The first method requires the user to know the total resistance of the X plate tablet (R_{XPLATE}). Three touch screen conversions are required: measurement of the X position, $X_{POSITION}(Y+$ input); measurement of the X+ input with the excitation voltage applied to Y+ and X- (Z1 measurement); and measurement of the Y- input with the excitation voltage applied to Y+ and X- (Z2 measurement). These three measurements are illustrated in following Figure. The controller have two special ADC channel settings to configure the X and Y switches for the Z1 and Z2 measurements and store the results in the Z1 and Z2 result registers. The touch resistance (R_{TOUCH}) can then be calculated using the following equation.

$$R_{TOUCH} = (R_{XPLATE}) \times (X_{POSITION}/4096) \times [(Z2/Z1) - 1]$$

• Second Method

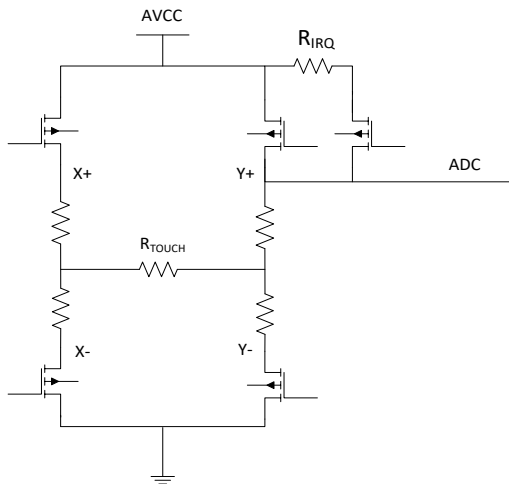
The second method requires the user to know the resistance of the X-plate and Y-plate tablets. Three touch screen conversions are required: a measurement of the X position ($X_{POSITION}$), the Y position ($Y_{POSITION}$), and the Z1 position. The following equation also calculates the touch resistance (R_{TOUCH}).

$$R_{TOUCH} = R_{XPLATE} \times (X_{POSITION}/4096) \times [(4096/Z1) - 1] - R_{YPLATE} \times [1 - (Y_{POSITION}/4096)]$$

15.6.4.5 Pen Down Detection

Pen down detection is used as an interrupt to the host. RIRQ is an internal pull-up resistor with a programmable value from 6 k Ω to 96 k Ω (default 48 k Ω).

Figure 15-36 Pen Down Detection for Touch Panel



The pen down IRQ output is pulled high by an internal pull-up. In the pen down detection, the Y- driver is enabled and connected to GND, and the pen down IRQ output is connected to the X+ input. When the panel is touched, the X+ input is pulled to ground through the touch screen, and the pen down IRQ output goes low because of the current path through the panel to GND, initiating an interrupt to the processor.

During the measurement cycle for X-, Y-, and Z-position, the X+ input is disconnected from the pen down IRQ pull-down transistor to eliminate any pull-up resistor leakage current from flowing through the touch screen, thus causing no errors.

15.6.4.6 Median and Averaging Filter

Touch screens are composed of two resistive layers, normally placed over an LCD screen. Because these layers are in close proximity to the LCD screen, noise can be coupled from the screen onto these resistive layers, causing errors in the touch screen positional measurements.

The controller contain a filtering block to process the data and discard the spurious noise before sending the information to the host. The purpose of this block is not only the suppression of noise; the on-chip filtering also greatly reduces the host processing loading.

The processing function consists of two filters that are applied to the converted results: the median filter and the averaging filter. The median filter suppresses the isolated out-of-range noise and sets the number of measurements to be taken. These measurements are arranged in a temporary array, where the first value is the smallest measurement and the last value is the largest measurement. Then the averaging filter size determines the number of values to average. There are four choices which is configured by [TP_CTRL_REG3](#) register (bit 1 and bit 0) to filtrate the ADC sampling data.

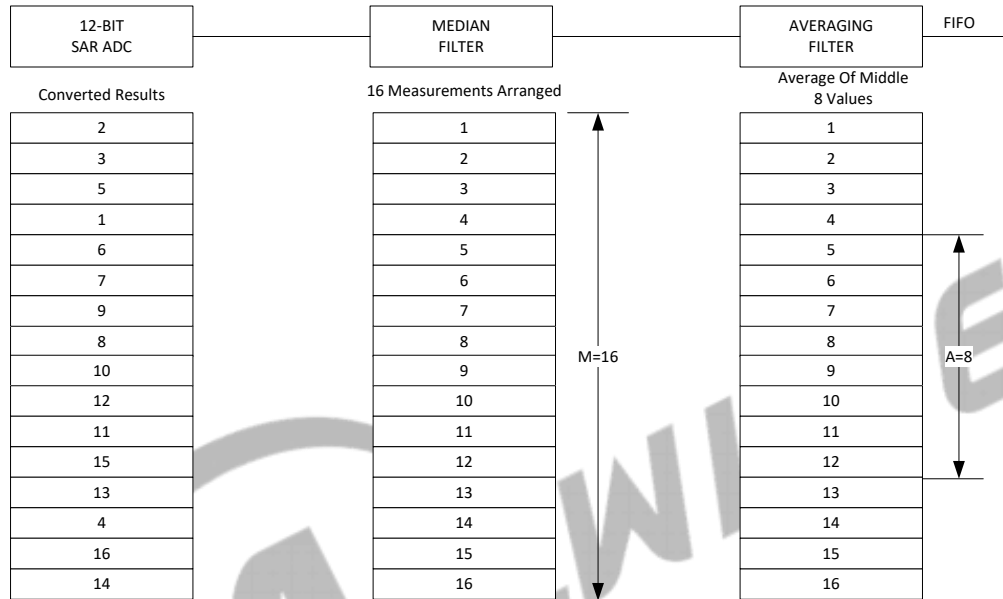
Table 15-11 Median and Averaging Filter Size

Bit1	Bit0	Averaging Filter Size	Median Filter Size
0	0	2	4
0	1	3	5
1	0	4	8

Bit1	Bit0	Averaging Filter Size	Median Filter Size
1	1	8	16

Example: In this example, the bit[1:0] of [TP_CTRL_REG3](#) is configured as 2'b11. So the median filter has a window size of 16. This means that 16 measurements are taken and arranged in descending order in a temporary array. The averaging window size in this example is 8. The output is the average of the middle eight values of the 16 measurements taken with the median filter.

Table 15-12 Median and Averaging Filter Example



15.6.5 Register List

Module Name	Base Address
TPADC	0x02009C00

Register Name	Offset	Description
TP_CTRL_REG0	0x0000	TP Control Register 0
TP_CTRL_REG1	0x0004	TP Control Register 1
TP_CTRL_REG2	0x0008	TP Control Register 2
TP_CTRL_REG3	0x000C	TP Control Register 3
TP_INT_FIFO_CTRL_REG	0x0010	TP Interrupt FIFO Control Register
TP_INT_FIFO_STAT_REG	0x0014	TP Interrupt FIFO Status Register
TP_CALI_DATA_REG	0x001C	TP Calibration Data Register
TP_DATA_REG	0x0024	TP Data Register

15.6.6 Register Description

15.6.6.1 0x0000 TP Control Register 0 (Default Value:0x0F80_0000)

Offset: 0x0000			Register Name: TP_CTRL_REG0
Bit	Read/Write	Default/Hex	Description
31:24	R/W	0xF	ADC_FIRST_DLY ADC First Convert Delay Time (T_FCDT) Setting Based on ADC first convert delay mode select (Bit 23) $T\_FCDT = ADC\_FIRST\_DLY * ADC\_FIRST\_DLY\_MODE$
23	R/W	0x1	ADC_FIRST_DLY_MODE ADC First Convert Delay Mode Select 0: CLK_IN/16 1: CLK_IN/16*256
22	/	/	/
21:20	R/W	0x0	ADC_CLK_DIVIDER ADC Clock Divider (CLK_IN) 00: CLK/2 01: CLK/3 10: CLK/6 11: CLK/1
19:16	R/W	0x0	FS_DIV ADC Sample Frequency Divider 0000: CLK_IN/2 ⁽²⁰⁻⁰⁾ 0001: CLK_IN/2 ⁽²⁰⁻¹⁾ 0010: CLK_IN/2 ⁽²⁰⁻²⁾ 1111: CLK_IN/2 ⁽²⁰⁻¹⁵⁾
15:0	R/W	0x0	TACQ Touch panel ADC acquire time CLK_IN/(16*(N+1))

15.6.6.2 0x0004 TP Control Register 1 (Default Value:0x0000_0101)

Offset: 0x0004			Register Name: TP_CTRL_REG1
Bit	Read/Write	Default/Hex	Description
31:20	/	/	/
19:12	R/W	0x0	STYLUS_UP_DEBOUNCE Stylus Up De-bounce Time Setting 0x00: 0 0xFF: 2N*(CLK_IN/16*256)

Offset: 0x0004			Register Name: TP_CTRL_REG1
Bit	Read/Write	Default/Hex	Description
11:10	/	/	/
9	R/W	0x0	STYLUS_UP_DEBOUCE_EN Stylus Up Debounce Function Select 0: Disable 1: Enable
8	R/W	0x1	CHOPPER_EN T-sensor Chopping Enable 0: Disable 1: Enable This field is not used when there is no T-sensor in TPADC
7	R/WAC	0x0	TOUCH_PAN_CALI_EN Touch Panel Calibration 1: Start calibration, it is cleared to 0 after calibration
6	R/W	0x0	TP_DUAL_EN Touch Panel Double Point Enable 0: Disable 1: Enable
5	R/W	0x0	TP_EN. TP Function Enable 0: Disable 1: Enable
4	R/W	0x0	TP_MODE_SELECT. Touch Panel Mode and Auxiliary ADC Mode Select 0: TP 1: Auxiliary ADC
3	R/W	0x0	ADC_CHAN3_SELECT Analog Input Channel 3 Select 0: Disable 1: Enable
2	R/W	0x0	ADC_CHAN2_SELECT Analog Input Channel 2 Select 0: Disable 1: Enable
1	R/W	0x0	ADC_CHAN1_SELECT Analog Input Channel 1 Select 0: Disable 1: Enable
0	R/W	0x1	ADC_CHAN0_SELECT Analog Input Channel 0 Select 0: Disable 1: Enable

CHAN0–3 can be selected at the same time. If N channel is selected, each channel has 1/N full speed of the ADC. If only one channel is selected, it has the full conversion rate. CHAN0–3 correspond to TP_YN, TP_YP, TP_XN, TP_XP.

15.6.6.3 0x0008 TP Control Register 2 (Default Value:0x8000_0FFF)

Offset: 0x0008			Register Name: TP_CTRL_REG2
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0x8	TP_SENSITIVE_ADJUST Internal Pull-up Resistor Control 0000: Least sensitive 1111: Most sensitive This field is used to adjust sensitivity of pen down detection.
27:26	R/W	0x0	TP_FIFO_MODE_SELECT TP FIFO Access Data Mode Select 00: FIFO store X1,Y1 data for single touch no pressure mode 01: FIFO store X1,Y1, ΔX , ΔY data for dual touch no pressure mode 10: FIFO store X1,Y1, X2,Y2 data for dual touch no pressure mode 11: FIFO store X1,Y1, X2,Y2,Z1,Z2 data for dual touch and pressure mode The ADC output data in single touch mode can store in FIFO with TP_FIFO_MODE_SELECT configured as 01,10,11. The data ΔX and ΔY are theoretically equal to 0, and X2,Y2 are equal to X1,Y1. When PRE_MEA_EN is set and TP_FIFO_MODE_SELECT is not configured as 2'b11, X and Y data will not be stored unless $x1*(z2-z1)/z1 < PRE\_MEA\_THRE\_CNT$. Z data will always be zero when TP_FIFO_MODE_SELECT is configured as 2'b11 but PRE_MEA_EN is not set.
25	/	/	/
24	R/W	0x0	PRE_MEA_EN TP Pressure Measurement Enable Control 0: Disable 1: Enable

Offset: 0x0008			Register Name: TP_CTRL_REG2
Bit	Read/Write	Default/Hex	Description
23:0	R/W	0xFF	PRE_MEA_THRE_CNT TP Pressure Measurement Threshold Control 0x000000: Least sensitive 0xFFFF: Most sensitive This field is used to adjust sensitivity of touch.

15.6.6.4 0x000C TP Control Register 3 (Default Value:0x0000_0001)

Offset: 0x000C			Register Name: TP_CTRL_REG3
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/
2	R/W	0x0	FILTER_EN Filter Enable 0: Disable 1: Enable
1:0	R/W	0x1	FILTER_TYPE Filter Type 00: 4/2 01: 5/3 10: 8/4 11: 16/8

15.6.6.5 0x0010 TP Interrupt& FIFO Control Register (Default Value:0x0000_0F00)

Offset: 0x0010			Register Name: TP_INT_FIFO_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17	R/W	0x0	TP_OVERRUN_IRQ_EN TP FIFO Overrun IRQ Enable 0: Disable 1: Enable
16	R/W	0x0	TP_DATA_IRQ_EN TP FIFO Data Available IRQ Enable 0: Disable 1: Enable
15:14	/	/	/
13	R/W	0x0	TP_DATA_XY_CHANGE TP FIFO X,Y Data Interchange Function Select 0: Disable 1: Enable

Offset: 0x0010			Register Name: TP_INT_FIFO_CTRL_REG
Bit	Read/Write	Default/Hex	Description
12:8	R/W	0xF	TP_FIFO_TRIG_LEVEL TP FIFO Data Available Trigger Level Interrupt and DMA request trigger level for TP or Auxiliary ADC Trigger Level = TXTL + 1
7	R/W	0x0	TP_DATA_DRQ_EN TP FIFO Data Available DRQ Enable 0: Disable 1: Enable
6:5	/	/	/
4	R/WAC	0x0	TP_FIFO_FLUSH TP FIFO Flush Write '1' to flush TX FIFO, self clear to '0'
3:2	/	/	/
1	R/W	0x0	TP_UP_IRQ_EN Touch Panel Last Touch (Stylus Up) IRQ Enable 0: Disable 1: Enable
0	R/W	0x0	TP_DOWN_IRQ_EN Touch Panel First Touch (Stylus Down) IRQ Enable 0: Disable 1: Enable

15.6.6.6 0x0014 TP Interrupt& FIFO Status Register (Default Value:0x0000_0000)

Offset: 0x0014			Register Name: TP_INT_FIFO_STAT_REG
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17	R/W1C	0x0	FIFO_OVERRUN_PENDING TP FIFO Overrun IRQ pending 0: No Pending IRQ 1: FIFO Overrun Pending IRQ Write '1' to clear this interrupt or automatically clear if interrupt condition fails.
16	R/W1C	0x0	FIFO_DATA_PENDING TP FIFO Data Available Pending Bit 0: NO Pending IRQ 1: FIFO Available Pending IRQ Write '1' to clear this interrupt or automatically clear if FIFO flushed.
15:14	/	/	/

Offset: 0x0014			Register Name: TP_INT_FIFO_STAT_REG
Bit	Read/Write	Default/Hex	Description
13:8	R	0x0	RXA_CNT TP FIFO Available Sample Word Counter
7:3	/	/	/
2	R	0x0	TP_IDLE_FLG Touch Panel Idle Flag 0: Idle 1: Not idle
1	R/W1C	0x0	TP_UP_PENDING Touch Panel Last Touch (Stylus Up) IRQ Pending bit 0: No IRQ 1: IRQ Writing 1 to the bit clears it and its corresponding interrupt if the interrupt is enabled.
0	R/W1C	0x0	TP_DOWN_PENDING Touch Panel First Touch (Stylus Down) IRQ Pending bit 0: No IRQ 1: IRQ Writing 1 to the bit clears it and its corresponding interrupt if the interrupt is enabled.

15.6.6.7 0x001C TP Calibration Data Register (Default Value:0x0000_0800)

Offset: 0x001C			Register Name: TP_CALI_DATA_REG
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R/W	0x800	TP_CDAT TP Common Data It is used to adjust the tolerance of the internal ADC.

15.6.6.8 0x0024 TP Data Register (Default Value:0x0000_0000)

In touch panel mode, the data stored in this register bases on TP_FIFO_MODE_SELECT. In Auxiliary ADC mode, the data stored in this register bases on ADC_CHAN_SELECT. If four channels are enabled, FIFO will access the input data in successive turn (ADC_CHAN0 -> ADC_CHAN1 -> ADC_CHAN2 -> ADC_CHAN3). If only two or three channels are selected, such as ADC_CHAN0 and ADC_CHAN3, firstly ADC_CHAN0 input data is accessed, then ADC_CHAN3 input data.

Offset: 0x0024			Register Name: TP_DATA_REG
Bit	Read/Write	Default/Hex	Description

Offset: 0x0024			Register Name: TP_DATA_REG
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R	0x0	TP_DATA Touch Panel X ,Y data or Auxiliary analogy input data converted by the internal ADC.

ALLWINNER

15.7 GPIO

15.7.1 Overview

The General Purpose Input/Output (GPIO) is one of the blocks controlling the chip multiplexing 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.

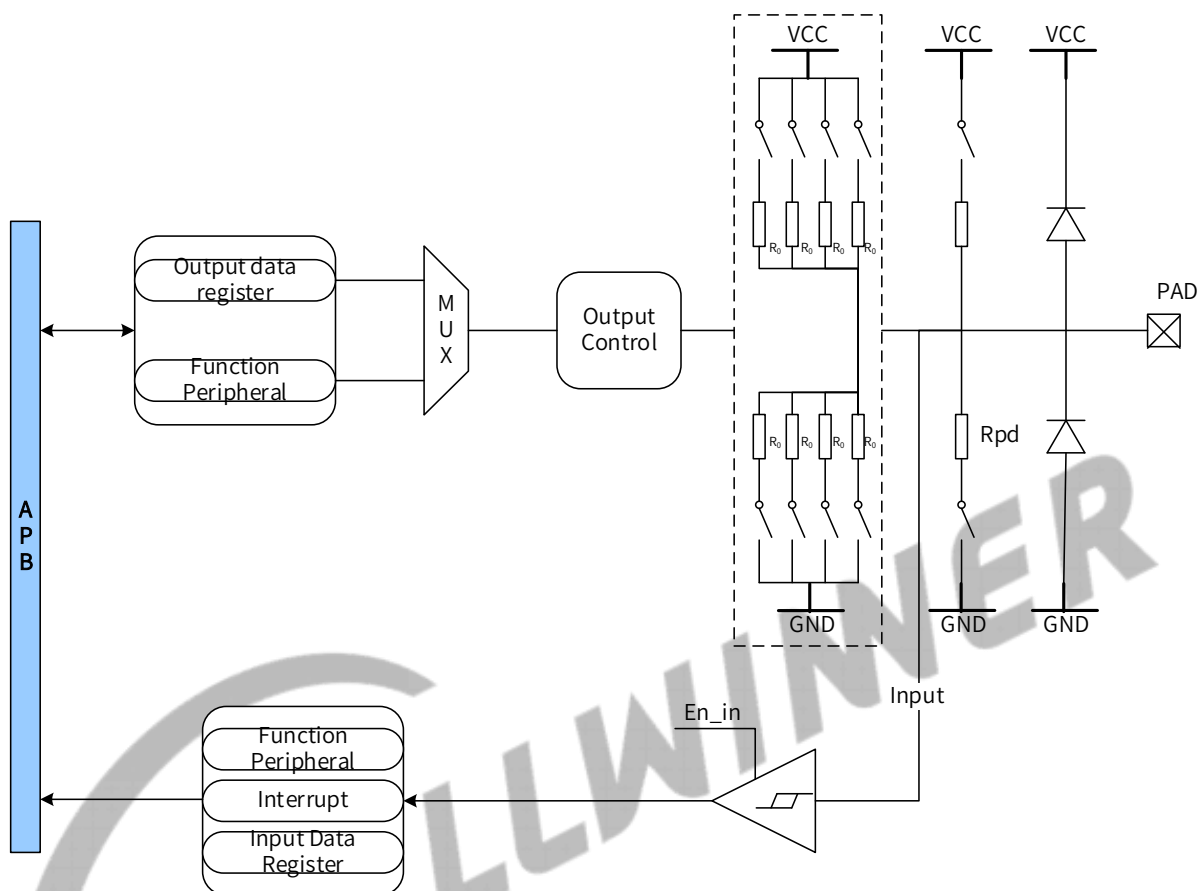
The GPIO has the following features:

- 6 groups of ports (PA, PB, PC, PD, PE, PF)
- Software control for each signal pin
- Each GPIO peripheral can produce an interrupt
- Pull-up/Pull-down/no-Pull register control
- Control the direction of every signal
- 4 drive strengths in each operating mode
- All ports can be configured as interrupts
- Configurable interrupt edges

15.7.2 Block Diagram

The following figure shows the block diagram of the GPIO.

Figure 15-37 GPIO Block Diagram



The overall I/O ports consists of the digital part (GPIO, peripheral interfaces) and IO analog part (output buffer, dual pull down, pad). The output of digital part can be selected by MUX, and the analog part can be used to configure pull-up/pull-down registers, driving strengths and so on.

When in the read operation, read the current pin level to the bus of internal registers. When not in the read operation, the external pins aren't connected to the bus of internal registers, which is called the high-impedance status.

15.7.3 Functional Description

15.7.3.1 Port Function

The Port Controller supports 6 GPIOs, every GPIO can be configured as Input, Output, Function Peripheral, I/O disable or Interrupt function. The configuration instruction of every function is as follows.

Table 15-13 Port Function

Category	Function	Buffer Strength	Pull Up	Pull Down

Category	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



NOTE

/: Configuration is unnecessary and invalid.

Y: Configuration is necessary. If no configuration, it will be configured to the default status.

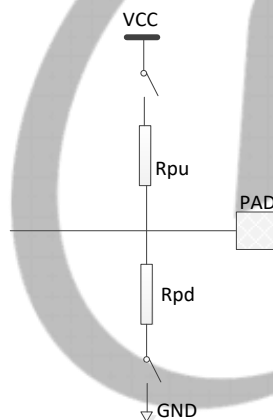
X: Configuration is selected according to the actual situation

N: Configuration is forbidden.

15.7.3.2 Pull-up, Pull-down, and High-impedance Input

Each IO can be configured as the internal pull-up, pull-down, or high-impedance input mode.

Figure 15-38 Pull-up, Pull-down, and High-impedance Input Mode



High-impedance: The output port is in the floating status. All the drives are turned off, and their levels are depended by the external levels.

Pull-up: The uncertain signals are pulled high by resistors, which performs the current-limiting function. In pull-up status, the IO is pulled up to VCC by Rpu.

Pull-down: The uncertain signals are pulled low by resistors. In pull-down status, the IO is pulled down to GND by Rpd.

NOTE

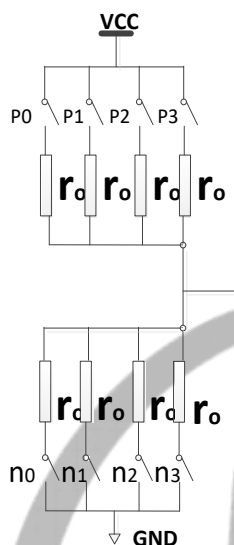
The pull-up/down resistor value of each IO is around 100K.

Whether pull-down, pull-up or high-impedance input is decided by the external circuit.

15.7.3.3 Output Driving Strength

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

Figure 15-39 IO Buffer Strength Diagram



High Level Output

In the process of high level output, the n0, n1, n2, n3 of NMOS is off, and the p0, p1, p2, p3 of PMOS is on.

When the driving strength is set to 0, only the p0 is turned on, during which the output impedance is maximum, that is, the driving capability is the weakest. The output impedance is the turned-on r_o .

When the driving strength is set to 1, only the p0 and p1 are turned on. The output impedance is $r_o/2$, equivalent to the parallel-connected two r_o .

When the driving strength is set to 2, three MOS of p0-p3 are turned on. The output impedance is $r_o/3$, equivalent to the parallel-connected three r_o .

When the driving strength is set to 3, p0-p3 are turned on. The output impedance is $r_o/4$, equivalent to the parallel-connected four r_o .

Low Level Output

In the process of low level output, the p0, p1, p2, p3 of PMOS is off, and the n0, n1, n2, n3 of NMOS is on.

When the driving strength is set to 0, only the n0 is turned on, during which the output impedance is maximum, that is, the driving capability is the weakest. The output impedance is the turned-on r0.

When the driving strength is set to 1, only the n0 and n1 are turned on. The output impedance is $r0/2$, equivalent to the parallel-connected two r0.

When the driving strength is set to 2, three MOS of n0-n3 are turned on. The output impedance is $r0/3$, equivalent to the parallel-connected three r0.

When the driving strength is set to 3, n0-n3 are turned on. The output impedance is $r0/4$, equivalent to the parallel-connected four r0.

15.7.3.4 Interrupts

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 IO will generate interrupts.
- **Negative Edge:** When a high level changes to a low level, the IO will generate interrupts.
- **High Level:** The IO will generate interrupts in the high level.
- **Low Level:** The IO will generate interrupts in the low level.
- **Double Edge:** The IO will generate interrupts both in the positive edge and negative edge.

The interrupt trigger types can be configured in the external interrupt configure registers.

The GPIO interrupts support hardware debounce function configuration in the external interrupt debounce Registers. In this way, the trigger signal will be sampled with a lower-frequency sample clock. As the dither frequency of signals is higher than their sample frequency, the debounce effect is reached.

The sample clock source can be set in the GPIO_INT_CLK_SELECT, and the prescale factor can be set in DEB_CLK_PRE_SCALE.

15.7.3.5 Multi-function Port

This device includes 54 multi-functional input/output port pins. There are 6 ports as listed below.

Table 15-14 Multi-function Port

Port Name	Number of Pins	Input Driver	Output Driver	Multiplex Pins	Typical Power Supply
PA	4	Schmitt	CMOS	TP/ PWM/	1.8V
PB	4	Schmitt	CMOS	BOOST/ TWI/ UART/ PWM/ SPIF/ IR/ SPI/	3.3V

Port Name	Number of Pins	Input Driver	Output Driver	Multiplex Pins	Typical Power Supply
PC	6	Schmitt	CMOS	SDC/ SPI/ SPIF/ BOOT/ UART/ TWI/ PWM/	3.3V
PD	23	Schmitt	CMOS	LCD/ SPI/ DBI/ TWI/ UART/ RJTAG/ PWM/ IR/ I2S/ OWA/ LVDS/ DSI/	3.3V/1.8V
PE	10	Schmitt	CMOS	LCD/ TWI/ I2S/ UART/ SPI/ DBI/ SDC/ CLK/ PWM/ IR/ OWA/	3.3V/1.8V
PF	7	Schmitt	CMOS	SDC/ DJTAG/ RJTAG/ PWM/ UART/ TWI/ IR/ SPI/ SPIF/	3.3V/1.8V



The following tables show the multiplex function pins.



NOTE

For each GPIO, Function0 is input function; Function1 is output function; Function11 to Function13 are reserved.

Table 15-15 Port A Multiplex Function

Pin Name	IO Type	Function2	Function3	Function4	Function5	Function6	Function7	Function8	Function9	Function10	Function14
PA0	A I/O	TP-X1	PWM0-0								PA-EINT0
PA1	A I/O	TP-X2	PWM0-1								PA-EINT1
PA2	A I/O	TP-Y1	PWM0-2								PA-EINT2
PA3	A I/O	TP-Y2	PWM0-3								PA-EINT3

Table 15-16 Port B Multiplex Function

Pin Name	IO Type	Function2	Function3	Function4	Function5	Function6	Function7	Function8	Function9	Function10	Function14
PB0	I/O	BOOST1-PWM	TWI1-SCK	UART1-TX	UART2-RTS		PWM0-0	SPIF0-CS1			PB-EINT0
PB1	I/O	BOOST1-FB	TWI1-SDA	UART1-RX	UART2-CTS		PWM0-1	IR-RX			PB-EINT1
PB2	I/O	BOOST0-PWM	TWI2-SCK	UART1-RTS	UART2-TX		PWM0-2	SPI0-CS1			PB-EINT2
PB3	I/O	BOOST0-FB	TWI2-SDA	UART1-CTS	UART2-RX		PWM0-3	IR-RX			PB-EINT3

Table 15-17 Port C Multiplex Function

Pin Name	IO Type	Function2	Function3	Function4	Function5	Function6	Function7	Function8	Function9	Function10	Function14
PC0	I/O	SDC2-CLK	SPI0-CLK	SPIF0-CLK							PC-EINT0
PC1	I/O	SDC2-CMD	SPI0-CS0	SPIF0-CS0							PC-EINT1
PC2	I/O	SDC2-D2	SPI0-MISO	SPIF0-MISO	BOOT-SEL0	UART3-RTS					PC-EINT2
PC3	I/O	SDC2-D1	SPI0-WP	SPIF0-WP	TWI1-SCK	UART3-TX	PWM0-0				PC-EINT3
PC4	I/O	SDC2-D0	SPI0-MOSI	SPIF0-MOSI	BOOT-SEL1	UART3-CTS					PC-EINT4
PC5	I/O	SDC2-D3	SPI0-HOLD	SPIF0-HOLD	TWI1-SDA	UART3-RX	PWM0-1				PC-EINT5

Table 15-18 Port D Multiplex Function

Pin Name	IO Type	Function2	Function3	Function4	Function5	Function6	Function7	Function8	Function9	Function10	Function14
PD0	I/O	LCD0-D2	SPI1-CS0/DBI-CSX	TWI1-SCK	UART3-TX	TWI0-SCK	RJTAG-MS				PD-EINT0
PD1	I/O	LCD0-D3	SPI1-CLK/DBI-SCLK	TWI1-SDA	UART3-RX	PWM0-0	RJTAG-DI				PD-EINT1
PD2	I/O	LCD0-D4	SPI1-MOSI/DBI-SDO	UART2-TX	TWI0-SDA	UART1-TX	RJTAG-DO				PD-EINT2
PD3	I/O	LCD0-D5	SPI1-MISO/DBI-SDI/DBI-TE/DBI-DCX	UART2-RX	UART3-RTS	UART1-RX	RJTAG-CK				PD-EINT3
PD4	I/O	LCD0-D6	SPI1-HOLD/DBI-DCX/DBI-WRX	UART2-RTS	UART3-CTS	TWI0-SCK					PD-EINT4
PD5	I/O	LCD0-D7	SPI1-WP/DBI-TE	UART2-CTS	IR-RX	TWI0-SDA					PD-EINT5
PD6	I/O	LCD0-D10	SPI1-CS1	I2S0-MCLK	UART4-TX	PWM0-0					PD-EINT6
PD7	I/O	LCD0-D11	IR-RX	I2S0-BCLK	UART4-RX	PWM0-1					PD-EINT7
PD8	I/O	LCD0-D12	TWI1-SCK	I2S0-LRCK	UART4-RTS	PWM0-2					PD-EINT8
PD9	I/O	LCD0-D13	TWI1-SDA	I2S0-DOUT0	UART4-CTS	PWM0-3	LCD0-VSYNC				PD-EINT9
PD10	I/O	LCD0-D14	TWI2-SCK	I2S0-DIN0	UART1-TX	UART5-TX	PWM0-3				PD-EINT10
PD11	I/O	LCD0-D15	TWI2-SDA	OWA-OUT	UART1-RX	UART5-RX					PD-EINT11
PD12	I/O	LCD0-D18	LVDS0-D0P	DSI-D0P	UART3-RX	TWI0-SDA					PD-EINT12
PD13	I/O	LCD0-D19	LVDS0-D0N	DSI-D0N	UART2-TX						PD-EINT13
PD14	I/O	LCD0-D20	LVDS0-D1P	DSI-D1P	UART2-RX						PD-EINT14
PD15	I/O	LCD0-D21	LVDS0-D1N	DSI-D1N	UART2-RTS						PD-EINT15
PD16	I/O	LCD0-D22	LVDS0-D2P	DSI-CKP	UART2-CTS						PD-EINT16

Pin Name	IO Type	Function2	Function3	Function4	Function5	Function6	Function7	Function8	Function9	Function10	Function14
PD17	I/O	LCD0-D23	LVDS0-D2N	DSI-CKN	UART5-TX						PD-EINT17
PD18	I/O	LCD0-CLK	LVDS0-CKP	DSI-D2P	UART5-RX						PD-EINT18
PD19	I/O	LCD0-HSYNC	LVDS0-CKN	DSI-D2N	UART5-RTS	UART4-TX					PD-EINT19
PD20	I/O	LCD0-VSYNC	LVDS0-D3P	DSI-D3P	UART5-CTS	UART4-RX					PD-EINT20
PD21	I/O	LCD0-DE	LVDS0-D3N	DSI-D3N	UART1-TX	UART4-RTS					PD-EINT21
PD22	I/O	OWA-OUT	IR-RX	PWM0-2	UART1-RX	UART4-CTS					PD-EINT22

Table 15-19 Port E Multiplex Function

Pin Name	IO Type	Function2	Function3	Function4	Function5	Function6	Function7	Function8	Function9	Function10	Function14
PE0	I/O		LCD0-D0	TWI1-SCK	I2S0-MCLK	UART0-TX		SPI1-CS0/DBI-CSX	SDC2-D1	CLK-FANOUT0	PE-EINT0
PE1	I/O		LCD0-D1	TWI1-SDA	I2S0-BCLK	UART0-RX		SPI1-CLK/DBI-SCLK	SDC2-D0		PE-EINT1
PE2	I/O		LCD0-D8	TWI0-SCK	I2S0-LRCK	UART4-TX		SPI1-MOSI/DBI-SDO	SDC2-CLK	PWM0-0	PE-EINT2
PE3	I/O		LCD0-D9	TWI0-SDA	I2S0-DOUT0	UART4-RX		SPI1-MISO/DBI-SDI/DBI-TE/DBI-DCX	SDC2-CMD	PWM0-1	PE-EINT3
PE4	I/O		LCD0-D16	TWI2-SCK	I2S0-DIN0	UART4-RTS	UART5-TX	SPI1-HOLD/DBI-DCX/DBI-WRX	SDC2-D3	PWM0-2	PE-EINT4
PE5	I/O		LCD0-D17	TWI2-SDA	IR-RX	UART4-CTS	UART5-RX	SPI1-WP/DBI-TE	SDC2-D2	PWM0-3	PE-EINT5
PE6	I/O		UART2-TX	LCD0-D14		OWA-OUT	IR-RX	SPI1-CS1			PE-EINT6
PE7	I/O		UART2-RX	LCD0-D15					CLK-FANOUT1		PE-EINT7
PE8	I/O		UART2-RTS	TWI0-SCK		UART3-TX	PWM0-0		CLK-FANOUT2		PE-EINT8
PE9	I/O		UART2-CTS	TWI0-SDA		UART3-RX	PWM0-1				PE-EINT9

Table 15-20 Port F Multiplex Function

Pin Name	IO Type	Function2	Function3	Function4	Function5	Function6	Function7	Function8	Function9	Function10	Function14
PF0	I/O	SDC0-D1	DJTAG-MS	RJTAG-MS	PWM0-0	UART1-CTS					PF-EINT0
PF1	I/O	SDC0-D0	DJTAG-DI	RJTAG-DI	PWM0-1						PF-EINT1
PF2	I/O	SDC0-CLK	UART0-TX	TWI0-SCK							PF-EINT2
PF3	I/O	SDC0-CMD	DJTAG-DO	RJTAG-DO	TWI1-SCK						PF-EINT3
PF4	I/O	SDC0-D3	UART0-RX	TWI0-SDA		UART1-TX	TWI2-SCK				PF-EINT4
PF5	I/O	SDC0-D2	DJTAG-CK	RJTAG-CK	TWI1-SDA	UART1-RX	TWI2-SDA				PF-EINT5
PF6	I/O	PWM0-3	IR-RX	SPI0-CS1	SPIF0-CS1	UART1-RTS					PF-EINT6

15.7.4 Register List

Module Name	Base Address
GPIO	0x02000000

Register Name	Offset	Description
PA_CFG0	0x0000	PA Configure Register 0
PA_DAT	0x0010	PA Data Register
PA_PUL0	0x0024	PA Pull Register 0
PB_CFG0	0x0030	PB Configure Register 0
PB_DAT	0x0040	PB Data Register
PB_DRV0	0x0044	PB Multi_Driving Register 0
PB_PUL0	0x0054	PB Pull Register 0
PC_CFG0	0x0060	PC Configure Register 0
PC_DAT	0x0070	PC Data Register
PC_DRV0	0x0074	PC Multi_Driving Register 0
PC_PUL0	0x0084	PC Pull Register 0
PD_CFG0	0x0090	PD Configure Register 0
PD_CFG1	0x0094	PD Configure Register 1
PD_CFG2	0x0098	PD Configure Register 2
PD_DAT	0x00A0	PD Data Register
PD_DRV0	0x00A4	PD Multi_Driving Register 0
PD_DRV1	0x00A8	PD Multi_Driving Register 1
PD_DRV2	0x00AC	PD Multi_Driving Register 2
PD_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_DAT	0x00D0	PE Data Register
PE_DRV0	0x00D4	PE Multi_Driving Register 0
PE_DRV1	0x00D8	PE Multi_Driving Register 1
PE_PUL0	0x00E4	PE Pull Register 0
PF_CFG0	0x00F0	PF Configure Register 0
PF_DAT	0x0100	PF Data Register
PF_DRV0	0x0104	PF Multi_Driving Register 0
PF_PUL0	0x0114	PF Pull Register 0
PA_INT_CFG0	0x0200	PA External Interrupt Configure Register 0
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
PB_INT_CFG0	0x0220	PB External Interrupt Configure Register 0
PB_INT_CTL	0x0230	PB External Interrupt Control Register

Register Name	Offset	Description
PB_INT_STA	0x0234	PB External Interrupt Status Register
PB_INT_DEB	0x0238	PB External Debounce Configure Register
PC_INT_CFG0	0x0240	PC External Interrupt Configure Register 0
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
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_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_CFG0	0x02A0	PF External Interrupt Configure Register 0
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
GPIO_POW_VAL	0x0348	GPIO Group Power Value Register
GPIO_POW_VOL_SET_CTL	0x0350	GPIO Group Power Voltage Select Control Register

15.7.5 Register Description

15.7.5.1 0x0000 PA Configure Register 0 (Default Value: 0x0000_FFFF)

Offset: 0x0000			Register Name: PA_CFG0
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/

Offset: 0x0000			Register Name: PA_CFG0
Bit	Read/Write	Default/Hex	Description
15:12	R/W	0xF	PA3_SELECT PA3 Select 0000: Input 0001: Output 0010: TP-Y2 0011: PWM-3 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA-EINT3 1111: Disable 0010-1101: Refer to Table 15-15 Port A Multiplex Function
11:8	R/W	0xF	PA2_SELECT PA2 Select 0000: Input 0001: Output 0010: TP-Y1 0011: PWM-2 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA-EINT2 1111: Disable
7:4	R/W	0xF	PA1_SELECT PA1 Select 0000: Input 0001: Output 0010: TP-X2 0011: PWM-1 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA-EINT1 1111: 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: TP-X1 0011: PWM-0 0100: Reserved 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PA-EINT0 1111: Disable

15.7.5.2 0x0010 PA Data Register (Default Value: 0x0000_0000)

Offset: 0x0010			Register Name: PA_DAT
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
3	R/W	0x0	PORT3_DAT PA3 Data If the function of PA3 port is configured as input, the PORT3_DAT bit indicates the state of the PA3 pin; If the function of PA3 port is configured as output, the state of the PA3 pin can be configured via software by modifying the value of the PORT3_DAT bit. In this case, the value obtained by reading PORT3_DAT bit is the value set by the software; If the PA3 port is configured as functional pin, this bit is invalid.
2	R/W	0x0	PORT2_DAT PA2 Data If the function of PA2 port is configured as input, the PORT2_DAT bit indicates the state of the PA2 pin; If the function of PA2 port is configured as output, the state of the PA2 pin can be configured via software by modifying the value of the PORT2_DAT bit. In this case, the value obtained by reading PORT2_DAT bit is the value set by the software; If the PA2 port is configured as functional pin, this bit is invalid.

Offset: 0x0010			Register Name: PA_DAT
Bit	Read/Write	Default/Hex	Description
1	R/W	0x0	PORT1_DAT PA1 Data If the function of PA1 port is configured as input, the PORT1_DAT bit indicates the sate of the PA1 pin; If the function of PA1 port is configured as output, the sate of the PA1 pin can be configured via software by modifying the value of the PORT1_DAT bit. In this case, the value obtained by reading PORT1_DAT bit is the value set by the software; If the PA1 port is configured as functional pin, this bit is invalid.
0	R/W	0x0	PORT0_DAT PA0 Data If the function of PA0 port is configured as input, the PORT0_DAT bit indicates the sate of the PA0 pin; If the function of PA0 port is configured as output, the sate of the PA0 pin can be configured via software by modifying the value of the PORT0_DAT bit. In this case, the value obtained by reading PORT0_DAT bit is the value set by the software; If the PA0 port is configured as functional pin, this bit is invalid.

15.7.5.3 0x0024 PA Pull Register 0 (Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: PA_PUL0
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
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

Offset: 0x0024			Register Name: PA_PUL0
Bit	Read/Write	Default/Hex	Description
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

15.7.5.4 0x0030 PB Configure Register 0 (Default Value: 0xFFFF_FFFF)

Offset: 0x0030			Register Name: PB_CFG0
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:12	R/W	0xF	PB3_SELECT PB3 Select 0000: Input 0001: Output 0010: BOOST0-FB 0011: TWI2-SDA 0100: UART1-CTS 0101: UART2-RX 0110: Reserved 0111: PWM0-3 1000: IR-RX 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PB-EINT3 1111: Disable
11:8	R/W	0xF	PB2_SELECT PB2 Select 0000: Input 0001: Output 0010: BOOST0-PWM 0011: TWI2-SCK 0100: UART1-RTS 0101: UART2-TX 0110: Reserved 0111: PWM0-2 1000: SPI0-CS1 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PB-EINT2 1111: Disable
7:4	R/W	0xF	PB1_SELECT PB1 Select 0000: Input 0001: Output 0010: BOOST1-FB 0011: TWI1-SDA 0100: UART1-RX 0101: UART2-CTS 0110: Reserved 0111: PWM0-1 1000: IR-RX 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PB-EINT1 1111: Disable

Offset: 0x0030			Register Name: PB_CFG0	
Bit	Read/Write	Default/Hex	Description	
3:0	R/W	0xF	PB0_SELECT	
			PB0 Select	
			0000: Input	0001: Output
			0010: BOOST1-PWM	0011: TWI1-SCK
			0100: UART1-TX	0101: UART2-RTS
			0110: Reserved	0111: PWM0-0
			1000: SPIF0-CS1	1001: Reserved
			1010: Reserved	1011: Reserved
			1100: Reserved	1101: Reserved
			1110: PB-EINT0	1111: Disable

15.7.5.5 0x0040 PB Data Register (Default Value: 0x0000_0000)

Offset: 0x0040			Register Name: PB_DAT
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	PORT7_DAT PB7 Data If the function of PB7 port is configured as input, the PORT7_DAT bit indicates the state of the PB7 pin; If the function of PB7 port is configured as output, the state of the PB7 pin can be configured via software by modifying the value of the PORT7_DAT bit. In this case, the value obtained by reading PORT7_DAT bit is the value set by the software; If the PB7 port is configured as functional pin, this bit is invalid.
6	R/W	0x0	PORT6_DAT PB6 Data If the function of PB6 port is configured as input, the PORT6_DAT bit indicates the state of the PB6 pin; If the function of PB6 port is configured as output, the state of the PB6 pin can be configured via software by modifying the value of the PORT6_DAT bit. In this case, the value obtained by reading PORT6_DAT bit is the value set by the software; If the PB6 port is configured as functional pin, this bit is invalid.

Offset: 0x0040			Register Name: PB_DAT
Bit	Read/Write	Default/Hex	Description
5	R/W	0x0	PORT5_DAT PB5 Data If the function of PB5 port is configured as input, the PORT5_DAT bit indicates the sate of the PB5 pin; If the function of PB5 port is configured as output, the sate of the PB5 pin can be configured via software by modifying the value of the PORT5_DAT bit. In this case, the value obtained by reading PORT5_DAT bit is the value set by the software; If the PB5 port is configured as functional pin, this bit is invalid.
4	R/W	0x0	PORT4_DAT PB4 Data If the function of PB4 port is configured as input, the PORT4_DAT bit indicates the sate of the PB4 pin; If the function of PB4 port is configured as output, the sate of the PB4 pin can be configured via software by modifying the value of the PORT4_DAT bit. In this case, the value obtained by reading PORT4_DAT bit is the value set by the software; If the PB4 port is configured as functional pin, this bit is invalid.
3	R/W	0x0	PORT3_DAT PB3 Data If the function of PB3 port is configured as input, the PORT3_DAT bit indicates the sate of the PB3 pin; If the function of PB3 port is configured as output, the sate of the PB3 pin can be configured via software by modifying the value of the PORT3_DAT bit. In this case, the value obtained by reading PORT3_DAT bit is the value set by the software; If the PB3 port is configured as functional pin, this bit is invalid.

Offset: 0x0040			Register Name: PB_DAT
Bit	Read/Write	Default/Hex	Description
2	R/W	0x0	PORT2_DAT PB2 Data If the function of PB2 port is configured as input, the PORT2_DAT bit indicates the state of the PB2 pin; If the function of PB2 port is configured as output, the state of the PB2 pin can be configured via software by modifying the value of the PORT2_DAT bit. In this case, the value obtained by reading PORT2_DAT bit is the value set by the software; If the PB2 port is configured as functional pin, this bit is invalid.
1	R/W	0x0	PORT1_DAT PB1 Data If the function of PB1 port is configured as input, the PORT1_DAT bit indicates the state of the PB1 pin; If the function of PB1 port is configured as output, the state of the PB1 pin can be configured via software by modifying the value of the PORT1_DAT bit. In this case, the value obtained by reading PORT1_DAT bit is the value set by the software; If the PB1 port is configured as functional pin, this bit is invalid.
0	/	/	/

15.7.5.6 0x0044 PB Multi_Driving Register 0 (Default Value: 0x1111_1111)

Offset: 0x0044			Register Name: PB_DRV0
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13:12	R/W	0x1	PB3_DRV PB3 Multi-driving Select 00: Level0 01: Level1 10: Level2 11: Level3
11:10	/	/	/
9:8	R/W	0x1	PB2_DRV PB2 Multi-driving Select 00: Level0 01: Level1 10: Level2 11: Level3
7:6	/	/	/

Offset: 0x0044			Register Name: PB_DRV0
Bit	Read/Write	Default/Hex	Description
5:4	R/W	0x1	PB1_DRV PB1 Multi-driving Select 00: Level0 01: Level1 10: Level2 11: Level3
3:2	/	/	/
1:0	R/W	0x1	Reserved

15.7.5.7 0x0054 PB Pull Register 0 (Default Value: 0x0000_0000)

Offset: 0x0054			Register Name: PB_PUL0
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:6	R/W	0x0	PB3_PULL PB3 Pull_up or down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
5:4	R/W	0x0	PB2_PULL PB2 Pull_up or down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
3:2	R/W	0x0	PB1_PULL PB1 Pull_up or down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
1:0	/	/	/

15.7.5.8 0x0060 PC Configure Register 0 (Default Value: 0xFFFF_FFFF)

Offset: 0x0060			Register Name: PC_CFG0
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/

Offset: 0x0060			Register Name: PC_CFG0
Bit	Read/Write	Default/Hex	Description
23:20	R/W	0xF	PC5_SELECT PC5 Select 0000: Input 0001: Output 0010: SDC2-D3 0011: SPI0-HOLD 0100: SPIF0-HOLD 0101: TWI1-SDA 0110: UART3-RX 0111: PWM0-1 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PC-EINT5 1111: Disable
19:16	R/W	0xF	PC4_SELECT PC4 Select 0000: Input 0001: Output 0010: SDC2-D0 0011: SPI0-MOSI 0100: SPIF0-MOSI 0101: BOOT-SEL1 0110: UART3-CTS 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PC-EINT4 1111: Disable
15:12	R/W	0xF	PC3_SELECT PC3 Select 0000: Input 0001: Output 0010: SDC2-D1 0011: SPI0-WP 0100: SPIF0-WP 0101: TWI1-SCK 0110: UART3-TX 0111: PWM0-0 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PC-EINT3 1111: Disable
11:8	R/W	0xF	PC2_SELECT PC2 Select 0000: Input 0001: Output 0010: SDC2-D2 0011: SPI0-MISO 0100: SPIF0-MISO 0101: BOOT-SEL0 0110: UART3-RTS 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PC-EINT2 1111: Disable

Offset: 0x0060			Register Name: PC_CFG0
Bit	Read/Write	Default/Hex	Description
7:4	R/W	0xF	PC1_SELECT PC1 Select 0000: Input 0001: Output 0010: SDC2-CMD 0011: SPI0-CS0 0100: SPIF0-CS0 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PC-EINT1 1111: Disable
3:0	R/W	0xF	PC0_SELECT PC0 Select 0000: Input 0001: Output 0010: SDC2-CLK 0011: SPI0-CLK 0100: SPIF0-CLK 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PC-EINT0 1111: Disable

15.7.5.9 0x0070 PC Data Register (Default Value: 0x0000_0000)

Offset: 0x0070			Register Name: PC_DAT
Bit	Read/Write	Default/Hex	Description
31:6	/	/	/
5	R/W	0x0	PORT5_DAT PC5 Data If the function of PC5 port is configured as input, the PORT5_DAT bit indicates the state of the PC5 pin; If the function of PC5 port is configured as output, the state of the PC5 pin can be configured via software by modifying the value of the PORT5_DAT bit. In this case, the value obtained by reading PORT5_DAT bit is the value set by the software; If the PC5 port is configured as functional pin, this bit is invalid.

Offset: 0x0070			Register Name: PC_DAT
Bit	Read/Write	Default/Hex	Description
4	R/W	0x0	PORT4_DAT PC4 Data If the function of PC4 port is configured as input, the PORT4_DAT bit indicates the sate of the PC4 pin; If the function of PC4 port is configured as output, the sate of the PC4 pin can be configured via software by modifying the value of the PORT4_DAT bit. In this case, the value obtained by reading PORT4_DAT bit is the value set by the software; If the PC4 port is configured as functional pin, this bit is invalid.
3	R/W	0x0	PORT3_DAT PC3 Data If the function of PC3 port is configured as input, the PORT3_DAT bit indicates the sate of the PC3 pin; If the function of PC3 port is configured as output, the sate of the PC3 pin can be configured via software by modifying the value of the PORT3_DAT bit. In this case, the value obtained by reading PORT3_DAT bit is the value set by the software; If the PC3 port is configured as functional pin, this bit is invalid.
2	R/W	0x0	PORT2_DAT PC2 Data If the function of PC2 port is configured as input, the PORT2_DAT bit indicates the sate of the PC2 pin; If the function of PC2 port is configured as output, the sate of the PC2 pin can be configured via software by modifying the value of the PORT2_DAT bit. In this case, the value obtained by reading PORT2_DAT bit is the value set by the software; If the PC2 port is configured as functional pin, this bit is invalid.

Offset: 0x0070			Register Name: PC_DAT
Bit	Read/Write	Default/Hex	Description
1	R/W	0x0	PORT1_DAT PC1 Data If the function of PC1 port is configured as input, the PORT1_DAT bit indicates the state of the PC1 pin; If the function of PC1 port is configured as output, the state of the PC1 pin can be configured via software by modifying the value of the PORT1_DAT bit. In this case, the value obtained by reading PORT1_DAT bit is the value set by the software; If the PC1 port is configured as functional pin, this bit is invalid.
0	R/W	0x0	PORT0_DAT PC0 Data If the function of PC0 port is configured as input, the PORT0_DAT bit indicates the state of the PC0 pin; If the function of PC0 port is configured as output, the state of the PC0 pin can be configured via software by modifying the value of the PORT0_DAT bit. In this case, the value obtained by reading PORT0_DAT bit is the value set by the software; If the PC0 port is configured as functional pin, this bit is invalid.

15.7.5.10 0x0074 PC Multi_Driving Register 0 (Default Value: 0x1111_1111)

Offset: 0x0074			Register Name: PC_DRV0
Bit	Read/Write	Default/Hex	Description
31:22	/	/	/
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	/	/	/

Offset: 0x0074			Register Name: PC_DRV0
Bit	Read/Write	Default/Hex	Description
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

15.7.5.11 0x0084 PC Pull Register 0 (Default Value: 0x0000_0054)

Offset: 0x0084			Register Name: PC_PUL0
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
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	0x1	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	0x0	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

Offset: 0x0084			Register Name: PC_PUL0
Bit	Read/Write	Default/Hex	Description
3:2	R/W	0x1	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

15.7.5.12 0x0090 PD Configure Register 0 (Default Value: 0xFFFF_FFFF)

Offset: 0x0090			Register Name: PD_CFG0
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0xF	PD7_SELECT PD7 Select 0000: Input 0001: Output 0010: LCD0-D11 0011: IR-RX 0100: I2S0-BCLK 0101: UART4-RX 0110: PWM0-1 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT7 1111: Disable
27:24	R/W	0xF	PD6_SELECT PD6 Select 0000: Input 0001: Output 0010: LCD0-D10 0011: SPI1-CS1 0100: I2S0-MCLK 0101: UART4-TX 0110: PWM0-0 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT6 1111: Disable

Offset: 0x0090			Register Name: PD_CFG0
Bit	Read/Write	Default/Hex	Description
23:20	R/W	0xF	PD5_SELECT PD5 Select 0000: Input 0001: Output 0010: LCD0-D7 0011: SPI1-WP/DBI-TE 0100: UART2-CTS 0101: IR-RX 0110: TWI0-SDA 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT5 1111: Disable
19:16	R/W	0xF	PD4_SELECT PD4 Select 0000: Input 0001: Output 0010: LCD0-D6 0011: SPI1-HOLD/DBI-DCX/DBI-WRX 0100: UART2-RTS 0101: UART3-CTS 0110: TWI0-SCK 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT4 1111: Disable
15:12	R/W	0xF	PD3_SELECT PD3 Select 0000: Input 0001: Output 0010: LCD0-D5 0011: SPI1-MISO/DBI-SDI/DBI-TE/DBI-DCX 0100: UART2-RX 0101: UART3-RTS 0110: UART1-RX 0111: RJTAG-CK 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT3 1111: Disable
11:8	R/W	0xF	PD2_SELECT PD2 Select 0000: Input 0001: Output 0010: LCD0-D4 0011: SPI1-MOSI/DBI-SDO 0100: UART2-TX 0101: TWI0-SDA 0110: UART1-TX 0111: RJTAG-DO 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT2 1111: Disable

Offset: 0x0090			Register Name: PD_CFG0
Bit	Read/Write	Default/Hex	Description
7:4	R/W	0xF	PD1_SELECT PD1 Select 0000: Input 0001: Output 0010: LCD0-D3 0011: SPI1-CLK/DBI-SCLK 0100: TWI1-SDA 0101: UART3-RX 0110: PWM0-0 0111: RJTAG-DI 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT1 1111: Disable
3:0	R/W	0xF	PD0_SELECT PD0 Select 0000: Input 0001: Output 0010: LCD0-D2 0011: SPI1-CS0/DBI-CSX 0100: TWI1-SCK 0101: UART3-TX 0110: TWI0-SCK 0111: RJTAG-MS 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT0 1111: Disable

15.7.5.13 0x0094 PD Configure Register 1 (Default Value: 0xFFFF_FFFF)

Offset: 0x0094			Register Name: PD_CFG1
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0xF	PD15_SELECT PD15 Select 0000: Input 0001: Output 0010: LCD0-D21 0011: LVDS0-D1N 0100: DSI-D1N 0101: UART2-RTS 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT15 1111: Disable

Offset: 0x0094			Register Name: PD_CFG1
Bit	Read/Write	Default/Hex	Description
27:24	R/W	0xF	PD14_SELECT PD14 Select 0000: Input 0001: Output 0010: LCD0-D20 0011: LVDS0-D1P 0100: DSI-D1P 0101: UART2-RX 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT14 1111: Disable
23:20	R/W	0xF	PD13_SELECT PD13 Select 0000: Input 0001: Output 0010: LCD0-D19 0011: LVDS0-D0N 0100: DSI-D0N 0101: UART2-TX 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT13 1111: Disable
19:16	R/W	0xF	PD12_SELECT PD12 Select 0000: Input 0001: Output 0010: LCD0-D18 0011: LVDS0-D0P 0100: DSI-D0P 0101: UART3-RX 0110: TWI0-SDA 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT12 1111: Disable
15:12	R/W	0xF	PD11_SELECT PD11 Select 0000: Input 0001: Output 0010: LCD0-D15 0011: TWI2-SDA 0100: OWA-OUT 0101: UART1-RX 0110: UART5-RX 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT11 1111: Disable

Offset: 0x0094			Register Name: PD_CFG1
Bit	Read/Write	Default/Hex	Description
11:8	R/W	0xF	PD10_SELECT PD10 Select 0000: Input 0001: Output 0010: LCD0-D14 0011: TWI2-SCK 0100: I2S0-DIN0 0101: UART1-TX 0110: UART5-TX 0111: PWM0-3 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT10 1111: Disable
7:4	R/W	0xF	PD9_SELECT PD9 Select 0000: Input 0001: Output 0010: LCD0-D13 0011: TWI1-SDA 0100: I2S0-DOUT0 0101: UART4-CTS 0110: PWM0-3 0111: LCD0-VSYNC 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT9 1111: Disable
3:0	R/W	0xF	PD8_SELECT PD8 Select 0000: Input 0001: Output 0010: LCD0-D12 0011: TWI1-SCK 0100: I2S0-LRCK 0101: UART4-RTS 0110: PWM0-2 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT8 1111: Disable

15.7.5.14 0x0098 PD Configure Register 2 (Default Value: 0x0FFF_FFFF)

Offset: 0x0098			Register Name: PD_CFG2
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/

Offset: 0x0098			Register Name: PD_CFG2
Bit	Read/Write	Default/Hex	Description
27:24	R/W	0xF	PD22_SELECT PD22 Select 0000: Input 0001: Output 0010: OWA-OUT 0011: IR-RX 0100: PWM0-2 0101: UART1-RX 0110: UART4-CTS 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT22 1111: Disable
23:20	R/W	0xF	PD21_SELECT PD21 Select 0000: Input 0001: Output 0010: LCD0-DE 0011: LVDS0-D3N 0100: DSI-D3N 0101: UART1-TX 0110: UART4-RTS 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT21 1111: Disable
19:16	R/W	0xF	PD20_SELECT PD20 Select 0000: Input 0001: Output 0010: LCD0-VSYNC 0011: LVDS0-D3P 0100: DSI-D3P 0101: UART5-CTS 0110: UART4-RX 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT20 1111: Disable
15:12	R/W	0xF	PD19_SELECT PD19 Select 0000: Input 0001: Output 0010: LCD0-HSYNC 0011: LVDS0-CKN 0100: DSI-D2N 0101: UART5-RTS 0110: UART4-TX 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT19 1111: Disable

Offset: 0x0098			Register Name: PD_CFG2
Bit	Read/Write	Default/Hex	Description
11:8	R/W	0xF	PD18_SELECT PD18 Select 0000: Input 0001: Output 0010: LCD0-CLK 0011: LVDS0-CKP 0100: DSI-D2P 0101: UART5-RX 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT18 1111: Disable
7:4	R/W	0xF	PD17_SELECT PD17 Select 0000: Input 0001: Output 0010: LCD0-D23 0011: LVDS0-D2N 0100: DSI-CKN 0101: UART5-TX 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT17 1111: Disable
3:0	R/W	0xF	PD16_SELECT PD16 Select 0000: Input 0001: Output 0010: LCD0-D22 0011: LVDS0-D2P 0100: DSI-CKP 0101: UART2-CTS 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PD-EINT16 1111: Disable

15.7.5.15 0x00A0 PD Data Register (Default Value: 0x0000_0000)

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

Offset: 0x00A0			Register Name: PD_DAT
Bit	Read/Write	Default/Hex	Description
22	R/W	0x0	PORT22_DAT PD22 Data If the function of PD22 port is configured as input, the PORT22_DAT bit indicates the state of the PD22 pin; If the function of PD22 port is configured as output, the state of the PD22 pin can be configured via software by modifying the value of the PORT22_DAT bit. In this case, the value obtained by reading PORT22_DAT bit is the value set by the software; If the PD22 port is configured as functional pin, this bit is invalid.
21	R/W	0x0	PORT21_DAT PD21 Data If the function of PD21 port is configured as input, the PORT21_DAT bit indicates the state of the PD21 pin; If the function of PD21 port is configured as output, the state of the PD21 pin can be configured via software by modifying the value of the PORT21_DAT bit. In this case, the value obtained by reading PORT21_DAT bit is the value set by the software; If the PD21 port is configured as functional pin, this bit is invalid.
20	R/W	0x0	PORT20_DAT PD20 Data If the function of PD20 port is configured as input, the PORT20_DAT bit indicates the state of the PD20 pin; If the function of PD20 port is configured as output, the state of the PD20 pin can be configured via software by modifying the value of the PORT20_DAT bit. In this case, the value obtained by reading PORT20_DAT bit is the value set by the software; If the PD20 port is configured as functional pin, this bit is invalid.

Offset: 0x00A0			Register Name: PD_DAT
Bit	Read/Write	Default/Hex	Description
19	R/W	0x0	PORT19_DAT PD19 Data If the function of PD19 port is configured as input, the PORT19_DAT bit indicates the state of the PD19 pin; If the function of PD19 port is configured as output, the state of the PD19 pin can be configured via software by modifying the value of the PORT19_DAT bit. In this case, the value obtained by reading PORT19_DAT bit is the value set by the software; If the PD19 port is configured as functional pin, this bit is invalid.
18	R/W	0x0	PORT18_DAT PD18 Data If the function of PD18 port is configured as input, the PORT18_DAT bit indicates the state of the PD18 pin; If the function of PD18 port is configured as output, the state of the PD18 pin can be configured via software by modifying the value of the PORT18_DAT bit. In this case, the value obtained by reading PORT18_DAT bit is the value set by the software; If the PD18 port is configured as functional pin, this bit is invalid.
17	R/W	0x0	PORT17_DAT PD17 Data If the function of PD17 port is configured as input, the PORT17_DAT bit indicates the state of the PD17 pin; If the function of PD17 port is configured as output, the state of the PD17 pin can be configured via software by modifying the value of the PORT17_DAT bit. In this case, the value obtained by reading PORT17_DAT bit is the value set by the software; If the PD17 port is configured as functional pin, this bit is invalid.

Offset: 0x00A0			Register Name: PD_DAT
Bit	Read/Write	Default/Hex	Description
16	R/W	0x0	PORT16_DAT PD16 Data If the function of PD16 port is configured as input, the PORT16_DAT bit indicates the state of the PD16 pin; If the function of PD16 port is configured as output, the state of the PD16 pin can be configured via software by modifying the value of the PORT16_DAT bit. In this case, the value obtained by reading PORT16_DAT bit is the value set by the software; If the PD16 port is configured as functional pin, this bit is invalid.
15	R/W	0x0	PORT15_DAT PD15 Data If the function of PD15 port is configured as input, the PORT15_DAT bit indicates the state of the PD15 pin; If the function of PD15 port is configured as output, the state of the PD15 pin can be configured via software by modifying the value of the PORT15_DAT bit. In this case, the value obtained by reading PORT15_DAT bit is the value set by the software; If the PD15 port is configured as functional pin, this bit is invalid.
14	R/W	0x0	PORT14_DAT PD14 Data If the function of PD14 port is configured as input, the PORT14_DAT bit indicates the state of the PD14 pin; If the function of PD14 port is configured as output, the state of the PD14 pin can be configured via software by modifying the value of the PORT14_DAT bit. In this case, the value obtained by reading PORT14_DAT bit is the value set by the software; If the PD14 port is configured as functional pin, this bit is invalid.

Offset: 0x00A0			Register Name: PD_DAT
Bit	Read/Write	Default/Hex	Description
13	R/W	0x0	PORT13_DAT PD13 Data If the function of PD13 port is configured as input, the PORT13_DAT bit indicates the state of the PD13 pin; If the function of PD13 port is configured as output, the state of the PD13 pin can be configured via software by modifying the value of the PORT13_DAT bit. In this case, the value obtained by reading PORT13_DAT bit is the value set by the software; If the PD13 port is configured as functional pin, this bit is invalid.
12	R/W	0x0	PORT12_DAT PD12 Data If the function of PD12 port is configured as input, the PORT12_DAT bit indicates the state of the PD12 pin; If the function of PD12 port is configured as output, the state of the PD12 pin can be configured via software by modifying the value of the PORT12_DAT bit. In this case, the value obtained by reading PORT12_DAT bit is the value set by the software; If the PD12 port is configured as functional pin, this bit is invalid.
11	R/W	0x0	PORT11_DAT PD11 Data If the function of PD11 port is configured as input, the PORT11_DAT bit indicates the state of the PD11 pin; If the function of PD11 port is configured as output, the state of the PD11 pin can be configured via software by modifying the value of the PORT11_DAT bit. In this case, the value obtained by reading PORT11_DAT bit is the value set by the software; If the PD11 port is configured as functional pin, this bit is invalid.

Offset: 0x00A0			Register Name: PD_DAT
Bit	Read/Write	Default/Hex	Description
10	R/W	0x0	PORT10_DAT PD10 Data If the function of PD10 port is configured as input, the PORT10_DAT bit indicates the sate of the PD10 pin; If the function of PD10 port is configured as output, the sate of the PD10 pin can be configured via software by modifying the value of the PORT10_DAT bit. In this case, the value obtained by reading PORT10_DAT bit is the value set by the software; If the PD10 port is configured as functional pin, this bit is invalid.
9	R/W	0x0	PORT9_DAT PD9 Data If the function of PD9 port is configured as input, the PORT9_DAT bit indicates the sate of the PD9 pin; If the function of PD9 port is configured as output, the sate of the PD9 pin can be configured via software by modifying the value of the PORT9_DAT bit. In this case, the value obtained by reading PORT9_DAT bit is the value set by the software; If the PD9 port is configured as functional pin, this bit is invalid.
8	R/W	0x0	PORT8_DAT PD8 Data If the function of PD8 port is configured as input, the PORT8_DAT bit indicates the sate of the PD8 pin; If the function of PD8 port is configured as output, the sate of the PD8 pin can be configured via software by modifying the value of the PORT8_DAT bit. In this case, the value obtained by reading PORT8_DAT bit is the value set by the software; If the PD8 port is configured as functional pin, this bit is invalid.

Offset: 0x00A0			Register Name: PD_DAT
Bit	Read/Write	Default/Hex	Description
7	R/W	0x0	PORT7_DAT PD7 Data If the function of PD7 port is configured as input, the PORT7_DAT bit indicates the sate of the PD7 pin; If the function of PD7 port is configured as output, the sate of the PD7 pin can be configured via software by modifying the value of the PORT7_DAT bit. In this case, the value obtained by reading PORT7_DAT bit is the value set by the software; If the PD7 port is configured as functional pin, this bit is invalid.
6	R/W	0x0	PORT6_DAT PD6 Data If the function of PD6 port is configured as input, the PORT6_DAT bit indicates the sate of the PD6 pin; If the function of PD6 port is configured as output, the sate of the PD6 pin can be configured via software by modifying the value of the PORT6_DAT bit. In this case, the value obtained by reading PORT6_DAT bit is the value set by the software; If the PD6 port is configured as functional pin, this bit is invalid.
5	R/W	0x0	PORT5_DAT PD5 Data If the function of PD5 port is configured as input, the PORT5_DAT bit indicates the sate of the PD5 pin; If the function of PD5 port is configured as output, the sate of the PD5 pin can be configured via software by modifying the value of the PORT5_DAT bit. In this case, the value obtained by reading PORT5_DAT bit is the value set by the software; If the PD5 port is configured as functional pin, this bit is invalid.

Offset: 0x00A0			Register Name: PD_DAT
Bit	Read/Write	Default/Hex	Description
4	R/W	0x0	PORT4_DAT PD4 Data If the function of PD4 port is configured as input, the PORT4_DAT bit indicates the sate of the PD4 pin; If the function of PD4 port is configured as output, the sate of the PD4 pin can be configured via software by modifying the value of the PORT4_DAT bit. In this case, the value obtained by reading PORT4_DAT bit is the value set by the software; If the PD4 port is configured as functional pin, this bit is invalid.
3	R/W	0x0	PORT3_DAT PD3 Data If the function of PD3 port is configured as input, the PORT3_DAT bit indicates the sate of the PD3 pin; If the function of PD3 port is configured as output, the sate of the PD3 pin can be configured via software by modifying the value of the PORT3_DAT bit. In this case, the value obtained by reading PORT3_DAT bit is the value set by the software; If the PD3 port is configured as functional pin, this bit is invalid.
2	R/W	0x0	PORT2_DAT PD2 Data If the function of PD2 port is configured as input, the PORT2_DAT bit indicates the sate of the PD2 pin; If the function of PD2 port is configured as output, the sate of the PD2 pin can be configured via software by modifying the value of the PORT2_DAT bit. In this case, the value obtained by reading PORT2_DAT bit is the value set by the software; If the PD2 port is configured as functional pin, this bit is invalid.

Offset: 0x00A0			Register Name: PD_DAT
Bit	Read/Write	Default/Hex	Description
1	R/W	0x0	PORT1_DAT PD1 Data If the function of PD1 port is configured as input, the PORT1_DAT bit indicates the state of the PD1 pin; If the function of PD1 port is configured as output, the state of the PD1 pin can be configured via software by modifying the value of the PORT1_DAT bit. In this case, the value obtained by reading PORT1_DAT bit is the value set by the software; If the PD1 port is configured as functional pin, this bit is invalid.
0	R/W	0x0	PORT0_DAT PD0 Data If the function of PD0 port is configured as input, the PORT0_DAT bit indicates the state of the PD0 pin; If the function of PD0 port is configured as output, the state of the PD0 pin can be configured via software by modifying the value of the PORT0_DAT bit. In this case, the value obtained by reading PORT0_DAT bit is the value set by the software; If the PD0 port is configured as functional pin, this bit is invalid.

15.7.5.16 0x00A4 PD Multi_Driving Register 0 (Default Value: 0x1111_1111)

Offset: 0x00A4			Register Name: PD_DRV0
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:28	R/W	0x1	PD7_DRV PD7 Multi-driving Select 00: 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	/	/	/

Offset: 0x00A4			Register Name: PD_DRV0
Bit	Read/Write	Default/Hex	Description
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	PD0_DRV PD0 Multi-driving Select 00: Level0 01: Level1 10: Level2 11: Level3

15.7.5.17 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	/	/	/

Offset: 0x00A8			Register Name: PD_DRV1
Bit	Read/Write	Default/Hex	Description
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

15.7.5.18 0x00AC PD Multi_Driving Register 2 (Default Value: 0x0111_1111)

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

Offset: 0x00AC			Register Name: PD_DRV2
Bit	Read/Write	Default/Hex	Description
25:24	R/W	0x1	PD22_DRV PD22 Multi-driving Select 00: 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

15.7.5.19 0x00B4 PD Pull Register 0 (Default Value: 0x0000_0000)

Offset: 0x00B4			Register Name: PD_PUL0
Bit	Read/Write	Default/Hex	Description

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	PD0_PULL PD0 Pull_up or down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

15.7.5.20 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

Offset: 0x00B8			Register Name: PD_PUL1
Bit	Read/Write	Default/Hex	Description
9:8	R/W	0x0	PD20_PULL PD20 Pull_up or down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
7:6	R/W	0x0	PD19_PULL PD19 Pull_up or down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
5:4	R/W	0x0	PD18_PULL PD18 Pull_up or down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
3:2	R/W	0x0	PD17_PULL PD17 Pull_up or down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
1:0	R/W	0x0	PD16_PULL PD16 Pull_up or down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

15.7.5.21 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: Reserved 0011: UART2-RX 0100: LCD0-D15 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: CLK-FANOUT1 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE-EINT7 1111: Disable

Offset: 0x00C0			Register Name: PE_CFG0
Bit	Read/Write	Default/Hex	Description
27:24	R/W	0xF	PE6_SELECT PE6 Select 0000: Input 0001: Output 0010: Reserved 0011: UART2-TX 0100: LCD0-D14 0101: Reserved 0110: OWA-OUT 0111: IR-RX 1000: SPI1-CS1 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE-EINT6 1111: Disable
23:20	R/W	0xF	PE5_SELECT PE5 Select 0000: Input 0001: Output 0010: Reserved 0011: LCD0-D17 0100: TWI2-SDA 0101: IR-RX 0110: UART4-CTS 0111: UART5-RX 1000: SPI1-WP/DBI-TE 1001: SDC2-D2 1010: PWM0-3 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE-EINT5 1111: Disable
19:16	R/W	0xF	PE4_SELECT PE4 Select 0000: Input 0001: Output 0010: Reserved 0011: LCD0-D16 0100: TWI2-SCK 0101: I2S0-DIN0 0110: UART4-RTS 0111: UART5-TX 1000: SPI1-HOLD/DBI-DCX/DBI-WRX 1001: SDC2-D3 1010: PWM0-2 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE-EINT4 1111: Disable
15:12	R/W	0xF	PE3_SELECT PE3 Select 0000: Input 0001: Output 0010: Reserved 0011: LCD0-D9 0100: TWI0-SDA 0101: I2S0-DOUT0 0110: UART4-RX 0111: Reserved 1000: SPI1-MISO/DBI-SDI/DBI-TE/DBI-DCX 1001: SDC2-CMD 1010: PWM0-1 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE-EINT3 1111: Disable

Offset: 0x00C0			Register Name: PE_CFG0
Bit	Read/Write	Default/Hex	Description
11:8	R/W	0xF	PE2_SELECT PE2 Select 0000: Input 0001: Output 00010: Reserved 0011: LCD0-D8 0100: TWI0-SCK 0101: I2S0-LRCK 0110: UART4-TX 0111: Reserved 1000: SPI1-MOSI/DBI-SDO 1001: SDC2-CLK 1010: PWM0-0 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE-EINT2 1111: Disable
7:4	R/W	0xF	PE1_SELECT PE1 Select 0000: Input 0001: Output 0010: Reserved 0011: LCD0-D1 0100: TWI1-SDA 0101: I2S0-BCLK 0110: UART0-RX 0111: Reserved 1000: SPI1-CLK/DBI-SCLK 1001: SDC2-D0 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE-EINT1 1111: Disable
3:0	R/W	0xF	PE0_SELECT PE0 Select 0000: Input 0001: Output 0010: Reserved 0011: LCD0-D0 0100: TWI1-SCK 0101: I2S0-MCLK 0110: UART0-TX 0111: Reserved 1000: SPI1-CS0/DBI-CSX 1001: SDC2-D1 1010: CLK-FANOUT0 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE-EINT0 1111: Disable

15.7.5.22 0x00C4 PE Configure Register 1 (Default Value: 0xFFFF_FFFF)

Offset: 0x00C4			Register Name: PE_CFG1
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/

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: Reserved 0011: UART2-CTS 0100: TWI0-SDA 0101: Reserved 0110: UART3-RX 0111: PWM0-1 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE-EINT9 1111: Disable
3:0	R/W	0xF	PE8_SELECT PE8 Select 0000: Input 0001: Output 0010: Reserved 0011: UART2-RTS 0100: TWI0-SCK 0101: Reserved 0110: UART3-TX 0111: PWM0-0 1000: Reserved 1001: CLK-FANOUT2 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PE-EINT8 1111: Disable

15.7.5.23 0x00D0 PE Data Register (Default Value: 0x0000_0000)

Offset: 0x00D0			Register Name: PE_DAT
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13	R/W	0x0	PORT13_DAT PE13 Data If the function of PE13 port is configured as input, the PORT13_DAT bit indicates the state of the PE13 pin; If the function of PE13 port is configured as output, the state of the PE13 pin can be configured via software by modifying the value of the PORT13_DAT bit. In this case, the value obtained by reading PORT13_DAT bit is the value set by the software; If the PE13 port is configured as functional pin, this bit is invalid.

Offset: 0x00D0			Register Name: PE_DAT
Bit	Read/Write	Default/Hex	Description
12	R/W	0x0	PORT12_DAT PE12 Data If the function of PE12 port is configured as input, the PORT12_DAT bit indicates the state of the PE12 pin; If the function of PE12 port is configured as output, the state of the PE12 pin can be configured via software by modifying the value of the PORT12_DAT bit. In this case, the value obtained by reading PORT12_DAT bit is the value set by the software; If the PE12 port is configured as functional pin, this bit is invalid.
11	R/W	0x0	PORT11_DAT PE11 Data If the function of PE11 port is configured as input, the PORT11_DAT bit indicates the state of the PE11 pin; If the function of PE11 port is configured as output, the state of the PE11 pin can be configured via software by modifying the value of the PORT11_DAT bit. In this case, the value obtained by reading PORT11_DAT bit is the value set by the software; If the PE11 port is configured as functional pin, this bit is invalid.
10	R/W	0x0	PORT10_DAT PE10 Data If the function of PE10 port is configured as input, the PORT10_DAT bit indicates the state of the PE10 pin; If the function of PE10 port is configured as output, the state of the PE10 pin can be configured via software by modifying the value of the PORT10_DAT bit. In this case, the value obtained by reading PORT10_DAT bit is the value set by the software; If the PE10 port is configured as functional pin, this bit is invalid.

Offset: 0x00D0			Register Name: PE_DAT
Bit	Read/Write	Default/Hex	Description
9	R/W	0x0	PORT9_DAT PE9 Data If the function of PE9 port is configured as input, the PORT9_DAT bit indicates the sate of the PE9 pin; If the function of PE9 port is configured as output, the sate of the PE9 pin can be configured via software by modifying the value of the PORT9_DAT bit. In this case, the value obtained by reading PORT9_DAT bit is the value set by the software; If the PE9 port is configured as functional pin, this bit is invalid.
8	R/W	0x0	PORT8_DAT PE8 Data If the function of PE8 port is configured as input, the PORT8_DAT bit indicates the sate of the PE8 pin; If the function of PE8 port is configured as output, the sate of the PE8 pin can be configured via software by modifying the value of the PORT8_DAT bit. In this case, the value obtained by reading PORT8_DAT bit is the value set by the software; If the PE8 port is configured as functional pin, this bit is invalid.
7	R/W	0x0	PORT7_DAT PE7 Data If the function of PE7 port is configured as input, the PORT7_DAT bit indicates the sate of the PE7 pin; If the function of PE7 port is configured as output, the sate of the PE7 pin can be configured via software by modifying the value of the PORT7_DAT bit. In this case, the value obtained by reading PORT7_DAT bit is the value set by the software; If the PE7 port is configured as functional pin, this bit is invalid.

Offset: 0x00D0			Register Name: PE_DAT
Bit	Read/Write	Default/Hex	Description
6	R/W	0x0	PORT6_DAT PE6 Data If the function of PE6 port is configured as input, the PORT6_DAT bit indicates the sate of the PE6 pin; If the function of PE6 port is configured as output, the sate of the PE6 pin can be configured via software by modifying the value of the PORT6_DAT bit. In this case, the value obtained by reading PORT6_DAT bit is the value set by the software; If the PE6 port is configured as functional pin, this bit is invalid.
5	R/W	0x0	PORT5_DAT PE5 Data If the function of PE5 port is configured as input, the PORT5_DAT bit indicates the sate of the PE5 pin; If the function of PE5 port is configured as output, the sate of the PE5 pin can be configured via software by modifying the value of the PORT5_DAT bit. In this case, the value obtained by reading PORT5_DAT bit is the value set by the software; If the PE5 port is configured as functional pin, this bit is invalid.
4	R/W	0x0	PORT4_DAT PE4 Data If the function of PE4 port is configured as input, the PORT4_DAT bit indicates the sate of the PE4 pin; If the function of PE4 port is configured as output, the sate of the PE4 pin can be configured via software by modifying the value of the PORT4_DAT bit. In this case, the value obtained by reading PORT4_DAT bit is the value set by the software; If the PE4 port is configured as functional pin, this bit is invalid.

Offset: 0x00D0			Register Name: PE_DAT
Bit	Read/Write	Default/Hex	Description
3	R/W	0x0	PORT3_DAT PE3 Data If the function of PE3 port is configured as input, the PORT3_DAT bit indicates the sate of the PE3 pin; If the function of PE3 port is configured as output, the sate of the PE3 pin can be configured via software by modifying the value of the PORT3_DAT bit. In this case, the value obtained by reading PORT3_DAT bit is the value set by the software; If the PE3 port is configured as functional pin, this bit is invalid.
2	R/W	0x0	PORT2_DAT PE2 Data If the function of PE2 port is configured as input, the PORT2_DAT bit indicates the sate of the PE2 pin; If the function of PE2 port is configured as output, the sate of the PE2 pin can be configured via software by modifying the value of the PORT2_DAT bit. In this case, the value obtained by reading PORT2_DAT bit is the value set by the software; If the PE2 port is configured as functional pin, this bit is invalid.
1	R/W	0x0	PORT1_DAT PE1 Data If the function of PE1 port is configured as input, the PORT1_DAT bit indicates the sate of the PE1 pin; If the function of PE1 port is configured as output, the sate of the PE1 pin can be configured via software by modifying the value of the PORT1_DAT bit. In this case, the value obtained by reading PORT1_DAT bit is the value set by the software; If the PE1 port is configured as functional pin, this bit is invalid.

Offset: 0x00D0			Register Name: PE_DAT
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	PORT0_DAT PE0 Data If the function of PE0 port is configured as input, the PORT0_DAT bit indicates the state of the PE0 pin; If the function of PE0 port is configured as output, the state of the PE0 pin can be configured via software by modifying the value of the PORT0_DAT bit. In this case, the value obtained by reading PORT0_DAT bit is the value set by the software; If the PE0 port is configured as functional pin, this bit is invalid.

15.7.5.24 0x00D4 PE Multi_Driving Register 0 (Default Value: 0x1111_1111)

Offset: 0x00D4			Register Name: PE_DRV0
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:28	R/W	0x1	PE7_DRV PE7 Multi-driving Select 00: Level0 01: Level1 10: Level2 11: Level3
27:26	/	/	/
25:24	R/W	0x1	PE6_DRV PE6 Multi-driving Select 00: Level0 01: Level1 10: Level2 11: Level3
23:22	/	/	/
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

Offset: 0x00D4			Register Name: PE_DRV0
Bit	Read/Write	Default/Hex	Description
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

15.7.5.25 0x00D8 PE Multi_Driving Register 1 (Default Value: 0x1111_1111)

Offset: 0x00D8			Register Name: PE_DRV1
Bit	Read/Write	Default/Hex	Description
31:22	/	/	/
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	/	/	/

Offset: 0x00D8			Register Name: PE_DRV1
Bit	Read/Write	Default/Hex	Description
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

15.7.5.26 0x00E4 PE Pull Register 0 (Default Value: 0x0000_0000)

Offset: 0x00E4			Register Name: PE_PUL0
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
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
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

Offset: 0x00E4			Register Name: PE_PUL0
Bit	Read/Write	Default/Hex	Description
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

15.7.5.27 0x00F0 PF Configure Register 0 (Default Value: 0x0FFF_FFFF)

Offset: 0x00F0			Register Name: PF_CFG0
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/

Offset: 0x00F0			Register Name: PF_CFG0
Bit	Read/Write	Default/Hex	Description
27:24	R/W	0xF	PF6_SELECT PF6 Select 0000: Input 0001: Output 0010: PWM0-3 0011: IR-RX 0100: SPI0-CS1 0101: SPIF0-CS1 0110: UART1-RTS 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PF-EINT6 1111: Disable
23:20	R/W	0xF	PF5_SELECT PF5 Select 0000: Input 0001: Output 0010: SDC0-D2 0011: DJTAG-CK 0100: RJTAG-CK 0101: TWI1-SDA 0110: UART1-RX 0111: TWI2-SDA 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PF-EINT5 1111: Disable
19:16	R/W	0xF	PF4_SELECT PF4 Select 0000: Input 0001: Output 0010: SDC0-D3 0011: UART0-RX 0100: TWI0-SDA 0101: Reserved 0110: UART1-TX 0111: TWI2-SCK 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PF-EINT4 1111: Disable
15:12	R/W	0xF	PF3_SELECT PF3 Select 0000: Input 0001: Output 0010: SDC0-CMD 0011: DJTAG-DO 0100: RJTAG-DO 0101: TWI1-SCK 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PF-EINT3 1111: Disable

Offset: 0x00F0			Register Name: PF_CFG0
Bit	Read/Write	Default/Hex	Description
11:8	R/W	0xF	PF2_SELECT PF2 Select 0000: Input 0001: Output 00010: SDC0-CLK 0011: UART0-TX 0100: TWI0-SCK 0101: Reserved 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PF-EINT2 1111: Disable
7:4	R/W	0xF	PF1_SELECT PF1 Select 0000: Input 0001: Output 0010: SDC0-D0 0011: DJTAG-DI 0100: RJTAG-DI 0101: PWM0-1 0110: Reserved 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PF-EINT1 1111: Disable
3:0	R/W	0xF	PF0_SELECT PF0 Select 0000: Input 0001: Output 0010: SDC0-D1 0011: DJTAG-MS 0100: RJTAG-MS 0101: PWM0-0 0110: UART1-CTS 0111: Reserved 1000: Reserved 1001: Reserved 1010: Reserved 1011: Reserved 1100: Reserved 1101: Reserved 1110: PF-EINT0 1111: Disable

15.7.5.28 0x0100 PF Data Register (Default Value: 0x0000_0000)

Offset: 0x0100			Register Name: PF_DAT
Bit	Read/Write	Default/Hex	Description
31:7	/	/	/

Offset: 0x0100			Register Name: PF_DAT
Bit	Read/Write	Default/Hex	Description
6	R/W	0x0	PORT6_DAT PF6 Data If the function of PF6 port is configured as input, the PORT6_DAT bit indicates the sate of the PF6 pin; If the function of PF6 port is configured as output, the sate of the PF6 pin can be configured via software by modifying the value of the PORT6_DAT bit. In this case, the value obtained by reading PORT6_DAT bit is the value set by the software; If the PF6 port is configured as functional pin, this bit is invalid.
5	R/W	0x0	PORT5_DAT PF5 Data If the function of PF5 port is configured as input, the PORT5_DAT bit indicates the sate of the PF5 pin; If the function of PF5 port is configured as output, the sate of the PF5 pin can be configured via software by modifying the value of the PORT5_DAT bit. In this case, the value obtained by reading PORT5_DAT bit is the value set by the software; If the PF5 port is configured as functional pin, this bit is invalid.
4	R/W	0x0	PORT4_DAT PF4 Data If the function of PF4 port is configured as input, the PORT4_DAT bit indicates the sate of the PF4 pin; If the function of PF4 port is configured as output, the sate of the PF4 pin can be configured via software by modifying the value of the PORT4_DAT bit. In this case, the value obtained by reading PORT4_DAT bit is the value set by the software; If the PF4 port is configured as functional pin, this bit is invalid.

Offset: 0x0100			Register Name: PF_DAT
Bit	Read/Write	Default/Hex	Description
3	R/W	0x0	PORT3_DAT PF3 Data If the function of PF3 port is configured as input, the PORT3_DAT bit indicates the sate of the PF3 pin; If the function of PF3 port is configured as output, the sate of the PF3 pin can be configured via software by modifying the value of the PORT3_DAT bit. In this case, the value obtained by reading PORT3_DAT bit is the value set by the software; If the PF3 port is configured as functional pin, this bit is invalid.
2	R/W	0x0	PORT2_DAT PF2 Data If the function of PF2 port is configured as input, the PORT2_DAT bit indicates the sate of the PF2 pin; If the function of PF2 port is configured as output, the sate of the PF2 pin can be configured via software by modifying the value of the PORT2_DAT bit. In this case, the value obtained by reading PORT2_DAT bit is the value set by the software; If the PF2 port is configured as functional pin, this bit is invalid.
1	R/W	0x0	PORT1_DAT PF1 Data If the function of PF1 port is configured as input, the PORT1_DAT bit indicates the sate of the PF1 pin; If the function of PF1 port is configured as output, the sate of the PF1 pin can be configured via software by modifying the value of the PORT1_DAT bit. In this case, the value obtained by reading PORT1_DAT bit is the value set by the software; If the PF1 port is configured as functional pin, this bit is invalid.

Offset: 0x0100			Register Name: PF_DAT
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	PORT0_DAT PF0 Data If the function of PF0 port is configured as input, the PORT0_DAT bit indicates the state of the PF0 pin; If the function of PF0 port is configured as output, the state of the PF0 pin can be configured via software by modifying the value of the PORT0_DAT bit. In this case, the value obtained by reading PORT0_DAT bit is the value set by the software; If the PF0 port is configured as functional pin, this bit is invalid.

15.7.5.29 0x0104 PF Multi_Driving Register 0 (Default Value: 0x0111_1111)

Offset: 0x0104			Register Name: PF_DRV0
Bit	Read/Write	Default/Hex	Description
31:26	/	/	/
25:24	R/W	0x1	PF6_DRV PF6 Multi-driving Select 00: Level0 01: Level1 10: Level2 11: Level3
23:22	/	/	/
21:20	R/W	0x1	PF5_DRV PF5 Multi-driving Select 00: Level0 01: Level1 10: Level2 11: Level3
19:18	/	/	/
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

Offset: 0x0104			Register Name: PF_DRV0
Bit	Read/Write	Default/Hex	Description
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	PF0_DRV PF0 Multi-driving Select 00: Level0 01: Level1 10: Level2 11: Level3

15.7.5.30 0x0114 PF Pull Register 0 (Default Value: 0x0000_0000)

Offset: 0x0114			Register Name: PF_PUL0
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
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

Offset: 0x0114			Register Name: PF_PUL0
Bit	Read/Write	Default/Hex	Description
1:0	R/W	0x0	PF0_PULL PF0 Pull_up or down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

15.7.5.31 0x0200 PA External Interrupt Configure Register 0 (Default Value: 0x0000_0000)

Offset: 0x0200			Register Name: PA_INT_CFG0
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:12	R/W	0x0	EINT3_CFG External INT3 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge(Positive/Negative) Others: Reserved
11:8	R/W	0x0	EINT2_CFG External INT2 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge(Positive/Negative) Others: Reserved
7:4	R/W	0x0	EINT1_CFG External INT1 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge(Positive/Negative) Others: Reserved

Offset: 0x0200			Register Name: PA_INT_CFG0
Bit	Read/Write	Default/Hex	Description
3:0	R/W	0x0	EINT0_CFG External INT0 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge(Positive/Negative) Others: Reserved

15.7.5.32 0x0210 PA External Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0210			Register Name: PA_INT_CTL
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
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

15.7.5.33 0x0214 PA External Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0214			Register Name: PA_INT_STA
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
3	R/C	0x0	EINT3_STATUS External INT3 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
2	R/C	0x0	EINT2_STATUS External INT2 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
1	R/C	0x0	EINT1_STATUS External INT1 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
0	R/C	0x0	EINT0_STATUS External INT0 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

15.7.5.34 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	/	/	/
0	R/W	0x0	GPIO_INT_CLK_SELECT GPIO Interrupt Clock Select 0: LOSC 32 kHz 1: HOSC 24 MHz

15.7.5.35 0x0220 PB External Interrupt Configure Register 0 (Default Value: 0x0000_0000)

Offset: 0x0220			Register Name: PB_INT_CFG0
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/

Offset: 0x0220			Register Name: PB_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	/	/	/

15.7.5.36 0x0230 PB External Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0230			Register Name: PB_INT_CTL
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
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: 0x0230			Register Name: PB_INT_CTL
Bit	Read/Write	Default/Hex	Description
1	R/W	0x0	EINT1_CTL External INT1 Enable 0: Disable 1: Enable
0	/	/	/

15.7.5.37 0x0234 PB External Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0234			Register Name: PB_INT_STA
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
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

15.7.5.38 0x0238 PB External Debounce Configure Register (Default Value: 0x0000_0000)

Offset: 0x0238			Register Name: PB_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.

Offset: 0x0238			Register Name: PB_INT_DEB
Bit	Read/Write	Default/Hex	Description
3:1	/	/	/
0	R/W	0x0	GPIO_INT_CLK_SELECT GPIO Interrupt Clock Select 0: LOSC 32 kHz 1: HOSC 24 MHz

15.7.5.39 0x0240 PC External Interrupt Configure Register 0 (Default Value: 0x0000_0000)

Offset: 0x0240			Register Name: PC_INT_CFG0
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
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: 0x0240			Register Name: PC_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

15.7.5.40 0x0250 PC External Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0250			Register Name: PC_INT_CTL
Bit	Read/Write	Default/Hex	Description
31:6	/	/	/
5	R/W	0x0	EINT5_CTL External INT5 Enable 0: Disable 1: Enable
4	R/W	0x0	EINT4_CTL External INT4 Enable 0: Disable 1: Enable

Offset: 0x0250			Register Name: PC_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

15.7.5.41 0x0254 PC External Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0254			Register Name: PC_INT_STA
Bit	Read/Write	Default/Hex	Description
31:6	/	/	/
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: 0x0254			Register Name: PC_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

15.7.5.42 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	/	/	/
0	R/W	0x0	GPIO_INT_CLK_SELECT GPIO Interrupt Clock Select 0: LOSC 32 kHz 1: HOSC 24 MHz

15.7.5.43 0x0260 PD External Interrupt Configure Register 0 (Default Value: 0x0000_0000)

Offset: 0x0260			Register Name: PD_INT_CFG0
Bit	Read/Write	Default/Hex	Description

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
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: 0x0260			Register Name: PD_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

15.7.5.44 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

Offset: 0x0264			Register Name: PD_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: 0x0264			Register Name: PD_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

15.7.5.45 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
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

Offset: 0x0268			Register Name: PD_INT_CFG2
Bit	Read/Write	Default/Hex	Description
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
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

15.7.5.46 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
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

Offset: 0x0270			Register Name: PD_INT_CTL
Bit	Read/Write	Default/Hex	Description
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
5	R/W	0x0	EINT5_CTL External INT5 Enable 0: Disable 1: Enable
4	R/W	0x0	EINT4_CTL External INT4 Enable 0: Disable 1: Enable

Offset: 0x0270			Register Name: PD_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

15.7.5.47 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
20	R/W	0x0	EINT20_STATUS External INT20 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
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
12	R/W	0x0	EINT12_STATUS External INT12 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
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
4	R/W	0x0	EINT4_STATUS External INT4 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
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

15.7.5.48 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	GPIO_INT_CLK_SELECT GPIO Interrupt Clock Select 0: LOSC 32 kHz 1: HOSC 24 MHz

15.7.5.49 0x0280 PE External Interrupt Configure Register 0 (Default Value: 0x0000_0000)

Offset: 0x0280			Register Name: PE_INT_CFG0
Bit	Read/Write	Default/Hex	Description

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

15.7.5.50 0x0284 PE External Interrupt Configure Register 1 (Default Value: 0x0000_0000)

Offset: 0x0284			Register Name: PE_INT_CFG1
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:20	R/W	0x0	EINT13_CFG External INT13 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge(Positive/Negative) Others: Reserved

Offset: 0x0284			Register Name: PE_INT_CFG1
Bit	Read/Write	Default/Hex	Description
19:16	R/W	0x0	EINT12_CFG External INT12 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge(Positive/Negative) Others: Reserved
15:12	R/W	0x0	EINT11_CFG External INT11 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge(Positive/Negative) Others: Reserved
11:8	R/W	0x0	EINT10_CFG External INT10 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge(Positive/Negative) Others: Reserved
7:4	R/W	0x0	EINT9_CFG External INT9 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge(Positive/Negative) Others: Reserved
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

15.7.5.51 0x0290 PE External Interrupt Control Register (Default Value: 0x0000_0000)

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

Offset: 0x0290			Register Name: PE_INT_CTL
Bit	Read/Write	Default/Hex	Description
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

15.7.5.52 0x0294 PE External Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0294			Register Name: PE_INT_STA
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
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

15.7.5.53 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	GPIO_INT_CLK_SELECT GPIO Interrupt Clock Select 0: LOSC 32 kHz 1: HOSC 24 MHz

15.7.5.54 0x02A0 PF External Interrupt Configure Register 0 (Default Value: 0x0000_0000)

Offset: 0x02A0			Register Name: PF_INT_CFG0
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/

Offset: 0x02A0			Register Name: PF_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: 0x02A0			Register Name: PF_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

15.7.5.55 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

Offset: 0x02B0			Register Name: PF_INT_CTL
Bit	Read/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

15.7.5.56 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

Offset: 0x02B4			Register Name: PF_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

15.7.5.57 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	GPIO_INT_CLK_SELECT GPIO Interrupt Clock Select 0: LOSC 32 kHz 1: HOSC 24 MHz

15.7.5.58 0x0348 GPIO Group Power Value Register (Default Value: 0x0100_1550)

Offset: 0x0348			Register Name: GPIO_POW_VAL
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:10	R	0x1	PF_PWR_VAL PF_PORT POWER VALUE. 00: 1.8V 01: 3.3V

Offset: 0x0348			Register Name: GPIO_POW_VAL
Bit	Read/Write	Default/Hex	Description
9:8	R	0x1	PE_PWR_VAL PE_PORT POWER VALUE. 00: 1.8V 01: 3.3V
7:6	R	0x1	PD_PWR_VAL PD_PORT POWER VALUE. 00: 1.8V 01: 3.3V
5:4	R	0x1	PC_PWR_VAL PC_PORT POWER VALUE. 00: 1.8V 01: 3.3V
3:0	/	/	/

15.7.5.59 0x0350 GPIO Group Power Voltage Select Control Register (Default Value: 0x0000_0001)

Offset: 0x0350			Register Name: GPIO_POW_VOL_SEL_CTL
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x1	VCC-PF POWER VOLTAGE SELECT CONTROL 0: 1.8V 1: 3.3V

15.8 UART

15.8.1 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.

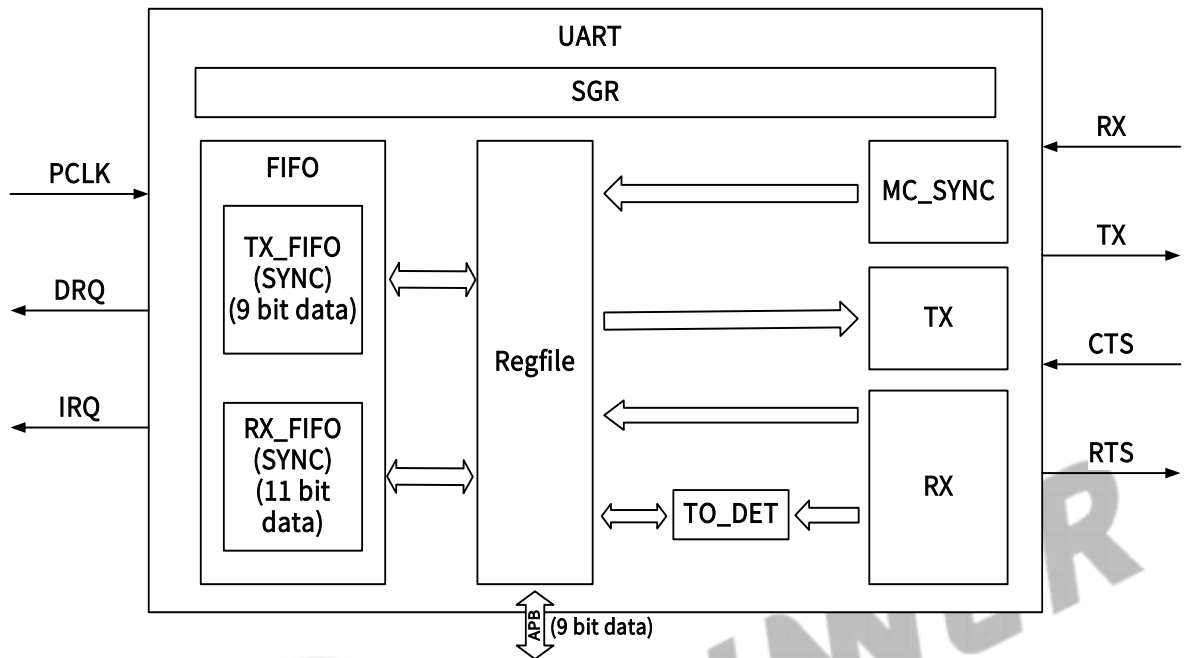
The UART has the following features:

- Up to 6 UART controllers: UART0, UART1, UART2, UART3, UART4, UART5
- 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 UART1-5
- Supports baud rate generator from APB bus clock
 - Speed up to 4 Mbit/s with 64 MHz APB clock
 - Speed up to 5 Mbit/s with 80 MHz APB clock
 - Speed up to 3.75 Mbit/s with 60 MHz APB clock
 - Speed up to 1.5 Mbit/s with 24 MHz APB clock
 - Speed up to 6.25 Mbit/s with 100 MHz APB clock
- Supports 5-8 data bits and 1/1.5/2 stop bits
- Supports Even, Odd or No Parity
- Supports TX/RX DMA Slave interface
- Supports Software/ Hardware Flow Control
- Supports IrDA 1.0 SIR
- Supports RS-485/9-bit mode
- Supports Auto-Flow by using CTS & RTS (excluding UART0)

15.8.2 Block Diagram

The following figure shows a block diagram of the UART.

Figure 15-40 UART Block Diagram



15.8.3 External Signals

The following table describes the external signals of UART.

Table 15-21 UART External Signals

Signal Name	Description	Type
UART0		
UART0-RX	UART0 Data Receiver	I
UART0-TX	UART0 Data Transmitter	O
UART1		
UART1-CTS	UART1 Clear to Send	I
UART1-RTS	UART1 Request to Send	O
UART1-RX	UART1 Data Receiver	I
UART1-TX	UART1 Data Transmitter	O
UART2		
UART2-CTS	UART2 Clear to Send	I
UART2-RTS	UART2 Request to Send	O
UART2-RX	UART2 Data Receiver	I
UART2-TX	UART2 Data Transmitter	O
UART3		
UART3-CTS	UART3 Clear to Send	I
UART3-RTS	UART3 Request to Send	O
UART3-RX	UART3 Data Receiver	I
UART3-TX	UART3 Data Transmitter	O

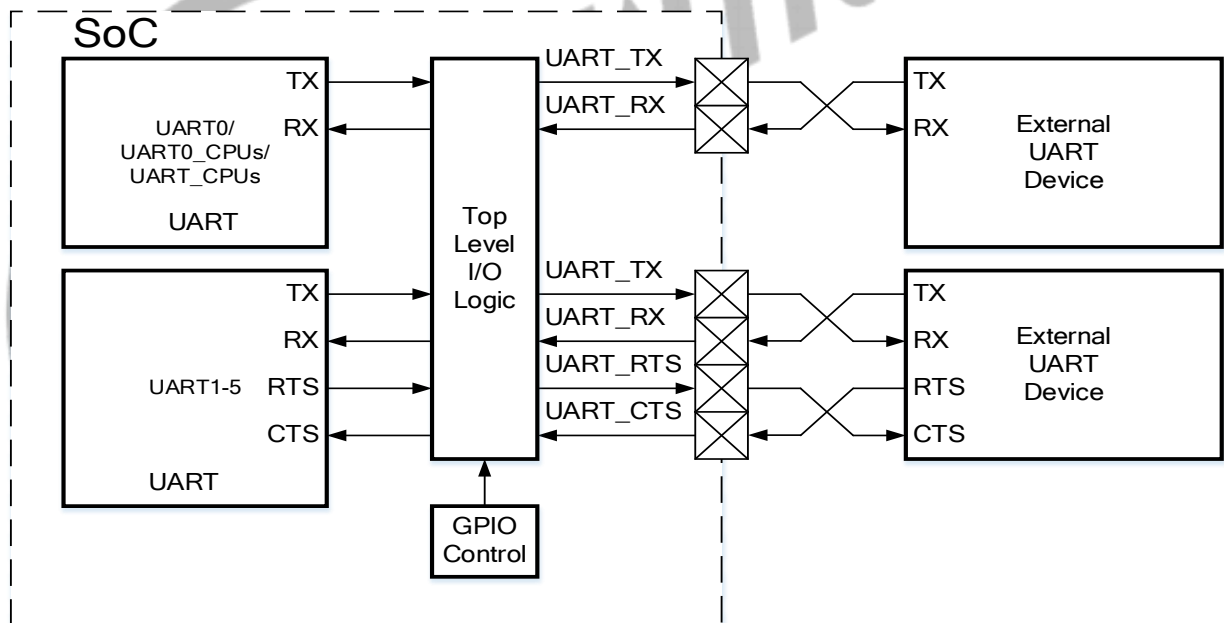
Signal Name	Description	Type
UART4		
UART4-CTS	UART4 Clear to Send	I
UART4-RTS	UART4 Request to Send	O
UART4-RX	UART4 Data Receiver	I
UART4-TX	UART4 Data Transmitter	O
UART5		
UART5-CTS	UART5 Clear to Send	I
UART5-RTS	UART5 Request to Send	O
UART5-RX	UART5 Data Receiver	I
UART5-TX	UART5 Data Transmitter	O

15.8.4 Functional Description

15.8.4.1 Typical Application

The following figure shows the I/O signals of UART controllers. To connect external signals, you need to configure GPIO controller to enable UART model.

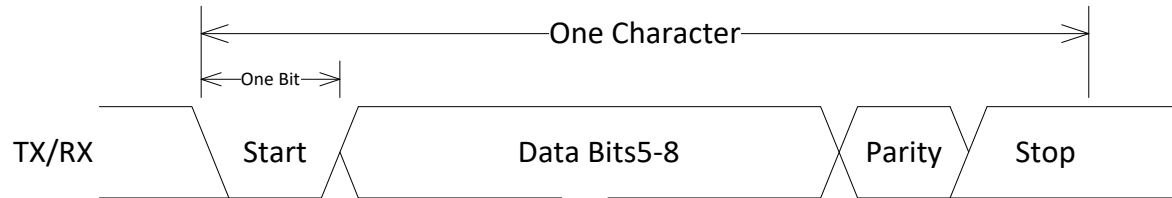
Figure 15-41 I/O Signals of UART Controllers (2/4-wire)



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 15-42 UART Serial Data Format



Start bit: It is the start flag of a data frame. In accordance with 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 bit: The data bit width can be configured as 5-bit, 6-bit, 7-bit, and 8-bit based on the 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](#) register. If RS-485 mode is enabled, the parity bit must be enabled.

Stop bit: It is the stop bit of a data frame. It can be set to 1-bit, 1.5-bit, and 2-bit by the [UART_LCR](#) register. The high level of the TXD signal indicates the end of a data frame.

RTS/CTS Auto Flow Control

The following figure shows the typical application diagram for RTS/CTS auto flow control. Figure 15-44 shows the data format of the RTS/CTS auto flow control.

Figure 15-43 Application Diagram for RTS/CTS Auto flow Control

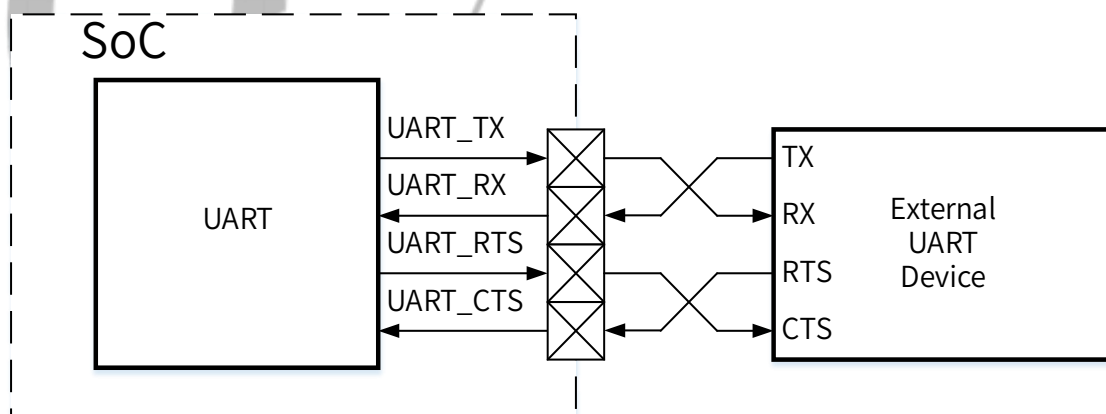
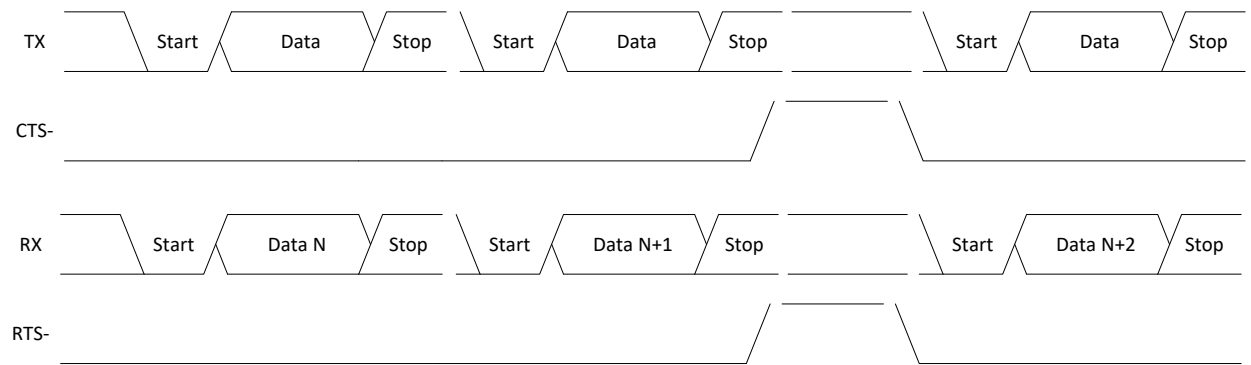


Figure 15-44 RTS/CTS Auto Flow Control Data Format



Serial IrDA

The following figure shows the application diagram for the IrDA transceiver. Figure 15-46 shows the data format of the serial IrDA.

Figure 15-45 Application Diagram for IrDA Transceiver

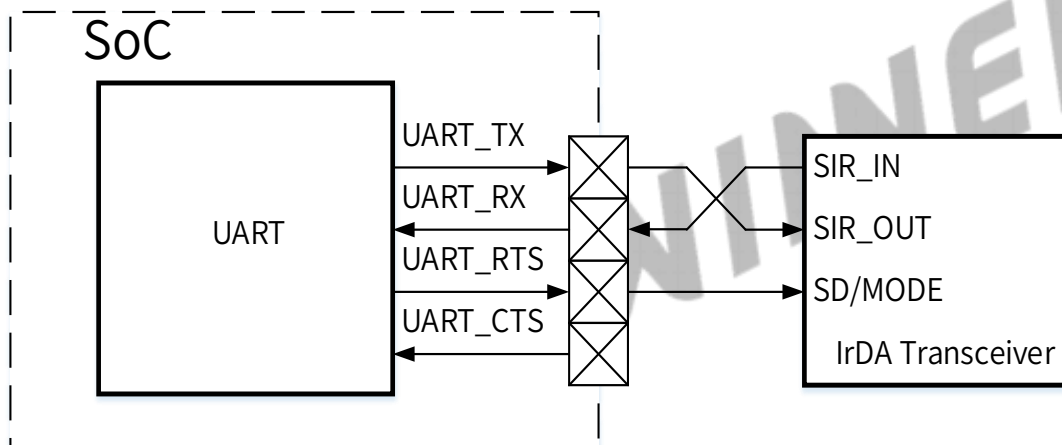
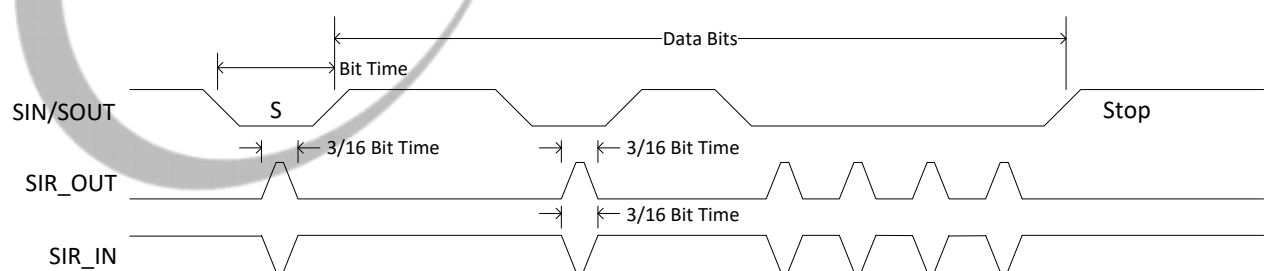


Figure 15-46 Serial IrDA Data Format



RS-485

Figure 15-47 shows the application diagram for the RS-485 transceiver. Figure 15-48 shows the data format of the RS-485.

Figure 15-47 Application Diagram for RS-485 Transceiver

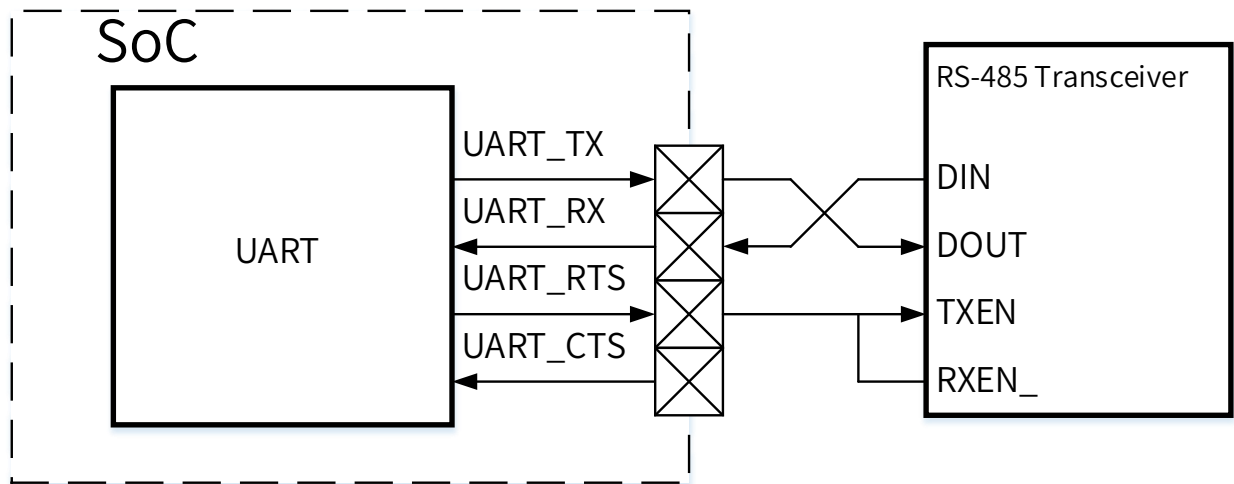
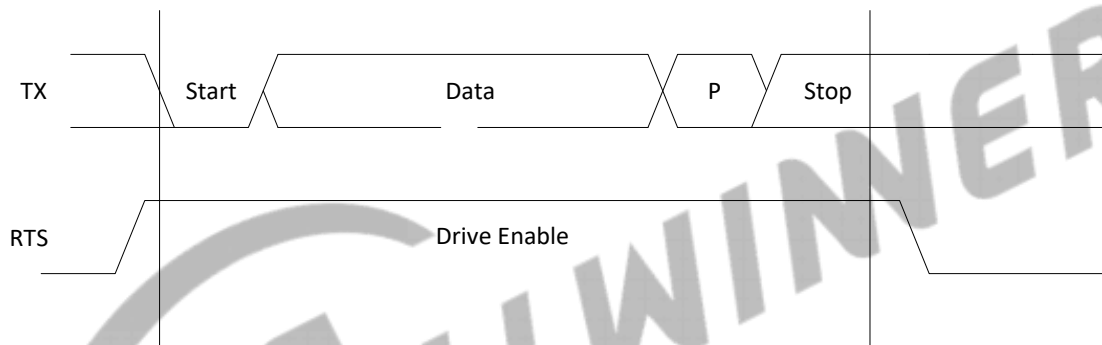


Figure 15-48 RS-485 Data Format



15.8.4.2 Baud and Error Rates

The baud rate is calculated as follows: Baud rate = APB_UART_CLK / (16 * divisor). APB_UART_CLK can be configured in section 4.1.6.26 0x0528 APB_UART Clock Register (Default Value: 0x0000_0000).

The divisor is frequency divider of UART, which has 16-bit. The low 8-bit is in the [UART_DLL](#) register, and the high 8-bit is in the [UART_DLH](#) register.

The following figures show the errors of baud rate in different clock sources.

Table 15-22 Errors of Baud Rate in UART Mode

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

Clock Source	Divisor	Baud Rate	Over Sampling	Error(%)
24000000	26	57600	16	0.16
24000000	13	115200	16	0.16
48000000	13	230400	16	0.16
60000000	1	3750000	16	0
75000000	5	921600	16	1.725
48000000	3	1000000	16	0
24000000	1	1500000	16	0
48000000	1	3000000	16	0
80000000	1	5000000	16	0
160000000	1	10000000	16	0

Table 15-23 Errors of Baud Rate in IrDA Mode

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 15-24 Errors of Baud Rate in RS485 Mode

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
48000000	13	230400	16	0.16
60000000	1	3750000	16	0
75000000	5	921600	16	1.725
48000000	3	1000000	16	0
24000000	1	1500000	16	0
48000000	1	3000000	16	0

Clock source	Divisor	Baud rate	Encoding	Error(%)
80000000	1	5000000	16	0
160000000	1	10000000	16	0

Table 15-25 Errors of Baud Rate in 19.2M and 26M Clock Source

Clock source	Divisor	Baud rate	Encoding	Error(%)
19200000	125	9600	16	0
26000000	14	115200	16	0.75%

NOTE

Some Clock source frequencies are not supported in this project. Please refer to 4.1 Clock Controller Unit (CCU).

15.8.4.3 DLAB Definition

The DLAB control bit ([UART_LCR\[7\]](#)) is the access control bit of the divisor Latch low/high register.

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

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

When the UART initializes, the divisor needs to be set. Write 1 to DLAB to configure [UART_DLL](#) and [UART_DLH](#), and then write 0 to DLAB to configure [UART_RBR/UART_THR](#).

15.8.4.4 CHCFG_AT_BUSY Definition

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

CHCFG_AT_BUSY: If this bit is enabled and UART is busy, the software can also set the UART controller, such as the [UART_LCR](#), [UART_DLH](#), [UART_DLL](#) 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 the update completes, the bit is cleared to 0 automatically.

To set the divisor, write 1 to CHCFG_AT_BUSY, and Write 1 to DLAB to configure [UART_DLL](#) and [UART_DLH](#), then Write 1 to CHANGE_UPDATE.

15.8.4.5 UART_USR Definition

UART_USR is the BUSY flag, used for indicating the UART controller is busy or not.

When TX is sending data, or RX is receiving data, or TX FIFO/RX FIFO is not null, the UART_USR will be asserted. It indicates the UART controller is busy.

15.8.5 Programming Guidelines

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

15.8.5.1 Initialization

Step 1 System Initialization

- Configure [APB_UART_CLK_REG](#) in the 4.1 Clock Controller Unit (CCU) to set the APB_UART clock.
- Set UARTx_BGR_REG[UARTx_GATING] to 1 to enable the module clock, and set UARTx_BGR_REG[UARTx_RST] to 1 to de-assert the module.

Step 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 15.7 GPIO).
- Baud-rate configuration:
 - Set UART baud-rate (refer to section 15.8.4.2 Baud and Error Rates);
 - Write [UART_FCR](#)[FIFOE] to 1 to enable TX/RX FIFO;
 - Write [UART_HALT](#)[HALT_TX] to 1 to disable TX transfer;
 - Set [UART_LCR](#)[DLAB] to 1, remain default configuration for other bits; set 0x00 offset address to the [UART_DLL](#) register, set 0x04 offset address to the [UART_DLH](#) register;
 - Write the high 8-bit of divisor to the [UART_DLH](#) register, and write the low 8-bit of divisor to the [UART_DLL](#) register;
 - Set [UART_LCR](#)[DLAB] to 0, remain default configuration for other bits; set 0x00 offset address to the [UART_RBR](#)/[UART_THR](#) register, set 0x04 offset address to the [UART_IER](#) register;
 - Set [UART_HALT](#)[HALT_TX] to 0 to enable TX transfer.

Step 3 Controller Parameter Configuration

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

Step 4 Interrupt Configuration

- Configure UART interrupt vector number to request UART interrupt (Refer to Section 10.1 Platform-Level Interrupt Controller (PLIC) for interrupt vector number).

- In DMA mode, write [UART_IER](#) to 0 to disable interrupt; write [UART_HSK](#)[Handshake configuration] to 0xE5 to set DMA handshake mode; write [UART_FCR](#)[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](#) to enable the corresponding interrupt according to requirements: such as transmit (TX) interrupt, receive (RX) interrupt, receive line status interrupt, RS485 interrupt, etc. (Here TX/RX interrupt is usually used).

15.8.5.2 Transmission in DMA Mode

- Step 1** Initialize UART model. Refer to section 15.8.5.1 Initialization for initialization steps.
- Step 2** Configure UART baud rate by setting [UART_DLL](#) and [UART_DLH](#).
- Step 3** Configure UART data format by setting [UART_LCR](#).
- Step 4** Initialize UART FIFO by setting [UART_FCR](#) and [UART_FCC](#).
- Step 5** Set PTE and DMA_PTE_RX by setting [UART_HALT](#).
- Step 6** Configure DMAC for UART.
- Step 7** Wait for DMA done by checking DMAC status.

15.8.5.3 Transmission in Interrupt Mode

- Step 1** Initialize UART model. Refer to section 15.8.5.1 for initialization steps.
- Step 2** Configure UART baud rate by setting [UART_DLL](#) and [UART_DLH](#).
- Step 3** Configure UART data format by setting [UART_LCR](#).
- Step 4** Initialize UART FIFO by setting [UART_FCR](#) and [UART_FCC](#).
- Step 5** Configure UART interrupt by [UART_IER](#) and [UART_IIR](#).
- Step 6** Wait for UART Interrupt service done.

15.8.5.4 Transmission in RS485 Mode

Select RS-485 Mode

- Step 1** Initialize UART model. Refer to section 15.8.5.1 for initialization steps.
- Step 2** Configure UART baud rate by setting [UART_DLL](#) and [UART_DLH](#).
- Step 3** Configure UART data format by setting [UART_LCR](#).
- Step 4** Initialize UART FIFO by setting [UART_FCR](#) and [UART_FCC](#).
- Step 5** Select RS485 Mode by setting [UART_MCR](#) [7:6].

Configure RS485 Receive Mode

Step 1 Configure UART as RS485 mode.

Step 2 Write 1 to [UART_LCR](#) [3].

Step 3 Configure UART RS485 Receive data format by setting [UART_485_CTL](#) [1:0].

-----If select ADD mode -----

Step 1 Configure AAD broadcast mode by setting [UART_485_CTL](#) [8].

In broadcast mode, UART will receive all data.

In non-broadcast mode, UART will not receive data if it does not receive a matching address.

Step 2 Configure the matching address by setting [UART_RS485_ADDR_MATCH](#).

Configure RS485 Half-Duplex Mode

Step 1 Configure UART as RS485 mode.

Step 2 Write 1 to [UART_485_CTL](#) [4] to select half-duplex mode.

Step 3 Write 1 to [UART_MCR](#) [5] to enable auto flow control mode.

Step 4 Write 1 to the [UART_FCR](#) [0] to enable FIFOs.

The processing logic of RS485 half-duplex hardware autotransition is as follows.

When there is data in the TXFIFO or TX shift register, the data will be sent first, and the RTS signal will be pulled high.

When there is no data in the TXFIFO or TX shift register, RTS is pulled low, indicating data reception. In this condition, you can also extend the duration that RTS is held high by configuring the [RTSN_RST_DLY_TIME](#) register. After the duration ends, the RTS will be pulled low.



NOTE

In RS485 half-duplex mode, only one UART can send data at a time.

15.8.5.5 FIFO Test Mode

Step 1 Configure FIFO test mode by setting [FIFO_TM](#).

Step 2 Write data to the TXFIFO by write [UART_THR](#) and read data from the TXFIFO by read [READ_TX_FIFO](#).

Step 3 Write data to the RXFIFO by write [WRITE_RX_FIFO](#) and read data from the RXFIFO by read [READ_RX_FIFO](#).

15.8.6 Register List

Module Name	Base Address
UART0	0x02500000
UART1	0x02500400
UART2	0x02500800
UART3	0x02500C00
UART4	0x02501000
UART5	0x02501400

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_FCR_1	0x00A0	UART FIFO Control Register1
UART_HALT	0x00A4	UART Halt TX Register
UART_DBG_DLL	0x00B0	UART Debug DLL Register
UART_DBG_DLH	0x00B4	UART Debug DLH Register
UART_485_CTL	0x00C0	UART RS485 Control Configure and Status Register
UART_RS485_ADDR_MATCH	0x00C4	UART RS485 Address Match Register
BUS_IDLE_CHECK	0x00C8	UART RS485 Bus Idle Check Register
TX_DELAY	0x00CC	UART TX Delay Register
FIFO_TM	0x00D4	UART FIFO Test Mode Register
READ_TX_FIFO	0x00D8	UART Read TX FIFO Register
WRITE_RX_FIFO	0x00DC	UART Write RX FIFO Register
READ_RX_FIFO	0x00E0	UART Read RX FIFO Register
RTSN_RST_DLY_TIME	0x00E4	UART RS485 RTSN Pull Down Delay Time Register
UART_FCC	0x00F0	UART FIFO Clock Control Register

15.8.7 Register Description

15.8.7.1 0x0000 UART Receive Buffer Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: UART_RBR
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8:0	R	0x0	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 the Line Status Register (LCR) is set. If in FIFO mode and FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO is preserved, but any incoming data are lost and an overrun error occurs.

15.8.7.2 0x0000 UART Transmit Holding Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: UART_THR
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8:0	W	0x0	THR Transmit Holding Register Data to be 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 THR when the THR Empty (THRE) bit (LSR[5]) is set. Any attempt to write data when the FIFO is full results in the write data being lost.

15.8.7.3 0x0000 UART Divisor Latch Low Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: UART_DLL
Bit	Read/Write	Default/Hex	Description

Offset: 0x0000			Register Name: UART_DLL
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0x0	DLL Divisor Latch Low Lower 8 bits of a 16-bit, read/write, Divisor Latch register that contains the baud rate divisor for the UART. This register may only be accessed when the DLAB bit (LCR[7]) is set and the UART is not busy (USR[0] is zero). 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). Note that with the Divisor Latch Registers (DLL and DLH) set to zero, 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.

15.8.7.4 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	DLH Divisor Latch High Upper 8 bits of a 16-bit, read/write, Divisor Latch register that contains the baud rate divisor for the UART. This register may only be accessed when the DLAB bit (LCR[7]) is set and the UART is not busy (USR[0] is zero). 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). Note that with the Divisor Latch Registers (DLL and DLH) set to zero, 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.

15.8.7.5 0x0004 UART Interrupt Enable Register (Default Value: 0x0000_0000)

Offset: 0x0004			Register Name: UART_IER
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	PTIME Programmable THRE Interrupt Mode Enable This is used to enable/disable the generation of THRE Interrupt. 0: Disable 1: Enable
6:5	/	/	/
4	R/W	0x0	RS485_INT_EN RS485 Interrupt Enable 0: Disable 1: Enable
3	R/W	0x0	EDSSI Enable Modem Status Interrupt This is used to enable/disable the generation of Modem Status Interrupt. This is the fourth highest priority interrupt. 0: Disable 1: Enable
2	R/W	0x0	ELSI Enable Receiver Line Status Interrupt This is used to enable/disable the generation of Receiver Line Status Interrupt. This is the highest priority interrupt. 0: Disable 1: Enable
1	R/W	0x0	ETBEI Enable Transmit Holding Register Empty Interrupt This is used to enable/disable the generation of Transmitter Holding Register Empty Interrupt. This is the third highest priority interrupt. 0: Disable 1: Enable

Offset: 0x0004			Register Name: UART_IER
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	ERBFI Enable Received Data Available Interrupt 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 interrupts. 0: Disable 1: Enable

15.8.7.6 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	FEFLAG FIFOs Enable Flag This is used to indicate whether the FIFOs are enabled or disabled. 00: Disable 11: Enable
5:4	/	/	/
3:0	R	0x1	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 (pending by UART_485_CTL[6:5]) 0100: Received data available 0110: Receiver line status 0111: Busy detect 1100: Character timeout Bit 3 indicates an interrupt can only occur when the FIFOs are enabled and used to distinguish a Character Timeout condition interrupt.

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)
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 1 character in it during This time	Reading the receiver buffer register
0010	Fifth	Transmit holding register empty	Transmitter holding register empty (Program THRE Mode disabled) or XMIT FIFO at or below threshold (Program THRE Mode enabled)	Reading the IIR register (if 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 auto flow 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

15.8.7.7 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	RT RCVR Trigger This is used to select the trigger level in the receiver FIFO at which the Received Data Available Interrupt is generated. In auto flow 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. 00: 1 character in the FIFO 01: FIFO ¼ full 10: FIFO ½ full 11: FIFO-2 less than full
5:4	W	0x0	TFT TX Empty Trigger This is used to select the empty threshold level at which the THRE Interrupts are generated when the mode is active. It also determines when the dma_tx_req_n signal is asserted when in certain modes of operation. 00: FIFO empty 01: 2 characters in the FIFO 10: FIFO ¼ full 11: FIFO ½ full

Offset: 0x0008			Register Name: UART_FCR
Bit	Read/Write	Default/Hex	Description
3	W	0x0	DMAM DMA Mode 0: Mode 0 In this mode, when PTE is high and TX FIFO is enable, the TX DMA request will be set when TFL is less than or equal to FIFO Trigger Level(otherwise it will be cleared). When PTE is high and TX FIFO is disabled, the TX DMA request will be set only if THR is empty. If 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 dma_pte_rx is high and RX FIFO is enabled, the rx drq will be set only if RFL is equal to or more than FIFO Trigger Level, otherwise it will be cleared. 1: Mode 1 In this mode, TX FIFO should be enable. If the PTE is high, the TX DMA request will be set when TFL is less than or equal to FIFO Trigger Level; If 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 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.
2	W	0x0	XFIFOR XMIT FIFO Reset This resets the control portion of the transmit FIFO and treats the FIFO as empty. This also de-asserts the DMA TX request. It is 'self-clearing'. It is not necessary to clear this bit.
1	W	0x0	RFIFOR RCVR FIFO Reset This resets the control portion of the receive FIFO and treats the FIFO as empty. This also de-asserts the DMA RX request. It is 'self-clearing'. It is not necessary to clear this bit.

Offset: 0x0008			Register Name: UART_FCR
Bit	Read/Write	Default/Hex	Description
0	W	0x0	FIFOE Enable FIFOs This enables/disables the transmit (XMIT) and receive (RCVR) FIFOs. Whenever the value of this bit is changed both the XMIT and RCVR controller portion of FIFOs is reset.

15.8.7.8 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 (USR[0] is zero) and always readable. This bit is used to enable reading and writing of the Divisor Latch register (DLL and DLH) to set the baud rate of the UART. This bit must be cleared after initial baud rate setup in order to access other registers. 0: Select RX Buffer Register (RBR) / TX Holding Register(THR) and Interrupt Enable Register (IER) 1: Select Divisor Latch LS Register (DLL) and Divisor Latch MS Register (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 one the serial output is forced to the spacing (logic 0) state. When not in Loopback Mode, as determined by MCR[4], the sout line is forced low until the Break bit is cleared. If SIR_MODE = Enabled and active (MCR[6] set to one) 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	EPS Even Parity Select It is writable only when UART is not busy (USR[0] is zero) and always writable readable. This is used to select between even and odd parity, when parity is enabled (PEN set to one). Setting the LCR[5] is unset to reverse the LCR[4]. 00: Odd Parity 01: Even Parity
3	R/W	0x0	PEN Parity Enable It is writable only when UART is not busy (USR[0] is zero) and always readable. This bit is used to enable and disable parity generation and detection in transmitted and received serial character respectively. 0: Parity disabled 1: Parity enabled Note: When use RS485 mode, PEN(LCR[3]) must set to 1.
2	R/W	0x0	STOP Number of stop bits It is writable only when UART is not busy (USR[0] is zero) and always readable. This is used to select the number of stop bits per character that the peripheral transmits and receives. If set to zero, one stop bit is transmitted in the serial data. If set to one and the data bits are set to 5 (LCR[1:0] set to zero) one and a half stop bits 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. 0: 1 stop bit 1: 1.5 stop bits when DLS (LCR[1:0]) is zero, else 2 stop bit

Offset: 0x000C			Register Name: UART_LCR
Bit	Read/Write	Default/Hex	Description
1:0	R/W	0x0	DLS Data Length Select It is writable only when UART is not busy (USR[0] is zero) and always readable. This is used to select the number of data bits per character that the peripheral transmits and receives. The number of bit that may be selected areas follows: 00: 5 bits 01: 6 bits 10: 7 bits 11: 8 bits Note: When use RS485 mode, DLS must be set to 11.

15.8.7.9 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	UART_FUNCTION Select IrDA or RS485 0: UART Mode 1: IrDA SIR Mode 2: RS485 Mode 3: Reversed
5	R/W	0x0	AFCE Auto Flow Control Enable When FIFOs are enabled and the Auto Flow Control Enable (AFCE) bit is set, Auto Flow Control features are enabled. 0: Auto Flow Control Mode is disabled 1: Auto Flow Control Mode is enabled

Offset: 0x0010			Register Name: UART_MCR
Bit	Read/Write	Default/Hex	Description
4	R/W	0x0	LOOP Loop Back Mode 0: Normal Mode 1: Loop Back Mode 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, MCR[6] set to zero), 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, MCR[6] set to one), data on the sir_out_n line is held low, while serial data output is inverted and looped back to the sir_in line.
3:2	/	/	/
1	R/W	0x0	RTS Request to Send This is used to directly control the Request to Send (rts_n) output. The Request To Send (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 (MCR[5] set to zero), the rts_n signal is set low by programming MCR[1] (RTS) to a high. In Auto Flow Control, AFCE_MODE == Enabled and active (MCR[5] set to one) and FIFOs enable (FCR[0] set to one), 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 MCR[1] is set low. 0: rts_n de-asserted (logic 1) 1: rts_n asserted (logic 0) Note that in Loopback mode (MCR[4] set to one), the rts_n output is held inactive high while the value of this location is internally looped back to an input.

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

15.8.7.10 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	FIFOERR RX Data Error in FIFO 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 a read from the LSR register provided there are no subsequent errors in the FIFO.
6	R	0x1	TEMT Transmitter Empty If the FIFOs are disabled, this bit is set to "1" whenever the TX Holding Register 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.

Offset: 0x0014			Register Name: UART_LSR
Bit	Read/Write	Default/Hex	Description
5	R	0x1	THRE TX Holding Register Empty If the FIFOs are disabled, this bit is set to "1" whenever the TX Holding Register 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.
4	R	0x0	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 whenever the serial input, <i>sir_in</i>, is held in a logic '0' state for longer than the sum of <i>start time</i>+ <i>data bits</i> + <i>parity</i>+ <i>stop bits</i>. If in infrared mode (SIR_MODE == Enabled), it is set whenever the serial input, <i>sir_in</i>, is continuously pulsed to logic '0' for longer than the sum of <i>start time</i>+ <i>data bits</i>+ <i>parity</i>+ <i>stop bits</i>. A break condition on serial input causes one and only one character, consisting of all zeros, to be received by the UART. 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 LSR clears the BI bit. In the non-FIFO mode, the BI indication occurs immediately and persists until the LSR is read.

Offset: 0x0014			Register Name: UART_LSR
Bit	Read/Write	Default/Hex	Description
3	RC	0x0	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 (LSR[3]) is set if a break interrupt has occurred, as indicated by Break Interrupt (BI) bit (LSR[4]). 0: No framing error 1: Framing error Reading the LSR clears the FE bit.
2	RC	0x0	PE Parity Error This is used to indicate the occurrence of a parity error in the receiver if the Parity Enable (PEN) bit (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 (LSR[2]) is set if a break interrupt has occurred, as indicated by Break Interrupt (BI) bit (LSR[4]). 0: No parity error 1: Parity error Reading the LSR or FIFO clears the PE bit.

Offset: 0x0014			Register Name: UART_LSR
Bit	Read/Write	Default/Hex	Description
1	RC	0x0	OE Overrun Error 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 RBR. When this happens, the data in the 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. 0: No overrun error 1: Overrun error Reading the LSR clears the OE bit.
0	R	0x0	DR Data Ready This is used to indicate that the receiver contains at least one character in the RBR or the receiver FIFO. 0: No data ready 1: Data ready This bit is cleared when the RBR is read in non-FIFO mode, or when the receiver FIFO is empty, in FIFO mode.

15.8.7.11 0x0018 UART Modem Status Register (Default Value: 0x0000_0000)

Offset: 0x0018			Register Name: UART_MSR
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R	0x0	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)

Offset: 0x0018			Register Name: UART_MSR
Bit	Read/Write	Default/Hex	Description
6	R	0x0	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 the modem or data set. 0: ri_n input is de-asserted (logic 1) 1: ri_n input is asserted (logic 0)
5	R	0x0	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 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 communications with UART. 0: dsr_n input is de-asserted (logic 1) 1: dsr_n input is asserted (logic 0) In Loopback Mode (MCR[4] set to one), DSR is the same as MCR[0] (DTR).
4	R	0x0	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 (MCR[4] = 1), CTS is the same as MCR[1] (RTS).

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

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

15.8.7.12 0x001C UART Scratch Register (Default Value: 0x0000_0000)

Offset: 0x001C			Register Name: UART_SCH
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:1	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.
0	R/W	0x0	RS485_AUTO_RTS_EN RS485_AUTO_RTS_ENABLE 1: Enable 0: Disable

15.8.7.13 0x007C UART Status Register (Default Value: 0x0000_0006)

Offset: 0x007C			Register Name: UART_USR
Bit	Read/Write	Default/Hex	Description
31:5	/	/	/

Offset: 0x007C			Register Name: UART_USR
Bit	Read/Write	Default/Hex	Description
4	R	0x0	RFF Receive FIFO Full This is used to indicate that the receive FIFO is completely full. 0: Receive FIFO not full 1: Receive FIFO Full This bit is cleared when the RX FIFO is no longer full.
3	R	0x0	RFNE Receive FIFO Not Empty This is used to indicate that the receive FIFO contains one or more entries. 0: Receive FIFO is empty 1: Receive FIFO is not empty This bit is cleared when the RX FIFO is empty.
2	R	0x1	TFE Transmit FIFO Empty This is used to indicate that the transmit FIFO is completely empty. 0: Transmit FIFO is not empty 1: Transmit FIFO is empty This bit is cleared when the TX FIFO is no longer empty.
1	R	0x1	TFNF Transmit FIFO Not Full This is used to indicate that the transmit FIFO is not full. 0: Transmit FIFO is full 1: Transmit 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

15.8.7.14 0x0080 UART Transmit FIFO Level Register (Default Value: 0x0000_0000)

Offset: 0x0080			Register Name: UART_TFL
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/

Offset: 0x0080			Register Name: UART_TFL
Bit	Read/Write	Default/Hex	Description
8:0	R	0x0	TFL Transmit FIFO Level This indicates the number of data entries in the transmit FIFO.

15.8.7.15 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 Receive FIFO Level This indicates the number of data entries in the receive FIFO.

15.8.7.16 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

15.8.7.17 0x008C UART DMA Request Enable Register (Default Value: 0x0000_0003)

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

15.8.7.18 0x00A0 UART FIFO Control Register1 (Default Value: 0x0001_0000)

Offset: 0x00A0			Register Name: UART_FCR_1
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28	R/W	0x0	TRIG_SEL 0: Use RT/TFT 1: Use RT_NEW/TFT_NEW The bit width of RT_NEW and TFT_NEW is 12 bits. When the FIFO depth is greater than 4095, RT_NEW and TFT_NEW cannot completely express all thresholds.
27:16	R/W	0x1	RT_NEW RCVR Trigger This is used to select the trigger level in the receiver FIFO at which the Received Data Available Interrupt is generated. In auto flow 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.
15:12	/	/	/
11:0	R/W	0x0	TFT_NEW TX Empty Trigger This is used to select the empty threshold level at which the THRE Interrupts are generated when the mode is active. It also determines when the dma_tx_req_n signal is asserted when in certain modes of operation.

15.8.7.19 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	DMA_PTE_RX The sending of RX_DRQ. In DMA1 mode, when RFL is more than or equal to trig or receive timeout, send DRQ. In DMA0 mode, when DMA_PTE_RX = 1 and FIFO on, if RFL is more than or equal to trig, send DRQ, else DRQ is cleared. In other cases, once the receive data is valid, send DRQ.

Offset: 0x00A4			Register Name: UART_HALT
Bit	Read/Write	Default/Hex	Description
6	R/W	0x0	PTE The sending of TX_REQ. In DMA1 mode (FIFO on), if PTE is set 1, when TFL is less than or equal to trig, send the DMA request. If PTE is set 0, when FIFO is empty, send the DMA request. The DMA request will stop when FIFO is full. In DMA0 mode, if PTE is set 1 and FIFO on, when TFL is less than or equal to trig, send DMA request. If PTE is set 1 and FIFO off, when THR is empty, send DMA request. If PTE is set 0, when FIFO(FIFO Enable) or THR(FIFO Enable) is empty, send DMA request. Otherwise, DMA request is cleared.
5	R/W	0x0	SIR_RX_INVERT SIR Receiver Pulse Polarity Invert 0: Not invert receiver signal 1: Invert receiver signal
4	R/W	0x0	SIR_TX_INVERT SIR Transmit Pulse Polarity Invert 0: Not invert transmit pulse 1: Invert transmit pulse
3	/	/	/
2	R/WAC	0x0	CHANGE_UPDATE After the user using HALT[1] to change the baudrate or LCR configuration, write 1 to update the configuration and waiting this bit self clear to 0 to finish update process. Write 0 to this bit has no effect. 1: Update trigger, Self clear to 0 when finish update.
1	R/W	0x0	CHCFG_AT_BUSY This is an enable bit for the user to change LCR register configuration and baudrate register (DLH and DLL) when the UART is busy (USB[0] is 1). 1: Enable change when busy
0	R/W	0x0	HALT_TX Halt TX This register is use to halt transmissions for testing, so that the transmit FIFO can be filled by the master when FIFOs are implemented and enabled. 0 : Halt TX disabled 1 : Halt TX enabled Note: If FIFOs are not enabled, the setting has no effect on operation.

15.8.7.20 0x00B0 UART Debug 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

15.8.7.21 0x00B4 UART Debug 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

15.8.7.22 0x00C0 UART RS485 Control Configure and Status Register (Default Value: 0x0000_0000)

Offset: 0x00C0			Register Name: UART_485_CTL
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8	R/W	0x0	AAD_BROADCAST_MODE AAD Mode Receive address detect broadcast mode 0: AAD broadcast mode 1: Not aad broadcast mode(when use this mode need set UART_485_CTL[1:0] is 01 then set UART_485_CTL[8] is 1)
7	R/W	0x0	ADD_BIT_POLARITY Address bit polarity 0: The address bit is 1 representative the data is address 1: The address bit is 0 representative the data is address
6	R/W1C	0x0	AAD_ADDR_DETECTED AAD Mode Receive address detected 0: Receive address is different with set 1: Receive address is same with set
5	R/W1C	0x0	ADDRESS_DETECTED RS485 Address detect 0: No address detected 1: Address detected

Offset: 0x00C0			Register Name: UART_485_CTL
Bit	Read/Write	Default/Hex	Description
4	R/W	0x0	DUPLEX RS485 Transfer mode 0: Full duplex 1: Half duplex
3:2	/	/	/
1:0	R/W	0x0	RCM RS485 Receive mode This is used to choose the RS485 Receive mode 00: RS_9BITM 01: AAD In RS-9bit mode, addresses and data are not distinguished. All data and addresses received by RX are written into RX FIFO. In AAD mode, addresses and data are distinguished. The data and addresses received by RX are filtered according to the AAD configuration.

15.8.7.23 0x00C4 UART RS485 Address Match Register (Default Value: 0x0000_0000)

Offset: 0x00C4			Register Name: UART_RS485_ADDR_MATCH
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0x0	AAD_RECEIVE_ADDR AAD Mode Receive address

15.8.7.24 0x00C8 UART RS485 Bus Idle Check Register (Default Value: 0x0000_0000)

Offset: 0x00C8			Register Name: BUS_IDLE_CHECK
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	BUS_IDLE_CHK_EN 1: Enable bus idle check function 0: Disable bus idle check function
6	R	0x0	BUS_STATUS The flag of bus status 1: Busy 0: Idle
5:0	R	0x0	ADJ_TIME How long the bus is idle. The unit is 8*16*Tbit

15.8.7.25 0x00CC UART TX Delay Register (Default Value: 0x0000_0000)

Offset: 0x00CC			Register Name: TX_DELAY
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0x0	DLY The delay time between the last stop bit and the next start bit. The unit is 16*Tbit. It is use to control the space between tow bytes in TX.

15.8.7.26 0x00D4 UART FIFO Test Mode Register (Default Value: 0x0000_0000)

Offset: 0x00D4			Register Name: FIFO_TM
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	FTM FIFO Test Mode 0: Normal mode 1: FIFO test mode

15.8.7.27 0x00D8 UART Read TX FIFO Register (Default Value: 0x0000_0000)

Offset: 0x00D8			Register Name: READ_TX_FIFO
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8:0	R	0x0	R_TXFIFO Read TX FIFO In FIFO test mode, reading the address could read the data from the txfifo.

15.8.7.28 0x00DC UART Write RX FIFO Register (Default Value: 0x0000_0000)

Offset: 0x00DC			Register Name: WRITE_RX_FIFO
Bit	Read/Write	Default/Hex	Description
31:11	/	/	/
10:0	W	0x0	W_RXFIFO Write RX FIFO In FIFO test mode, writing the address could write the data to the rxfifo.

15.8.7.29 0x00E0 UART Read RX FIFO Register (Default Value: 0x0000_0000)

Offset: 0x00E0			Register Name: READ_RX_FIFO
Bit	Read/Write	Default/Hex	Description
31:11	/	/	/
10:0	R	0x0	R_RXFIFO Read RX FIFO In FIFO test mode, read the address could read the data from the rxfifo.

15.8.7.30 0x00E4 UART RS485 RTSN Pull Down Delay Time Register (Default Value: 0x0000_0000)

Offset: 0x00E4			Register Name: RTSN_RST_DLY_TIME
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8	R/W	0x0	RTSN_RST_DLYT_TIME_ENABLE Rts_n reset delay time enable 0: Not enable 1: Enable
7:0	R/W	0x0	RTSN_RST_DLYT Rts_n Reset Delay Time Rts_n delay number of Tbit to reset in RS485 auto flow control.

15.8.7.31 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, R/W clocks use APB clock 1: Async mode, write clock uses APB clock, read 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

15.9 SPI

15.9.1 Overview

The Serial Peripheral Interface (SPI) is a full-duplex, synchronous, four-wire serial communication interface between a CPU and SPI-compliant external devices. It can work in master mode and slave mode.

The SPI has the following features:

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



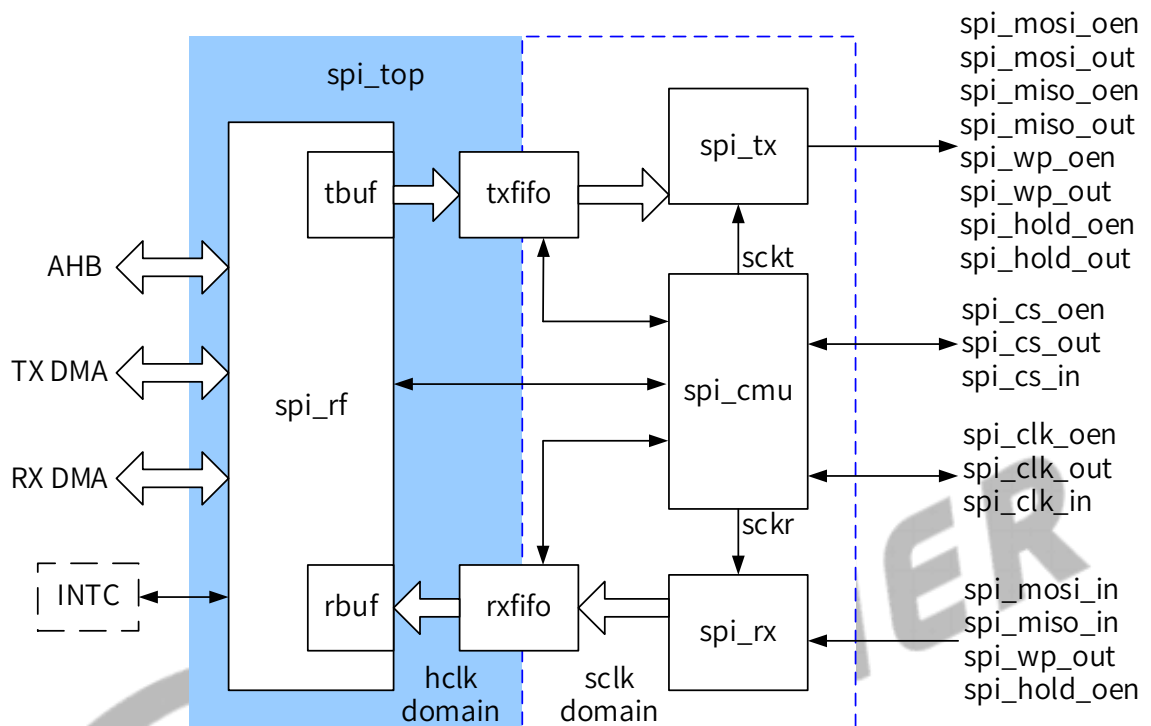
NOTE

This chapter only describes SPI0. For detailed information of SPI1 (supports SPI mode and DBI mode), please refer to Section 15.10 SPI_DBI

15.9.2 Block Diagram

The following figure shows a block diagram of the SPI.

Figure 15-49 SPI Block Diagram



SPI contains the following sub-blocks:

Table 15-26 SPI Sub-blocks

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

15.9.3 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 15.7 GPIO.

Table 15-27 SPI External Signals

Signal Name	Description	Type
SPI0-CS0	SPI0 Chip Select 0 (Active Low)	I/O
SPI0-CLK	SPI0 Clock	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 Protection (Active Low)/ Serial Data Input and Output for Quad Input or Quad Output	I/O
SPI0-HOLD	SPI0 Hold Signal/ Serial Data Input and Output for Quad Input or Quad Output	I/O

15.9.4 Functional Description

15.9.4.1 SPI Master and Slave Mode

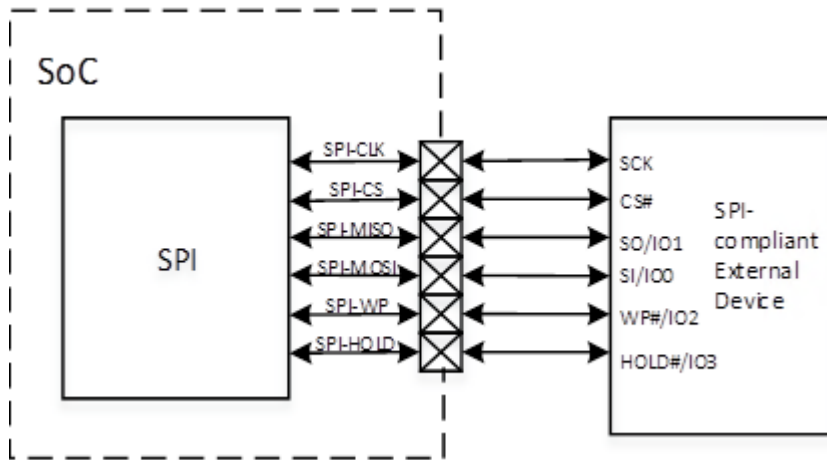
The SPI controller can be configured to a master or slave device. The master mode is selected by setting the MODE bit ([SPI_GCR](#) [1]); the slave mode is selected by clearing the MODE bit.

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

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

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

Figure 15-50 SPI Application Block Diagram



15.9.4.2 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_CLK 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_CLK). The SPI_CLK 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_CLK 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 15-28 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 waveforms for SPI_CLK.

Figure 15-51 SPI Phase 0 Timing Diagram

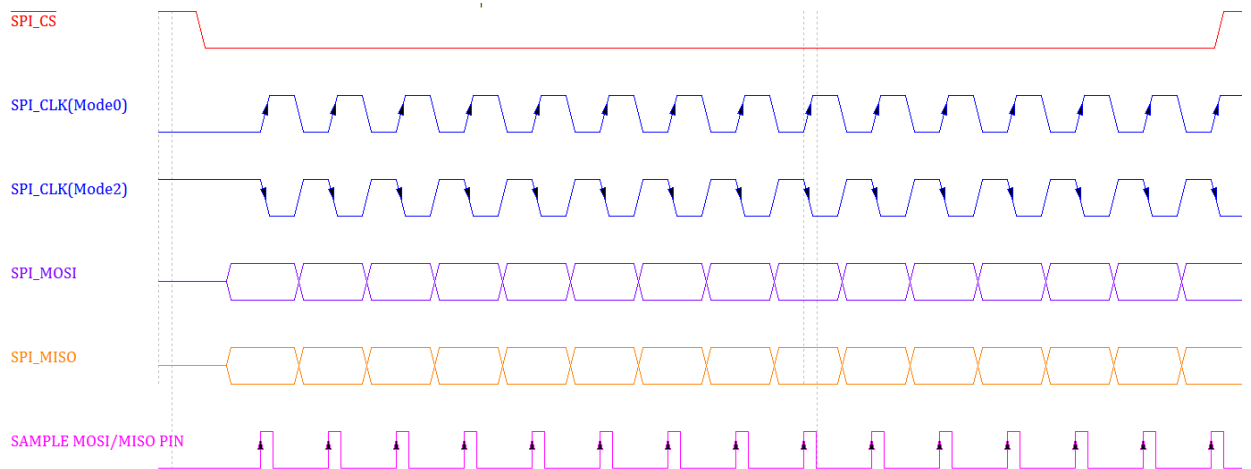
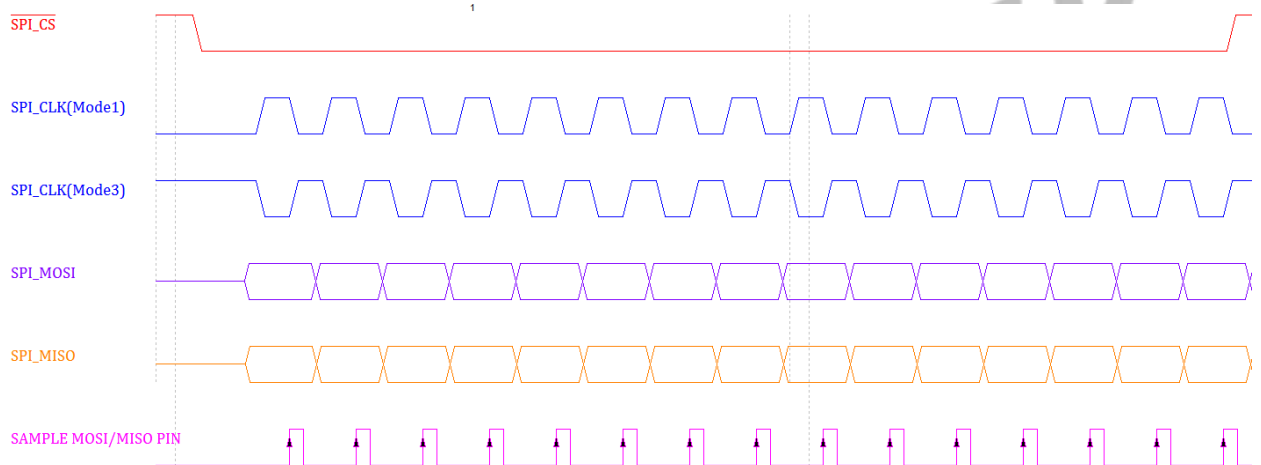


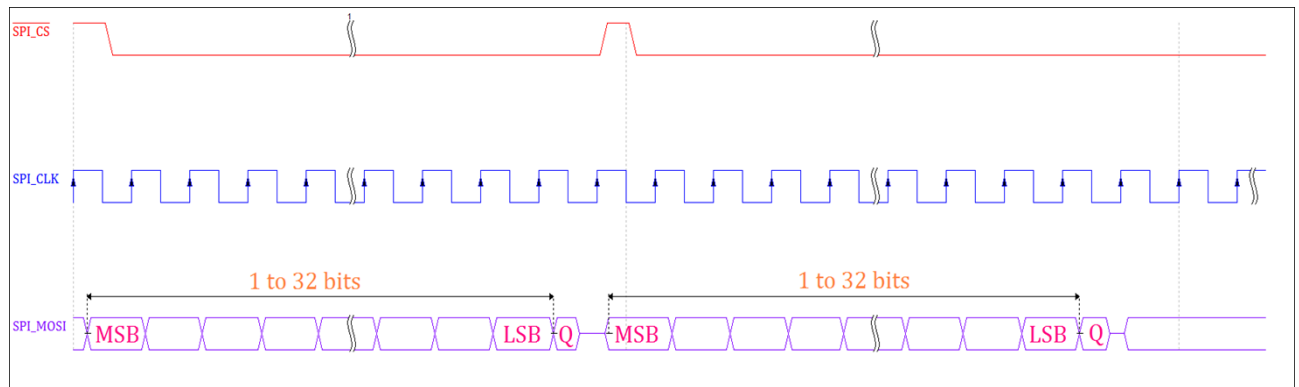
Figure 15-52 SPI Phase 1 Timing Diagram



15.9.4.3 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.

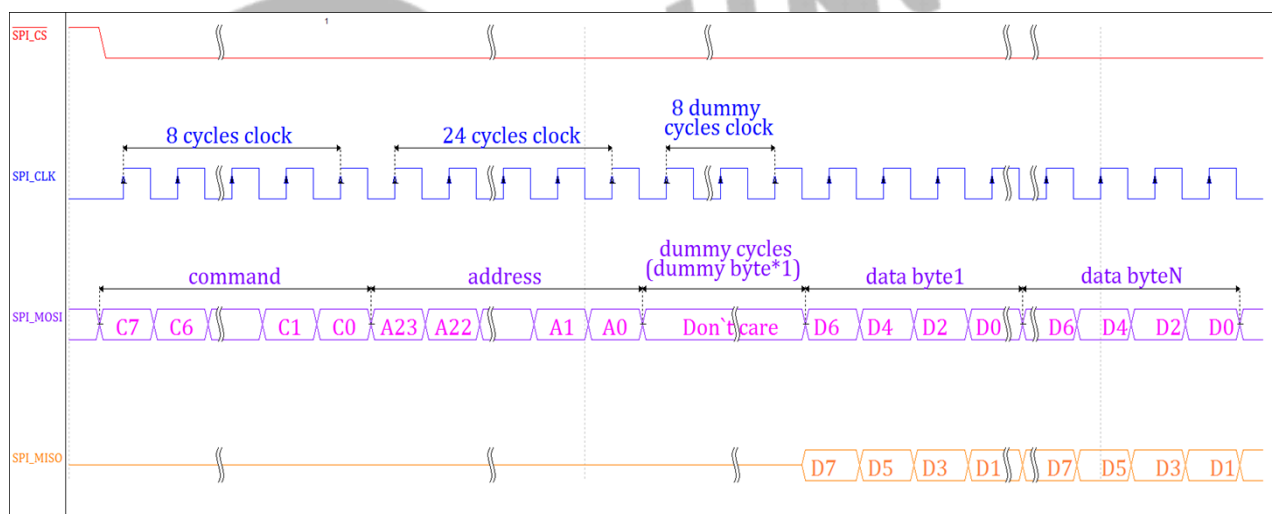
Figure 15-53 SPI 3-Wire Mode



15.9.4.4 SPI Dual Mode

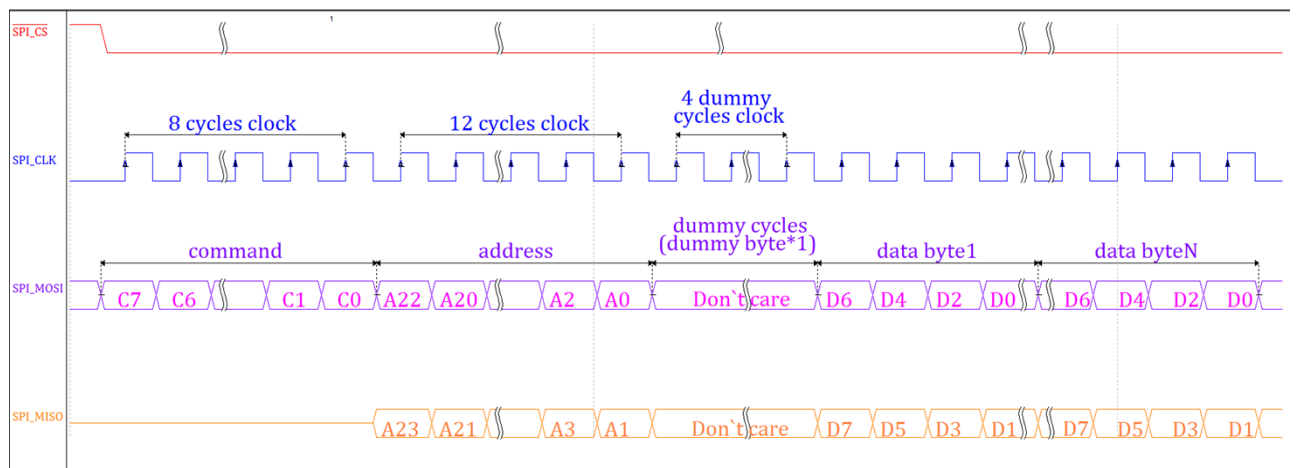
The dual read mode (SPI x2) is selected when the DRM is set in [SPI_BCC](#) (Offset: 0x0038) [28]. Using the dual mode allows data to be transferred to or from the device at double the rate of standard single mode, the data can be read at fast speed using two data bits (MOSI and MISO) at a time. The following figure describes the dual-input/dual-output SPI and the dual I/O SPI.

Figure 15-54 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 15-55 SPI Dual I/O Mode

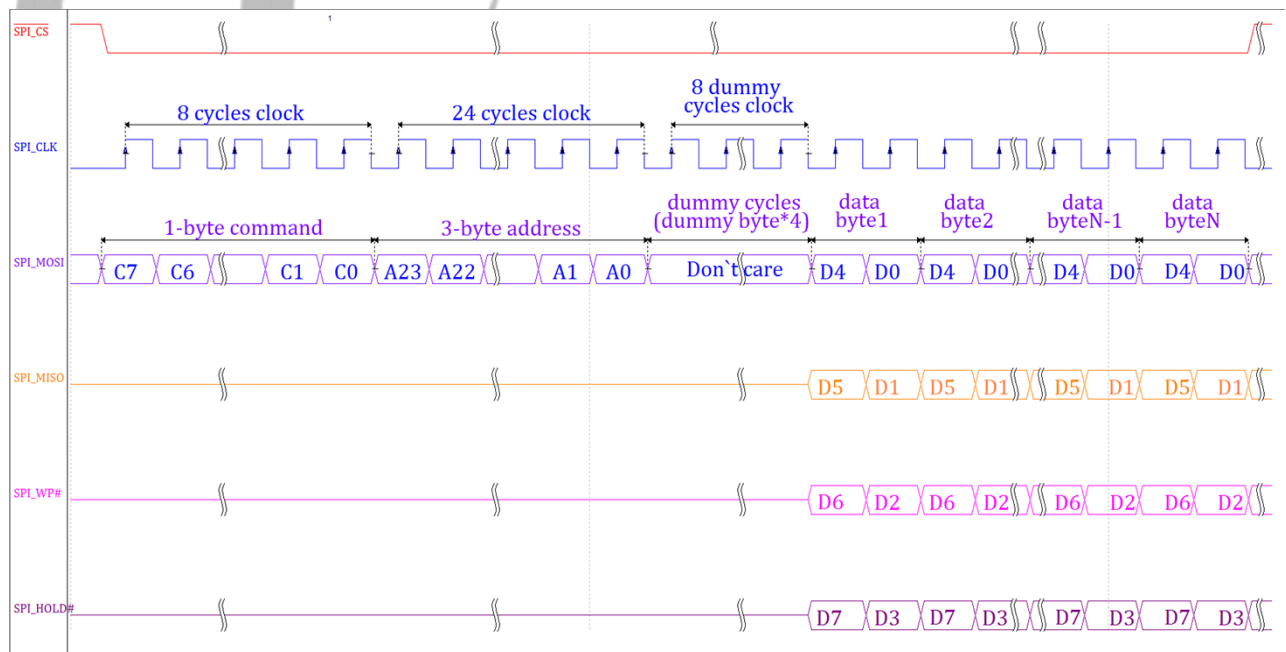


In the dual I/O SPI mode, only the command bytes output in a unit of a single bit in serial mode through the SPI_MOSI line. The address bytes and the dummy bytes output in a unit of dual bits through the SPI_MOSI and SPI_MISO. And the data bytes output (write) and input (read) in a unit of dual bits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

15.9.4.5 SPI Quad 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 15-56 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#.

15.9.4.6 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 the empty TX FIFO. 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 full TX FIFO. 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).

15.9.5 Programming Guidelines

15.9.5.1 Writing/Reading Data Process

The SPI transfers serial data between the processor and peripheral devices. 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 peripheral devices. The RX channel has the path from peripheral devices 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.

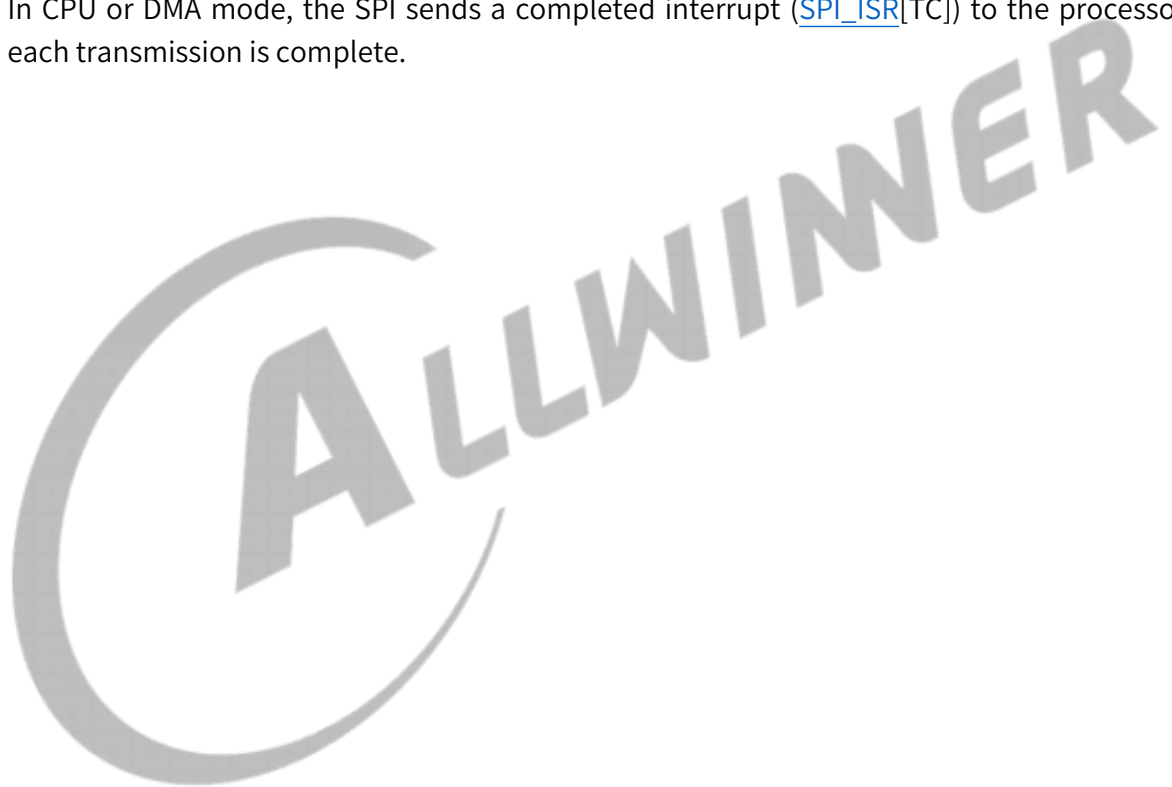
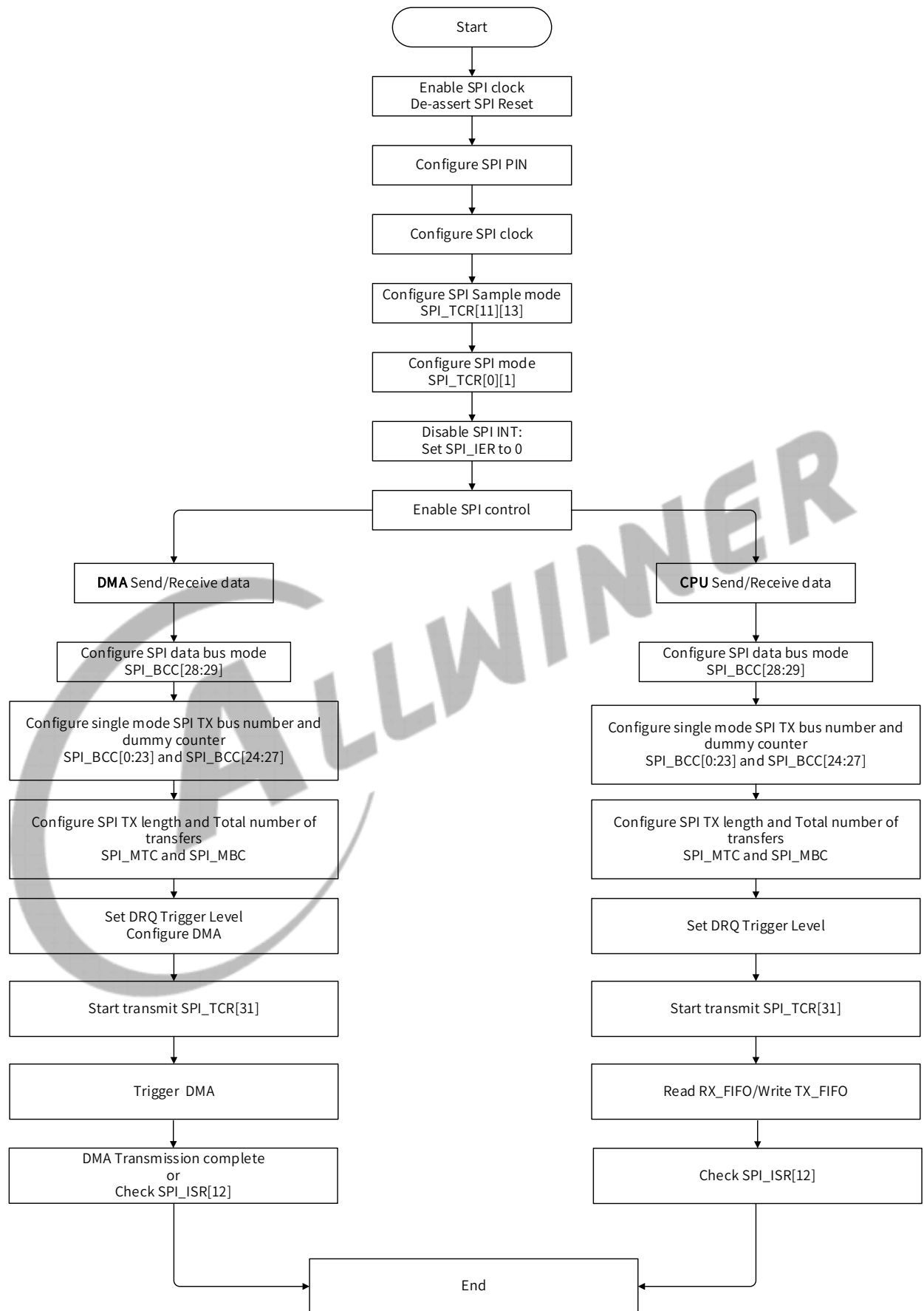


Figure 15-57 Data Transfer in Query Mode(including CPU and DMA mode)



15.9.5.2 Master Transmit and Receive Burst

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

15.9.5.3 SPI Run Clock and Sample Mode

The SPI Controller runs at 24KHz-100MHz at its interface to external SPI devices. The internal SPI Clock should run at the same frequency as the outgoing clock in master mode. The SPI Clock is selected different clock sources, and SPI must configure different work mode. There are three work mode: normal sample mode, delay half cycle sample mode, delay one cycle sample mode. Delay half cycle sample mode is the default mode of SPI controller. When SPI runs at 48MHz or below 48MHz, SPI can work at normal sample mode or delay half cycle sample mode. When SPI runs over 48MHz, set SDC in SPI Transfer Control Register to '1' to make the internal read sample point with a delay of half cycle of SPI_CLK, which is used in high-speed read operation to reduce the error caused by the time delay of SPI_CLK propagating between master and slave device. The different configuration of new/old SPI sample mode shows in the following tables.

Table 15-29 SPI Old Sample Mode

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



NOTE

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

Table 15-30 SPI New Sample Mode

SPI Sample Mode	SDM(bit13)	SDC(bit11)	SDC1(bit15)
normal sample	1	0	0

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

15.9.6 Register List

Module Name	Base Address
SPI0	0x04025000

Register Name	Offset	Description
SPI_GCR	0x0004	SPI Global Control Register
SPI_TCR	0x0008	SPI Transfer Control Register
SPI_IER	0x0010	SPI Interrupt Control Register
SPI_ISR	0x0014	SPI Interrupt Status Register
SPI_FCR	0x0018	SPI FIFO Control Register
SPI_FSR	0x001C	SPI FIFO Status Register
SPI_WCR	0x0020	SPI Wait Clock Counter Register
SPI_MBC	0x0030	SPI Master Burst Counter Register
SPI_MTC	0x0034	SPI Master Transmit Counter Register
SPI_BCC	0x0038	SPI Master Burst Control Counter 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
SPI_BSR	0x0400	SPI BUF Status Register

15.9.7 Register Description

15.9.7.1 0x0004 SPI Global Control Register (Default Value: 0x0000_0080)

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

Offset: 0x0004			Register Name: SPI_GCR
Bit	Read/Write	Default/Hex	Description
31	R/WAC	0x0	SRST Soft reset Write '1' to this bit will clear the SPI controller, and auto clear to '0' when reset operation completes Write '0' has no effect.
30:8	/	/	/
7	R/W	0x1	TP_EN Transmit Pause Enable In master mode, it is used to control transmit state machine to stop smart burst sending when RX FIFO is full. 1: Stop transmit data when RXFIFO is full 0: Normal operation. Ignore RXFIFO status Note: When XCH=1, do not write this bit.
6:3	/	/	/
2	R/W	0x0	MODE_SEL Sample Timing Mode Select 0: Old mode of Sample Timing 1: New mode of Sample Timing Note: When XCH=1, do not write this bit.
1	R/W	0x0	MODE SPI Function Mode Select 0: Slave Mode 1: Master Mode Note: When XCH=1, do not write this bit.
0	R/W	0x0	EN SPI Module Enable Control 0: Disable 1: Enable Note: After transforming from bit_mode to byte_mode, it must Enable the SPI Module again.

15.9.7.2 0x0008 SPI Transfer Control Register (Default Value: 0x0000_0087)

Offset: 0x0008			Register Name: SPI_TCR
Bit	Read/Write	Default/Hex	Description

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

Offset: 0x0008			Register Name: SPI_TCR
Bit	Read/Write	Default/Hex	Description
12	R/W	0x0	FBS First Transmit Bit Select 0: MSB first 1: LSB first Note: When XCH=1, do not write this bit.
11	R/W	0x0	SDC Master Sample Data Control 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 propagating between master and slave. 0: Normal operation, do not delay internal read sample point 1: Delay internal read sample point Note: When XCH=1, do not write this bit.
10	R/W	0x0	RPSM Rapids mode select Select Rapids mode for high speed write. 0: Normal write mode 1: Rapids write mode Note: When XCH=1, do not write this bit.
9	R/W	0x0	DDB Dummy Burst Type 0: The bit value of dummy SPI burst is zero 1: The bit value of dummy SPI burst is one Note: When XCH=1, do not write this bit.
8	R/W	0x0	DHB Discard Hash Burst In master mode it controls whether discarding unused SPI bursts 0: Receiving all SPI bursts in BC period 1: Discard unused SPI bursts, only fetching the SPI bursts during dummy burst period. The bursts number is specified by TC. Note: When XCH=1, do not write this bit.

Offset: 0x0008			Register Name: SPI_TCR
Bit	Read/Write	Default/Hex	Description
7	R/W	0x1	SS_LEVEL When control CS signal manually (SPI_CTRL_REG.SS_CTRL==1), set this bit to '1' or '0' to control the level of CS signal. 0: Set CS to low 1: Set CS to high Note: When XCH=1, do not write this bit.
6	R/W	0x0	SS_OWNER CS Output Owner Select Usually, controller sends CS signal automatically with data together. When this bit is set to 1, software must manually write SPI_CTL_REG.SS_LEVEL to 1 or 0 to control the level of CS signal. 0: SPI controller 1: Software Note: When XCH=1, do not write this bit.
5:4	R/W	0x0	SS_SEL SPI Chip Select Select one of four external SPI Master/Slave Devices 00: SPI_CS0 will be asserted 01: SPI_CS1 will be asserted (not used) 10: SPI_CS2 will be asserted (not used) 11: SPI_CS3 will be asserted (not used) Note: When XCH=1, do not write this bit.
3	R/W	0x0	SSCTL In master mode, this bit selects the output wave form for the SPI_CSx signal. Only valid when SS_OWNER = 0. 0: SPI_CSx remains asserted between SPI bursts 1: Negate SPI_CSx between SPI bursts Note: When XCH=1, do not write this bit.
2	R/W	0x1	SPOL SPI Chip Select Signal Polarity Control 0: Active high polarity (0 = Idle) 1: Active low polarity (1 = Idle) Note: When XCH=1, do not write this bit.

Offset: 0x0008			Register Name: SPI_TCR
Bit	Read/Write	Default/Hex	Description
1	R/W	0x1	CPOL SPI Clock Polarity Control 0: Active high polarity (0 = Idle) 1: Active low polarity (1 = Idle) Note: When XCH=1, do not write this bit.
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) Note: When XCH=1, do not write this bit.

15.9.7.3 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 (CSx) from valid state to 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 under run 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 under run Interrupt Enable 0: Disable 1: Enable

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

15.9.7.4 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 CS Invalid Interrupt When SSI is 1, it indicates that SS has changed from valid state to invalid state. Writing 1 to this bit clears it.

Offset: 0x0014			Register Name: SPI_ISR
Bit	Read/Write	Default/Hex	Description
12	R/W1C	0x0	TC Transfer Completed In master mode, it indicates that all bursts specified by BC has been exchanged. In other condition, when set, this bit indicates that all the data in TXFIFO has been loaded in the Shift register, and the Shift register has shifted out all the bits. Writing 1 to this bit clears it. 0: Busy 1: Transfer Completed
11	R/W1C	0x0	TF_UDF TXFIFO underrun This bit is set when if the TXFIFO is underrun. Writing 1 to this bit clears it. 0: TXFIFO is not underrun 1: TXFIFO is underrun
10	R/W1C	0x0	TF_OVF TXFIFO Overflow This bit is set when if the TXFIFO is overflow. Writing 1 to this bit clears it. 0: TXFIFO is not overflow 1: TXFIFO is overflowed
9	R/W1C	0x0	RX_UDF RXFIFO Underrun When set, this bit indicates that RXFIFO has underrun. Writing 1 to this bit clears it.
8	R/W1C	0x0	RX_OVF RXFIFO Overflow When set, this bit indicates that RXFIFO has overflowed. Writing 1 to this bit clears it. 0: RXFIFO is available. 1: RXFIFO has overflowed.
7	/	/	/
6	R/W1C	0x0	TX_FULL TXFIFO Full This bit is set when if the TXFIFO is full. Writing 1 to this bit clears it. 0: TXFIFO is not Full 1: TXFIFO is Full

Offset: 0x0014			Register Name: SPI_ISR
Bit	Read/Write	Default/Hex	Description
5	R/W1C	0x1	TX_EMP TXFIFO Empty This bit is set if the TXFIFO is empty. Writing 1 to this bit clears it. 0: TXFIFO contains one or more words. 1: TXFIFO is empty
4	R/W1C	0x1	TX_READY TXFIFO Ready 0: TX_WL > TX_TRIG_LEVEL 1: TX_WL <= TX_TRIG_LEVEL This bit is set any time if TX_WL <= TX_TRIG_LEVEL. Writing "1" to this bit clears it. Where TX_WL is the water level of RXFIFO
3	/	/	/
2	R/W1C	0x0	RX_FULL RX Full This bit is set when the RXFIFO is full. Writing 1 to this bit clears it. 0: Not Full 1: Full
1	R/W1C	0x1	RX_EMP RX Empty This bit is set when the RXFIFO is empty. Writing 1 to this bit clears it. 0: Not empty 1: Empty
0	R/W1C	0x0	RX_RDY RX Ready 0: RX_WL <= RX_TRIG_LEVEL 1: RX_WL > RX_TRIG_LEVEL This bit is set any time if RX_WL > RX_TRIG_LEVEL. Writing "1" to this bit clears it. Where RX_WL is the total water level of the RXFIFO.

15.9.7.5 0x0018 SPI FIFO Control Register (Default Value: 0x0040_0001)

Offset: 0x0018			Register Name: SPI_FCR
Bit	Read/Write	Default/Hex	Description

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

Offset: 0x0018			Register Name: SPI_FCR
Bit	Read/Write	Default/Hex	Description
8	R/W	0x0	RF_DRQ_EN RX FIFO DMA Request Enable 0: Disable 1: Enable
7:0	R/W	0x1	RX_TRIG_LEVEL RX FIFO Ready Request Trigger Level

15.9.7.6 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
30:24	/	/	/
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
14:8	/	/	/
7:0	R	0x0	RF_CNT RX FIFO Counter These bits indicate the number of words in RX FIFO + RX BUFFER 0: 0 byte in RX FIFO & RX BUFFER 1: 1 byte in RX FIFO & RX BUFFER ... 64: 64 bytes in RX FIFO & RX BUFFER ... 192:192bytes in RX FIFO & RX BUFFER other: Reserved

15.9.7.7 0x0020 SPI Wait Clock Counter Register (Default Value: 0x0000_0000)

Offset: 0x0020	Register Name: SPI_WCR

Bit	Read/Write	Default/Hex	Description
31:20	/	/	/
19:16	R/W	0x0	SWC Dual mode direction switch waits clock counter (for master mode only). 0: No wait states inserted n: n SPI_CLK wait states inserted Note: These bits control the number of wait states to be inserted before start dual data transfer in dual SPI mode. The SPI module counts SPI_CLK by SWC for delaying next word data transfer. Note: When XCH=1, do not write this bit.
15:0	R/W	0x0	WCC Wait Clock Counter (In Master mode) These bits control the number of wait states to be inserted in data transfers. The SPI module counts SPI_CLK by WCC for delaying next word data transfer. 0: No wait states inserted N: N SPI_CLK wait states inserted Note: When XCH=1, do not write this bit.

15.9.7.8 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. 0: 0 burst 1: 1 burst ... N: N bursts Note: - When XCH=1, do not write this bit. - Total transfer data, include the TXD, RXD and dummy burst.

15.9.7.9 0x0034 SPI Master Transmit Counter Register (Default Value: 0x0000_0000)

Offset: 0x0034	Register Name: SPI_MTC

Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:0	R/W	0x0	MWTC Master Write Transmit Counter In master mode, this field specifies the burst number that should be sent to TXFIFO before automatically sending dummy burst. For saving bus bandwidth, the dummy burst (all zero bits or all one bits) is sent by SPI Controller automatically. 0: 0 burst 1: 1 burst ... N: N bursts Note: When XCH=1, do not write this bit.

15.9.7.10 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	Quad_EN Master Quad_Mode_EN 0: Quad mode disable 1: Quad mode enable Note: - When XCH=1, do not write this bit. - Quad mode includes Quad-Input and Quad-Output.
28	R/W	0x0	DRM Master Dual Mode RX Enable 0: RX use single-bit mode 1: RX use dual mode Note: - When XCH=1, do not write this bit. - It is only valid when Quad_Mode_EN=0.

Offset: 0x0038			Register Name: SPI_BCC
Bit	Read/Write	Default/Hex	Description
27:24	R/W	0x0	DBC Master Dummy Burst Counter In master mode, this field specifies the burst number that should be sent before receive in dual SPI mode. The data is don't care by the device. 0: 0 burst 1: 1 burst ... N: N bursts Note: When XCH=1, do not write this bit.
23:0	R/W	0x0	STC Master Single Mode Transmit Counter In master mode, this field specifies the burst number that should be sent in single mode before automatically sending dummy burst. This is the first transmit counter in all bursts. 0: 0 burst 1: 1 burst ... N: N bursts Note: When XCH=1, do not write this bit.

15.9.7.11 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	TCE Transfer Control Enable In master mode, it is used to start to transfer the serial bit's frame, it is only valid when Work Mode Select==0x2/0x3. 0: Idle 1: Initiates transfer. Write "1" to this bit will start to transfer serial bits' frame (the value comes from the SPI TX Bit Register or SPI RX Bit Register), and will auto clear after the bursts transfer completely. Write '0' to this bit has no effect.

Offset: 0x0040			Register Name: SPI_BATCR
Bit	Read/Write	Default/Hex	Description
30	R/W	0x0	MSMS Master Sample Standard 0: Standard Sample Mode 1: Delay Sample Mode In Standard Sample Mode, SPI master samples the data at the standard rising edge of CLK for each SPI mode; In Delay Sample Mode, SPI master samples data at the edge that is half cycle delayed by the standard rising edge of CLK defined in respective SPI mode.
29:26	/	/	/
25	R/W1C	0x0	TBC Transfer Bits Completed 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. 0: Busy 1: Transfer Completed Note: It is only valid when Work Mode Select is 0x2/0x3.
24	R/W	0x0	TBC_INT_EN Transfer Bits Completed Interrupt Enable 0: Disable 1: Enable Note: It is only valid when Work Mode Select is 0x2/0x3.
23:22	/	/	/
21:16	R/W	0x00	RX_FEM_LEN Configure the length of serial data frame(burst) of RX 000000: 0bit 000001: 1bit ... 100000: 32bits Other values: Reserved Note: It is only valid when Work Mode Select==0x2/0x3. and cannot be written when TCE=1.
15:14	/	/	/

Offset: 0x0040			Register Name: SPI_BATCR
Bit	Read/Write	Default/Hex	Description
13:8	R/W	0x00	TX_FRM_LEN Configure the length of serial data frame(burst) of TX 000000: 0bit 000001: 1bit ... 100000: 32bits Other values: Reserved Note: It is only valid when Work Mode Select==0x2/0x3, and cannot be written when TCE=1.
7	R/W	0x1	SS_LEVEL When control CS signal manually, set this bit to '1' or '0' to control the level of CS signal. 0: Set CS to low 1: Set CS to high Note: It is only valid when Work Mode Select==0x2/0x3, and only work in Mode0, cannot be written when TCE=1.
6	R/W	0x0	SS_OWNER CS Output Owner Select Usually, controller sends CS signal automatically with data together. When this bit is set to 1, software must manually write SPI_CTL_REG.SS_LEVEL to 1 or 0 to control the level of CS signal. 0: SPI controller 1: Software Note: It is only valid when Work Mode Select==0x2/0x3, and only work in Mode0, cannot be written when TCE=1.
5	R/W	0x1	SPOL SPI Chip Select Signal Polarity Control 0: Active high polarity (0 = Idle) 1: Active low polarity (1 = Idle) Note: It is only valid when Work Mode Select==0x2/0x3, and only work in Mode0, cannot be written when TCE=1.
4	/	/	/

Offset: 0x0040			Register Name: SPI_BATCR
Bit	Read/Write	Default/Hex	Description
3:2	R/W	0x0	SS_SEL SPI Chip Select Select one of four external SPI Master/Slave Devices 00: SPI_CS0 will be asserted 01: SPI_CS1 will be asserted (not used) 10: SPI_CS2 will be asserted (not used) 11: SPI_CS3 will be asserted (not used) Note: It is only valid when Work Mode Select= =0x2/0x3, and only work in Mode0, cannot be written when TCE=1.
1:0	R/W	0x0	WMS 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: Reserve 10: Data frame is bit aligned in 3-Wire SPI 11: Data frame is bit aligned in Standard SPI

15.9.7.12 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	CDR_N Clock Divide Rate (Master Mode Only) The SPI_CLK is determined according to the following equation: $SPI\_CLK = Source\_CLK / (2 * (CDR\_N + 1))$. This register is only valid when Work Mode Select==0x2/0x3.

15.9.7.13 0x0048 SPI TX Bit Register (Default Value: 0x0000_0000)

Offset: 0x0048			Register Name: SPI_TBR
Bit	Read/Write	Default/Hex	Description

Offset: 0x0048			Register Name: SPI_TBR
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	VTB The Value of the Transmit Bits This register is used to store the value of the transmitted serial data frame. Note: In the process of transmission, the LSB is transmitted first. This register is only valid when Work Mode Select==0x2/0x3.

15.9.7.14 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	VRB The Value of the Receive Bits This register is used to store the value of the received serial data frame. Note: In the process of transmission, the LSB is transmitted first. This register is only valid when Work Mode Select==0x2/0x3.

15.9.7.15 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	0x3	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
5	R/W	0x1	SPI_ACK_M SPI NDMA Acknowledge Mode 0: Active fall do not care ACK 1: Active fall must after detect ACK is high

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

15.9.7.16 0x0200 SPI TX Data Register (Default Value: 0x0000_0000)

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

15.9.7.17 0x0300 SPI RX Data Register (Default Value: 0x0000_0000)

Offset: 0x0300			Register Name: SPI_RXD
Bit	Read/Write	Default/Hex	Description

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

15.9.7.18 0x0400 SPI BUF Status Register (Default Value: 0x0000_0000)

Offset: 0x0400			Register Name: SPI_BSR
Bit	Read/Write	Default/Hex	Description
31:27	/	/	/
26:24	R	0x0	TX_SHIFT_BUFFER_CNT TX SHIFT Buffer Counter These bits indicate the number of bytes in TX Shift Buffer, which is after TX FIFO, and maximum depth is 4 level with 8 bits.
23:19	/	/	/
18:16	R	0x0	TB_CNT TX FIFO Write Buffer Counter These bits indicate the number of bytes in TX FIFO Write Buffer
15:8	/	/	/
7:0	R	0x0	RB_CNT RX FIFO Read Buffer Counter These bits indicate the number of bytes in RX FIFO Read Buffer

15.10 SPI_DBI

15.10.1 Overview

The SoC provides a 3/4 line SPI display bus interface (SPI_DBI) for video data transmission. It supports DBI mode or SPI mode. The DBI mode can be configured with 3-wire/4-wire transmission and compatible with multiple video data formats at the same time. The integrated SPI mode is used for low-cost display schemes.

The SPI mode has the following features:

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

The DBI mode has the following features:

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

Supports software flexible control video frame rate



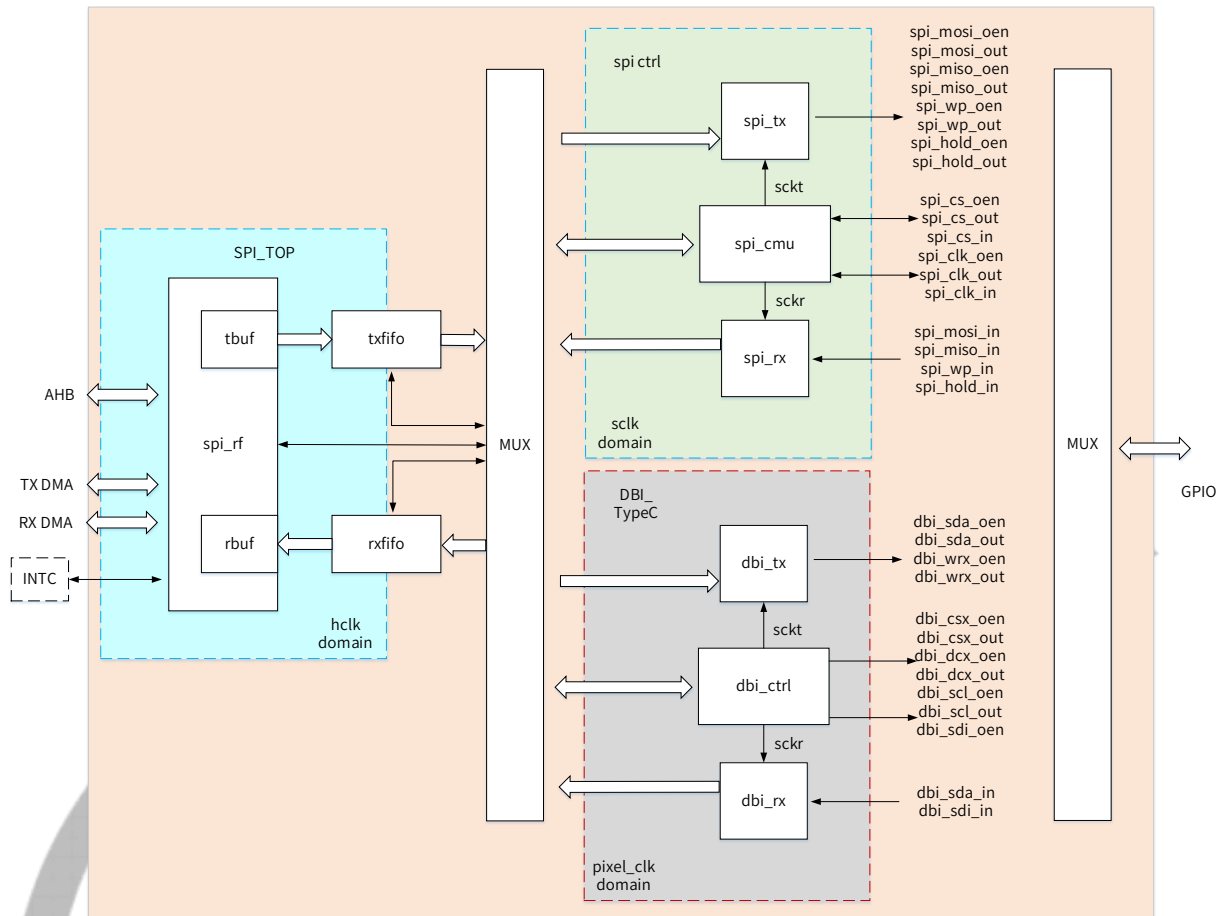
NOTE

This chapter only describes SPI1 (SPI mode and DBI mode). For detailed information of SPI0, please refer to Section 15.9 SPI.

15.10.2 Block Diagram

The following figure shows a block diagram of the SPI_DBI.

Figure 15-58 SPI_DBI Block Diagram



SPI_DBI contains the following sub-blocks:

Table 15-31 SPI_DBI Sub-blocks

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

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

15.10.3 External Signals

The following table describes the external signals of SPI_DBI. When using SPI_DBI, the corresponding PADS are selected as SPI_DBI function via Section 15.7 GPIO.

Table 15-32 GPIO multiplexing of SPI1 and DBI

DBI	SPI1
DBI-CSX	SPI1-CS0
DBI-SCLK	SPI1-CLK
DBI-SDO	SPI1-MOSI
DBI-SDI/ DBI-TE/ DBI-DCX	SPI1-MISO
DBI-DCX/DBI-WRX	SPI1-HOLD
DBI-TE	SPI1-WP

Table 15-33 SPI_DBI External Signals

Signal Name	Description	Type
SPI Mode		
SPI1-CS0	SPI1 Chip Select Signal, 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 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
DBI Mode		
DBI-CSX	Chip Select Signal, Low Active	I/O

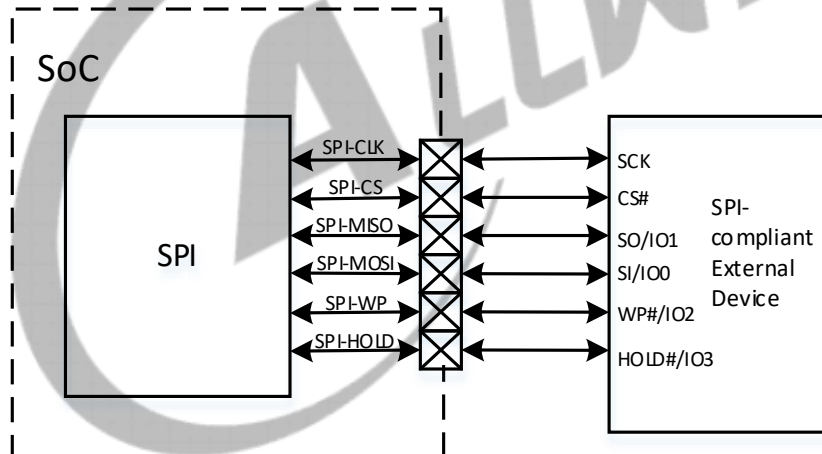
Signal Name	Description	Type
DBI-SCLK	Serial Clock Signal	I/O
DBI-SDO	Data Output Signal	I/O
DBI-SDI	Data Input Signal The data is sampled on the rising edge and the falling edge	I/O
DBI-TE	Tearing Effect Input It is used to capture the external TE signal edge. The rising and falling edge is configurable.	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

15.10.4 Functional Description

15.10.4.1 Typical Application

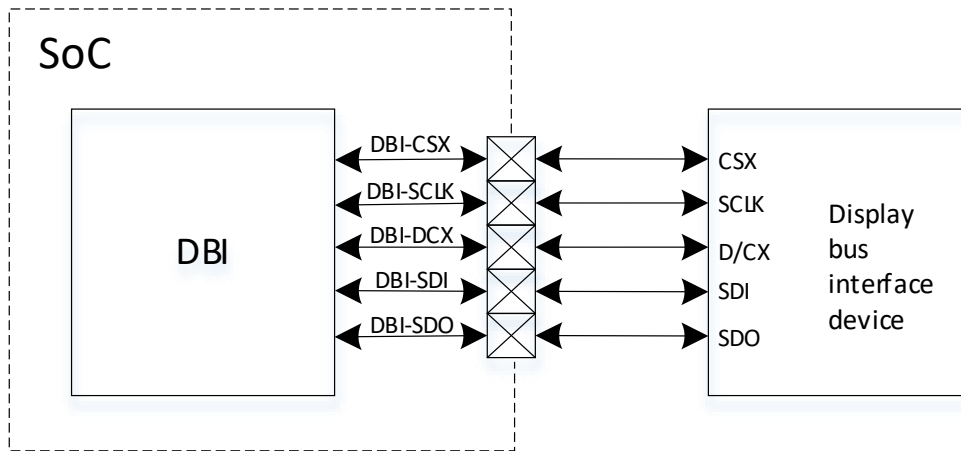
The following figure shows the application block diagram when the SPI master device is connected to a slave device.

Figure 15-59 SPI Application Block Diagram



The following figure shows the application block diagram when the DBI master device is connected to a display bus interface device.

Figure 15-60 DBI Application Block Diagram



NOTE

For the function descriptions in SPI mode, refer to the section SPI.

15.10.4.2 DBI 3-Line Interface Write and Read Timing

DBI 3-Line Interface Write Timing

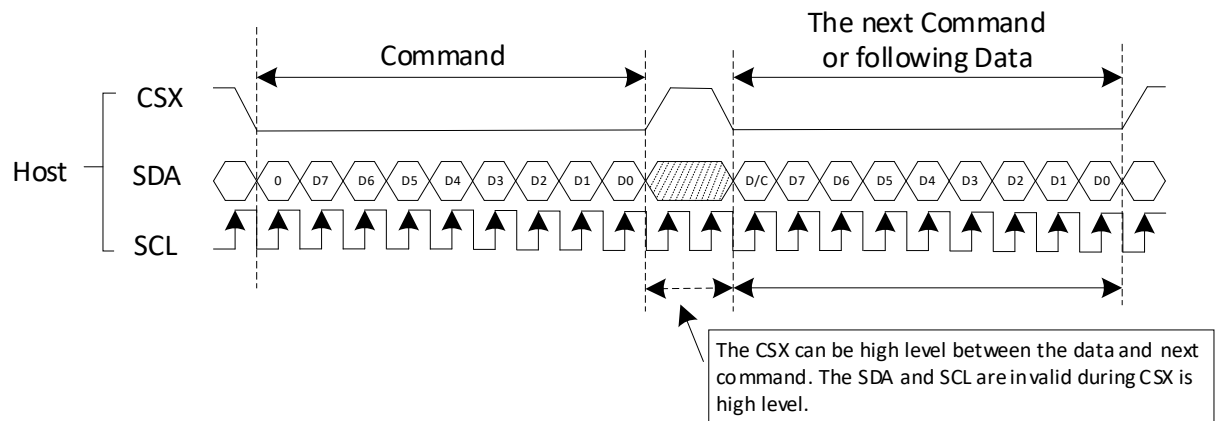
The 3-line DBI Interface I contains CSX, SDA, and SCL, where SDA shares this port for bidirectional port data input and output.

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

As there is no Data/Command data line in the 3-line display bus mode, whether Data or Command is currently being transmitted cannot be confirmed. Thus, an extra bit is added to the data-stream before MSB to indicate whether Data or Command is currently being transmitted. (0: Command, 1: Data)

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

Figure 15-61 DBI 3-Line Display Bus Serial Interface Write Operation Format

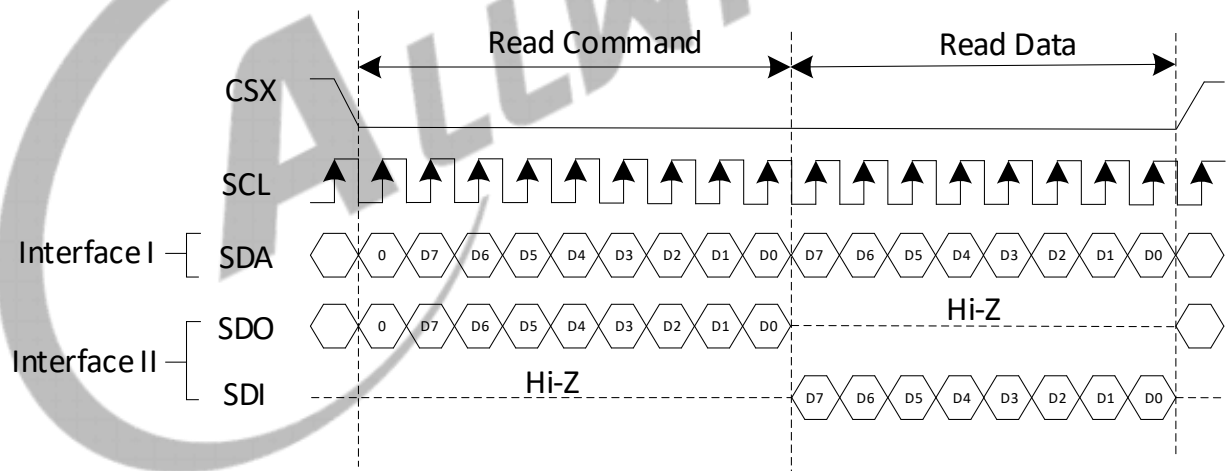


DBI 3-Line Interface Read Timing

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

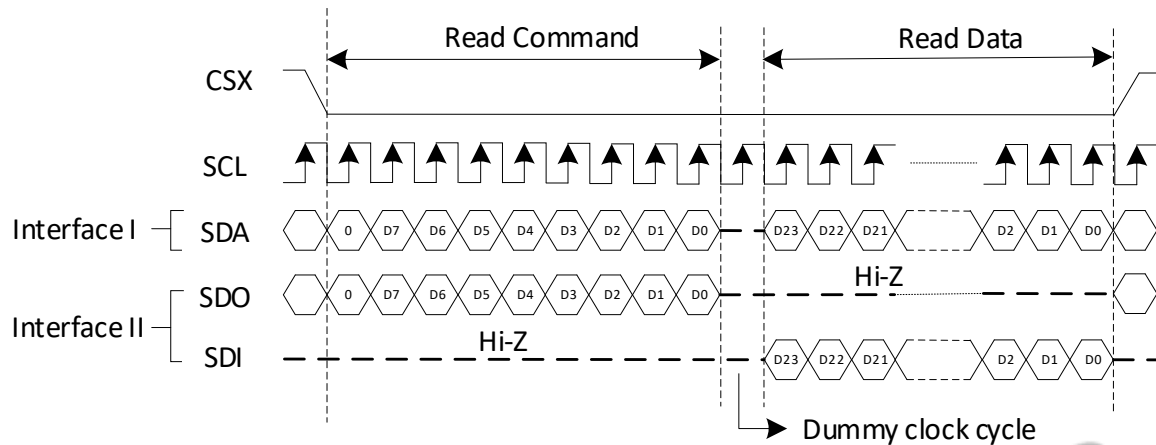
The following figure shows the 8 bits read operation format of 3-line DBI Interface I and Interface II. After the read command is transmitted, the data is read immediately with no dummy period.

Figure 15-62 DBI 3-Line Display Bus Serial Interface 8-bit Read Operation Format



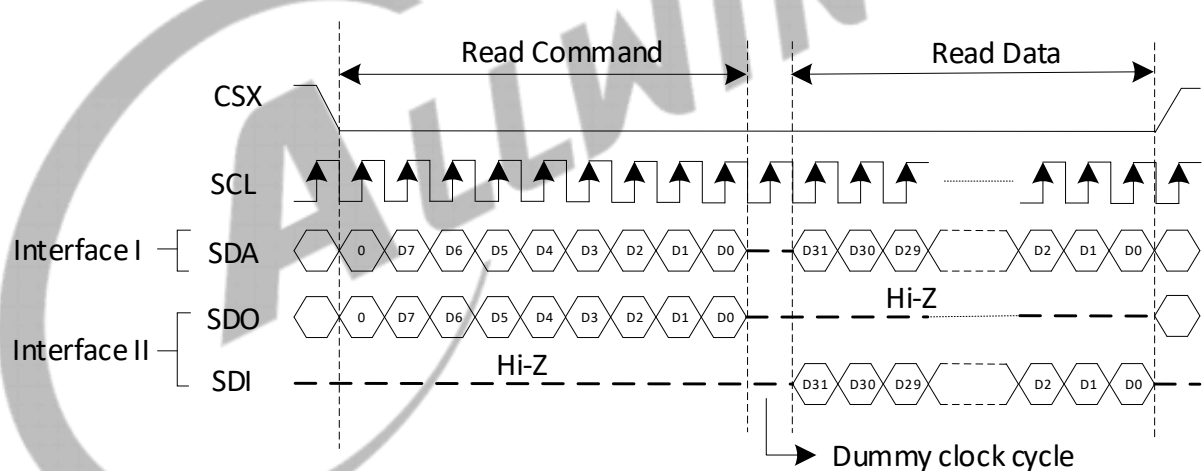
The following figure shows the 24 bits read operation format of 3-line DBI Interface I and Interface II. When the read command is transmitted, the data will be read after the dummy clock cycle.

Figure 15-63 DBI 3-Line Display Bus Serial Interface 24-bit Read Operation Format



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

Figure 15-64 DBI 3-Line Display Bus Serial Interface 32-bit Read Operation Format



15.10.4.3 DBI 4-Line Interface Write and Read Timing

DBI 4-Line Interface Write Timing

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

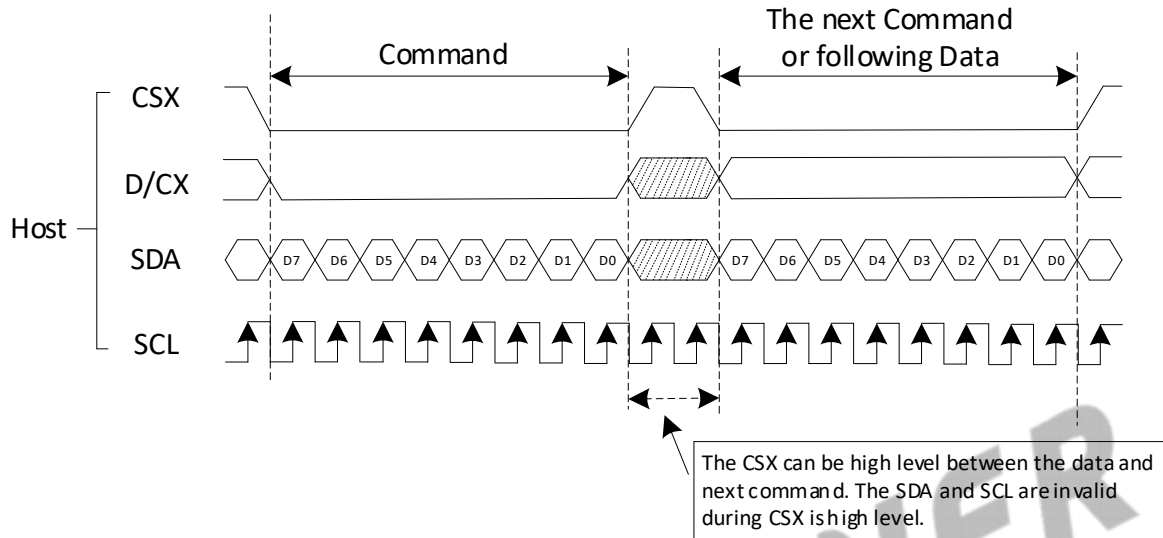
The 4-line DBI Interface II contains CSX, D/CX, SDA, SCL, and SDI; Data input and output ports are independent of each other.

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

There is a Data/Command data line in the 4-line display bus mode to indicate whether Data or Command is currently being transmitted. (0: Command, 1: Data) Thus, no extra bit is needed.

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

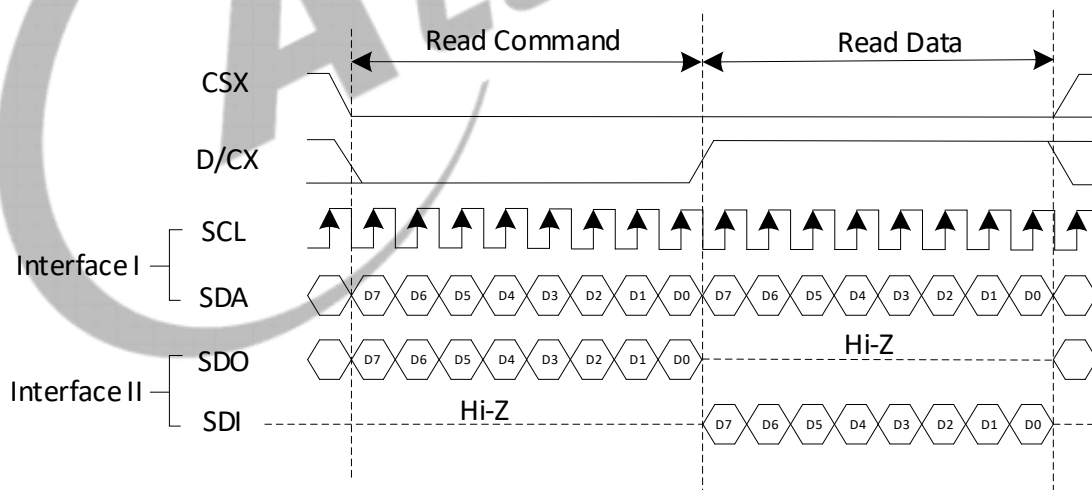
Figure 15-65 DBI 4-Line Display Bus Serial Interface Write Operation Format



DBI 4-Line Interface Read Timing

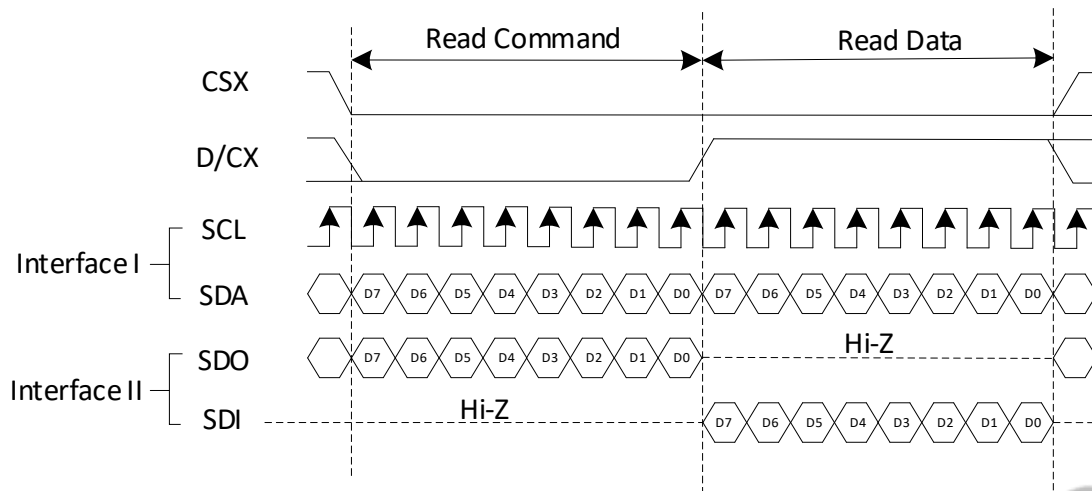
The following figure shows the 8 bits read operation format of 4-line DBI Interface I and Interface II.

Figure 15-66 DBI 4-Line Display Bus Serial Interface 8-bit Read Operation Format



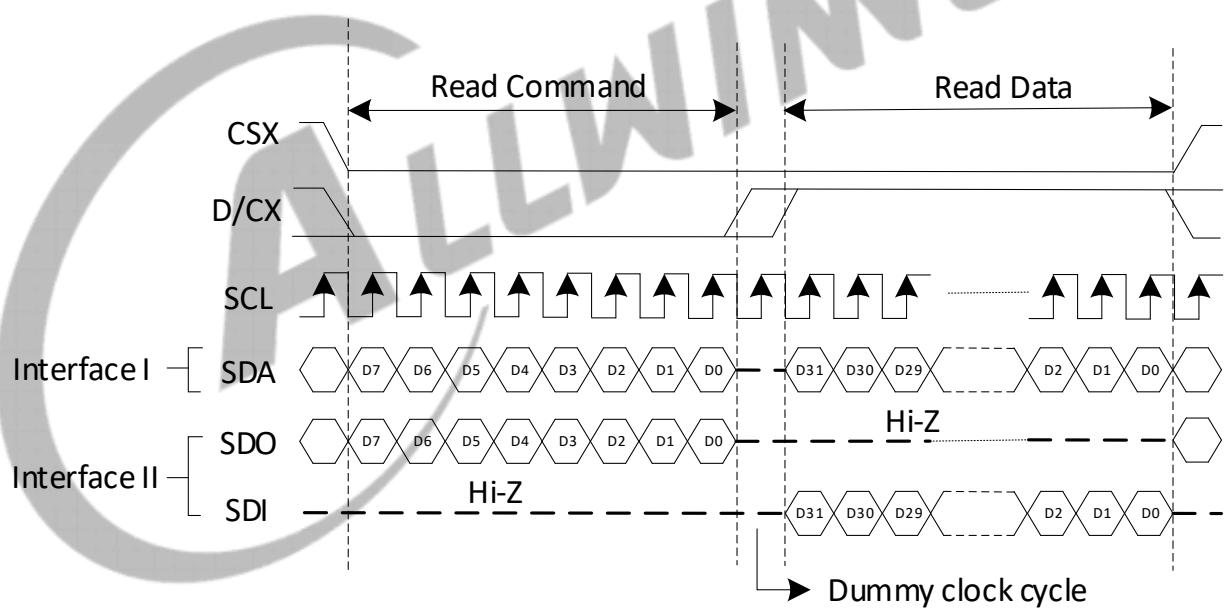
The following figure shows the 24 bits read operation format of 4-line DBI Interface I and Interface II.

Figure 15-67 DBI 4-Line Display Bus Serial Interface 24-bit Read Operation Format



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

Figure 15-68 DBI 4-Line Display Bus Serial Interface 32-bit Read Operation Format



15.10.4.4 DBI 3-Line Interface Transmit Video Format

Figure 15-69 RGB111 3-Line Interface Transmit Video Format

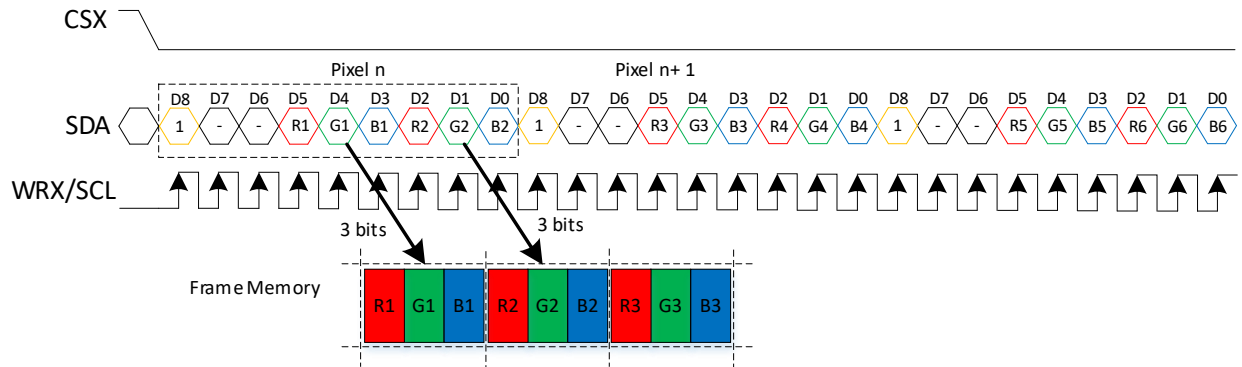


Figure 15-70 RGB444 3-Line Interface Transmit Video Format

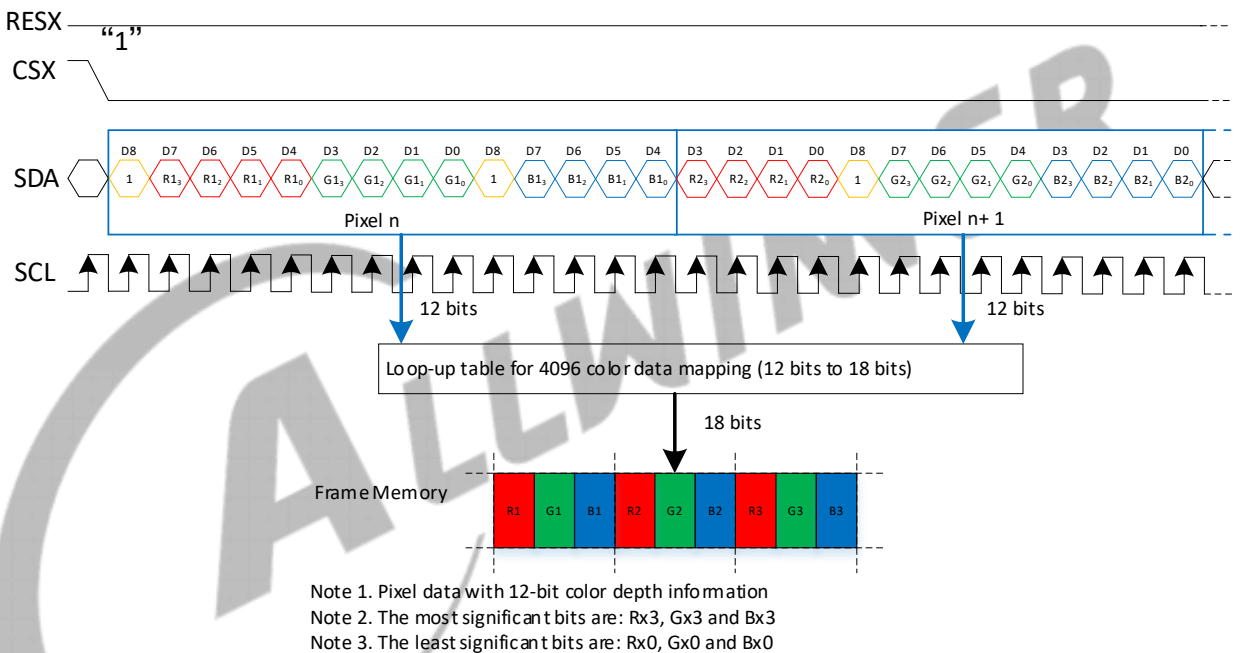


Figure 15-71 RGB565 3-Line Interface Transmit Video Format

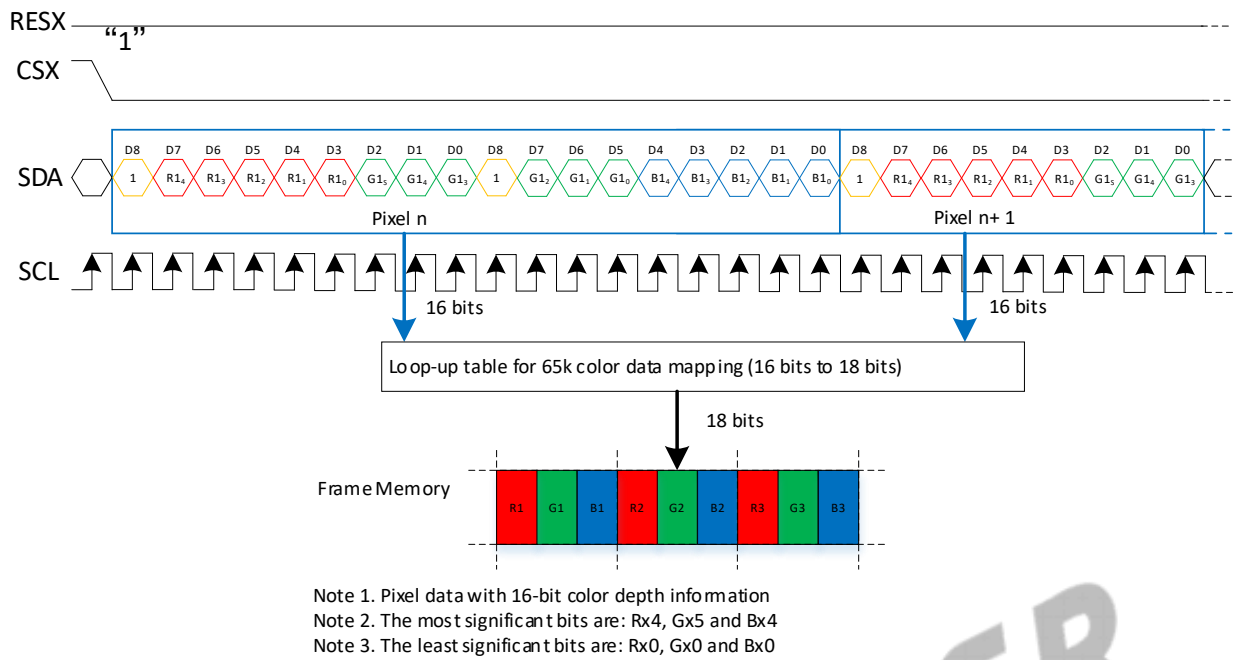
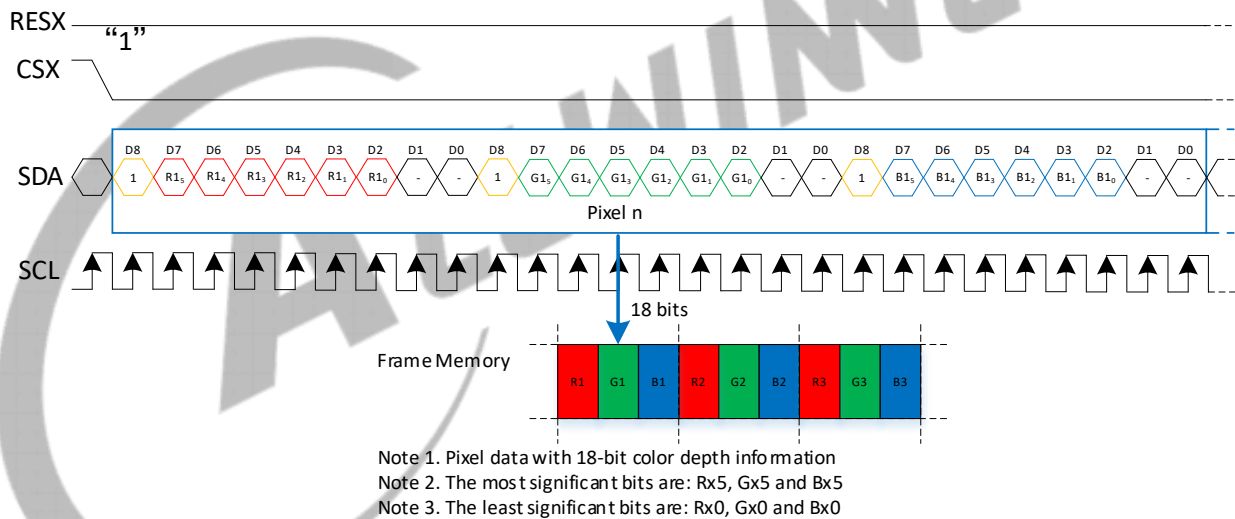


Figure 15-72 RGB666 3-Line Interface Transmit Video Format



15.10.4.5 DBI 4-Line Interface Transmit Video Format

Figure 15-73 RGB111 4-Line Interface Transmit Video Format

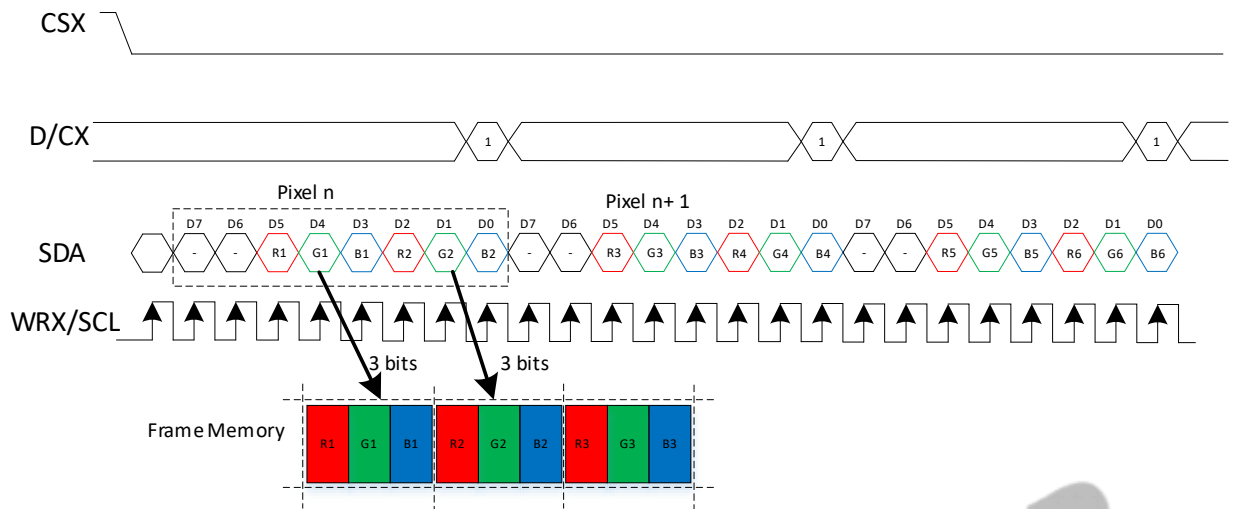
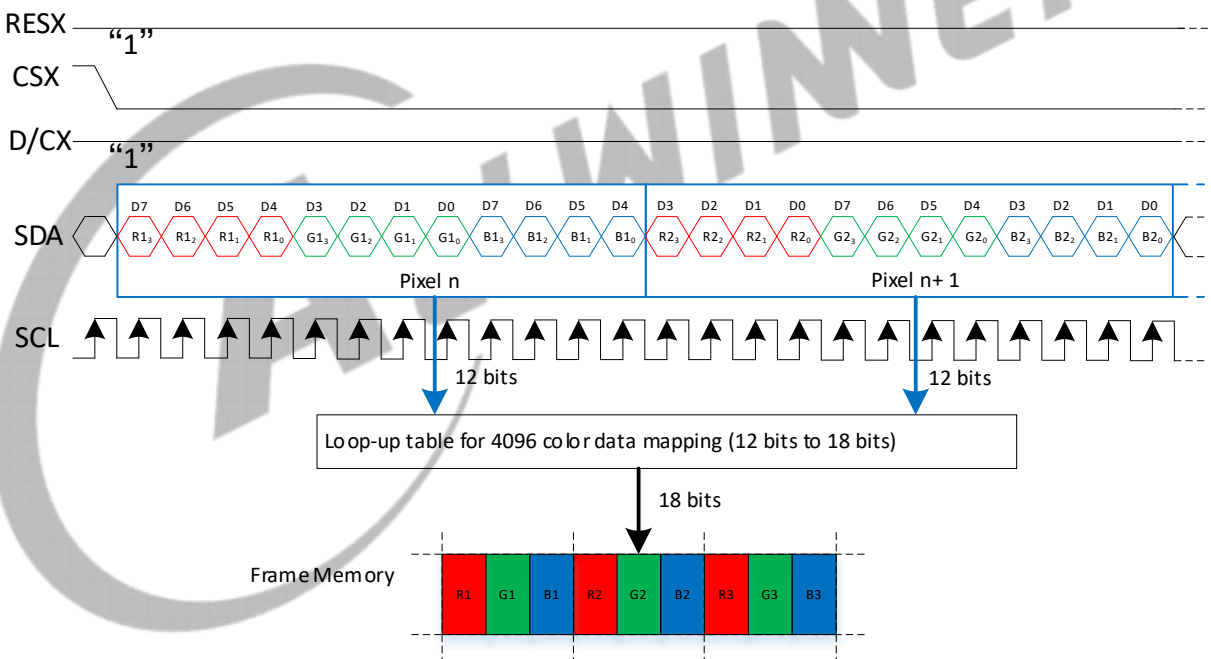


Figure 15-74 RGB444 4-Line Interface Transmit Video Format



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

RESX "1"

CSX

D/CX "1"

SDA

SCL

Pixel n

Pixel n+1

16 bits

16 bits

Loop-up table for 65k color data mapping (16 bits to 18 bits)

18 bits

Frame Memory

R1 G1 B1 R2 G2 B2 R3 G3 B3

Note 1. Pixel data with 16-bit color depth information

Note 2. The most significant t bits are: Rx4, Gx5 and Bx4

Note 3. The least significant t bits are: Rx0, Gx0 and Bx0

The diagram illustrates the timing of the SDA interface. It shows four signals: RESX, CSX, D/CX, and SDA. RESX, CSX, and D/CX are all high (labeled "1"). The SDA signal is a serial data bus. A sequence of 18 bits is transferred, labeled "Pixel n". The bits are grouped into three sets of six bits each, representing the color channels: Red (R), Green (G), and Blue (B). The SCL signal is a clock signal with 18 pulses, each corresponding to a bit on the SDA line. A blue arrow points from the 18th bit of the SDA signal to the "Frame Memory" block, which shows the data being stored: R1, G1, B1, R2, G2, B2, R3, G3, B3. The Frame Memory is divided into three columns, each containing three rows of data.

RESX "1"

CSX

D/CX "1"

SDA

SCL

Pixel n

18 bits

Frame Memory

R1 G1 B1 R2 G2 B2 R3 G3 B3

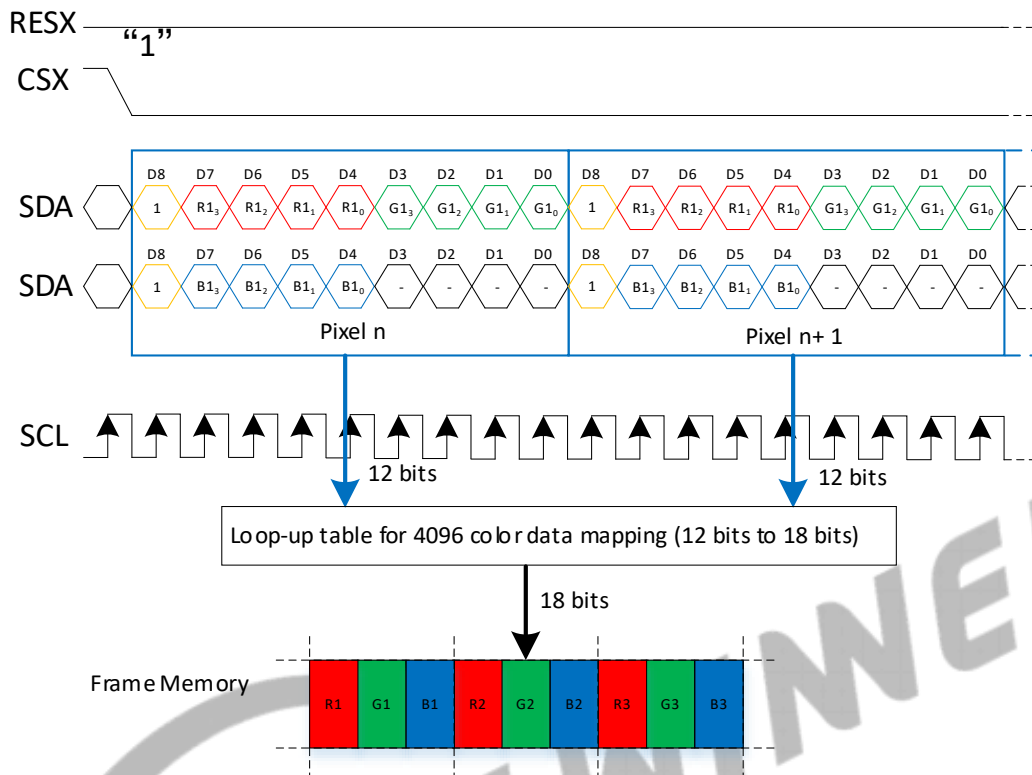
Note 1. Pixel data with 18-bit color depth information

Note 2. The most significant bits are: Rx5, Gx5 and Bx5

Note 3. The least significant bits are: Rx0, Gx0 and Bx0

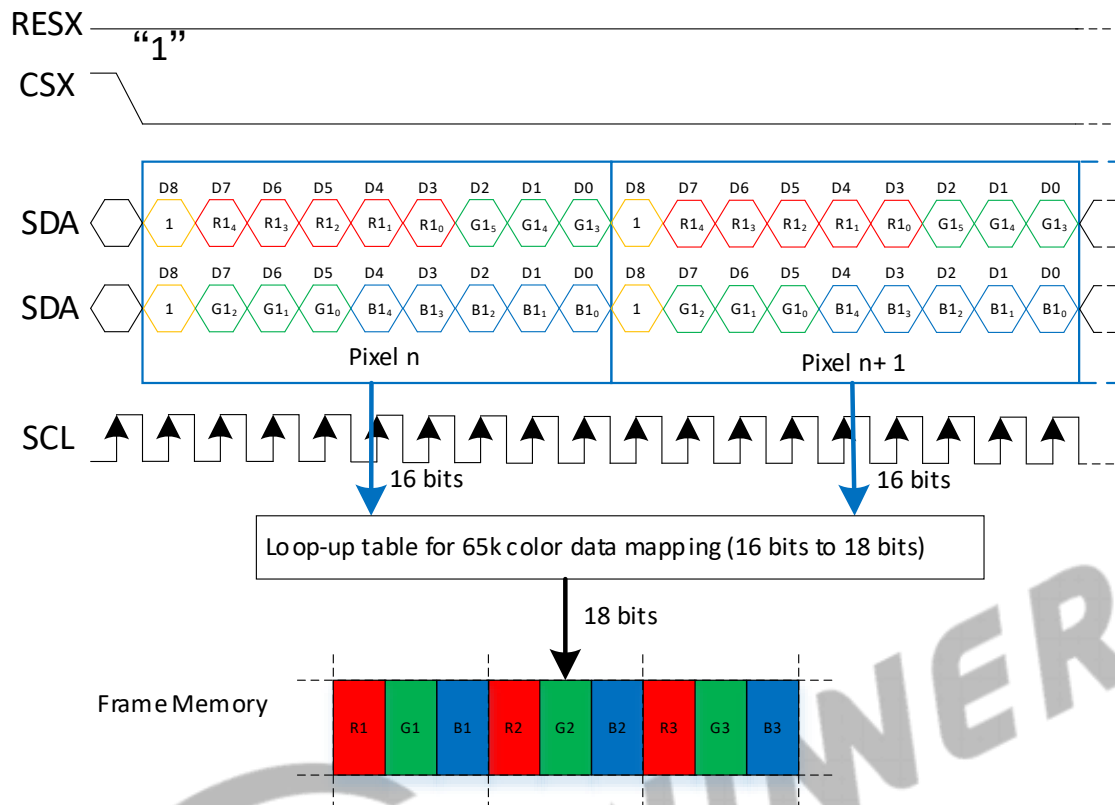
15.10.4.6 DBI 2 Data Lane Interface Transmit Video Format

Figure 15-77 RGB444 2 Data Lane Interface Transmit Video Format



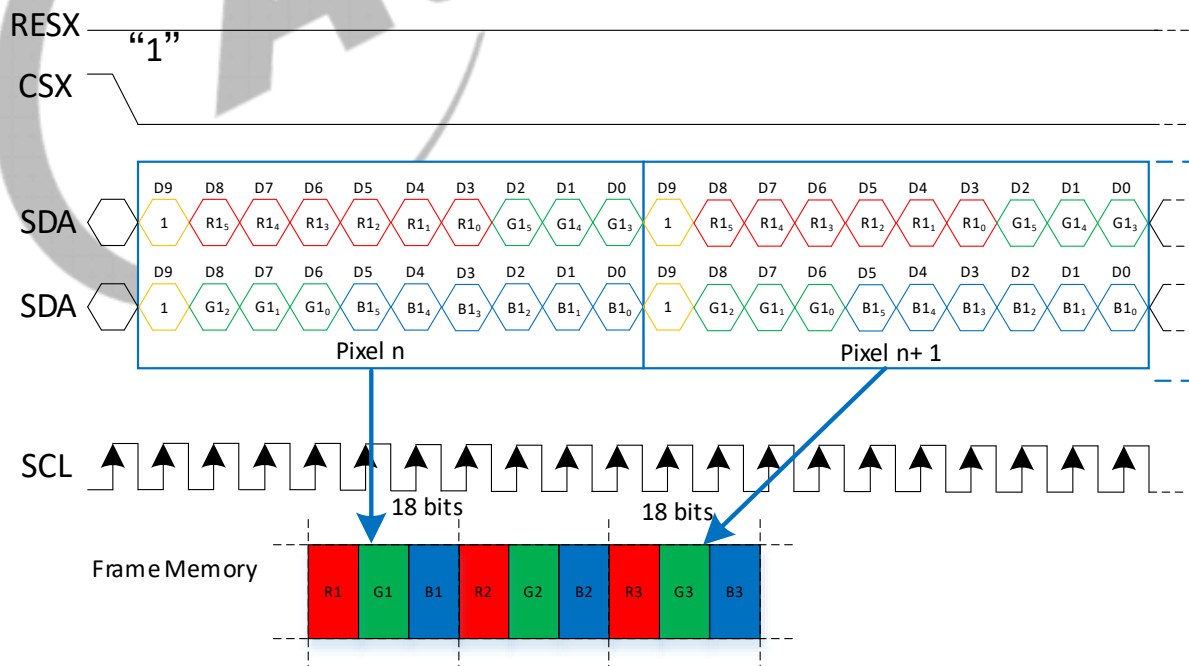
Note 1. Pixel data with 12-bit color information
 Note 2. The most significant bits are: Rx3, Gx3 and Bx3
 Note 3. The least significant bits are: Rx0, Gx0 and Bx0

Figure 15-78 RGB565 2 Data Lane Interface Transmit Video Format



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

Figure 15-79 RGB666 2 Data Lane Interface Transmit Video Format 0



- Note 1. Pixel data with 18-bit color information
- Note 2. The most significant bits are: Rx5, Gx5 and Bx5
- Note 3. The least significant bits are: Rx0, Gx0 and Bx0

Figure 15-80 RGB666 2 Data Lane Interface Transmit Video Format 1 (ilitek)

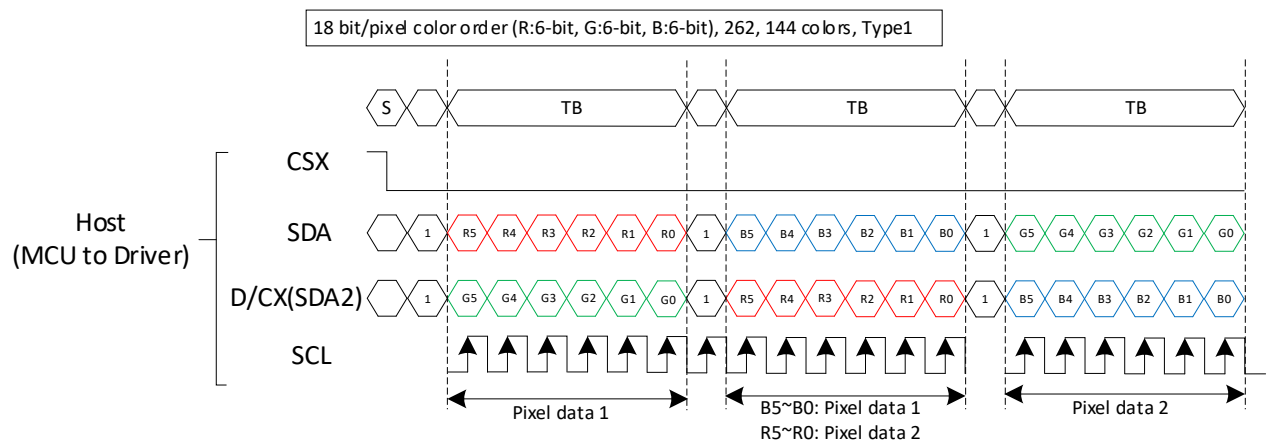


Figure 15-81 RGB666 2 Data Lane Interface Transmit Video Format 2 (New vision)

RGB666,mdt=01

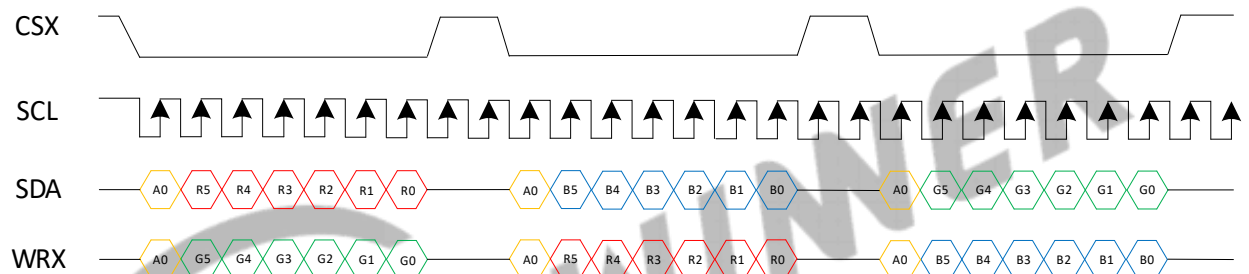
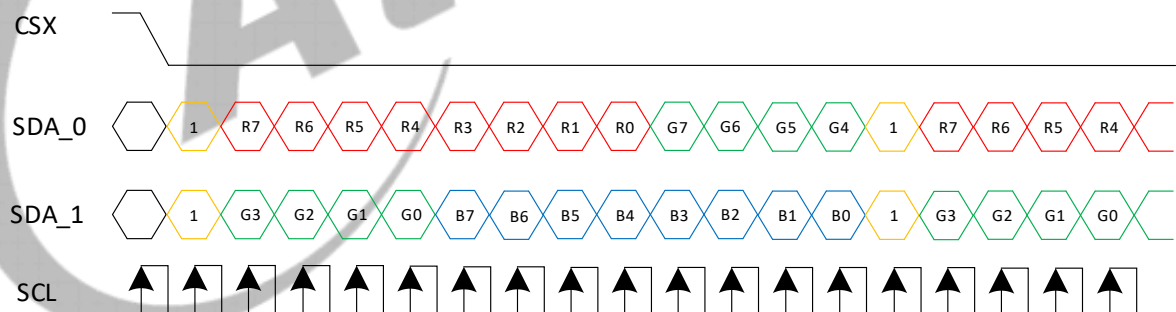


Figure 15-82 RGB888 2 Data Lane Interface Transmit Video Format

RGB888



Note 1. Pixel data with 24-bit color information

Note 2. The most significant bits are: R7, G7 and B7

Note 3. The least significant bits are: R0, G0 and B0

15.10.4.7 SPI_DBI Clock Frequency

The equation of SPI_DBI clock frequency is

$$f_{\text{SPI\_CLK}} = \text{frame\_rate} * \text{one\_pixel\_clock\_cycle} * \text{one\_frame\_pixel\_value}$$

The meaning of each sub-item is as follows:

- frame_rate: The frame rate at the unit of Hz.

- one_pixel_clock_cycle: The clock period cycle of transferring a pixel. The period cycle differs from different transferring format.
 - 2-data lane-RGB444: 9 cycles
 - 2-data lane-RGB565: 9 cycles
 - 2-data lane-RGB666-Format0: 10 cycles
 - 2-data lane-RGB888: 13 cycles
- one_frame_pixel_value: The number of pixel in one frame.
 - 240*320
 - 320*480

For example, RGB data is transferred with 2 data line and RGB666 mode, and the resolution and frame rate of the display are 240*320 and 60Hz, then the SPI clock frequency is $60 \times 10 \times 240 \times 320 = 46.08\text{MHz}$.

The following tables show the SCLK frequency in different RGB modes.

Table 15-34 2 data lane sclk frequency

2 data lane					
RGB	Frame	30Hz		60Hz	
	Resolution	240*320	320*480	240*320	320*480
RGB111		/	/	/	/
RGB444		20.736	41.472	41.472	82.944
RGB565		20.736	41.472	41.472	82.944
RGB666		23.04	46.08	46.08	92.16
RGB888		29.952	59.904	59.904	/

Table 15-35 3 line sclk frequency

3 line					
RGB	Frame	30Hz		60Hz	
	Resolution	240*320	320*480	240*320	320*480
RGB111		10.368	20.736	20.736	41.472
RGB444		31.104	62.208	62.208	/
RGB565		41.472	82.944	82.944	/
RGB666		62.208	/	/	/
RGB888		/	/	/	/

Table 15-36 4 line sclk frequency

4 line					
RGB	Frame	30Hz		60Hz	
	Resolution	240*320	320*480	240*320	320*480
RGB111		9.216	18.432	18.432	36.864
RGB444		27.648	55.296	55.296	/
RGB565		36.864	73.728	73.728	/
RGB666		55.296	/	/	/

4 line					
RGB	Frame	30Hz		60Hz	
	Resolution	240*320	320*480	240*320	320*480
RGB888		/	/	/	/

15.10.5 Programming Guidelines



The following are the programming guidelines in DBI mode. For those in SPI mode, see the Section 15.9 SPI.

15.10.5.1 Write Command Transfer

- Step 1** Set the SPI_DBI_MODE_SEL (bit3) of [SPI_GCR](#) (0x0004) to 1 to select DBI mode.
- Step 2** Set the DBI_EN_MODE_SEL (bit[30:29]) of [DBI_CTL_1](#) (0x0104) to 0 to select the trigger mode of DBI.
- Step 3** Configure the [DBI_CTL_0](#) (0x0100).
- Set [DBI_CTL_0](#)[Command Type] (bit31) to 0 to configure the writing command.
 - Set [DBI_CTL_0](#)[Write Command Dummy Cycles] (bit[30:20]) to configure the number of dummy cycles between commands.
 - Set [DBI_CTL_0](#)[Output Data Sequence] (bit19) to select the MSB or LSB.
 - Set [DBI_CTL_0](#)[Transmit Mode] (bit15) to 0 to select the command path.
 - Set [DBI_CTL_0](#)[Output Data Format] (bit[14:12]) to 0 to transmit the command.
 - Set [DBI_CTL_0](#)[DBI interface Select] (bit[10:8]) to select the DBI interface type.
 - The remaining values of the [DBI_CTL_0](#) register remain the default value.
- Step 4** Set [DBI_CTL_1](#)[DCX_DATA] (bit22) to 0 to send the command.
- Step 5** DMA Path: Configure the [SPI_FCR](#) register (0x0018).
- Set [SPI_FCR](#)[TF_DRQ_EN] (bit24) to 1 to enable TXFIFO DMA.
 - Set [SPI_FCR](#)[TX_TRIG_LEVEL] (bit[23:16]) to 255. It indicates the controller requests data from DMA if the remaining space of TX FIFO is greater than 255.
- CPU Path: Write the command to be sent to the 0x200 address.
- Step 6** Set [SPI_GCR](#)[DBI_EN] (bit4) to 1 to start transmitting the command.
- Step 7** Wait until the TX FIFO underrun interrupt ([SPI_ISR](#)[TF_UDF]) is 1. It indicates that the command written to the TX FIFO is transmitted completely.

15.10.5.2 Parameter Transfer

- Step 1** Set the SPI_DBI_MODE_SEL (bit3) of [SPI_GCR](#) (0x0004) to 1 to select DBI mode.
- Step 2** Set the DBI_EN_MODE_SEL (bit[30:29]) of [DBI_CTL_1](#) (0x0104) to 0 to select the trigger mode of DBI.
- Step 3** Configure the [DBI_CTL_0](#) register (0x0100).
- Set [DBI_CTL_0](#)[Command Type] (bit31) to 0 to configure the writing command.
 - Set [DBI_CTL_0](#)[Write Command Dummy Cycles] (bit[30:20]) to configure the number of dummy cycles between commands.
 - Set [DBI_CTL_0](#)[Output Data Sequence] (bit19) to select the MSB or LSB.
 - Set [DBI_CTL_0](#)[Transmit Mode] (bit15) to 0 to select the command path.
 - Set [DBI_CTL_0](#)[Output Data Format] (bit[14:12]) to 0 to transmit the command.
 - Set [DBI_CTL_0](#)[DBI interface Select] (bit[10:8]) to select the DBI interface type.
 - The remaining values of the [DBI_CTL_0](#) register remain the default value.
- Step 4** Set [DBI_CTL_1](#)[DCX_DATA] (bit22) to 1 to send the parameter.
- Step 5** DMA Path: Configure the [SPI_FCR](#) register (0x0018).
- Set [SPI_FCR](#)[TF_DRQ_EN] (bit24) to 1 to enable TXFIFO DMA.
 - Set [SPI_FCR](#)[TX_TRIG_LEVEL] (bit[23:16]) to 255. It indicates the controller requests data from DMA if the remaining space of TX FIFO is greater than 255.
- CPU Path: Write the command to be sent to the 0x200 address.
- Step 6** Set [SPI_GCR](#)[DBI_EN] (bit4) to 1 to start transmitting the command.
- Step 7** Wait until the TX FIFO underrun interrupt ([SPI_ISR](#)[TF_UDF]) is 1. It indicates that the command written to the TX FIFO is transmitted completely.

15.10.5.3 Video Transfer

Before the video transmission configuration, confirm whether the data is from CPU path or DMA path.

- If the data is from the CPU path, the controller writes the command to be sent to the 0x0200 address by the AHB bus.
- If the data is from the DMA path, configure [DBI_CTL_2](#) [DBI_FIFO_DRQ_EN] (bit15) to 1 and [DBI_CTL_2](#) [TX_TRIG_LEVEL] (bit[14:8]) to 64. The controller will request data from DMA if the remaining space of TX FIFO is greater than 64. The empty & full interrupts and the status bit of DBI FIFO can be configured in the [DBI_INT](#) and [DBI_DEBUG_0](#).

Software Trigger Mode

The software enables DBI_en_trigger when the edge interrupt of TE is detected.

After transmitting a frame of image, the controller automatically clears the line_cnt, pixel_cnt and stops data transmission.

The operation process is as follows.

- Step 1** Set the SPI_DBI_MODE_SEL (bit3) of [SPI_GCR](#) (0x0004) to 1 to select DBI mode.
- Step 2** Set the DBI EN MODE SEL (bit[30:29]) of [DBI_CTL_1](#) (0x0104) to 01 to select the software trigger mode.
- Step 3** Configure the [DBI_CTL_0](#) register (0x0100).
 - a) Set [DBI_CTL_0](#)[Command Type] (bit31) to 0 to set the writing command.
 - b) Set [DBI_CTL_0](#)[Write Command Dummy Cycles] (bit[30:20]) to configure the number of dummy cycles between commands.
 - c) Set [DBI_CTL_0](#)[Output Data Sequence] (bit19) to select the MSB or LSB.
 - d) Set [DBI_CTL_0](#)[Transmit Mode] (bit15) to 1 to select the image path.
 - e) Set [DBI_CTL_0](#)[Output Data Format] (bit[14:12]) to select RGB111/444/565/666/888.
 - f) Set [DBI_CTL_0](#)[DBI interface Select] (bit[10:8]) to select the DBI interface type.
 - g) The remaining values of the [DBI_CTL_0](#) register remain the default value.
- Step 4** Set [DBI_CTL_1](#)[DCX_DATA] (bit22) to 0 to send the image data.
- Step 5** Configure [DBI_Video_Size](#) (0x110) according to the sent image size.
- Step 6** Configure [DBI_CTL_2](#) (0x0108) to set the TE-related parameter.
- Step 7** Detect the TE interrupt of the [DBI_INT](#) (0x0120) register.
- Step 8** Configure [DBI_CTL_1](#)[DBI_soft_trigger] to 1.

Timer Trigger Mode

The software configures timer_en to enable timer counting, and when the counter reaches the specified value, the DBI_EN automatically will be enabled to start data transmission.

After transmitting a frame of image, the controller will automatically clear the line_cnt, pixel_cnt, and stop data transmission.

The timer starts counting again. When the counter reaches the specified value, the controller automatically enables DBI_EN again until the software turns off the timer_en.

The operation process is as follows.

- Step 1** Set the SPI_DBI_MODE_SEL (bit3) of [SPI_GCR](#) (0x0004) to 1 to select DBI mode.
- Step 2** Set the DBI EN MODE SEL (bit30:29) of [DBI_CTL_1](#) (0x0104) to 10 to select the timer trigger mode.

Step 3 Configure the [DBI_CTL_0](#) register (0x0100).

- a) Set [DBI_CTL_0](#)[Command Type] (bit31) to 0 to set the writing command.
- b) Set [DBI_CTL_0](#)[Write Command Dummy Cycles] (bit[30:20]) to configure the number of dummy cycles between commands.
- c) Set [DBI_CTL_0](#)[Output Data Sequence] (bit19) to select the MSB or LSB.
- d) Set [DBI_CTL_0](#)[Transmit Mode] (bit15) to 1 to select the image path.
- e) Set [DBI_CTL_0](#)[Output Data Format] (bit[14:12]) to select RGB111/444/565/666/888.
- f) Set [DBI_CTL_0](#)[DBI interface Select] (bit[10:8]) to select the DBI interface type.
- g) The remaining values of the [DBI_CTL_0](#) register remain the default value.

Step 4 Set [DBI_CTL_1](#)[DCX_DATA] (bit22) to 0 to send the image data.

Step 5 Configure [DBI_Video_Size](#) (0x110) to transmit the image size.

Step 6 Configure the related parameter of DBI_Timer (0x10C).

TE Trigger Mode

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

After transmitting a frame of image, the controller will automatically clear the line_cnt, pixel_cnt, and stop transmitting data.

When the edge changes of the TE are detected (The rising and falling edges are optional), the DBI_EN can be automatically enabled to restart data transmission until the software shuts down TE_EN or the screen no longer sends TE signals.

The operation process is as follows.

Step 1 Set the SPI_DBI_MODE_SEL (bit3) of [SPI_GCR](#) (0x0004) to 1 to select DBI mode.

Step 2 Set the DBI EN MODE SEL (bit30:29) of [DBI_CTL_1](#) (0x0104) to 11 to select the TE. Configure the [DBI_CTL_0](#) register (0x0100).

Step 3 Set [DBI_CTL_0](#)[Command Type] (bit31) to 0 to set the writing command.

- a) Set [DBI_CTL_0](#)[Write Command Dummy Cycles] (bit[30:20]) to configure the number of dummy cycles between commands.
- b) Set [DBI_CTL_0](#)[Output Data Sequence] (bit19) to select the MSB or LSB.
- c) Set [DBI_CTL_0](#)[Transmit Mode] (bit15) to 1 to select the image path.
- d) Set [DBI_CTL_0](#)[Output Data Format] (bit[14:12]) to select RGB111/444/565/666/888.
- e) Set [DBI_CTL_0](#)[DBI interface Select] (bit[10:8]) to select the DBI interface type.
- f) The remaining values of the [DBI_CTL_0](#) register remain the default value.

Step 4 Configure [DBI_CTL_1](#)[DCX_DATA] (bit22) to 0 to send the image data.

Step 5 Configure [DBI_Video_Size](#) (0x0110) to transmit the image size.

Step 6 Configure [DBI_CTL_2](#) (0x0108) to set the TE-related parameter.

15.10.5.4 Read Command & Read Data Transfer

Step 1 Set the SPI_DBI_MODE_SEL (bit3) of [SPI_GCR](#) (0x0004) to 1 to select DBI mode.

Step 2 Set the DBI EN MODE SEL (bit[30:29]) of [DBI_CTL_1](#) (0x0104) to 0.

Step 3 Configure the [DBI_CTL_0](#) register (0x0100).

- Set [DBI_CTL_0](#)[Command Type] (bit31) to 0 to set the reading command.
- Set [DBI_CTL_0](#)[Output Data Sequence] (bit19) to select the MSB or LSB.
- Set [DBI_CTL_0](#)[Transmit Mode] (bit15) to 0 to select the command path.
- Set [DBI_CTL_0](#)[Output Data Format] (bit[14:12]) to 0.
- Set [DBI_CTL_0](#)[DBI interface Select] (bit[10:8]) to select the DBI interface type.
- The remaining values of the [DBI_CTL_0](#) register remain the default value.

Step 4 Configure the [DBI_CTL_1](#) register (0x0104).

- Configure [DBI_CTL_1](#)[DCX_DATA] (bit22) to 0 to send the command.
- Configure [DBI_CTL_1](#)[Read_MSB_First] (bit20) to select whether the first bit of the read data is the highest or lowest bit of data.
- Configure [DBI_CTL_1](#)[Read Data Number of Bytes] to set the byte number to be read.
- Configure [DBI_CTL_1](#)[Read Command Dummy Cycles] to set the dummy cycle between the read command and the read data. When the dummy cycle is complete, the data starts to be sampled.

Step 5 DMA Path: Configure the [SPI_FCR](#) register (0x0018).

- Set [SPI_FCR](#)[RF_DRQ_EN] (bit8) to 1 to enable RXFIFO DMA.
- Set [SPI_FCR](#)[RX_TRIG_LEVEL] (bit[7:0]) to 32, which indicates the controller requests receiving data from DMA if the data of the RX FIFO is greater than 64.

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

Step 6 Set [SPI_GCR](#)[DBI_EN] (bit4) to 1 to start transmitting command.

Step 7 Wait until [DBI_INT](#)[RD_DONE_INT] is 1. It indicates that the data is read completely.

15.10.6 Register List

Module Name	Base Address
SPI1	0x04026000

Register Name	Offset	Description
SPI_GCR	0x0004	SPI Global Control Register
SPI_TCR	0x0008	SPI Transfer Control Register
SPI_IER	0x0010	SPI Interrupt Control Register
SPI_ISR	0x0014	SPI Interrupt Status Register
SPI_FCR	0x0018	SPI FIFO Control Register
SPI_FSR	0x001C	SPI FIFO Status Register
SPI_WCR	0x0020	SPI Wait Clock Counter Register
SPI_MBC	0x0030	SPI Master Burst Counter Register
SPI_MTC	0x0034	SPI Master Transmit Counter Register
SPI_BCC	0x0038	SPI Master Burst Control Counter Register
SPI_BATCR	0x0040	SPI Bit-Aligned Transfer Configure Register
SPI_BA_CCR	0x0044	SPI Bit-Aligned CLOCK Configuration Register
SPI_TBR	0x0048	SPI TX Bit Register
SPI_RBR	0x004C	SPI RX Bit Register
SPI_NDMA_MODE_CTL	0x0088	SPI Normal DMA Mode Control Register
DBI_CTL_0	0x0100	DBI Control Register 0
DBI_CTL_1	0x0104	DBI Control Register 1
DBI_CTL_2	0x0108	DBI Control Register 2
DBI_TIMER	0x010C	DBI Timer Control Register
DBI_VIDEO_SIZE	0x0110	DBI Video Size Register
DBI_INT	0x0120	DBI Interrupt Register
DBI_DEBUG_0	0x0124	DBI Debug Register 0
DBI_DEBUG_1	0x0128	DBI Debug Register 1
SPI_TXD	0x0200	SPI TX Data Register
SPI_RXD	0x0300	SPI RX Data Register
SPI_BSR	0x0400	SPI BUF Status Register

15.10.7 Register Description

15.10.7.1 0x0004 SPI Global Control Register (Default Value: 0x0000_0080)

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

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

15.10.7.2 0x0008 SPI Transfer Control Register (Default Value: 0x0000_0087)

Offset: 0x0008			Register Name: SPI_TCR
Bit	Read/Write	Default/Hex	Description

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

Offset: 0x0008			Register Name: SPI_TCR
Bit	Read/Write	Default/Hex	Description
12	R/W	0x0	FBS First Transmit Bit Select 0: MSB first 1: LSB first Note: When XCH=1, do not write this bit.
11	R/W	0x0	SDC Master Sample Data Control 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 propagating between master and slave. 0: Normal operation, do not delay internal read sample point 1: Delay internal read sample point Note: When XCH=1, do not write this bit.
10	R/W	0x0	RPSM Rapids mode select Select Rapids mode for high speed write. 0: Normal write mode 1: Rapids write mode Note: When XCH=1, do not write this bit.
9	R/W	0x0	DDB Dummy Burst Type 0: The bit value of dummy SPI burst is zero 1: The bit value of dummy SPI burst is one Note: When XCH=1, do not write this bit.
8	R/W	0x0	DHB Discard Hash Burst In master mode it controls whether discarding unused SPI bursts 0: Receiving all SPI bursts in BC period 1: Discard unused SPI bursts, only fetching the SPI bursts during dummy burst period. The bursts number is specified by TC. Note: When XCH=1, do not write this bit.

Offset: 0x0008			Register Name: SPI_TCR
Bit	Read/Write	Default/Hex	Description
7	R/W	0x1	SS_LEVEL When control CS signal manually (SPI_CTRL_REG.SS_CTRL==1), set this bit to '1' or '0' to control the level of CS signal. 0: Set CS to low 1: Set CS to high Note: When XCH=1, do not write this bit.
6	R/W	0x0	SS_OWNER CS Output Owner Select Usually, controller sends CS signal automatically with data together. When this bit is set to 1, software must manually write SPI_CTL_REG.SS_LEVEL to 1 or 0 to control the level of CS signal. 0: SPI controller 1: Software Note: When XCH=1, do not write this bit.
5:4	R/W	0x0	SS_SEL SPI Chip Select Select one of four external SPI Master/Slave Devices 00: SPI_CS0 will be asserted 01: SPI_CS1 will be asserted (not used) 10: SPI_CS2 will be asserted (not used) 11: SPI_CS3 will be asserted (not used) Note: When XCH=1, do not write this bit.
3	R/W	0x0	SSCTL In master mode, this bit selects the output wave form for the SPI_CSx signal. Only valid when SS_OWNER = 0. 0: SPI_CSx remains asserted between SPI bursts 1: Negate SPI_CSx between SPI bursts Note: When XCH=1, do not write this bit.
2	R/W	0x1	SPOL SPI Chip Select Signal Polarity Control 0: Active high polarity (0 = Idle) 1: Active low polarity (1 = Idle) Note: When XCH=1, do not write this bit.

Offset: 0x0008			Register Name: SPI_TCR
Bit	Read/Write	Default/Hex	Description
1	R/W	0x1	CPOL SPI Clock Polarity Control 0: Active high polarity (0 = Idle) 1: Active low polarity (1 = Idle) Note: When XCH=1, do not write this bit.
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) Note: When XCH=1, do not write this bit.

15.10.7.3 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 (CSx) from valid state to 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 under run 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 under run Interrupt Enable 0: Disable 1: Enable

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

15.10.7.4 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 CS Invalid Interrupt When SSI is 1, it indicates that CS has changed from valid state to invalid state. Writing 1 to this bit clears it.

Offset: 0x0014			Register Name: SPI_ISR
Bit	Read/Write	Default/Hex	Description
12	R/W1C	0x0	TC Transfer Completed In master mode, it indicates that all bursts specified by BC has been exchanged. In other condition, When set, this bit indicates that all the data in TXFIFO has been loaded in the Shift register, and the Shift register has shifted out all the bits. Writing 1 to this bit clears it. 0: Busy 1: Transfer Completed
11	R/W1C	0x0	TF_UDF TXFIFO under run This bit is set when if the TXFIFO is underrun. Writing 1 to this bit clears it. 0: TXFIFO is not underrun 1: TXFIFO is underrun
10	R/W1C	0x0	TF_OVF TXFIFO Overflow This bit is set when if the TXFIFO is overflow. Writing 1 to this bit clears it. 0: TXFIFO is not overflow 1: TXFIFO is overflowed
9	R/W1C	0x0	RX_UDF RXFIFO Underrun When set, this bit indicates that RXFIFO has underrun. Writing 1 to this bit clears it.
8	R/W1C	0x0	RX_OVF RXFIFO Overflow When set, this bit indicates that RXFIFO has overflowed. Writing 1 to this bit clears it. 0: RXFIFO is available. 1: RXFIFO has overflowed.
7	/	/	/
6	R/W1C	0x0	TX_FULL TXFIFO Full This bit is set when if the TXFIFO is full . Writing 1 to this bit clears it. 0: TXFIFO is not full 1: TXFIFO is Full

Offset: 0x0014			Register Name: SPI_ISR
Bit	Read/Write	Default/Hex	Description
5	R/W1C	0x1	TX_EMP TXFIFO Empty This bit is set if the TXFIFO is empty. Writing 1 to this bit clears it. 0: TXFIFO contains one or more words. 1: TXFIFO is empty
4	R/W1C	0x1	TX_READY TXFIFO Ready 0: TX_WL > TX_TRIG_LEVEL 1: TX_WL <= TX_TRIG_LEVEL This bit is set any time if TX_WL <= TX_TRIG_LEVEL. Writing "1" to this bit clears it. Where TX_WL is the water level of RXFIFO
3	/	/	/
2	R/W1C	0x0	RX_FULL RX Full This bit is set when the RX FIFO is full. Writing 1 to this bit clears it. 0: Not Full 1: Full
1	R/W1C	0x1	RX_EMP RX Empty This bit is set when the RX FIFO is empty. Writing 1 to this bit clears it. 0: Not empty 1: Empty
0	R/W1C	0x0	RX_RDY RX Ready 0: RX_WL <= RX_TRIG_LEVEL 1: RX_WL > RX_TRIG_LEVEL This bit is set any time if RX_WL > RX_TRIG_LEVEL. Writing "1" to this bit clears it. Where RX_WL is the water level of RX FIFO.

15.10.7.5 0x0018 SPI FIFO Control Register (Default Value: 0x0040_0001)

Offset: 0x0018			Register Name: SPI_FCR
Bit	Read/Write	Default/Hex	Description

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

Offset: 0x0018			Register Name: SPI_FCR
Bit	Read/Write	Default/Hex	Description
8	R/W	0x0	RF_DRQ_EN RX FIFO DMA Request Enable 0: Disable 1: Enable
7:0	R/W	0x1	RX_TRIG_LEVEL RX FIFO Ready Request Trigger Level

15.10.7.6 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	/	/	/

Offset: 0x001C			Register Name: SPI_FSR
Bit	Read/Write	Default/Hex	Description
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

15.10.7.7 0x0020 SPI Wait Clock Counter Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: SPI_WCR
Bit	Read/Write	Default/Hex	Description
31:20	/	/	/
19:16	R/W	0x0	SWC Dual mode direction switch wait clock counter (for master mode only). 0: No wait states inserted n: n SPI_CLK wait states inserted These bits control the number of wait states to be inserted before start dual data transfer in dual SPI mode. The SPI module counts SPI_CLK by SWC for delaying next word data transfer. Note: When XCH=1, do not write this bit.
15:0	R/W	0x0	WCC Wait Clock Counter (In Master mode) These bits control the number of wait states to be inserted in data transfers. The SPI module counts SPI_CLK by WCC for delaying next word data transfer. 0: No wait states inserted N: N SPI_CLK wait states inserted Note: When XCH=1, do not write this bit.

15.10.7.8 0x0030 SPI Master Burst Counter Register (Default Value: 0x0000_0000)

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

Offset: 0x0030			Register Name: SPI_MBC
Bit	Read/Write	Default/Hex	Description
23:0	R/W	0x0	MBC Master Burst Counter In master mode, this field specifies the total burst number. 0: 0 burst 1: 1 burst ... N: N bursts Note: - When XCH=1, do not write this bit. - Total transfer data include the TXD, RXD and dummy burst.

15.10.7.9 0x0034 SPI Master Transmit Counter Register (Default Value: 0x0000_0000)

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

15.10.7.10 0x0038 SPI Master Burst Control Counter Register (Default Value: 0x0000_0000)

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

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

15.10.7.11 0x0040 SPI Bit-Aligned Transfer Configure Register (Default Value: 0x0000_00A0)

Offset: 0x0040			Register Name: SPI_BATCR
Bit	Read/Write	Default/Hex	Description

Offset: 0x0040			Register Name: SPI_BATCR
Bit	Read/Write	Default/Hex	Description
31	R/WAC	0x0	TCE Transfer Control Enable In master mode, it is used to start t1o transfer the serial bits frame, it is only valid when Work Mode Select==0x2/0x3. 0: Idle 1: Initiates transfer. Write “1” to this bit will start to transfer serial bits frame (the value comes from the SPI TX Bit Register or SPI RX Bit Register), and will auto clear after the bursts transfer completely. Write ‘0’ to this bit has no effect.
30	R/W	0x0	MSMS Master Sample Standard 0: Standard Sample Mode 1: Delay Sample Mode In Standard Sample Mode, SPI master samples the data at the standard rising edge of CLK for each SPI mode. In Delay Sample Mode, SPI master samples data at the edge that is half cycle delayed by the standard rising edge of CLK defined in respective SPI mode.
29:26	/	/	/
25	R/W1C	0x0	TBC Transfer Bits Completed 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. 0: Busy 1: Transfer Completed Note: It is only valid when Work Mode Select=0x2/0x3.
24	R/W	0x0	TBC_INT_EN Transfer Bits Completed Interrupt Enable 0: Disable 1: Enable Note: It is only valid when Work Mode Select=0x2/0x3.
23:22	/	/	/

Offset: 0x0040			Register Name: SPI_BATCR
Bit	Read/Write	Default/Hex	Description
21:16	R/W	0x00	RX_FEM_LEN Configure the length of serial data frame(burst) of RX 000000: 0bit 000001: 1bit ... 100000: 32bits Other values: Reserved Note: It is only valid when Work Mode Select=0x2/0x3, and cannot be written when TCE=1.
15:14	/	/	/
13:8	R/W	0x00	TX_FRM_LEN Configure the length of serial data frame(burst) of TX 000000: 0bit 000001: 1bit ... 100000: 32bits Other values: Reserved Note: It is only valid when Work Mode Select=0x2/0x3, and cannot be written when TCE=1.
7	R/W	0x1	SS_LEVEL When control CS signal manually, set this bit to '1' or '0' to control the level of CS signal. 0: Set CS to low 1: Set CS to high Note: It is only valid when Work Mode Select=0x2/0x3, and only work in Mode0, cannot be written when TCE=1.

Offset: 0x0040			Register Name: SPI_BATCR
Bit	Read/Write	Default/Hex	Description
6	R/W	0x0	SS_OWNER CS Output Owner Select Usually, controller sends CS signal automatically with data together. When this bit is set to 1, software must manually write SPI_CTL_REG.SS_LEVEL to 1 or 0 to control the level of CS signal. 0: SPI controller 1: Software Note: It is only valid when Work Mode Select=0x2/0x3, and only work in Mode0, cannot be written when TCE=1.
5	R/W	0x1	SPOL SPI Chip Select Signal Polarity Control 0: Active high polarity (0 = Idle) 1: Active low polarity (1 = Idle) Note: It is only valid when Work Mode Select=0x2/0x3, and only work in Mode0, cannot be written when TCE=1.
4	/	/	/
3:2	R/W	0x0	SS_SEL SPI Chip Select Select one of four external SPI Master/Slave Devices 00: SPI_CS0 will be asserted 01: SPI_CS1 will be asserted (not used) 10: SPI_CS2 will be asserted (not used) 11: SPI_CS3 will be asserted (not used) Note: It is only valid when Work Mode Select =0x2/0x3, and only work in Mode0, cannot be written when TCE=1.
1:0	R/W	0x0	WMS 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

15.10.7.12 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	CDR_N Clock Divide Rate (Master Mode Only) The SPI_CLK is determined according to the following equation: $SPI\_CLK = Source\_CLK / (2 * (CDR\_N + 1))$.

15.10.7.13 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	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. Note: This register is only valid when Work Mode Select=0x2/0x3.

15.10.7.14 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	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. Note: This register is only valid when Work Mode Select=0x2/0x3.

15.10.7.15 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	/	/	/

Offset: 0x0088			Register Name: SPI_NDMA_MODE_CTL
Bit	Read/Write	Default/Hex	Description
7:6	R/W	0x3	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
5	R/W	0x1	SPI_ACK_M SPI NDMA Acknowledge Mode 0: Active fall do not care ack 1: Active fall must after detect ack is high
4:0	R/W	0x05	SPI_DMA_WAIT The counts of hold cycles from DMA last signal high to DMA_ACTIVE high.

15.10.7.16 0x0100 DBI Control Register 0 (Default Value: 0x0010_0000)

Offset: 0x0100			Register Name: DBI_CTL_0
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	CMDT Command Type 0: Write Command 1: Read Command
30:20	R/W	0x1	WCDC Write Command Dummy Cycles Controls dummy cycles between two write commands Range 1-255 Default Condition: There is a dbi_clk cycle between each command or parameter.
19	R/W	0x0	DAT_SEQ Output Data Sequence 0: MSB First 1: LSB First

Offset: 0x0100			Register Name: DBI_CTL_0
Bit	Read/Write	Default/Hex	Description
18:16	R/W	0x0	RGB_SEQ Output RGB Sequence 000: RGB 001: RBG 010: GRB 011: GBR 100: BRG 101: BGR 110-111: Reserved
15	R/W	0x0	TRAN_MOD Transmit Mode 0: Command / Parameter 1: Video
14:12	R/W	0x0	DAT_FMT Output Data Format 000: RGB111 001: RGB444 010: RGB565 011: RGB666 100: RGB888 (only for 2 Data Lane Interface) 101-111: Reserved
11	/	/	/
10:8	R/W	0x0	DBI_INTF DBI Interface 000: 3 Line Interface I 001: 3 Line Interface II 010: 4 Line Interface I 011: 4 Line Interface II 100: 2 Data Lane Interface

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

15.10.7.17 0x0104 DBI Control Register 1 (Default Value: 0x0000_0001)

Offset: 0x0104	Register Name: DBI_CTL_1

Bit	Read/Write	Default/Hex	Description
31	R/WAC	0x0	DBI_SOFT_TRG It is only available for software trigger mode. Write '1' to this bit will start DBI TX module and auto clear to '0' when completing start operation, write to '0' has no effect.
30:29	R/W	0x0	DBI_EN_MOD_SEL DBI Enable Mode Select 0: Always on mode 1: Software trigger mode 2: Timer trigger mode 3: Te trigger mode
28	/	/	/
27:26	R/W	0x0	RGB666_FMT 2 Data Lane RGB666 Format 00: Normal Format 01: Special Format for ILITEK 10: Special Format for New Vision
25	R/W	0x0	DBI_RXCLK_INV DBI rx clock inverse 0: Sample data by using the positive edge of the output clock 1: Sample data by using the negative edge of the output clock
24	R/W	0x0	DBI_CLKO_MOD DBI output clock mode 0: DBI clock always on(DCX Setup/hold equals one clock cycle) 1: DBI clock auto gating(DCX Setup/hold equals to a half clock cycle)
23	R/W	0x0	DBI_CLKO_INV DBI clock output inverse When the bit24 (DBI output clock mode) is 0. 0: The falling edge releases the CSX signal, and the falling edge releases data 1: The rising edge releases the CSX signal, and the rising edge releases data When the bit24 (DBI output clock mode) is 1. 0: The rising edge releases the CSX signal, and the falling edge releases data 1: The falling edge releases the CSX signal, and the rising edge releases data

22	R/W	0x0	DCX_DATA DCX Data Value 0: DCX Value equal to 0 1: DCX Value equal to 1
21	R/W	0x0	RGB 16 Data Source Select RGB 16 Data Source Select 0: Pixel1 is stored in the higher bit of address, and Pixel0 is stored in the lower bit of address 1: Pixel0 is stored in the higher bit of address, and Pixel1 is stored in the lower bit of address
20	R/W	0x0	RDAT_LSB Bit Order of Read Data 0: A reading data is the higher bit 1: A reading data is the lower bit
19:16	/	/	/
15:8	R/W	0x0	RCDC Read Command Dummy Cycles The dummy cycle between the read command and read data Reading 1-byte (8 bits) data has not dummy cycle.
7:0	R/W	0x1	RDBN Read Data Number of Bytes Sample Bytes data based on configuration.

15.10.7.18 0x0108 DBI Control Register 2 (Default Value: 0x0000_4000)

Offset: 0x0108			Register Name: DBI_CTL_2
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15	R/W	0x0	DBI_FIFO_DRQ_EN DBI FIFO DMA Request Enable 0: Disable 1: Enable
14:8	R/W	0x40	DBI_TRIG_LEVEL DBI FIFO Empty Request Trigger Level
7	/	/	/
6	R/W	0x0	DBI_SDI_OUT_SEL DBI SDI PIN Output Select The signal is used with the DBI SDI PIN Function Sel bit. 0: Output WRX (When DBI DCX PIN Function Sel = 0, the SDI pin outputs data) 1: Output DCX

Offset: 0x0108			Register Name: DBI_CTL_2
Bit	Read/Write	Default/Hex	Description
5	R/W	0x0	DBI_DCX_SEL DBI DCX PIN Function Sel 0: DBI DCX Function 1: WRX(2 Data Lane Interface)
4:3	R/W	0x0	DBI_SDI_SEL DBI SDI PIN Function sel 0: DBI_SDI (Interface II) , WRX(2 Data Lane Interface) 1: DBI_TE 2: DBI_DCX 3: Reserved
2	R/W	0x0	TE_DBC_SEL TE debounce function select 0: Debounce 1: No-debounce
1	R/W	0x0	TE_TRIG_SEL TE edge trigger select 0: TE rising edge 1: TE falling edge
0	R/W	0x0	TE_EN TE enable 0: TE Disable 1: TE Enable

15.10.7.19 0x010C DBI Timer Control Register (Default Value: 0x0000_0000)

Offset: 0x010C			Register Name: DBI_TIMER
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	DBI_TM_EN DBI Timer enable 0: Enable 1: Disable

Offset: 0x010C			Register Name: DBI_TIMER
Bit	Read/Write	Default/Hex	Description
30:0	R/W	0x0	DBI_TM_VAL DBI Timer Value It is used to set the time at which the interrupt of the DBI Timer is triggered. When the Timer_EN is 1, the timer starts to count (the counting clock is CLK). When the counter reaches the target value to trigger the Timer_INT of DBI, the data will start to be sent in serially. Note: Counting will stop when the data is being sent serially. The interval between two data transfers set by DBI Timer Value is called frame blanking.

15.10.7.20 0x0110 DBI Video Size Register (Default Value: 0x01E0_0140)

Offset: 0x0110			Register Name: DBI_VIDEO_SIZE
Bit	Read/Write	Default/Hex	Description
31:27	/	/	/
26:16	R/W	0x1e0	V_SIZE It is used to generate the Frame int.
15:11	/	/	/
10:0	R/W	0x140	H_SIZE It is used to generate the Line int.

15.10.7.21 0x0120 DBI Interrupt Register (Default Value: 0x0000_4000)

Offset: 0x0120			Register Name: DBI_INT
Bit	Read/Write	Default/Hex	Description
31:15	/	/	/
14	W1C	0x1	DBI_FIFO_EMPTY_INT DBI FIFO Empty Interrupt Status 0: DBI_FIFO is not empty 1: DBI_FIFO is empty
13	W1C	0x0	DBI_FIFO_FULL_INT DBI FIFO Full Interrupt Status 0: DBI_FIFO is not full. 1: DBI_FIFO is full.

Offset: 0x0120			Register Name: DBI_INT
Bit	Read/Write	Default/Hex	Description
12	W1C	0x0	TIMER_INT It indicates that timer has been count clk cycles to the value of DBI_Timer Register[30:0] . Writing 1 to this bit clears it. 0: Timer hasn't been achieved objective 1: Timer has been achieved objective
11	W1C	0x0	RD_DONE_INT It indicates that the number of byte setting in DBI_Control Register 1 [19:8] has been read. Writing 1 to this bit clears it. 0: All of data hasn't been read. 1: All of data has been read.
10	W1C	0x0	TE_INT It indicates that TE signal has been changed. Writing 1 to this bit clears it. 0: TE signal isn't changed. 1: TE signal has been changed.
9	W1C	0x0	FRAM_DONE_INT It indicates that a frame video data has been send. Writing 1 to this bit clears it. 0: A frame of video has not been send. 1: A frame of video has been send.
8	W1C	0x0	LINE_DONE_INT It indicates that a line of video data has been send. Writing 1 to this bit clears it. 0: A line of video data has not been send 1: A line of video data has been send
7	/	/	/
6	R/W	0x0	DBI_FIFO_EMPTY_INT_EN DBI FIFO Empty Interrupt Enable 0: Disable 1: Enable
5	R/W	0x0	DBI_FIFO_FULL_INT_EN DBI FIFO Full Interrupt Enable 0: Disable 1: Enable
4	R/W	0x0	TIMER_INT_EN Timer Interrupt Enable 0: Disable 1: Enable

Offset: 0x0120			Register Name: DBI_INT
Bit	Read/Write	Default/Hex	Description
3	R/W	0x0	RD_DONE_INT_EN Read Done Interrupt Enable 0: Disable 1: Enable
2	R/W	0x0	TE_INT_EN TE Interrupt Enable 0: Disable 1: Enable
1	R/W	0x0	FRAM_DONE_INT_EN Frame Done Interrupt Enable 0: Disable 1: Enable
0	R/W	0x0	LINE_DONE_INT_EN Line Done Interrupt Enable 0: Disable 1: Enable

15.10.7.22 0x0124 DBI Debug Register 0 (Default Value: 0x007F_0000)

Offset: 0x0124			Register Name: DBI_DEBUG_0
Bit	Read/Write	Default/Hex	Description
31:23	/	/	/
22:16	R	0x7F	DBI_FIFO_AVAIL DBI FIFO ROOM VALID 0-127 Words
15:13	/	/	/
12	R	0x0	TE_VAL TE input value 0: TE not Trigger 1: TE Trigger
11:8	R	0x0	DBI_RXCS FSM for DBI Receive RX_BS0 - RX_BS6 , Gray - Code
7:4	R	0x0	SH_CS FSM for shifter 0000-0011: SH0-SH11

Offset: 0x0124			Register Name: DBI_DEBUG_0
Bit	Read/Write	Default/Hex	Description
3:2	R	0x0	DBI_TXCS FSM for DBI Transmit 00: IDLE 01: SHIF 10: DUMY 11: READ
1:0	R	0x0	MEM_CS FSM for DBI Memory 00: IDLE_FRM 01: FRM_POS 10: FRM_RDY

15.10.7.23 0x0128 DBI Debug Register 1 (Default Value: 0x0000_0000)

Offset: 0x0128			Register Name: DBI_DEBUG_1
Bit	Read/Write	Default/Hex	Description
31:26	/	/	/
25:16	R	0x0	LCNT Line counter The number of pixel lines that are currently sent
15:12	/	/	/
11:0	R	0x0	CCNT Component counter The number of RGB components that are currently sent The field is equal to pixel_cnt *3.

15.10.7.24 0x0200 SPI TX Data Register (Default Value: 0x0000_0000)

Offset: 0x0200			Register Name: SPI_TXD
Bit	Read/Write	Default/Hex	Description

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

15.10.7.25 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	RDATA Receive Data This register can be accessed in byte, half-word or word unit by AHB. In byte accessing method, if there are data in RXFIFO, the top word is returned and the RXFIFO depth is decreased by 1. In half-word accessing method, two SPI bursts are returned and the RXFIFO depth is decrease by 2. In word accessing method, the four SPI bursts are returned and the RXFIFO depth is decreased by 4. Note: This address is read-only if RF_TEST is '0', and if RF_TEST is set to '1', this address is readable and writable to test the RX FIFO through the AHB bus.

15.10.7.26 0x0400 SPI BUF Status Register (Default Value: 0x0000_0000)

Offset: 0x0400			Register Name: SPI_BSR
Bit	Read/Write	Default/Hex	Description
31:27	/	/	/

Offset: 0x0400			Register Name: SPI_BSR
Bit	Read/Write	Default/Hex	Description
26:24	R	0x0	TX_SHIFT_BUFFER_CNT TX SHIFT Buffer Counter These bits indicate the number of bytes in TX Shift Buffer, which is after TX FIFO, and maximum depth is 4 level with 8 bits.
23:19	/	/	/
18:16	R	0x0	TB_CNT TX FIFO Write Buffer Counter These bits indicate the number of bytes in TX FIFO Write Buffer
15:8	/	/	/
7:0	R	0x0	RB_CNT RX FIFO Read Buffer Counter These bits indicate the number of bytes in RX FIFO Read Buffer

15.11 CIR Receiver (IR_RX)

15.11.1 Overview

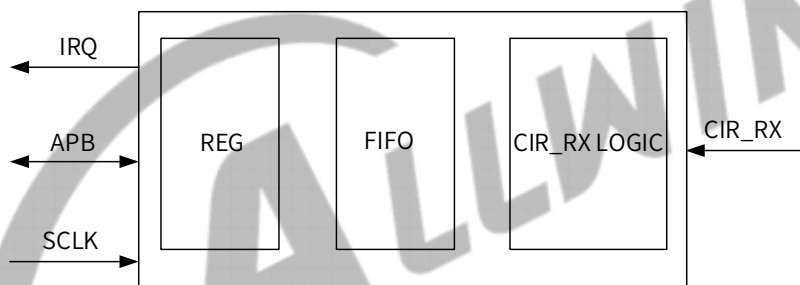
The Consumer Infrared (CIR) receiver captures pulse from the IR Receiver module and uses the Run-Length Code (RLC) to encode the pulse.

The CIR receiver has the following features:

- One IR_RX interface
- Supports CIR for remote control
- 64x8 bits FIFO for data buffer
- Sample clock up to 1 MHz
- Programmable FIFO thresholds

15.11.2 Block Diagram

Figure 15-83 CIR Receiver Block Diagram



The CIR receiver samples the input signal on the programmable frequency and records these samples into RX FIFO when one CIR signal is found on the air. The CIR receiver uses Run-Length Code (RLC) to encode pulse width. The encoded data is buffered in 64 levels and 8-bit width RX FIFO; the MSB bit is used to record the polarity of the receiving CIR signal, the rest 7 bits are used for the length of RLC. The maximum length of the RLC is 128. If the duration of one level (high or low level) is more than 128, another byte is used.

15.11.3 External Signals

The following table describes the external signals of CIR Receiver.

Table 15-37 CIR Receiver External Signals

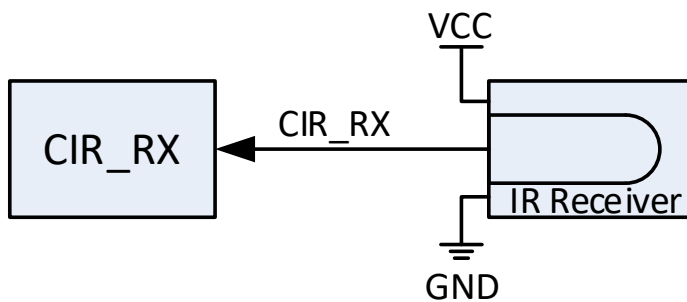
Signal Name	Description	Type
IR-RX	Consumer Infrared Receiver	I

15.11.4 Functional Description

15.11.4.1 Typical Application

The following figure shows IR_RX controller IO signals. To connect with external devices, you could configure GPIO controller to enable IR_RX model.

Figure 15-84 CIR Receiver Application Diagram



15.11.4.2 NEC Protocol

Figure 15-85 A Typical Pulse Train of the NEC



The CIR receiver module is a timer with a capture function.

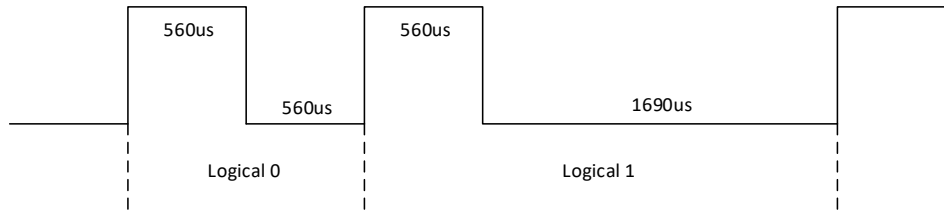
When IR_RX signals satisfy the Active Threshold (ATHR), the CIR receiver starts capturing. The signals whose pulse width less than Noise Threshold (NTHR) will be ignored. When the signals satisfy Idle Threshold (ITHR), the capture process will be stopped and the Receiver Packet End interrupt will be generated, then the Receiver Packet End Flag will be asserted.

In a capture process, every effective pulse is buffered to FIFO in bytes according to the form of the Run-Length Code. The MSB bit of a byte indicates the polarity of pulse, and the rest 7 bits indicates the pulse width by taking Sample Clock as a basic unit. This is the code form of the RLC-Byte. When the level changes or the pulse width counter overflows, the RLC-Byte is buffered to FIFO. Afterwards, the IR_RX module receives the infrared signals transmitted by the infrared remote control, the software decodes the signals.

15.11.4.3 Operating Mode

Sample Clock

Figure 15-86 Logical '0' and Logical '1' of NEC Protocol



For NEC protocol, a logical "1" takes 2.25 ms (560 us+1680 us) to transmit, while a logical "0" is only half of that, being 1.12 ms (560 us+560 us).

For example, if the sample clock is 31.25 kHz, and a sample cycle is 32 us, then 18 sample cycles are 560 us. So the RLC of 560 us low level is 0x12 (b'00010010), the RLC of 560 us high level is 0x92 (b'10010010). Then a logical "1" takes code 0x12 (b'00010010) and code 0xb5 (b'10110101) to transmit, a logical "0" takes code 0x12 and code 0x92 to transmit.



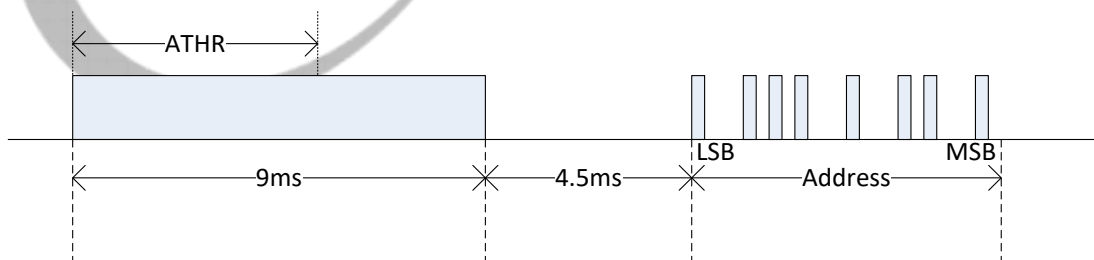
NOTE

Input signals can be inverted by setting the RPPI of the [CIR_RXCTL](#) register to 1.

ATHR

When the CIR receiver is in the idle state, if the electrical level of the IR_RX signal changes (positive jump or negative jump), and the duration reaches this threshold, then the CIR receiver takes the starting of the signal as a lead code, and the CIR receiver turns into an active state and starts to capture IR_RX signals.

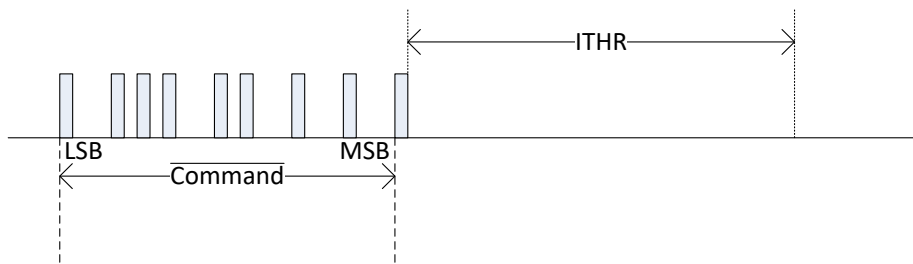
Figure 15-87 ATHR Definition



ITHR

If the electrical level of IR_RX signals has no change, and the duration reaches this threshold, then the CIR receiver enters into Idle state and ends this capture.

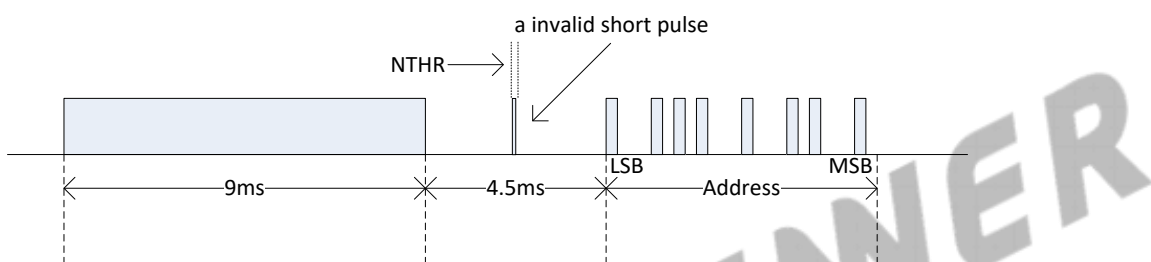
Figure 15-88 ITHR Definition



NTHR

In the capture process, the pulse is ignored if the pulse width is less than the Noise Threshold.

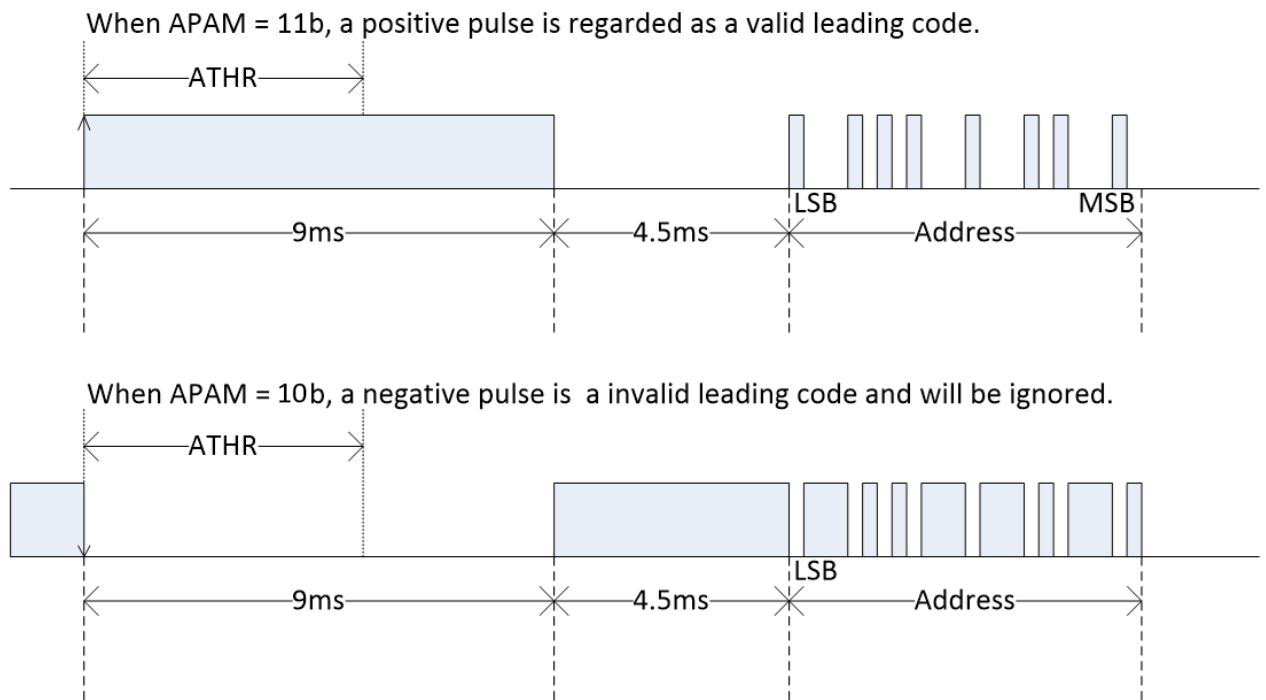
Figure 15-89 NTHR Definition



APAM

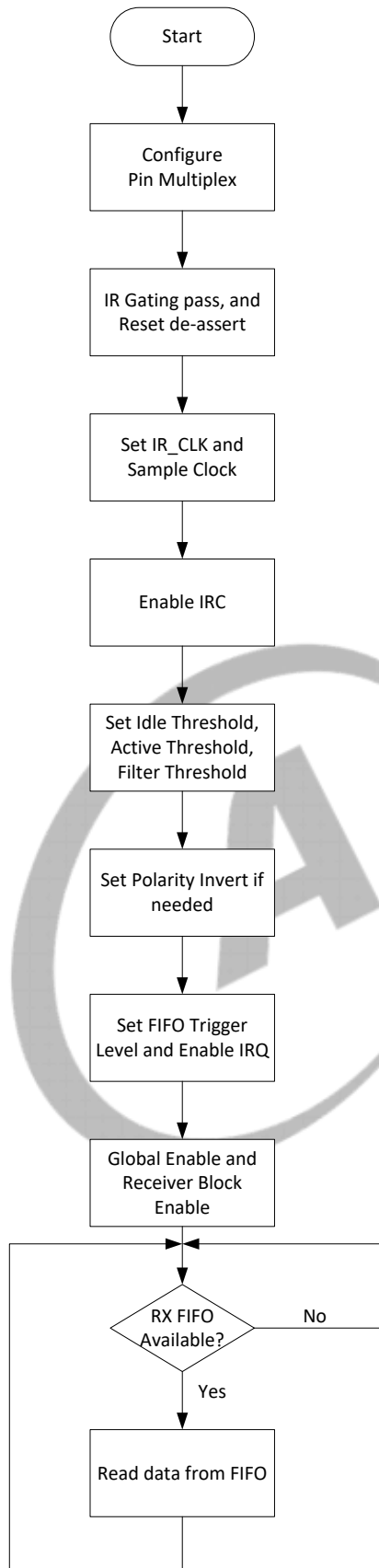
The APAM is used to fit the type of lead code. If a pulse does not fit the type of lead code, it is not regarded as a lead code even if the pulse width reaches ATHR.

Figure 15-90 APAM Definition



15.11.5 Operation Guidelines

Figure 15-91 CIR Receiver Process



15.11.6 Register List

Module Name	Base Address
IR_RX	0x02037000

Register Name	Offset	Description
CIR_CTL	0x0000	CIR Receiver Control Register
CIR_RXCTL	0x0010	CIR Receiver Configure Register
CIR_RXFIFO	0x0020	CIR Receiver FIFO Register
CIR_RXINT	0x002C	CIR Receiver Interrupt Control Register
CIR_RXSTA	0x0030	CIR Receiver Status Register
CIR_CONFIG	0x0034	CIR Receiver Configure Register

15.11.7 Register Description

15.11.7.1 0x0000 CIR Receiver Control Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: CIR_CTL
Bit	Read/Write	Default/HEX	Description
31:8	/	/	/
7:6	R/W	0x0	APAM Active Pulse Accept Mode 00, 01: Both positive and negative pulses are valid as a leading code. 10: Only negative pulse is valid as a leading code. 11: Only positive pulse is valid as a leading code.
5:4	R/W	0x0	CEN CIR ENABLE 00-10: Reserved 11: CIR mode enable
3:2	/	/	/
1	R/W	0x0	RXEN Receiver Block Enable 0: Disable 1: Enable
0	R/W	0x0	GEN Global Enable A disable on this bit overrides any other block or channel enables and flushes all FIFOs. 0: Disable 1: Enable

15.11.7.2 0x0010 CIR Receiver Configure Register (Default Value: 0x0000_0004)

Offset: 0x0010			Register Name: CIR_RXCTL
Bit	Read/Write	Default/HEX	Description
31:3	/	/	/
2	R/W	0x1	RPPI Receiver Pulse Polarity Invert. 0: Do not invert receiver signal 1: Invert receiver signal
1:0	/	/	/

15.11.7.3 0x0020 CIR Receiver FIFO Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: CIR_RXFIFO
Bit	Read/Write	Default/HEX	Description
31:8	/	/	/
7:0	R	0x0	RBF Receiver Byte FIFO

15.11.7.4 0x002C CIR Receiver Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x002C			Register Name: CIR_RXINT
Bit	Read/Write	Default/HEX	Description
31:14	/	/	/
13:8	R/W	0x0	RAL RX FIFO Available Received Byte Level for interrupt and DMA request TRIGGER_LEVEL = RAL + 1
5	R/W	0x0	DRQ_EN RX FIFO DMA Enable 0: Disable 1: Enable When set to '1', the Receiver FIFO DRQ is asserted if reaching RAL. The DRQ is de-asserted when condition fails.
4	R/W	0x0	RAI_EN RX FIFO Available Interrupt Enable 0: Disable 1: Enable When set to '1', the Receiver FIFO IRQ is asserted if reaching RAL. The IRQ is de-asserted when condition fails.

Offset: 0x002C			Register Name: CIR_RXINT
Bit	Read/Write	Default/HEX	Description
3:2	/	/	/
1	R/W	0x0	RPEI_EN Receiver Packet End Interrupt Enable 0: Disable 1: Enable
0	R/W	0x0	ROI_EN Receiver FIFO Overrun Interrupt Enable 0: Disable 1: Enable

15.11.7.5 0x0030 CIR Receiver Status Register (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: CIR_RXSTA
Bit	Read/Write	Default/HEX	Description
31:15	/	/	/
14:8	R	0x0	RAC RX FIFO Available Counter 0: No available data in RX FIFO 1: 1-byte available data in RX FIFO 2: 2-byte available data in RX FIFO ... 64: 64 byte available data in RX FIFO
7	R	0x0	STAT Status of CIR 0x0: Idle 0x1: Busy
6:5	/	/	/
4	R/W1C	0x0	RA RX FIFO Available 0: RX FIFO not available according its level 1: RX FIFO available according its level This bit is cleared by writing a '1'.
3:2	/	/	/
1	R/W1C	0x0	RPE Receiver Packet End Flag 0: STO was not detected. In CIR mode, one CIR symbol is receiving or not detected. 1: STO field or packet abort symbol (7' b0000,000 and 8' b0000,0000 for MIR and FIR) is detected. In CIR mode, one CIR symbol is received. This bit is cleared by writing a '1'.

Offset: 0x0030			Register Name: CIR_RXSTA
Bit	Read/Write	Default/HEX	Description
0	R/W1C	0x0	ROI Receiver FIFO Overrun 0: Receiver FIFO not overrun 1: Receiver FIFO overrun This bit is cleared by writing a '1' .

15.11.7.6 0x0034 CIR Receiver Configure Register (Default Value: 0x0000_1828)

Offset: 0x0034			Register Name: CIR_CONFIG
Bit	Read/Write	Default/HEX	Description
31:25	/	/	/
24	R/W	0x0	SCS2 Bit2 of Sample Clock Select for CIR This bit is defined by SCS bits below.
23	R/W	0x0	ATHC Active Threshold Control for CIR 0x0: ATHR in Unit of (Sample Clock) 0x1: ATHR in Unit of (128*Sample Clocks)
22:16	R/W	0x0	ATHR Active Threshold for CIR These bits control the duration of CIR from Idle to Active State. The duration can be calculated by $((ATHR + 1) * (ATHC ? \text{Sample Clock: } 128 * \text{Sample Clock}))$.
15:8	R/W	0x18	ITHR Idle Threshold for CIR The Receiver uses it to decide whether the CIR command has been received. If there is no CIR signal on the air, the receiver is staying in IDLE status. One active pulse will bring the receiver from IDLE status to Receiving status. After the CIR is end, the inputting signal will keep the specified level (high or low level) for a long time. The receiver can use this idle signal duration to decide that it has received the CIR command. The corresponding flag is asserted. If the corresponding interrupt is enabled, the interrupt line is asserted to CPU. When the duration of signal keeps one status (high or low level) for the specified duration $((ITHR + 1) * 128 \text{ sample\_clk})$, this means that the previous CIR command has been finished.

Offset: 0x0034			Register Name: CIR_CONFIG																																				
Bit	Read/Write	Default/HEX	Description																																				
7:2	R/W	0xa	NTHR Noise Threshold for CIR When the duration of signal pulse (high or low level) is less than NTHR, the pulse is taken as noise and should be discarded by hardware. 0: All samples are recorded into RX FIFO 1: If the signal is only one sample duration, it is taken as noise and discarded. 2: If the signal is less than ($\leq$) two sample duration, it is taken as noise and discarded. ... 61: If the signal is less than ($\leq$) sixty-one sample duration, it is taken as noise and discarded.																																				
1:0	R/W	0x0	SCS Sample Clock Select for CIR <table border="1"> <thead> <tr> <th>SCS2</th><th>SCS[1]</th><th>SCS[0]</th><th>Sample Clock</th></tr> </thead> <tbody> <tr> <td>0</td><td>0</td><td>0</td><td>IR_CLK/64</td></tr> <tr> <td>0</td><td>0</td><td>1</td><td>IR_CLK/128</td></tr> <tr> <td>0</td><td>1</td><td>0</td><td>IR_CLK/256</td></tr> <tr> <td>0</td><td>1</td><td>1</td><td>IR_CLK/512</td></tr> <tr> <td>1</td><td>0</td><td>0</td><td>IR_CLK</td></tr> <tr> <td>1</td><td>0</td><td>1</td><td>Reserved</td></tr> <tr> <td>1</td><td>1</td><td>0</td><td>Reserved</td></tr> <tr> <td>1</td><td>1</td><td>1</td><td>Reserved</td></tr> </tbody> </table>	SCS2	SCS[1]	SCS[0]	Sample Clock	0	0	0	IR_CLK/64	0	0	1	IR_CLK/128	0	1	0	IR_CLK/256	0	1	1	IR_CLK/512	1	0	0	IR_CLK	1	0	1	Reserved	1	1	0	Reserved	1	1	1	Reserved
SCS2	SCS[1]	SCS[0]	Sample Clock																																				
0	0	0	IR_CLK/64																																				
0	0	1	IR_CLK/128																																				
0	1	0	IR_CLK/256																																				
0	1	1	IR_CLK/512																																				
1	0	0	IR_CLK																																				
1	0	1	Reserved																																				
1	1	0	Reserved																																				
1	1	1	Reserved																																				

16 Security

16.1 Crypto Engine(CE)

16.1.1 Overview

The CE module is an accelerator for encryption and decryption algorithms, supporting multiple modes and cpu and dma access.

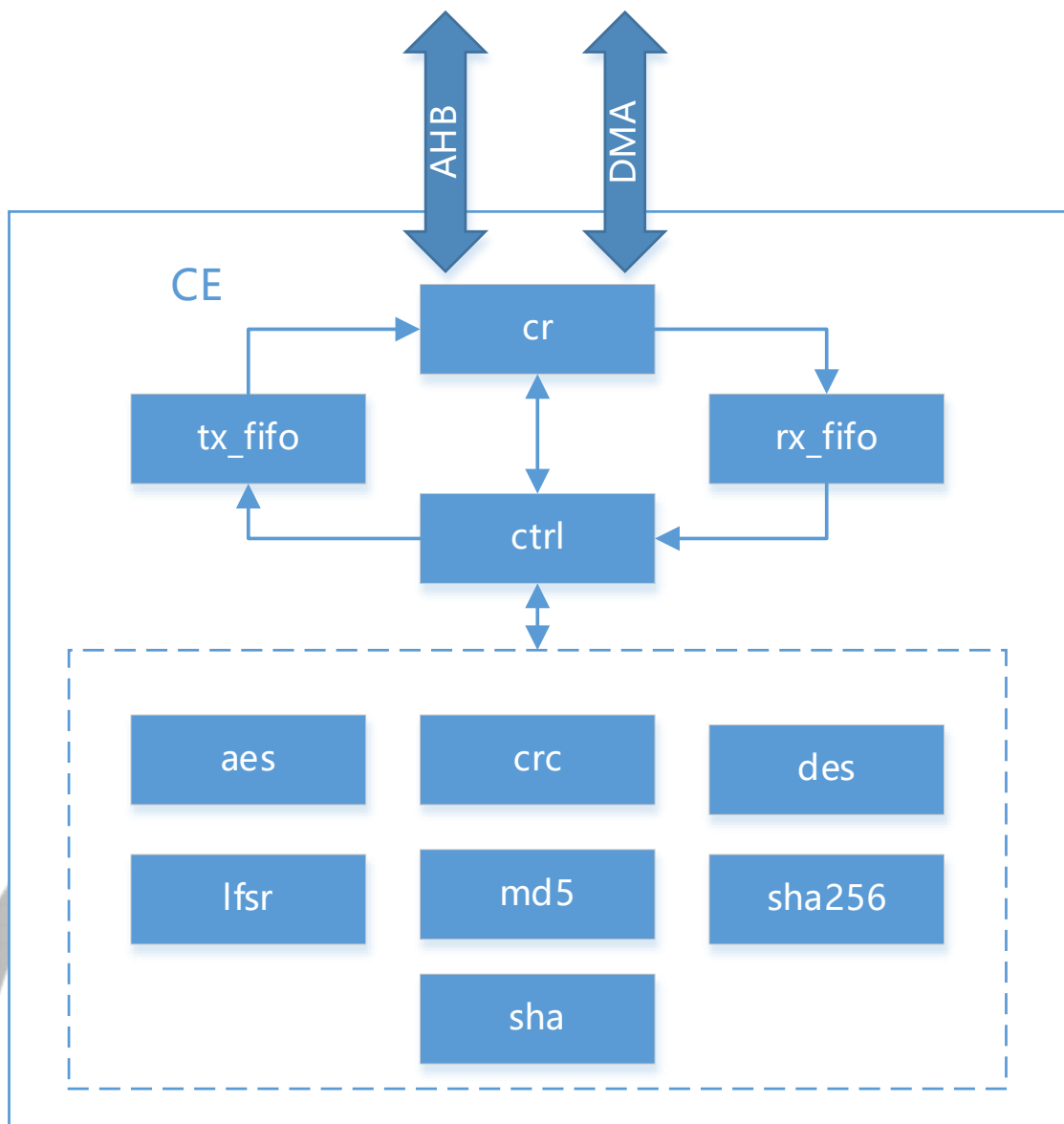
The CE has the following features:

- Supports AES, DES, 3DES, SHA-1, MD5, PRNG, CRC32/16, SHA256 encryption and decryption protocols
- AES/DES/3DES support ECB, CBC, CTR modes
- AES supports CTS mode
- AES key size supports 128bits, 192bits and 256bits
- Can generate a 160-bit pseudo-random number (PRNG) based on a 192-bit seed by hardware.
- Supports RX FIFO with a size of 32x32
- Supports TX FIFO with a size of 32x32
- Supports AHB interface and CPU access
- Supports DMA interface and external DMA access

16.1.2 Block Diagram

The following figure shows the block diagram of CE.

Figure 16-1 CE Block Diagram



The cr module is a register module with an AHB interface and a DMA interface. The input and output data are cached by rx fifo and tx fifo respectively. The ctrl module is responsible for controlling the entire encryption and decryption process. Modules such as aes and crc are hardware accelerators for different encryption and decryption protocols, implementing various encryption and decryption algorithms through hardware.

16.1.3 Functional Discription

16.1.3.1 SSK in DES/3DES mode

SSK is only valid for [127:0] bits, but the 3DES mode requires 192 bits of key, so the 3DES mode does not support the use of SSK.

When SSK is used in 3DES mode, [127:0]bit inner key is the efuse value, and [191:128]bit inner key is 0;

The key size of DES mode is 64-bit, and DES mode only uses the [63:0]bit of SSK.

16.1.4 Operation Guide

Step 1 Configure the registers and set parameters such as the working mode of ce;

Step 2 Enable the ce module;

Step 3 Carry out corresponding data transmission according to different working modes.



NOTE

The bus clock must not exceed six times the module clock; otherwise, the CRC algorithm will be abnormal.

16.1.5 Register List

Module Name	Base Address	Comments
CE	0x0304 0800	

Register Name	Offset	Description
CE		
CE_CTL	0x000	CE Control Register
CE_KEY0	0x004	CE Input Key 0 Register
CE_KEY1	0x008	CE Input Key 1 Register
CE_KEY2	0x00C	CE Input Key 2 Register
CE_KEY3	0x010	CE Input Key 3 Register
CE_KEY4	0x014	CE Input Key 4 Register
CE_KEY5	0x018	CE Input Key 5 Register
CE_KEY6	0x01C	CE Input Key 6 Register
CE_KEY7	0x020	CE Input Key 7 Register
CE_INIT_VEC0	0x024	CE Initialization Vector 0 Register
CE_INIT_VEC1	0x028	CE Initialization Vector 1 Register
CE_INIT_VEC2	0x02C	CE Initialization Vector 2 Register
CE_INIT_VEC3	0x030	CE Initialization Vector 3 Register
CE_CNT0	0x034	CE Counter 0 Register
CE_CNT1	0x038	CE Counter 1 Register
CE_CNT2	0x03C	CE Counter 2 Register
CE_CNT3	0x040	CE Counter 3 Register
CE_FCTL	0x044	CE FIFO Control/ Status Register
CE_INT	0x048	CE Interrupt Control/ Status Register
CE_MSG_DIGEST0	0x04C	CE Message Digest 0 Register
CE_MSG_DIGEST1	0x050	CE Message Digest 1 Register
CE_MSG_DIGEST2	0x054	CE Message Digest 2 Register
CE_MSG_DIGEST3	0x058	CE Message Digest 3 Register

Register Name	Offset	Description
CE_MSG_DIGEST4	0x05C	CE Message Digest 4 Register
CE_CTS_LEN	0x060	CE AES-CTS text length Register
CE_CRC_POLY	0x070	CE CRC Poly Register
CE_RESULT	0x074	CE Result Register
CE_MSG_DIGEST5	0x0a0	CE Message Digest 5 Register
CE_MSG_DIGEST6	0x0a4	CE Message Digest 6 Register
CE_MSG_DIGEST7	0x0a8	CE Message Digest 7 Register
CE_RXFIFO	0x200	CE Rx Fifo Register
CE_TXFIFO	0x204	CE Tx Fifo Register

16.1.6 Register Description

16.1.6.1 0x000 CE Control Register (Default Value: 0x0000_0000)

Offset: 0x000			Register Name: CE_CTL
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	CRC_XOROUT crc result XOR enabled 0: The crc result is directly output 1: The crc result is XORED with 0xffffffff (CRC32) or 0xffff (CRC16) and then output
30	R/W	0x0	CRC_REFOUT Output data inversion enable 0: The output data remains unchanged 1: The output data is reversed bit by bit
29	R/W	0x0	CRC_REFIN Input data inversion enable 0: The input data remains unchanged 1: The input data is reversed bit by bit
28	R/W	0x0	CRC_INIT Selection of the initial crc value 0: The initial value is 0 1: The initial value is 0xffffffff (CRC32) or 0xffff (CRC16).

Offset: 0x000			Register Name: CE_CTL
Bit	Read/Write	Default/Hex	Description
27:24	R/W	0x0	INKEY_SEL AES/DES/3DES key selection 0000: Select the input CE_KEYx(regular mode) 0001: Select the ssk from SID 0010: / 0011-1010: Select the internal Key n (n is inkey_sel-3) Others: Reserved Note 1) The ssk from SID is only 128 bits 2) When using an ssk from SID, only the lower 128-bit key is valid, and the upper 128-bit key is 0
23:19	R	0x0	Reserved
18:16	R	0x0	chip_id
15	R/W	0x0	SS_PRNG_MODE PRNG mode selection /CRC package continuity selection 0: PRNG single mode /CRC Last package 1: PRNG continuous mode /CRC is not the last package
14	R/W	0x0	SS_USE_IV Enable the IV mode of the SHA-1/MD5 initial constant 0: Use the constant defined by FIPS-180 as the initial constant 1: Use the input IV as the initial constant
13:12	R/W	0x0	SS_MODE Selection of CE operation mode 00: Electronic Code Book (ECB) mode 01: Cipher Block Chaining (CBC) mode 10: Counter (CTR) mode 11: Cipher Text Stealing (CTS) Pattern
11:10	R/W	0x0	SS_CNT_WIDTH The counter bit width of CTR mode 00:16 bits 01:32 bits 10:64 bits 11:128 bits

Offset: 0x000			Register Name: CE_CTL
Bit	Read/Write	Default/Hex	Description
9:8	R/W	0x0	SS_KEY_SIZE The key bit width of AES 00:128 bits 01:192 bits 10:256 bits 11: Reserved
7	R/W	0x0	SS_DIR Direction selection 0: Encryption 1: Decryption Note: Ss_mode is valid in non-CTR mode. In CTR mode, regardless of the value of this bit, it is encrypted
6:4	R/W	0x0	SS_TYPE Selection of CE encryption and decryption methods 000: AES 001: DES 010:3DES 011: SHA-1 100: MD5 101: PRNG 110: SHA256 111: CRC
3	R/W	0x0	GP_WIDTH CRC width 0: Width is 16 1: Width is 32
2	R/WAC	0x0	SS_SM_END SHA-1/MD5/SHA256/CRC data termination signal The software notifies the SHA-1/MD5/SHA256/CRC engine data end by writing "1". If the RX fifo is not empty, the engine will process the data in the RX fifo. After the information processing is completed, this bit is automatically zeroed by the hardware, and the processed data can be read from the digest register. Note: This bit is only used by the SHA-1/MD5/SHA256/CRC engine

Offset: 0x000			Register Name: CE_CTL
Bit	Read/Write	Default/Hex	Description
1	R/WAC	0x0	SS_PRNG_RUN PRNG start signal In PRNG single-mode, write "1" to enable PRNG. After generating a set of random numbers (5 words), the hardware automatically clears them to zero Note: When ss_enb is set to 0, this bit is automatically cleared to 0
0	R/W	0x0	SS_ENB CE enable signal 0: Disabled 1: Enable

16.1.6.2 0x004 CE Key 0 Register (Default Value: 0x0000_0000)

Offset: 0x004			Register Name: CE_KEY0
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	ss_key0

Note:

When the input CE_KEYx (inkey_sel is 0) is selected, these 8 registers are valid. When other keys are selected, these registers are invalid and the read value is 0

16.1.6.3 0x008 CE Key 1 Register (Default Value: 0x0000_0000)

Offset: 0x008			Register Name: CE_KEY1
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	ss_key1

Note: Same as CE_KEY0

16.1.6.4 0x00C CE Key 2 Register (Default Value: 0x0000_0000)

Offset: 0x00C			Register Name: CE_KEY2
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	ss_key2

Note: Same as CE_KEY0

16.1.6.5 0x010 CE Key 3 Register (Default Value: 0x0000_0000)

Offset: 0x010			Register Name: CE_KEY3

Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	ss_key3

Note: Same as CE_KEY0

16.1.6.6 0x014 CE Key 4 Register (Default Value: 0x0000_0000)

Offset: 0x014			Register Name: CE_KEY4
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	ss_key4

Note: Same as CE_KEY0

16.1.6.7 0x018 CE Key 5 Register (Default Value: 0x0000_0000)

Offset: 0x018			Register Name: CE_KEY5
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	ss_key5

Note: Same as CE_KEY0

16.1.6.8 0x01C CE Key 6 Register (Default Value: 0x0000_0000)

Offset: 0x01C			Register Name: CE_KEY6
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	ss_key6

Note: Same as CE_KEY0

16.1.6.9 0x020 CE Key 7 Register (Default Value: 0x0000_0000)

Offset: 0x020			Register Name: CE_KEY7
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	ss_key7

Note: Same as CE_KEY0

16.1.6.10 0x024 CE Initialization Vector 0 Register (Default Value: 0x0000_0000)

Offset: 0x024			Register Name: CE_INIT_VEC0
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	ss_iv0

16.1.6.11 0x028 CE Initialization Vector 1 Register (Default Value: 0x0000_0000)

Offset: 0x028			Register Name: CE_INIT_VEC1
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	ss_iv1

16.1.6.12 0x02C CE Initialization Vector 2 Register (Default Value: 0x0000_0000)

Offset: 0x02C			Register Name: CE_INIT_VEC2
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	ss_iv2

16.1.6.13 0x030 CE Initialization Vector 3 Register (Default Value: 0x0000_0000)

Offset: 0x030			Register Name: CE_INIT_VEC3
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	ss_iv3

16.1.6.14 0x034 CE Counter 0 Register (Default Value: 0x0000_0000)

Offset: 0x034			Register Name: CE_CNT0
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	ss_cnt0

16.1.6.15 0x038 CE Counter 1 Register (Default Value: 0x0000_0000)

Offset: 0x038			Register Name: CE_CNT1
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	ss_cnt1

16.1.6.16 0x03C CE Counter 2 Register (Default Value: 0x0000_0000)

Offset: 0x03C			Register Name: CE_CNT2
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	ss_cnt2

16.1.6.17 0x040 CE Counter 3 Register (Default Value: 0x0000_0000)

Offset: 0x040			Register Name: CE_CNT3
Bit	Read/Write	Default/Hex	Description

Offset: 0x040			Register Name: CE_CNT3
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	ss_cnt3

16.1.6.18 0x044 CE FIFO Control/Status Register (Default Value: 0x6000_0F0F)

Offset: 0x044			Register Name: CE_FCTL
Bit	Read/Write	Default/Hex	Description
31	R	0x0	Reserved
30	R	0x1	RX_FULL_N rx fifo state 0: Full 1: Non-full
29:24	R	0x20	RX_LEVEL_SPC rx fifo Remaining space counter
23	R	0x0	Reserved
22	R	0x0	TX_EMPTY_N tx fifo is in an empty or full state 0: Empty 1: Non-empty
21:16	R	0x0	TX_LEVEL_DAT tx fifo receiving data counter
15:13	R	0x0	Reserved
12:8	R/W	0x0f	ss_rxth
7:5	R	0x0	Reserved
4:0	R/W	0xf	ss_txth

16.1.6.19 0x048 CE Interrupt Control/Status Register (Default Value: 0x0000_0000)

Offset: 0x048			Register Name: CE_INT
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11	R/W1C	0x0	HASH_END Hash crc terminates the interrupt state
10	R/W1C	0x0	RX_INT_PD rx interrupt status
9	/	/	/
8	R/W1C	0x0	TX_INT_PD tx interrupt status
7:5	/	/	/

Offset: 0x048			Register Name: CE_INT
Bit	Read/Write	Default/Hex	Description
4	R/W	0x0	DRQ_EN drq enable signal 0: Disabled 1: Enable
3	R/W	0x0	HINT_EN Hash crc terminates the interrupt enable signal 0: Disabled 1: Enable
2	R/W	0x0	RX_INT_ENB rx interrupts the enable signal 0: Disabled 1: Enable
1	R/W	0x0	Reserved
0	R/W	0x0	TX_INT_EN tx interrupts the enable signal 0: Disabled 1: Enable

16.1.6.20 0x4c CE Message Digest 0 Register (Default Value: 0x0000_0000)

Offset: 0x4c			Register Name: CE_MSG_DIGEST0
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	MD_RES0 SHA1/ MD5 Message digest for SHA1/MD5

16.1.6.21 0x50 CE Message Digest 1 Register (Default Value: 0x0000_0000)

Offset: 0x50			Register Name: CE_MSG_DIGEST1
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	MD_RES1 SHA1/ MD5 Message digest for SHA1/MD5

16.1.6.22 0x54 CE Message Digest 2 Register (Default Value: 0x0000_0000)

Offset: 0x54			Register Name: CE_MSG_DIGEST2
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	MD_RES2 SHA1/ MD5 Message digest for SHA1/MD5

16.1.6.23 0x58 CE Message Digest 3 Register (Default Value: 0x0000_0000)

Offset: 0x58			Register Name: CE_MSG_DIGEST3
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	MD_RES3 SHA1/ MD5 Message digest for SHA1/MD5

16.1.6.24 0x5C CE Message Digest 4 Register (Default Value: 0x0000_0000)

Offset: 0x5C			Register Name: CE_MSG_DIGEST4
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	MD_RES4 SHA1/ MD5 Message digest for SHA1/MD5

16.1.6.25 0x060 CE AES-CTS text length Register (Default Value: 0x0000_0000)

Offset: 0x060			Register Name: CE_CTS_LEN
Bit	Read/Write	Default/Hex	Description
31:0	W	0x0	CS_LEN The length of AES-CTS, in bytes. A value of 0 indicates that there is no data

16.1.6.26 0x70 CE CRC Poly Register (Default Value: 0x0000_0000)

Offset: 0x70			Register Name: CE_CRC_POLY
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	CRC_GP CRC poly nominal configuration

16.1.6.27 0x74 CE Result Register (Default Value: 0x0000_0000)

Offset: 0x74			Register Name: CE_RESULT
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	CRC_OUT crc output

16.1.6.28 0xa0 CE Message Digest 5 Register (Default Value: 0x0000_0000)

Offset: 0xa0			Register Name: CE_MSG_DIGEST5
Bit	Read/Write	Default/Hex	Description

Offset: 0xa0			Register Name: CE_MSG_DIGEST5
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	MD_RES5 SHA256 Message digest for SHA1/MD5

16.1.6.29 0xa4 CE Message Digest 6 Register (Default Value: 0x0000_0000)

Offset: 0xa4			Register Name: CE_MSG_DIGEST6
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	MD_RES6 SHA256 Message digest for SHA1/MD5

16.1.6.30 0xa8 CE Message Digest 7 Register (Default Value: 0x0000_0000)

Offset: 0xa8			Register Name: CE_MSG_DIGEST7
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	MD_RES7 SHA256 Message digest for SHA1/MD5

16.1.6.31 0x200 CE Rx Fifo Register (Default Value: 0x0000_0000)

Offset: 0x200			Register Name: CE_RXFIFO
Bit	Read/Write	Default/Hex	Description
31:0	W	0x0	The input port of the RX fifo, only supports 32-bit write

16.1.6.32 0x204 CE Tx Fifo Register (Default Value: 0x0000_0000)

Offset: 0x204			Register Name: CE_TXFIFO
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	The output port of the TX fifo, only supports 32-bit write

Copyright©2025 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 © 2025 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