Rockchip
RK1808
Technical Reference Manual

Revision 1.2
Aug. 2019

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Revision History

Date	Revision	Description
2019-8-26	1.2	Cancel Instruction Chapter
2019-3-1	1.1	Update DDR3 capacity from 2GB to 4GB
2019-1-21	1.0	Initial Release

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Table of Content

Table of Content 3
Figure Index 9
Table Index 12
NOTICE 14
Chapter 1 System Overview 15
1.1 Address Mapping 15
1.2 System Boot 17
1.3 System Interrupt connection 18
1.4 System DMA hardware request connection 21
Chapter 2 Clock & Reset Unit (CRU) 23
2.1 Overview 23
2.2 Block Diagram 23
2.3 System Reset Solution 23
2.4 Function Description 24
2.5 PLL Introduction 24
2.6 Register Description 25
2.7 Application Notes 126
Chapter 3 System Debug 129
3.1 Overview 129
3.2 Block Diagram 129
3.3 Function Description 129
3.4 Register Description 129
3.5 Interface Description 129
Chapter 4 General Register File (GRF) 131
4.1 Overview 131
4.2 Function Description 131
4.3 BUS_GRF Register Description 131
4.4 USB2PHY_GRF Register Description 234
4.5 PCIe_USB3_PHY_GRF Register Description 249
4.6 PMU_GRF Register Description 257
4.7 DDR_GRF Register Description 287
4.8 PCIe_USB_GRF Register Description 293
4.9 CORE_GRF Register Description 299

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 5 Cortex-A35 307
5.1 Overview 307
5.2 Block Diagram 307
5.3 Function Description 308
Chapter 6 Direct Memory Access Controller (DMAC) 309
6.1 Overview 309
6.2 Block Diagram 310
6.3 Function Description 310
6.4 Register Description 311
6.5 Timing Diagram 348
6.6 Interface Description 349
6.7 Application Notes 351
Chapter 7 Generic Interrupt Controller (GIC) 357
7.1 Overview 357
7.2 Block Diagram 357
7.3 Function Description 358
Chapter 8 Power Management Unit (PMU) 359
8.1 Overview 359
8.2 Block Diagram 359
8.3 Function Description 360
8.4 Register Description 361
8.5 Timing Diagram 392
8.6 Application Note 393
Chapter 9 Process Voltage Temperature Monitor (PVTM) 395
9.1 Overview 395
9.2 Block Diagram 395
9.3 Function Description 395
9.4 Application Notes 397
Chapter 10 Advanced eXtensible Interface Performance (AXI PERF) 398
10.1 Overview 398
10.2 Block Diagram 398
10.3 Register Description 398
10.4 Application Notes 404
Chapter 11 Mobile Storage Host Controller 407

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

11.1 Overview 407
11.2 Block Diagram 408
11.3 Function Description 409
11.4 Register Description 430
11.5 Interface Description 455
11.6 Application Notes 457
Chapter 12 Serial Flash Controller (SFC) 479
12.1 Overview 479
12.2 Block Diagram 479
12.3 Function Description 479
12.4 Register Description 480
12.5 Interface Description 488
12.6 Application Notes 489
Chapter 13 Embedded SRAM 492
13.1 Overview 492
13.2 Block Diagram 492
13.3 Function Description 492
Chapter 14 Spinlock 493
14.1 Overview 493
14.2 Block Diagram 493
14.3 Function Description 493
14.4 Register Description 493
14.5 Register Description 494
Chapter 15 Neural Process Unit (NPU) 495
15.1 Overview 495
15.2 Block Diagram 495
15.3 Function Description 496
15.4 Register Description 497
Chapter 16 Video Input Processor(VIP) 498
16.1 Overview 498
16.2 Block Diagram 498
16.3 Function Description 498
16.4 Register Description 499
16.5 Interface Description 528
16.6 Application Notes 529

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 17 GMAC Ethernet Interface 530
17.1 Overview 530
17.2 Block Diagram 531
17.3 Function Description 532
17.4 Register Description 536
17.5 Interface Description 590
17.6 Application Notes 591
Chapter 18 Pulse Density Modulation Interface Controller 604
18.1 Overview 604
18.2 Block Diagram 604
18.3 Function Description 605
18.4 Register Description 607
18.5 Interface Description 615
18.6 Application Notes 616
Chapter 19 I2S 2-Chanel 617
19.1 Overview 617
19.2 Block Diagram 617
19.3 Function description 618
19.4 Register Description 621
19.5 Interface Description 630
19.6 Application Notes 631
Chapter 20 I2S 8-Channel 633
20.1 Overview 633
20.2 Block Diagram 634
20.3 Function description 634
20.4 Register description 640
20.5 Interface Description 654
20.6 Application Notes 656
Chapter 21 SPI2APB 658
21.1 Overview 658
21.2 Block Diagram 658
21.3 Function Description 658
21.4 Register Description 660
21.5 Interface Description 664
21.6 Application Notes 664

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 22 Serial Peripheral Interface (SPI) 665
22.1 Overview 665
22.2 Block Diagram 665
22.3 Function Description 666
22.4 Register Description 668
22.5 Interface Description 678
22.6 Application Notes 679
Chapter 23 Universal Asynchronous Receiver/Transmitter (UART) 682
23.1 Overview 682
23.2 Block Diagram 682
23.3 Function Description 683
23.4 Register Description 686
23.5 Interface Description 703
23.6 Application Notes 705
Chapter 24 I2C Interface 709
24.1 Overview 709
24.2 Block Diagram 709
24.3 Function Description 709
24.4 Register Description 712
24.5 Interface Description 722
24.6 Application Notes 723
Chapter 25 GPIO 727
25.1 Overview 727
25.2 Block Diagram 727
25.3 Function Description 727
25.4 Register Description 729
25.5 Interface Description 732
25.6 Application Notes 733
Chapter 26 Timer 735
26.1 Overview 735
26.2 Block Diagram 735
26.3 Function Description 735
26.4 Register Description 736
26.5 Application Notes 738
26.6 Register Base Address 738

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

26.7 Clock and Enable 738
Chapter 27 Pulse Width Modulation (PWM) 740
27.1 Overview 740
27.2 Block Diagram 740
27.3 Function Description 741
27.4 Register Description 742
27.5 Interface Description 760
27.6 Application Notes 761
Chapter 28 Watchdog Timer(WDT) 763
28.1 Overview 763
28.2 Block Diagram 763
28.3 Function Description 763
28.4 Register Description 764
28.5 Application Notes 767
Chapter 29 Successive Approximation Register ADC(SARADC) 768
29.1 Overview 768
29.2 Block Diagram 768
29.3 Function Description 768
29.4 Register description 768
29.5 Timing Diagram 770
29.6 Application Notes 770
Chapter 30 Temperature Sensor ADC(TSADC) 771
30.1 Overview 771
30.2 Block Diagram 771
30.3 Function Description 771
30.4 Register description 772
30.5 Application Notes 776

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Figure Index

Fig. 1-1 RK1808 boot procedure flow 18
Fig. 2-1 CRU Block Diagram 23
Fig. 2-2 Reset Architecture Diagram 23
Fig. 2-3 PLL Block Diagram 25
Fig. 3-1 Debug system structure 129
Fig. 3-2 DAP SWJ interface 130
Fig. 3-3 SW-DP acknowledgement timing 130
Fig. 5-1 Block Diagram 308
Fig. 6-1 Block diagram of DMAC 310
Fig. 6-2 DMAC operation states 311
Fig. 6-3 DMAC request and acknowledge timing 349
Fig. 7-1 Block Diagram 358
Fig. 8-1 Power Domain Partition 359
Fig. 8-2 PMU Bock Diagram 360
Fig. 8-3 Each Domain Power Switch Timing 392
Fig. 8-4 External Wakeup Source PAD Timing 393
Fig. 9-1 PVTM Block Diagram 395
Fig. 10-1 AXI_PERF block diagram 398
Fig. 11-1 Host Controller Block Diagram 408
Fig. 11-2 SD/MMC Card-Detect Signal 413
Fig. 11-3 Host Controller Command Path State Machine 415
Fig. 11-4 Host Controller Data Transmit State Machine 417
Fig. 11-5 Host Controller Data Receive State Machine 419
Fig. 11-6 Dual-Buffer Descriptor Structure 425
Fig. 11-7 Chain Descriptor Structure 426
Fig. 11-8 Descriptor Formats for 32-bit AHB Address Bus Width 426
Fig. 11-9 SD/MMC Card-Detect and Write-Protect 457
Fig. 11-10 SD/MMC Card Termination 457
Fig. 11-11 Host Controller Initialization Sequence 458
Fig. 11-12 Voltage Switching Command Flow Diagram 468
Fig. 11-13 ACMD41 Argument 468
Fig. 11-14 ACMD41 Response(R3) 469
Fig. 11-15 Voltage Switch Normal Scenario 469
Fig. 11-16 Voltage Switch Error Scenario 470
Fig. 11-17 CASES for eMMC 4.5 START bit 472
Fig. 11-18 Clock Generation Unit 475
Fig. 11-19 Card Detection Method 2 477
Fig. 11-20 Card Detection Method 4 477
Fig. 12-1 SFC Architecture 479
Fig. 12-2 Idle cycles 480
Fig. 12-3 SPI mode 480
Fig. 12-4 Slave mode write 489
Fig. 12-5 Slave mode read 490
Fig. 12-6 Master mode flow 491
Fig. 12-7 SPI mode 491
Fig. 13-1 Embedded SRAM block diagram 492
Fig. 14-1 Spinlock Block Diagram 493
Fig. 15-1 NPU Block Diagram 496
Fig. 16-1 VIP Block Diagram 498
Fig. 17-1 GMACArchitecture 531
Fig. 17-2 GMAC Architecture 531
Fig. 17-3 MAC Block Diagram 532
Fig. 17-4 RMII transmission bit ordering 532
Fig. 17-5 End of MII and RMII Transmission in 10-Mbps Mode 533
Fig. 17-6 RMII receive bit ordering 533

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Fig. 17-7 MDIO frame structure 534
Fig. 17-8 Descriptor Ring and Chain Structure 591
Fig. 17-9 Rx/Tx Descriptors definition 591
Fig. 17-10 RMII clock architecture when clock source from CRU 600
Fig. 17-11 RMII clock architecture when clock source from external OSC 600
Fig. 17-12 RGMII clock architecture when clock source from CRU 601
Fig. 17-13 Wake-Up Frame Filter Register 601
Fig. 18-1 PDM Block Diagram 604
Fig. 18-2 PDM with Eight Mono MIC 605
Fig. 18-3 PDM with Four Stereo MIC 606
Fig. 18-4 PDM interface diagram with external MIC 606
Fig. 18-5 PDM Clock Structure 607
Fig. 18-6 PDM operation flow 616
Fig. 19-1 I2S/PCM controller (2 channel) Block Diagram 617
Fig. 19-2 I2S transmitter-master & receiver-slave condition 618
Fig. 19-3 I2S transmitter-slave& receiver-master condition 618
Fig. 19-4 I2S normal mode timing format 619
Fig. 19-5 I2S left justified mode timing format 619
Fig. 19-6 I2S right justified mode timing format 619
Fig. 19-7 PCM early mode timing format 619
Fig. 19-8 PCM late1 mode timing format 620
Fig. 19-9 PCM late2 mode timing format 620
Fig. 19-10 PCM late3 mode timing format 621
Fig. 19-11 I2S/PCM controller transmit operation flow chart 631
Fig. 19-12 I2S/PCM controller receive operation flow chart 632
Fig. 20-1 I2S/PCM/TDM controller (8 channel) Block Diagram 634
Fig. 20-2 I2S transmitter-master & receiver-slave condition 635
Fig. 20-3 I2S transmitter-slave & receiver-master condition 635
Fig. 20-4 I2S normal mode timing format 635
Fig. 20-5 I2S left justified mode timing format 635
Fig. 20-6 I2S right justified mode timing format 636
Fig. 20-7 PCM early mode timing format 636
Fig. 20-8 PCM late1 mode timing format 636
Fig. 20-9 PCM late2 mode timing format 637
Fig. 20-10 PCM late3 mode timing format 637
Fig. 20-11 I2S/PCM/TDM controller transmit operation flow chart 656
Fig. 20-12 I2S/PCM/TDM controller receive operation flow chart 657
Fig. 21-1 SPI2APBBlock Diagram 658
Fig. 21-2 Write operation 658
Fig. 21-3 Read operation 659
Fig. 21-4 Query operation 659
Fig. 21-5 Write message operation 660
Fig. 22-1 SPI Controller Block diagram 666
Fig. 22-2 SPI Master and Slave Interconnection 666
Fig. 22-3 SPI Format (SCPH=0 SCPOL=0) 667
Fig. 22-4 SPI Format (SCPH=0 SCPOL=1) 667
Fig. 22-5 SPI Format (SCPH=1 SCPOL=0) 668
Fig. 22-6 SPI Format (SCPH=1 SCPOL=1) 668
Fig. 22-7 SPI Master transfer flow diagram 680
Fig. 22-8 SPI Slave transfer flow diagram 681
Fig. 23-1 UART Architecture 682
Fig. 23-2 UART Serial protocol 683
Fig. 23-3 IrDA 1.0 683
Fig. 23-4 UART baud rate 684
Fig. 23-5 UART Auto flow control block diagram 685
Fig. 23-6 UART AUTO RTS TIMING 685
Fig. 23-7 UART AUTO CTS TIMING 685

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Fig. 23-8 UART none fifo mode 705
Fig. 23-9 UART fifo mode 706
Fig. 23-10 UART clock generation 707
Fig. 24-1 I2C architecture 709
Fig. 24-2 I2C DATA Validity 711
Fig. 24-3 I2C Start and stop conditions 712
Fig. 24-4 I2C Acknowledge 712
Fig. 24-5 I2C byte transfer 712
Fig. 24-6 I2C Flow chat for transmit only mode 724
Fig. 24-7 I2C Flow chat for receive only mode 725
Fig. 24-8 I2C Flow chat for mix mode 726
Fig. 25-1 GPIO block diagram 727
Fig. 25-2 GPIO Interrupt RTL Block Diagram 728
Fig. 26-1 Timer Block Diagram 735
Fig. 26-2 Timer Usage Flow 736
Fig. 26-3 Timing between timer_en and timer_clk 739
Fig. 27-1 PWM Block Diagram 740
Fig. 27-2 PWM Capture Mode 741
Fig. 27-3 PWM Continuous Left-aligned Output Mode 741
Fig. 27-4 PWM Continuous Center-aligned Output Mode 742
Fig. 27-5 PWM One-shot Center-aligned Output Mode 742
Fig. 28-1 WDT block diagram 763
Fig. 28-2 WDT Operation Flow 764
Fig. 29-1 SARADC block diagram 768
Fig. 29-2 SAR-ADC timing diagram in single-sample conversion mode 770
Fig. 30-1 TSADC Controller Block Diagram 771
Fig. 30-2 The start flow to enable the Sensor and ADC 776
Fig. 30-3 TSADC timing diagram in bypass mode 777
Fig. 30-4 TSADC timing diagram in normal mode with tsadc_clk_sel = 1'b0 777
Fig. 30-5 TSADC timing diagram in normal mode with tsadc_clk_sel = 1'b1 777

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Table Index

Table 1-1 RK1808 Address Mapping....16
Table 1-2 RK1808 remap function....17
Table 1-3 RK1808 Interrupt connection list....18
Table 1-4 RK1808 DMAC Hardware request connection list....21
Table 2-1 Source Clock Limitation of Fractional Divider....128
Table 3-1 SW-DP Interface Description....130
Table 4-1 GRF Address Mapping Table....131
Table 5-1 CPU Configuration....307
Table 6-1 DMAC Request Mapping Table....309
Table 6-2 DMAC boot interface....349
Table 6-3Source size in CCRn....355
Table 6-4 DMAC Instruction sets....355
Table 6-5 DMAC instruction encoding....356
Table 7-1 GIC-500 configuration....357
Table 8-1 Power Domain and Voltage Domain Summary....359
Table 8-2 Low Power State....393
Table 8-3 Low Power State....394
Table 9-1 Core_pvtm control source and result destination....395
Table 9-2 PMU_pvtm control source and result destination....396
Table 11-1 Bits in Interrupt Status Register....410
Table 11-2 Auto-Stop Generation....420
Table 11-3 Non-data Transfer Commands and Requirements....422
Table 11-4 Bits in IDMAC DES0 Element....426
Table 11-5 Bits in IDMAC DES1 Element....427
Table 11-6 Bits in IDMAC DES2 Element....427
Table 11-7 Bits in IDMAC DES3 Element....428
Table 11-8 SDMMC Interface Description....456
Table 11-9 SDIO Interface Description....456
Table 11-10 EMMC Interface Description....456
Table 11-11 Command Settings for No-Data Command....461
Table 11-12 Command Setting for Single or Multiple-Block Read....463
Table 11-13 Command Settings for Single or Multiple-Block Write....464
Table 11-14 PBL and Watermark Levels....473
Table 11-15 Configuration for SDMMC Clock Generation....475
Table 11-16 Configuration for SDIO Clock Generation....475
Table 11-17 Configuration for EMMC Clock Generation....476
Table 11-18 Register for SDMMC Card Detection Method 3....477
Table 12-1 SFC interface description....488
Table 15-1 NPU Address Mapping....497
Table 16-1 VIP Interface Description....528
Table 17-1 RGMII Interface Description....590
Table 17-2 Receive Descriptor 0....592
Table 17-3 Receive Descriptor 1....594
Table 17-4 Receive Descriptor 2....594
Table 17-5 Receive Descriptor 3....594
Table 17-6 Transmit Descriptor 0....595
Table 17-7 Transmit Descriptor 1....596
Table 17-8 Transmit Descriptor 2....597
Table 17-9 Transmit Descriptor 3....598
Table 18-1 Relation between PDM_CLK and sample rate....607
Table 18-2 PDM Interface Description....615
Table 19-1 I2S_2CH Interface Description....630
Table 20-1 I2S_8CH_TDM Interface Description....654
Table 21-1 SPI2APB interface description....664
Table 22-1 SPI1/SPI2 interface description....678

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Table 23-1 UART Interface Description 703
Table 23-2 UART baud rate configuration 707
Table 24-1 I2C Interface Description 722
Table 25-1 GPIO interface description 732
Table 26-1 Register Base Address 738
Table 27-1 PWM Interface Description 760
Table 30-1 Temperature Code Mapping 777

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

NOTICE

Copyright © 2019, Fuzhou Rockchip Electronics Co., Ltd. All rights reserved.

1. By using this document, you hereby unequivocally acknowledge that you have read and agreed to be bound by the contents of this notice.

2. Fuzhou Rockchip Electronics Co., Ltd. ("Rockchip") may make changes to any information in this document at any time without any prior notice. The information herein is subject to change without notice. Do not finalize a design with this information.

3. Information in this document is provided in connection with Rockchip products.

4. THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT ANY WARRANTY OR CONDITION OF ANY KIND, EITHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING, WITHOUT LIMITATION, ANY WARRANTY OR CONDITION WITH RESPECT TO MERCHANTABILITY, FITNESS FOR ANY PARTICULAR PURPOSE, OR NON-INFRINGEMENT.ROCKCHIP DOES NOT ASSUME ANY RESPONSIBILITY AND LIABILITY FOR ITS USE NOR FOR ANY INFRINGEMENT OF PATENTS OR OTHER RIGHTS OF THE THIRD PARTIES WHICH MAY RESULT FROM ITS USE.

5. Rockchip products described in this document are not designed, intended for use in medical, life saving, life sustaining, critical control or safety systems, or in nuclear facility application.

6. Rockchip and Rockchip logo are trademarks or registered trademarks of Rockchip in China and other countries. All referenced brands, product names, service names and trademarks in this document are the property by their respective owners.

Chapter 1 System Overview

1.1 Address Mapping

RK1808 boot from internal BootRom, which supports remap function by software programming. Remap is controlled by PMU_SGRF_SOC_CON0[13]. When remap is set to 2'b01, the BootRom is un-accessible and PMU_SRAM is mapped to address 0xFFFF0000. When remap is set to 2'b10, the BootRom is un-accessible and SYSTEM_SRAM is mapped to address 0xFFFF0000.

Module	Start Address	Size	Module	Start Address	Size
PCIE_S	0xF8000000	64MB	I2C1	0XFF500000	16KB
PCIE_DBI	0xFC000000	4MB	I2C2	0XFF504000	16KB
PCIE_APB	0xFC400000	64KB	I2C3	0XFF508000	16KB
RESERVED	0xFC410000	11.9375MB	I2C4	0XFF50C000	16KB
USB3	0xFD000000	2MB	I2C5	0XFF510000	16KB
RESERVED	0xFD200000	4MB	RESERVED	0XFF514000	0.046875MB
RESERVED	0xFD600000	10MB	SPI0(SPI2APB)	0XFF520000	64KB
BUS_GRF	0XFE000000	64KB	SPI1	0XFF530000	64KB
USB2PHY_GRF	0XFE010000	32KB	UART1	0XFF540000	64KB
USB3PHY_GRF	0XFE018000	32KB	UART2	0XFF550000	64KB
PMU_GRF	0XFE020000	64KB	UART3	0XFF560000	64KB
DDR_GRF	0XFE030000	64KB	UART4	0XFF570000	64KB
USB_GRF	0XFE040000	64KB	SPI2	0XFF580000	64KB
CORE_GRF	0XFE050000	64KB	FIREWALL_APB	0XFF590000	64KB
RESERVED	0XFE060000	3.625MB	UART5	0XFF5A0000	64KB
BUS_SGRF	0XFE400000	64KB	UART6	0XFF5B0000	64KB
PMU_SGRF	0XFE410000	64KB	UART7	0XFF5C0000	64KB
RESERVED	0XFE420000	3.875MB	PWM2	0XFF5D0000	64KB
SERVICE_MSCH	0XFE800000	64KB	RESERVED	0XFF5E0000	64KB
SERVICE_AUDIO	0XFE810000	64KB	RESERVED	0XFF5F0000	256KB
SERVICE_BUS	0XFE820000	64KB	CRYPTO	0XFF630000	64KB
SERVICE_CPU	0XFE830000	64KB	DCF	0XFF640000	64KB
SERVICE_GIC	0XFE840000	64KB	RESERVED	0XFF650000	256KB
SERVICE_NPU	0XFE850000	64KB	GPIO1	0XFF690000	64KB
SERVICE_GMAC	0XFE860000	64KB	GPIO2	0XFF6A0000	64KB
SERVICE_MMC	0XFE870000	64KB	GPIO3	0XFF6B0000	64KB
SERVICE_PCIE	0XFE880000	64KB	GPIO4	0XFF6C0000	64KB
SERVICE_USB	0XFE890000	64KB	RESERVED	0XFF6D0000	192KB
SERVICE_VIO_0	0XFE8A0000	64KB	TIMER0-5	0XFF700000	64KB
SERVICE_VIO_1	0XFE8B0000	64KB	STIMER0-1	0XFF710000	64KB
SERVICE_VPU	0XFE8C0000	64KB	WDTNS	0XFF720000	64KB
SERVICE_DMA_CRPT	0XFE8D0000	64KB	WDTS	0XFF730000	64KB
RESERVED	0XFE8E0000	3.125MB	RESERVED	0XFF740000	0.625MB
SYSTEM_SRAM	0XFEC00000	2MB	I2S0	0XFF7E0000	64KB
RESERVED	0XFEE00000	2MB	I2S1	0XFF7F0000	64KB

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Module	Start Address	Size	Module	Start Address	Size
RESERVED	0XFF000000	0KB	PDM	0XFF800000	64KB
DEBUG	0XFF000000	256KB	VAD	0XFF810000	64KB
SPINLOCK	0XFF040000	64KB	RESERVED	0XFF820000	512KB
RESERVED	0XFF050000	0.6875MB	RESERVED	0XFF8A0000	512KB
GIC500	0XFF100000	2MB	RESERVED	0XFF920000	512KB
CPU_GIC_INTERFACE	0XFF300000	256KB	RESERVED	0XFF9A0000	64KB
DDR_PHY	0XFF340000	64KB	DDR_STDBY	0XFF9B0000	32KB
CRU	0XFF350000	16KB	DDR_DFICTRL	0XFF9B8000	32KB
PMU_CRU	0XFF354000	16KB	DDR_MONITOR	0XFF9C0000	32KB
RESERVED	0XFF358000	32KB	RESERVED	0XFF9C8000	32KB
CSI_PHY	0XFF360000	64KB	RESERVED	0XFF9D0000	512KB
DSI_PHY	0XFF370000	64KB	UPCTL	0XFFA50000	64KB
USB3_PHY	0XFF380000	64KB	RESERVED	0XFFA60000	512KB
RESERVED	0XFF390000	0KB	CIF	0XFFAE0000	64KB
RESERVED	0XFF390000	64KB	RGA	0XFFAF0000	64KB
TSADC	0XFF3A0000	64KB	VOP_LITE	0XFFB00000	64KB
EFUSE_NS	0XFF3B0000	32KB	CSI HOST	0XFFB10000	64KB
EFUSE_S	0XFF3B8000	32KB	CSI2TX	0XFFB20000	64KB
SARADC	0XFF3C0000	64KB	DSI HOST	0XFFB30000	64KB
PWM0	0XFF3D0000	32KB	VOP_RAW	0XFFB40000	64KB
PWM1	0XFF3D8000	32KB	ISP	0XFFB50000	64KB
PMU	0XFF3E0000	64KB	RESERVED	0XFFB60000	128KB
RESERVED	0XFF3F0000	64KB	VPU	0XFFB80000	64KB
RESERVED	0XFF400000	64KB	RESERVED	0XFFB90000	192KB
I2C0	0XFF410000	64KB	NPU	0XFFBC0000	512KB
RESERVED	0XFF420000	64KB	RESERVED	0XFFC40000	64KB
UART0	0XFF430000	64KB	SFC	0XFFC50000	64KB
RESERVED	0XFF440000	448KB	SDIO	0XFFC60000	64KB
PMU_SRAM	0XFF4B0000	64KB	RESERVED	0XFFC70000	512KB
GPIO0	0XFF4C0000	64KB	SDMMC	0XFFCF0000	64KB
RESERVED	0XFF4D0000	64KB	EMMC	0XFFD00000	64KB
DMAC	0XFF4E0000	64KB	RESERVED	0XFFD10000	192KB
SDMAC	0XFF4F0000	64KB	RESERVED	0XFFD40000	256KB
			USB2.0_HOST	0XFFD80000	320KB
			GMAC	0XFFDD0000	64KB

Table 1-1 RK1808 Address Mapping

Notes:
PMU_SRAM only has 8KB memory size
Service_xxx are interconnect register. Refer to chapter interconnect for more details.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

The following table show the boot address when before remap and after remap

remap[1]=0 remap[0]=0			remap[1]=0 remap[0]=1			remap[1]=1 remap[0]=0	
			Not accessible	BootRom 64K		Not accessible	BootRom 64K
0xFFFF0000	BootRom 64K		0xFFFF0000	PMU_SRAM 64K		0xFF4B0000	PMU_SRAM 64K
0xFF4B0000	PMU_SRAM 64K		0xFF4B0000	PMU_SRAM 64K		0xFFFF0000	SYSTEM_SRAM 64K

Table 1-2 RK1808 remap function

1.2 System Boot

RK1808 provides system boot from off-chip devices such as SDMMC card, eMMC memory, serial nand or nor flash. When boot code is not ready in these devices, also provide system code download into them by USB OTG interface. Besides of the above boot ways, RK1808 provide download boot code through SPI0, which follows the SPI's slave protocol and has an APB master interface that can access SYSTEM_SRAM. All of the boot code will be stored in internal BootRom. The following is the whole boot procedure for boot code, which will be stored in BootRom in advance.

The following features are supported.

- Support system boot from the following device:
  - Serial Nor Flash, 1bit data width
  - eMMC Interface, 8bits data width
  - SDMMC Card, 4bits data width
- Support system code download by USB OTG
- Support system code download by SPI0 module

Following figure shows RK1808 boot procedure flow.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

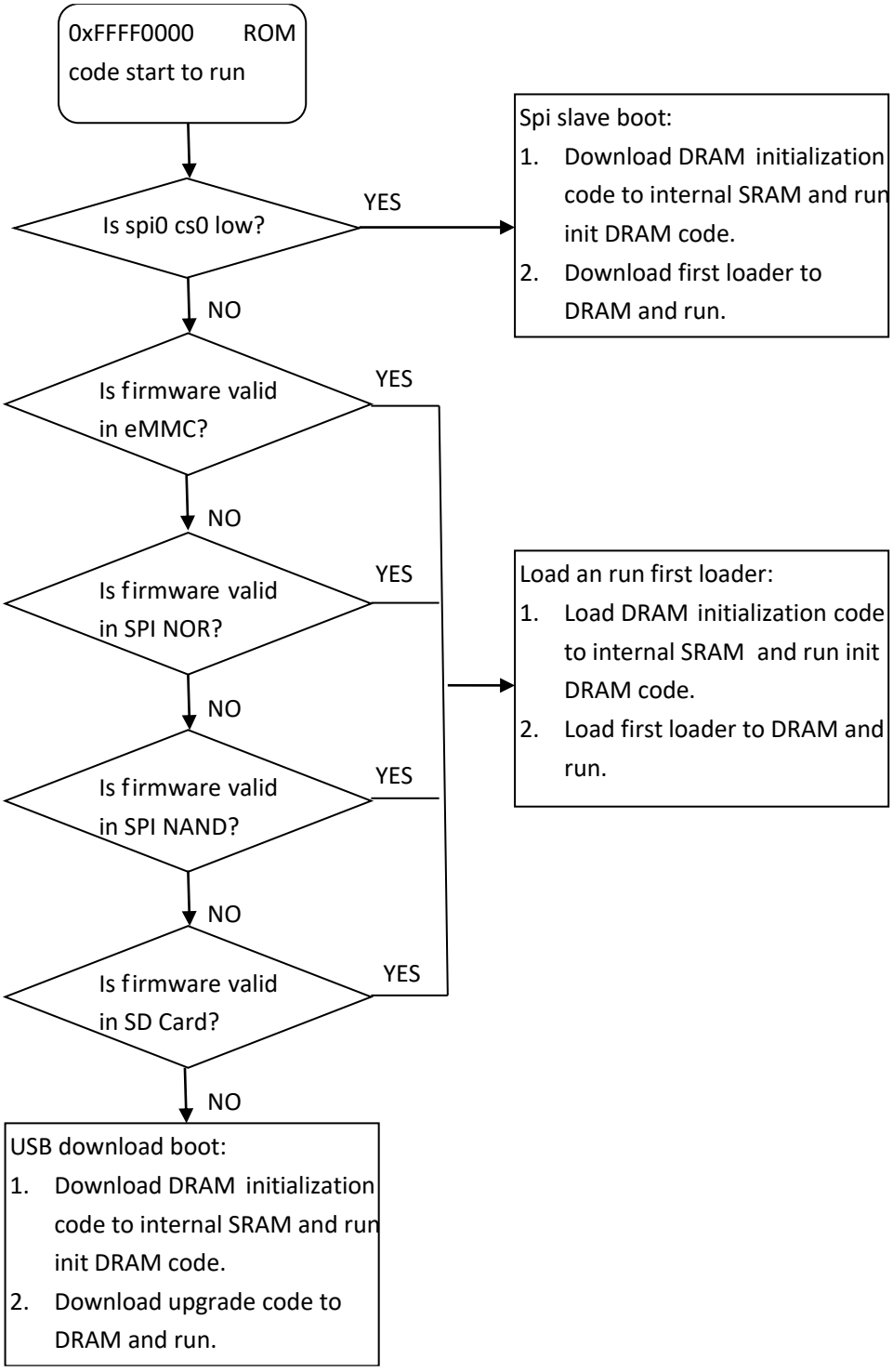


Fig. 1-1 RK1808 boot procedure flow

1.3 System Interrupt connection

RK1808 provides an general interrupt controller(GIC) for CPU, which has 256 SPI (shared peripheral interrupts) interrupt sources and 3 PPI(Private peripheral interrupt) interrupt source and separately generates one nIRQ and one nFIQ to CPU. The triggered type for each interrupts is high level sensitive, not programmable. The detailed interrupt sources connection is in the following table. For detailed GIC setting, please refer to Chapter 9.

Table 1-3 RK1808 Interrupt connection list

Number	Source	Polarity
0-31PPI		High level

Number	Source	Polarity
32	dcf_int	High level
33	dmac_irq	High level
34	dmac_irq_abort	High level
35	gpio0_int	High level
36	gpio1_int	High level
37	gpio2_int	High level
38	gpio3_int	High level
39	i2c_irq_i2c0	High level
40	i2c_irq_i2c1	High level
41	i2c_irq_i2c2	High level
42	i2s_intr_i2s08ch	High level
43	i2s_intr_i2s12ch	High level
44	pdm_irq	High level
45	uart0_intr	High level
46	uart1_intr	High level
47	uart2_intr	High level
48	uart3_intr	High level
49	uart4_intr	High level
50	efusens_int	High level
51	pwm_int_pwr_pwm0	High level
52	pwm_int_pwm0	High level
53	spi_intr_spi0	High level
54	spi_intr_spi1	High level
55	spi_intr_spi2	High level
56	timer0_int_stimer	High level
57	timer1_int_stimer	High level
58	timer0_int_rktimer	High level
59	timer1_int_rktimer	High level
60	timer2_int_rktimer	High level
61	timer3_int_rktimer	High level
62	timer4_int_rktimer	High level
63	timer5_int_rktimer	High level
64	tsadc_int_tsadc	High level
65	wdt_intr_wdtns	High level
66	wdt_intr_wdts	High level
67	upctl_arpoison_int	High level
68	upctl_awpoison_int	High level
69	upctl_alert_err_int	High level
70	ddrmon_int	High level
71	gmac_intr_gmac2io	High level
72	pmt_intr_gmac2io	High level
73	msch_alarm_irq	High level
74	efuses_int	High level
75	npu_int	High level

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Number	Source	Polarity
76	uart7_intr	High level
77	uart6_intr	High level
78	uart5_intr	High level
79	sdmmc_int_emmc	High level
80	sdmmc_int_sdmmc	High level
81	sdmmc_int_sdio	High level
82	sfc_int_sfc	High level
83	pmu_int	High level
84	host_arb_int_usb2host	High level
85	host_ehci_int_usb2host	High level
86	host_ohci_int_usb2host	High level
87	cru_hwffc_int	High level
88	cif_int_out_cif	High level
89	interrupt_csitx	High level
90	cpu_slv_err_obsrv_mainFault_0	High level
91	mipi_dsi_host_irq_dsihost	High level
92	rga_irq	High level
93	vop_intr_vopl	High level
94	vop_intr_vopraw	High level
95	crypto_irq	High level
96	saradc_irq	High level
97	sdmmc_detectn_irq_grf	High level
98	sdcard_dectn_in_flt	High level
99	global_int_disable	High level
100	global_int_disable_0	High level
101	global_int_disable_1	High level
102	global_int_disable_2	High level
103	global_int_disable_3	High level
104	int_vad_vad	High level
105	perf_int_ca35	High level
106	usb3otg_host_legacy_smi_interrupt_usb3otg	High level
107	usb3otg_int_usb3otg	High level
108	usbphy_otg_linestate_irq	High level
109	usbphy_otg_id_irq	High level
110	usbphy_otg_bvalid_irq	High level
111	usbphy_host_linestate_irq	High level
112	usbphy_otg_disconnect_irq	High level
113	usbphy_host_disconnect_irq	High level
114	gpio_intr_flag_gpio4	High level
115	i2c_irq_i2c3	High level
116	pwm_int_pwr_pwm1	High level
117	pwm_int_pwm1	High level
118	pcie_sys_int	High level
119	pcie_legacy_int	High level

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Number	Source	Polarity
120	pcie_msg_rx_int	High level
121	pcie_err_int	High level
122	pwm_int_pwr_pwm2	High level
123	pwm_int_pwm2	High level
124	vpu_mmu_irq	High level
125	vpu_enc_irq	High level
126	vpu_dec_irq	High level
127	mipi_irq_isp	High level
128	mi_irq_isp	High level
129	isp_irq	High level
130	isp_mmu_2_irq	High level
131	usb3otg_pme_generation_irq	High level
132	usb3otg_host_sys_err_irq	High level
133	i2c4_irq	High level
134	i2c5_irq	High level
135	pciephy_lfps_beacon_irq	High level
136	pcie_pmc_int	High level
137	intr2_csirx	High level
138	intr1_csirx	High level
260	a35_pmuirq_0	High level
261	a35_pmuirq_1	High level
264	a35_vcpumntirq_0	High level
265	a35_vcpumntirq_1	High level
269	a35_comirq_0	High level
270	a35_comirq_1	High level
273	a35_axierrirq	High level

1.4 System DMA hardware request connection

RK1808 provides one DMA controller(DMAC) inside the system. For detailed descriptions of DMAC, please refer to Chapter 8.

Table 1-4 RK1808 DMAC Hardware request connection list

Request Index	Source	Polarity
0	Uart0 tx	High level
1	Uart0 rx	High level
2	Uart1 tx	High level
3	Uart1 rx	High level
4	Uart2 tx	High level
5	Uart2 rx	High level
6	Uart3 tx	High level
7	Uart3 rx	High level
8	Uart4 tx	High level
9	Uart4 rx	High level
10	SPI0 tx	High level
11	SPI0 rx	High level

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Request Index	Source	Polarity
12	SPI1 tx	High level
13	SPI1 rx	High level
14	SPI2 tx	High level
15	SPI2 rx	High level
16	I2S0_8ch tx	High level
17	I2S0_8ch rx	High level
18	I2S1_2ch_tx	High level
19	I2S1_2ch_rx	High level
21	PWM0	High level
22	PWM1	High level
23	PWM2	High level
24	PDM	High level
25	Uart5 tx	High level
26	Uart5 rx	High level
27	Uart6 tx	High level
28	Uart6 rx	High level
29	Uart7 tx	High level
30	Uart7 rx	High level

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 2 Clock & Reset Unit (CRU)

2.1 Overview

The CRU is an APB slave module that is designed for generating all of the internal and system clocks, resets of chip. CRU generates system clocks from PLL output clock or external clock source, and generates system reset from external power-on-reset, watchdog timer reset or software reset or temperature sensor.
CRU supports the following features:

- Compliance with AMBA APB interface
- Embedded with 6 PLLs
- Flexible selection of clock source
- Support dividing clock separately
- Support gating clock separately
- Support software reset each module separately

2.2 Block Diagram

CRU comprises with:

- PLL
- Register configuration unit
- Clock generate unit
- Reset generate unit

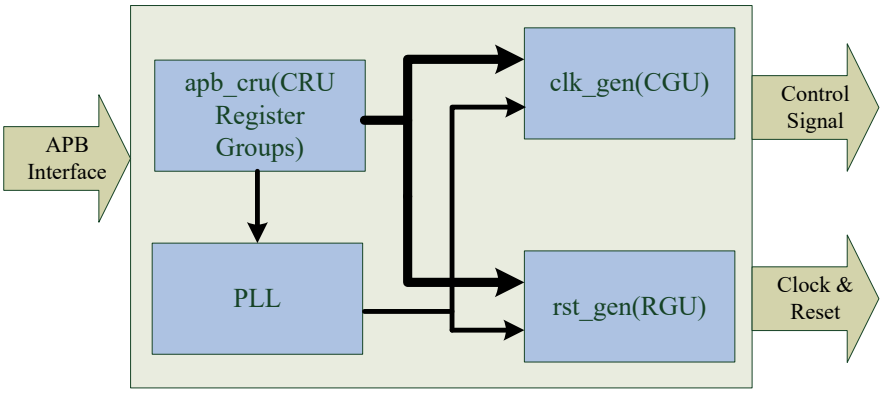


Fig. 2-1 CRU Block Diagram

2.3 System Reset Solution

The following diagram shows reset architecture.

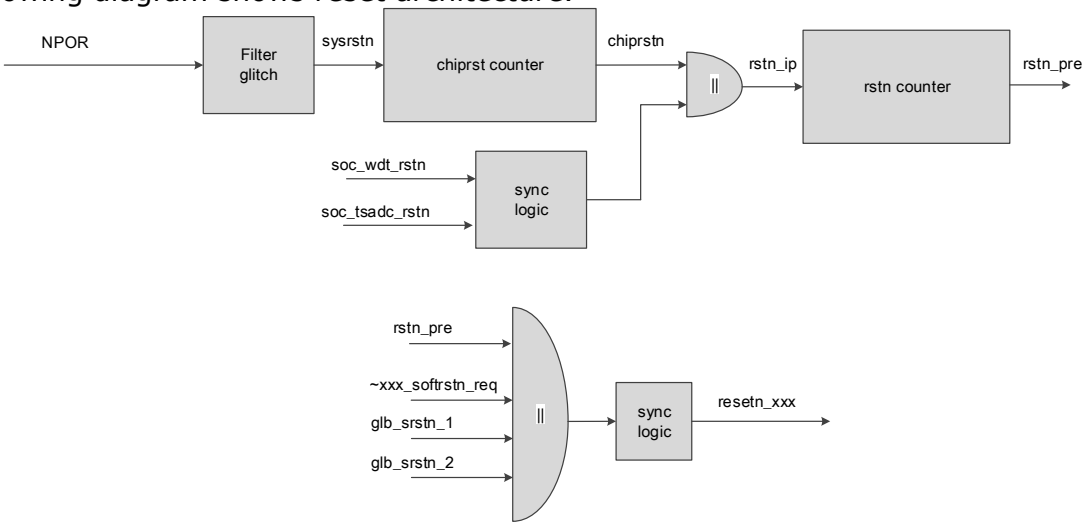


Fig. 2-2 Reset Architecture Diagram

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Reset source of each reset signal includes:

- NPOR: external power on reset
- soc_wdt_rstn: reset from watch-dog module
- soc_tsadc_rstn: reset from tsadc module
- xxx_softrstn_req: software reset request (xxx represents module name)
- glb_srstn_1: global software first reset
- glb_srstn_2: global software second reset

glb_srstn_1 and glb_srstn_2 are the global software reset by programming CRU register. When setting register CRU_GLB_SRST_FST as 0xfdb9, glb_srstn_1 will be asserted, and when programming registerCRU_GLB_SRST_SND as 0xeca8, glb_srstn_2 will be asserted. These two kinds of software resets will be self-cleared by hardware. glb_srstn_1 will reset all the logic, and glb_srstn_2 will reset all the logic except GRF, SGRF and all GPIOs.

2.4 Function Description

There are 6 PLLs in RK1808: ARM PLL, NEW PLL, DDR PLL, CODEC PLL, GENERAL PLL and PMU PLL, and each PLL can only receive 24MHz oscillator as input reference clock.
Each PLL can be set to three work modes: normal mode, slow mode and deep slow mode. When power on or changing PLL setting, we must program PLL into slow mode or deep slow mode.
To maximize the flexibility, some of clocks can select divider source from multiple PLLs.
To provide some specific frequency, another solution is integrated: fractional divider.
Divfree50 divider and divfreeNP5 divider are also provided for some modules.
All clocks can be gated and all resets can be generated by software.

2.5 PLL Introduction

2.5.1 Overview

The PLLs inside RK1808 output clock's frequency up to 5GHz. The 5GHz PLL is a general purpose, high-performance PLL-based clock generator. The PLL is a multi-function, general purpose frequency synthesizer. Ultra-wide input and output ranges along with best-in-class jitter performance allow the PLL to be used for almost any clocking application. With excellent supply noise immunity, the PLL is ideal for use in noisy mixed signal SoC environments.
5GHz PLL supports the following features:

- Input frequency range: 2MHz to 1250MHz(Integer Mode) and 10MHz to 1250MHz (Fractional Mode)
- Output Frequency Range:25MHz to 5GHz
- VCO output clock from 1250MHz to 5GHz
- 24 bit fractional accuracy, and fractional mode jitter performance to nearly match integer mode performance.
- 4:1 VCO frequency range allows PLL to be optimized for minimum jitter or minimum power.
- Isolated analog supply (1.8V) allows for excellent supply rejection in noisy SoC applications.
- Lock Detect Signal indicates when frequency lock has been achieved.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

2.5.2 Block diagram

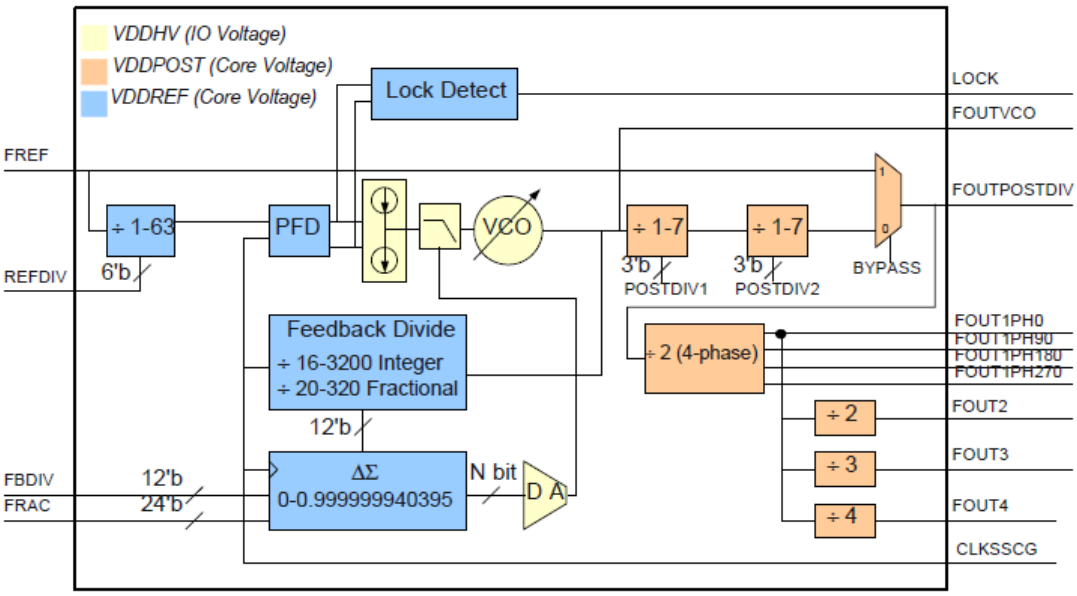


Fig. 2-3 PLL Block Diagram

2.6 Register Description

2.6.1 Internal Address Mapping

Slave address can be divided into different length for different usage, which is shown as follows.

2.6.2 Registers Summary

Name	Offset	Size	Reset Value	Description
CRU_APLL_CON0	0x0000	W	0x000020fa	APLL configuration register0
CRU_APLL_CON1	0x0004	W	0x00001043	APLL configuration register1
CRU_APLL_CON2	0x0008	W	0x00000001	APLL configuration register2
CRU_APLL_CON3	0x000c	W	0x00000007	APLL configuration register3
CRU_APLL_CON4	0x0010	W	0x00007f00	APLL configuration register4
CRU_DPLL_CON0	0x0020	W	0x000010c8	DPLL configuration register0
CRU_DPLL_CON1	0x0024	W	0x00001043	DPLL configuration register1
CRU_DPLL_CON2	0x0028	W	0x00000001	DPLL configuration register2
CRU_DPLL_CON3	0x002c	W	0x00000007	DPLL configuration register3
CRU_DPLL_CON4	0x0030	W	0x00007f00	DPLL configuration register4
CRU_CPLL_CON0	0x0040	W	0x000020fa	CPLL configuration register0
CRU_CPLL_CON1	0x0044	W	0x00001043	CPLL configuration register1
CRU_CPLL_CON2	0x0048	W	0x00000001	CPLL configuration register2
CRU_CPLL_CON3	0x004c	W	0x00000007	CPLL configuration register3
CRU_CPLL_CON4	0x0050	W	0x00007f00	CPLL configuration register4
CRU_GPLL_CON0	0x0060	W	0x00002064	GPLL configuration register0
CRU_GPLL_CON1	0x0064	W	0x00001041	GPLL configuration register1
CRU_GPLL_CON2	0x0068	W	0x00000001	GPLL configuration register2
CRU_GPLL_CON3	0x006c	W	0x00000007	GPLL configuration register3
CRU_GPLL_CON4	0x0070	W	0x00007f00	GPLL configuration register4
CRU_NPLL_CON0	0x0080	W	0x000020c8	NPLL configuration register0

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
CRU_NPLL_CON1	0x0084	W	0x00001043	NPLL configuration register1
CRU_NPLL_CON2	0x0088	W	0x00000001	NPLL configuration register2
CRU_NPLL_CON3	0x008c	W	0x00000007	NPLL configuration register3
CRU_NPLL_CON4	0x0090	W	0x00007f00	NPLL configuration register4
CRU_MODE	0x00a0	W	0x00000000	MODE
CRU_MISC	0x00a4	W	0x00000000	MISC
CRU_MISC1	0x00a8	W	0x00000000	MISC1
CRU_GLB_CNT_TH	0x00b0	W	0x3a980064	GLB_CNT_TH
CRU_GLB_RST_ST	0x00b4	W	0x00000000	GLB_RST_ST
CRU_GLB_SRST_FST	0x00b8	W	0x00000000	GLB_SRST_FST
CRU_GLB_SRST_SND	0x00bc	W	0x00000000	GLB_SRST_SND
CRU_GLB_RST_CON	0x00c0	W	0x00000000	GLB_RST_CON
CRU_CLKSEL_CON0	0x0100	W	0x00001300	Clock select and divide register0
CRU_CLKSEL_CON1	0x0104	W	0x00000011	Clock select and divide register1
CRU_CLKSEL_CON2	0x0108	W	0x00000711	Clock select and divide register2
CRU_CLKSEL_CON3	0x010c	W	0x00000b03	Clock select and divide register3
CRU_CLKSEL_CON4	0x0110	W	0x00000103	Clock select and divide register4
CRU_CLKSEL_CON5	0x0114	W	0x00000403	Clock select and divide register5
CRU_CLKSEL_CON6	0x0118	W	0x0bb8ea60	Clock select and divide register6
CRU_CLKSEL_CON7	0x011c	W	0x00000407	Clock select and divide register7
CRU_CLKSEL_CON8	0x0120	W	0x0bb8ea60	Clock select and divide register8
CRU_CLKSEL_CON9	0x0124	W	0x0000003c	Clock select and divide register9
CRU_CLKSEL_CON10	0x0128	W	0x00000203	Clock select and divide register10
CRU_CLKSEL_CON11	0x012c	W	0x00000202	Clock select and divide register11
CRU_CLKSEL_CON12	0x0130	W	0x00000503	Clock select and divide register12
CRU_CLKSEL_CON13	0x0134	W	0x00000000	Clock select and divide register13
CRU_CLKSEL_CON14	0x0138	W	0x00000063	Clock select and divide register14
CRU_CLKSEL_CON15	0x013c	W	0x00003023	Clock select and divide register15
CRU_CLKSEL_CON16	0x0140	W	0x00000103	Clock select and divide register16
CRU_CLKSEL_CON17	0x0144	W	0x00000303	Clock select and divide register17
CRU_CLKSEL_CON18	0x0148	W	0x00001305	Clock select and divide register18
CRU_CLKSEL_CON19	0x014c	W	0x00000705	Clock select and divide register19

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
CRU_CLKSEL_CON20	0x0150	W	0x00000007	Clock select and divide register20
CRU_CLKSEL_CON21	0x0154	W	0x00000007	Clock select and divide register21
CRU_CLKSEL_CON22	0x0158	W	0x00000007	Clock select and divide register22
CRU_CLKSEL_CON23	0x015c	W	0x00000007	Clock select and divide register23
CRU_CLKSEL_CON24	0x0160	W	0x00000007	Clock select and divide register24
CRU_CLKSEL_CON25	0x0164	W	0x00000007	Clock select and divide register25
CRU_CLKSEL_CON26	0x0168	W	0x0000070b	Clock select and divide register26
CRU_CLKSEL_CON27	0x016c	W	0x00000302	Clock select and divide register27
CRU_CLKSEL_CON28	0x0170	W	0x00000b05	Clock select and divide register28
CRU_CLKSEL_CON29	0x0174	W	0x00000305	Clock select and divide register29
CRU_CLKSEL_CON30	0x0178	W	0x0000000b	Clock select and divide register30
CRU_CLKSEL_CON31	0x017c	W	0x0bb8ea60	Clock select and divide register31
CRU_CLKSEL_CON32	0x0180	W	0x0000000b	Clock select and divide register32
CRU_CLKSEL_CON33	0x0184	W	0x0bb8ea60	Clock select and divide register33
CRU_CLKSEL_CON34	0x0188	W	0x0000000b	Clock select and divide register34
CRU_CLKSEL_CON35	0x018c	W	0x0bb8ea60	Clock select and divide register35
CRU_CLKSEL_CON36	0x0190	W	0x0000000b	Clock select and divide register36
CRU_CLKSEL_CON37	0x0194	W	0x0bb8ea60	Clock select and divide register37
CRU_CLKSEL_CON38	0x0198	W	0x0000000b	Clock select and divide register38
CRU_CLKSEL_CON39	0x019c	W	0x0000c00b	Clock select and divide register39
CRU_CLKSEL_CON40	0x01a0	W	0x0bb8ea60	Clock select and divide register40
CRU_CLKSEL_CON41	0x01a4	W	0x0000000b	Clock select and divide register41

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
CRU_CLKSEL_CON42	0x01a8	W	0x0000c00b	Clock select and divide register42
CRU_CLKSEL_CON43	0x01ac	W	0x0bb8ea60	Clock select and divide register43
CRU_CLKSEL_CON44	0x01b0	W	0x0000000b	Clock select and divide register44
CRU_CLKSEL_CON45	0x01b4	W	0x0000c00b	Clock select and divide register45
CRU_CLKSEL_CON46	0x01b8	W	0x0bb8ea60	Clock select and divide register46
CRU_CLKSEL_CON47	0x01bc	W	0x0000000b	Clock select and divide register47
CRU_CLKSEL_CON48	0x01c0	W	0x0000c00b	Clock select and divide register48
CRU_CLKSEL_CON49	0x01c4	W	0x0bb8ea60	Clock select and divide register49
CRU_CLKSEL_CON50	0x01c8	W	0x0000000b	Clock select and divide register50
CRU_CLKSEL_CON51	0x01cc	W	0x0000c00b	Clock select and divide register51
CRU_CLKSEL_CON52	0x01d0	W	0x0bb8ea60	Clock select and divide register52
CRU_CLKSEL_CON53	0x01d4	W	0x0000000b	Clock select and divide register53
CRU_CLKSEL_CON54	0x01d8	W	0x0000c00b	Clock select and divide register54
CRU_CLKSEL_CON55	0x01dc	W	0x0bb8ea60	Clock select and divide register55
CRU_CLKSEL_CON56	0x01e0	W	0x0000000b	Clock select and divide register56
CRU_CLKSEL_CON57	0x01e4	W	0x0000c00b	Clock select and divide register57
CRU_CLKSEL_CON58	0x01e8	W	0x0bb8ea60	Clock select and divide register58
CRU_CLKSEL_CON59	0x01ec	W	0x00000505	Clock select and divide register59
CRU_CLKSEL_CON60	0x01f0	W	0x00000305	Clock select and divide register60
CRU_CLKSEL_CON61	0x01f4	W	0x00000b0b	Clock select and divide register61
CRU_CLKSEL_CON62	0x01f8	W	0x0000001f	Clock select and divide register62
CRU_CLKSEL_CON63	0x01fc	W	0x00000017	Clock select and divide register63

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
CRU_CLKSEL_CON64	0x0200	W	0x00000b0b	Clock select and divide register64
CRU_CLKSEL_CON65	0x0204	W	0x00000001	Clock select and divide register65
CRU_CLKSEL_CON66	0x0208	W	0x00000001	Clock select and divide register66
CRU_CLKSEL_CON67	0x020c	W	0x00000001	Clock select and divide register67
CRU_CLKSEL_CON68	0x0210	W	0x00000001	Clock select and divide register68
CRU_CLKSEL_CON69	0x0214	W	0x00000b0b	Clock select and divide register69
CRU_CLKSEL_CON70	0x0218	W	0x0000000b	Clock select and divide register70
CRU_CLKSEL_CON71	0x021c	W	0x00000505	Clock select and divide register71
CRU_CLKSEL_CON72	0x0220	W	0x00001f00	Clock select and divide register72
CRU_CLKGATE_CON0	0x0230	W	0x00000000	Clock gating register0
CRU_CLKGATE_CON1	0x0234	W	0x00000000	Clock gating register1
CRU_CLKGATE_CON2	0x0238	W	0x00000000	Clock gating register2
CRU_CLKGATE_CON3	0x023c	W	0x00000000	Clock gating register3
CRU_CLKGATE_CON4	0x0240	W	0x00000000	Clock gating register4
CRU_CLKGATE_CON5	0x0244	W	0x00000000	Clock gating register5
CRU_CLKGATE_CON6	0x0248	W	0x00000000	Clock gating register6
CRU_CLKGATE_CON7	0x024c	W	0x00000000	Clock gating register7
CRU_CLKGATE_CON8	0x0250	W	0x00000000	Clock gating register8
CRU_CLKGATE_CON9	0x0254	W	0x00000000	Clock gating register9
CRU_CLKGATE_CON10	0x0258	W	0x00000000	Clock gating register10
CRU_CLKGATE_CON11	0x025c	W	0x00000000	Clock gating register11
CRU_CLKGATE_CON12	0x0260	W	0x00000000	Clock gating register12
CRU_CLKGATE_CON13	0x0264	W	0x00000000	Clock gating register13
CRU_CLKGATE_CON14	0x0268	W	0x00000000	Clock gating register14
CRU_CLKGATE_CON15	0x026c	W	0x00000000	Clock gating register15
CRU_CLKGATE_CON16	0x0270	W	0x00000000	Clock gating register16
CRU_CLKGATE_CON17	0x0274	W	0x00000000	Clock gating register17
CRU_CLKGATE_CON18	0x0278	W	0x00000000	Clock gating register18
CRU_CLKGATE_CON19	0x027c	W	0x00000000	Clock gating register19
CRU_SSGTBL0_3	0x0280	W	0x00000000	External wave table register0
CRU_SSGTBL4_7	0x0284	W	0x00000000	External wave table register1
CRU_SSGTBL8_11	0x0288	W	0x00000000	External wave table register2
CRU_SSGTBL12_15	0x028c	W	0x00000000	External wave table register3
CRU_SSGTBL16_19	0x0290	W	0x00000000	External wave table register4

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
CRU_SSGTBL20_23	0x0294	W	0x00000000	External wave table register5
CRU_SSGTBL24_27	0x0298	W	0x00000000	External wave table register6
CRU_SSGTBL28_31	0x029c	W	0x00000000	External wave table register7
CRU_SSGTBL32_35	0x02a0	W	0x00000000	External wave table register8
CRU_SSGTBL36_39	0x02a4	W	0x00000000	External wave table register9
CRU_SSGTBL40_43	0x02a8	W	0x00000000	External wave table register10
CRU_SSGTBL44_47	0x02ac	W	0x00000000	External wave table register11
CRU_SSGTBL48_51	0x02b0	W	0x00000000	External wave table register12
CRU_SSGTBL52_55	0x02b4	W	0x00000000	External wave table register13
CRU_SSGTBL56_59	0x02b8	W	0x00000000	External wave table register14
CRU_SSGTBL60_63	0x02bc	W	0x00000000	External wave table register15
CRU_SSGTBL64_67	0x02c0	W	0x00000000	External wave table register16
CRU_SSGTBL68_71	0x02c4	W	0x00000000	External wave table register17
CRU_SSGTBL72_75	0x02c8	W	0x00000000	External wave table register18
CRU_SSGTBL76_79	0x02cc	W	0x00000000	External wave table register19
CRU_SSGTBL80_83	0x02d0	W	0x00000000	External wave table register20
CRU_SSGTBL84_87	0x02d4	W	0x00000000	External wave table register21
CRU_SSGTBL88_91	0x02d8	W	0x00000000	External wave table register22
CRU_SSGTBL92_95	0x02dc	W	0x00000000	External wave table register23
CRU_SSGTBL96_99	0x02e0	W	0x00000000	External wave table register24
CRU_SSGTBL100_103	0x02e4	W	0x00000000	External wave table register25
CRU_SSGTBL104_107	0x02e8	W	0x00000000	External wave table register26
CRU_SSGTBL108_111	0x02ec	W	0x00000000	External wave table register27
CRU_SSGTBL112_115	0x02f0	W	0x00000000	External wave table register28
CRU_SSGTBL116_119	0x02f4	W	0x00000000	External wave table register29
CRU_SSGTBL120_123	0x02f8	W	0x00000000	External wave table register30
CRU_SSGTBL124_127	0x02fc	W	0x00000000	External wave table register31
CRU_SOFTRST_CON0	0x0300	W	0x00000000	Software reset control register0
CRU_SOFTRST_CON1	0x0304	W	0x00000000	Software reset control register1
CRU_SOFTRST_CON2	0x0308	W	0x00000000	Software reset control register2
CRU_SOFTRST_CON3	0x030c	W	0x00000010	Software reset control register3
CRU_SOFTRST_CON4	0x0310	W	0x00000000	Software reset control register4
CRU_SOFTRST_CON5	0x0314	W	0x00000000	Software reset control register5
CRU_SOFTRST_CON6	0x0318	W	0x00000000	Software reset control register6
CRU_SOFTRST_CON7	0x031c	W	0x00000000	Software reset control register7
CRU_SOFTRST_CON8	0x0320	W	0x00000000	Software reset control register8
CRU_SOFTRST_CON9	0x0324	W	0x00000000	Software reset control register9
CRU_SOFTRST_CON10	0x0328	W	0x00000000	Software reset control register10
CRU_SOFTRST_CON11	0x032c	W	0x00000000	Software reset control register11
CRU_SOFTRST_CON12	0x0330	W	0x00000000	Software reset control register12
CRU_SOFTRST_CON13	0x0334	W	0x00000000	Software reset control register13
CRU_SOFTRST_CON14	0x0338	W	0x00000000	Software reset control register14
CRU_SOFTRST_CON15	0x033c	W	0x00000000	Software reset control register15

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
CRU_SDMMC_CON0	0x0380	W	0x00000004	SDMMC control0
CRU_SDMMC_CON1	0x0384	W	0x00000000	SDMMC control1
CRU_SDIO_CON0	0x0388	W	0x00000004	SDIO control0
CRU_SDIO_CON1	0x038c	W	0x00000000	SDIO control1
CRU_EMMC_CON0	0x0390	W	0x00000004	EMMC control0
CRU_EMMC_CON1	0x0394	W	0x00000000	EMMC control1
CRU_PMUPLL_CON0	0x4000	W	0x000020fa	PPLL configuration register0
CRU_PMUPLL_CON1	0x4004	W	0x00001043	PPLL configuration register1
CRU_PMUPLL_CON2	0x4008	W	0x00000001	PPLL configuration register2
CRU_PMUPLL_CON3	0x400c	W	0x00000007	PPLL configuration register3
CRU_PMUPLL_CON4	0x4010	W	0x00007f00	PPLL configuration register4
CRU_PMU_MODE	0x4020	W	0x00000000	PMU_MODE
CRU_PMU_CLKSEL_CON0	0x4040	W	0x00000009	PMU Clock select and divide register0
CRU_PMU_CLKSEL_CON1	0x4044	W	0x0bb8ea60	PMU Clock select and divide register1
CRU_PMU_CLKSEL_CON2	0x4048	W	0x00003131	PMU Clock select and divide register2
CRU_PMU_CLKSEL_CON3	0x404c	W	0x00000009	PMU Clock select and divide register3
CRU_PMU_CLKSEL_CON4	0x4050	W	0x0000c009	PMU Clock select and divide register4
CRU_PMU_CLKSEL_CON5	0x4054	W	0x0bb8ea60	PMU Clock select and divide register5
CRU_PMU_CLKSEL_CON6	0x4058	W	0x00000001	PMU Clock select and divide register6
CRU_PMU_CLKSEL_CON7	0x405c	W	0x00000403	PMU Clock select and divide register7
CRU_PMU_CLKGATE_CON0	0x4080	W	0x00000000	PMU Clock gating register0
CRU_PMU_CLKGATE_CON1	0x4084	W	0x00000000	PMU Clock gating register1

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

2.6.3 Detail Register Description

CRU_APLL_CON0

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	bypass PLL Bypass. FREF bypasses PLL to FOUTPOSTDIV 1'b0: No bypass 1'b1: Bypass
14:12	RW	0x2	postdiv1 First Post Divide Value, (1-7)
11:0	RW	0x0fa	fbdiv Feedback Divide Value, valid divider settings are: [16, 2500] in integer mode [20, 500] in fractional mode Tips: No plus one operation

CRU_APLL_CON1

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	pllpdsel PLL global power down source selection If pllpdsel == 1, PLL can be power down only by pllpd1, otherwise pll is power down when any one of refdiv/fbdiv/fracdiv is changed or pllpd0 is asserted
14	RW	0x0	pllpd1 PLL global power down request 1'b0: No power down 1'b1: Power down
13	RW	0x0	pllpd0 PLL global power down request 1'b0: No power down 1'b1: Power down
12	RW	0x1	dsmpd PLL delta sigma modulator enable 1'b0: Modulator is enable 1'b1: Modulator is disabled
11	RO	0x0	Reserved
10	RO	0x0	pll_lock PLL lock status 1'b0: Unlock 1'b1: Lock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
9	RO	0x0	Reserved
8:6	RW	0x1	postdiv2 Second Post Divide Value, (1-7)
5:0	RW	0x03	refdiv Reference Clock Divide Value, (1-63)

CRU_APLL_CON2

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:28	RO	0x0	Reserved
27	RW	0x0	fout4phasepd Power down 4-phase clocks and 2X, 3X, 4X clocks 1'b0: No power down 1'b1: Power down
26	RW	0x0	foutvcopd Power down buffered VCO clock 1'b0: No power down 1'b1: Power down
25	RW	0x0	foutpostdivpd Power down all outputs except for buffered VCO clock 1'b0: No power down 1'b1: Power down
24	RW	0x0	dacpd Power down quantization noise cancellation DAC 1'b0: No power down 1'b1: Power down
23:0	RW	0x000001	fracdiv Fractional part of feedback divide (fraction = FRAC/2^24)

CRU_APLL_CON3

Address: Operational Base + offset (0x000c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:13	RO	0x0	Reserved
12:8	WO	0x00	ssmod_spread spread amplitude % = 0.1 * SPREAD[4: 0]
7:4	WO	0x0	ssmod_divval Divider required to set the modulation frequency
3	WO	0x0	ssmod_downspread Selects center spread or downs pread 1'b0: Down spread 1'b1: Center spread
2	WO	0x1	ssmod_reset Reset modulator state 1'b0: No reset 1'b1: Reset
1	WO	0x1	ssmod_disable_sscg Bypass SSMOD by module 1'b0: No bypass 1'b1: Bypass
0	WO	0x1	ssmod_bp Bypass SSMOD by integration 1'b0: No bypass 1'b1: Bypass

CRU_APLL_CON4

Address: Operational Base + offset (0x0010)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	WO	0x7f	ssmod_ext_maxaddr External wave table data inputs (0-255)
7:1	RO	0x0	Reserved
0	WO	0x0	ssmod_sel_ext_wave select external wave 1'b0: No select ext_wave 1'b1: Select ext_wave

CRU_DPLL_CON0

Address: Operational Base + offset (0x0020)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	bypass PLL Bypass. FREF bypasses PLL to FOUTPOSTDIV 1'b0: No bypass 1'b1: Bypass
14:12	RW	0x1	postdiv1 First Post Divide Value, (1-7)
11:0	RW	0x0c8	fbdiv Feedback Divide Value, valid divider settings are: [16, 2500] in integer mode [20, 500] in fractional mode Tips: No plus one operation

CRU_DPLL_CON1

Address: Operational Base + offset (0x0024)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	pllpdsel PLL global power down source selection If pllpdsel == 1, PLL can be power down only by pllpd1, otherwise pll is power down when any one of refdiv/fbdiv/fracdiv is changed or pllpd0 is asserted
14	RW	0x0	pllpd1 PLL global power down request 1'b0: No power down 1'b1: Power down
13	RW	0x0	pllpd0 PLL global power down request 1'b0: No power down 1'b1: Power down
12	RW	0x1	dsmpd PLL delta sigma modulator enable 1'b0: Modulator is enable, 1'b1: Modulator is disabled
11	RO	0x0	Reserved
10	RO	0x0	pll_lock PLL lock status 1'b0: Unlock 1'b1: Lock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
9	RO	0x0	Reserved
8:6	RW	0x1	postdiv2 Second Post Divide Value (1-7)
5:0	RW	0x03	refdiv Reference Clock Divide Value (1-63)

CRU_DPLL_CON2

Address: Operational Base + offset (0x0028)

Bit	Attr	Reset Value	Description
31:28	RO	0x0	Reserved
27	RW	0x0	fout4phasepd Power down 4-phase clocks and 2X, 3X, 4X clocks 1'b0: No power down 1'b1: Power down
26	RW	0x0	foutvcopd Power down buffered VCO clock 1'b0: No power down 1'b1: Power down
25	RW	0x0	foutpostdivpd Power down all outputs except for buffered VCO clock 1'b0: No power down 1'b1: Power down
24	RW	0x0	dacpd Power down quantization noise cancellation DAC 1'b0: No power down 1'b1: Power down
23:0	RW	0x000001	fracdiv Fractional part of feedback divide (fraction = FRAC/2^24)

CRU_DPLL_CON3

Address: Operational Base + offset (0x002c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:13	RO	0x0	Reserved
12:8	WO	0x00	ssmod_spread spread amplitude % = 0.1 * SPREAD[4: 0]
7:4	WO	0x0	ssmod_divval Divider required to set the modulation frequency
3	WO	0x0	ssmod_downspread Selects center spread or downs pread 1'b0: Down spread 1'b1: Center spread
2	WO	0x1	ssmod_reset Reset modulator state 1'b0: No reset 1'b1: Reset
1	WO	0x1	ssmod_disable_sscg Bypass SSMOD by module 1'b0: No bypass 1'b1: Bypass
0	WO	0x1	ssmod_bp Bypass SSMOD by integration 1'b0: No bypass 1'b1: Bypass

CRU_DPLL_CON4

Address: Operational Base + offset (0x0030)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	WO	0x7f	ssmod_ext_maxaddr External wave table data inputs, (0-255)
7:1	RO	0x0	Reserved
0	WO	0x0	ssmod_sel_ext_wave select external wave 1'b0: No select ext_wave 1'b1: Select ext_wave

CRU_CPLL_CON0

Address: Operational Base + offset (0x0040)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	bypass PLL Bypass. FREF bypasses PLL to FOUTPOSTDIV 1'b0: No bypass 1'b1: Bypass
14:12	RW	0x2	postdiv1 First Post Divide Value, (1-7)
11:0	RW	0x0fa	fbdiv Feedback Divide Value, valid divider settings are: [16, 2500] in integer mode [20, 500] in fractional mode Tips: No plus one operation

CRU_CPLL_CON1

Address: Operational Base + offset (0x0044)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	pllpdsel PLL global power down source selection If pllpdsel == 1, PLL can be power down only by pllpd1, otherwise pll is power down when any one of refdiv/fbdiv/fracdiv is changed or pllpd0 is asserted
14	RW	0x0	pllpd1 PLL global power down request 1'b0: No power down 1'b1: Power down
13	RW	0x0	pllpd0 PLL global power down request 1'b0: No power down 1'b1: Power down
12	RW	0x1	dsmpd PLL delta sigma modulator enable 1'b0: Modulator is enable 1'b1: Modulator is disabled
11	RO	0x0	Reserved
10	RO	0x0	pll_lock PLL lock status 1'b0: Unlock 1'b1: Lock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
9	RO	0x0	Reserved
8:6	RW	0x1	postdiv2 Second Post Divide Value (1-7)
5:0	RW	0x03	refdiv Reference Clock Divide Value (1-63)

CRU_CPLL_CON2

Address: Operational Base + offset (0x0048)

Bit	Attr	Reset Value	Description
31:28	RO	0x0	Reserved
27	RW	0x0	fout4phasepd Power down 4-phase clocks and 2X, 3X, 4X clocks 1'b0: No power down 1'b1: Power down
26	RW	0x0	foutvcopd Power down buffered VCO clock 1'b0: No power down 1'b1: Power down
25	RW	0x0	foutpostdivpd Power down all outputs except for buffered VCO clock 1'b0: No power down 1'b1: Power down
24	RW	0x0	dacpd Power down quantization noise cancellation DAC 1'b0: No power down 1'b1: Power down
23:0	RW	0x000001	fracdiv Fractional part of feedback divide (fraction = FRAC/2^24)

CRU_CPLL_CON3

Address: Operational Base + offset (0x004c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:13	RO	0x0	Reserved
12:8	WO	0x00	ssmod_spread spread amplitude % = 0.1 * SPREAD[4: 0]

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7:4	WO	0x0	ssmod_divval Divider required to set the modulation frequency
3	WO	0x0	ssmod_downspread Selects center spread or downs pread 1'b0: Down spread 1'b1: Center spread
2	WO	0x1	ssmod_reset Reset modulator state 1'b0: No reset 1'b1: Reset
1	WO	0x1	ssmod_disable_sscg Bypass SSMOD by module 1'b0: No bypass 1'b1: Bypass
0	WO	0x1	ssmod_bp Bypass SSMOD by integration 1'b0: No bypass 1'b1: Bypass

CRU_CPLL_CON4

Address: Operational Base + offset (0x0050)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	WO	0x7f	ssmod_ext_maxaddr External wave table data inputs (0-255)
7:1	RO	0x0	Reserved
0	WO	0x0	ssmod_sel_ext_wave select external wave 1'b0: No select ext_wave 1'b1: Select ext_wave

CRU_GPLL_CON0

Address: Operational Base + offset (0x0060)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15	RW	0x0	bypass PLL Bypass. FREF bypasses PLL to FOUTPOSTDIV 1'b0: No bypass 1'b1: Bypass
14:12	RW	0x2	postdiv1 First Post Divide Value, (1-7)
11:0	RW	0x064	fbdiv Feedback Divide Value, valid divider settings are: [16, 2500] in integer mode [20, 500] in fractional mode Tips: No plus one operation

CRU_GPLL_CON1

Address: Operational Base + offset (0x0064)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	pllpdsel PLL global power down source selection If pllpdsel == 1, PLL can be power down only by pllpd1, otherwise pll is power down when any one of refdiv/fbdiv/fracdiv is changed or pllpd0 is asserted
14	RW	0x0	pllpd1 PLL global power down request 1'b0: No power down 1'b1: Power down
13	RW	0x0	pllpd0 PLL global power down request 1'b0: No power down 1'b1: Power down
12	RW	0x1	dsmpd PLL delta sigma modulator enable 1'b0: Modulator is enable 1'b1: Modulator is disabled
11	RO	0x0	Reserved
10	RO	0x0	pll_lock PLL lock status 1'b0: Unlock 1'b1: Lock
9	RO	0x0	Reserved
8:6	RW	0x1	postdiv2 Second Post Divide Value, (1-7)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5:0	RW	0x01	refdiv Reference Clock Divide Value, (1-63)

CRU_GPLL_CON2

Address: Operational Base + offset (0x0068)

Bit	Attr	Reset Value	Description
31:28	RO	0x0	Reserved
27	RW	0x0	fout4phasepd Power down 4-phase clocks and 2X, 3X, 4X clocks 1'b0: No power down 1'b1: Power down
26	RW	0x0	foutvcopd Power down buffered VCO clock 1'b0: No power down 1'b1: Power down
25	RW	0x0	foutpostdivpd Power down all outputs except for buffered VCO clock 1'b0: No power down 1'b1: Power down
24	RW	0x0	dacpd Power down quantization noise cancellation DAC 1'b0: No power down 1'b1: Power down
23:0	RW	0x000001	fracdiv Fractional part of feedback divide (fraction = FRAC/2^24)

CRU_GPLL_CON3

Address: Operational Base + offset (0x006c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:13	RO	0x0	Reserved
12:8	WO	0x00	ssmod_spread spread amplitude % = 0.1 * SPREAD[4: 0]
7:4	WO	0x0	ssmod_divval Divider required to set the modulation frequency
3	WO	0x0	ssmod_downspread Selects center spread or downs pread 1'b0: Down spread 1'b1: Center spread

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2	WO	0x1	ssmod_reset Reset modulator state 1'b0: No reset 1'b1: Reset
1	WO	0x1	ssmod_disable_sscg Bypass SSMOD by module 1'b0: No bypass 1'b1: Bypass
0	WO	0x1	ssmod_bp Bypass SSMOD by integration 1'b0: No bypass 1'b1: Bypass

CRU_GPLL_CON4

Address: Operational Base + offset (0x0070)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	WO	0x7f	ssmod_ext_maxaddr External wave table data inputs (0-255)
7:1	RO	0x0	Reserved
0	WO	0x0	ssmod_sel_ext_wave select external wave 1'b0: No select ext_wave 1'b1: Select ext_wave

CRU_NPLL_CON0

Address: Operational Base + offset (0x0080)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	bypass PLL Bypass. FREF bypasses PLL to FOUTPOSTDIV 1'b0: No bypass 1'b1: Bypass
14:12	RW	0x2	postdiv1 First Post Divide Value, (1-7)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
11:0	RW	0x0c8	fbdiv Feedback Divide Value, valid divider settings are: [16, 2500] in integer mode [20, 500] in fractional mode Tips: No plus one operation

CRU_NPLL_CON1

Address: Operational Base + offset (0x0084)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	pllpdsel PLL global power down source selection If pllpdsel == 1, PLL can be power down only by pllpd1, otherwise pll is power down when any one of refdiv/fbdiv/fracdiv is changed or pllpd0 is asserted
14	RW	0x0	pllpd1 PLL global power down request 1'b0: No power down 1'b1: Power down
13	RW	0x0	pllpd0 PLL global power down request 1'b0: No power down 1'b1: Power down
12	RW	0x1	dsmpd PLL delta sigma modulator enable 1'b0: Modulator is enable 1'b1: Modulator is disabled
11	RO	0x0	Reserved
10	RO	0x0	pll_lock PLL lock status 1'b0: Unlock 1'b1: Lock
9	RO	0x0	Reserved
8:6	RW	0x1	postdiv2 Second Post Divide Value, (1-7)
5:0	RW	0x03	refdiv Reference Clock Divide Value, (1-63)

CRU_NPLL_CON2

Address: Operational Base + offset (0x0088)

Bit	Attr	Reset Value	Description
31:28	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
27	RW	0x0	fout4phasepd Power down 4-phase clocks and 2X, 3X, 4X clocks 1'b0: No power down 1'b1: Power down
26	RW	0x0	foutvcopd Power down buffered VCO clock 1'b0: No power down 1'b1: Power down
25	RW	0x0	foutpostdivpd Power down all outputs except for buffered VCO clock 1'b0: No power down 1'b1: Power down
24	RW	0x0	dacpd Power down quantization noise cancellation DAC 1'b0: No power down 1'b1: Power down
23:0	RW	0x000001	fracdiv Fractional part of feedback divide (fraction = FRAC/2^24)

CRU_NPLL_CON3

Address: Operational Base + offset (0x008c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:13	RO	0x0	Reserved
12:8	WO	0x00	ssmod_spread spread amplitude % = 0.1 * SPREAD[4: 0]
7:4	WO	0x0	ssmod_divval Divider required to set the modulation frequency
3	WO	0x0	ssmod_downspread Selects center spread or downs pread 1'b0: Down spread 1'b1: Center spread
2	WO	0x1	ssmod_reset Reset modulator state 1'b0: No reset 1'b1: Reset
1	WO	0x1	ssmod_disable_sscg Bypass SSMOD by module 1'b0: No bypass 1'b1: Bypass

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
0	WO	0x1	ssmod_bp Bypass SSMOD by integration 1'b0: No bypass 1'b1: Bypass

CRU_NPLL_CON4

Address: Operational Base + offset (0x0090)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	WO	0x7f	ssmod_ext_maxaddr External wave table data inputs (0-255)
7:1	RO	0x0	Reserved
0	WO	0x0	ssmod_sel_ext_wave select external wave 1'b0: No select ext_wave 1'b1: Select ext_wave

CRU_MODE

Address: Operational Base + offset (0x00a0)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RO	0x0	Reserved
11:10	RW	0x0	usbphy480m_work_mode 2'b00: Clock from xin_osc0_func_div 2'b01: Clock from usbphy 480M 2'b10: Clock from clk_rtc_32k 2'b11: Reserved
9:8	RW	0x0	npll_work_mode 2'b00: Clock from xin_osc0_func_div 2'b01: Clock from newpll 2'b10: Clock from clk_rtc_32k 2'b11: Reserved
7:6	RW	0x0	gpll_work_mode 2'b00: Clock from xin_osc0_func_div 2'b01: Clock from generalpll 2'b10: Clock from clk_rtc_32k 2'b11: Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5:4	RW	0x0	cpll_work_mode 2'b00: Clock from xin_osc0_func_div 2'b01: Clock from codecpll 2'b10: Clock from clk_rtc_32k 2'b11: Reserved
3:2	RW	0x0	dpll_work_mode 2'b00: Clock from xin_osc0_func_div 2'b01: Clock from ddrpll 2'b10: Clock from clk_rtc_32k 2'b11: Reserved
1:0	RW	0x0	apll_work_mode 2'b00: Clock from xin_osc0_func_div 2'b01: Clock from armpll 2'b10: Clock from clk_rtc_32k 2'b11: Reserved

CRU_MISC

Address: Operational Base + offset (0x00a4)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RO	0x0	Reserved
13:12	RW	0x0	core_high_freq_rst_en 1'b0: Disable core high frequency reset gate function 1'b1: Enable core high frequency reset gate function Each bit for each core, eg. bit0 for core0
11:5	RO	0x0	Reserved
4	RW	0x0	corepo_wrst_wfien 1'b0: Disable core0/1 warm reset for cpu power on reset 1'b1: Enable core0/1 warm reset for cpu power on reset
3	RW	0x0	corepo_srst_wfien 1'b0: Disable core0/1 wif reset for cpu power on reset 1'b1: Enable core0/1 wfi reset for cpu power on reset
2	RW	0x0	core_wrst_wfien 1'b0: Disable core0/1 warm reset for cpu reset 1'b1: Enable core0/1 warm reset for cpu reset
1	RW	0x0	core_srst_wfien 1'b0: Disable core0/1 wif reset for cpu reset 1'b1: Enable core0/1 wfi reset for cpu reset
0	RW	0x0	warmrst_en 1'b0: Disable cpu warm reset 1'b1: Enable cpu warm reset

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

CRU_MISC1

Address: Operational Base + offset (0x00a8)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RO	0x0	Reserved
14	RW	0x0	clk_i2s1_out_mclk_oen clk_i2s1_out_mclk pad output enable, low active 1'b0: Enable clk_i2s1_out_mclk output to pad 1'b1: Disable clk_i2s1_out_mclk output to pad
13	RW	0x0	clk_i2s0_rx_out_mclk_oen clk_i2s0_rx_out_mclk pad output enable, low active 1'b0: Enable clk_i2s0_rx_out_mclk output to pad 1'b1: Disable clk_i2s0_rx_out_mclk output to pad
12	RW	0x0	clk_i2s0_tx_out_mclk_oen clk_i2s0_tx_out_mclk pad output enable, low active 1'b0: Enable clk_i2s0_tx_out_mclk output to pad 1'b1: Disable clk_i2s0_tx_out_mclk output to pad
11	RO	0x0	Reserved
10	RW	0x0	pd_vpu_dwn_clk_en_dis 1'b1: Disable PMU control of clock off when pd_vpu power down
9:7	RO	0x0	Reserved
6	RW	0x0	pd_ddr_dwn_clk_en_dis 1'b1: Disable PMU control of clock off when pd_ddr power down
5	RO	0x0	Reserved
4	RW	0x0	pd_npu_dwn_clk_en_dis 1'b1: Disable PMU control of clock off when pd_npu power down
3	RW	0x0	pd_vio_dwn_clk_en_dis 1'b1: Disable PMU control of clock off when pd_vio power down
2	RW	0x0	pd_pcie_dwn_clk_en_dis 1'b1: Disable PMU control of clock off when pd_pcie power down
1	RO	0x0	Reserved
0	RW	0x0	pd_core_dwn_clk_en_dis 1'b1: Disable PMU control of clock off when pd_core power down

CRU_GLB_CNT_TH

Address: Operational Base + offset (0x00b0)

Bit	Attr	Reset Value	Description
31:16	RW	0x3a98	pll_lockperiod PLL lock filtered period time, measured in OSC clock cycles
15:0	RW	0x0064	global_reset_counter_threshold Global soft reset, wdt reset or tsadc_shut reset asserted time counter threshold. Measured in OSC clock cycles

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

CRU_GLB_RST_ST

Address: Operational Base + offset (0x00b4)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	Reserved
19	RW	0x0	resetn_corepo1_src_st power on resetn source status of core1
18	RW	0x0	resetn_corepo0_src_st power on resetn source status of core0
17	RW	0x0	resetn_core1_src_st core resetn source status of core1
16	RW	0x0	resetn_core0_src_st core resetn source status of core0
15:6	RO	0x0	Reserved
5	W1C	0x0	snd_glb_tsadc_rst_st sencond global TSADC triggered reset flag 1'b0: Last hot reset is not sencond global TSADC triggered reset 1'b1: Last hot reset is sencond global TSADC triggered reset
4	W1C	0x0	fst_glb_tsadc_rst_st first global TSADC triggered reset flag 1'b0: Last hot reset is not first global TSADC triggered reset 1'b1: Last hot reset is first global TSADC triggered reset
3	W1C	0x0	snd_glb_wdt_rst_st sencond global WDT triggered reset flag 1'b0: Last hot reset is not sencond global WDT triggered reset 1'b1: Last hot reset is sencond global WDT triggered reset
2	W1C	0x0	fst_glb_wdt_rst_st first global WDT triggered reset flag 1'b0: Last hot reset is not first global WDT triggered reset 1'b1: Last hot reset is first global WDT triggered reset
1	W1C	0x0	snd_glb_rst_st second global rst flag 1'b0: Last hot reset is not sencond global reset 1'b1: Last hot reset is sencond global reset
0	W1C	0x0	fst_glb_rst_st first global rst flag 1'b0: Last hot reset is not first global reset 1'b1: Last hot reset is first global reset

CRU_GLB_SRST_FST

Address: Operational Base + offset (0x00b8)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	Reserved
15:0	RW	0x0000	GLB_SRST_FST The first global software reset config value

CRU_GLB_SRST_SND

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x00bc)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	Reserved
15:0	RW	0x0000	GLB_SRST_SND The second global software reset config value

CRU_GLB_RST_CON

Address: Operational Base + offset (0x00c0)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	Reserved
7	RW	0x0	wdt_reset_ext_en 1'b0: Disable wdt reset extend 1'b1: Enable wdt reset extend, reset extend time depend on bit15~0 of GLB_CNT_TH
6	RW	0x0	tsadc_shut_reset_ext_en 1'b0: Disable tsadc_shut reset extend 1'b1: Enable tsadc_shut reset extend, reset extend time depend on bit15~0 of GLB_CNT_TH
5	RO	0x0	Reserved
4	RW	0x0	pmu_srst_wdt_en 1'b0: Enable wdt reset as pmu reset source 1'b1: Disable wdt reset as pmu reset source
3:2	RO	0x0	Reserved
1	RW	0x0	wdt_glb_srst_ctrl 1'b0: WDT trigger second global reset 1'b1: WDT trigger first global reset
0	RW	0x0	tsadc_glb_srst_ctrl 1'b0: TSADC trigger second global reset 1'b1: TSADC trigger first global reset

CRU_CLKSEL_CON0

Address: Operational Base + offset (0x0100)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RO	0x0	Reserved
14:12	RW	0x1	aclk_core_div_con aclk_core=clk_core/(div_con+1)
11:8	RW	0x3	core_dbg_div_con pclk_dbg=clk_core/(div_con+1)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7:6	RW	0x0	core_clk_pll_sel 2'b00: APLL 2'b01: CPLL 2'b10: GPLL 2'b11: Reserved
5:4	RO	0x0	Reserved
3:0	RW	0x0	clk_core_div_con clk_core=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON1

Address: Operational Base + offset (0x0104)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	npu_src_clk_sel 1'b0: Select clk_npu_div 1'b1: Select clk_npu_np5
14:12	RO	0x0	Reserved
11:10	RW	0x0	npu_np5_pll_clk_sel 2'b00: GPLL 2'b01: CPLL 2'b10: APLL 2'b11: Reserved
9:8	RW	0x0	npu_pll_clk_sel 2'b00: GPLL 2'b01: CPLL 2'b10: APLL 2'b11: Reserved
7:4	RW	0x1	clk_npu_np5_div_con clk_npu_np5=2*npu_np5_pll_src/(2*div_con+3)
3:0	RW	0x1	clk_npu_div_con clk_npu_div=npu_pll_src/(div_con+1)

CRU_CLKSEL_CON2

Address: Operational Base + offset (0x0108)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	npu_hclk_pll_sel 1'b0: GPLL 1'b1: CPLL

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
14	RW	0x0	npu_aclk_pll_sel 1'b0: GPLL 1'b1: CPLL
13:12	RO	0x0	Reserved
11:8	RW	0x7	hclk_npu_div_con hclk_npu=pll_src/(div_con+1)
7:4	RW	0x1	aclk_npu2mem_div_con aclk_npu2mem=aclk_npu/(div_con+1)
3:0	RW	0x1	aclk_npu_div_con aclk_npu=pll_src/(div_con+1)

CRU_CLKSEL_CON3

Address: Operational Base + offset (0x010c)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:13	RO	0x0	Reserved
12:8	RW	0x0b	pclk_ddr_div_con pclk_ddr=gpll_clk_src/(div_con+1)
7	RW	0x0	ddrphy1x_pll_clk_sel 1'b0: DPLL 1'b1: GPLL
6:5	RO	0x0	Reserved
4:0	RW	0x03	ddrphy1x_div_con clk_ddrphy1x=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON4

Address: Operational Base + offset (0x0110)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RO	0x0	Reserved
11:8	RW	0x1	lsclk_vio_div_con lsclk_vio=hsclk_vio/(div_con+1)
7	RW	0x0	hsclk_vio_pll_sel 1'b0: GPLL 1'b1: CPLL
6:5	RO	0x0	Reserved
4:0	RW	0x03	hsclk_vio_div_con hsclk_vio=pll_clk_src/(div_con+1)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

CRU_CLKSEL_CON5

Address: Operational Base + offset (0x0114)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	dclk_vopraw_sel 2'b00: Select dclk_vopraw 2'b01: Select dclk_vopraw_frac_out 2'b10: Select xin_osc0 2'b11: Reserved
13:12	RO	0x0	Reserved
11:10	RW	0x1	dclk_vopraw_pll_sel 2'b00: CPLL 2'b01: GPLL 2'b10: NPLL 2'b11: Reserved
9:8	RO	0x0	Reserved
7:0	RW	0x03	dclk_vopraw_div_con dclk_vopraw=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON6

Address: Operational Base + offset (0x0118)

Bit	Attr	Reset Value	Description
31:0	RW	0x0bb8ea60	dclk_vopraw_frac_div_con High 16-bit for numerator, Low 16-bit for denominator, clock source is dclk_vopraw

CRU_CLKSEL_CON7

Address: Operational Base + offset (0x011c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	dclk_voplite_sel 2'b00: Select dclk_voplite 2'b01: Select dclk_voplite_frac_out 2'b10: Select xin_osc0 2'b11: Reserved
13:12	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
11:10	RW	0x1	dclk_voplite_pll_sel 2'b00: CPLL 2'b01: GPLL 2'b10: NPLL 2'b11: Reserved
9:8	RO	0x0	Reserved
7:0	RW	0x07	dclk_voplite_div_con dclk_voplite=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON8

Address: Operational Base + offset (0x0120)

Bit	Attr	Reset Value	Description
31:0	RW	0x0bb8ea60	dclk_voplite_frac_div_con High 16-bit for numerator, Low 16-bit for denominator, clock source is dclk_voplite

CRU_CLKSEL_CON9

Address: Operational Base + offset (0x0124)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RO	0x0	Reserved
11:0	RW	0x03c	clk_txesc_div_con clk_txesc=gpll_clk_src/(div_con+1)

CRU_CLKSEL_CON10

Address: Operational Base + offset (0x0128)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	clk_isp_pll_sel 2'b00: GPLL 2'b01: CPLL 2'b10: NPLL 2'b11: Reserved
13	RO	0x0	Reserved
12:8	RW	0x02	clk_isp_div_con clk_isp=pll_clk_src/(div_con+1)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7:6	RW	0x0	clk_rga_core_pll_sel 2'b00: GPLL 2'b01: CPLL 2'b10: NPLL 2'b11: Reserved
5	RO	0x0	Reserved
4:0	RW	0x03	clk_rga_core_div_con clk_rga_core=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON11

Address: Operational Base + offset (0x012c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	dclk_cif_pll_sel 2'b00: CPLL 2'b01: GPLL 2'b10: NPLL 2'b11: Reserved
13	RO	0x0	Reserved
12:8	RW	0x02	dclk_cif_div_con dclk_cif=pll_clk_src/(div_con+1)
7:6	RW	0x0	clk_cif_out_pll_sel 2'b00: Select xin_osc0 2'b01: Select NPLL 2'b10: Select GPLL 2'b11: Select USBPHY480M
5:0	RW	0x02	clk_cif_out_div_con clk_cif_out=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON12

Address: Operational Base + offset (0x0130)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	pdclk_pcie_pll_sel 1'b0: GPLL 1'b1: CPLL
14:13	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
12:8	RW	0x05	lsclk_pcie_div_con lsclk_pcie=pll_clk_src/(div_con+1)
7:5	RO	0x0	Reserved
4:0	RW	0x03	hsclk_pcie_div_con hsclk_pcie=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON13

Address: Operational Base + offset (0x0134)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:13	RO	0x0	Reserved
12	RW	0x0	usb3_otg_suspend_src_sel 1'b1: Select clk_rtc_32k 1'b0: Select xin_osc0
11:10	RO	0x0	Reserved
9:0	RW	0x000	clk_usb3_otg_suspend_div_con clk_usb3_otg_suspend=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON14

Address: Operational Base + offset (0x0138)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:13	RO	0x0	Reserved
12	RW	0x0	clk_pcie_aux_src_sel 1'b0: Select xin_osc0 1'b1: Select clk_pcie_aux_src
11:10	RO	0x0	Reserved
9:8	RW	0x0	clk_pcie_aux_pll_sel 2'b00: CPLL 2'b01: GPLL 2'b10: NPLL 2'b11: Reserved
7	RO	0x0	Reserved
6:0	RW	0x63	clk_pcie_aux_div_con clk_pcie_aux_src=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON15

Address: Operational Base + offset (0x013c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x3	aclk_gic_div_con aclk_gic=pll_clk_src/(div_con+1)
11	RW	0x0	gic_pll_clk_sel 1'b0: GPLL 1'b1: CPLL
10:9	RO	0x0	Reserved
8	RW	0x0	aclk_pcie_pll_sel 1'b0: GPLL 1'b1: CPLL
7:4	RW	0x2	pclk_pcie_div_con pclk_pcie=aclk_pcie/(div_con+1)
3:0	RW	0x3	aclk_pcie_div_con aclk_pcie=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON16

Address: Operational Base + offset (0x0140)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RO	0x0	Reserved
11:8	RW	0x1	pclk_vpu_div_con pclk_vpu=aclk_vpu/(div_con+1)
7	RW	0x0	aclk_vpu_pll_clk_sel 1'b0: GPLL 1'b1: CPLL
6:5	RO	0x0	Reserved
4:0	RW	0x03	aclk_vpu_div_con aclk_vpu=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON17

Address: Operational Base + offset (0x0144)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15	RW	0x0	hsclk_system_sram_pll_clk_sel 1'b0: GPLL 1'b1: CPLL
14:13	RO	0x0	Reserved
12:8	RW	0x03	hsclk_system_sram_div_con hsclk_system_sram=pll_clk_src/(div_con+1)
7	RW	0x0	aclk_system_sram_pll_clk_sel 1'b0: GPLL 1'b1: CPLL
6:5	RO	0x0	Reserved
4:0	RW	0x03	aclk_system_sram_div_con aclk_system_sram=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON18

Address: Operational Base + offset (0x0148)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	clk_gmac_out_pll_sel 2'b00: CPLL 2'b01: NPLL 2'b10: PMUPLL 2'b11: Reserved
13	RO	0x0	Reserved
12:8	RW	0x13	clk_gmac_out_div_con clk_gmac_out=pll_clk_src/(div_con+1)
7:5	RO	0x0	Reserved
4:0	RW	0x05	msclk_core_niu_div_con msclk_core_niu=gpll_clk_src/(div_con+1)

CRU_CLKSEL_CON19

Address: Operational Base + offset (0x014c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	pd_peri_pll_sel 1'b0: GPLL 1'b1: CPLL
14:13	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
12:8	RW	0x07	lsclk_peri_div_con lsclk_peri=pll_clk_src/(div_con+1)
7:5	RO	0x0	Reserved
4:0	RW	0x05	msclk_peri_div_con msclk_peri=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON20

Address: Operational Base + offset (0x0150)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	clk_sdmmc_pll_sel 2'b00: Select GPLL 2'b01: Select CPLL 2'b10: Select NPLL 2'b11: Select xin_osc0
13:8	RO	0x0	Reserved
7:0	RW	0x07	clk_sdmmc_div_con clk_sdmmc=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON21

Address: Operational Base + offset (0x0154)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	clk_sdmmc_sel 1'b0: Select clk_sdmmc 1'b1: Select clk_sdmmc_div50
14:8	RO	0x0	Reserved
7:0	RW	0x07	clk_sdmmc_div50_div_con clk_sdmmc_div50=clk_sdmmc/(div_con+1), duty cycle is 50% for any value

CRU_CLKSEL_CON22

Address: Operational Base + offset (0x0158)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable

Bit	Attr	Reset Value	Description
15:14	RW	0x0	clk_sdio_pll_sel 2'b00: Select GPLL 2'b01: Select CPLL 2'b10: Select NPLL 2'b11: Select xin_osc0
13:8	RO	0x0	Reserved
7:0	RW	0x07	clk_sdio_div_con clk_sdio=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON23

Address: Operational Base + offset (0x015c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	clk_sdio_sel 1'b0: Select clk_sdio 1'b1: Select clk_sdio_div50
14:8	RO	0x0	Reserved
7:0	RW	0x07	clk_sdio_div50_div_con clk_sdio_div50=clk_sdio/(div_con+1), duty cycle is 50% for any value

CRU_CLKSEL_CON24

Address: Operational Base + offset (0x0160)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	clk_emmc_pll_sel 2'b00: Select GPLL 2'b01: Select CPLL 2'b10: Select NPLL 2'b11: Select xin_osc0
13:8	RO	0x0	Reserved
7:0	RW	0x07	clk_emmc_div_con clk_emmc=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON25

Address: Operational Base + offset (0x0164)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	clk_emmc_sel 1'b0: Select clk_emmc 1'b1: Select clk_emmc_div50
14:8	RO	0x0	Reserved
7:0	RW	0x07	clk_emmc_div50_div_con clk_emmc_div50=clk_emmc/(div_con+1), duty cycle is 50% for any value

CRU_CLKSEL_CON26

Address: Operational Base + offset (0x0168)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	clk_gmac_pll_sel 2'b00: CPLL 2'b01: NPLL 2'b10: PMUPLL 2'b11: Reserved
13	RO	0x0	Reserved
12:8	RW	0x07	clk_gmac_div_con clk_gmac=pll_clk_src/(div_con+1)
7	RW	0x0	clk_sfc_pll_sel 1'b0: GPLL 1'b1: CPLL
6:0	RW	0x0b	clk_sfc_div_con clk_sfc=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON27

Address: Operational Base + offset (0x016c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	pdbus_clk_pll_sel 1'b0: GPLL 1'b1: CPLL
14:13	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
12:8	RW	0x03	hsclk_bus_div_con hsclk_bus=pll_clk_src/(div_con+1)
7:5	RO	0x0	Reserved
4	RW	0x0	rmii_mode 1'b0: Rgmii mode 1'b1: Rmii mode
3:2	RW	0x0	gmac_clk_sel gmac tx clk speed when rgmii mode 2'b00: 125M 2'b01: 125M 2'b10: 2.5M 2'b11: 25M
1	RW	0x1	rmii_clk_sel gmac tx/rx clk speed when rmii mode 1'b0: 2.5M 1'b1: 25M
0	RW	0x0	rmii_extclksrc_sel 1'b0: Select internal soc clk_gmac as gmac controller clock source 1'b1: Select external phy clock as gmac controller clock source

CRU_CLKSEL_CON28

Address: Operational Base + offset (0x0170)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:13	RO	0x0	Reserved
12:8	RW	0x0b	lsclk_bus_div_con lsclk_bus=pll_clk_src/(div_con+1)
7:5	RO	0x0	Reserved
4:0	RW	0x05	msclk_bus_div_con msclk_bus=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON29

Address: Operational Base + offset (0x0174)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15	RW	0x0	clk_crypto_pka_sel 1'b0: GPLL 1'b1: CPLL
14:13	RO	0x0	Reserved
12:8	RW	0x03	clk_crypto_pka_div_con clk_crypto_pka=pll_clk_src/(div_con+1)
7	RW	0x0	clk_crypto_pll_sel 1'b0: GPLL 1'b1: CPLL
6:5	RO	0x0	Reserved
4:0	RW	0x05	clk_crypto_div_con clk_crypto=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON30

Address: Operational Base + offset (0x0178)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	clk_pdm_sel 1'b0: Select clk_pdm 1'b1: Select clk_pdm_frac_out
14:10	RO	0x0	Reserved
9:8	RW	0x0	clk_pdm_pll_sel 2'b00: Select GPLL 2'b01: Select xin_osc0 2'b10: Select CPLL 2'b11: Select NPLL
7	RO	0x0	Reserved
6:0	RW	0x0b	clk_pdm_div_con clk_pdm=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON31

Address: Operational Base + offset (0x017c)

Bit	Attr	Reset Value	Description
31:0	RW	0x0bb8ea60	clk_pdm_frac_div_con High 16-bit for numerator, Low 16-bit for denominator, clock source is clk_pdm

CRU_CLKSEL_CON32

Address: Operational Base + offset (0x0180)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	clk_i2s0_tx_out_mclk_sel 2'b00: Select selected clock by clk_i2s0_tx_rx_clk_sel 2'b01: Select xin_osc0_half 2'b10: Select clk_i2s0_rx 2'b11: Reserved
13	RO	0x0	Reserved
12	RW	0x0	clk_i2s0_tx_rx_clk_sel 1'b0: Select clk_i2s0_tx_clk 1'b1: Select clk_i2s0_rx_clk
11:10	RW	0x0	clk_i2s0_tx_sel 2'b00: Select clk_i2s0_tx 2'b01: Select clk_i2s0_tx_frac_out 2'b10: Select mclk_i2s0_tx_in 2'b11: Select xin_osc0_half
9:8	RW	0x0	clk_i2s0_tx_pll_sel 2'b00: GPLL 2'b01: CPLL 2'b10: NPLL 2'b11: Reserved
7	RO	0x0	Reserved
6:0	RW	0x0b	clk_i2s0_tx_div_con clk_i2s0_tx=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON33

Address: Operational Base + offset (0x0184)

Bit	Attr	Reset Value	Description
31:0	RW	0x0bb8ea60	clk_i2s0_tx_frac_div_con High 16-bit for numerator, Low 16-bit for denominator, clock source is clk_i2s0_tx

CRU_CLKSEL_CON34

Address: Operational Base + offset (0x0188)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15:14	RW	0x0	clk_i2s0_rx_out_mclk_sel 2'b00: Select selected clock by clk_i2s0_rx_tx_clk_sel 2'b01: Select xin_osc0_half 2'b10: Select clk_i2s0_tx 2'b11: Reserved
13	RO	0x0	Reserved
12	RW	0x0	clk_i2s0_rx_tx_clk_sel 1'b0: Select clk_i2s0_rx_clk 1'b1: Select clk_i2s0_tx_clk
11:10	RW	0x0	clk_i2s0_rx_sel 2'b00: Select clk_i2s0_rx 2'b01: Select clk_i2s0_rx_frac_out 2'b10: Select mclk_i2s0_rx_in 2'b11: Select xin_osc0_half
9:8	RW	0x0	clk_i2s0_rx_pll_sel 2'b00: GPLL 2'b01: CPLL 2'b10: NPLL 2'b11: Reserved
7	RO	0x0	Reserved
6:0	RW	0x0b	clk_i2s0_rx_div_con clk_i2s0_rx=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON35

Address: Operational Base + offset (0x018c)

Bit	Attr	Reset Value	Description
31:0	RW	0x0bb8ea60	clk_i2s0_rx_frac_div_con High 16-bit for numerator, Low 16-bit for denominator, clock source is clk_i2s0_rx

CRU_CLKSEL_CON36

Address: Operational Base + offset (0x0190)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	clk_i2s1_out_mclk_sel 1'b0: Select selected clock by clk_i2s1_sel 1'b1: Select xin_osc0_half
14:12	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
11:10	RW	0x0	clk_i2s1_sel 2'b00: Select clk_i2s1 2'b01: Select clk_i2s1_frac_out 2'b10: Select mclk_i2s1_in 2'b11: Select xin_osc0_half
9:8	RW	0x0	clk_i2s1_pll_sel 2'b00: GPLL 2'b01: CPLL 2'b10: NPLL 2'b11: Reserved
7	RO	0x0	Reserved
6:0	RW	0x0b	clk_i2s1_div_con clk_i2s1=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON37

Address: Operational Base + offset (0x0194)

Bit	Attr	Reset Value	Description
31:0	RW	0x0bb8ea60	clk_i2s1_frac_div_con High 16-bit for numerator, Low 16-bit for denominator, clock source is clk_i2s1

CRU_CLKSEL_CON38

Address: Operational Base + offset (0x0198)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	clk_uart1_pll_sel 2'b00: GPLL 2'b01: USBPHY480M 2'b10: CPLL 2'b11: NPLL
13:7	RO	0x0	Reserved
6:0	RW	0x0b	clk_uart1_div_con clk_uart1=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON39

Address: Operational Base + offset (0x019c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x3	clk_uart1_sel 2'b00: Select clk_uart1 2'b01: Select clk_uart1_np5 2'b10: Select clk_uart1_frac_out 2'b11: Select xin_osc0
13:7	RO	0x0	Reserved
6:0	RW	0x0b	clk_uart1_divnp5_div_con clk_uart1_np5=2*clk_uart1/(2*div_con+3)

CRU_CLKSEL_CON40

Address: Operational Base + offset (0x01a0)

Bit	Attr	Reset Value	Description
31:0	RW	0x0bb8ea60	clk_uart1_frac_div_con High 16-bit for numerator, Low 16-bit for denominator, clock source is clk_uart1

CRU_CLKSEL_CON41

Address: Operational Base + offset (0x01a4)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	clk_uart2_pll_sel 2'b00: GPLL 2'b01: USBPHY480M 2'b10: CPLL 2'b11: NPLL
13:7	RO	0x0	Reserved
6:0	RW	0x0b	clk_uart2_div_con clk_uart2=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON42

Address: Operational Base + offset (0x01a8)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x3	clk_uart2_sel 2'b00: Select clk_uart2 2'b01: Select clk_uart2_np5 2'b10: Select clk_uart2_frac_out 2'b11: Select xin_osc0
13:7	RO	0x0	Reserved
6:0	RW	0x0b	clk_uart2_divnp5_div_con clk_uart2_np5=2*clk_uart2/(2*div_con+3)

CRU_CLKSEL_CON43

Address: Operational Base + offset (0x01ac)

Bit	Attr	Reset Value	Description
31:0	RW	0x0bb8ea60	clk_uart2_frac_div_con High 16-bit for numerator, Low 16-bit for denominator, clock source is clk_uart2

CRU_CLKSEL_CON44

Address: Operational Base + offset (0x01b0)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	clk_uart3_pll_sel 2'b00: GPLL 2'b01: USBPHY480M 2'b10: CPLL 2'b11: NPLL
13:7	RO	0x0	Reserved
6:0	RW	0x0b	clk_uart3_div_con clk_uart3=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON45

Address: Operational Base + offset (0x01b4)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x3	clk_uart3_sel 2'b00: Select clk_uart3 2'b01: Select clk_uart3_np5 2'b10: Select clk_uart3_frac_out 2'b11: Select xin_osc0
13:7	RO	0x0	Reserved
6:0	RW	0x0b	clk_uart3_divnp5_div_con clk_uart3_np5=2*clk_uart3/(2*div_con+3)

CRU_CLKSEL_CON46

Address: Operational Base + offset (0x01b8)

Bit	Attr	Reset Value	Description
31:0	RW	0x0bb8ea60	clk_uart3_frac_div_con High 16-bit for numerator, Low 16-bit for denominator, clock source is clk_uart3

CRU_CLKSEL_CON47

Address: Operational Base + offset (0x01bc)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	clk_uart4_pll_sel 2'b00: GPLL 2'b01: USBPHY480M 2'b10: CPLL 2'b11: NPLL
13:7	RO	0x0	Reserved
6:0	RW	0x0b	clk_uart4_div_con clk_uart4=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON48

Address: Operational Base + offset (0x01c0)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x3	clk_uart4_sel 2'b00: Select clk_uart4 2'b01: Select clk_uart4_np5 2'b10: Select clk_uart4_frac_out 2'b11: Select xin_osc0
13:7	RO	0x0	Reserved
6:0	RW	0x0b	clk_uart4_divnp5_div_con clk_uart4_np5=2*clk_uart4/(2*div_con+3)

CRU_CLKSEL_CON49

Address: Operational Base + offset (0x01c4)

Bit	Attr	Reset Value	Description
31:0	RW	0x0bb8ea60	clk_uart4_frac_div_con High 16-bit for numerator, Low 16-bit for denominator, clock source is clk_uart4

CRU_CLKSEL_CON50

Address: Operational Base + offset (0x01c8)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	clk_uart5_pll_sel 2'b00: GPLL 2'b01: USBPHY480M 2'b10: CPLL 2'b11: NPLL
13:7	RO	0x0	Reserved
6:0	RW	0x0b	clk_uart5_div_con clk_uart5=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON51

Address: Operational Base + offset (0x01cc)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x3	clk_uart5_sel 2'b00: Select clk_uart5 2'b01: Select clk_uart5_np5 2'b10: Select clk_uart5_frac_out 2'b11: Select xin_osc0
13:7	RO	0x0	Reserved
6:0	RW	0x0b	clk_uart5_divnp5_div_con clk_uart5_np5=2*clk_uart5/(2*div_con+3)

CRU_CLKSEL_CON52

Address: Operational Base + offset (0x01d0)

Bit	Attr	Reset Value	Description
31:0	RW	0x0bb8ea60	clk_uart5_frac_div_con High 16-bit for numerator, Low 16-bit for denominator, clock source is clk_uart5

CRU_CLKSEL_CON53

Address: Operational Base + offset (0x01d4)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	clk_uart6_pll_sel 2'b00: GPLL 2'b01: USBPHY480M 2'b10: CPLL 2'b11: NPLL
13:7	RO	0x0	Reserved
6:0	RW	0x0b	clk_uart6_div_con clk_uart6=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON54

Address: Operational Base + offset (0x01d8)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x3	clk_uart6_sel 2'b00: Select clk_uart6 2'b01: Select clk_uart6_np5 2'b10: Select clk_uart6_frac_out 2'b11: Select xin_osc0
13:7	RO	0x0	Reserved
6:0	RW	0x0b	clk_uart6_divnp5_div_con clk_uart6_np5=2*clk_uart6/(2*div_con+3)

CRU_CLKSEL_CON55

Address: Operational Base + offset (0x01dc)

Bit	Attr	Reset Value	Description
31:0	RW	0x0bb8ea60	clk_uart6_frac_div_con High 16-bit for numerator, Low 16-bit for denominator, clock source is clk_uart6

CRU_CLKSEL_CON56

Address: Operational Base + offset (0x01e0)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	clk_uart7_pll_sel 2'b00: GPLL 2'b01: USBPHY480M 2'b10: CPLL 2'b11: NPLL
13:7	RO	0x0	Reserved
6:0	RW	0x0b	clk_uart7_div_con clk_uart7=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON57

Address: Operational Base + offset (0x01e4)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15:14	RW	0x3	clk_uart7_sel 2'b00: Select clk_uart7 2'b01: Select clk_uart7_np5 2'b10: Select clk_uart7_frac_out 2'b11: Select xin_osc0
13:7	RO	0x0	Reserved
6:0	RW	0x0b	clk_uart7_divnp5_div_con clk_uart7_np5=2*clk_uart7/(2*div_con+3)

CRU_CLKSEL_CON58

Address: Operational Base + offset (0x01e8)

Bit	Attr	Reset Value	Description
31:0	RW	0x0bb8ea60	clk_uart7_frac_div_con High 16-bit for numerator, Low 16-bit for denominator, clock source is clk_uart7

CRU_CLKSEL_CON59

Address: Operational Base + offset (0x01ec)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	clk_i2c2_pll_sel 1'b0: Select GPLL 1'b1: Select xin_osc0
14:8	RW	0x05	clk_i2c2_div_con clk_i2c2=pll_clk_src/(div_con+1)
7	RW	0x0	clk_i2c1_pll_sel 1'b0: Select GPLL 1'b1: Select xin_osc0
6:0	RW	0x05	clk_i2c1_div_con clk_i2c1=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON60

Address: Operational Base + offset (0x01f0)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	clk_spi0_pll_sel 1'b0: Select GPLL 1'b1: Select xin_osc0

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
14:8	RW	0x03	clk_spi0_div_con clk_spi0=pll_clk_src/(div_con+1)
7	RW	0x0	clk_i2c3_pll_sel 1'b0: Select GPLL 1'b1: Select xin_osc0
6:0	RW	0x05	clk_i2c3_div_con clk_i2c3=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON61

Address: Operational Base + offset (0x01f4)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	clk_spi2_pll_sel 1'b0: Select GPLL 1'b1: Select xin_osc0
14:8	RW	0x0b	clk_spi2_div_con clk_spi2=pll_clk_src/(div_con+1)
7	RW	0x0	clk_spi1_pll_sel 1'b0: Select GPLL 1'b1: Select xin_osc0
6:0	RW	0x0b	clk_spi1_div_con clk_spi1=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON62

Address: Operational Base + offset (0x01f8)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:11	RO	0x0	Reserved
10:0	RW	0x01f	clk_tsadc_div_con clk_tsadc=xin_osc0/(div_con+1)

CRU_CLKSEL_CON63

Address: Operational Base + offset (0x01fc)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15:11	RO	0x0	Reserved
10:0	RW	0x017	clk_saradc_div_con clk_saradc=xin_osc0/(div_con+1)

CRU_CLKSEL_CON64

Address: Operational Base + offset (0x0200)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	clk_efuse_ns_pll_clk_sel 2'b00: GPLL 2'b01: CPLL 2'b10: Select xin_osc0 2'b11: Reserved
13:8	RW	0x0b	clk_efuse_ns_div_con clk_efuse_ns=pll_clk_src/(div_con+1)
7:6	RW	0x0	clk_efuse_s_pll_clk_sel 2'b00: GPLL 2'b01: CPLL 2'b10: Select xin_osc0 2'b11: Reserved
5:0	RW	0x0b	clk_efuse_s_div_con clk_efuse_s=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON65

Address: Operational Base + offset (0x0204)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	dbclk_gpio1_pll_sel 1'b0: Select xin_osc0 1'b1: Select clk_rtc_32k
14:11	RO	0x0	Reserved
10:0	RW	0x001	dbclk_gpio1_div_con dbclk_gpio1=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON66

Address: Operational Base + offset (0x0208)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	dbclk_gpio2_pll_sel 1'b0: Select xin_osc0 1'b1: Select clk_rtc_32k
14:11	RO	0x0	Reserved
10:0	RW	0x001	dbclk_gpio2_div_con dbclk_gpio2=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON67

Address: Operational Base + offset (0x020c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	dbclk_gpio3_pll_sel 1'b0: Select xin_osc0 1'b1: Select clk_rtc_32k
14:11	RO	0x0	Reserved
10:0	RW	0x001	dbclk_gpio3_div_con dbclk_gpio3=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON68

Address: Operational Base + offset (0x0210)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	dbclk_gpio4_pll_sel 1'b0: Select xin_osc0 1'b1: Select clk_rtc_32k
14:11	RO	0x0	Reserved
10:0	RW	0x001	dbclk_gpio4_div_con dbclk_gpio4=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON69

Address: Operational Base + offset (0x0214)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	clk_pwm1_pll_sel 1'b0: Select GPLL 1'b1: Select xin_osc0
14:8	RW	0x0b	clk_pwm1_div_con clk_pwm1=pll_clk_src/(div_con+1)
7	RW	0x0	clk_pwm0_pll_sel 1'b0: Select GPLL 1'b1: Select xin_osc0
6:0	RW	0x0b	clk_pwm0_div_con clk_pwm0=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON70

Address: Operational Base + offset (0x0218)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	Reserved
7	RW	0x0	clk_pwm2_pll_sel 1'b0: Select GPLL 1'b1: Select xin_osc0
6:0	RW	0x0b	clk_pwm2_div_con clk_pwm2=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON71

Address: Operational Base + offset (0x021c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	clk_i2c5_pll_sel 1'b0: Select GPLL 1'b1: Select xin_osc0
14:8	RW	0x05	clk_i2c5_div_con clk_i2c5=pll_clk_src/(div_con+1)
7	RW	0x0	clk_i2c4_pll_sel 1'b0: Select GPLL 1'b1: Select xin_osc0

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
6:0	RW	0x05	clk_i2c4_div_con clk_i2c4=pll_clk_src/(div_con+1)

CRU_CLKSEL_CON72

Address: Operational Base + offset (0x0220)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:13	RO	0x0	Reserved
12:8	RW	0x1f	test_div_con clk_test_out=test_clk_src/(div_con+1)
7:5	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4:0	RW	0x00	testclk_sel 5'h00: xin_osc0 5'h01: clk_rtc_32k 5'h02: clk_core 5'h03: aclk_gic 5'h04: clk_npu 5'h05: aclk_npu 5'h06: ddrphy1x 5'h07: clk_isp 5'h08: hsclk_pcie 5'h09: clk_pcie_pipe 5'h0a: hsclk_system_sram_niu 5'h0b: aclk_system_sram 5'h0c: hsclk_vio 5'h0d: clk_rga_core 5'h0e: dclk_vopraw 5'h0f: dclk_voplite 5'h10: msclk_peri 5'h11: clk_spi0 5'h12: clk_gmac 5'h13: clk_sdmmc 5'h14: clk_sdio 5'h15: clk_emmc 5'h16: hsclk_bus 5'h17: clk_crypto_pka 5'h18: clk_uart1 5'h19: dclk_cif 5'h1a: aclk_vpu 5'h1b: clk_crypto_core 5'h1c: clk_i2s0_rx 5'h1d: clk_i2s1 5'h1e: clk_pdm 5'h1f: clk_i2s0_tx

CRU_CLKGATE_CON0

Address: Operational Base + offset (0x0230)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	clk_pvtm_npu_en When HIGH, disable clock
14:13	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
12	RW	0x0	pclk_core_grf_en When HIGH, disable clock
11	RW	0x0	aclk_core_prf_en When HIGH, disable clock
10	RW	0x0	aclk_adb400_core2gic_en When HIGH, disable clock
9	RW	0x0	aclk_adb400_gic2core_en When HIGH, disable clock
8	RW	0x0	pclk_dbg_daplite_en When HIGH, disable clock
7	RW	0x0	pclk_dbg_niu_en When HIGH, disable clock
6	RW	0x0	aclk_core_niu_en When HIGH, disable clock
5	RW	0x0	clk_pvtm_core_en When HIGH, disable clock
4	RW	0x0	clk_jtag_en When HIGH, disable clock
3	RW	0x0	pclk_core_en When HIGH, disable clock
2	RW	0x0	aclk_core_en When HIGH, disable clock
1	RW	0x0	msclk_core_niu_clk_en When HIGH, disable clock
0	RW	0x0	core_pll_clk_en When HIGH, disable clock

CRU_CLKGATE_CON1

Address: Operational Base + offset (0x0234)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	aclk_npu2mem_en When HIGH, disable clock
14	RW	0x0	hclk_npu_niu_en When HIGH, disable clock
13	RW	0x0	aclk_npu_niu_en When HIGH, disable clock
12	RW	0x0	hclk_npu_en When HIGH, disable clock
11	RW	0x0	aclk_npu_en When HIGH, disable clock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
10	RW	0x0	clk_npu_core_en When HIGH, disable clock
9	RW	0x0	npu_pll_hclk_en When HIGH, disable clock
8	RW	0x0	npu_pll_aclk_en When HIGH, disable clock
7	RW	0x0	npu_np5_pll_clk_en When HIGH, disable clock
6	RW	0x0	npu_pll_clk_en When HIGH, disable clock
5	RW	0x0	aclk_spinlock_en When HIGH, disable clock
4	RW	0x0	aclk_adb400_gic2core_en When HIGH, disable clock
3	RW	0x0	aclk_adb400_core2gic_en When HIGH, disable clock
2	RW	0x0	aclk_gic500_en When HIGH, disable clock
1	RW	0x0	aclk_gic_niu_en When HIGH, disable clock
0	RW	0x0	gic_pll_clk_en When HIGH, disable clock

CRU_CLKGATE_CON2

Address: Operational Base + offset (0x0238)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	aclk_axi_split_en When HIGH, disable clock
14	RW	0x0	pclk_ddrgrf_en When HIGH, disable clock
13	RW	0x0	clk_ddrstdby_en When HIGH, disable clock
12	RW	0x0	pclk_ddrstdby_en When HIGH, disable clock
11	RW	0x0	clk_ddrc_mon_en When HIGH, disable clock
10	RW	0x0	pclk_ddr_mon_en When HIGH, disable clock
9	RW	0x0	pclk_ddr_msch_en When HIGH, disable clock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
8	RW	0x0	clk_ddr_msch_en When HIGH, disable clock
7	RW	0x0	pclk_ddr_upctl_en When HIGH, disable clock
6	RW	0x0	clk_ddr_upctl_en When HIGH, disable clock
5	RW	0x0	aclk_ddr_upctl_en When HIGH, disable clock
4	RO	0x0	Reserved
3	RW	0x0	clk_ddrdfi_ctl_en When HIGH, disable clock
2	RW	0x0	pclk_ddrdfi_ctl_en When HIGH, disable clock
1	RW	0x0	pclk_ddr_pll_clk_en When HIGH, disable clock
0	RW	0x0	clk_ddrmon24_en When HIGH, disable clock

CRU_CLKGATE_CON3

Address: Operational Base + offset (0x023c)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RO	0x0	Reserved
13	RW	0x0	pclkin_cif_en When HIGH, disable clock
12	RW	0x0	lsclk_vio_src_en When HIGH, disable clock
11	RW	0x0	dclk_cif_pll_clk_en When HIGH, disable clock
10	RW	0x0	clk_isp_pll_clk_en When HIGH, disable clock
9	RW	0x0	clk_cifout_pll_clk_en When HIGH, disable clock
8	RW	0x0	clk_rga_pll_clk_en When HIGH, disable clock
7	RW	0x0	clk_txesc_pll_en When HIGH, disable clock
6	RW	0x0	dclkvoplite_clk_en When HIGH, disable clock
5	RW	0x0	dclkvoplite_fracdiv_clk_en When HIGH, disable clock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4	RW	0x0	dclkvoplite_pll_clk_en When HIGH, disable clock
3	RW	0x0	dclkvopraw_clk_en When HIGH, disable clock
2	RW	0x0	dclkvopraw_fracdiv_clk_en When HIGH, disable clock
1	RW	0x0	dclkvopraw_pll_clk_en When HIGH, disable clock
0	RW	0x0	hsclk_vio_pll_clk_en When HIGH, disable clock

CRU_CLKGATE_CON4

Address: Operational Base + offset (0x0240)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RO	0x0	Reserved
14	RW	0x0	hclk_isp_en When HIGH, disable clock
13	RW	0x0	aclk_isp_en When HIGH, disable clock
12	RW	0x0	pclk_csirx_en When HIGH, disable clock
11	RW	0x0	hclk_cif_en When HIGH, disable clock
10	RW	0x0	aclk_cif_en When HIGH, disable clock
9	RW	0x0	hclk_rga_en When HIGH, disable clock
8	RW	0x0	aclk_rga_en When HIGH, disable clock
7	RW	0x0	pclk_csitx_en When HIGH, disable clock
6	RW	0x0	pclk_mipi_dsi_en When HIGH, disable clock
5	RW	0x0	hclk_voplite_en When HIGH, disable clock
4	RW	0x0	aclk_voplite_en When HIGH, disable clock
3	RW	0x0	hclk_vopraw_en When HIGH, disable clock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2	RW	0x0	aclk_vopraw_en When HIGH, disable clock
1	RW	0x0	lsclk_vio_niu_en When HIGH, disable clock
0	RW	0x0	hsclk_vio_niu_en When HIGH, disable clock

CRU_CLKGATE_CON5

Address: Operational Base + offset (0x0244)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:7	RO	0x0	Reserved
6	RW	0x0	pclk_pcie_src_clk_en When HIGH, disable clock
5	RW	0x0	aclk_pcie_pll_clk_en When HIGH, disable clock
4	RW	0x0	clk_pcie_aux_src_clk_en When HIGH, disable clock
3	RW	0x0	clk_pcie_aux_pll_clk_en When HIGH, disable clock
2	RW	0x0	clk_usb3otg_suspend_clk_en When HIGH, disable clock
1	RW	0x0	clk_usb3otg_ref_clk_en When HIGH, disable clock
0	RW	0x0	pdclk_pcie_pll_clk_en When HIGH, disable clock

CRU_CLKGATE_CON6

Address: Operational Base + offset (0x0248)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RO	0x0	Reserved
11	RW	0x0	pclk_pcie_niu_en When HIGH, disable clock
10	RW	0x0	aclk_pcie_niu_en When HIGH, disable clock
9	RW	0x0	pclk_pcie_en When HIGH, disable clock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
8	RW	0x0	hclk_usb2host_arb_en When HIGH, disable clock
7	RW	0x0	hclk_usb2host_en When HIGH, disable clock
6	RW	0x0	aclk_usb3otg_en When HIGH, disable clock
5	RW	0x0	pclk_pcie_grf_en When HIGH, disable clock
4	RW	0x0	aclk_pcie_dbi_en When HIGH, disable clock
3	RW	0x0	aclk_pcie_slv_en When HIGH, disable clock
2	RW	0x0	aclk_pcie_mst_en When HIGH, disable clock
1	RW	0x0	lsclk_pcie_niu_en When HIGH, disable clock
0	RW	0x0	hsclk_pcie_niu_en When HIGH, disable clock

CRU_CLKGATE_CON7

Address: Operational Base + offset (0x024c)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RO	0x0	Reserved
13	RW	0x0	aclk_system_sram_3_niu_en When HIGH, disable clock
12	RW	0x0	aclk_system_sram_2_niu_en When HIGH, disable clock
11	RW	0x0	aclk_system_sram_1_niu_en When HIGH, disable clock
10	RW	0x0	aclk_system_sram_0_niu_en When HIGH, disable clock
9	RW	0x0	aclk_system_sram_3_en When HIGH, disable clock
8	RW	0x0	aclk_system_sram_2_en When HIGH, disable clock
7	RW	0x0	aclk_system_sram_1_en When HIGH, disable clock
6	RW	0x0	aclk_system_sram_0_en When HIGH, disable clock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5	RW	0x0	system_sram_hsclk_pll_clk_en When HIGH, disable clock
4	RW	0x0	system_sram_3_src_clk_en When HIGH, disable clock
3	RW	0x0	system_sram_2_src_clk_en When HIGH, disable clock
2	RW	0x0	system_sram_1_src_clk_en When HIGH, disable clock
1	RW	0x0	system_sram_0_src_clk_en When HIGH, disable clock
0	RW	0x0	system_sram_pll_clk_en When HIGH, disable clock

CRU_CLKGATE_CON8

Address: Operational Base + offset (0x0250)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RO	0x0	Reserved
13	RW	0x0	hclk_vpu_en When HIGH, disable clock
12	RW	0x0	aclk_vpu_en When HIGH, disable clock
11	RW	0x0	hclk_vpu_niu_en When HIGH, disable clock
10	RW	0x0	aclk_npu_niu_en When HIGH, disable clock
9	RW	0x0	hclk_vpu_src_en When HIGH, disable clock
8	RW	0x0	aclk_vpu_pll_clk_en When HIGH, disable clock
7	RW	0x0	ddrphy1x_clk_en When HIGH, disable clock
6	RW	0x0	ddrphy_gpll_clk_en When HIGH, disable clock
5	RW	0x0	ddrphy_dpll_clk_en When HIGH, disable clock
4	RW	0x0	lsclk_peri_niu_en When HIGH, disable clock
3	RW	0x0	msclk_peri_niu_en When HIGH, disable clock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2	RW	0x0	lsclk_peri_clk_en When HIGH, disable clock
1	RW	0x0	msclk_peri_clk_en When HIGH, disable clock
0	RW	0x0	pd_peri_pll_clk_en When HIGH, disable clock

CRU_CLKGATE_CON9

Address: Operational Base + offset (0x0254)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RO	0x0	Reserved
13	RW	0x0	hclk_sfc_clk_en When HIGH, disable clock
12	RW	0x0	hclk_emmc_clk_en When HIGH, disable clock
11	RW	0x0	hclk_mmc_sfc_niu_en When HIGH, disable clock
10	RW	0x0	clk_sfc_pll_clk_en When HIGH, disable clock
9	RW	0x0	clk_sdmmc_clk_en When HIGH, disable clock
8	RW	0x0	clk_sdmmc_div50_clk_en When HIGH, disable clock
7	RW	0x0	clk_sdmmc_pll_clk_en When HIGH, disable clock
6	RW	0x0	clk_emmc_clk_en When HIGH, disable clock
5	RW	0x0	clk_emmc_div50_clk_en When HIGH, disable clock
4	RW	0x0	clk_emmc_pll_clk_en When HIGH, disable clock
3	RW	0x0	clk_sdio_clk_en When HIGH, disable clock
2	RW	0x0	clk_sdio_div50_clk_en When HIGH, disable clock
1	RW	0x0	clk_sdio_pll_clk_en When HIGH, disable clock
0	RW	0x0	hclk_mmc_sfc_clk_en When HIGH, disable clock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

CRU_CLKGATE_CON10

Address: Operational Base + offset (0x0258)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	clk_gmac_out_pll_clk_en When HIGH, disable clock
14	RW	0x0	hclk_sdmmc_clk_en When HIGH, disable clock
13	RW	0x0	hclk_sdio_clk_en When HIGH, disable clock
12	RW	0x0	pclk_gmac_en When HIGH, disable clock
11	RW	0x0	aclk_gmac_en When HIGH, disable clock
10	RW	0x0	pclk_gmac_niu_en When HIGH, disable clock
9	RW	0x0	hclk_gmac_niu_en When HIGH, disable clock
8	RW	0x0	aclk_gmac_niu_en When HIGH, disable clock
7	RW	0x0	clk_mac_tx_en When HIGH, disable clock
6	RW	0x0	clk_mac_rx_en When HIGH, disable clock
5	RW	0x0	clk_mac_refout_en When HIGH, disable clock
4	RW	0x0	clk_mac_ref_en When HIGH, disable clock
3	RW	0x0	clk_gmac_pll_clk_en When HIGH, disable clock
2	RW	0x0	pclk_sd_gmac_clk_en When HIGH, disable clock
1	RW	0x0	hclk_sd_gmac_clk_en When HIGH, disable clock
0	RW	0x0	aclk_sd_gmac_clk_en When HIGH, disable clock

CRU_CLKGATE_CON11

Address: Operational Base + offset (0x025c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	clk_uar2_clk_en When HIGH, disable clock
14	RW	0x0	clk_uart2_frac_src_clk_en When HIGH, disable clock
13	RW	0x0	clk_uart2_divnp5_clk_en When HIGH, disable clock
12	RW	0x0	clk_uart2_pll_clk_en When HIGH, disable clock
11	RW	0x0	clk_uart1_clk_en When HIGH, disable clock
10	RW	0x0	clk_uart1_frac_src_clk_en When HIGH, disable clock
9	RW	0x0	clk_uart1_divnp5_clk_en When HIGH, disable clock
8	RW	0x0	clk_uart1_pll_clk_en When HIGH, disable clock
7	RO	0x0	Reserved
6	RW	0x0	clk_crypto_pka_pll_clk_en When HIGH, disable clock
5	RW	0x0	clk_crypto_pll_clk_en When HIGH, disable clock
4	RW	0x0	pclk_top_clk_en When HIGH, disable clock
3	RW	0x0	lsclk_bus_clk_en When HIGH, disable clock
2	RW	0x0	msclk_bus_clk_en When HIGH, disable clock
1	RW	0x0	hsclk_bus_clk_en When HIGH, disable clock
0	RW	0x0	pd_bus_pll_clk_en When HIGH, disable clock

CRU_CLKGATE_CON12

Address: Operational Base + offset (0x0260)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	clk_uart6_clk_en When HIGH, disable clock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
14	RW	0x0	clk_uart6_frac_src_clk_en When HIGH, disable clock
13	RW	0x0	clk_uart6_divnp5_clk_en When HIGH, disable clock
12	RW	0x0	clk_uart6_pll_clk_en When HIGH, disable clock
11	RW	0x0	clk_uart5_clk_en When HIGH, disable clock
10	RW	0x0	clk_uart5_frac_src_clk_en When HIGH, disable clock
9	RW	0x0	clk_uart5_divnp5_clk_en When HIGH, disable clock
8	RW	0x0	clk_uart5_pll_clk_en When HIGH, disable clock
7	RW	0x0	clk_uart4_clk_en When HIGH, disable clock
6	RW	0x0	clk_uart4_frac_src_clk_en When HIGH, disable clock
5	RW	0x0	clk_uart4_divnp5_clk_en When HIGH, disable clock
4	RW	0x0	clk_uart4_pll_clk_en When HIGH, disable clock
3	RW	0x0	clk_uart3_clk_en When HIGH, disable clock
2	RW	0x0	clk_uart3_frac_src_clk_en When HIGH, disable clock
1	RW	0x0	clk_uart3_divnp5_clk_en When HIGH, disable clock
0	RW	0x0	clk_uart3_pll_clk_en When HIGH, disable clock

CRU_CLKGATE_CON13

Address: Operational Base + offset (0x0264)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RO	0x0	Reserved
14	RW	0x0	clk_saradc_pll_clk_en When HIGH, disable clock
13	RW	0x0	clk_tsadc_pll_clk_en When HIGH, disable clock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
12	RW	0x0	clk_pwm2_pll_clk_en When HIGH, disable clock
11	RW	0x0	clk_pwm1_pll_clk_en When HIGH, disable clock
10	RW	0x0	clk_pwm0_pll_clk_en When HIGH, disable clock
9	RW	0x0	clk_spi2_pll_clk_en When HIGH, disable clock
8	RW	0x0	clk_spi1_pll_clk_en When HIGH, disable clock
7	RW	0x0	clk_spi0_pll_clk_en When HIGH, disable clock
6	RW	0x0	clk_i2c3_pll_clk_en When HIGH, disable clock
5	RW	0x0	clk_i2c2_pll_clk_en When HIGH, disable clock
4	RW	0x0	clk_i2c1_pll_clk_en When HIGH, disable clock
3	RW	0x0	clk_uart7_clk_en When HIGH, disable clock
2	RW	0x0	clk_uart7_frac_src_clk_en When HIGH, disable clock
1	RW	0x0	clk_uart7_divnp5_clk_en When HIGH, disable clock
0	RW	0x0	clk_uart7_pll_clk_en When HIGH, disable clock

CRU_CLKGATE_CON14

Address: Operational Base + offset (0x0268)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RO	0x0	Reserved
13	RW	0x0	clk_timer5_clk_en When HIGH, disable clock
12	RW	0x0	clk_timer4_clk_en When HIGH, disable clock
11	RW	0x0	clk_timer3_clk_en When HIGH, disable clock
10	RW	0x0	clk_timer2_clk_en When HIGH, disable clock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
9	RW	0x0	clk_timer1_clk_en When HIGH, disable clock
8	RW	0x0	clk_timer0_clk_en When HIGH, disable clock
7	RW	0x0	clk_i2c5_pll_clk_en When HIGH, disable clock
6	RW	0x0	clk_i2c4_pll_clk_en When HIGH, disable clock
5	RW	0x0	dbclk_gpio4_pll_clk_en When HIGH, disable clock
4	RW	0x0	dbclk_gpio3_pll_clk_en When HIGH, disable clock
3	RW	0x0	dbclk_gpio2_pll_clk_en When HIGH, disable clock
2	RW	0x0	dbclk_gpio1_pll_clk_en When HIGH, disable clock
1	RW	0x0	clk_efuse_ns_pll_clk_en When HIGH, disable clock
0	RW	0x0	clk_efuse_s_pll_clk_en When HIGH, disable clock

CRU_CLKGATE_CON15

Address: Operational Base + offset (0x026c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	pclk_uart7_clk_en When HIGH, disable clock
14	RW	0x0	pclk_uart6_clk_en When HIGH, disable clock
13	RW	0x0	pclk_uart5_clk_en When HIGH, disable clock
12	RW	0x0	pclk_uart4_clk_en When HIGH, disable clock
11	RW	0x0	pclk_uart3_clk_en When HIGH, disable clock
10	RW	0x0	pclk_uart2_clk_en When HIGH, disable clock
9	RW	0x0	pclk_uart1_clk_en When HIGH, disable clock
8	RW	0x0	pclk_dcf_clk_en When HIGH, disable clock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7	RW	0x0	aclk_dcf_clk_en When HIGH, disable clock
6	RW	0x0	hclk_crypto_en When HIGH, disable clock
5	RW	0x0	aclk_crypto_en When HIGH, disable clock
4	RW	0x0	hclk_rom_clk_en When HIGH, disable clock
3	RW	0x0	lsclk_bus_niu_en When HIGH, disable clock
2	RW	0x0	msclk_bus_sub_niu_clk_en When HIGH, disable clock
1	RW	0x0	msclk_bus_niu_clk_en When HIGH, disable clock
0	RW	0x0	hsclk_bus_niu_clk_en When HIGH, disable clock

CRU_CLKGATE_CON16

Address: Operational Base + offset (0x0270)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	pclk_gpio4_clk_en When HIGH, disable clock
14	RW	0x0	pclk_gpio3_clk_en When HIGH, disable clock
13	RW	0x0	pclk_gpio2_clk_en When HIGH, disable clock
12	RW	0x0	pclk_gpio1_clk_en When HIGH, disable clock
11	RW	0x0	pclk_efuse_ns_clk_en When HIGH, disable clock
10	RW	0x0	pclk_saradc_en When HIGH, disable clock
9	RW	0x0	pclk_tsadc_en When HIGH, disable clock
8	RW	0x0	pclk_pwm2_clk_en When HIGH, disable clock
7	RW	0x0	pclk_pwm1_clk_en When HIGH, disable clock
6	RW	0x0	pclk_pwm0_clk_en When HIGH, disable clock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5	RW	0x0	pclk_spi2_en When HIGH, disable clock
4	RW	0x0	pclk_spi1_en When HIGH, disable clock
3	RW	0x0	pclk_spi0_en When HIGH, disable clock
2	RW	0x0	pclk_i2c3_clk_en When HIGH, disable clock
1	RW	0x0	pclk_i2c2_clk_en When HIGH, disable clock
0	RW	0x0	pclk_i2c1_clk_en When HIGH, disable clock

CRU_CLKGATE_CON17

Address: Operational Base + offset (0x0274)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	clk_i2s0_tx_out_mclk_en When HIGH, disable clock
14	RW	0x0	clk_i2s0_tx_clk_en When HIGH, disable clock
13	RW	0x0	clk_i2s0_tx_frac_src_clk_en When HIGH, disable clock
12	RW	0x0	clk_i2s0_tx_pll_clk_en When HIGH, disable clock
11	RW	0x0	clk_pdm_clk_en When HIGH, disable clock
10	RW	0x0	clk_pdm_frac_src_clk_en When HIGH, disable clock
9	RW	0x0	clk_pdm_pll_clk_en When HIGH, disable clock
8	RW	0x0	hclk_pd_audio_clk_en When HIGH, disable clock
7:6	RO	0x0	Reserved
5	RW	0x0	pclk_i2c5_clk_en When HIGH, disable clock
4	RW	0x0	pclk_i2c4_clk_en When HIGH, disable clock
3	RW	0x0	pclk_bus_sgrf_clk_en When HIGH, disable clock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2	RW	0x0	pclk_bus_grf_clk_en When HIGH, disable clock
1	RW	0x0	pclk_wdt_ns_clk_en When HIGH, disable clock
0	RW	0x0	pclk_timer0_en When HIGH, disable clock

CRU_CLKGATE_CON18

Address: Operational Base + offset (0x0278)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	hclk_i2s1_en When HIGH, disable clock
14	RW	0x0	hclk_i2s0_en When HIGH, disable clock
13	RW	0x0	hclk_pdm_en When HIGH, disable clock
12	RW	0x0	hclk_vad_en When HIGH, disable clock
11	RW	0x0	hclk_audio_niu_en When HIGH, disable clock
10:8	RO	0x0	Reserved
7	RW	0x0	clk_i2s1_out_mclk_en When HIGH, disable clock
6	RW	0x0	clk_i2s1_clk_en When HIGH, disable clock
5	RW	0x0	clk_i2s1_frac_src_clk_en When HIGH, disable clock
4	RW	0x0	clk_i2s1_pll_clk_en When HIGH, disable clock
3	RW	0x0	clk_i2s0_rx_out_mclk_en When HIGH, disable clock
2	RW	0x0	clk_i2s0_rx_clk_en When HIGH, disable clock
1	RW	0x0	clk_i2s0_rx_frac_src_clk_en When HIGH, disable clock
0	RW	0x0	clk_i2s0_rx_pll_clk_en When HIGH, disable clock

CRU_CLKGATE_CON19

Address: Operational Base + offset (0x027c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	testclk_clk_en When HIGH, disable clock
14:9	RO	0x0	Reserved
8	RW	0x0	pclk_usb_grf_en When HIGH, disable clock
7	RW	0x0	pclk_usb3_grf_en When HIGH, disable clock
6	RW	0x0	pclk_usb3phy_pipe_en When HIGH, disable clock
5	RO	0x0	Reserved
4	RW	0x0	pclk_mipicsiphy_clk_en When HIGH, disable clock
3	RW	0x0	pclk_mipidsiphy_clk_en When HIGH, disable clock
2	RW	0x0	pclk_ddrphy_clk_en When HIGH, disable clock
1	RW	0x0	pclk_top_cru_clk_en When HIGH, disable clock
0	RW	0x0	pclk_top_niu_clk_en When HIGH, disable clock

CRU_SSGTBL0_3

Address: Operational Base + offset (0x0280)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl0_3 Extern wave table 0-3 7-0: Table0 15-8: Table1 23-16: Table2 31-24: Table3

CRU_SSGTBL4_7

Address: Operational Base + offset (0x0284)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl4_7 Extern wave table 4-7 7-0: Table4 15-8: Table5 23-16: Table6 31-24: Table7

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

CRU_SSGTBL8_11

Address: Operational Base + offset (0x0288)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl8_11 Extern wave table 8-11 7-0: Table8 15-8: Table9 23-16: Table10 31-24: Table11

CRU_SSGTBL12_15

Address: Operational Base + offset (0x028c)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl12_15 Extern wave table 12-15 7-0: Table12 15-8: Table13 23-16: Table14 31-24: Table15

CRU_SSGTBL16_19

Address: Operational Base + offset (0x0290)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl16_19 Extern wave table 16-19 7-0: Table16 15-8: Table17 23-16: Table18 31-24: Table19

CRU_SSGTBL20_23

Address: Operational Base + offset (0x0294)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl20_23 Extern wave table 20-23 7-0: Table20 15-8: Table21 23-16: Table22 31-24: Table23

CRU_SSGTBL24_27

Address: Operational Base + offset (0x0298)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl24_27 Extern wave table 24-27 7-0: Table24 15-8: Table25 23-16: Table26 31-24: Table27

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

CRU_SSGTBL28_31

Address: Operational Base + offset (0x029c)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl28_31 Extern wave table 28-31 7-0: Table28 15-8: Table29 23-16: Table30 31-24: Table31

CRU_SSGTBL32_35

Address: Operational Base + offset (0x02a0)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl32_35 Extern wave table 32-35 7-0: Table32 15-8: Table33 23-16: Table34 31-24: Table35

CRU_SSGTBL36_39

Address: Operational Base + offset (0x02a4)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl36_39 Extern wave table 36-39 7-0: Table36 15-8: Table37 23-16: Table38 31-24: Table39

CRU_SSGTBL40_43

Address: Operational Base + offset (0x02a8)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl40_43 Extern wave table 40-43 7-0: Table40 15-8: Table41 23-16: Table42 31-24: Table43

CRU_SSGTBL44_47

Address: Operational Base + offset (0x02ac)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl44_47 Extern wave table 44-47 7-0: Table44 15-8: Table45 23-16: Table46 31-24: Table47

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

CRU_SSGTBL48_51

Address: Operational Base + offset (0x02b0)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl48_51 Extern wave table 48-51 7-0: Table48 15-8: Table49 23-16: Table50 31-24: Table51

CRU_SSGTBL52_55

Address: Operational Base + offset (0x02b4)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl52_55 Extern wave table 52-55 7-0: Table52 15-8: Table53 23-16: Table54 31-24: Table55

CRU_SSGTBL56_59

Address: Operational Base + offset (0x02b8)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl56_59 Extern wave table 56-59 7-0: Table56 15-8: Table57 23-16: Table58 31-24: Table59

CRU_SSGTBL60_63

Address: Operational Base + offset (0x02bc)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl60_63 Extern wave table 60-63 7-0: Table60 15-8: Table61 23-16: Table62 31-24: Table63

CRU_SSGTBL64_67

Address: Operational Base + offset (0x02c0)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl64_67 Extern wave table 64-67 7-0: Table64 15-8: Table65 23-16: Table66 31-24: Table67

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

CRU_SSGTBL68_71

Address: Operational Base + offset (0x02c4)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl68_71 Extern wave table 68-71 7-0: Table68 15-8: Table69 23-16: Table70 31-24: Table71

CRU_SSGTBL72_75

Address: Operational Base + offset (0x02c8)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl72_75 Extern wave table 72-75 7-0: Table72 15-8: Table73 23-16: Table74 31-24: Table75

CRU_SSGTBL76_79

Address: Operational Base + offset (0x02cc)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl76_79 Extern wave table 76-79 7-0: Table76 15-8: Table77 23-16: Table78 31-24: Table79

CRU_SSGTBL80_83

Address: Operational Base + offset (0x02d0)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl80_83 Extern wave table 76-79 7-0: Table80 15-8: Table81 23-16: Table82 31-24: Table83

CRU_SSGTBL84_87

Address: Operational Base + offset (0x02d4)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl84_87 Extern wave table 84-87 7-0: Table84 15-8: Table85 23-16: Table86 31-24: Table87

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

CRU_SSGTBL88_91

Address: Operational Base + offset (0x02d8)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl88_91 Extern wave table 88-91 7-0: Table88 15-8: Table89 23-16: Table90 31-24: Table91

CRU_SSGTBL92_95

Address: Operational Base + offset (0x02dc)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl92_95 Extern wave table 92-95 7-0: Table92 15-8: Table93 23-16: Table94 31-24: Table95

CRU_SSGTBL96_99

Address: Operational Base + offset (0x02e0)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl96_99 Extern wave table 96-99 7-0: Table96 15-8: Table97 23-16: Table98 31-24: Table99

CRU_SSGTBL100_103

Address: Operational Base + offset (0x02e4)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl100_103 Extern wave table 100-103 7-0: Table100 15-8: Table101 23-16: Table102 31-24: Table103

CRU_SSGTBL104_107

Address: Operational Base + offset (0x02e8)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl104_107 Extern wave table 104-107 7-0: Table104 15-8: Table105 23-16: Table106 31-24: Table107

CRU_SSGTBL108_111

Address: Operational Base + offset (0x02ec)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl108_111 Extern wave table 108-111 7-0: Table108 15-8: Table109 23-16: Table110 31-24: Table111

CRU_SSGTBL112_115

Address: Operational Base + offset (0x02f0)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl112_115 Extern wave table 112-115 7-0: Table112 15-8: Table113 23-16: Table114 31-24: Table115

CRU_SSGTBL116_119

Address: Operational Base + offset (0x02f4)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl116_119 Extern wave table 116-119 7-0: Table116 15-8: Table117 23-16: Table118 31-24: Table119

CRU_SSGTBL120_123

Address: Operational Base + offset (0x02f8)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl120_123 Extern wave table 120-123 7-0: Table120 15-8: Table121 23-16: Table122 31-24: Table123

CRU_SSGTBL124_127

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x02fc)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ssgtbl124_127 Extern wave table 124-127 7-0: Table124 15-8: Table125 23-16: Table126 31-24: Table127

CRU_SOFTRST_CON0

Address: Operational Base + offset (0x0300)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	core_grf_psrstn_req When HIGH, reset relative logic
14	RW	0x0	core_prf_asrstn_req When HIGH, reset relative logic
13	RW	0x0	pdcore_adb400_core2gic_asrstn_req When HIGH, reset relative logic
12	RW	0x0	pdcore_adb400_gic2core_asrstn_req When HIGH, reset relative logic
11	RW	0x0	core_msniu_srstn_req When HIGH, reset relative logic
10	RW	0x0	dap_srstn_req When HIGH, reset relative logic
9	RW	0x0	l2_srstn_req When HIGH, reset relative logic
8	R/WSC	0x0	strc_sys_asrstn_req When HIGH, reset relative logic
7	R/WSC	0x0	core_noc_srstn_req When HIGH, reset relative logic
6	RW	0x0	topdbg_srstn_req When HIGH, reset relative logic
5	RW	0x0	core1_dbg_srstn_req When HIGH, reset relative logic
4	RW	0x0	core0_dbg_srstn_req When HIGH, reset relative logic
3	RW	0x0	core1_srstn_req When HIGH, reset relative logic
2	R/WSC	0x0	core0_srstn_req When HIGH, reset relative logic
1	RW	0x0	corepo1_srstn_req When HIGH, reset relative logic

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
0	R/W SC	0x0	corepo0_srstn_req When HIGH, reset relative logic

CRU_SOFTRST_CON1

Address: Operational Base + offset (0x0304)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	ddrdfi_ctl_psrstn_req When HIGH, reset relative logic
14	RW	0x0	ddrdfi_ctl_srstn_req When HIGH, reset relative logic
13	RW	0x0	axi_split_asrstn_req When HIGH, reset relative logic
12	RW	0x0	ddrgrf_psrstn_req When HIGH, reset relative logic
11	RW	0x0	ddrstdby_srstn_req When HIGH, reset relative logic
10	RW	0x0	ddrstdby_psrstn_req When HIGH, reset relative logic
9	RW	0x0	ddrmon_psrstn_req When HIGH, reset relative logic
8	RW	0x0	msch_psrstn_req When HIGH, reset relative logic
7	RW	0x0	msch_srstn_req When HIGH, reset relative logic
6	RW	0x0	upctl2_prstn_req When HIGH, reset relative logic
5	RW	0x0	upctl2_asrstn_req When HIGH, reset relative logic
4	RW	0x0	upctl2_srstn_req When HIGH, reset relative logic
3	RO	0x0	Reserved
2	RW	0x0	ddrphy_psrstn_req When HIGH, reset relative logic
1	RO	0x0	Reserved
0	RW	0x0	ddrphy_srstn_req When HIGH, reset relative logic

CRU_SOFTRST_CON2

Address: Operational Base + offset (0x0308)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	pdgic_spinlock_asrstn_req When HIGH, reset relative logic
14:12	RO	0x0	Reserved
11	RW	0x0	core_pvtm_srstn_req When HIGH, reset relative logic
10	RW	0x0	npu_pvtm_srstn_req When HIGH, reset relative logic
9	RW	0x0	npu2mem_asrstn_req When HIGH, reset relative logic
8	RW	0x0	npu_niu_hsrstn_req When HIGH, reset relative logic
7	RW	0x0	npu_niu_asrstn_req When HIGH, reset relative logic
6	RW	0x0	npu_hsrstn_req When HIGH, reset relative logic
5	RW	0x0	npu_asrstn_req When HIGH, reset relative logic
4	RW	0x0	npu_core_srstn_req When HIGH, reset relative logic
3	RW	0x0	pdgic_adb400_gic2core_asrstn_req When HIGH, reset relative logic
2	RW	0x0	pdgic_adb400_core2gic_asrstn_req When HIGH, reset relative logic
1	RW	0x0	gic500_asrstn_req When HIGH, reset relative logic
0	RW	0x0	gic500_niu_asrstn_req When HIGH, reset relative logic

CRU_SOFTRST_CON3

Address: Operational Base + offset (0x030c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	pcie_niu_psrstn_req When HIGH, reset relative logic
14	RW	0x0	pcie_niu_asrstn_req When HIGH, reset relative logic

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
13	RW	0x0	pciectl_pwr_srstn_req When HIGH, reset relative logic
12	RW	0x0	pciectl_sticky_srstn_req When HIGH, reset relative logic
11	RW	0x0	pciectl_nsticky_srstn_req When HIGH, reset relative logic
10	RW	0x0	pciectl_core_srstn_req When HIGH, reset relative logic
9	RW	0x0	pciectl_pe_srstn_req When HIGH, reset relative logic
8	RW	0x0	pciectl_button_srstn_req When HIGH, reset relative logic
7	RW	0x0	pciectl_dbi_asrstn_req When HIGH, reset relative logic
6	RW	0x0	pciectl_slv_asrstn_req When HIGH, reset relative logic
5	RW	0x0	pciectl_mst_asrstn_req When HIGH, reset relative logic
4	RW	0x1	pciectl_powerup_srstn_req When HIGH, reset relative logic
3	RW	0x0	pciectl_psrstn_req When HIGH, reset relative logic
2	RW	0x0	pciegrf_psrstn_req When HIGH, reset relative logic
1	RW	0x0	pcie_niu_lssrstn_req When HIGH, reset relative logic
0	RW	0x0	pcie_niu_hssrstn_req When HIGH, reset relative logic

CRU_SOFTRST_CON4

Address: Operational Base + offset (0x0310)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RO	0x0	Reserved
11	RW	0x0	usb2host_utmi_srstn_req When HIGH, reset relative logic
10	RW	0x0	usb2host_arb_hsrstn_req When HIGH, reset relative logic
9	RW	0x0	usb2host_hsrstn_req When HIGH, reset relative logic

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
8	RW	0x0	usb3_otg_asrstn_req When HIGH, reset relative logic
7	RW	0x0	usb2phy_grf_psrstn_req When HIGH, reset relative logic
6	RW	0x0	usb3phy_grf_psrstn_req When HIGH, reset relative logic
5	RW	0x0	usbphy_host_port_srstn_req When HIGH, reset relative logic
4	RW	0x0	usbphy_otg_port_srstn_req When HIGH, reset relative logic
3	RW	0x0	usbphy_por_srstn_req When HIGH, reset relative logic
2	RW	0x0	pciephy_pipe_srstn_req When HIGH, reset relative logic
1	RW	0x0	pciephy_psrstn_req When HIGH, reset relative logic
0	RW	0x0	pciephy_por_srstn_req When HIGH, reset relative logic

CRU_SOFTRST_CON5

Address: Operational Base + offset (0x0314)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	vpu_hsrstn_req When HIGH, reset relative logic
14	RW	0x0	vpu_asrstn_req When HIGH, reset relative logic
13	RW	0x0	vpu_niu_hsrstn_req When HIGH, reset relative logic
12	RW	0x0	vpu_niu_asrstn_req When HIGH, reset relative logic
11:9	RO	0x0	Reserved
8	RW	0x0	system_sram_niu_hssrstn_req When HIGH, reset relative logic
7	RW	0x0	system_sram_3_niu_asrstn_req When HIGH, reset relative logic
6	RW	0x0	system_sram_2_niu_asrstn_req When HIGH, reset relative logic
5	RW	0x0	system_sram_1_niu_asrstn_req When HIGH, reset relative logic
4	RW	0x0	system_sram_0_niu_asrstn_req When HIGH, reset relative logic
3	RW	0x0	system_sram_3_asrstn_req When HIGH, reset relative logic
2	RW	0x0	system_sram_2_asrstn_req When HIGH, reset relative logic
1	RW	0x0	system_sram_1_asrstn_req When HIGH, reset relative logic
0	RW	0x0	system_sram_0_asrstn_req When HIGH, reset relative logic

CRU_SOFTRST_CON6

Address: Operational Base + offset (0x0318)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RO	0x0	Reserved
13	RW	0x0	csitx_isrstn_req When HIGH, reset relative logic
12	RW	0x0	csitx_cam_srstn_req When HIGH, reset relative logic
11	RW	0x0	csitx_txesc_srstn_req When HIGH, reset relative logic
10	RW	0x0	csitx_txbytehs_srstn_req When HIGH, reset relative logic
9	RW	0x0	csitx_psrstn_req When HIGH, reset relative logic
8	RW	0x0	mipidsi_host_psrstn_req When HIGH, reset relative logic
7	RW	0x0	vopl_drstn_req When HIGH, reset relative logic
6	RW	0x0	vopl_hsrstn_req When HIGH, reset relative logic
5	RW	0x0	vopl_asrstn_req When HIGH, reset relative logic
4	RW	0x0	vopb_drstn_req When HIGH, reset relative logic
3	RW	0x0	vopb_hsrstn_req When HIGH, reset relative logic
2	RW	0x0	vopb_asrstn_req When HIGH, reset relative logic
1	RW	0x0	vio_niu_lssrstn_req When HIGH, reset relative logic
0	RW	0x0	vio_niu_hssrstn_req When HIGH, reset relative logic

CRU_SOFTRST_CON7

Address: Operational Base + offset (0x031c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15:14	RO	0x0	Reserved
13	RW	0x0	mipidsiphy_psrstn_req When HIGH, reset relative logic
12	RW	0x0	mipicsiphy_psrstn_req When HIGH, reset relative logic
11	RO	0x0	Reserved
10	RW	0x0	isp_srstn_req When HIGH, reset relative logic
9	RW	0x0	isp_hsrstn_req When HIGH, reset relative logic
8	RW	0x0	cif_dsrstn_req When HIGH, reset relative logic
7	RW	0x0	cif_pclkin_srstn_req When HIGH, reset relative logic
6	RW	0x0	cif_isrstn_req When HIGH, reset relative logic
5	RW	0x0	cif_hsrstn_req When HIGH, reset relative logic
4	RW	0x0	cif_asrstn_req When HIGH, reset relative logic
3	RW	0x0	csirx_psrstn_req When HIGH, reset relative logic
2	RW	0x0	rga_srstn_req When HIGH, reset relative logic
1	RW	0x0	rga_hsrstn_req When HIGH, reset relative logic
0	RW	0x0	rga_asrstn_req When HIGH, reset relative logic

CRU_SOFTRST_CON8

Address: Operational Base + offset (0x0320)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	gmac_asrstn_req When HIGH, reset relative logic
14	RW	0x0	pd_sd_gmac_niu_psrstn_req When HIGH, reset relative logic
13	RW	0x0	pd_sd_gmac_niu_hsrstn_req When HIGH, reset relative logic
12	RW	0x0	pd_sd_gmac_niu_asrstn_req When HIGH, reset relative logic

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
11:10	RO	0x0	Reserved
9	RW	0x0	sfc_srstn_req When HIGH, reset relative logic
8	RW	0x0	sfc_hsrstn_req When HIGH, reset relative logic
7	RW	0x0	emmc_hsrstn_req When HIGH, reset relative logic
6	RW	0x0	sdio_hsrstn_req When HIGH, reset relative logic
5	RW	0x0	sdmmc_hsrstn_req When HIGH, reset relative logic
4	RW	0x0	pd_mmc_sfc_niu_hsrstn_req When HIGH, reset relative logic
3:2	RO	0x0	Reserved
1	RW	0x0	pdperi_niu_lssrstn_req When HIGH, reset relative logic
0	RW	0x0	pdperi_niu_mssrstn_req When HIGH, reset relative logic

CRU_SOFTRST_CON9

Address: Operational Base + offset (0x0324)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RO	0x0	Reserved
14	RW	0x0	pmu_gpio0_dbsrstn_req When HIGH, reset relative logic
13	RW	0x0	pmu_i2c0_srstn_req When HIGH, reset relative logic
12	RW	0x0	pmu_i2c0_psrstn_req When HIGH, reset relative logic
11	RW	0x0	pmu_ddr_fail_save_srstn_req When HIGH, reset relative logic
10	RO	0x0	Reserved
9	RW	0x0	pmu_uart0_srstn_req When HIGH, reset relative logic
8	RW	0x0	pmu_pvtm_srstn_req When HIGH, reset relative logic
7	RO	0x0	Reserved
6	RW	0x0	pmu_uart0_psrstn_req When HIGH, reset relative logic

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5	RW	0x0	pmu_gpio0_psrstn_req When HIGH, reset relative logic
4	RW	0x0	pmu_sram_psrstn_req When HIGH, reset relative logic
3	RW	0x0	pmu_pmu_srstn_req When HIGH, reset relative logic
2	RW	0x0	pmu_grf_psrstn_req When HIGH, reset relative logic
1	RO	0x0	Reserved
0	RW	0x0	pmu_niu_psrstn_req When HIGH, reset relative logic

CRU_SOFTRST_CON10

Address: Operational Base + offset (0x0328)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:9	RO	0x0	Reserved
8	RW	0x0	i2s0_rx_srstn_req When HIGH, reset relative logic
7	RW	0x0	i2s1_srstn_req When HIGH, reset relative logic
6	RW	0x0	i2s1_hsrstn_req When HIGH, reset relative logic
5	RW	0x0	i2s0_tx_srstn_req When HIGH, reset relative logic
4	RW	0x0	i2s0_hsrstn_req When HIGH, reset relative logic
3	RW	0x0	pdm_srstn_req When HIGH, reset relative logic
2	RW	0x0	pdm_hsrstn_req When HIGH, reset relative logic
1	RW	0x0	vad_hsrstn_req When HIGH, reset relative logic
0	RW	0x0	pd_audio_niu_hsrstn_req When HIGH, reset relative logic

CRU_SOFTRST_CON11

Address: Operational Base + offset (0x032c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	uart3_srstn_req When HIGH, reset relative logic
14	RW	0x0	uart3_psrstn_req When HIGH, reset relative logic
13	RW	0x0	uart2_srstn_req When HIGH, reset relative logic
12	RW	0x0	uart2_psrstn_req When HIGH, reset relative logic
11	RW	0x0	uart1_srstn_req When HIGH, reset relative logic
10	RW	0x0	uart1_psrstn_req When HIGH, reset relative logic
9	RW	0x0	dcf_psrstn_req When HIGH, reset relative logic
8	RW	0x0	dcf_asrstn_req When HIGH, reset relative logic
7	RW	0x0	crypto_pka_srstn_req When HIGH, reset relative logic
6	RW	0x0	crypto_core_srstn_req When HIGH, reset relative logic
5	RW	0x0	crypto_hsrstn_req When HIGH, reset relative logic
4	RW	0x0	crypto_asrstn_req When HIGH, reset relative logic
3	RW	0x0	rom_hsrstn_req When HIGH, reset relative logic
2	RW	0x0	top_niu_psrstn_req When HIGH, reset relative logic
1	RW	0x0	bus_niu_lssrstn_req When HIGH, reset relative logic
0	RW	0x0	bus_niu_mssrstn_req When HIGH, reset relative logic

CRU_SOFTRST_CON12

Address: Operational Base + offset (0x0330)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15	RW	0x0	pwm0_srstn_req When HIGH, reset relative logic
14	RW	0x0	pwm0_psrstn_req When HIGH, reset relative logic
13	RW	0x0	i2c3_srstn_req When HIGH, reset relative logic
12	RW	0x0	i2c3_psrstn_req When HIGH, reset relative logic
11	RW	0x0	i2c2_srstn_req When HIGH, reset relative logic
10	RW	0x0	i2c2_psrstn_req When HIGH, reset relative logic
9	RW	0x0	i2c1_srstn_req When HIGH, reset relative logic
8	RW	0x0	i2c1_psrstn_req When HIGH, reset relative logic
7	RW	0x0	uart7_srstn_req When HIGH, reset relative logic
6	RW	0x0	uart7_psrstn_req When HIGH, reset relative logic
5	RW	0x0	uart6_srstn_req When HIGH, reset relative logic
4	RW	0x0	uart6_psrstn_req When HIGH, reset relative logic
3	RW	0x0	uart5_srstn_req When HIGH, reset relative logic
2	RW	0x0	uart5_psrstn_req When HIGH, reset relative logic
1	RW	0x0	uart4_srstn_req When HIGH, reset relative logic
0	RW	0x0	uart4_psrstn_req When HIGH, reset relative logic

CRU_SOFTRST_CON13

Address: Operational Base + offset (0x0334)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	timer2_srstn_req When HIGH, reset relative logic
14	RW	0x0	timer1_srstn_req When HIGH, reset relative logic

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
13	RW	0x0	timer0_srstn_req When HIGH, reset relative logic
12	RW	0x0	timer_psrstn_req When HIGH, reset relative logic
11	RW	0x0	bus_grf_psrstn_req When HIGH, reset relative logic
10	RO	0x0	Reserved
9	RW	0x0	spi2_srstn_req When HIGH, reset relative logic
8	RW	0x0	spi2_psrstn_req When HIGH, reset relative logic
7	RW	0x0	spi1_srstn_req When HIGH, reset relative logic
6	RW	0x0	spi1_psrstn_req When HIGH, reset relative logic
5	RW	0x0	spi0_srstn_req When HIGH, reset relative logic
4	RW	0x0	spi0_psrstn_req When HIGH, reset relative logic
3	RW	0x0	pwm2_srstn_req When HIGH, reset relative logic
2	RW	0x0	pwm2_psrstn_req When HIGH, reset relative logic
1	RW	0x0	pwm1_srstn_req When HIGH, reset relative logic
0	RW	0x0	pwm1_psrstn_req When HIGH, reset relative logic

CRU_SOFTRST_CON14

Address: Operational Base + offset (0x0338)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RO	0x0	Reserved
14	RW	0x0	bus_sub_niu_mssrstn_req When HIGH, reset relative logic
13	RW	0x0	gpio4_dbsrstn_req When HIGH, reset relative logic
12	RW	0x0	gpio4_psrstn_req When HIGH, reset relative logic
11	RW	0x0	gpio3_dbsrstn_req When HIGH, reset relative logic

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
10	RW	0x0	gpio3_psrstn_req When HIGH, reset relative logic
9	RW	0x0	gpio2_dbsrstn_req When HIGH, reset relative logic
8	RW	0x0	gpio2_psrstn_req When HIGH, reset relative logic
7	RW	0x0	gpio1_dbsrstn_req When HIGH, reset relative logic
6	RW	0x0	gpio1_psrstn_req When HIGH, reset relative logic
5	RW	0x0	efuse_ns_srstn_req When HIGH, reset relative logic
4	RW	0x0	efuse_ns_psrstn_req When HIGH, reset relative logic
3	RW	0x0	wdt_ns_psrstn_req When HIGH, reset relative logic
2	RW	0x0	timer5_srstn_req When HIGH, reset relative logic
1	RW	0x0	timer4_srstn_req When HIGH, reset relative logic
0	RW	0x0	timer3_srstn_req When HIGH, reset relative logic

CRU_SOFTRST_CON15

Address: Operational Base + offset (0x033c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	tsadc_srstn_req When HIGH, reset relative logic
14	RW	0x0	tsadc_psrstn_req When HIGH, reset relative logic
13	RW	0x0	saradc_psrstn_req When HIGH, reset relative logic
12:4	RO	0x0	Reserved
3	RW	0x0	i2c5_srstn_req When HIGH, reset relative logic
2	RW	0x0	i2c5_psrstn_req When HIGH, reset relative logic
1	RW	0x0	i2c4_srstn_req When HIGH, reset relative logic

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
0	RW	0x0	i2c4_psrstn_req When HIGH, reset relative logic

CRU_SDMMC_CON0

Address: Operational Base + offset (0x0380)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RO	0x0	Reserved
11	RW	0x0	drv_sel drv_sel
10:3	RW	0x00	drv_delaynum drv_delaynum
2:1	RW	0x2	drv_degree drv_degree
0	RW	0x0	init_state init_state

CRU_SDMMC_CON1

Address: Operational Base + offset (0x0384)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RO	0x0	Reserved
11	RW	0x0	sample_sel sample_sel
10:3	RW	0x00	sample_delaynum sample_delaynum
2:1	RW	0x0	sample_degree sample_degree
0	RO	0x0	Reserved

CRU_SDIO_CON0

Address: Operational Base + offset (0x0388)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
11	RW	0x0	drv_sel drv_sel
10:3	RW	0x00	drv_delaynum drv_delaynum
2:1	RW	0x2	drv_degree drv_degree
0	RW	0x0	init_state init_state

CRU_SDIO_CON1

Address: Operational Base + offset (0x038c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RO	0x0	Reserved
11	RW	0x0	sample_sel sample_sel
10:3	RW	0x00	sample_delaynum sample_delaynum
2:1	RW	0x0	sample_degree sample_degree
0	RO	0x0	Reserved

CRU_EMMC_CON0

Address: Operational Base + offset (0x0390)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RO	0x0	Reserved
11	RW	0x0	drv_sel drv_sel
10:3	RW	0x00	drv_delaynum drv_delaynum
2:1	RW	0x2	drv_degree drv_degree
0	RW	0x0	init_state init_state

CRU_EMMC_CON1

Address: Operational Base + offset (0x0394)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RO	0x0	Reserved
11	RW	0x0	sample_sel sample_sel
10:3	RW	0x00	sample_delaynum sample_delaynum
2:1	RW	0x0	sample_degree sample_degree
0	RO	0x0	Reserved

CRU_PMUPLL_CON0

Address: Operational Base + offset (0x4000)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	bypass PLL Bypass. FREF bypasses PLL to FOUTPOSTDIV 1'b0: No bypass 1'b1: Bypass
14:12	RW	0x2	postdiv1 First Post Divide Value, (1-7)
11:0	RW	0x0fa	fbdiv Feedback Divide Value, valid divider settings are: [16, 2500] in integer mode [20, 500] in fractional mode Tips: No plus one operation

CRU_PMUPLL_CON1

Address: Operational Base + offset (0x4004)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	pllpdsel PLL global power down source selection If pllpdsel == 1, PLL can be power down only by pllpd1, otherwise pll is power down when any one of refdiv/fbdiv/fracdiv is changed or pllpd0 is asserted

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
14	RW	0x0	pllpd1 PLL global power down request 1'b0: No power down 1'b1: Power down
13	RW	0x0	pllpd0 PLL global power down request 1'b0: No power down 1'b1: Power down
12	RW	0x1	dsmpd PLL delta sigma modulator enable 1'b0: Modulator is enable 1'b1: Modulator is disabled
11	RO	0x0	Reserved
10	RO	0x0	pll_lock PLL lock status 1'b0: Unlock 1'b1: Lock
9	RO	0x0	Reserved
8:6	RW	0x1	postdiv2 Second Post Divide Value, (1-7)
5:0	RW	0x03	refdiv Reference Clock Divide Value, (1-63)

CRU_PMUPLL_CON2

Address: Operational Base + offset (0x4008)

Bit	Attr	Reset Value	Description
31:28	RO	0x0	Reserved
27	RW	0x0	fout4phasepd Power down 4-phase clocks and 2X, 3X, 4X clocks 1'b0: No power down 1'b1: Power down
26	RW	0x0	foutvcopd Power down buffered VCO clock 1'b0: No power down 1'b1: Power down
25	RW	0x0	foutpostdivpd Power down all outputs except for buffered VCO clock 1'b0: No power down 1'b1: Power down
24	RW	0x0	dacpd Power down quantization noise cancellation DAC 1'b0: No power down 1'b1: Power down

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
23:0	RW	0x000001	fracdiv Fractional part of feedback divide (fraction = FRAC/2^24)

CRU_PMUPLL_CON3

Address: Operational Base + offset (0x400c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:13	RO	0x0	Reserved
12:8	WO	0x00	ssmod_spread spread amplitude % = 0.1 * SPREAD[4: 0]
7:4	WO	0x0	ssmod_divval Divider required to set the modulation frequency
3	WO	0x0	ssmod_downspread Selects center spread or downs pread 1'b0: Down spread 1'b1: Center spread
2	WO	0x1	ssmod_reset Reset modulator state 1'b0: No reset 1'b1: Reset
1	WO	0x1	ssmod_disable_sscg Bypass SSMOD by module 1'b0: No bypass 1'b1: Bypass
0	WO	0x1	ssmod_bp Bypass SSMOD by integration 1'b0: No bypass 1'b1: Bypass

CRU_PMUPLL_CON4

Address: Operational Base + offset (0x4010)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	WO	0x7f	ssmod_ext_maxaddr External wave table data inputs, (0-255)
7:1	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
0	WO	0x0	ssmod_sel_ext_wave 1'b0: No select ext_wave 1'b1: Select ext_wave

CRU_PMU_MODE

Address: Operational Base + offset (0x4020)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:2	RO	0x0	Reserved
1:0	RW	0x0	pmupll_work_mode 2'b00: Clock from xin_osc0_func_div 2'b01: Clock from pll 2'b10: Clock from clk_rtc_32k 2'b11: Reserved

CRU_PMU_CLKSEL_CON0

Address: Operational Base + offset (0x4040)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	clk_rtc32k_clk_sel 2'b00: Select clk_32k_from_io as clk_rtc_32k 2'b01: Select clk_32k_from_pvtm as clk_rtc_32k 2'b10: Select clk_div32p768khz as clk_rtc_32k 2'b11: Reserved
13	RO	0x0	Reserved
12:8	RW	0x00	xin_osc0_func_div_con xin_osc0_func_div=xin_osc0/(div_con+1)
7:5	RO	0x0	Reserved
4:0	RW	0x09	pclk_pdpmu_div_con pclk_pdpmu=pmupll_clk_src/(div_con+1)

CRU_PMU_CLKSEL_CON1

Address: Operational Base + offset (0x4044)

Bit	Attr	Reset Value	Description
31:0	RW	0x0bb8ea60	clk_div32p768khz_div_con High 16-bit for numerator, Low 16-bit for denominator, clock source is xin_osc0

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

CRU_PMU_CLKSEL_CON2

Address: Operational Base + offset (0x4048)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	clk_wifi_sel 1'b0: Select xin_osc0 as clk_wifi_out 1'b1: Select clk_wifi_div as clk_wifi_out
14	RO	0x0	Reserved
13:8	RW	0x31	clk_wifi_div_con clk_wifi_div=pmupll_clk_src/(div_con+1)
7	RW	0x0	mipidsiphy_ref_sel 1'b0: Select xin_osc0 as mipidsi phy reference clock 1'b1: Select clk_ref24m as mipidsi phy reference clock
6	RW	0x0	usbphy_ref_sel 1'b0: Select xin_osc0 as usbphy reference clock 1'b1: Select clk_ref24m as usbphy reference clock
5:0	RW	0x31	clk_ref24m_div_con clk_ref24m=pmupll_clk_src/(div_con+1)

CRU_PMU_CLKSEL_CON3

Address: Operational Base + offset (0x404c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	clk_uart0_pll_sel 2'b00: GPLL 2'b01: USBPHY480M 2'b10: CPLL 2'b11: PMUPLL
13:7	RO	0x0	Reserved
6:0	RW	0x09	clk_uart0_div_con clk_uart0=pll_clk_src/(div_con+1)

CRU_PMU_CLKSEL_CON4

Address: Operational Base + offset (0x4050)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15:14	RW	0x3	clk_uart0_sel 2'b00: Select clk_uart0 2'b01: Select clk_uart0_np5 2'b10: Select clk_uart0_frac_out 2'b11: Select xin_osc0
13:7	RO	0x0	Reserved
6:0	RW	0x09	clk_uart0_divnp5_div_con clk_uart0_np5=2*clk_uart0/(2*div_con+3)

CRU_PMU_CLKSEL_CON5

Address: Operational Base + offset (0x4054)

Bit	Attr	Reset Value	Description
31:0	RW	0x0bb8ea60	clk_uart0_frac_div_con High 16-bit for numerator, Low 16-bit for denominator, clock source is clk_uart0

CRU_PMU_CLKSEL_CON6

Address: Operational Base + offset (0x4058)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	dbclk_gpio0_pll_sel 1'b0: Select xin_osc0 1'b1: Select clk_rtc_32k
14:11	RO	0x0	Reserved
10:0	RW	0x001	dbclk_gpio0_div_con dbclk_gpio0=pll_clk_src/(div_con+1)

CRU_PMU_CLKSEL_CON7

Address: Operational Base + offset (0x405c)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	clk_i2c0_pll_sel 1'b0: Select PMUPLL 1'b1: Select xin_osc0
14:8	RW	0x04	clk_i2c0_div_con clk_i2c0=pll_clk_src/(div_con+1)
7:5	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4	RW	0x0	clk_pciephy_ref_clk_sel 1'b0: Select xin_osc0 1'b1: Select clk_pciephy_ref_src
3:2	RO	0x0	Reserved
1:0	RW	0x3	clk_pciephy_ref_div_con clk_pciephy_ref_src= clk_ppll_ph0 /(div_con+1) clk_ppll_ph0 is half the frequency of foutpostdiv of PMUPLL.

CRU_PMU_CLKGATE_CON0

Address: Operational Base + offset (0x4080)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	clk_wifi_clk_en When HIGH, disable clock
14	RW	0x0	clk_wifi_pll_clk_en When HIGH, disable clock
13	RW	0x0	clk_div32p768khz_src_clk_en When HIGH, disable clock
12	RW	0x0	xin_osc0_func_div_src_clk_en When HIGH, disable clock
11:10	RO	0x0	Reserved
9	RW	0x0	pclk_pmu_i2c0_clk_en When HIGH, disable clock
8	RW	0x0	pclk_pmu_cru_clk_en When HIGH, disable clock
7	RW	0x0	pclk_pmu_uart0_clk_en When HIGH, disable clock
6	RW	0x0	pclk_pmu_gpio0_clk_en When HIGH, disable clock
5	RW	0x0	pclk_pmu_sram_clk_en When HIGH, disable clock
4	RW	0x0	pclk_pmu_pmu_clk_en When HIGH, disable clock
3	RW	0x0	pclk_pmu_grf_clk_en When HIGH, disable clock
2	RW	0x0	pclk_pmu_sgrf_clk_en When HIGH, disable clock
1	RW	0x0	pclk_pmu_niu_clk_en When HIGH, disable clock
0	RW	0x0	pclk_pdpmu_pll_clk_en When HIGH, disable clock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

CRU_PMU_CLKGATE_CON1

Address: Operational Base + offset (0x4084)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:13	RO	0x0	Reserved
12	RW	0x0	clk_pciephy_ref_clk_en When HIGH, disable clock
11	RW	0x0	clk_pciephy_ref_div_clk_en When HIGH, disable clock
10	RW	0x0	mipidsiphy_ref_cclk_en When HIGH, disable clock
9	RW	0x0	usbphy_ref_clk_en When HIGH, disable clock
8	RW	0x0	clk_ref24m_pll_clk_en When HIGH, disable clock
7	RO	0x0	Reserved
6	RW	0x0	dbclk_gpio0_pmu_pll_clk_en When HIGH, disable clock
5	RW	0x0	clk_i2c0_pmu_pll_clk_en When HIGH, disable clock
4	RW	0x0	clk_pvtm_pmu_clk_en When HIGH, disable clock
3	RW	0x0	clk_uart0_pmu_clk_en When HIGH, disable clock
2	RW	0x0	clk_uart0_pmu_frac_clk_en When HIGH, disable clock
1	RW	0x0	clk_uart0_pmu_divnp5_clk_en When HIGH, disable clock
0	RW	0x0	clk_uart0_pmu_pll_clk_en When HIGH, disable clock

2.7 Application Notes

2.7.1 PLL usage

A. PLL output frequency configuration

FBDIV, POSTDIV1, BYPASS can be configured by programming CRU_xPLL_CON0.
DSMPD, REFDIV, POSTDIV2 can be configured by programming CRU_xPLL_CON1.
FRAC can be configured by programming CRU_xPLL_CON2.

If DSMPD = 1 (DSM is disabled, "integer mode")
FOUTVCO = (FREF / REFDIV) * FBDIV
FOUTPOSTDIV = FOUTVCO / (POSTDIV1*POSTDIV2)
When FREF is 24MHz, and if 700MHz FOUTPOSTDIV is needed. The configuration can be:

DSMPD = 1
REFDIV = 6
FBDIV = 175

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

POSTDIV1=1
POSTDIV2=1
And then
FOUTVCO = (FREF / REFDIV) * FBDIV = 24/6*175=700
FOUTPOSTDIV = FOUTVCO / (POSTDIV1*POSTDIV2)=700/1/1=700

If DSMPD = 0 (DSM is enabled, "fractional mode")
FOUTVCO = (FREF / REFDIV) * (FBDIV + FRAC / (2^24))
FOUTPOSTDIV = FOUTVCO / (POSTDIV1*POSTDIV2)
When FREF is 24MHz, and if 491.52MHz FOUTPOSTDIV is needed. The configuration can be:
DSMPD = 0
REFDIV = 1
FBDIV = 40
FRAC = 24'hf5c28f
POSTDIV1=2
POSTDIV2=1
And then
FOUTVCO = (FREF / REFDIV) * (FBDIV + FRAC / (2^24)) = 983.04
FOUTPOSTDIV = FOUTVCO / (POSTDIV1*POSTDIV2)=983.04/(2*1)=491.52

B. PLL setting consideration

- If the POSTDIV value is changed during operation a short pulse (glitch) may occur on FOUTPOSTDIV. The minimum width of the short pulse will be equal to twice the period of the VCO. Therefore, if the circuitry clocked by the PLL is sensitive to short pulses, the new divide value should be re-timed so that it is synchronous with the rising edge of the output clock (FOUTPOSTDIV). Glitches cannot occur on any of the other outputs.
- For lowest power operation, the minimum VCO and FREF frequencies should be used. For minimum jitter operation, the highest VCO and FREF frequencies should be used. The normal operating range for the VCO is described above in.
- The supply rejection will be worse at the low end of the VCO range so care should be taken to keep the supply clean for low power applications.
- The feedback divider is not capable of dividing by all possible settings due to the use of a power-saving architecture. The following settings are valid for FBDIV:
- The PD input places the PLL into the lowest power mode. In this case, all analog circuits are turned off and FREF will be "ignored". The FOUTPOSTDIV and FOUTVCO pins are forced to logic low (0V).
- The BYPASS pin controls FREF to be passed to the FOUTPOSTDIV when active high. However, the PLL continues to run as it normally would if bypass were low. This is a useful feature for PLL testing since the clock path can be verified without the PLL being required to work. Also, the effect that the PLL induced supply noise has on the output buffering can be evaluated. It is not recommended to switch between BYPASS mode and normal mode for regular chip operation since this may result in a glitch. Also, FOUTPOSTDIVPD should be set low if the PLL is to be used in BYPASS mode.

2.7.2 PLL frequency change and lock check

The PLL programming supports changed on-the-fly and the PLL will simply slew to the new frequency.
PLL lock state can be checked in CRU_APLL_CON1[10], CRU_DPLL_CON1[10], CRU_CPLL_CON1[10], CRU_GPLL_CON1[10], CRU_NPLL_CON1[10], CRU_PMUPLL_CON1[10] register. The lock state is high when both original hardware PLL lock and PLL counter lock are high. The PLL counter lock initial value is CRU_GLB_CNT_TH[31:16].
The max delay time is 1000*REFDIV/FREF.
PLL locking consists of three phases.

- Phase 1 is control voltage slewing. During this phase one of the clocks (reference or divide) is much faster than the other, and the PLL frequency adjusts almost

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

continuously. When locking from power down, the divide clock is initially very slow and steadily increases frequency. It will take slightly longer for faster VCO settings when locking from power down, since the PLL must slew further.

- Phase 2 is small signal phase acquisition. During this phase, the internal up/down signals alternate semi-chaotically as the phase slowly adjusts until the two signals are aligned. The duration of this phase depends on the loop bandwidth and is faster with higher bandwidth. Bandwidth can be estimated as FREF / REFDIV / 20 for integer mode and FREF /REFDIV / 40 for fractional mode. The duration of small signal locking is about 1/Bandwidth.
- Phase 3 is the digital cycle count. After the last cycle slip is detected, an internal counter waits 256 FREF / REFDIV cycles before the lock signal goes high. This is frequently the dominant factor in lock time – especially for slower reference clock signals or large reference divide settings. This time can be calculated as 256*REFDIV/FREF.

2.7.3 Fractional divider usage

To get specific frequency, clocks of I2S, PDM, UART can be generated by fractional divider. Generally you must set that denominator 20 times larger than numerator to generate precise clock frequency. So the fractional divider applies only to generate low frequency clock like I2S, UART and PDM. For implementation issue, the input source clocks of fractional divider also have the following limitation.

Table 2-1 Source Clock Limitation of Fractional Divider

Clock Name	Fractional divider source clock Limit
clk_pdm	330MHz
clk_i2s0_tx/rx	600MHz/594MHz
clk_i2s1	600MHz
clk_uart0~7	1000MHz
vopraw_dclk	327MHz
voplite_dclk	400MHz

2.7.4 Divfree50 divider usage

Some modules like EMMC, SDIO and SDMMC need clock of 50% duty cycle, divfree50 can generate clock of 50% duty cycle even in odd value divisor.

2.7.5 DivFreeNP5 divider usage

Some modules like NPU and UART need some special frequency can use this divider. Frequency of this divider=clk_src/((2*n+1)/2). Eg, UART with baud rate of 4Mbps need use this divider to generate 64MHz clock from 480MHz of usbphypll.

2.7.6 Global software reset

Two global software resets are designed in RK1808, you can program CRU_GLB_SRST_FST_VALUE[15:0] as 0xfdb9 to assert the first global software reset glb_srstn_1 and program CRU_GLB_SRST_SND_VALUE[15:0] as 0xeca8 to assert the second global software reset glb_srstn_2. These two software resets are self-de-asserted by hardware. Resetting hold timing of global software reset (glb_srstn_1, glb_srstn_2, soc_wdt_rstn, soc_tsadc_rstn) can be programmable up to 1ms.
Glb_srstn_1 resets almost all logic.
Glb_srstn_2 resets almost all logic except GRF and GPIOs.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 3 System Debug

3.1 Overview

The chip uses the DAPLITE Technology to support real-time debug.

3.1.1 Features

- Invasive debug with core halted
- SW-DP

3.1.2 Debug components address map

The following table shows the debug components address in memory map:

Module	Base Address
DAP_ROM	0xff000000

3.2 Block Diagram

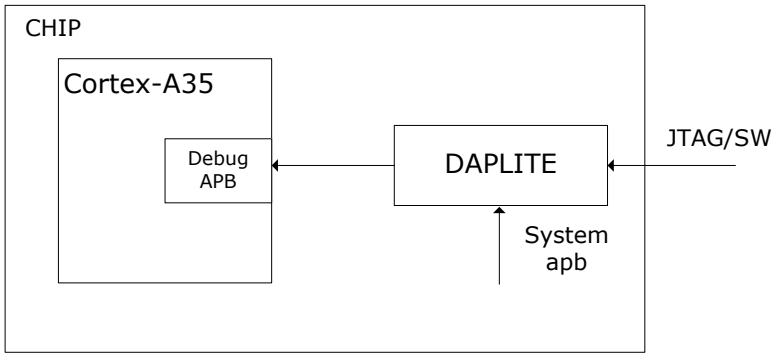


Fig. 3-1 Debug system structure

3.3 Function Description

3.3.1 DAP

The DAP has following components:

- Serial Wire JTAG Debug Port(SWJ-DP)
- APB Access Port(APB-AP)
- ROM table

The debug port is the host tools interface to access the DAP-Lite. This interface controls any access ports provided within the DAP-Lite. The DAP-Lite supports a combined debug port which includes both JTAG and Serial Wire Debug(SWD), with a mechanism that supports switching between them.

The APB-AP acts as a bridge between SWJ-DP and APB bus which translate the Debug request to APB bus.

The DAP provides an internal ROM table connected to the master Debug APB port of the APB-Mux. The Debug ROM table is loaded at address 0x00000000 and 0x80000000 of this bus and is accessible from both APB-AP and the system APB input. Bit[31] of the address bus is not connected to the ROM Table, ensuring that both views read the same value. The ROM table stores the locations of the components on the Debug APB.

More information please refer to the document CoreSight_DAPLite_TRM.pdf for the debug detail description.

3.4 Register Description

Please refer to the documentCoreSight_DAPLite_TRM.pdf for the debug detail description.

3.5 Interface Description

3.5.1 DAP SWJ-DP Interface

The following figure is the DAP SWJ-DP interface, the SWJ-DP is a combined JTAG-DP and SW-DP that enable you connect either a Serial Wire Debug(SWJ) to JTAG probe to a target.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

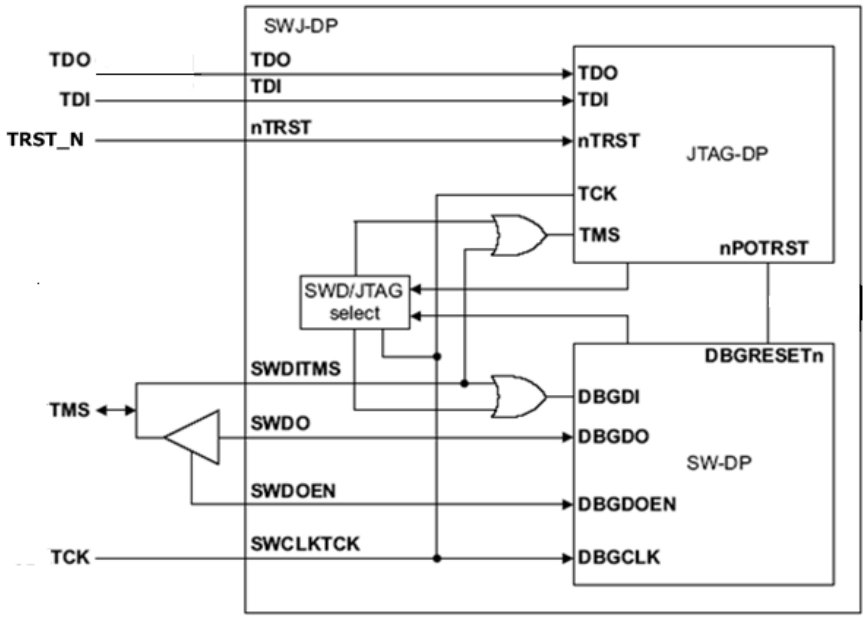


Fig. 3-2 DAP SWJ interface

3.5.2 DAP SW-DP Interface

This implementation is taken from ADIv5.1 and operates with a synchronous serial interface. This uses a single bidirectional data signal, and a clock signal.

The figure below describes the interaction between the timing of transactions on the serial wire interface, and the DAP internal bus transfers. It shows when the target responds with a WAIT acknowledgement.

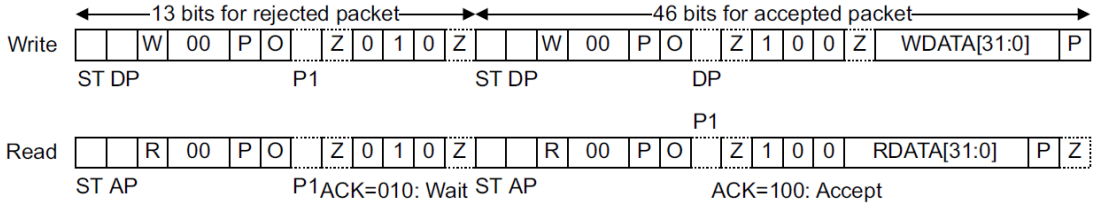


Fig. 3-3 SW-DP acknowledgement timing

Table 3-1 SW-DP Interface Description

Module pin	Direction	Pad name	IOMUX
jtag_tck	I	IO_SDMMC0d2_JTAGtck_GPIO4A4vccio6	GRF_GPIO4A_IOMUX_SEL_H[2:0]=3'b2
jtag_tms	I/O	IO_SDMMC0d3_JTAGtms_GPIO4A5vcio6	GRF_GPIO4A_IOMUX_SEL_H[6:4]=3'b2

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 4 General Register File (GRF)

4.1 Overview

The general register file is used to do system control register configuration and status register reading by software, which is located at several addresses.

- BUS_GRF, used for general non-secure system
- USB2_PHY_GRF, used for USB2 PHY configuration
- PCIe_USB3_PHY_GRF, used for PCIe/USB3 combo PHY configuration
- PMU_GRF, used for always on system
- DDR_GRF, used for configuration in PD_DDR
- PCIe_USB_GRF, use for configuration of USB2 Host, USB3 OTG and PCIe Controller

CORE_GRF, used for core PVTM and performance monitor

4.2 Function Description

The function of general register file is：

- IOMUX control
- Control the state of GPIO in power-down mode
- GPIO PAD pull down and pull up control
- Used for common system control
- Used to record the system state

Table 4-1 GRF Address Mapping Table

Name	Address Base
BUS_GRF	0xFE000000
USB2PHY_GRF	0xFE010000
PCIe_USB3_PHY_GRF	0xFE018000
PMU_GRF	0xFE020000
DDR_GRF	0xFE030000
PCIe_USB_GRF	0xFE040000
CORE_GRF	0xFE050000

4.3 BUS_GRF Register Description

4.3.1 Registers Summary

Name	Offset	Size	Reset Value	Description
BUS_GRF_GPIO1A_IOMUX_L	0x0000	W	0x00000000	GPIO1A IOMUX control low bits
BUS_GRF_GPIO1A_IOMUX_H	0x0004	W	0x00000000	GPIO1A IOMUX control high bits
BUS_GRF_GPIO1B_IOMUX_L	0x0008	W	0x00000000	GPIO1B IOMUX control low bits
BUS_GRF_GPIO1B_IOMUX_H	0x000c	W	0x00000000	GPIO1B IOMUX control high bits
BUS_GRF_GPIO2A_IOMUX_L	0x0020	W	0x00000000	GPIO2A IOMUX control low bits
BUS_GRF_GPIO2A_IOMUX_H	0x0024	W	0x00000000	GPIO2A IOMUX control high bits
BUS_GRF_GPIO2B_IOMUX_L	0x0028	W	0x00000000	GPIO2B IOMUX control low bits
BUS_GRF_GPIO2B_IOMUX_H	0x002c	W	0x00000000	GPIO2B IOMUX control high bits
BUS_GRF_GPIO2C_IOMUX_L	0x0030	W	0x00000000	GPIO2C IOMUX control low bits
BUS_GRF_GPIO2C_IOMUX_H	0x0034	W	0x00000000	GPIO2C IOMUX control high bits
BUS_GRF_GPIO2D_IOMUX_L	0x0038	W	0x00000000	GPIO2D IOMUX control low bits
BUS_GRF_GPIO3A_IOMUX_L	0x0040	W	0x00000000	GPIO3A IOMUX control low bits
BUS_GRF_GPIO3A_IOMUX_H	0x0044	W	0x00000000	GPIO3A IOMUX control high bits

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
BUS_GRF_GPIO3B_IOMUX_L	0x0048	W	0x00000000	GPIO3B IOMUX control low bits
BUS_GRF_GPIO3B_IOMUX_H	0x004c	W	0x00000000	GPIO3B IOMUX control high bits
BUS_GRF_GPIO3C_IOMUX_L	0x0050	W	0x00000000	GPIO3C IOMUX control low bits
BUS_GRF_GPIO3C_IOMUX_H	0x0054	W	0x00000000	GPIO3C IOMUX control high bits
BUS_GRF_GPIO3D_IOMUX_L	0x0058	W	0x00000000	GPIO3D IOMUX control low bits
BUS_GRF_GPIO4A_IOMUX_L	0x0060	W	0x00002200	GPIO4A IOMUX control low bits
BUS_GRF_GPIO4A_IOMUX_H	0x0064	W	0x00000022	GPIO4A IOMUX control high bits
BUS_GRF_GPIO4B_IOMUX_L	0x0068	W	0x00000000	GPIO4B IOMUX control low bits
BUS_GRF_GPIO4B_IOMUX_H	0x006c	W	0x00000000	GPIO4B IOMUX control high bits
BUS_GRF_GPIO4C_IOMUX_L	0x0070	W	0x00000000	GPIO4C IOMUX control low bits
BUS_GRF_GPIO4C_IOMUX_H	0x0074	W	0x00000000	GPIO4C IOMUX control high bits
BUS_GRF_GPIO1A_P	0x0080	W	0x00005555	GPIO1A PU/PD control
BUS_GRF_GPIO1B_P	0x0084	W	0x00009595	GPIO1B PU/PD control
BUS_GRF_GPIO2A_P	0x0090	W	0x0000aaaa	GPIO2A PU/PD control
BUS_GRF_GPIO2B_P	0x0094	W	0x0000aaaa	GPIO2B PU/PD control
BUS_GRF_GPIO2C_P	0x0098	W	0x0000aaaa	GPIO2C PU/PD control
BUS_GRF_GPIO2D_P	0x009c	W	0x00000005	GPIO2D PU/PD control
BUS_GRF_GPIO3A_P	0x00a0	W	0x0000aaaa	GPIO3A PU/PD control
BUS_GRF_GPIO3B_P	0x00a4	W	0x0000aaaa	GPIO3B PU/PD control
BUS_GRF_GPIO3C_P	0x00a8	W	0x0000aaaa	GPIO3C PU/PD control
BUS_GRF_GPIO3D_P	0x00ac	W	0x000000aa	GPIO3D PU/PD control
BUS_GRF_GPIO4A_P	0x00b0	W	0x00009559	GPIO4A PU/PD control
BUS_GRF_GPIO4B_P	0x00b4	W	0x00005555	GPIO4B PU/PD control
BUS_GRF_GPIO4C_P	0x00b8	W	0x00000155	GPIO4C PU/PD control
BUS_GRF_GPIO1A_SR	0x00c0	W	0x000000ff	GPIO1A slew rate control
BUS_GRF_GPIO1B_SR	0x00c4	W	0x000000ff	GPIO1B slew rate control
BUS_GRF_GPIO2A_SR	0x00d0	W	0x000000ff	GPIO2A slew rate control
BUS_GRF_GPIO2B_SR	0x00d4	W	0x000000ff	GPIO2B slew rate control
BUS_GRF_GPIO2C_SR	0x00d8	W	0x000000ff	GPIO2C slew rate control
BUS_GRF_GPIO2D_SR	0x00dc	W	0x000000ff	GPIO2D slew rate control
BUS_GRF_GPIO3A_SR	0x00e0	W	0x000000ff	GPIO3A slew rate control
BUS_GRF_GPIO3B_SR	0x00e4	W	0x000000ff	GPIO3B slew rate control
BUS_GRF_GPIO3C_SR	0x00e8	W	0x000000ff	GPIO3C slew rate control
BUS_GRF_GPIO3D_SR	0x00ec	W	0x000000ff	GPIO3D slew rate control
BUS_GRF_GPIO4A_SR	0x00f0	W	0x000000ff	GPIO4A slew rate control
BUS_GRF_GPIO4B_SR	0x00f4	W	0x000000ff	GPIO4B slew rate control
BUS_GRF_GPIO4C_SR	0x00f8	W	0x000000ff	GPIO4C slew rate control
BUS_GRF_GPIO1A_SMT	0x0100	W	0x00000000	GPIO1A Schmitt control
BUS_GRF_GPIO1B_SMT	0x0104	W	0x00000000	GPIO1B Schmitt control
BUS_GRF_GPIO1C_SMT	0x0108	W	0x00000000	GPIO1C Schmitt control
BUS_GRF_GPIO1D_SMT	0x010c	W	0x00000000	GPIO1D Schmitt control
BUS_GRF_GPIO2A_SMT	0x0110	W	0x00000000	GPIO2A Schmitt control

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
BUS_GRF_GPIO2B_SMT	0x0114	W	0x00000000	GPIO2B Schmitt control
BUS_GRF_GPIO2C_SMT	0x0118	W	0x00000000	GPIO2C Schmitt control
BUS_GRF_GPIO2D_SMT	0x011c	W	0x00000000	GPIO2D Schmitt control
BUS_GRF_GPIO3A_SMT	0x0120	W	0x00000000	GPIO3A Schmitt control
BUS_GRF_GPIO3B_SMT	0x0124	W	0x00000000	GPIO3B Schmitt control
BUS_GRF_GPIO3C_SMT	0x0128	W	0x00000000	GPIO3C Schmitt control
BUS_GRF_GPIO3D_SMT	0x012c	W	0x00000000	GPIO3D Schmitt control
BUS_GRF_GPIO4A_SMT	0x0130	W	0x00000000	GPIO4A Schmitt control
BUS_GRF_GPIO4B_SMT	0x0134	W	0x00000000	GPIO4B Schmitt control
BUS_GRF_GPIO4C_SMT	0x0138	W	0x00000000	GPIO4C Schmitt control
BUS_GRF_GPIO1A_E	0x0140	W	0x0000aaaa	GPIO1A driver strength control
BUS_GRF_GPIO1B_E	0x0144	W	0x000055a2	GPIO1B driver strength control
BUS_GRF_GPIO2A_E	0x0150	W	0x00005555	GPIO2A driver strength control
BUS_GRF_GPIO2B_E	0x0154	W	0x00005555	GPIO2B driver strength control
BUS_GRF_GPIO2C_E	0x0158	W	0x00005555	GPIO2C driver strength control
BUS_GRF_GPIO2D_E	0x015c	W	0x00000000	GPIO2D driver strength control
BUS_GRF_GPIO3A_E	0x0160	W	0x0000a955	GPIO3A driver strength control
BUS_GRF_GPIO3B_E	0x0164	W	0x0000aaaa	GPIO3B driver strength control
BUS_GRF_GPIO3C_E	0x0168	W	0x0000aaaa	GPIO3C driver strength control
BUS_GRF_GPIO3D_E	0x016c	W	0x000000aa	GPIO3D driver strength control
BUS_GRF_GPIO4A_E	0x0170	W	0x0000aaaa	GPIO4A driver strength control
BUS_GRF_GPIO4B_E	0x0174	W	0x000055aa	GPIO4B driver strength control
BUS_GRF_GPIO4C_E	0x0178	W	0x00000155	GPIO4C driver strength control
BUS_GRF_IOFUNC_CON0	0x0190	W	0x00000000	muti-IOMUX function control register0
BUS_GRF_SOC_CON0	0x0400	W	0x00000000	SOC control register0
BUS_GRF_SOC_CON1	0x0404	W	0x00000000	SOC control register1
BUS_GRF_SOC_CON2	0x0408	W	0x00001000	SOC control register2
BUS_GRF_SOC_CON3	0x040c	W	0x00000000	SOC control register3
BUS_GRF_SOC_CON4	0x0410	W	0x00000000	SOC control register4
BUS_GRF_SOC_CON5	0x0414	W	0x0003a980	SOC control register5
BUS_GRF_VI_CON0	0x0430	W	0x00000000	video input block register0
BUS_GRF_VI_CON1	0x0434	W	0x00000000	video input block register1
BUS_GRF_VI_STATUS	0x0438	W	0x00000000	video input block status register
BUS_GRF_VO_CON0	0x0440	W	0x00000000	video output block register0
BUS_GRF_VO_CON1	0x0444	W	0x00000000	video output block register1
BUS_GRF_SOC_STATUS0	0x0480	W	0x00000000	SOC status register0
BUS_GRF_CPU_CON0	0x0500	W	0x00000000	CPU control register0
BUS_GRF_CPU_CON1	0x0504	W	0x00000384	CPU control register1
BUS_GRF_CPU_CON2	0x0508	W	0x00000000	CPU control register2

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
BUS_GRF_CPU_STATUS0	0x0520	W	0x00000000	CPU status register0
BUS_GRF_CPU_STATUS1	0x0524	W	0x00000000	CPU status register1
BUS_GRF_SOC_NOC_CON0	0x0530	W	0x00000000	NOC control register0
BUS_GRF_SOC_NOC_CON1	0x0534	W	0x00000000	NOC control register1
BUS_GRF_SOC_NOC_CON2	0x0538	W	0x00000000	NOC control register2
BUS_GRF_RAM_CON0	0x0600	W	0x00000000	SRAM control register0
BUS_GRF_RAM_CON1	0x0604	W	0x00000000	SRAM control register1
BUS_GRF_RAM_CON2	0x0608	W	0x00000000	SRAM control register2
BUS_GRF_RAM_CON3	0x060c	W	0x00000000	SRAM control register3
BUS_GRF_RAM_CON4	0x0610	W	0x00000000	SRAM control register4
BUS_GRF_NPUPVTM_CON0	0x0780	W	0x00000000	NPU PVTM control register0
BUS_GRF_NPUPVTM_CON1	0x0784	W	0x00000000	NPU PVTM control register1
BUS_GRF_NPUPVTM_STATUS0	0x0788	W	0x00000000	NPU PVTM status register0
BUS_GRF_NPUPVTM_STATUS1	0x078c	W	0x00000000	NPU PVTM status register1
BUS_GRF_CHIP_ID	0x0800	W	0x00001808	CHIP ID
BUS_GRF_MAC_CON0	0x0900	W	0x00000000	GMAC control register0
BUS_GRF_MAC_CON1	0x0904	W	0x00000000	GMAC control register1

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

4.3.2 Detail Register Description

BUS_GRF_GPIO1A_IOMUX_L

Address: Operational Base + offset (0x0000)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio1a3_sel 4'h0: GPIO 4'h1: EMMC_D3 4'h2: SFC_SIO3 4'h3: Reserved
11:8	RW	0x0	gpio1a2_sel 4'h0: GPIO 4'h1: EMMC_D2 4'h2: SFC_SIO2 4'h3: Reserved
7:4	RW	0x0	gpio1a1_sel 4'h0: GPIO 4'h1: EMMC_D1 4'h2: SFC_SIO1 4'h3: Reserved
3:0	RW	0x0	gpio1a0_sel 4'h0: GPIO 4'h1: EMMC_D0 4'h2: SFC_SIO0 4'h3: Reserved

BUS_GRF_GPIO1A_IOMUX_H
Address: Operational Base + offset (0x0004)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio1a7_sel 4'h0: GPIO 4'h1: EMMC_D7 4'h2: SPI2M0_CLK 4'h3: Reserved
11:8	RW	0x0	gpio1a6_sel 4'h0: GPIO 4'h1: EMMC_D6 4'h2: SPI2M0_MISO 4'h3: Reserved
7:4	RW	0x0	gpio1a5_sel 4'h0: GPIO 4'h1: EMMC_D5 4'h2: SFC_CLK 4'h3: Reserved
3:0	RW	0x0	gpio1a4_sel 4'h0: GPIO 4'h1: EMMC_D4 4'h2: SFC_CSN0 4'h3: Reserved

BUS_GRF_GPIO1B_IOMUX_L

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio1b3_sel 4'h0: GPIO 4'h1: EMMC_RSTN 4'h2: Reserved 4'h3: Reserved
11:8	RW	0x0	gpio1b2_sel 4'h0: GPIO 4'h1: EMMC_CMD 4'h2: Reserved 4'h3: Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7:4	RW	0x0	gpio1b1_sel 4'h0: GPIO 4'h1: EMMC_CLKOUT 4'h2: SPI2M0_CSN 4'h3: Reserved
3:0	RW	0x0	gpio1b0_sel 4'h0: GPIO 4'h1: EMMC_PWREN 4'h2: SPI2M0_MOSI 4'h3: Reserved

BUS_GRF_GPIO1B_IOMUX_H

Address: Operational Base + offset (0x000c)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio1b7_sel 4'h0: GPIO 4'h1: SPI0_CLK 4'h2: PWM_5 4'h3: Reserved
11:8	RW	0x0	gpio1b6_sel 4'h0: GPIO 4'h1: SPI0_CSN 4'h2: PWM_4 4'h3: Reserved
7:4	RW	0x0	gpio1b5_sel 4'h0: GPIO 4'h1: SPI0_MISO 4'h2: I2C2M1_SDA 4'h3: UART1_TXM1
3:0	RW	0x0	gpio1b4_sel 4'h0: GPIO 4'h1: SPI0_MOSI 4'h2: I2C2M1_SCL 4'h3: UART1_RXM1

BUS_GRF_GPIO2A_IOMUX_L

Address: Operational Base + offset (0x0020)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio2a3_sel 4'h0: GPIO 4'h1: CIF_D15M0 4'h2: RGMII_TXD0 4'h3: LCDC_D1M0
11:8	RW	0x0	gpio2a2_sel 4'h0: GPIO 4'h1: CIF_D14M0 4'h2: RGMII_TXD1 4'h3: LCDC_D0M0
7:4	RW	0x0	gpio2a1_sel 4'h0: GPIO 4'h1: CIF_D13M0 4'h2: RGMII_TXEN 4'h3: LCDC_D7M0
3:0	RW	0x0	gpio2a0_sel 4'h0: GPIO 4'h1: CIF_D12M0 4'h2: RGMII_CRS 4'h3: LCDC_D6M0

BUS_GRF_GPIO2A_IOMUX_H

Address: Operational Base + offset (0x0024)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio2a7_sel 4'h0: GPIO 4'h1: CIF_D5M0 4'h2: RGMII_RXDV 4'h3: SPI2M1_CSN
11:8	RW	0x0	gpio2a6_sel 4'h0: GPIO 4'h1: CIF_D4M0 4'h2: RGMII_RXER 4'h3: SPI2M1_MOSI

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7:4	RW	0x0	gpio2a5_sel 4'h0: GPIO 4'h1: CIF_D3M0 4'h2: RGMII_RXD1 4'h3: SPI2M1_CLK
3:0	RW	0x0	gpio2a4_sel 4'h0: GPIO 4'h1: CIF_D2M0 4'h2: RGMII_RXD0 4'h3: SPI2M1_MISO

BUS_GRF_GPIO2B_IOMUX_L

Address: Operational Base + offset (0x0028)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio2b3_sel 4'h0: GPIO 4'h1: CIF_D9M0 4'h2: RGMII_TXD3 4'h3: LCDC_VSYNCM0
11:8	RW	0x0	gpio2b2_sel 4'h0: GPIO 4'h1: CIF_D8M0 4'h2: RGMII_MDC 4'h3: LCDC_HSYNCM0
7:4	RW	0x0	gpio2b1_sel 4'h0: GPIO 4'h1: CIF_D7M0 4'h2: RGMII_COL 4'h3: Reserved
3:0	RW	0x0	gpio2b0_sel 4'h0: GPIO 4'h1: CIF_D6M0 4'h2: RGMII_MDIO 4'h3: Reserved

BUS_GRF_GPIO2B_IOMUX_H

Address: Operational Base + offset (0x002c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio2b7_sel 4'h0: GPIO 4'h1: CIF_CLKOUTM0 4'h2: RGMII_CLK 4'h3: Reserved
11:8	RW	0x0	gpio2b6_sel 4'h0: GPIO 4'h1: CIF_CLKINM0 4'h2: RGMII_RXD3 4'h3: Reserved
7:4	RW	0x0	gpio2b5_sel 4'h0: GPIO 4'h1: CIF_HREFM0 4'h2: RGMII_RXD2 4'h3: Reserved
3:0	RW	0x0	gpio2b4_sel 4'h0: GPIO 4'h1: CIF_VSYNCM0 4'h2: RGMII_TXD2 4'h3: Reserved

BUS_GRF_GPIO2C_IOMUX_L

Address: Operational Base + offset (0x0030)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio2c3_sel 4'h0: GPIO 4'h1: CIF_D11M0 4'h2: Reserved 4'h3: LCDC_D3M0
11:8	RW	0x0	gpio2c2_sel 4'h0: GPIO 4'h1: CIF_D10M0 4'h2: RGMII_RXCLK 4'h3: LCDC_D2M0

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7:4	RW	0x0	gpio2c1_sel 4'h0: GPIO 4'h1: CIF_D1M0 4'h2: RGMII_TXCLK 4'h3: Reserved
3:0	RW	0x0	gpio2c0_sel 4'h0: GPIO 4'h1: CIF_D0M0 4'h2: CLK_OUT_ETHERNET 4'h3: Reserved

BUS_GRF_GPIO2C_IOMUX_H

Address: Operational Base + offset (0x0034)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio2c7_sel 4'h0: GPIO 4'h1: Reserved 4'h2: Reserved 4'h3: LCDC_DENM0
11:8	RW	0x0	gpio2c6_sel 4'h0: GPIO 4'h1: Reserved 4'h2: Reserved 4'h3: LCDC_CLKM0
7:4	RW	0x0	gpio2c5_sel 4'h0: GPIO 4'h1: Reserved 4'h2: Reserved 4'h3: LCDC_D5M0
3:0	RW	0x0	gpio2c4_sel 4'h0: GPIO 4'h1: Reserved 4'h2: Reserved 4'h3: LCDC_D4M0

BUS_GRF_GPIO2D_IOMUX_L

Address: Operational Base + offset (0x0038)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7:4	RW	0x0	gpio2d1_sel 4'h0: GPIO 4'h1: I2C3_SDA 4'h2: UART2_RXM1 4'h3: Reserved
3:0	RW	0x0	gpio2d0_sel 4'h0: GPIO 4'h1: I2C3_SCL 4'h2: UART2_TXM1 4'h3: Reserved

BUS_GRF_GPIO3A_IOMUX_L

Address: Operational Base + offset (0x0040)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio3a3_sel 4'h0: GPIO 4'h1: I2S1_2CH_SDO 4'h2: UART2_TXM2 4'h3: Reserved
11:8	RW	0x0	gpio3a2_sel 4'h0: GPIO 4'h1: I2S1_2CH_MCLK 4'h2: PWM_7 4'h3: Reserved
7:4	RW	0x0	gpio3a1_sel 4'h0: GPIO 4'h1: I2S1_2CH_SCLK 4'h2: PWM_6 4'h3: Reserved
3:0	RW	0x0	gpio3a0_sel 4'h0: GPIO 4'h1: I2S1_2CH_LRCK 4'h2: Reserved 4'h3: Reserved

BUS_GRF_GPIO3A_IOMUX_H

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x0044)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio3a7_sel 4'h0: GPIO 4'h1: I2S0_8CH_SDI1 4'h2: PDM_SDI1 4'h3: Reserved
11:8	RW	0x0	gpio3a6_sel 4'h0: GPIO 4'h1: I2S0_8CH_SDI2 4'h2: PDM_SDI2 4'h3: Reserved
7:4	RW	0x0	gpio3a5_sel 4'h0: GPIO 4'h1: I2S0_8CH_SDI3 4'h2: PDM_SDI3 4'h3: Reserved
3:0	RW	0x0	gpio3a4_sel 4'h0: GPIO 4'h1: I2S1_2CH_SDI 4'h2: UART2_RXM2 4'h3: Reserved

BUS_GRF_GPIO3B_IOMUX_L

Address: Operational Base + offset (0x0048)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio3b3_sel 4'h0: GPIO 4'h1: I2S0_8CH_SDO2 4'h2: I2C2M0_SCL 4'h3: LCDC_VSYNCM1
11:8	RW	0x0	gpio3b2_sel 4'h0: GPIO 4'h1: I2S0_8CH_SDO3 4'h2: ISP_FLASHTRIGIN 4'h3: LCDC_HSYNCM1

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7:4	RW	0x0	gpio3b1_sel 4'h0: GPIO 4'h1: I2S0_8CH_LRCKRX 4'h2: PDM_CLK1 4'h3: Reserved
3:0	RW	0x0	gpio3b0_sel 4'h0: GPIO 4'h1: I2S0_8CH_SCLKRX 4'h2: PDM_CLK0 4'h3: Reserved

BUS_GRF_GPIO3B_IOMUX_H

Address: Operational Base + offset (0x004c)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio3b7_sel 4'h0: GPIO 4'h1: I2S0_8CH_SCLKTX 4'h2: ISP_PRELIGHTTRIG 4'h3: Reserved
11:8	RW	0x0	gpio3b6_sel 4'h0: GPIO 4'h1: I2S0_8CH_LRCKTX 4'h2: ISP_FLASHTRIGOUT 4'h3: Reserved
7:4	RW	0x0	gpio3b5_sel 4'h0: GPIO 4'h1: I2S0_8CH_MCLK 4'h2: ISP_SHUTTEREN 4'h3: Reserved
3:0	RW	0x0	gpio3b4_sel 4'h0: GPIO 4'h1: I2S0_8CH_SDO1 4'h2: I2C2M0_SDA 4'h3: Reserved

BUS_GRF_GPIO3C_IOMUX_L

Address: Operational Base + offset (0x0050)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio3c3_sel 4'h0: GPIO 4'h1: LCDC_D9 4'h2: UART5_RX 4'h3: I2C4_SDA
11:8	RW	0x0	gpio3c2_sel 4'h0: GPIO 4'h1: LCDC_D8 4'h2: UART5_TX 4'h3: I2C4_SCL
7:4	RW	0x0	gpio3c1_sel 4'h0: GPIO 4'h1: I2S0_8CH_SDI0 4'h2: PDM_SDI0 4'h3: Reserved
3:0	RW	0x0	gpio3c0_sel 4'h0: GPIO 4'h1: I2S0_8CH_SDO0 4'h2: ISP_SHUTTERTRIG 4'h3: Reserved

BUS_GRF_GPIO3C_IOMUX_H

Address: Operational Base + offset (0x0054)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio3c7_sel 4'h0: GPIO 4'h1: LCDC_D13 4'h2: UART7_RX 4'h3: SPI1M1_CLK
11:8	RW	0x0	gpio3c6_sel 4'h0: GPIO 4'h1: LCDC_D12 4'h2: UART7_TX 4'h3: Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7:4	RW	0x0	gpio3c5_sel 4'h0: GPIO 4'h1: LCDC_D11 4'h2: UART6_RX 4'h3: Reserved
3:0	RW	0x0	gpio3c4_sel 4'h0: GPIO 4'h1: LCDC_D10 4'h2: UART6_TX 4'h3: Reserved

BUS_GRF_GPIO3D_IOMUX_L

Address: Operational Base + offset (0x0058)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio3d3_sel 4'h0: GPIO 4'h1: LCDC_D17 4'h2: PWM_11 4'h3: SPI1M1_CSN1
11:8	RW	0x0	gpio3d2_sel 4'h0: GPIO 4'h1: LCDC_D16 4'h2: PWM_10 4'h3: SPI1M1_MISO
7:4	RW	0x0	gpio3d1_sel 4'h0: GPIO 4'h1: LCDC_D15 4'h2: PWM_9 4'h3: SPI1M1_CSN0
3:0	RW	0x0	gpio3d0_sel 4'h0: GPIO 4'h1: LCDC_D14 4'h2: PWM_8 4'h3: SPI1M1_MOSI

BUS_GRF_GPIO4A_IOMUX_L

Address: Operational Base + offset (0x0060)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x2	gpio4a3_sel 4'h0: GPIO 4'h1: SDMMC0_D1 4'h2: UART2_RXM0 4'h3: Reserved
11:8	RW	0x2	gpio4a2_sel 4'h0: GPIO 4'h1: SDMMC0_D0 4'h2: UART2_TXM0 4'h3: Reserved
7:4	RW	0x0	gpio4a1_sel 4'h0: GPIO 4'h1: SDMMC0_CLK 4'h2: Reserved 4'h3: Reserved
3:0	RW	0x0	gpio4a0_sel 4'h0: GPIO 4'h1: SDMMC0_CMD 4'h2: TEST_CLK0 4'h3: Reserved

BUS_GRF_GPIO4A_IOMUX_H

Address: Operational Base + offset (0x0064)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio4a7_sel 4'h0: GPIO 4'h1: SDMMC1_CLK 4'h2: Reserved 4'h3: Reserved
11:8	RW	0x0	gpio4a6_sel 4'h0: GPIO 4'h1: SDMMC1_CMD 4'h2: Reserved 4'h3: Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7:4	RW	0x2	gpio4a5_sel 4'h0: GPIO 4'h1: SDMMC0_D3 4'h2: JTAG_TMS 4'h3: Reserved
3:0	RW	0x2	gpio4a4_sel 4'h0: GPIO 4'h1: SDMMC0_D2 4'h2: JTAG_TCK 4'h3: Reserved

BUS_GRF_GPIO4B_IOMUX_L

Address: Operational Base + offset (0x0068)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio4b3_sel 4'h0: GPIO 4'h1: SDMMC1_D3 4'h2: UART1_RTS 4'h3: Reserved
11:8	RW	0x0	gpio4b2_sel 4'h0: GPIO 4'h1: SDMMC1_D2 4'h2: UART1_CTS 4'h3: Reserved
7:4	RW	0x0	gpio4b1_sel 4'h0: GPIO 4'h1: SDMMC1_D1 4'h2: UART1_TXM0 4'h3: Reserved
3:0	RW	0x0	gpio4b0_sel 4'h0: GPIO 4'h1: SDMMC1_D0 4'h2: UART1_RXM0 4'h3: Reserved

BUS_GRF_GPIO4B_IOMUX_H

Address: Operational Base + offset (0x006c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio4b7_sel 4'h0: GPIO 4'h1: UART4_RTS 4'h2: SPI1_MISO 4'h3: PCIUSB_DEBUG3
11:8	RW	0x0	gpio4b6_sel 4'h0: GPIO 4'h1: UART4_CTS 4'h2: SPI1_CSN0 4'h3: PCIUSB_DEBUG2
7:4	RW	0x0	gpio4b5_sel 4'h0: GPIO 4'h1: UART4_TX 4'h2: SPI1_MOSI 4'h3: PCIUSB_DEBUG1
3:0	RW	0x0	gpio4b4_sel 4'h0: GPIO 4'h1: UART4_RX 4'h2: SPI1_CLK 4'h3: PCIUSB_DEBUG0

BUS_GRF_GPIO4C_IOMUX_L

Address: Operational Base + offset (0x0070)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RW	0x0	gpio4c3_sel 4'h0: GPIO 4'h1: Reserved 4'h2: Reserved 4'h3: PCIUSB_DEBUG7
11:8	RW	0x0	gpio4c2_sel 4'h0: GPIO 4'h1: I2C5_SDA 4'h2: Reserved 4'h3: PCIUSB_DEBUG6

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7:4	RW	0x0	gpio4c1_sel 4'h0: GPIO 4'h1: I2C5_SCL 4'h2: Reserved 4'h3: PCIUSB_DEBUG5
3:0	RW	0x0	gpio4c0_sel 4'h0: GPIO 4'h1: Reserved 4'h2: SPI1_CSN1 4'h3: PCIUSB_DEBUG4

BUS_GRF_GPIO4C_IOMUX_H

Address: Operational Base + offset (0x0074)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:4	RO	0x0	reserved
3:0	RW	0x0	gpio4c4_sel 4'h0: GPIO 4'h1: Reserved 4'h2: Reserved 4'h3: Reserved

BUS_GRF_GPIO1A_P

Address: Operational Base + offset (0x0080)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x1	gpio1a7_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
13:12	RW	0x1	gpio1a6_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
11:10	RW	0x1	gpio1a5_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
9:8	RW	0x1	gpio1a4_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
7:6	RW	0x1	gpio1a3_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
5:4	RW	0x1	gpio1a2_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
3:2	RW	0x1	gpio1a1_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
1:0	RW	0x1	gpio1a0_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;

BUS_GRF_GPIO1B_P

Address: Operational Base + offset (0x0084)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x2	gpio1b7_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
13:12	RW	0x1	gpio1b6_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
11:10	RW	0x1	gpio1b5_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
9:8	RW	0x1	gpio1b4_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
7:6	RW	0x2	gpio1b3_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
5:4	RW	0x1	gpio1b2_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
3:2	RW	0x1	gpio1b1_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
1:0	RW	0x1	gpio1b0_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;

BUS_GRF_GPIO2A_P

Address: Operational Base + offset (0x0090)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15:14	RW	0x2	gpio2a7_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
13:12	RW	0x2	gpio2a6_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
11:10	RW	0x2	gpio2a5_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
9:8	RW	0x2	gpio2a4_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
7:6	RW	0x2	gpio2a3_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
5:4	RW	0x2	gpio2a2_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
3:2	RW	0x2	gpio2a1_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
1:0	RW	0x2	gpio2a0_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;

BUS_GRF_GPIO2B_P

Address: Operational Base + offset (0x0094)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x2	gpio2b7_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
13:12	RW	0x2	gpio2b6_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
11:10	RW	0x2	gpio2b5_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
9:8	RW	0x2	gpio2b4_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
7:6	RW	0x2	gpio2b3_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
5:4	RW	0x2	gpio2b2_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
3:2	RW	0x2	gpio2b1_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
1:0	RW	0x2	gpio2b0_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

BUS_GRF_GPIO2C_P

Address: Operational Base + offset (0x0098)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x2	gpio2c7_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
13:12	RW	0x2	gpio2c6_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
11:10	RW	0x2	gpio2c5_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
9:8	RW	0x2	gpio2c4_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
7:6	RW	0x2	gpio2c3_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
5:4	RW	0x2	gpio2c2_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
3:2	RW	0x2	gpio2c1_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
1:0	RW	0x2	gpio2c0_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;

BUS_GRF_GPIO2D_P

Address: Operational Base + offset (0x009c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	gpio2d7_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
13:12	RW	0x0	gpio2d6_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
11:10	RW	0x0	gpio2d5_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
9:8	RW	0x0	gpio2d4_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
7:6	RW	0x0	gpio2d3_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
5:4	RW	0x0	gpio2d2_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
3:2	RW	0x1	gpio2d1_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
1:0	RW	0x1	gpio2d0_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;

BUS_GRF_GPIO3A_P

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x00a0)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x2	gpio3a7_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
13:12	RW	0x2	gpio3a6_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
11:10	RW	0x2	gpio3a5_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
9:8	RW	0x2	gpio3a4_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
7:6	RW	0x2	gpio3a3_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
5:4	RW	0x2	gpio3a2_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
3:2	RW	0x2	gpio3a1_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
1:0	RW	0x2	gpio3a0_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

BUS_GRF_GPIO3B_P
Address: Operational Base + offset (0x00a4)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x2	gpio3b7_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
13:12	RW	0x2	gpio3b6_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
11:10	RW	0x2	gpio3b5_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
9:8	RW	0x2	gpio3b4_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
7:6	RW	0x2	gpio3b3_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
5:4	RW	0x2	gpio3b2_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
3:2	RW	0x2	gpio3b1_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
1:0	RW	0x2	gpio3b0_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;

BUS_GRF_GPIO3C_P

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x00a8)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x2	gpio3c7_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
13:12	RW	0x2	gpio3c6_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
11:10	RW	0x2	gpio3c5_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
9:8	RW	0x2	gpio3c4_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
7:6	RW	0x2	gpio3c3_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
5:4	RW	0x2	gpio3c2_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
3:2	RW	0x2	gpio3c1_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
1:0	RW	0x2	gpio3c0_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

BUS_GRF_GPIO3D_P

Address: Operational Base + offset (0x00ac)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	gpio3d7_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
13:12	RW	0x0	gpio3d6_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
11:10	RW	0x0	gpio3d5_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
9:8	RW	0x0	gpio3d4_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
7:6	RW	0x2	gpio3d3_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
5:4	RW	0x2	gpio3d2_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
3:2	RW	0x2	gpio3d1_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
1:0	RW	0x2	gpio3d0_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;

BUS_GRF_GPIO4A_P

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x00b0)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x2	gpio4a7_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
13:12	RW	0x1	gpio4a6_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
11:10	RW	0x1	gpio4a5_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
9:8	RW	0x1	gpio4a4_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
7:6	RW	0x1	gpio4a3_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
5:4	RW	0x1	gpio4a2_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
3:2	RW	0x2	gpio4a1_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
1:0	RW	0x1	gpio4a0_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

BUS_GRF_GPIO4B_P
Address: Operational Base + offset (0x00b4)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x1	gpio4b7_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
13:12	RW	0x1	gpio4b6_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
11:10	RW	0x1	gpio4b5_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
9:8	RW	0x1	gpio4b4_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
7:6	RW	0x1	gpio4b3_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
5:4	RW	0x1	gpio4b2_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
3:2	RW	0x1	gpio4b1_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
1:0	RW	0x1	gpio4b0_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;

BUS_GRF_GPIO4C_P

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x00b8)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	gpio4c7_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
13:12	RW	0x0	gpio4c6_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
11:10	RW	0x0	gpio4c5_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
9:8	RW	0x1	gpio4c4_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
7:6	RW	0x1	gpio4c3_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
5:4	RW	0x1	gpio4c2_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
3:2	RW	0x1	gpio4c1_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;
1:0	RW	0x1	gpio4c0_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved;

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

BUS_GRF_GPIO1A_SR

Address: Operational Base + offset (0x00c0)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x1	gpio1a7_sr 1'b0: Slow(half frequency) 1'b1: Fast
6	RW	0x1	gpio1a6_sr 1'b0: Slow(half frequency) 1'b1: Fast
5	RW	0x1	gpio1a5_sr 1'b0: Slow(half frequency) 1'b1: Fast
4	RW	0x1	gpio1a4_sr 1'b0: Slow(half frequency) 1'b1: Fast
3	RW	0x1	gpio1a3_sr 1'b0: Slow(half frequency) 1'b1: Fast
2	RW	0x1	gpio1a2_sr 1'b0: Slow(half frequency) 1'b1: Fast
1	RW	0x1	gpio1a1_sr 1'b0: Slow(half frequency) 1'b1: Fast
0	RW	0x1	gpio1a0_sr 1'b0: Slow(half frequency) 1'b1: Fast

BUS_GRF_GPIO1B_SR

Address: Operational Base + offset (0x00c4)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x1	gpio1b7_sr 1'b0: Slow(half frequency) 1'b1: Fast

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
6	RW	0x1	gpio1b6_sr 1'b0: Slow(half frequency) 1'b1: Fast
5	RW	0x1	gpio1b5_sr 1'b0: Slow(half frequency) 1'b1: Fast
4	RW	0x1	gpio1b4_sr 1'b0: Slow(half frequency) 1'b1: Fast
3	RW	0x1	gpio1b3_sr 1'b0: Slow(half frequency) 1'b1: Fast
2	RW	0x1	gpio1b2_sr 1'b0: Slow(half frequency) 1'b1: Fast
1	RW	0x1	gpio1b1_sr 1'b0: Slow(half frequency) 1'b1: Fast
0	RW	0x1	gpio1b0_sr 1'b0: Slow(half frequency) 1'b1: Fast

BUS_GRF_GPIO2A_SR

Address: Operational Base + offset (0x00d0)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x1	gpio2a7_sr 1'b0: Slow(half frequency) 1'b1: Fast
6	RW	0x1	gpio2a6_sr 1'b0: Slow(half frequency) 1'b1: Fast
5	RW	0x1	gpio2a5_sr 1'b0: Slow(half frequency) 1'b1: Fast
4	RW	0x1	gpio2a4_sr 1'b0: Slow(half frequency) 1'b1: Fast

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
3	RW	0x1	gpio2a3_sr 1'b0: Slow(half frequency) 1'b1: Fast
2	RW	0x1	gpio2a2_sr 1'b0: Slow(half frequency) 1'b1: Fast
1	RW	0x1	gpio2a1_sr 1'b0: Slow(half frequency) 1'b1: Fast
0	RW	0x1	gpio2a0_sr 1'b0: Slow(half frequency) 1'b1: Fast

BUS_GRF_GPIO2B_SR
Address: Operational Base + offset (0x00d4)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x1	gpio2b7_sr 1'b0: Slow(half frequency) 1'b1: Fast
6	RW	0x1	gpio2b6_sr 1'b0: Slow(half frequency) 1'b1: Fast
5	RW	0x1	gpio2b5_sr 1'b0: Slow(half frequency) 1'b1: Fast
4	RW	0x1	gpio2b4_sr 1'b0: Slow(half frequency) 1'b1: Fast
3	RW	0x1	gpio2b3_sr 1'b0: Slow(half frequency) 1'b1: Fast
2	RW	0x1	gpio2b2_sr 1'b0: Slow(half frequency) 1'b1: Fast
1	RW	0x1	gpio2b1_sr 1'b0: Slow(half frequency) 1'b1: Fast
0	RW	0x1	gpio2b0_sr 1'b0: Slow(half frequency) 1'b1: Fast

BUS_GRF_GPIO2C_SR

Address: Operational Base + offset (0x00d8)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x1	gpio2c7_sr 1'b0: Slow(half frequency) 1'b1: Fast
6	RW	0x1	gpio2c6_sr 1'b0: Slow(half frequency) 1'b1: Fast
5	RW	0x1	gpio2c5_sr 1'b0: Slow(half frequency) 1'b1: Fast
4	RW	0x1	gpio2c4_sr 1'b0: Slow(half frequency) 1'b1: Fast
3	RW	0x1	gpio2c3_sr 1'b0: Slow(half frequency) 1'b1: Fast
2	RW	0x1	gpio2c2_sr 1'b0: Slow(half frequency) 1'b1: Fast
1	RW	0x1	gpio2c1_sr 1'b0: Slow(half frequency) 1'b1: Fast
0	RW	0x1	gpio2c0_sr 1'b0: Slow(half frequency) 1'b1: Fast

BUS_GRF_GPIO2D_SR

Address: Operational Base + offset (0x00dc)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x1	gpio2d7_sr 1'b0: Slow(half frequency) 1'b1: Fast
6	RW	0x1	gpio2d6_sr 1'b0: Slow(half frequency) 1'b1: Fast

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5	RW	0x1	gpio2d5_sr 1'b0: Slow(half frequency) 1'b1: Fast
4	RW	0x1	gpio2d4_sr 1'b0: Slow(half frequency) 1'b1: Fast
3	RW	0x1	gpio2d3_sr 1'b0: Slow(half frequency) 1'b1: Fast
2	RW	0x1	gpio2d2_sr 1'b0: Slow(half frequency) 1'b1: Fast
1	RW	0x1	gpio2d1_sr 1'b0: Slow(half frequency) 1'b1: Fast
0	RW	0x1	gpio2d0_sr 1'b0: Slow(half frequency) 1'b1: Fast

BUS_GRF_GPIO3A_SR

Address: Operational Base + offset (0x00e0)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x1	gpio3a7_sr 1'b0: Slow(half frequency) 1'b1: Fast
6	RW	0x1	gpio3a6_sr 1'b0: Slow(half frequency) 1'b1: Fast
5	RW	0x1	gpio3a5_sr 1'b0: Slow(half frequency) 1'b1: Fast
4	RW	0x1	gpio3a4_sr 1'b0: Slow(half frequency) 1'b1: Fast
3	RW	0x1	gpio3a3_sr 1'b0: Slow(half frequency) 1'b1: Fast

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2	RW	0x1	gpio3a2_sr 1'b0: Slow(half frequency) 1'b1: Fast
1	RW	0x1	gpio3a1_sr 1'b0: Slow(half frequency) 1'b1: Fast
0	RW	0x1	gpio3a0_sr 1'b0: Slow(half frequency) 1'b1: Fast

BUS_GRF_GPIO3B_SR

Address: Operational Base + offset (0x00e4)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x1	gpio3b7_sr 1'b0: Slow(half frequency) 1'b1: Fast
6	RW	0x1	gpio3b6_sr 1'b0: Slow(half frequency) 1'b1: Fast
5	RW	0x1	gpio3b5_sr 1'b0: Slow(half frequency) 1'b1: Fast
4	RW	0x1	gpio3b4_sr 1'b0: Slow(half frequency) 1'b1: Fast
3	RW	0x1	gpio3b3_sr 1'b0: Slow(half frequency) 1'b1: Fast
2	RW	0x1	gpio3b2_sr 1'b0: Slow(half frequency) 1'b1: Fast
1	RW	0x1	gpio3b1_sr 1'b0: Slow(half frequency) 1'b1: Fast
0	RW	0x1	gpio3b0_sr 1'b0: Slow(half frequency) 1'b1: Fast

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

BUS_GRF_GPIO3C_SR

Address: Operational Base + offset (0x00e8)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x1	gpio3c7_sr 1'b0: Slow(half frequency) 1'b1: Fast
6	RW	0x1	gpio3c6_sr 1'b0: Slow(half frequency) 1'b1: Fast
5	RW	0x1	gpio3c5_sr 1'b0: Slow(half frequency) 1'b1: Fast
4	RW	0x1	gpio3c4_sr 1'b0: Slow(half frequency) 1'b1: Fast
3	RW	0x1	gpio3c3_sr 1'b0: Slow(half frequency) 1'b1: Fast
2	RW	0x1	gpio3c2_sr 1'b0: Slow(half frequency) 1'b1: Fast
1	RW	0x1	gpio3c1_sr 1'b0: Slow(half frequency) 1'b1: Fast
0	RW	0x1	gpio3c0_sr 1'b0: Slow(half frequency) 1'b1: Fast

BUS_GRF_GPIO3D_SR

Address: Operational Base + offset (0x00ec)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x1	gpio3d7_sr 1'b0: Slow(half frequency) 1'b1: Fast

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
6	RW	0x1	gpio3d6_sr 1'b0: Slow(half frequency) 1'b1: Fast
5	RW	0x1	gpio3d5_sr 1'b0: Slow(half frequency) 1'b1: Fast
4	RW	0x1	gpio3d4_sr 1'b0: Slow(half frequency) 1'b1: Fast
3	RW	0x1	gpio3d3_sr 1'b0: Slow(half frequency) 1'b1: Fast
2	RW	0x1	gpio3d2_sr 1'b0: Slow(half frequency) 1'b1: Fast
1	RW	0x1	gpio3d1_sr 1'b0: Slow(half frequency) 1'b1: Fast
0	RW	0x1	gpio3d0_sr 1'b0: Slow(half frequency) 1'b1: Fast

BUS_GRF_GPIO4A_SR

Address: Operational Base + offset (0x00f0)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x1	gpio4a7_sr 1'b0: Slow(half frequency) 1'b1: Fast
6	RW	0x1	gpio4a6_sr 1'b0: Slow(half frequency) 1'b1: Fast
5	RW	0x1	gpio4a5_sr 1'b0: Slow(half frequency) 1'b1: Fast
4	RW	0x1	gpio4a4_sr 1'b0: Slow(half frequency) 1'b1: Fast

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
3	RW	0x1	gpio4a3_sr 1'b0: Slow(half frequency) 1'b1: Fast
2	RW	0x1	gpio4a2_sr 1'b0: Slow(half frequency) 1'b1: Fast
1	RW	0x1	gpio4a1_sr 1'b0: Slow(half frequency) 1'b1: Fast
0	RW	0x1	gpio4a0_sr 1'b0: Slow(half frequency) 1'b1: Fast

BUS_GRF_GPIO4B_SR

Address: Operational Base + offset (0x00f4)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x1	gpio4b7_sr 1'b0: Slow(half frequency) 1'b1: Fast
6	RW	0x1	gpio4b6_sr 1'b0: Slow(half frequency) 1'b1: Fast
5	RW	0x1	gpio4b5_sr 1'b0: Slow(half frequency) 1'b1: Fast
4	RW	0x1	gpio4b4_sr 1'b0: Slow(half frequency) 1'b1: Fast
3	RW	0x1	gpio4b3_sr 1'b0: Slow(half frequency) 1'b1: Fast
2	RW	0x1	gpio4b2_sr 1'b0: Slow(half frequency) 1'b1: Fast
1	RW	0x1	gpio4b1_sr 1'b0: Slow(half frequency) 1'b1: Fast
0	RW	0x1	gpio4b0_sr 1'b0: Slow(half frequency) 1'b1: Fast

BUS_GRF_GPIO4C_SR

Address: Operational Base + offset (0x00f8)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x1	gpio4c7_sr 1'b0: Slow(half frequency) 1'b1: Fast
6	RW	0x1	gpio4c6_sr 1'b0: Slow(half frequency) 1'b1: Fast
5	RW	0x1	gpio4c5_sr 1'b0: Slow(half frequency) 1'b1: Fast
4	RW	0x1	gpio4c4_sr 1'b0: Slow(half frequency) 1'b1: Fast
3	RW	0x1	gpio4c3_sr 1'b0: Slow(half frequency) 1'b1: Fast
2	RW	0x1	gpio4c2_sr 1'b0: Slow(half frequency) 1'b1: Fast
1	RW	0x1	gpio4c1_sr 1'b0: Slow(half frequency) 1'b1: Fast
0	RW	0x1	gpio4c0_sr 1'b0: Slow(half frequency) 1'b1: Fast

BUS_GRF_GPIO1A_SMT

Address: Operational Base + offset (0x0100)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio1a7_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
6	RW	0x0	gpio1a6_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5	RW	0x0	gpio1a5_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
4	RW	0x0	gpio1a4_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
3	RW	0x0	gpio1a3_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
2	RW	0x0	gpio1a2_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
1	RW	0x0	gpio1a1_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
0	RW	0x0	gpio1a0_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

BUS_GRF_GPIO1B_SMT

Address: Operational Base + offset (0x0104)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio1b7_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
6	RW	0x0	gpio1b6_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
5	RW	0x0	gpio1b5_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
4	RW	0x0	gpio1b4_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
3	RW	0x0	gpio1b3_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2	RW	0x0	gpio1b2_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
1	RW	0x0	gpio1b1_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
0	RW	0x0	gpio1b0_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

BUS_GRF_GPIO1C_SMT

Address: Operational Base + offset (0x0108)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio1c7_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
6	RW	0x0	gpio1c6_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
5	RW	0x0	gpio1c5_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
4	RW	0x0	gpio1c4_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
3	RW	0x0	gpio1c3_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
2	RW	0x0	gpio1c2_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
1	RW	0x0	gpio1c1_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
0	RW	0x0	gpio1c0_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

BUS_GRF_GPIO1D_SMT

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x010c)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio1d7_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
6	RW	0x0	gpio1d6_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
5	RW	0x0	gpio1d5_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
4	RW	0x0	gpio1d4_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
3	RW	0x0	gpio1d3_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
2	RW	0x0	gpio1d2_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
1	RW	0x0	gpio1d1_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
0	RW	0x0	gpio1d0_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

BUS_GRF_GPIO2A_SMT

Address: Operational Base + offset (0x0110)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio2a7_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
6	RW	0x0	gpio2a6_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5	RW	0x0	gpio2a5_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
4	RW	0x0	gpio2a4_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
3	RW	0x0	gpio2a3_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
2	RW	0x0	gpio2a2_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
1	RW	0x0	gpio2a1_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
0	RW	0x0	gpio2a0_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

BUS_GRF_GPIO2B_SMT

Address: Operational Base + offset (0x0114)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio2b7_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
6	RW	0x0	gpio2b6_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
5	RW	0x0	gpio2b5_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
4	RW	0x0	gpio2b4_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
3	RW	0x0	gpio2b3_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2	RW	0x0	gpio2b2_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
1	RW	0x0	gpio2b1_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
0	RW	0x0	gpio2b0_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

BUS_GRF_GPIO2C_SMT

Address: Operational Base + offset (0x0118)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio2c7_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
6	RW	0x0	gpio2c6_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
5	RW	0x0	gpio2c5_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
4	RW	0x0	gpio2c4_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
3	RW	0x0	gpio2c3_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
2	RW	0x0	gpio2c2_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
1	RW	0x0	gpio2c1_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
0	RW	0x0	gpio2c0_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

BUS_GRF_GPIO2D_SMT

Address: Operational Base + offset (0x011c)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio2d7_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
6	RW	0x0	gpio2d6_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
5	RW	0x0	gpio2d5_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
4	RW	0x0	gpio2d4_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
3	RW	0x0	gpio2d3_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
2	RW	0x0	gpio2d2_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
1	RW	0x0	gpio2d1_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
0	RW	0x0	gpio2d0_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

BUS_GRF_GPIO3A_SMT

Address: Operational Base + offset (0x0120)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio3a7_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
6	RW	0x0	gpio3a6_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
5	RW	0x0	gpio3a5_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
4	RW	0x0	gpio3a4_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
3	RW	0x0	gpio3a3_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
2	RW	0x0	gpio3a2_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
1	RW	0x0	gpio3a1_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
0	RW	0x0	gpio3a0_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

BUS_GRF_GPIO3B_SMT

Address: Operational Base + offset (0x0124)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio3b7_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
6	RW	0x0	gpio3b6_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
5	RW	0x0	gpio3b5_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
4	RW	0x0	gpio3b4_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
3	RW	0x0	gpio3b3_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
2	RW	0x0	gpio3b2_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
1	RW	0x0	gpio3b1_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
0	RW	0x0	gpio3b0_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

BUS_GRF_GPIO3C_SMT

Address: Operational Base + offset (0x0128)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio3c7_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
6	RW	0x0	gpio3c6_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
5	RW	0x0	gpio3c5_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
4	RW	0x0	gpio3c4_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
3	RW	0x0	gpio3c3_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
2	RW	0x0	gpio3c2_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
1	RW	0x0	gpio3c1_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
0	RW	0x0	gpio3c0_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

BUS_GRF_GPIO3D_SMT

Address: Operational Base + offset (0x012c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio3d7_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
6	RW	0x0	gpio3d6_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
5	RW	0x0	gpio3d5_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
4	RW	0x0	gpio3d4_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
3	RW	0x0	gpio3d3_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
2	RW	0x0	gpio3d2_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
1	RW	0x0	gpio3d1_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
0	RW	0x0	gpio3d0_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

BUS_GRF_GPIO4A_SMT

Address: Operational Base + offset (0x0130)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio4a7_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
6	RW	0x0	gpio4a6_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5	RW	0x0	gpio4a5_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
4	RW	0x0	gpio4a4_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
3	RW	0x0	gpio4a3_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
2	RW	0x0	gpio4a2_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
1	RW	0x0	gpio4a1_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
0	RW	0x0	gpio4a0_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

BUS_GRF_GPIO4B_SMT

Address: Operational Base + offset (0x0134)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio4b7_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
6	RW	0x0	gpio4b6_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
5	RW	0x0	gpio4b5_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
4	RW	0x0	gpio4b4_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
3	RW	0x0	gpio4b3_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2	RW	0x0	gpio4b2_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
1	RW	0x0	gpio4b1_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
0	RW	0x0	gpio4b0_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

BUS_GRF_GPIO4C_SMT

Address: Operational Base + offset (0x0138)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio4c7_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
6	RW	0x0	gpio4c6_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
5	RW	0x0	gpio4c5_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
4	RW	0x0	gpio4c4_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
3	RW	0x0	gpio4c3_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
2	RW	0x0	gpio4c2_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
1	RW	0x0	gpio4c1_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
0	RW	0x0	gpio4c0_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

BUS_GRF_GPIO1A_E

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x0140)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x2	gpio1a7_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
13:12	RW	0x2	gpio1a6_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
11:10	RW	0x2	gpio1a5_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
9:8	RW	0x2	gpio1a4_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
7:6	RW	0x2	gpio1a3_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
5:4	RW	0x2	gpio1a2_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
3:2	RW	0x2	gpio1a1_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
1:0	RW	0x2	gpio1a0_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

BUS_GRF_GPIO1B_E

Address: Operational Base + offset (0x0144)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x1	gpio1b7_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
13:12	RW	0x1	gpio1b6_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
11:10	RW	0x1	gpio1b5_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
9:8	RW	0x1	gpio1b4_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7:6	RW	0x2	gpio1b3_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
5:4	RW	0x2	gpio1b2_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
3:2	RW	0x0	gpio1b1_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
1:0	RW	0x2	gpio1b0_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

BUS_GRF_GPIO2A_E

Address: Operational Base + offset (0x0150)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x1	gpio2a7_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
13:12	RW	0x1	gpio2a6_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
11:10	RW	0x1	gpio2a5_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
9:8	RW	0x1	gpio2a4_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
7:6	RW	0x1	gpio2a3_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
5:4	RW	0x1	gpio2a2_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
3:2	RW	0x1	gpio2a1_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
1:0	RW	0x1	gpio2a0_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

BUS_GRF_GPIO2B_E

Address: Operational Base + offset (0x0154)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15:14	RW	0x1	gpio2b7_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
13:12	RW	0x1	gpio2b6_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
11:10	RW	0x1	gpio2b5_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
9:8	RW	0x1	gpio2b4_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
7:6	RW	0x1	gpio2b3_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
5:4	RW	0x1	gpio2b2_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
3:2	RW	0x1	gpio2b1_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
1:0	RW	0x1	gpio2b0_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

BUS_GRF_GPIO2C_E

Address: Operational Base + offset (0x0158)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x1	gpio2c7_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
13:12	RW	0x1	gpio2c6_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
11:10	RW	0x1	gpio2c5_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
9:8	RW	0x1	gpio2c4_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
7:6	RW	0x1	gpio2c3_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5:4	RW	0x1	gpio2c2_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
3:2	RW	0x1	gpio2c1_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
1:0	RW	0x1	gpio2c0_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

BUS_GRF_GPIO2D_E

Address: Operational Base + offset (0x015c)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	gpio2d7_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
13:12	RW	0x0	gpio2d6_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
11:10	RW	0x0	gpio2d5_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
9:8	RW	0x0	gpio2d4_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
7:6	RW	0x0	gpio2d3_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
5:4	RW	0x0	gpio2d2_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
3:2	RW	0x0	gpio2d1_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
1:0	RW	0x0	gpio2d0_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

BUS_GRF_GPIO3A_E

Address: Operational Base + offset (0x0160)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x2	gpio3a7_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
13:12	RW	0x2	gpio3a6_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
11:10	RW	0x2	gpio3a5_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
9:8	RW	0x1	gpio3a4_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
7:6	RW	0x1	gpio3a3_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
5:4	RW	0x1	gpio3a2_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
3:2	RW	0x1	gpio3a1_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
1:0	RW	0x1	gpio3a0_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

BUS_GRF_GPIO3B_E

Address: Operational Base + offset (0x0164)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x2	gpio3b7_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
13:12	RW	0x2	gpio3b6_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
11:10	RW	0x2	gpio3b5_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
9:8	RW	0x2	gpio3b4_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
7:6	RW	0x2	gpio3b3_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
5:4	RW	0x2	gpio3b2_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
3:2	RW	0x2	gpio3b1_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
1:0	RW	0x2	gpio3b0_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

BUS_GRF_GPIO3C_E

Address: Operational Base + offset (0x0168)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x2	gpio3c7_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
13:12	RW	0x2	gpio3c6_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
11:10	RW	0x2	gpio3c5_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
9:8	RW	0x2	gpio3c4_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7:6	RW	0x2	gpio3c3_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
5:4	RW	0x2	gpio3c2_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
3:2	RW	0x2	gpio3c1_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
1:0	RW	0x2	gpio3c0_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

BUS_GRF_GPIO3D_E

Address: Operational Base + offset (0x016c)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	gpio3d7_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
13:12	RW	0x0	gpio3d6_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
11:10	RW	0x0	gpio3d5_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
9:8	RW	0x0	gpio3d4_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
7:6	RW	0x2	gpio3d3_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
5:4	RW	0x2	gpio3d2_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
3:2	RW	0x2	gpio3d1_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
1:0	RW	0x2	gpio3d0_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

BUS_GRF_GPIO4A_E

Address: Operational Base + offset (0x0170)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15:14	RW	0x2	gpio4a7_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
13:12	RW	0x2	gpio4a6_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
11:10	RW	0x2	gpio4a5_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
9:8	RW	0x2	gpio4a4_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
7:6	RW	0x2	gpio4a3_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
5:4	RW	0x2	gpio4a2_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
3:2	RW	0x2	gpio4a1_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
1:0	RW	0x2	gpio4a0_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

BUS_GRF_GPIO4B_E

Address: Operational Base + offset (0x0174)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x1	gpio4b7_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
13:12	RW	0x1	gpio4b6_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
11:10	RW	0x1	gpio4b5_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
9:8	RW	0x1	gpio4b4_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
7:6	RW	0x2	gpio4b3_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5:4	RW	0x2	gpio4b2_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
3:2	RW	0x2	gpio4b1_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
1:0	RW	0x2	gpio4b0_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

BUS_GRF_GPIO4C_E

Address: Operational Base + offset (0x0178)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	gpio4c7_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
13:12	RW	0x0	gpio4c6_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
11:10	RW	0x0	gpio4c5_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
9:8	RW	0x1	gpio4c4_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
7:6	RW	0x1	gpio4c3_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
5:4	RW	0x1	gpio4c2_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
3:2	RW	0x1	gpio4c1_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
1:0	RW	0x1	gpio4c0_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

BUS_GRF_IOFUNC_CON0

Address: Operational Base + offset (0x0190)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	uart2_sel uart2 rx/tx io selection: 2'b00: RX(GPIO4A3), TX(GPIO4A2) 2'b01: RX(GPIO2D1), TX(GPIO2D0) 2'b10: RX(GPIO3A4), TX(GPIO3A3) 2'b11: USBPHY uart debug port;

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
13	RW	0x0	uart1_sel uart1 rx/tx io selection: 1'b0: RX(GPIO4B0), TX(GPIO4B1) 1'b1: RX(GPIO1B4), TX(GPIO1B5)
12:10	RO	0x0	reserved
9	RW	0x0	i2s1_2ch_lrck_sel i2s1_2ch lrck selection: 1'b0: From rx, 1'b1: From tx
8	RO	0x0	reserved
7:6	RW	0x0	pciephy_debug_sel pcie debug io selection: 2'b00: PCIEPHY LANE0 debug 2'b01: PCIEPHY LANE1 debug 2'b10: USB2PHY status 2'b11: Reserved
5	RW	0x0	spi1_sel SPI1 IO selection 1'b0: MISO(GPIO4B7), SPICLK(GPIO4B4), MOSI(GPIO4B5), CSN0(GPIO4B6), CSN1(GPIO4C0) 1'b1: MISO(GPIO3D2), SPICLK(GPIO3C7), MOSI(GPIO3D0), CSN0(GPIO3D1), CSN1(GPIO3D3)
4	RW	0x0	spi2_sel SPI2 IO selection 1'b0: MISO(GPIO1A6), SPICLK(GPIO1A7), MOSI(GPIO1B0), CSN(GPIO1B1) 1'b1: MISO(GPIO2A4), SPICLK(GPIO2A5), MOSI(GPIO2A6), CSN(GPIO2A7)
3	RW	0x0	i2c2_sel i2c2 IO selection 1'b0: SCL(GPIO3B3), SDA(GPIO3B4) 1'b1: SCL(GPIO1B4), SDA(GPIO1B5)
2	RW	0x0	i2s0_mclk_sel I2S0 mclk source selection from chip CRU to IO 1'b0: I2s0_mclk_rx from cru 1'b1: I2s0_mclk_tx from cru

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
1:0	RW	0x0	i2s0_tx_lrck_sel SCLK input of I2S0 source selection 2'b00: sclk_rx is from GPIO3B0, sclk_tx is from GPIO3B7 2'b01: Both sclk_rx and sclk/tx are from GPIO3B7 2'b10: Both sclk_rx and sclk/tx are from GPIO3B0 2'b11: sclk_rx is from GPIO3B7, sclk_tx is from GPIO3B0 LRCK input of I2S0 source selection 2'b00: lrck_rx is from GPIO3B1, lrck_tx is from GPIO3B6 2'b01: lrck_rx/lrch_tx are both from GPIO3B6 2'b10: lrck_rx/lrck_tx are from GPIO3B1 2'b11: lrck_rx/lrck_tx are from GPIO3B1

BUS_GRF_SOC_CON0

Address: Operational Base + offset (0x0400)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:3	RO	0x0	reserved
2	RW	0x0	tsadc_ana_reg_2 The enable signal of internal bandgap chopper function in temperature sensor 1'b0: Disable 1'b1: Enable
1:0	RO	0x0	reserved

BUS_GRF_SOC_CON1

Address: Operational Base + offset (0x0404)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RO	0x0	reserved
11	RW	0x0	uart7_rts_inv uart7_rts polarity selection 1'b0: Low asserted 1'b1: High asserted
10	RW	0x0	uart7_cts_inv uart7_cts polarity selection 1'b0: Low asserted 1'b1: High asserted

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
9	RW	0x0	uart6_rts_inv uart6_rts polarity selection 1'b0: Low asserted 1'b1: High asserted
8	RW	0x0	uart6_cts_inv uart6_cts polarity selection 1'b0: Low asserted 1'b1: High asserted
7:5	RO	0x0	reserved
4	RW	0x0	tsadc_clk_sel tsadc_clk_sel register of TSADC
3	RW	0x0	tsadc_dig_bypass tsadc_dig_bypass bit register of TSADC
2	RW	0x0	tsadc_tsen_pd_0 tsadc_tsen_pd bit register of TSADC
1:0	RW	0x0	tsadc_chsel tsadc_chsel bit register of TSADC

BUS_GRF_SOC_CON2

Address: Operational Base + offset (0x0408)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	system_mem_intl system memory interleave control 1'b0: Disable 1'b1: Enable
14	RW	0x0	npu_disable_ramclk_gate NPU ram clock gating enable 1'b0: Disable 1'b1: Enable
13	RW	0x0	system_mem_cs_clk_gate_en AutoGate SYSTEM_MEM clock enable 1'b1: Auto gate enable, 1'b0: Disable
12	RW	0x1	wdtns_glb_reset_en WDT NS global reset enable. 1'b0: WDTNS cann't trigger reset. 1'b1: WDTNS can trigger reset

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
11	RW	0x0	uart5_rts_inv uart5_rts polarity selection 1'b0: Low asserted 1'b1: High asserted
10	RW	0x0	uart5_cts_inv uart5_cts polarity selection 1'b0: Low asserted 1'b1: High asserted
9	RW	0x0	uart4_rts_inv uart4_rts polarity selection 1'b0: Low asserted 1'b1: High asserted
8	RW	0x0	uart4_cts_inv uart4_cts polarity selection 1'b0: Low asserted 1'b1: High asserted
7	RW	0x0	uart3_rts_inv uart3_rts polarity selection 1'b0: Low asserted 1'b1: High asserted
6	RW	0x0	uart3_cts_inv uart3_cts polarity selection 1'b0: Low asserted 1'b1: High asserted
5	RW	0x0	uart2_rts_inv uart2_rts polarity selection 1'b0: Low asserted 1'b1: High asserted
4	RW	0x0	uart2_cts_inv uart2_cts polarity selection 1'b0: Low asserted 1'b1: High asserted
3	RW	0x0	uart1_rts_inv uart1_rts polarity selection 1'b0: Low asserted 1'b1: High asserted
2	RW	0x0	uart1_cts_inv uart1_cts polarity selection 1'b0: Low asserted 1'b1: High asserted
1:0	RO	0x0	reserved

BUS_GRF_SOC_CON3

Address: Operational Base + offset (0x040c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0000	saradc_ana_reg SARADC testmode control bits

BUS_GRF_SOC_CON4

Address: Operational Base + offset (0x0410)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	gic_pwr_idle_req 1'b1, Requst to idle niu in PD_GIC 1'b0, No request
14	RW	0x0	pmu_pwr_idle_req 'b1, Requst to idle niu in PD_PMU. 1'b0, No request
13:0	RO	0x0	reserved

BUS_GRF_SOC_CON5

Address: Operational Base + offset (0x0414)

Bit	Attr	Reset Value	Description
31:0	RW	0x0003a980	sdcard_dectn_dly Delay counter setting after sdcard plug out. Count by 24M clock

BUS_GRF_VI_CON0

Address: Operational Base + offset (0x0430)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RO	0x0	reserved
14:13	RW	0x0	isp_cif_if_datawidth 2'b00: 8bit; 2'b01: 10bit; others: 12bit;
12	RW	0x0	dvp_clk_inv_sel 1'b1: DVP path clock inverted; 1'b0: DVP path clock is not inverted
11:10	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
9	RW	0x0	csiphy_clkinv_selection CSI PHY rx byte clock inverted enable 1'b1: Enable 1'b0: Disable
8	RW	0x0	csiphy_clklane_en 1'b1: Enable csiphy clock lane 1'b0: Disable csiphy clock lane
7	RW	0x0	csiphy_datalane_en_3 1'b1: Enable csiphy lane3; 1'b0: Disable csiphy_lane3
6	RW	0x0	csiphy_datalane_en_2 1'b1: Enable csiphy lane2; 1'b0: Disable csiphy_lane2
5	RW	0x0	csiphy_datalane_en_1 1'b1: Enable csiphy lane1; 1'b0: Disable csiphy_lane1
4	RW	0x0	csiphy_datalane_en_0 1'b1: Enable csiphy lane0; 1'b0: Disable csiphy_lane0
3	RW	0x0	csiphy_forcerxmode_3 1'b1: Force to rx mode; 1'b0: Disable force control
2	RW	0x0	csiphy_forcerxmode_2 1'b1: Force to rx mode; 1'b0: Disable force control
1	RW	0x0	csiphy_forcerxmode_1 1'b1: Force to rx mode; 1'b0: Disable force control
0	RW	0x0	csiphy_forcerxmode_0 1'b1: Force to rx mode; 1'b0: Disable force control

BUS_GRF_VI_CON1

Address: Operational Base + offset (0x0434)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RO	0x0	reserved
14:13	RW	0x0	dvp_mainpress QoS control for DVP
12	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
11:10	RW	0x0	isp_mainpress QoS control for ISP
9	RW	0x0	cif_datapath pixel input data path selection for cif controller: 1'b0: Single edge sampling 1'b1: Double edge sampling
8	RW	0x0	cif_clk_delay_en cif clock input delay line enable: 1'b1: Delayline is enabled 1'b0: Delayline is disabled
7	RO	0x0	reserved
6:0	RW	0x00	csiphy_clklane_num The delay value of dvp path clock.This register is valid only when cif_datapath(BUS_GRF_VI_CON1[9]) is set to high

BUS_GRF_VI_STATUS
Address: Operational Base + offset (0x0438)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:11	RO	0x0	reserved
10	RW	0x0	csiphy_errcontentionlp1_0 LP1 Contention error status
9	RO	0x0	csiphy_errcontentionlp0_0 LP0 Contention error status
8	RO	0x0	csiphy_rxskewcalhs_3 Lane3 high-speed receive skew calibration status
7	RO	0x0	csiphy_rxskewcalhs_2 Lane1 high-speed receive skew calibration status
6	RO	0x0	csiphy_rxskewcalhs_1 Lane1 high-speed receive skew calibration status
5	RO	0x0	csiphy_rxskewcalhs_0 Lane0 high-speed receive skew calibration status
4	RO	0x0	csiphy_direction Transmit/Receive direction. 1'b0: transmit mode 1'b1: receive mode
3	RW	0x0	csiphy_ulpsactivenot_3 Lane3 ULP active status This active low signal is asserted to indicate that the Lane is in ULP state
2	RO	0x0	csiphy_ulpsactivenot_2 Lane2 ULP active status This active low signal is asserted to indicate that the Lane is in ULP state
1	RO	0x0	csiphy_ulpsactivenot_1 Lane1 ULP active status This active low signal is asserted to indicate that the Lane is in ULP state
0	RO	0x0	csiphy_ulpsactivenot_0 Lane0 ULP active status This active low signal is asserted to indicate that the Lane is in ULP state

BUS_GRF_VO_CON0

Address: Operational Base + offset (0x0440)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	voplite_press QoS control for voplite
13:12	RW	0x0	vopraw_press QoS control for vopraw
11:10	RW	0x0	dcf_vop_standby_sel 2'b00: ((aclk_vopm_en | dsp_hold_vopm) & (aclk_vops_en | dsp_hold_vops)) 2'b01: Aclk_vopm_en | dsp_hold_vopm Others: Aclk_vops_en | dsp_hold_vops
9	RO	0x0	reserved
8	RW	0x0	dsiphy_selection 1'b0: DSIPHY is connected to csitx 1'b1: DSIPHY is conneted to DSIHOST
7	RW	0x0	dsihost_dpiupdatecfg dsihost_dpiupdatecfg
6:4	RO	0x0	reserved
3	RW	0x0	dsihost_dpicolorm dsihost_dpicolorm
2	RW	0x0	dsihost_dpishutdn dsihost_dpishutdn
1	RW	0x0	voplite_dma_finish_enable 1'b1: Dma_finish from voplite to dcf is enabled 1'b0: Dma_finish from voplite to dcf is disable
0	RW	0x0	vopraw_dma_finish_enable 1'b1: Dma_finish from vopraw to dcf is enabled; 1'b0: Dma_finish from vopraw to dcf is disable

BUS_GRF_VO_CON1

Address: Operational Base + offset (0x0444)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	dsiphy_txskewcalhs_3 Request to transmit skew calibration on lane3. 1'b1: Request; 1'b0: Idle

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
14	RW	0x0	dsiphy_txskewcalhs_2 Request to transmit skew calibration on lane2. 1'b1: Request; 1'b0: Idle
13	RW	0x0	dsiphy_txskewcalhs_1 Request to transmit skew calibration on lane1. 1'b1: Request; 1'b0: Idle
12	RW	0x0	dsiphy_txskewcalhs_0 Request to transmit skew calibration on lane0. 1'b1: Request; 1'b0: Idle
11	RW	0x0	dsiphy_txskewcalhs_ck Request to transmit skew calibration on clock lane. 1'b1: Request; 1'b4: Idle
10	RW	0x0	dsiphy_lane3_frctxstpm Force DSI TX PHY lane3 into transmit mode and generate stop state 1'b0: Disable 1'b1: Enable
9	RW	0x0	dsiphy_lane2_frctxstpm Force DSI TX PHY lane2 into transmit mode and generate stop state 1'b0: Disable 1'b1: Enable
8	RW	0x0	dsiphy_lane1_frctxstpm Force DSI TX PHY lane1 into transmit mode and generate stop state 1'b0: Disable 1'b1: Enable
7	RW	0x0	dsiphy_lane0_frctxstpm Force DSI TX PHY lane0 into transmit mode and generate stop state 1'b0: Disable 1'b1: Enable
6	RW	0x0	dsiphy_forcerxmode Force DSI TX PHY lane module into Receive mode, wait for stop state 1'b0: Disable 1'b1: Enable
5	RW	0x0	dsiphy_lane0_turndisable Disable Turn-around. This signal is used to prevent Lane from going into transmit mode, even if it observes a turn-around request on the Lane interconnect

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4	RW	0x0	lcdc_dclk_inv_sel 1'b0: Normal clock 1'b1: Inverted clock
3	RW	0x0	rgb_bypass 1'b1: Bypass data sync 1'b0: Use data sync
2:0	RO	0x0	reserved

BUS_GRF_SOC_STATUS0

Address: Operational Base + offset (0x0480)

Bit	Attr	Reset Value	Description
31:24	RO	0x00	npu_debug NPU debug state bit
23	RO	0x0	gpio1b6_datain Input data value of GPIO1B6 IO
22	RW	0x0	gic_pwr_idle_status GIC NIU idle acknowledge status
21	RW	0x0	gic_pwr_idle GIC NIU idle status
20	RO	0x0	pmu_pwr_idle_ack PMU NIU idle acknowledge status
19	RW	0x0	pmu_pwr_idle PMU NIU idle status
18	RO	0x0	voplite_dma_finish voplite_dma_finish status
17	RO	0x0	vopraw_dma_finish voprawb_dma_finish status
16	RO	0x0	timer_en_status5 timer5_en status
15	RO	0x0	timer_en_status4 timer4_en status
14	RO	0x0	timer_en_status3 timer3_en status
13	RO	0x0	timer_en_status2 timer2_en status
12	RO	0x0	timer_en_status1 timer1_en status
11	RO	0x0	timer_en_status0 timer0_en status
10:7	RO	0x0	reserved
6	RO	0x0	ddr_cmd_pll_lock DDR CMDPLL lock status
5	RO	0x0	npll_lock NPLL lock status

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4	RO	0x0	ppll_lock PPLL lock status
3	RO	0x0	gpll_lock GPLL lock status
2	RO	0x0	cpll_lock CPLL lock status
1	RO	0x0	dpll_lock DPLL lock status
0	RO	0x0	apll_lock APLL lock status

BUS_GRF_CPU_CON0

Address: Operational Base + offset (0x0500)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RO	0x0	reserved
13:12	RW	0x0	cfgte Enable T32 exception. It sets the initial value of the TE bit in the SCTLR and HSCTLR register. Each bit is defined below. 1'b0: TE bit is low 1'b1: TE bit is high These bits are sampled only during reset of the core. Tie low for the ARM instruction set for exception handling. Tie high for the Thumb instruction set for exception handling. Only change it when the core is in the reset state
11:10	RO	0x0	reserved
9:8	RW	0x0	cfgend Endianness configuration at reset. It sets the initial value of the EE bits in the SCTLR, HSCTLR, SCTLR_EL1, SCTLR_EL2, and SCTLR_EL3 registers. Each bit is defined below. 1'b0: EE bit is low 1'b1: EE bit is high These bits are sampled only during reset of the core. Tie high for big-endian data during exception handling. Tie it low for little-endian data during exception handling. Only change it when the cores are in the state
7:5	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4	RW	0x0	l2rstdisable Disable automatic L2 cache invalidate at reset. 1'b0: enable 1'b1: disable Assert high to disable L2 cache invalidate at reset. Assert low to enable L2 cache invalidate at reset. Only change it when the processor is in the reset state
3:2	RO	0x0	reserved
1:0	RW	0x0	l1rstdisable Disable L1 data cache automatic invalidate on reset functionality. 1'b0: enable 1'b1: disable This value is sampled only during reset of the processor. Assert low for normal L1 data cache behavior on rest. Assert it high to disable automatic invalidation of L1 data cache on reset for debugging purpose only. This bit must be driven Low during normal processor powerup sequences

BUS_GRF_CPU_CON1

Address: Operational Base + offset (0x0504)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RO	0x0	reserved
11	RW	0x0	awuser_mode AXI bus awuser bit4 control of CPU
10	RW	0x0	gic500_axim_err_ack gic500 axim_err_ack port control 1'b0: Disable 1'b1: Enable
9:8	RW	0x3	gic500_cpu_active_0 gic500 cpu_active port control 1'b1: Enable 1'b0: Disable
7	RW	0x1	force_jtag Force select jtag function for GPIO4A4/GPIO4A5 1'b1: Enable 1'b0: Disable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
6	RW	0x0	cpu_ema_detect_en 1'b1: When ema_l2d/emaw_l2d/ema_ra/emaw_ra/emas_ra changed, hardware automaticly stops cpu and make it valiable to cpu; 1'b0: Disable
5	RW	0x0	evento_clear Set this bit to clear the evento rising edge state
4	RW	0x0	eventi Event input for processor wake-up from WFE state. This pin must be asserted for at least one CLKIN clock cycles. When this signal is asserted, it acts as WFE wake-up event to all the cores in the cluster
3	RO	0x0	reserved
2	RW	0x1	dbgromaddrv Debug ROM address valid control 1'b0: Disable 1'b1: Enable
1	RW	0x0	cfgsdisable Prevents modification of certain Secure registers, including bits that correspond to the Lockable SPIs. CFGSDISABLE is typically de-asserted from reset until Secure software has configured the GIC-400 and then subsequently asserted permanently to provide extra security
0	RW	0x0	clrexmonreq Request to clear the external global exclusive monitor. This sends a WFE wake-up event to all cores in the cluster. When set high the global exclusive monitor in the system is requesting the processor EVENT registers to be set high

BUS_GRF_CPU_CON2

Address: Operational Base + offset (0x0508)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	ca35_l2d_ma_wrasd ca35_l2d_ma_wrasd memory bit control
6	RW	0x0	ca35_l2d_ma_wras ca35_l2d_ma_wras memory bit control
5	RW	0x0	ca35_l2d_ma_wl ca35_l2d_ma_wl memory bit control

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4	RW	0x0	ca35_l2d_ma_sawl ca35_l2d_ma_sawl memory bit control
3	RW	0x0	ca35_ma_wrasd ca35_ma_wrasd memory bit control
2	RW	0x0	ca35_ma_wras ca35_ma_wras memory bit control
1	RW	0x0	ca35_ma_wl ca35_ma_wl memory bit control
0	RW	0x0	ca35_ma_sawl ca35_ma_sawl memory bit control

BUS_GRF_CPU_STATUS0

Address: Operational Base + offset (0x0520)

Bit	Attr	Reset Value	Description
31:17	RO	0x0	reserved
16	RW	0x0	l2flushdone High indicates L2 hardware flush complete
15	RW	0x0	clrexmonack High indicates the ack from clrexmonreq
14	RW	0x0	jtagnsw Current TAP mode of operation. High if JTAG selected, low if SWD selected SWJ-DP
13	RW	0x0	jtagtop JTAG TAP controller in one of top 4 states (TLR, RTI, Sel-DR or Sel-IR)
12	RW	0x0	evento_rising_edge When event output rising edge is detected, this bit keeps high until set evento_clear to clear this bit
11:6	RO	0x0	reserved
5	RW	0x0	gic500_axim_err gic500_axim_err status
4	RO	0x0	gic500_ecc_fatal gic500_ecc_fatal status
3:2	RO	0x0	reserved
1:0	RO	0x0	smpnamp Indicate whether a core is taking part in coherency

BUS_GRF_CPU_STATUS1

Address: Operational Base + offset (0x0524)

Bit	Attr	Reset Value	Description
31:13	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
12	RO	0x0	standbywfil2 Indicates whether the L2 memory system is in WFI low-power state. This signal is active when the following conditions are met. 1. All cores are in WFI low-power state, held in reset, or nL2RESET is asserted LOW. 2. If ACE has been configured, ACINACTM is asserted high. 3. If ACP has been configured, AINACTS is asserted high. 4. If CHI has been configured, SINACT is asserted high. 5. L2 memory system is idle. The system power controller must not remove power from the processor when this signal is LOW. This restriction includes configurations where you have set the L2_CACHE_PRESENT parameter to 0
11:6	RO	0x0	reserved
5:4	RO	0x0	standbywfi Indicates whether a core is in WFI low-power state. 1'b0: Core not in WFI low-power state 1'b1: Core in WFI low-power state. This is the reset condition The system power controller must not remove power from an individual core when the corresponding bit of this signal is low
3:2	RO	0x0	reserved
1:0	RO	0x0	standbywfe Indicates whether a core is in WFE low-power state. 1'b0: Core not in WFE low-power state 1'b1: Core in WFE low-power state

BUS_GRF_SOC_NOC_CON0

Address: Operational Base + offset (0x0530)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	system_mem_fwd_system_mem0_system_mem_s0_TPwrStall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
14	RW	0x0	gic_fwd_bus_bus_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
13	RW	0x0	cpu_req_msch_msch_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
12	RW	0x0	cpu_fwd_gic_gic_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
11	RW	0x0	cpu_fwd_bus_Switch18PwrStall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
10	RW	0x0	bus_req_msch_msch_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
9	RW	0x0	bus_mst_fwd_bus_Switch21PwrStall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
8	RW	0x0	bus_fwd_srvmsch_Switch10PwrStall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
7	RW	0x0	bus_fwd_pmu_pm_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
6	RW	0x0	bus_fwd_npu_npu_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
5	RW	0x0	bus_fwd_system_mem_system_mem_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
4	RW	0x0	bus_fwd_gic_gic_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
3	RW	0x0	bus_fwd_ddrc_ddrc_apb_TPwrStall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
2	RW	0x0	bus_fwd_bus_mst_dm_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
1	RW	0x0	bus_fwd_audio_audio_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
0	RW	0x0	audio_fwd_bus_bus_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus

BUS_GRF_SOC_NOC_CON1

Address: Operational Base + offset (0x0534)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	vio_fwd_bus_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
14	RW	0x0	php_mid_fwd_mmcstall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
13	RW	0x0	php_mid_fwd_gmac_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
12	RW	0x0	php_fwd_bus_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
11	RW	0x0	pcie_fwd_bus_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
10	RW	0x0	mmc_fwd_php_mid_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
9	RW	0x0	gmac_fwd_php_mid_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
8	RO	0x0	reserved
7	RW	0x0	bus_fwd_vio_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
6	RW	0x0	bus_fwd_php_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
5	RW	0x0	bus_fwd_pcie_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
4	RW	0x0	npu_req_msch_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
3	RW	0x0	npu_fwd_system_mem_system_mem_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
2:0	RO	0x0	reserved

BUS_GRF_SOC_NOC_CON2

Address: Operational Base + offset (0x0538)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15:14	RW	0x0	msch_split_size This register will decide the splitting size of the interconnect for a transaction from a master and must be set to a proper value according to 'ddrConf' in memory scheduler register. 2'b00: Splitting size is 64 bytes. Must be set to this value when 'ddrConf' is 0~6 or 12~15. 2'b01: Splitting size is 32 bytes. Must be set to this value when 'ddrConf' is 9~10. 2'b10: Splitting size is 16 bytes. Must be set to this value when 'ddrConf' is 7~8 or 11. 2'b11: Reserved, Do not set to this value, otherwise, the system will crash
13:4	RO	0x0	reserved
3	RW	0x0	bus_fwd_vpu_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
2	RW	0x0	pcie_usb_fwd_pcie_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
1	RW	0x0	pcie_fwd_pcie_usb_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus
0	RW	0x0	vpu_fwd_bus_bus_stall Response type when bus is force to idle state 1'b0: bus return error response 1'b1: bus return ok response and hold the bus

BUS_GRF_RAM_CON0

Address: Operational Base + offset (0x0600)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RO	0x0	reserved
13	RW	0x0	npu_ma_tpb Sram timing control register, margin adjust port A
12	RW	0x0	npu_ma_tpa Sram timing control register, margin adjust port A
11	RW	0x0	npu_ma_wrasd Sram timing control register, Write assist function disable
10	RW	0x0	npu_ma_wras Sram timing control register, Write Assist timing
9	RW	0x0	npu_ma_wl Sram timing control register, Wordline pulse only
8	RW	0x0	npu_ma_sawl Sram timing control register, Sense-Amp timing and word line pulse width
7	RO	0x0	reserved
6:4	RW	0x0	system_mem_emaw Sram timing control register, Write Assist timing
3:2	RW	0x0	system_mem_ema Sram timing control register, Wordline pulse only
1:0	RW	0x0	system_mem_emas Sram timing control register, Sense-Amp timing and word line pulse width

BUS_GRF_RAM_CON1

Address: Operational Base + offset (0x0604)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RO	0x0	reserved
13:12	RW	0x0	vio_sp_emaw Sram timing control register
11	RW	0x0	vio_sp_emas Sram timing control register
10:8	RW	0x0	vio_sp_emaa Sram timing control register
7	RW	0x0	vio_dp_emasa Sram timing control register

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
6:4	RW	0x0	vio_dp_emab Sram timing control register
3	RO	0x0	reserved
2:0	RW	0x0	vio_dp_emaa Sram timing control register

BUS_GRF_RAM_CON2

Address: Operational Base + offset (0x0608)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RW	0x0	bus_emaw Sram timing control register
13:11	RW	0x0	bus_ema Sram timing control register
10	RW	0x0	bus_emas Sram timing control register
9	RW	0x0	gic500_ma_wrasd Sram timing control register
8	RW	0x0	gic500_ma_wras Sram timing control register
7	RW	0x0	gic500_ma_wl Sram timing control register
6	RW	0x0	gic500_ma_sawl Sram timing control register
5	RW	0x0	gmac_emab_2 Sram timing control register
4	RW	0x0	gmac_emab_1 Sram timing control register
3	RW	0x0	gmac_emab_0 Sram timing control register
2	RW	0x0	gmac_emaa_1 Sram timing control register
1	RW	0x0	gmac_emaa_0 Sram timing control register
0	RW	0x0	gmac_emasa Sram timing control register

BUS_GRF_RAM_CON3

Address: Operational Base + offset (0x060c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RO	0x0	reserved
11	RW	0x0	audio_ma_wrasd Sram timing control register
10	RW	0x0	audio_ma_wras Sram timing control register
9	RW	0x0	audio_ma_wl Sram timing control register
8	RW	0x0	audio_ma_sawl Sram timing control register
7	RO	0x0	reserved
6:4	RW	0x0	bus_emab Sram timing control register
3	RW	0x0	bus_emasa Sram timing control register
2:0	RW	0x0	bus_emaa Sram timing control register

BUS_GRF_RAM_CON4

Address: Operational Base + offset (0x0610)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:13	RW	0x0	vpu_ema_rom Sram timing control register
13:11	RW	0x0	vpu_emaw Sram timing control register
10	RW	0x0	vpu_emas Sram timing control register
9	RW	0x0	vpu_emasa Sram timing control register
8:6	RW	0x0	vpu_emab Sram timing control register
5:3	RW	0x0	vpu_emaa Sram timing control register
2:0	RW	0x0	vpu_ema Sram timing control register

BUS_GRF_NPUPVTM_CON0

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x0780)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:5	RO	0x0	reserved
4:2	RW	0x0	npupvtm_osc_sel osc_ring selection. 3'b000: osc_ring 0 3'b001: osc_ring 1 3'b010: osc_ring 2 3'b011: osc_ring 3 3'b100: osc_ring 4 Others: reserved
1	RW	0x0	npupvtm_osc_en Set high to enable the osc_ring in the PVTM
0	RW	0x0	npupvtm_start Set high to start PVTM

BUS_GRF_NPUPVTM_CON1

Address: Operational Base + offset (0x0784)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	npupvtm_cal_cnt pvtm calculation counter

BUS_GRF_NPUPVTM_STATUS0

Address: Operational Base + offset (0x0788)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	npupvtm_freq_done High indicates npu pvtm frequency count done

BUS_GRF_NPUPVTM_STATUS1

Address: Operational Base + offset (0x078c)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	npupvtm_freq_cnt Indicates the cycle counts of the osc ring clock

BUS_GRF_CHIP_ID

Address: Operational Base + offset (0x0800)

Bit	Attr	Reset Value	Description
31:0	RO	0x00001808	chip_id 32'h1808

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

BUS_GRF_MAC_CON0

Address: Operational Base + offset (0x0900)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RW	0x00	gmac2io_clk_rx_dl_cfg gmac2io_clk_rx clock delay line control
7:0	RW	0x00	gmac2io_clk_tx_dl_cfg gmac2io_clk_tx clock delay line control

BUS_GRF_MAC_CON1

Address: Operational Base + offset (0x0904)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:7	RO	0x0	reserved
6:4	RW	0x0	gmac2io_phy_intf_sel PHY interface select 3'b001: RGMII 3'b100: RMII All others: Reserved
3	RW	0x0	gmac2io_flowctrl GMAC transmit flow control When set high, instructs the GMAC to transmit PAUSE Control frame in Full-duplex mode. In Half-duplex mode, the GMAC enables the Back-pressure function until this signal is made low again
2	RW	0x0	gmac2io_mac_speed MAC speed 1'b1:100-Mbps 1'b0:10-Mbps
1	RW	0x0	gmac2io_rxclk_dly_ena RGMII RX clock delayline enable 1'b1: Enable 1'b0: Disable
0	RW	0x0	gmac2io_txclk_dly_ena RGMII TX clock delayline enable 1'b1: Enable 1'b0: Disable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

4.4 USB2PHY_GRF Register Description

4.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
USB2PHY_GRF_REG0	0x0000	W	0x00008518	USB PHY Register0
USB2PHY_GRF_REG1	0x0004	W	0x0000e007	USB PHY Register1
USB2PHY_GRF_REG2	0x0008	W	0x000082e7	USB PHY Register2
USB2PHY_GRF_REG3	0x000c	W	0x000002ac	USB PHY Register3
USB2PHY_GRF_REG4	0x0010	W	0x00005556	USB PHY Register4
USB2PHY_GRF_REG5	0x0014	W	0x00005555	USB PHY Register5
USB2PHY_GRF_REG6	0x0018	W	0x00000005	USB PHY Register6
USB2PHY_GRF_REG7	0x001c	W	0x000068c0	USB PHY Register7
USB2PHY_GRF_REG8	0x0020	W	0x00000000	USB PHY Register8
USB2PHY_GRF_REG9	0x0024	W	0x00000000	USB PHY Register9
USB2PHY_GRF_REG10	0x0028	W	0x00000000	USB PHY Register10
USB2PHY_GRF_REG11	0x002c	W	0x00000440	USB PHY Register11
USB2PHY_GRF_REG12	0x0030	W	0x00008518	USB PHY Register12
USB2PHY_GRF_REG13	0x0034	W	0x0000e007	USB PHY Register13
USB2PHY_GRF_REG14	0x0038	W	0x000082e7	USB PHY Register14
USB2PHY_GRF_REG15	0x003c	W	0x000002ac	USB PHY Register15
USB2PHY_GRF_REG16	0x0040	W	0x00005556	USB PHY Register16
USB2PHY_GRF_REG17	0x0044	W	0x00005555	USB PHY Register17
USB2PHY_GRF_REG18	0x0048	W	0x00000005	USB PHY Register18
USB2PHY_GRF_REG19	0x004c	W	0x000068c0	USB PHY Register19
USB2PHY_GRF_REG20	0x0050	W	0x00000000	USB PHY Register20
USB2PHY_GRF_REG21	0x0054	W	0x00000000	USB PHY Register21
USB2PHY_GRF_REG22	0x0058	W	0x00000000	USB PHY Register22
USB2PHY_GRF_REG23	0x005c	W	0x00000440	USB PHY Register23
USB2PHY_GRF_CON0	0x0100	W	0x00000c52	USB PHY control register0
USB2PHY_GRF_CON1	0x0104	W	0x000001d2	USB PHY control register1
USB2PHY_GRF_CON2	0x0108	W	0x00000000	USB PHY control register2
USB2PHY_GRF_CON3	0x010c	W	0x00000019	USB PHY control register3
USB2PHY_GRF_INT_MASK	0x0110	W	0x00000000	USB PHY interrupt mask register
USB2PHY_GRF_INT_STATUS	0x0114	W	0x00000000	USB PHY interrupt status register
USB2PHY_GRF_INT_STATUS_CLR	0x0118	W	0x00000000	USB PHY interrupt status clear register
USB2PHY_GRF_STATUS	0x0120	W	0x00000000	USB PHY status register
USB2PHY_GRF_LS_CON	0x0130	W	0x00030100	USB PHY linestate control register
USB2PHY_GRF_DIS_CON	0x0134	W	0x00030100	USB PHY disconnect control register
USB2PHY_GRF_BVALID_CON	0x0138	W	0x00030100	USB PHY bvalid control register

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
USB2PHY_GRF_ID_CON	0x013c	W	0x00030100	USB PHY id control register
USB2PHY_GRF_CON4	0x0140	W	0x00000000	USB PHY control register3

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

4.4.2 Detail Register Description

USB2PHY_GRF_REG0

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x8518	usbphy_reg0 usbcomb phy control reg. BIT15 to 0

USB2PHY_GRF_REG1

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0xe007	usbphy_reg1 usbcomb phy control reg. BIT31 to 16

USB2PHY_GRF_REG2

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x82e7	usbphy_reg2 usbcomb phy control reg. BIT47 to 32

USB2PHY_GRF_REG3

Address: Operational Base + offset (0x000c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x02ac	usbphy_reg3 usbcomb phy control reg. BIT63 to 48

USB2PHY_GRF_REG4

Address: Operational Base + offset (0x0010)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x5556	usbphy_reg4 usbcomb phy control reg. BIT79 to 64

USB2PHY_GRF_REG5

Address: Operational Base + offset (0x0014)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x5555	usbphy_reg5 usbcomb phy control reg. BIT95 to 80

USB2PHY_GRF_REG6

Address: Operational Base + offset (0x0018)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x0005	usbphy_reg6 usbcomb phy control reg. BIT111 to 96

USB2PHY_GRF_REG7

Address: Operational Base + offset (0x001c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x68c0	usbphy_reg7 usbcomb phy control reg. BIT127 to 112

USB2PHY_GRF_REG8

Address: Operational Base + offset (0x0020)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x0000	usbphy_reg8 usbcomb phy control reg. BIT143 to 128

USB2PHY_GRF_REG9

Address: Operational Base + offset (0x0024)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x0000	usbphy_reg9 usbcomb phy control reg. BIT159 to 144

USB2PHY_GRF_REG10

Address: Operational Base + offset (0x0028)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x0000	usbphy_reg10 usbcomb phy control reg. BIT175 to 160

USB2PHY_GRF_REG11

Address: Operational Base + offset (0x002c)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15:0	RW	0x0440	usbphy_reg11 usbcomb phy control reg. BIT191 to 176

USB2PHY_GRF_REG12

Address: Operational Base + offset (0x0030)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x8518	usbphy_reg12 usbcomb phy control reg. BIT207 to 192

USB2PHY_GRF_REG13

Address: Operational Base + offset (0x0034)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0xe007	usbphy_reg13 usbcomb phy control reg. BIT223 to 208

USB2PHY_GRF_REG14

Address: Operational Base + offset (0x0038)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x82e7	usbphy_reg14 usbcomb phy control reg. BIT239 to 224

USB2PHY_GRF_REG15

Address: Operational Base + offset (0x003c)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x02ac	usbphy_reg15 usbcomb phy control reg. BIT255 to 240

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

USB2PHY_GRF_REG16

Address: Operational Base + offset (0x0040)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x5556	usbphy_reg16 usbcomb phy control reg. BIT271 to 256

USB2PHY_GRF_REG17

Address: Operational Base + offset (0x0044)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x5555	usbphy_reg17 usbcomb phy control reg. BIT287 to 272

USB2PHY_GRF_REG18

Address: Operational Base + offset (0x0048)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x0005	usbphy_reg18 usbcomb phy control reg. BIT303 to 288

USB2PHY_GRF_REG19

Address: Operational Base + offset (0x004c)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x68c0	usbphy_reg19 usbcomb phy control reg. BIT319 to 304

USB2PHY_GRF_REG20

Address: Operational Base + offset (0x0050)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15:0	RW	0x0000	usbphy_reg20 usbcomb phy control reg. BIT335 to 320

USB2PHY_GRF_REG21

Address: Operational Base + offset (0x0054)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x0000	usbphy_reg21 usbcomb phy control reg. BIT351 to 336

USB2PHY_GRF_REG22

Address: Operational Base + offset (0x0058)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x0000	usbphy_reg22 usbcomb phy control reg. BIT367 to 352

USB2PHY_GRF_REG23

Address: Operational Base + offset (0x005c)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:0	RW	0x0440	usbphy_reg23 usbcomb phy control reg. BIT383 to 368

USB2PHY_GRF_CON0

Address: Operational Base + offset (0x0100)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:14	RO	0x0	reserved
13	RW	0x0	usbotg_utmi_dischrgvbus usbotg_utmi_dischrgvbus

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
12	RW	0x0	usbotg_utmi_chrgvbus usbotg_utmi_chrgvbus
11	RW	0x1	usbotg_utmi_idpullup usbotg_utmi_idpullup
10	RW	0x1	usbotg_utmi_iddig GRF USB OTG Plug iddig Indicator
9	RW	0x0	usbotg_utmi_iddig_sel USB OTG plug indicator output selection 1'b0:Select phy iddig status to controller 1'b1:Select GRF plug iddig indicator to controller
8	RW	0x0	usbotg_utmi_dmpulldown GRF OTG DM pulldown resistor
7	RW	0x0	usbotg_utmi_dppulldown GRF OTG DP pulldown resistor
6	RW	0x1	usbotg_utmi_termselect GRF OTG termination select between FS/LS/HS speed
5:4	RW	0x1	usbotg_utmi_xcvrselect GRF OTG transceiver select between FS/LS/HS speed
3:2	RW	0x0	usbotg_utmi_opmode GRF OTG operational mode selection
1	RW	0x1	usbotg_utmi_suspend_n GRF OTG suspend mode 1'b0:Suspend 1'b1:Normal
0	RW	0x0	usbotg_utmi_sel 1'b0:Select OTG controller utmi interface to phy 1'b1:Select GRF utmi interface to phy

USB2PHY_GRF_CON1

Address: Operational Base + offset (0x0104)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:9	RO	0x0	reserved
8	RW	0x1	usbhost_utmi_dmpulldown GRF HOST DM pulldown resistor
7	RW	0x1	usbhost_utmi_dppulldown GRF HOST DP pulldown resistor
6	RW	0x1	usbhost_utmi_termselect GRF HOST termination select between FS/LS/HS speed
5:4	RW	0x1	usbhost_utmi_xcvrselect GRF HOST transceiver select between FS/LS/HS speed

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
3:2	RW	0x0	usbhost_utmi_opmode GRF HOST operational mode selection
1	RW	0x1	usbhost_utmi_suspend_n GRF HOST suspend mode 1'b0:Suspend 1'b1:Normal
0	RW	0x0	usbhost_utmi_sel 1'b0:Select HOST controller utmi interface to phy 1'b1:Select GRF utmi interface to phy

USB2PHY_GRF_CON2

Address: Operational Base + offset (0x0108)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:13	RO	0x0	reserved
12	RW	0x0	vdm_src_en_usbotg open dm voltage source
11	RW	0x0	vdp_src_en_usbotg open dp voltage source
10	RW	0x0	rdm_pdwn_en_usbotg open dm pull down resistor
9	RW	0x0	idp_src_en_usbotg open dm source current
8	RW	0x0	idm_sink_en_usbotg open dm sink current
7	RW	0x0	idp_sink_en_usbotg open dp sink current
6:5	RO	0x0	reserved
4	RW	0x0	usbphy_commononn configure PLL clock output in suspend mode 1'b0:480MHz clock always on 1'b1:480MHz clock will turn off when both ports suspend asserted. If the supsend of any port deassert, it will wait 1ms to make 480MHz clock stable
3	RW	0x0	bypasssel_usbotg bypass select
2	RW	0x0	bypassdmen_usbotg bypass dm enable
1	RW	0x0	usbotg_disable_1 bypass OTG function
0	RW	0x0	usbotg_disable_0 bypass OTG function

USB2PHY_GRF_CON3

Address: Operational Base + offset (0x010c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15	RW	0x0	usbphy_hostport_wakeup_irq_en 1'b0:Disable wakeup irq 1'b1:Enable wakeup irq
14	RO	0x0	reserved
13	RW	0x0	usbotg_utmi_chrgvbus USB OTG GRF charge vbus
12	RO	0x0	reserved
11	RW	0x0	usbotg_utmi_drvvbus USB OTG GRF utmi_drvvbus
10	RW	0x0	usbotg_utmi_drvvbus_sel USB OTG utmi_drvvbus_sel bit control 1'b0:Select OTG controller drvvbus to phy 1'b1:Select OTG GRF utmidrvvbus to phy
9	RW	0x0	usbotg_utmi_fs_se0 USB OTG utmi_fs_se0 bit control
8	RW	0x0	usbotg_utmi_fs_data USB OTG utmi_fs_data bit control
7	RW	0x0	usbotg_utmi_fs_oe USB OTG utmi_fs_oe bit control
6	RW	0x0	usbotg_utmi_fs_xver_own USB OTG utmi_fs_xver_own bit control
5	RW	0x0	usbhost_utmi_idpullup USB HOST utmi_idpullup bit control
4	RW	0x1	usbhost_utmi_dmpulldown Enable DMINUS Pull Down resistor
3	RW	0x1	usbhost_utmi_dppulldown Enable DPLUS Pull Down resistor
2	RW	0x0	usbhost_utmi_dischrgvbus USB HOST utmi_dischrgvbus bit control
1	RW	0x0	usbhost_utmi_chrgvbus USB HOST utmi_chrgvbus bit control
0	RW	0x1	usbhost_utmi_drvvbus USB HOST utmi_drvvbus bit control

USB2PHY_GRF_INT_MASK

Address: Operational Base + offset (0x0110)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1:Write access enable
15:10	RO	0x0	reserved
9:8	RW	0x0	host0_disconnect_irq_en host0_disconnect_irq edge status enable 2'bx1: Disconnect rising edge irq status enable 2'b1x: Disconnect falling edge irq status enable
7:6	RW	0x0	otg0_disconnect_irq_en otg0_disconnect_irq edge status enable 2'bx1: Disconnect rising edge irq status enable 2'b1x: Disconnect falling edge irq status enable
5:4	RW	0x0	otg0_id_irq_en otg0_id edge status enable 2'bx1:Id rising edge irq status enable 2'b1x:Id falling edge irq status enable
3:2	RW	0x0	otg0_bvalid_irq_en otg0_bvalid edge status irq enable 2'bx1:Bvalid rising edge irq status enable 2'b1x:Bvalid falling edge irq status enable
1	RW	0x0	host0_linestate_irq_en host0_linestate change status irq enable
0	RW	0x0	otg0_linestate_irq_en otg0_linestate change status irq enable

USB2PHY_GRF_INT_STATUS

Address: Operational Base + offset (0x0114)

Bit	Attr	Reset Value	Description
31:10	RO	0x0	reserved
9:8	RW	0x0	host0_disconnect_irq host0_disconnect edge irq status 2'bx1: Disconnect rising edge irq status 2'b1x: Disconnect falling edge irq status
7:6	RO	0x0	otg0_disconnect_irq otg0_disconnect edge irq status 2'bx1: Disconnect rising edge irq status 2'b1x: Disconnect falling edge irq status
5:4	RW	0x0	otg0_id_irq otg0_id edge irq status 2'bx1:Id rising edge irq status 2'b1x:Id falling edge irq status

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
3:2	RO	0x0	otg0_bvalid_irq otg0_bvalid edge irq status 2'bx1:Bvalid rising edge irq status 2'b1x:Bvalid falling edge irq status
1	RO	0x0	host0_linestate_irq host0_linestate change irq status
0	RO	0x0	otg0_linestate_irq otg0_linestate change irq status

USB2PHY_GRF_INT_STATUS_CLR

Address: Operational Base + offset (0x0118)

Bit	Attr	Reset Value	Description
31:10	RO	0x0	reserved
9:8	RW	0x0	host0_disconnect_irq_clr host0_disconnect_irq_clr irq status clear 2'b01: Disconnect rising edge irq status clear 2'b10: Disconnect falling edge irq status clear
7:6	WO	0x0	otg0_disconnect_irq_clr otg0_disconnect_irq_clr irq status clear 2'b01: Disconnect rising edge irq status clear 2'b10: Disconnect falling edge irq status clear
5:4	WO	0x0	otg0_id_irq_clr otg0_id edge irq status clear 2'b01:Id rising edge irq status clear 2'b10:Id falling edge irq status clear
3:2	WO	0x0	otg0_bvalid_irq_clr otg0_bvalid edge irq status clear 2'b01:Bvalid rising edge irq status clear 2'b10:Bvalid falling edge irq status clear
1	WO	0x0	host0_linestate_irq_clr host0_linestate change irq status clear, write 1 to clear irq status
0	WO	0x0	otg0_linestate_irq_clr otg0_linestate change irq status clear, write 1 to clear irq status

USB2PHY_GRF_STATUS

Address: Operational Base + offset (0x0120)

Bit	Attr	Reset Value	Description
31:26	RO	0x0	reserved
25	RO	0x0	usbphy_dp_detected usbphy_dp_detected bit status
24	RO	0x0	usbphy_cp_detected usbphy_cp_detected bit status
23	RW	0x0	usbphy_dcp_detected usbphy_dcp_detected bit status

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
22	RO	0x0	usbhost_phy_ls_fs_rcv host_phy_ls_fs_rcv status
21	RO	0x0	usbhost_utmi_avalid host_utmi_avalid status
20	RO	0x0	usbhost_utmi_bvalid host_utmi_bvalid status
19	RO	0x0	usbhost_utmi_hostdisconnect host_utmi_hostdisconnect status
18	RO	0x0	usbhost_utmi_iddig_o host_utmi_iddig status
17:16	RO	0x0	usbhost_utmi_linestate host_utmi_linestate status
15	RO	0x0	usbhost_utmi_sessend host_utmi_sessend status
14	RO	0x0	usbhost_utmi_vbusvalid host_utmi_vbusvalid status
13	RO	0x0	usbhost_utmi_vmi host_utmi_vmi status
12	RO	0x0	usbhost_utmi_vpi host_utmi_vpi status
11	RO	0x0	usbotg_phy_ls_fs_rcv utmi_phy_ls_fs_rcv_out status
10	RO	0x0	usbotg_utmi_avalid otg_utmi avalid bit status
9	RO	0x0	usbotg_utmi_bvalid otg_utmi bvalid bit status
8	RO	0x0	usbotg_utmi_fs_xver_own OTG utmi_fs_xver_own status
7	RO	0x0	usbotg_utmi_hostdisconnect otg_utmi_hostdisconnect status
6	RO	0x0	usbotg_utmi_iddig usbotg_utmi_iddig status
5:4	RO	0x0	usbotg_utmi_linestate otg_utmi_linestate status
3	RO	0x0	usbotg_utmi_sessend otg_utmi_sessend bit status
2	RO	0x0	usbotg_utmi_vbusvalid otg_utmi_vbusvalid bit status
1	RO	0x0	usbotg_utmi_vmi otg_utmi_vmi bit status
0	RO	0x0	usbotg_utmi_vpi otg_utmi_vpi bit status

USB2PHY_GRF_LS_CON

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x0130)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x30100	linestate_filter_con host/otg port linestate filter time control register. Unit:Pclk(up to 100MHz)

USB2PHY_GRF_DIS_CON

Address: Operational Base + offset (0x0134)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x30100	disconnect_filter_con host/otg port hostdisconnect filter time control register. Unit:Pclk(up to 100MHz)

USB2PHY_GRF_BVALID_CON

Address: Operational Base + offset (0x0138)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x30100	bvalid_filter_con otg port bvalid filter time control register. Unit:Pclk(up to 100MHz)

USB2PHY_GRF_ID_CON

Address: Operational Base + offset (0x013c)

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:0	RW	0x0030100	id_filter_con otg ID port filter time control register. Unit:Pclk(up to 100MHz)

USB2PHY_GRF_CON4

Address: Operational Base + offset (0x0140)

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6	RW	0x0	usbotg_utmi_suspend_n usbotg_utmi_suspend_n GRF value
5	RW	0x0	usbotg_utmi_suspend_sel1 1'b0:grf_usbphy_con4[6] 1'b1:Usbotg_utmi_suspend_n & usbotg_utmi_l1_suspend_n
4	RW	0x0	usbotg_utmi_suspend_sel0 1'b0:~usbotg_utmi_suspend_com_n & ~usbotg_utmi_l1_suspend_com_n 1'b1:grf_usbphy_con4[5]?(usbotg_utmi_suspend_n & sbotg_utmi_l1_suspend_n) :grf_usbphy_con4[6];
3:0	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

4.5 PCIe_USB3_PHY_GRF Register Description

4.5.1 Registers Summary

Name	Offset	Size	Reset Value	Description
PCIe_USB3_PHY_GRF_PIPE_CON0	0x0000	W	0x00000000	PCIe_USB3 PHY configuration register0
PCIe_USB3_PHY_GRF_PIPE_CON1	0x0004	W	0x00001452	PCIe_USB3 PHY configuration register1
PCIe_USB3_PHY_GRF_PIPE_CON2	0x0008	W	0x00000000	PCIe_USB3 PHY configuration register2
PCIe_USB3_PHY_GRF_PIPE_CON3	0x000c	W	0x00000000	PCIe_USB3 PHY configuration register3
PCIe_USB3_PHY_GRF_PIPE_CON4	0x0010	W	0x00000000	PCIe_USB3 PHY configuration register3
PCIe_USB3_PHY_GRF_PIPE_STATUS0	0x0030	W	0x00000000	PCIe_USB3 PHY status register0
PCIe_USB3_PHY_GRF_PIPE_STATUS1	0x0034	W	0x00000000	PCIe_USB3 PHY status register1

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

4.5.2 Detail Register Description

PCIe_USB3_PHY_GRF_PIPE_CON0

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15	RW	0x0	sel_pipe_l1_txoneszeros pipe_l1_txoneszeros selection. 1'b0: From PCIe controller 1'b1: From grf Note: Need set 1 when USB3 mode
14	RW	0x0	sel_pipe_l1_txelecidle pipe_l1_txelecidle selection. 1'b0: From PCIe controller 1'b1: From grf Note: Need set 1 when USB3 mode
13	RW	0x0	sel_pipe_l1_txdectrx_loopback pipe_l1_txdectrx_loopback selection. 1'b0: From PCIe controller 1'b1: From grf Note: Need set 1 when USB3 mode

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
12	RW	0x0	sel_pipe_l1_rxtermination pipe_l1_rxtermination selection. 1'b0: From PCIe controller 1'b1: From grf Note: Need set 1 when USB3 mode
11	RW	0x0	sel_pipe_l1_powerdown pipe_l1_powerdown selection. 1'b0: From PCIe controller 1'b1: From grf Note: Need set 1 when USB3 mode
10	RW	0x0	sel_pipe_l0_txoneszeros pipe_l0_txoneszeros selection. 1'b0: From PCIe or USB3 controller 1'b1: From grf note: PCIe does not use this pipe signal, output 0, same to lane1
9	RW	0x0	sel_pipe_l0_txelecidle pipe_l0_txelecidle selection. 1'b0: From PCIe or USB3 controller 1'b1: From grf
8	RW	0x0	sel_pipe_l0_txdectrx_loopback pipe_l0_txdectrx_loopback selection. 1'b0: From PCIe or USB3 controller 1'b1: From grf
7	RW	0x0	sel_pipe_l0_rxtermination pipe_l0_rxtermination selection. 1'b0: From PCIe or USB3 controller 1'b1: From grf note: PCIe does not use this pipe signal, output 0, same to lane1
6	RW	0x0	sel_pipe_l0_powerdown pipe_l0_powerdown selection. 1'b0: From PCIe or USB3 controller 1'b1: From grf
5:4	RO	0x0	reserved
3	RW	0x0	sel_pipe_mac_pclkreq_n pipe_mac_pclkreq_n selection. 1'b0: From PCIe or USB3 controller 1'b1: From grf
2	RW	0x0	sel_pipe_rate pipe_rate selection. 1'b0: From PCIe or USB3 controller 1'b1: From grf
1	RW	0x0	sel_pipe_phymode pipe_phymode selection. 1'b0: From PCIe or USB3 controller 1'b1: From grf

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
0	RW	0x0	sel_pipe_databuswidth pipe_databuswidth selection. 1'b0: From PCIe or USB3 controller 1'b1: From grf

PCIe_USB3_PHY_GRF_PIPE_CON1

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15	RW	0x0	pipe_txcompliance_l1 Lane 1. The MAC asserts this signal for a single clock cycle that matches with the compliance pattern data to start a negative disparity. It is active HIGH. This signal is not required for USB3.0 application
14	RW	0x0	pipe_txcompliance_l0 Lane 0. The MAC asserts this signal for a single clock cycle that matches with the compliance pattern data to start a negative disparity. It is active HIGH. This signal is not required for USB3.0 application
13	RW	0x0	sel_pipe_txcompliance_l1 pipe_txcompliance_l1 selection. 1'b0: From PCIe controller 1'b1: From grf
12	RW	0x1	sel_pipe_txcompliance_l0 pipe_txcompliance_l0 selection. 1'b0: From PCIe controller 1'b1: From grf
11	RW	0x0	sel_pipe_txcommonmode_disable_l1 pipe_txcommonmode_disable_l1 selection. 1'b0: From PCIe controller 1'b1: From grf Note: Need set 1 when USB3 mode
10	RW	0x1	sel_pipe_rxelecidle_disable_l1 pipe_rxelecidle_disable_l1 selection. 1'b0: From PCIe controller 1'b1: From grf Note: Need set 1 when USB3 mode
9	RW	0x0	sel_pipe_txcommonmode_disable_l0 pipe_txcommonmode_disable_l0 selection. 1'b0: From PCIe controller 1'b1: From grf Note: Need set 1 when USB3 mode

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
8	RW	0x0	sel_pipe_rxelecidle_disable_l0 pipe_rxelecidle_disable_l0 selection. 1'b0: From PCIe controller 1'b1: From grf Note: Need set 1 when USB3 mode
7:6	RW	0x1	pipe_mac_pclkreq_n 2'b00: Don't request pipe_clk removal 2'b10: Request pipe_clk removal for executing L1 substates Note: USB3 mode must set to 2'b00
5:4	RW	0x1	pipe_rate phy bit rate: 2'b00: 2.5Gbps, must set to 2.5Gbps during reset in PCIe mode. 2'b01: 5.0Gbps, USB3 only support 5.0Gbps
3:2	RW	0x0	pipe_phymode phy mode: 2'b00: PCIe mode 2'b01: USB3 mode
1:0	RW	0x2	pipe_databuswidth phy pipe data bus width: 2'b00: 32-bit, USB3 only support 32-bit. 2'b01: 16-bit, PCIe only support 16-bit

PCIe_USB3_PHY_GRF_PIPE_CON2

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15	RW	0x0	pipe_txcommonmode_disable_l1 pipe_txcommonmode_disable_l1 Note: Need set 1 when USB3 mode
14	RW	0x0	pipe_rxelecidle_disable_l1 pipe_rxelecidle_disable_l1 Note: Need set 1 when USB3 mode
13	RW	0x0	pipe_l1_txoneszeros pipe_l1_txoneszeros Note: Need set 0 in both PCIe and USB3 mode
12	RW	0x0	pipe_l1_txelecidle pipe_l1_txelecidle Note: Need set 1 when USB3 mode
11	RW	0x0	pipe_l1_txdectrx_loopback pipe_l1_txdectrx_loopback Note: Need set 0 when USB3 mode

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
10	RW	0x0	pipe_l1_rxtermination pipe_l1_rxtermination Note: Need set 1 in both PCIe and USB3 mode
9:8	RW	0x0	pipe_l1_powerdown PIPE Power Up/Down for lane1. Note: Need set 3 when USB3 mode
7	RW	0x0	pipe_txcommonmode_disable_l0 pipe_txcommonmode_disable_l0 Note: Need set 0 when USB3 mode
6	RW	0x0	pipe_rxelecidle_disable_l0 pipe_rxelecidle_disable_l0 Note: Need set 0 when USB3 mode
5	RW	0x0	pipe_l0_txoneszeros pipe_l0_txoneszeros Note: Need set 0 when PCIe mode
4	RW	0x0	pipe_l0_txelecidle pipe_l0_txelecidle
3	RW	0x0	pipe_l0_txdectrx_loopback pipe_l0_txdectrx_loopback
2	RW	0x0	pipe_l0_rxtermination pipe_l0_rxtermination Note: Need set 1 when PCIe mode
1:0	RW	0x0	pipe_l0_powerdown PIPE Power Up/Down for lane0: 2'b00: P0 (L0): normal 2'b01: P0s (L0s): low recovery time, power saving. 2'b10: P1 (L1): longer recovery time, additional power saving. 2'b11: P2 (L2): lowest power state

PCIe_USB3_PHY_GRF_PIPE_CON3

Address: Operational Base + offset (0x000c)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:13	RO	0x0	reserved
12:10	RW	0x0	pipe_txmargin_l1 Transmit margin control for lane 1 3'b000: Normal operation range 3'b001: 800-1200mV for full swing or 400-700mV for half swing 3'b001: 200-400mV for full swing or 100-200mV for half swing Ohter: Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
9:8	RW	0x0	pipe_txdeemph_l1 De-emphasis control for lane 1 2'b00: -6dB 2'b01: -3.5dB 2'b10: No de-emphasis 2'b11: Reserved
7	RW	0x0	pipe_txswing_l1 controls transmitter voltage swing level, for lane 1 1'b0: Full swing 1'b1: Low swing
6:4	RW	0x0	pipe_txmargin_l0 Transmit margin control for lane 0 3'b000: Normal operation range 3'b001: 800-1200mV for full swing or 400-700mV for half swing 3'b001: 200-400mV for full swing or 100-200mV for half swing Ohter: Reserved
3:2	RW	0x0	pipe_txdeemph_l0 De-emphasis control for lane 0 2'b00: -6dB 2'b01: -3.5dB 2'b10: No de-emphasis 2'b11: Reserved
1	RW	0x0	pipe_txswing_l0 controls transmitter voltage swing level, for lane 0 1'b0: Full swing 1'b1: Low swing
0	RW	0x0	PHY mode select: 1’b0: PHY pipe signal from PCIE controller 1’b1: PHY pipe signal from USB3 controller

PCIe_USB3_PHY_GRF_PIPE_CON4

Address: Operational Base + offset (0x0010)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:6	RO	0x0	reserved
5	RW	0x0	sel_pipe_txmargin_l1 PHY control source selection 1'b0: From Controller 1'b1: From PIPE_CON3 register

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4	RW	0x0	sel_pipe_txdeemph_l1 PHY control source selection 1'b0: From Controller 1'b1: From PIPE_CON3 register
3	RW	0x0	sel_pipe_txswing_l1 PHY control source selection 1'b0: From Controller 1'b1: From PIPE_CON3 register
2	RW	0x0	sel_pipe_txmargin_l0 PHY control source selection 1'b0: From Controller 1'b1: From PIPE_CON3 register
1	RW	0x0	sel_pipe_txdeemph_l0 PHY control source selection 1'b0: From Controller 1'b1: From PIPE_CON3 register
0	RW	0x0	sel_pipe_txswing_l0 PHY control source selection 1'b0: From Controller 1'b1: From PIPE_CON3 register

PCIe_USB3_PHY_GRF_PIPE_STATUS0

Address: Operational Base + offset (0x0030)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RW	0x0000	pcie_phy_obs pcie_phy status bit for debug

PCIe_USB3_PHY_GRF_PIPE_STATUS1

Address: Operational Base + offset (0x0034)

Bit	Attr	Reset Value	Description
31:15	RO	0x0	reserved
14	RW	0x0	tx_pll_lock tx pll lock status
13	RW	0x0	rx_cdr_lock_l0 status of rx cdr lock of lane 0
12	RW	0x0	rx_cdr_lock_l1 status of rx cdr lock of lane 1
11	RW	0x0	pipe_clkreq_n pipe_clkreq_n

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
10	RW	0x0	pipe_power_presetn PIPE Power Present This PIPE output indicates the presence of VBUS. If V BUS is connected to the PHY, this PIPE interface signal is generated by an internal VBUS comparator. This is an asynchronous signal
9	RO	0x0	pipe_rxelecidle_o_l0 PIPE Receiver Electrical Idle. Lane 0 Indicates receiver detection of an electrical idle. Signal desertion while the PHY is in a P0, P1, P2, or P3 state indicates the detection of Low Frequency Periodic Signaling (LFPS)
8	RW	0x0	pipe_rxelecidle_o_l1 PIPE Receiver Electrical Idle. Lane 1 Indicates receiver detection of an electrical idle. Signal desertion while the PHY is in a P0, P1, P2, or P3 state indicates the detection of Low Frequency Periodic Signaling (LFPS)
7	RO	0x0	pipe_phystatus_o_l0 PIPE PHY Status, Lane 1 Communicates completion of several PHY functions including power management state transitions, rate change, and receiver detection. When this signal transitions during entry and exit from P3 states and PCLK is not running, the signaling is asynchronous. In error situations (where the PHY fails to assert PhyStatus), the MAC can take MAC-specific error recovery actions
6	RO	0x0	pipe_phystatus_o_l1 PIPE PHY Status, Lane 1 Communicates completion of several PHY functions including power management state transitions, rate change, and receiver detection. When this signal transitions during entry and exit from P3 states and PCLK is not running, the signaling is asynchronous. In error situations (where the PHY fails to assert PhyStatus), the MAC can take MAC-specific error recovery actions
5:3	RO	0x0	pipe_rxstatus_o_l0 PIPE Receiver Status. Lane 0 Encodes receiver status and error codes for the received data stream when receiving data. 3'b000: received data OK 3'b001: one SKP Ordered set added 3'b010: one SKP Ordered set removed 3'b011: receiver detection 3'b100: 8B/10B decode error 3'b101: elastic buffer overflow 3'b110: elastic buffer underflow 3'b111: receive disparity error, overwritten by decode error

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2:0	RW	0x0	pipe_rxstatus_o_l1 PIPE Receiver Status. Lane 1 Encodes receiver status and error codes for the received data stream when receiving data. 3'b000: received data OK 3'b001: one SKP Ordered set added 3'b010: one SKP Ordered set removed 3'b011: receiver detection 3'b100: 8B/10B decode error 3'b101: elastic buffer overflow 3'b110: elastic buffer underflow 3'b111: receive disparity error, overwritten by decode error

4.6 PMU_GRF Register Description

4.6.1 Registers Summary

Name	Offset	Size	Reset Value	Description
PMU_GRF_GPIO0A_IOMUX	0x0000	W	0x00001144	GPIO0A IOMUX control low bits
PMU_GRF_GPIO0B_IOMUX	0x0004	W	0x00001000	GPIO0B IOMUX control low bits
PMU_GRF_GPIO0C_IOMUX	0x0008	W	0x00000000	GPIO0C IOMUX control low bits
PMU_GRF_GPIO0A_P	0x0010	W	0x00004652	GPIO0A PU/PD control
PMU_GRF_GPIO0B_P	0x0014	W	0x000095a5	GPIO0B PU/PD control
PMU_GRF_GPIO0C_P	0x0018	W	0x0000aa8a	GPIO0C PU/PD control
PMU_GRF_GPIO0A_E	0x0020	W	0x00000001	GPIO0A driver strength control
PMU_GRF_GPIO0B_E	0x0024	W	0x00000000	GPIO0B driver strength control
PMU_GRF_GPIO0C_E	0x0028	W	0x00000000	GPIO0C driver strength control
PMU_GRF_GPIO0A_SR	0x0030	W	0x00000000	GPIO0A slow rate control
PMU_GRF_GPIO0B_SR	0x0034	W	0x00000000	GPIO0B slow rate control
PMU_GRF_GPIO0C_SR	0x0038	W	0x00000000	GPIO0C slow rate control
PMU_GRF_GPIO0A_SMT	0x0040	W	0x00000000	GPIO0A Schmitt control
PMU_GRF_GPIO0B_SMT	0x0044	W	0x00000000	GPIO0B Schmitt control
PMU_GRF_GPIO0C_SMT	0x0048	W	0x00000000	GPIO0C Schmitt control
PMU_GRF_SOC_CON0	0x0100	W	0x00000010	PMU SOC control register0
PMU_GRF_SOC_CON1	0x0104	W	0x00000000	PMU SOC control register1
PMU_GRF_SOC_CON2	0x0108	W	0x000012c0	PMU SOC control register2
PMU_GRF_SOC_CON3	0x010c	W	0x0000001c	PMU SOC control register3
PMU_GRF_SOC_CON4	0x0110	W	0x00000000	PMU SOC control register4
PMU_GRF_SOC_STATUS	0x0120	W	0x00000000	PMU SOC status register
PMU_GRF_IO_VSEL0	0x0140	W	0x00000000	IO voltage seletion register0
PMU_GRF_IO_VSEL1	0x0144	W	0x00000000	IO voltage seletion register1
PMU_GRF_IO_VSEL_STATUS	0x0148	W	0x00000000	IO voltage status register
PMU_GRF_PVTM_CON0	0x0180	W	0x00000003	PVTM control register0

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
PMU_GRF_PVTM_CON1	0x0184	W	0x00000100	PVTM control register1
PMU_GRF_PVTM_STATUS0	0x0190	W	0x00000000	PVTM status register0
PMU_GRF_PVTM_STATUS1	0x0194	W	0x00000000	PVTM status register1
PMU_GRF_OS_REG0	0x0200	W	0x00000000	PMU GRF OS register0
PMU_GRF_OS_REG1	0x0204	W	0x00000000	PMU GRF OS register1
PMU_GRF_OS_REG2	0x0208	W	0x00000000	PMU GRF OS register2
PMU_GRF_OS_REG3	0x020c	W	0x00000000	PMU GRF OS register3
PMU_GRF_OS_REG4	0x0210	W	0x00000000	PMU GRF OS register4
PMU_GRF_OS_REG5	0x0214	W	0x00000000	PMU GRF OS register5
PMU_GRF_OS_REG6	0x0218	W	0x00000000	PMU GRF OS register6
PMU_GRF_OS_REG7	0x021c	W	0x00000000	PMU GRF OS register7
PMU_GRF_OS_REG8	0x0220	W	0x00000000	PMU GRF OS register8
PMU_GRF_OS_REG9	0x0224	W	0x00000000	PMU GRF OS register9
PMU_GRF_OS_REG10	0x0228	W	0x00000000	PMU GRF OS register10
PMU_GRF_OS_REG11	0x022c	W	0x00000000	PMU GRF OS register11
PMU_GRF_RESET_FUNCTION_STATUS	0x0230	W	0x00000000	system reset status register
PMU_GRF_RESET_FUNCTION_CLR	0x0234	W	0x00000000	system reset status clear register
PMU_GRF_SIG_DETECT_CON	0x0380	W	0x00000000	sdmmc detect control reg
PMU_GRF_SIG_DETECT_STATUS	0x0390	W	0x00000000	sdmmc detect status reg
PMU_GRF_SIG_DETECT_STATUS_CLEAR	0x03a0	W	0x00000000	sdmmc irq clear reg
PMU_GRF_SDMMC_DET_COUNTER	0x03b0	W	0x000003e8	sdmmc detect counter reg

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

4.6.2 Detail Register Description

PMU_GRF_GPIO0A_IOMUX

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable rite enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:14	RW	0x0	gpio0a7_sel 2'b00: GPIO 2'b01: PCIe_WAKEm0 2'b10: Reserved 2'b11: Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
13:12	RW	0x1	gpio0a6_sel 2'b00: GPIO 2'b01: TSADC_SHUTM0 2'b10: TSADC_SHUTORG 2'b11: Reserved
11:10	RW	0x0	gpio0a5_sel 2'b00: GPIO 2'b01: PCIe_PRSTNm0 2'b10: Reserved 2'b11: Reserved
9:8	RW	0x1	gpio0a4_sel 2'b00: GPIO 2'b01: PMIC_SLEEP 2'b10: Reserved 2'b11: Reserved
7:6	RW	0x1	gpio0a3_sel 2'b00: GPIO 2'b01: SDMMC0_DETN 2'b10: PCIe_REQNm0 2'b11: Reserved
5:4	RW	0x0	gpio0a2_sel 2'b00: GPIO 2'b01: PCIe_BUTTONRST 2'b10: Reserved 2'b11: Reserved
3:2	RW	0x1	gpio0a1_sel 2'b00: GPIO 2'b01: TSADC_SHUTM1 2'b10: Reserved 2'b11: Reserved
1:0	RW	0x0	gpio0a0_sel 2'b00: GPIO 2'b01: CLK_OUT_WIFI 2'b10: Reserved 2'b11: Reserved

PMU_GRF_GPIO0B_IOMUX

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15:14	RW	0x0	gpio0b7_sel 2'b00: GPIO 2'b01: PWM_0 2'b10: OTG_DRV 2'b11: Reserved
13:12	RW	0x1	gpio0b6_sel 2'b00: GPIO 2'b01: FLASH_VOL_SEL 2'b10: PCIe_PRSTNm1 2'b11: Reserved
11:10	RW	0x0	gpio0b5_sel 2'b00: GPIO 2'b01: UART0_RTS 2'b10: TEST_CLK1 2'b11: Reserved
9:8	RW	0x0	gpio0b4_sel 2'b00: GPIO 2'b01: UART0_CTS 2'b10: PMU_DEBUG2 2'b11: PMU_DEBUG_SOUT
7:6	RW	0x0	gpio0b3_sel 2'b00: GPIO 2'b01: UART0_RX 2'b10: PMU_DEBUG1 2'b11: Reserved
5:4	RW	0x0	gpio0b2_sel 2'b00: GPIO 2'b01: UART0_TX 2'b10: PMU_DEBUG0 2'b11: Reserved
3:2	RW	0x0	gpio0b1_sel 2'b00: GPIO 2'b01: I2C0_SDA 2'b10: Reserved 2'b11: Reserved
1:0	RW	0x0	gpio0b0_sel 2'b00: GPIO 2'b01: I2C0_SCL 2'b10: Reserved 2'b11: Reserved

PMU_GRF_GPIO0C_IOMUX

Address: Operational Base + offset (0x0008)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:14	RW	0x0	gpio0c7_sel 2'b00: GPIO 2'b01: Reserved 2'b10: UART3_RTSm0 2'b11: Reserved
13:12	RW	0x0	gpio0c6_sel 2'b00: GPIO 2'b01: PCIe_REQNm1 2'b10: UART3_CTSm0 2'b11: Reserved
11:10	RW	0x0	gpio0c5_sel 2'b00: GPIO 2'b01: PWM_2 2'b10: PCIe_WAKEm1 2'b11: Reserved
9:8	RW	0x0	gpio0c4_sel 2'b00: GPIO 2'b01: PWM_3 2'b10: UART3_RXm0 2'b11: PMU_DEBUG4
7:6	RW	0x0	gpio0c3_sel 2'b00: GPIO 2'b01: PWM_1 2'b10: UART3_TXM0 2'b11: PMU_DEBUG3
5:4	RW	0x0	gpio0c2_sel 2'b00: GPIO 2'b01: CLK_INOUT_32K 2'b10: Reserved 2'b11: Reserved
3:2	RW	0x0	gpio0c1_sel 2'b00: GPIO 2'b01: I2C1_SDA 2'b10: Reserved 2'b11: Reserved
1:0	RW	0x0	gpio0c0_sel 2'b00: GPIO 2'b01: I2C1_SCL 2'b10: Reserved 2'b11: PMU_DEBUG5

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

PMU_GRF_GPIO0A_P
Address: Operational Base + offset (0x0010)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:14	RW	0x1	gpio0a7_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
13:12	RW	0x0	gpio0a6_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
11:10	RW	0x1	gpio0a5_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
9:8	RW	0x2	gpio0a4_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
7:6	RW	0x1	gpio0a3_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
5:4	RW	0x1	gpio0a2_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
3:2	RW	0x0	gpio0a1_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
1:0	RW	0x2	gpio0a0_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

PMU_GRF_GPIO0B_P
Address: Operational Base + offset (0x0014)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:14	RW	0x2	gpio0b7_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
13:12	RW	0x1	gpio0b6_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
11:10	RW	0x1	gpio0b5_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
9:8	RW	0x1	gpio0b4_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
7:6	RW	0x2	gpio0b3_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
5:4	RW	0x2	gpio0b2_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
3:2	RW	0x1	gpio0b1_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
1:0	RW	0x1	gpio0b0_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved

PMU_GRF_GPIO0C_P

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x0018)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:14	RW	0x2	gpio0c7_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
13:12	RW	0x2	gpio0c6_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
11:10	RW	0x2	gpio0c5_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
9:8	RW	0x2	gpio0c4_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
7:6	RW	0x2	gpio0c3_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
5:4	RW	0x0	gpio0c2_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
3:2	RW	0x2	gpio0c1_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved
1:0	RW	0x2	gpio0c0_p 2'b00: Z(Normal operation); 2'b01: Weak 1(pull-up); 2'b10: Weak 0(pull-down); 2'b11: Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

PMU_GRF_GPIO0A_E

Address: Operational Base + offset (0x0020)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:14	RW	0x0	gpio0a7_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
13:12	RW	0x0	gpio0a6_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
11:10	RW	0x0	gpio0a5_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
9:8	RW	0x0	gpio0a4_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
7:6	RW	0x0	gpio0a3_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
5:4	RW	0x0	gpio0a2_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
3:2	RW	0x0	gpio0a1_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
1:0	RW	0x1	gpio0a0_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

PMU_GRF_GPIO0B_E

Address: Operational Base + offset (0x0024)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:14	RW	0x0	gpio0b7_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
13:12	RW	0x0	gpio0b6_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
11:10	RW	0x0	gpio0b5_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
9:8	RW	0x0	gpio0b4_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7:6	RW	0x0	gpio0b3_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
5:4	RW	0x0	gpio0b2_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
3:2	RW	0x0	gpio0b1_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
1:0	RW	0x0	gpio0b0_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

PMU_GRF_GPIO0C_E

Address: Operational Base + offset (0x0028)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:14	RW	0x0	gpio0c7_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
13:12	RW	0x0	gpio0c6_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
11:10	RW	0x0	gpio0c5_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
9:8	RW	0x0	gpio0c4_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
7:6	RW	0x0	gpio0c3_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
5:4	RW	0x0	gpio0c2_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

3:2	RW	0x0	gpio0c1_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF
1:0	RW	0x0	gpio0c0_e drive strength control 2'b00: 2pF 2'b01: 4pF 2'b10: 8pF 2'b11: 16pF

PMU_GRF_GPIO0A_SR

Address: Operational Base + offset (0x0030)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio0a7_sr 1'b0: Slow(half frequency) 1'b1: Fast
6	RW	0x0	gpio0a6_sr 1'b0: Slow(half frequency) 1'b1: Fast
5	RW	0x0	gpio0a5_sr 1'b0: Slow(half frequency) 1'b1: Fast
4	RW	0x0	gpio0a4_sr 1'b0: Slow(half frequency) 1'b1: Fast
3	RW	0x0	gpio0a3_sr 1'b0: Slow(half frequency) 1'b1: Fast
2	RW	0x0	gpio0a2_sr 1'b0: Slow(half frequency) 1'b1: Fast
1	RW	0x0	gpio0a1_sr 1'b0: Slow(half frequency) 1'b1: Fast
0	RW	0x0	gpio0a0_sr 1'b0: Slow(half frequency) 1'b1: Fast

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

PMU_GRF_GPIO0B_SR

Address: Operational Base + offset (0x0034)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio0b7_sr 1'b0: Slow(half frequency) 1'b1: Fast
6	RW	0x0	gpio0b6_sr 1'b0: Slow(half frequency) 1'b1: Fast
5	RW	0x0	gpio0b5_sr 1'b0: Slow(half frequency) 1'b1: Fast
4	RW	0x0	gpio0b4_sr 1'b0: Slow(half frequency) 1'b1: Fast
3	RW	0x0	gpio0b3_sr 1'b0: Slow(half frequency) 1'b1: Fast
2	RW	0x0	gpio0b2_sr 1'b0: Slow(half frequency) 1'b1: Fast
1	RW	0x0	gpio0b1_sr 1'b0: Slow(half frequency) 1'b1: Fast
0	RW	0x0	gpio0b0_sr 1'b0: Slow(half frequency) 1'b1: Fast

PMU_GRF_GPIO0C_SR

Address: Operational Base + offset (0x0038)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio0c7_sr 1'b0: Slow(half frequency) 1'b1: Fast
6	RW	0x0	gpio0c6_sr 1'b0: Slow(half frequency) 1'b1: Fast
5	RW	0x0	gpio0c5_sr 1'b0: Slow(half frequency) 1'b1: Fast
4	RW	0x0	gpio0c4_sr 1'b0: Slow(half frequency) 1'b1: Fast
3	RW	0x0	gpio0c3_sr 1'b0: Slow(half frequency) 1'b1: Fast
2	RW	0x0	gpio0c2_sr 1'b0: Slow(half frequency) 1'b1: Fast
1	RW	0x0	gpio0c1_sr 1'b0: Slow(half frequency) 1'b1: Fast
0	RW	0x0	gpio0c0_sr 1'b0: Slow(half frequency) 1'b1: Fast

PMU_GRF_GPIO0A_SMT

Address: Operational Base + offset (0x0040)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio0a7_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
6	RW	0x0	gpio0a6_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
5	RW	0x0	gpio0a5_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
4	RW	0x0	gpio0a4_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
3	RW	0x0	gpio0a3_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
2	RW	0x0	gpio0a2_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
1	RW	0x0	gpio0a1_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
0	RW	0x0	gpio0a0_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

PMU_GRF_GPIO0B_SMT

Address: Operational Base + offset (0x0044)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio0b7_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
6	RW	0x0	gpio0b6_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5	RW	0x0	gpio0b5_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
4	RW	0x0	gpio0b4_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
3	RW	0x0	gpio0b3_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
2	RW	0x0	gpio0b2_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
1	RW	0x0	gpio0b1_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
0	RW	0x0	gpio0b0_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

PMU_GRF_GPIO0C_SMT

Address: Operational Base + offset (0x0048)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	gpio0c7_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
6	RW	0x0	gpio0c6_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
5	RW	0x0	gpio0c5_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
4	RW	0x0	gpio0c4_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
3	RW	0x0	gpio0c3_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2	RW	0x0	gpio0c2_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
1	RW	0x0	gpio0c1_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled
0	RW	0x0	gpio0c0_smt 1'b0: No hysteresis 1'b1: Schmitt trigger enabled

PMU_GRF_SOC_CON0

Address: Operational Base + offset (0x0100)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:14	RO	0x0	reserved
13	RW	0x0	ddrphy_bufferen_core 1'b0: Enable ddrphy io retention; 1'b1: Disable ddrphy io retention;
12	RW	0x0	ddrphy_bufferen_sel 1'b0: Ddrphy_bufferen from pmu and ddr_fail_safe 1'b1: Ddrphy_bufferen from ddrphy_bufferen_core
11	RO	0x0	reserved
10	RW	0x0	grf_con_pcie_reqn_sel 1'b0: gpio0c6 selected 1'b1: gpio0a3 selected
9	RW	0x0	grf_pci_wake_sel 1'b0: gpio0c5 selected 1'b1: gpio0a7 selected
8	RW	0x0	grf_con_pcie_rstn_sel 1'b0: gpio0b6 selected 1'b1: gpio0a5 selected
7	RW	0x0	grf_con_pmic_sleep_sel pmic sleep function selection 1'b0: From reset pulse generator, can reset external PMIC 1'b1: From pmu block, only suppport sleep function for external PMIC
6	RW	0x0	uart0_cts_sel uart0_cts polarity selection 1'b0: Low asserted 1'b1: High asserted

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5	RW	0x0	uart0_rts_sel uart0_rts polarity selection 1'b0: Low asserted 1'b1: High asserted
4:1	RW	0x8	osc_gm_sel OSC amplifier gain setting (compensates the basic architecture of the crystal oscilator pads): 4'h0: 2mS 4'h1: 4mS 4'h2: 6mS … 4'h15: 32mS
0	RW	0x0	con_32k_ioe 1'b0: Output mode 1'b1: Input mode;

PMU_GRF_SOC_CON1

Address: Operational Base + offset (0x0104)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15	RW	0x0	hold_the_ddrfailsafe hold ddrfailsafe reset
14:0	RW	0x0000	resetn_hold Please refer to cru_softrst_con9. Each bit has a hold

PMU_GRF_SOC_CON2

Address: Operational Base + offset (0x0108)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:13	RO	0x0	reserved
12:0	RW	0x12c0	npor_out2chip_pulse_width Pulse width of triggered NPOR output. Multiplied with XIN_OSC clock period

PMU_GRF_SOC_CON3

Address: Operational Base + offset (0x010c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:9	RO	0x0	reserved
8	RW	0x0	upctl_c_sysreq_cfg 1'b0: After ddr failsafe module enters self-refresh status, then request DDR controller to enter low power state 1'b1: Always enable requesting DDR controller to enter low power state, when ddr failsafe module is working
7	RW	0x0	ddr_io_ret_oen_cfg ddr_io_ret_oen_cfg bit control 1'b0: Ddr_io_ret output enable 1'b1: Ddr_io_ret_output disable
6	RW	0x0	ddr_io_ret_cfg 1'b0: Ddr io retention managed by hardware automatically; 1'b1: Enable ddr io retention manually
5	RW	0x0	ddr_io_ret_de_req Request to enter retention, during system failure 1'b0: Disable 1'b1: Enable
4	RW	0x1	ddrc_gating_en 1'b0: Disable ddr clock gating during system failure 1'b1: Enable ddr clock gating during system failure
3	RW	0x1	sref_enter_en 1'b0: Disable ddr self-refresh enter when system is failed 1'b1: Enable ddr self-refresh enter when system is failed
2	RW	0x1	ddrio_ret_en 1'b0: Remain ddr io status when system is failed 1'b1: Enable ddr io retension when system is failed
1	RW	0x0	wdt_shut_reset_trigger_en Enable failsafe wdt input 1'b0: Disable; 1'b1: Enable;
0	RW	0x0	tsadc_shut_reset_trigger_en Enable failsafe tsadc input 1'b0: Disable; 1'b1: Enable;

PMU_GRF_SOC_CON4

Address: Operational Base + offset (0x0110)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:7	RO	0x0	reserved
6	RW	0x0	pmu_sram_ma_wrasd Write assist function disable
5:4	RW	0x0	pmu_sram_ma_wras Write Assist timing
3:2	RW	0x0	pmu_sram_ma_wl Wordline pulse only
1:0	RW	0x0	pmu_sram_ma_sawl Sense-Amp timing and word line pulse width

PMU_GRF_SOC_STATUS

Address: Operational Base + offset (0x0120)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	grf_pmuvccio2_voltage PMU VCC IO2 voltage detected 1'b0:3.3V 1'b1:1.8V

PMU_GRF_IO_VSEL0

Address: Operational Base + offset (0x0140)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:9	RO	0x0	reserved
8	RW	0x0	grf_vccio0_vsel_src VCC IO0 voltage source select 1'b0: Controlled by IO_FLASHvol_sel_GPIO0B6pmuio2 1'b1: Controlled by grf_vccio0_vsel
7	RW	0x0	grf_vccio7_vsel VCC IO7 voltage select 1'b0:3.3V 1'b1:1.8V
6	RW	0x0	grf_vccio6_vsel VCC IO6 voltage select 1'b0:3.3V 1'b1:1.8V

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5	RW	0x0	grf_vccio5_vsel VCC IO5 voltage select 1'b0:3.3V 1'b1:1.8V
4	RW	0x0	grf_vccio4_vsel VCC IO4 voltage select 1'b0:3.3V 1'b1:1.8V
3	RW	0x0	grf_vccio3_vsel VCC IO3 voltage select 1'b0:3.3V 1'b1:1.8V
2	RW	0x0	grf_vccio2_vsel VCC IO2 voltage select 1'b0:3.3V 1'b1:1.8V
1	RW	0x0	grf_vccio1_vsel VCC IO1 voltage select 1'b0:3.3V 1'b1:1.8V
0	RW	0x0	grf_vccio0_vsel VCC IO0 voltage select 1'b0:3.3V 1'b1:1.8V

PMU_GRF_IO_VSEL1

Address: Operational Base + offset (0x0144)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7	RW	0x0	grf_vccio7_vsel_auto VCC IO7 voltage detect enable 1'b0: Auto-selection independent of grf_io_vsel0 1'b1: Manual selection by grf_io_vsel0
6	RW	0x0	grf_vccio6_vsel_auto VCC IO6 voltage detect enable 1'b0: Auto-selection independent of grf_io_vsel0 1'b1: Manual selection by grf_io_vsel0
5	RW	0x0	grf_vccio5_vsel_auto VCC IO5 voltage detect enable 1'b0: Auto-selection independent of grf_io_vsel0 1'b1: Manual selection by grf_io_vsel0

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4	RW	0x0	grf_vccio4_vsel_auto VCC IO4 voltage detect enable 1'b0: Auto-selection independent of grf_io_vsel0 1'b1: Manual selection by grf_io_vsel0
3	RW	0x0	grf_vccio3_vsel_auto VCC IO3 voltage detect enable 1'b0: Auto-selection independent of grf_io_vsel0 1'b1: Manual selection by grf_io_vsel0
2	RW	0x0	grf_vccio2_vsel_auto VCC IO2 voltage detect enable 1'b0: Auto-selection independent of grf_io_vsel0 1'b1: Manual selection by grf_io_vsel0
1	RW	0x0	grf_vccio1_vsel_auto VCC IO1 voltage detect enable 1'b0: Auto-selection independent of grf_io_vsel0 1'b1: Manual selection by grf_io_vsel0
0	RW	0x0	grf_vccio0_vsel_auto VCC IO0 voltage detect enable 1'b0: Auto-selection independent of grf_io_vsel0 1'b1: Manual selection by grf_io_vsel0

PMU_GRF_IO_VSEL_STATUS

Address: Operational Base + offset (0x0148)

Bit	Attr	Reset Value	Description
31:9	RO	0x0	reserved
8	RW	0x0	grf_vccio2_voltage1 VCC IO2 voltage detected VCC IO2 voltage detected, two voltage detection circuits are used, this is the status bit for second detection circuit, please check the bit2 of PMU_GRF_IO_VSEL_STATUS for another VCC IO2 voltage detection circuit. 1'b0:3.3V 1'b1:1.8V
7	RO	0x0	grf_vccio7_voltage VCC IO7 voltage detected 1'b0:3.3V 1'b1:1.8V
6	RO	0x0	grf_vccio6_voltage VCC IO6 voltage detected 1'b0:3.3V 1'b1:1.8V
5	RO	0x0	grf_vccio5_voltage VCC IO5 voltage detected 1'b0:3.3V 1'b1:1.8V

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4	RO	0x0	grf_vccio4_voltage VCC IO4 voltage detected 1'b0:3.3V 1'b1:1.8V
3	RO	0x0	grf_vccio3_voltage VCC IO3 voltage detected 1'b0:3.3V 1'b1:1.8V
2	RO	0x0	grf_vccio2_voltage0 VCC IO2 voltage detected VCC IO2 voltage detected, two voltage detection circuits are used, this is the status bit for second detection circuit, please check the bit8 of PMU_GRF_IO_VSEL_STATUS for another VCC IO2 voltage detection circuit. 1'b0:3.3V 1'b1:1.8V
1	RO	0x0	grf_vccio1_voltage VCC IO1 voltage detected 1'b0:3.3V 1'b1:1.8V
0	RO	0x0	grf_vccio0_voltage VCC IO0 voltage detected 1'b0:3.3V 1'b1:1.8V

PMU_GRF_PVTM_CON0

Address: Operational Base + offset (0x0180)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:8	RO	0x0	reserved
7:2	RW	0x00	pvtm_clkout_div pvtm_clkout_div Frequence is pvtm_clkout_div*4+1
1	RW	0x1	pvtm_pmu_osc_en Set high to enable the osc_ring in the PVTM
0	RW	0x1	pvtm_pmu_start Set high to start PVTM

PMU_GRF_PVTM_CON1

Address: Operational Base + offset (0x0184)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RW	0x00000100	pvtm_pmu_cal_cnt pvtm calculation counter

PMU_GRF_PVTM_STATUS0

Address: Operational Base + offset (0x0190)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	pvtm_pmu_freq_done High indicates pmu pvtm frequency count done

PMU_GRF_PVTM_STATUS1

Address: Operational Base + offset (0x0194)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pvtm_pmu_freq_cnt Indicates the cycle counts of the osc ring clock

PMU_GRF_OS_REG0

Address: Operational Base + offset (0x0200)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pmu_os_reg0 OS register

PMU_GRF_OS_REG1

Address: Operational Base + offset (0x0204)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pmu_os_reg1 OS register

PMU_GRF_OS_REG2

Address: Operational Base + offset (0x0208)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pmu_os_reg2 OS register

PMU_GRF_OS_REG3

Address: Operational Base + offset (0x020c)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pmu_os_reg3 OS register

PMU_GRF_OS_REG4

Address: Operational Base + offset (0x0210)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pmu_os_reg4 OS register

PMU_GRF_OS_REG5

Address: Operational Base + offset (0x0214)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pmu_os_reg5 OS register

PMU_GRF_OS_REG6

Address: Operational Base + offset (0x0218)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pmu_os_reg6 OS register

PMU_GRF_OS_REG7

Address: Operational Base + offset (0x021c)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pmu_os_reg7 OS register

PMU_GRF_OS_REG8

Address: Operational Base + offset (0x0220)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pmu_os_reg8 OS register, once this reg is wrotten, it can't be reset

PMU_GRF_OS_REG9

Address: Operational Base + offset (0x0224)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pmu_os_reg9 OS register, once this reg is wrotten, it can't be reset

PMU_GRF_OS_REG10

Address: Operational Base + offset (0x0228)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pmu_os_reg10 OS register, once this reg is wrotten, it can't be reset

PMU_GRF_OS_REG11

Address: Operational Base + offset (0x022c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pmu_os_reg11 OS register, once this reg is wrotten, it can't be reset

PMU_GRF_RESET_FUNCTION_STATUS

Address: Operational Base + offset (0x0230)

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3	RW	0x0	ddr_fail_safe_src ddr_fail_safe is active
2	RW	0x0	tsadc_shut_reset_src Reset by tsadc shut trigger
1	RW	0x0	wdt_reset_src Reset by wdt trigger
0	RW	0x0	first_reset_src Reset by first reset trigger

PMU_GRF_RESET_FUNCTION_CLR

Address: Operational Base + offset (0x0234)

Bit	Attr	Reset Value	Description
31:17	RO	0x0	reserved
16	RW	0x0	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:4	RO	0x0	reserved
3	RW	0x0	ddr_fail_safe_src_clr Clear bit for ddr_fail_safe is active 1'b1: Clear enable 1'b0: Clear disable
2	RW	0x0	tsadc_shut_reset_src_clr Clear bit for reset by tsadc shut trigger 1'b1: Clear enable 1'b0: Clear disable
1	RW	0x0	wdt_reset_src_clr Clear bit for reset by wdt trigger 1'b1: Clear enable 1'b0: Clear disable
0	RW	0x0	first_reset_src_clr Clear bit for reset by first reset trigger 1'b1: Clear enable 1'b0: Clear disable

PMU_GRF_SIG_DETECT_CON

Address: Operational Base + offset (0x0380)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:2	RO	0x0	reserved
1	RW	0x0	sdmmc_detectn_neg_irq_msk Enable sdmmc detectn negedge irq 1'b0: Disable 1'b1: Enable
0	RW	0x0	sdmmc_detectn_pos_irq_msk Enable sdmmc detectn posedge irq 1'b0: Disable 1'b1: Enable

PMU_GRF_SIG_DETECT_STATUS

Address: Operational Base + offset (0x0390)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x0	sdmmc_detectn_neg_irq 1'b1: Irq asserted; 1'b0: No irq
0	RW	0x0	sdmmc_detectn_pos_irq 1'b1: Irq asserted; 1'b0: No irq

PMU_GRF_SIG_DETECT_STATUS_CLEAR

Address: Operational Base + offset (0x03a0)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	WO	0x0	sdmmc_detectn_neg_irq_clr 1'b1: Clear irq
0	WO	0x0	sdmmc_detectn_pos_irq_clr 1'b1: Clear irq

PMU_GRF_SDMMC_DET_COUNTER

Address: Operational Base + offset (0x03b0)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x003e8	sdmmc_detectn_count sdmmc_detectn_count bit register

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

4.7 DDR_GRF Register Description

4.7.1 Registers Summary

Name	Offset	Size	Reset Value	Description
DDR_GRF_CON0	0x0000	W	0x00000000	DDR Control Register0
DDR_GRF_CON1	0x0004	W	0x00002600	DDR Control Register1
DDR_GRF_SPLIT_CON	0x0008	W	0x00000110	DDR AXI SPLIT Control Register
DDR_GRF_LP_CON	0x0020	W	0x00001101	DDR PHY Lower Power Control Register
DDR_GRF_STATUS0	0x0100	W	0x00000000	DDR Status Register0
DDR_GRF_STATUS1	0x0104	W	0x00000000	DDR Status Register1
DDR_GRF_STATUS2	0x0108	W	0x00000000	DDR Status Register2
DDR_GRF_STATUS3	0x010c	W	0x00000000	DDR Status Register3
DDR_GRF_STATUS4	0x0110	W	0x00000000	DDR Status Register4
DDR_GRF_STATUS5	0x0114	W	0x00000000	DDR Status Register5
DDR_GRF_STATUS6	0x0118	W	0x00000000	DDR Status Register6
DDR_GRF_STATUS7	0x011c	W	0x00000000	DDR Status Register7
DDR_GRF_STATUS8	0x0120	W	0x00000000	DDR Status Register8
DDR_GRF_STATUS9	0x0124	W	0x00000000	DDR Status Register9

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

4.7.2 Detail Register Description

DDR_GRF_CON0

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RO	0x0	reserved
14	RW	0x0	awpoison AXI write poison control
13	RW	0x0	awurgent AXI write urgent control
12	RW	0x0	arpoison AXI read poison control
11	RW	0x0	arurgent AXI bus read urgent control
10	RW	0x0	pa_wmask When asserted(active high), it will prevent the corresponding write to PA
9:8	RW	0x0	pa_rmask When asserted(active high), it will prevent the corresponding read to PA
7:6	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5	RW	0x0	csysreq_upctl_ddrstdby 1'b0: Disable stdby controls upctl csysreq_ddrc 1'b1: Enable stdby control upctl csysreq_ddrc
4	RW	0x0	csysreq_upctl_pmu 1'b0: Disable pmu controls upctl csysreq_ddrc 1'b1: Enable pmu controls upctl csysreq_ddrc
3	RW	0x0	csysreq_aclk 1'b0: Request upctl aclk enter low power 1'b1: Request upctl aclk exit low power
2	RW	0x0	dfi_init_start dfi_init start value
1	RW	0x0	dfi_init_start_sel 1'b0: Upctl controls dfi_init_start 1'b1: Grf_dfi_init_start controls dfi_init_start
0	RW	0x0	upctl_slverr_enable 1'b0: Disable upctl apb slverr response 1'b1: Enable upcttl apb slverr response

DDR_GRF_CON1

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	con_clkouten_dqcmd_sel ddr phy clkouten_dqcmd source selection 1'b0: From ddr_stdby block 1'b1: From grf regsiter
14	RW	0x0	con_pllpd_dqcmd_sel ddr phy pllpd_dqcmd source selection 1'b0: From ddr_stdby block 1'b1: From grf regsiter
13	RW	0x1	con_clkouten_dqcmd clkouten_dqcmd bit control for DDR PHY, control DDR PHY clock enable 1'b0: Disable 1'b1: Eanable
12	RW	0x0	con_pllpd_dqcmd pllpd_dqcmd bit control for DDR PHY, control DDR PHY PLL power down 1'b0: Disable 1'b1: Eanable
11:8	RW	0x6	auto_sr_dly The delay of auto gated ddrc_core_clk. It should be to be 0x6

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7:6	RO	0x0	reserved
5	RW	0x0	con_upctl2_pdsrlp_cg_en DDR clock gating control after enter PD/SR state 1'b0: Enable clock gating 1'b1: Disable clock gating
4	RW	0x0	upctl_syscreq_cg_en 1'b0: Disable force ddrc_core_clk ungating when external ddrc_csysreq asserted 1'b1: Enable force ddrc_core ungating when external ddrc_csysreq asserted
3	RW	0x0	selfref_type2_en 1'b0: Disable ddrc_core_clk auto gating in type2 selfrefresh 1'b1: Enable ddrc_core_clk auto gating in type2 selfrefresh
2	RW	0x0	upctl_core_cg_en 1'b0: Disable ddrc_core_clk auto gating 1'b1: Enable ddrc_core_clk auto gating
1	RW	0x0	upctl_apb_cg_en 1'b0: Disable function of force aclk/ddrc_core_clk ungated when apb access is going 1'b1: Enable function of force aclk/ddrc_core_clk ungated when apb access is going
0	RW	0x0	upctl_axi_cg_en 1'b0: Disable aclk auto gating 1'b1: Enable aclk auto gating

DDR_GRF_SPLIT_CON

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:11	RO	0x0	reserved
10:9	RW	0x0	SPMODE Split mode select. 2'b00:DDR controller and phy works at 32 bits mode. Low 16 bits are valid if access address is above split address. 2'b01:DDR controller and phy works at 32 bits mode. High 16 bits are valid if access address is above split address. 2'b10:DDR controller and phy works at 16 bits mode. Low 8 bits are valid if access address is above split address. 2'b11:DDR controller and phy works at 16 bits mode. High 8 bits are valid if access address is above split address

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
8	RW	0x1	BYPASS 1'b0: Enable axi split 1'b1: Bypass axi split
7:0	RW	0x10	SPADDR Split address high 8 bits of 32bit address. For example, if SPADDR=0x10, then the split address is 0x10000000. The axi_split module will be bypassed if reading or writing DDR below split address, otherwise axi burst will be split

DDR_GRF_LP_CON

Address: Operational Base + offset (0x0020)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:14	RO	0x0	reserved
13	RW	0x0	sr_ctl_en 1'b0: Disable sr exit/enter reload/inverse lpckdis_ini 1'b1: Enable sr exit/eneter reload/inverse lpckdis_ini
12	RW	0x1	pd_ctl_en 1'b0: Disable pd exit/enter reload/inverse lpckdis_ini 1'b1: Enable pd exit/eneter reload/inverse lpckdis_ini
11:10	RO	0x0	reserved
9	RW	0x0	lpckdis_en 1'b0: Disable ddr phy low power fuction 1'b1: Enable ddr phy low power function
8	RW	0x1	lpckdis_ini lpckdis intial value
7:3	RO	0x0	reserved
2	RW	0x0	lp23_mode 1'b0: Disable LPDDR2/LPDDR3 mode 1'b1: Enable LPDDR2/LPDDR3 mode
1	RO	0x0	reserved
0	RW	0x1	ddr23_mode 1'b0: Disable DDR2/DDR3 mode 1'b1: Enable DDR2/DDR3 mode

DDR_GRF_STATUS0

Address: Operational Base + offset (0x0100)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	mrr_data0[31:0] DDR_STATUS0~DDR_STATUS7 are Mode Register Read Data. mrr_data0[31:0] data status.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

DDR_GRF_STATUS1

Address: Operational Base + offset (0x0104)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	mrr_data0[63:32] mrr_data0[63:32] data status. See DDR_STATUS0

DDR_GRF_STATUS2

Address: Operational Base + offset (0x0108)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	mrr_data0[95:64] mrr_data0[95:64] data status. See DDR_STATUS0

DDR_GRF_STATUS3

Address: Operational Base + offset (0x010c)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	mrr_data0[127:96] mrr_data0[127:96] data status. See DDR_STATUS0

DDR_GRF_STATUS4

Address: Operational Base + offset (0x0110)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	mrr_data1[31:0] mrr_data1[31:0] data status. See DDR_STATUS0

DDR_GRF_STATUS5

Address: Operational Base + offset (0x0114)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	mrr_data1[63:32] mrr_data1[63:32] data status. See DDR_STATUS0

DDR_GRF_STATUS6

Address: Operational Base + offset (0x0118)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	mrr_data1[95:64] mrr_data1[95:64] data status. See DDR_STATUS0

DDR_GRF_STATUS7

Address: Operational Base + offset (0x011c)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	mrr_data1[127:96] mrr_data1[127:96] data status. See DDR_STATUS0

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

DDR_GRF_STATUS8

Address: Operational Base + offset (0x0120)

Bit	Attr	Reset Value	Description
31:19	RO	0x0	reserved
18	RO	0x0	cpu_port_probe_mainStatAlarm The status of cpu_port_probe_mainStatAlarm
17	RO	0x0	cpu_port_probe_mainTraceAlarm The status of cpu_port_probe_mainTraceAlarm
16	RO	0x0	gpu_port_probe_mainStatAlarm The status of gpu_port_probe_mainStatAlarm
15	RO	0x0	gpu_port_probe_mainTraceAlarm The status of gpu_port_probe_mainTraceAlarm
14	RO	0x0	mmip_port_probe_mainStatAlarm The status of mmip_port_probe_mainStatAlarm
13	RO	0x0	mmip_port_probe_mainTraceAlarm The status of mmip_port_probe_mainTraceAlarm
12	RO	0x0	peri_port_probe_mainStatAlarm The status of peri_port_probe_mainStatAlarm
11	RW	0x0	peri_port_probe_mainTraceAlarm The status of peri_port_probe_mainTraceAlarm
10	RW	0x0	dfi_scramble_read_of The status of dfi_scramble_read_of
9	RW	0x0	dfi_scramble_write_of The status of dfi_scramble_write_of
8	RW	0x0	dfi_scramble_key_ready The status of dfi_scramble_key_ready
7:6	RW	0x0	ddrc_reg_selfref_type The status of ddrc_reg_selfref_type
5	RW	0x0	cactive_aclk The status of cactive_aclk
4	RW	0x0	csysaclk_aclk The status of csysaclk_aclk
3	RO	0x0	con_csysreq_aclk The status of con_csysreq_aclk
2	RO	0x0	cactive_ddrc The status of external cactive_ddrc
1	RO	0x0	csysack_ddrc The status of external csysack_ddrc
0	RO	0x0	csysreq_ddrc The status of external csysreq_ddrc

DDR_GRF_STATUS9

Address: Operational Base + offset (0x0124)

Bit	Attr	Reset Value	Description
31	RW	0x0	dfi_lp_ck_disable The status of low power of ddr phy

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
30	RO	0x0	reserved
29:24	RO	0x00	hif_refresh_req_bank The status of hif_refresh_req_bank
23	RO	0x0	reserved
22:16	RO	0x00	wr_credit_cnt The status of wr_credit_cnt
15	RO	0x0	reserved
14:8	RO	0x00	hpr_credit_cnt The status of hpr_credit_cnt
7	RO	0x0	reserved
6:0	RO	0x00	lpr_credit_cnt The status of lpr_credit_cnt

4.8 PCIe_USB_GRF Register Description

4.8.1 Registers Summary

Name	Offset	Size	Reset Value	Description
PCIe_USB_GRF_PCIe_CON0	0x0000	W	0x00000000	PCIe control register0
PCIe_USB_GRF_PCIe_STATUS0	0x0010	W	0x00000000	PCIe status register0
PCIe_USB_GRF_USB3OTG_CON0	0x0430	W	0x00002000	usb3otg control register0
PCIe_USB_GRF_USB3OTG_CON1	0x0434	W	0x00001100	usb3otg control register1
PCIe_USB_GRF_USB3OTG_STATUS0	0x0450	W	0x00000000	usb3otg status register0
PCIe_USB_GRF_USB3OTG_STATUS1	0x0454	W	0x00000000	usb3otg status register1
PCIe_USB_GRF_USB3OTG_STATUS2	0x0458	W	0x00000000	usb3otg status register2
PCIe_USB_GRF_HOST0_CON0	0x0700	W	0x00000820	USB host control register0
PCIe_USB_GRF_HOST0_CON1	0x0704	W	0x000004bc	USB host control register1
PCIe_USB_GRF_HOST0_STATUS	0x0890	W	0x00000000	USB host status register

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

4.8.2 Detail Register Description

PCIe_USB_GRF_PCIe_CON0

Address: Operational Base + offset (0x0000)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:3	RO	0x0	reserved
2	RW	0x0	pcie_link_rst_grt PCIe link rest grant control
1	RW	0x0	pcie_pwr_idle_req PCIe axi niu idle request
0	RO	0x0	reserved

PCIe_USB_GRF_PCIe_STATUS0

Address: Operational Base + offset (0x0010)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RO	0x0	pcie_pwr_idle PCIe axi niu idle status
0	RO	0x0	pcie_pwr_idle_ack PCIe axi niu idle acknowledge status

PCIe_USB_GRF_USB3OTG_CON0

Address: Operational Base + offset (0x0430)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	usb3otg_host_u2_port_disable USB2.0 Port Disable control. 1'b0: Port Enabled 1'b1: Port Disabled When 1, this signal stops reporting connect/disconnect events the port and keeps the port in disabled state
14	RW	0x0	usb3otg_host_port_power_control_present This indicates whether the host controller implementation includes port power control. 1'b0: Indicates that the port does not have port power switches. 1'b1: Indicates that the port has port power switches
13:8	RW	0x20	usb3otg_fladj_30mhz_reg usb3otg_fladj_30mhz_reg bit control

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7:6	RW	0x0	usb3otg_hub_port_perm_attach Indicates if the device attached to a downstream port is permanently attached or not. 1'b0: Not permanently attached 1'b1: Permanently attached Bit0 is for USB2.0 port and bit1 are for USB 3.0 SS port
5:4	RW	0x0	usb3otg_hub_port_overcurrent This is the per port Overcurrent indication of the root-hub ports: 1'b0: No Overcurrent 1'b1: Overcurrent Bit0 is for USB 2.0 port and bit1 are for USB 3.0 SS port
3:0	RW	0x0	usb3otg_bus_filter_bypass It is expected that this signal is set or reset at power-on reset and is not changed during the normal operation of the core. The function of each bit is: bus_filter_bypass[3]: Bypass the filter for utmiotg_iddig bus_filter_bypass[2]: Bypass the filters for utmisrp_bvalid and utmisrp_sessend bus_filter_bypass[1]: Bypass the filter for pipe3_PowerPresent all U3 ports bus_filter_bypass[0]: Bypass the filter for utmiotg_vbusvalid all U2 ports In non-OTG Host-only mode, internal bus filters are not needed. Values: 1'b0: Bus filter(s) enabled 1'b1: Bus filter(s) disabled (bypassed)

PCIe_USB_GRF_USB3OTG_CON1

Address: Operational Base + offset (0x0434)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:12	RW	0x1	usb3otg_host_num_u3_port xHCI usb3 port number, default as 1
11:8	RW	0x1	usb3otg_host_num_u2_port xHCI host USB2 Port number, default as 1
7:6	RO	0x0	reserved
5	RW	0x0	usb3otg_host_legacy_smi_bar Use this register to support SMI on BAR defined in xHCI spec. SW must set this register, then clear this register to indicate Base Address Register written

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4	RW	0x0	usb3otg_host_legacy_smi_pci_cmd Use this register to support SMI on PCI Command defined in xHCI spec. SW must set this register, then clear this register to indicate PCI command register written
3:2	RO	0x0	reserved
1	RW	0x0	usb3otg_pme_en Enable signal for the pme_generation. Enable the core to assert pme_generation
0	RW	0x0	usb3otg_host_u3_port_disable USB 3.0 SS Port Disable control. 1'b0: Port Enabled 1'b1: Port Disabled

PCIe_USB_GRF_USB3OTG_STATUS0

Address: Operational Base + offset (0x0450)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	usb3otg_logic_analyzer_trace[31:0] usb3otg_logic_analyzer_trace[31:0] bit status

PCIe_USB_GRF_USB3OTG_STATUS1

Address: Operational Base + offset (0x0454)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	usb3otg_logic_analyzer_trace[63:32] usb3otg_logic_analyzer_trace[63:32] bit status

PCIe_USB_GRF_USB3OTG_STATUS2

Address: Operational Base + offset (0x0458)

Bit	Attr	Reset Value	Description
31:12	RO	0x0	reserved
11:0	RO	0x000	usb3otg_host_current_belt[11:0] usb3otg_host_current_belt[11:0] bit status

PCIe_USB_GRF_HOST0_CON0

Address: Operational Base + offset (0x0700)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:12	RO	0x0	reserved
11:6	RW	0x20	host0_fladj_val_common USB HOST0 fladj_val_common bit control
5:0	RW	0x20	host0_fladj_val USB HOST0 fladj bit control

PCIe_USB_GRF_HOST0_CON1

Address: Operational Base + offset (0x0704)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0:Write access disable 1'b1: Write access enable
15:14	RO	0x0	reserved
13	RW	0x0	host0_arb_pause host0 ehci/ohci arbiter pause control
12	RW	0x0	host0_ohci_susp_lgcy USB HOST0 ohci_susp_lgcy bit control
11	RW	0x0	host0_ohci_cntsel USB HOST0 ohci_cntsel bit control
10	RW	0x1	host0_ohci_clkcktrst USB HOST0 ohci_clkcktrst bit control
9	RW	0x0	host0_app_prt_ovrcur USB HOST0 app_prt_ovrcur bit control
8	RW	0x0	host0_autoppd_on_overcur_en USB HOST0 autoppd_on_overcur_en bit control
7	RW	0x1	host0_word_if USB HOST0 word_if bit control
6	RW	0x0	host0_sim_mode USB HOST0 sim_mode bit control 1'b0: Disable 1'b1: Enable
5	RW	0x1	host0_incrx_en USB HOST0 incrx_en bit control 1'b0: Disable 1'b1: Enable
4	RW	0x1	host0_incr8_en USB HOST0 incr8_en bit control 1'b0: Disable 1'b1: Enable
3	RW	0x1	host0_incr4_en USB HOST0 incr4_en bit control 1'b0: Disable 1'b1: Enable
2	RW	0x1	host0_incr16_en USB HOST0 incr16_en bit control 1'b0: Disable 1'b1: Enable
1	RW	0x0	host0_hubsetup_min USB HOST0 bubsetup_min bit control
0	RW	0x0	host0_app_start_clk USB HOST0 app_start_clk bit control

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

PCIe_USB_GRF_HOST0_STATUS

Address: Operational Base + offset (0x0890)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30	RO	0x0	host0_ehci_power_state_ack host0_ehci_power_state_ack bit status
29	RO	0x0	host0_ehci_pme_status host0_ehci_pme_status bit status
28	RO	0x0	host0_ehci_bufacc host0_ehci_bufacc bit status
27	RO	0x0	host0_ehci_xfer_prdc host0_ehci_xfer_prdc bit status
26	RO	0x0	host0_ohci_ccs host0_ohci_ccs bit status
25	RO	0x0	host0_ohci_rwe host0_ohci_rwe bit status
24	RO	0x0	host0_ohci_drwe host0_ohci_drwe bit status
23	RO	0x0	host0_ohci_globalsuspend host0_ohci_globalsuspend bit status
22	RO	0x0	host0_ohci_bufacc host0_ohci_bufacc bit status
21	RO	0x0	host0_ohci_rmtwkp host0_ohci_rmtwkp bit status
20:17	RO	0x0	host0_ehci_lpsmc_state host0_ehci_lpsmc_state bit status
16:11	RO	0x00	host0_ehci_usbsts host0_ehci_usbsts bit status
10:0	RW	0x000	host0_ehci_xfer_cnt host0_ehci_xfer_cnt bit status

4.9 CORE_GRF Register Description

4.9.1 Registers Summary

Name	Offset	Size	Reset Value	Description
CORE_GRF_CPU_PEFF_CON0	0x0000	W	0x00000000	CPU performance monitor control register0
CORE_GRF_CPU_PEFF_CON1	0x0004	W	0x00000000	CPU performance monitor control register1
CORE_GRF_CPU_PEFF_CON2	0x0008	W	0x00000000	CPU performance monitor control register2
CORE_GRF_CPU_PEFF_CON3	0x000c	W	0x00000000	CPU performance monitor control register3

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
CORE_GRF_CPU_PEFF_CON4	0x0010	W	0x00000000	CPU performance monitor control register4
CORE_GRF_CPU_PEFF_CON5	0x0014	W	0x00000000	CPU performance monitor control register5
CORE_GRF_CPU_PEFF_CON6	0x0018	W	0x00000000	CPU performance monitor control register6
CORE_GRF_CPU_PEFF_CON7	0x001c	W	0x00000000	CPU performance monitor control register7
CORE_GRF_CPU_PEFF_CON8	0x0020	W	0x00000000	CPU performance monitor control register8
CORE_GRF_GIC_BASE_CON	0x0024	W	0x00000000	GIC base address
CORE_GRF_CPU_PERF_RD_MAX_LATENCY_NUM	0x0030	W	0x00000000	CPU performance monitor status register
CORE_GRF_CPU_PERF_RD_LATENCY_SAMP_NUM	0x0034	W	0x00000000	CPU performance monitor status register
CORE_GRF_CPU_PERF_RD_LATENCY_ACC_NUM	0x0038	W	0x00000000	CPU performance monitor status register
CORE_GRF_CPU_PERF_RD_AXI_TOTAL_BYTE	0x003c	W	0x00000000	CPU performance monitor status register
CORE_GRF_CPU_PERF_WR_AXI_TOTAL_BYTE	0x0040	W	0x00000000	CPU performance monitor status register
CORE_GRF_CPU_PERF_WORKING_CNT	0x0044	W	0x00000000	CPU performance monitor status register
CORE_GRF_CPU_PERF_INT_STATUS	0x0048	W	0x00000000	CPU performance monitor status register
CORE_GRF_COREPVTM_CON0	0x0080	W	0x00000000	CORE PVTM control register0
CORE_GRF_COREPVTM_CON1	0x0084	W	0x00000000	CORE PVTM control register1
CORE_GRF_COREPVTM_STATUS0	0x0088	W	0x00000000	CORE PVTM status register0
CORE_GRF_COREPVTM_STATUS1	0x008c	W	0x00000000	CORE PVTM status register1

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

4.9.2 Detail Register Description

CORE_GRF_CPU_PEFF_CON0

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15	RW	0x0	cpu_sw_rd_latency_id_range_e Axi read channel id for latency axi_performance test
14	RO	0x0	reserved
13:8	RW	0x00	cpu_sw_rd_latency_id 2'h0: 16-Byte align 2'h1: 32-Byte align 2'h2: 64-Byte align 2'h3: 128-Byte align
7:6	RW	0x0	cpu_sw_ddr_align_type axi_perf counter id control 1'b0: Count all write channel id 1'b1: Count sw_ar_count_id write channel only
5	RW	0x0	cpu_sw_aw_cnt_id_type axi_perf counter id control 1'b0: Count all write channel id 1'b1: Count sw_aw_count_id read channel only
4	RW	0x0	cpu_sw_ar_cnt_id_type axi_perf counter id control 1'b0: Count all read channel id 1'b1: Count sw_ar_count_id read channel only
3	RW	0x0	cpu_sw_axi_cnt_type_wrap axi_perf counter type wrap 1'b0: No wrap test 1'b1: Wrap test
2	RW	0x0	cpu_sw_axi_cnt_type axi_perf counter type 1'b0: axi transfer test 1'b1: ddr align transfer test
1	RW	0x0	cpu_sw_axi_perf_clr axi_perf clear bit 1'b0: Disable 1'b1: Enable
0	RW	0x0	cpu_sw_axi_perf_work axi_perf enable bit 1'b0: Disable 1'b1: Enable

CORE_GRF_CPU_PEFF_CON1

Address: Operational Base + offset (0x0004)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RO	0x0	reserved
11:0	RW	0x000	cpu_sw_rd_latency_thr Axi read channel id for latency axi_performance test

CORE_GRF_CPU_PEFF_CON2

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	cpu_sw_axi_perf_int_clr interrupt clear 1'b0: No op 1'b1: Clear
14	RW	0x0	cpu_sw_axi_perf_int_e interrupt enable 1'b0: Disable 1'b1: Enable
13	RO	0x0	reserved
12:8	RW	0x00	cpu_sw_aw_count_id axi write channel ID to be counted
7:6	RO	0x0	reserved
5:0	RW	0x00	cpu_sw_ar_count_id axi read channel ID to be counted

CORE_GRF_CPU_PEFF_CON3

Address: Operational Base + offset (0x000c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:10	RW	0x00	cpu_sw_ar_mon_id the axi read ID to be monitored which address in the range between sw_araddr_st and sw_araddr_end
9:4	RW	0x00	cpu_sw_ar_mon_id_bmsk read channel mask bit control when sw_ar_mon_id_type=1
3:2	RO	0x0	reserved
1	RW	0x0	cpu_sw_ar_mon_id_type mon_id_type bit control 1'b0: Monitor all IDs in axi read channel 1'b1: Only monitor the ID defined in cpu_sw_ar_mon_id
0	RW	0x0	cpu_sw_ar_mon_id_msk AXI read channel monitor interrupt mask control 1'b0: Don't mask interrupt 1'b1: Mask interrupt

CORE_GRF_CPU_PEFF_CON4

Address: Operational Base + offset (0x0010)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:12	RO	0x0	reserved
11:7	RW	0x00	cpu_sw_aw_mon_id the axi read ID to be monitored which address in the range between sw_araddr_st and sw_araddr_end
6:2	RW	0x00	cpu_sw_aw_mon_id_bmsk write channel mask bit control when sw_ar_mon_id_type=1
1	RW	0x0	cpu_sw_aw_mon_id_type mon_id_type bit control 1'b0: Monitor all IDs in axi read channel 1'b1: Only monitor the ID defined in cpu_sw_ar_mon_id
0	RW	0x0	cpu_sw_aw_mon_id_msk AXI write channel monitor interrupt mask control 1'b0: Don't mask interrupt 1'b1: Mask interrupt

CORE_GRF_CPU_PEFF_CON5

Address: Operational Base + offset (0x0014)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	cpu_sw_araddr_mon_st monitor read start address

CORE_GRF_CPU_PEFF_CON6

Address: Operational Base + offset (0x0018)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	cpu_sw_araddr_mon_end monitor read end address

CORE_GRF_CPU_PEFF_CON7

Address: Operational Base + offset (0x001c)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	cpu_sw_awaddr_mon_st monitor write start address

CORE_GRF_CPU_PEFF_CON8

Address: Operational Base + offset (0x0020)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	cpu_sw_awaddr_mon_end monitor write end address

CORE_GRF_GIC_BASE_CON

Address: Operational Base + offset (0x0024)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RW	0x0000	gic_base_addr GIC interface base address of CPU

CORE_GRF_CPU_PERF_RD_MAX_LATENCY_NUM

Address: Operational Base + offset (0x0030)

Bit	Attr	Reset Value	Description
31:13	RO	0x0	reserved
12:0	RW	0x0000	rd_max_latency_r axi read max latency output

CORE_GRF_CPU_PERF_RD_LATENCY_SAMP_NUM

Address: Operational Base + offset (0x0034)

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved
26:0	RO	0x0000000	rd_latency_samp_r AXI read latency total sample number

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

CORE_GRF_CPU_PERF_RD_LATENCY_ACC_NUM

Address: Operational Base + offset (0x0038)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	rd_latency_acc_cnt_r AXI read latency (>sw_rd_latency_thr) total number

CORE_GRF_CPU_PERF_RD_AXI_TOTAL_BYTE

Address: Operational Base + offset (0x003c)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	rd_axi_total_byte AXI active total read bytes/ddr align read bytes

CORE_GRF_CPU_PERF_WR_AXI_TOTAL_BYTE

Address: Operational Base + offset (0x0040)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	wr_axi_total_byte AXI active total write bytes/ddr align write bytes

CORE_GRF_CPU_PERF_WORKING_CNT

Address: Operational Base + offset (0x0044)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	working_cnt_r working counter

CORE_GRF_CPU_PERF_INT_STATUS

Address: Operational Base + offset (0x0048)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:24	RO	0x00	a35_aw_mon_axi_id_status The ID be monitored read from the specific addr area
23:17	RO	0x0	reserved
16	RO	0x0	a35_aw_mon_axi_hit_flag Write from the specific addr area interrupt status
15	RO	0x0	reserved
14:8	RO	0x00	a35_ar_mon_axi_id_status The ID be monitored read from the specific addr area
7:1	RO	0x0	reserved
0	RO	0x0	a35_ar_mon_axi_hit_flag Read from the specific addr area interrupt status

CORE_GRF_COREPVTM_CON0

Address: Operational Base + offset (0x0080)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Write enable for lower 16bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:5	RO	0x0	reserved
4:2	RW	0x0	corepvtm_osc_sel osc_ring selection. 3'b000: Osc_ring 0 3'b001: Osc_ring 1 3'b010: Osc_ring 2 3'b011: Osc_ring 3 3'b100: Osc_ring 4 Others: Reserved
1	RW	0x0	corepvtm_osc_en Set high to enable the osc_ring in the PVTM
0	RW	0x0	corepvtm_start Set high to start PVTM

CORE_GRF_COREPVTM_CON1

Address: Operational Base + offset (0x0084)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	corepvtm_cal_cnt pvtm calculation counter

CORE_GRF_COREPVTM_STATUS0

Address: Operational Base + offset (0x0088)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	corepvtm_freq_done High indicates pmu pvtm frequency count done

CORE_GRF_COREPVTM_STATUS1

Address: Operational Base + offset (0x008c)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	corepvtm_freq_cnt Indicates the cycle counts of the osc ring clock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 5 Cortex-A35

5.1 Overview

The RK1808 has a dual-core Cortex-A35 cluster with 128K L2 memory. Cortex-A35 processor, which is a mid-range, low-power processor that implements the ARMv8-A architecture.

The Cortex-A35 processor includes following features:

- Full implementation of the ARMv8-A A64, A32, and T32 instruction sets
- Both the AArch32 and AArch64 execution states at all Exception levels (EL0 to EL3)
- In-order pipeline with direct and indirect branch prediction
- Separate Level 1 (L1) data and instruction side memory systems with a Memory Management Unit(MMU)
- Level 2 (L2) memory system that provides cluster memory coherency
- L2 cache
- TrustZone
- Support data engine that implements the Advanced SIMD and floating-point architecture support
- Support Cryptographic Extension
- ARMv8 debug logic
- Support Generic Interrupt Controller (GIC) CPU interface to connect to an external distributor
- Generic Timers supporting 64-bit count input from an external system counter

The configuration details are shown in following tables

Table 5-1 CPU Configuration

Number of CPU	2
L1 I cache size	32K
L1 D cache size	32K
L2 cache size	128K
L2 data RAM output latency	3 cycles
L2 data RAM input latency	2 cycles
CPU cache protection	No
SCU L2 cache protection	No
BUS master interface	AXI4
NEON and floating point support	Yes
Cryptography extension	Yes

5.2 Block Diagram

The Cortex-A35 subsystem is shown in Figure 1-1. As illustrated, dual-core Cortex-A35 connects to system bus through SCU-L2 which can handle with CDC(clock domain crossing) issue.

The Cortex-A35 is connected with system counter, which can run under a constant frequency clock, for PPI interrupt generation.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

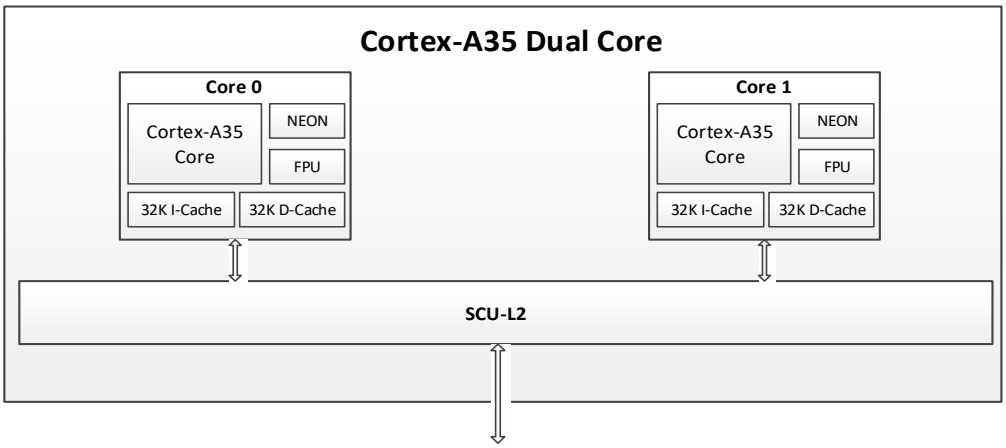


Fig. 5-1 Block Diagram

5.3 Function Description

Please refer to the document cortex_a35_r0p2_trm.pdf for the detail function description.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 6 Direct Memory Access Controller (DMAC)

6.1 Overview

This device supports 1 Direct Memory Access (DMA) Controller. It (DMAC) supports transfers between memory and memory, peripheral and memory. DMAC is under Non-secure state after reset, and the secure state can be changed by configuring SGRF module. DMAC supports the following features:

- Support Trust-Zone technology
- Support 31 peripheral request
- Support 64-bit AXI bus width
- 8 channel at the same time
- Support burst length up to 16
- Support 16 interrupts and 1 abort interrupt
- Supports 128 MFIFO depth

Following table shows the DMAC request mapping scheme.

Table 6-1 DMAC Request Mapping Table

Req number	Source	Polarity
0	UART0_TX	High level
1	UART0_RX	High level
2	UART1_TX	High level
3	UART1_RX	High level
4	UART2_TX	High level
5	UART2_RX	High level
6	UART3_TX	High level
7	UART3_RX	High level
8	UART4_TX	High level
9	UART4_RX	High level
10	SPI0_TX	High level
11	SPI0_RX	High level
12	SPI1_TX	High level
13	SPI1_RX	High level
14	SPI2_TX	High level
15	SPI2_RX	High level
16	I2S0_8CH_TX	High level
17	I2S0_8CH_RX	High level
18	I2S1_2CH_TX	High level
19	I2S1_2CH_RX	High level
20	I2S2_2CH_TX	High level
21	PWM0_TX	High level
22	PWM1_TX	High level
23	PWM2_TX	High level
24	PDM	High level
25	UART5_TX	High level
26	UART5_RX	High level
27	UART6_TX	High level
28	UART6_RX	High level
29	UART7_TX	High level
30	UART7_RX	High level

DMAC support incrementing-address burst and fixed-address burst. But in the case of access SPI and UART at byte or half-word size, DMAC only support fixed-address burst and the address must be aligned to word.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

6.2 Block Diagram

Following figure shows the block diagram of DMAC.

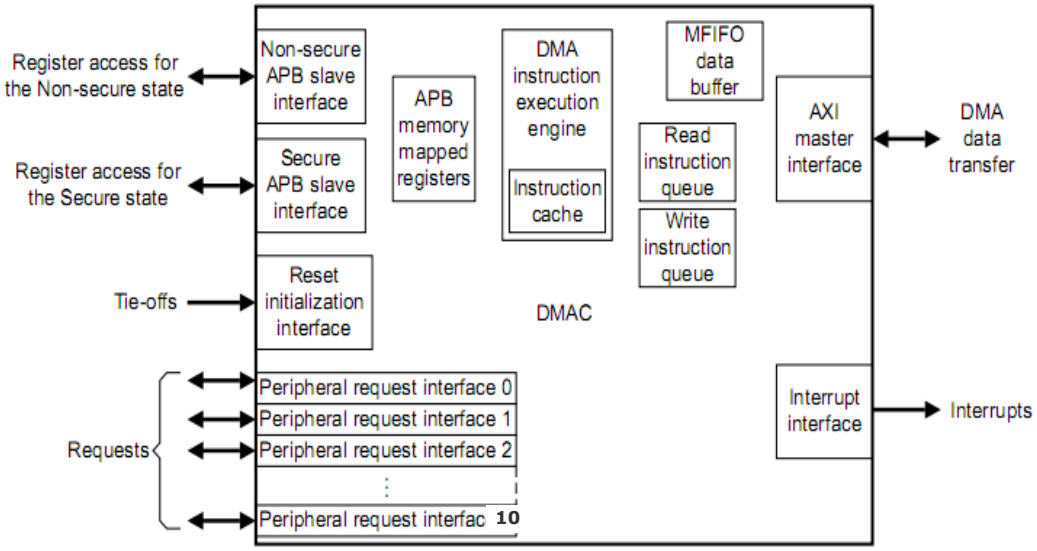


Fig. 6-1 Block diagram of DMAC

As the DMAC supports Trust-Zone technology, so dual APB interfaces enable the operation of the DMAC to be partitioned into the secure state and non-secure state. You can use the APB interfaces to access status registers and also directly execute instructions in the DMAC. The default interface after reset is non-secure APB interface.

6.3 Function Description

6.3.1 Introduction

The DMAC contains an instruction processing block that enables it to process program code that controls a DMA transfer. The program code is stored in a region of system memory that the DMAC accesses using its AXI interface. The DMAC stores instructions temporarily in a cache. It supports 8 channels, each channel capable of supporting a single concurrent thread of DMA operation. In addition, a single DMA manager thread exists, and you can use it to initialize the DMA channel threads. The DMAC executes up to one instruction for each AXI clock cycle. To ensure that it regularly executes each active thread, it alternates by processing the DMA manager thread and then a DMA channel thread. It uses a round-robin process when selecting the next active DMA channel thread to execute.

The DMAC uses variable-length instructions that consist of one to six bytes. It provides a separate Program Counter (PC) register for each DMA channel. When a thread requests an instruction from an address, the cache performs a look-up. If a cache hit occurs, then the cache immediately provides the data. Otherwise, the thread is stalled while the DMAC uses the AXI interface to perform a cache line fill. If an instruction is greater than 4 bytes, or spans the end of a cache line, the DMAC performs multiple cache accesses to fetch the instruction.

When a cache line fill is in progress, the DMAC enables other threads to access the cache, but if another cache miss occurs, this stalls the pipeline until the first line fill is complete. When a DMA channel thread executes a load or store instruction, the DMAC adds the instruction to the relevant read or write queue. The DMAC uses these queues as an instruction storage buffer prior to it issuing the instructions on the AXI bus. The DMAC also contains a Multi First-In-First-Out (MFIFO) data buffer that it uses to store data that it reads, or writes, during a DMA transfer.

6.3.2 Operating states

Following figure shows the operating states for the DMA manager thread and DMA channel threads.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

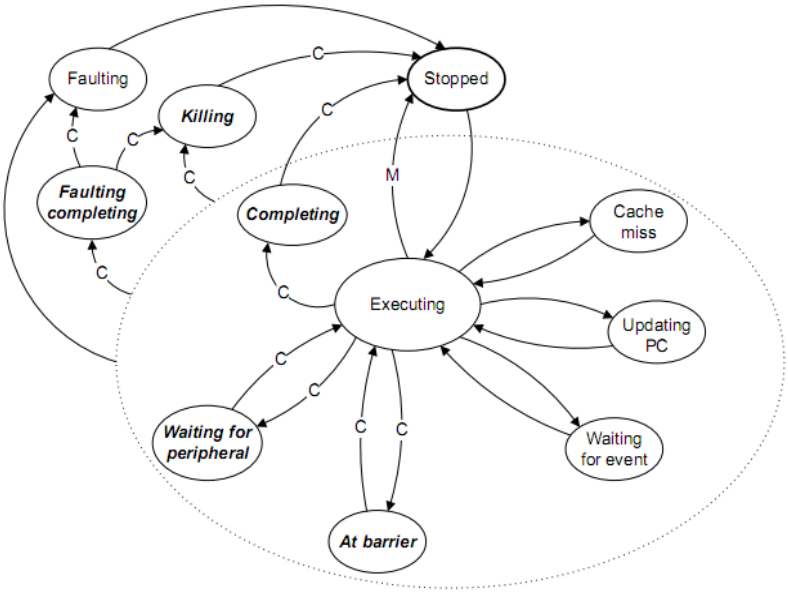


Fig. 6-2 DMAC operation states

Notes: arcs with no letter designator indicate state transitions for the DMA manager and DMA channel threads, otherwise use is restricted as follows:
C DMA channel threads only.
M DMA manager thread only.

After the DMAC exits from reset, it sets all DMA channel threads to the stopped state, and DMA manager thread moves to the Stopped state.

6.4 Register Description

6.4.1 Internal Address Mapping

Slave address can be divided into different length for different usage, which is shown as follows.

6.4.2 Registers Summary

Name	Offset	Size	Reset Value	Description
DMA_DSR	0x0000	W	0x00000000	DMA Manager Status Register
DMA_DPC	0x0004	W	0x00000000	DMA Program Counter Register
DMA_INTEN	0x0020	W	0x00000000	Interrupt Enable Register
DMA_EVENT_RIS	0x0024	W	0x00000000	Event-Interrupt Raw Status Register
DMA_INTMIS	0x0028	W	0x00000000	Interrupt Status Register
DMA_INTCLR	0x002c	W	0x00000000	Interrupt Clear Register
DMA_FSRD	0x0030	W	0x00000000	Fault Status DMA Manager Register
DMA_FSRC	0x0034	W	0x00000000	Fault Status DMA Channel Register
DMA_FTRD	0x0038	W	0x00000000	Fault Type DMA Manager Register
DMA_FTR0	0x0040	W	0x00000000	Fault Type DMA Channel Register
DMA_FTR1	0x0044	W	0x00000000	Fault Type DMA Channel Register
DMA_FTR2	0x0048	W	0x00000000	Fault Type DMA Channel Register

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
DMA_FTR3	0x004c	W	0x00000000	Fault Type DMA Channel Register
DMA_FTR4	0x0050	W	0x00000000	Fault Type DMA Channel Register
DMA_FTR5	0x0054	W	0x00000000	Fault Type DMA Channel Register
DMA_CSR0	0x0100	W	0x00000000	Channel Status Registers
DMA_CPC0	0x0104	W	0x00000000	Channel Program Counter Registers
DMA_CSR1	0x0108	W	0x00000000	Channel Status Registers
DMA_CPC1	0x010c	W	0x00000000	Channel Program Counter Registers
DMA_CSR2	0x0110	W	0x00000000	Channel Status Registers
DMA_CPC2	0x0114	W	0x00000000	Channel Program Counter Registers
DMA_CSR3	0x0118	W	0x00000000	Channel Status Registers
DMA_CPC3	0x011c	W	0x00000000	Channel Program Counter Registers
DMA_CSR4	0x0120	W	0x00000000	Channel Status Registers
DMA_CPC4	0x0124	W	0x00000000	Channel Program Counter Registers
DMA_CSR5	0x0128	W	0x00000000	Channel Status Registers
DMA_CPC5	0x012c	W	0x00000000	Channel Program Counter Registers
DMA_SAR0	0x0400	W	0x00000000	Source Address Registers
DMA_DAR0	0x0404	W	0x00000000	Destination Address Registers
DMA_CCR0	0x0408	W	0x00000000	Channel Control Registers
DMA_LC0_0	0x040c	W	0x00000000	Loop Counter 0 Registers
DMA_LC1_0	0x0410	W	0x00000000	Loop Counter 1 Registers
DMA_SAR1	0x0420	W	0x00000000	Source Address Registers
DMA_DAR1	0x0424	W	0x00000000	Destination Address Registers
DMA_CCR1	0x0428	W	0x00000000	Channel Control Registers
DMA_LC0_1	0x042c	W	0x00000000	Loop Counter 0 Registers
DMA_LC1_1	0x0430	W	0x00000000	Loop Counter 1 Registers
DMA_SAR2	0x0440	W	0x00000000	Source Address Registers
DMA_DAR2	0x0444	W	0x00000000	Destination Address Registers
DMA_CCR2	0x0448	W	0x00000000	Channel Control Registers
DMA_LC0_2	0x044c	W	0x00000000	Loop Counter 0 Registers
DMA_LC1_2	0x0450	W	0x00000000	Loop Counter 1 Registers
DMA_SAR3	0x0460	W	0x00000000	Source Address Registers
DMA_DAR3	0x0464	W	0x00000000	Destination Address Registers
DMA_CCR3	0x0468	W	0x00000000	Channel Control Registers
DMA_LC0_3	0x046c	W	0x00000000	Loop Counter 0 Registers
DMA_LC1_3	0x0470	W	0x00000000	Loop Counter 1 Registers
DMA_SAR4	0x0480	W	0x00000000	Source Address Registers
DMA_DAR4	0x0484	W	0x00000000	Destination Address Registers

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
DMA_CCR4	0x0488	W	0x00000000	Channel Control Registers
DMA_LC0_4	0x048c	W	0x00000000	Loop Counter 0 Registers
DMA_LC1_4	0x0490	W	0x00000000	Loop Counter 1 Registers
DMA_SAR5	0x04a0	W	0x00000000	Source Address Registers
DMA_DAR5	0x04a4	W	0x00000000	Destination Address Registers
DMA_CCR5	0x04a8	W	0x00000000	Channel Control Registers
DMA_LC0_5	0x04ac	W	0x00000000	Loop Counter 0 Registers
DMA_LC1_5	0x04b0	W	0x00000000	Register0000 Description
DMA_DBGSTATUS	0x0d00	W	0x00000000	Debug Status Register
DMA_DBGCMD	0x0d04	W	0x00000000	Debug Command Register
DMA_DBGINST0	0x0d08	W	0x00000000	Debug Instruction-0 Register
DMA_DBGINST1	0x0d0c	W	0x00000000	Debug Instruction-1 Register
DMA_CR0	0x0e00	W	0x00047051	Configuration Register 0
DMA_CR1	0x0e04	W	0x00000057	Configuration Register 1
DMA_CR2	0x0e08	W	0x00000000	Configuration Register 2
DMA_CR3	0x0e0c	W	0x00000000	Configuration Register 3
DMA_CR4	0x0e10	W	0x00000006	Configuration Register 4
DMA_CRDn	0x0e14	W	0x02094733	Configuration Register n
DMA_WD	0x0e80	W	0x00000000	DMA Watchdog Register

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

6.4.3 Detail Register Description

DMA_DSR

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:10	RO	0x0	Reserved
9	RO	0x0	0 = DMA manager operates in the Secure state 1 = DMA manager operates in the Non-secure state
8:4	RO	0x00	b00000 = event[0] b00001 = event[1] b00010 = event[2] … b11111 = event[31]
3:0	RO	0x0	b0000 = Stopped b0001 = Executing b0010 = Cache miss b0011 = Updating PC b0100 = Waiting for event b0101-b1110 = Reserved b1111 = Faulting

DMA_DPC

Address: Operational Base + offset (0x0004)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Program counter for the DMA manager thread

DMA_INTEN

Address: Operational Base + offset (0x0020)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	Bit [N] = 0 If the DMAC executes DMASEV for the event-interrupt resource N then the DMAC signals event N to all of the threads. Set bit [N] to 0 if your system design does not use irq[N] to signal an interrupt request. Bit [N] = 1 If the DMAC executes DMASEV for the event-interrupt resource N then the DMAC sets irq[N] HIGH. Set bit [N] to 1 if your system designer requires irq[N] to signal an interrupt request

DMA_EVENT_RIS

Address: Operational Base + offset (0x0024)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Bit [N] = 0 Event N is inactive or irq[N] is LOW. Bit [N] = 1 Event N is active or irq[N] is HIGH

DMA_INTMIS

Address: Operational Base + offset (0x0028)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Bit [N] = 0 Interrupt N is inactive and therefore irq[N] is LOW. Bit [N] = 1 Interrupt N is active and therefore irq[N] is HIGH

DMA_INTCLR

Address: Operational Base + offset (0x002c)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	Bit [N] = 0 The status of irq[N] does not change. Bit [N] = 1 The DMAC sets irq[N] LOW if the INTEN Register programs the DMAC to signal an interrupt. Otherwise, the status of irq[N] does not change

DMA_FSRD

Address: Operational Base + offset (0x0030)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	0 = the DMA manager thread is not in the Faulting state 1 = the DMA manager thread is in the Faulting state

DMA_FSRC

Address: Operational Base + offset (0x0034)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Bit [N] = 0 No fault is present on DMA channel N. Bit [N] = 1 DMA channel N is in the Faulting or Faulting completing state

DMA_FTRD

Address: Operational Base + offset (0x0038)

Bit	Attr	Reset Value	Description
31	RO	0x0	Reserved
30	RO	0x0	memory or from the debug interface: 0 = instruction that generated an abort was read from system memory 1 = instruction that generated an abort was read from the debug interface
29:17	RO	0x0	Reserved
16	RO	0x0	performs an instruction fetch: 0 = OKAY response 1 = EXOKAY, SLVERR, or DECERR response
15:6	RO	0x0	Reserved
5	RO	0x0	0 = DMA manager has appropriate security to execute DMAWFE or DMASEV 1 = a DMA manager thread in the Non-secure state attempted to execute either: o DMAWFE to wait for a secure event o DMASEV to create a secure event or secure interrupt
4	RO	0x0	0 = DMA manager has appropriate security to execute DMAGO 1 = a DMA manager thread in the Non-secure state attempted to execute DMAGO to create a DMA channel operating in the Secure state
3:2	RO	0x0	Reserved
1	RO	0x0	the configuration of the DMAC: 0 = valid operand 1 = invalid operand
0	RW	0x0	0 = defined instruction 1 = undefined instruction

DMA_FTR0

Address: Operational Base + offset (0x0040)

Bit	Attr	Reset Value	Description
31	RO	0x0	0 = DMA channel has adequate resources 1 = DMA channel has locked-up because of insufficient resources. This fault is an imprecise abort

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
30	RO	0x0	memory or from the debug interface: 0 = instruction that generated an abort was read from system memory 1 = instruction that generated an abort was read from the debug interface. This fault is an imprecise abort but the bit is only valid when a precise abort occurs
29:19	RO	0x0	Reserved
18	RO	0x0	thread performs a data read: 0 = OKAY response 1 = EXOKAY, SLVERR, or DECERR response. This fault is an imprecise abort
17	RO	0x0	thread performs a data write: 0 = OKAY response 1 = EXOKAY, SLVERR, or DECERR response. This fault is an imprecise abort
16	RO	0x0	thread performs an instruction fetch: 0 = OKAY response 1 = EXOKAY, SLVERR, or DECERR response. This fault is a precise abort
15:14	RO	0x0	Reserved
13	RO	0x0	0 = MFIFO contains all the data to enable the DMAST to complete 1 = previous DMALDs have not put enough data in the MFIFO to enable the DMAST to complete. This fault is a precise abort
12	RO	0x0	DMALD 0 = MFIFO contains sufficient space 1 = MFIFO is too small to hold the data that DMALD requires. DMAST 0 = MFIFO contains sufficient data 1 = MFIFO is too small to store the data to enable DMAST to complete. This fault is an imprecise abort
11:8	RO	0x0	Reserved
7	RO	0x0	to perform a secure read or secure write: 0 = a DMA channel thread in the Non-secure state is not violating the security permissions 1 = a DMA channel thread in the Non-secure state attempted to perform a secure read or secure write. This fault is a precise abort

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
6	RO	0x0	DMASTP, or DMAFLUSHP with inappropriate security permissions: 0 = a DMA channel thread in the Non-secure state is not violating the security permissions 1 = a DMA channel thread in the Non-secure state attempted to execute either: o DMAWFP to wait for a secure peripheral o DMALDP or DMASTP to notify a secure peripheral o DMAFLUSHP to flush a secure peripheral. This fault is a precise abort
5	RO	0x0	0 = a DMA channel thread in the Non-secure state is not violating the security permissions 1 = a DMA channel thread in the Non-secure state attempted to execute either: o DMAWFE to wait for a secure event o DMASEV to create a secure event or secure interrupt. This fault is a precise abort
4:2	RO	0x0	Reserved
1	RO	0x0	valid for the configuration of the DMAC: 0 = valid operand 1 = invalid operand. This fault is a precise abort
0	RO	0x0	0 = defined instruction 1 = undefined instruction. This fault is a precise abort

DMA_FTR1

Address: Operational Base + offset (0x0044)

Bit	Attr	Reset Value	Description
31	RO	0x0	0 = DMA channel has adequate resources 1 = DMA channel has locked-up because of insufficient resources. This fault is an imprecise abort
30	RO	0x0	memory or from the debug interface: 0 = instruction that generated an abort was read from system memory 1 = instruction that generated an abort was read from the debug interface. This fault is an imprecise abort but the bit is only valid when a precise abort occurs
29:19	RO	0x0	Reserved
18	RO	0x0	thread performs a data read: 0 = OKAY response 1 = EXOKAY, SLVERR, or DECERR response. This fault is an imprecise abort

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
17	RO	0x0	thread performs a data write: 0 = OKAY response 1 = EXOKAY, SLVERR, or DECERR response. This fault is an imprecise abort
16	RO	0x0	thread performs an instruction fetch: 0 = OKAY response 1 = EXOKAY, SLVERR, or DECERR response. This fault is a precise abort
15:14	RO	0x0	Reserved
13	RO	0x0	0 = MFIFO contains all the data to enable the DMAST to complete 1 = previous DMALDs have not put enough data in the MFIFO to enable the DMAST to complete. This fault is a precise abort
12	RO	0x0	DMALD 0 = MFIFO contains sufficient space 1 = MFIFO is too small to hold the data that DMALD requires. DMAST 0 = MFIFO contains sufficient data 1 = MFIFO is too small to store the data to enable DMAST to complete. This fault is an imprecise abort
11:8	RO	0x0	Reserved
7	RO	0x0	to perform a secure read or secure write: 0 = a DMA channel thread in the Non-secure state is not violating the security permissions 1 = a DMA channel thread in the Non-secure state attempted to perform a secure read or secure write. This fault is a precise abort
6	RO	0x0	DMASTP, or DMAFLUSHP with inappropriate security permissions: 0 = a DMA channel thread in the Non-secure state is not violating the security permissions 1 = a DMA channel thread in the Non-secure state attempted to execute either: o DMAWFP to wait for a secure peripheral o DMALDP or DMASTP to notify a secure peripheral o DMAFLUSHP to flush a secure peripheral. This fault is a precise abort
5	RO	0x0	0 = a DMA channel thread in the Non-secure state is not violating the security permissions 1 = a DMA channel thread in the Non-secure state attempted to execute either: o DMAWFE to wait for a secure event o DMASEV to create a secure event or secure interrupt. This fault is a precise abort
4:2	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
1	RO	0x0	valid for the configuration of the DMAC: 0 = valid operand 1 = invalid operand. This fault is a precise abort
0	RO	0x0	0 = defined instruction 1 = undefined instruction. This fault is a precise abort

DMA_FTR2

Address: Operational Base + offset (0x0048)

Bit	Attr	Reset Value	Description
31	RO	0x0	0 = DMA channel has adequate resources 1 = DMA channel has locked-up because of insufficient resources. This fault is an imprecise abort
30	RO	0x0	memory or from the debug interface: 0 = instruction that generated an abort was read from system memory 1 = instruction that generated an abort was read from the debug interface. This fault is an imprecise abort but the bit is only valid when a precise abort occurs
29:19	RO	0x0	Reserved
18	RO	0x0	thread performs a data read: 0 = OKAY response 1 = EXOKAY, SLVERR, or DECERR response. This fault is an imprecise abort
17	RO	0x0	thread performs a data write: 0 = OKAY response 1 = EXOKAY, SLVERR, or DECERR response. This fault is an imprecise abort
16	RO	0x0	thread performs an instruction fetch: 0 = OKAY response 1 = EXOKAY, SLVERR, or DECERR response. This fault is a precise abort
15:14	RO	0x0	Reserved
13	RO	0x0	0 = MFIFO contains all the data to enable the DMAST to complete 1 = previous DMALDs have not put enough data in the MFIFO to enable the DMAST to complete. This fault is a precise abort

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
12	RO	0x0	DMALD 0 = MFIFO contains sufficient space 1 = MFIFO is too small to hold the data that DMALD requires. DMAST 0 = MFIFO contains sufficient data 1 = MFIFO is too small to store the data to enable DMAST to complete. This fault is an imprecise abort
11:8	RO	0x0	Reserved
7	RO	0x0	to perform a secure read or secure write: 0 = a DMA channel thread in the Non-secure state is not violating the security permissions 1 = a DMA channel thread in the Non-secure state attempted to perform a secure read or secure write. This fault is a precise abort
6	RO	0x0	DMASTP, or DMAFLUSHP with inappropriate security permissions: 0 = a DMA channel thread in the Non-secure state is not violating the security permissions 1 = a DMA channel thread in the Non-secure state attempted to execute either: o DMAWFP to wait for a secure peripheral o DMALDP or DMASTP to notify a secure peripheral o DMAFLUSHP to flush a secure peripheral. This fault is a precise abort
5	RO	0x0	0 = a DMA channel thread in the Non-secure state is not violating the security permissions 1 = a DMA channel thread in the Non-secure state attempted to execute either: o DMAWFE to wait for a secure event o DMASEV to create a secure event or secure interrupt. This fault is a precise abort
4:2	RO	0x0	Reserved
1	RO	0x0	valid for the configuration of the DMAC: 0 = valid operand 1 = invalid operand. This fault is a precise abort
0	RO	0x0	0 = defined instruction 1 = undefined instruction. This fault is a precise abort

DMA_FTR3

Address: Operational Base + offset (0x004c)

Bit	Attr	Reset Value	Description
31	RO	0x0	0 = DMA channel has adequate resources 1 = DMA channel has locked-up because of insufficient resources. This fault is an imprecise abort

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
30	RO	0x0	memory or from the debug interface: 0 = instruction that generated an abort was read from system memory 1 = instruction that generated an abort was read from the debug interface. This fault is an imprecise abort but the bit is only valid when a precise abort occurs
29:19	RO	0x0	Reserved
18	RO	0x0	thread performs a data read: 0 = OKAY response 1 = EXOKAY, SLVERR, or DECERR response. This fault is an imprecise abort
17	RO	0x0	thread performs a data write: 0 = OKAY response 1 = EXOKAY, SLVERR, or DECERR response. This fault is an imprecise abort
16	RO	0x0	thread performs an instruction fetch: 0 = OKAY response 1 = EXOKAY, SLVERR, or DECERR response. This fault is a precise abort
15:14	RO	0x0	Reserved
13	RO	0x0	0 = MFIFO contains all the data to enable the DMAST to complete 1 = previous DMALDs have not put enough data in the MFIFO to enable the DMAST to complete. This fault is a precise abort
12	RO	0x0	DMALD 0 = MFIFO contains sufficient space 1 = MFIFO is too small to hold the data that DMALD requires. DMAST 0 = MFIFO contains sufficient data 1 = MFIFO is too small to store the data to enable DMAST to complete. This fault is an imprecise abort
11:8	RO	0x0	Reserved
7	RO	0x0	to perform a secure read or secure write: 0 = a DMA channel thread in the Non-secure state is not violating the security permissions 1 = a DMA channel thread in the Non-secure state attempted to perform a secure read or secure write. This fault is a precise abort

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
6	RO	0x0	DMASTP, or DMAFLUSHP with inappropriate security permissions: 0 = a DMA channel thread in the Non-secure state is not violating the security permissions 1 = a DMA channel thread in the Non-secure state attempted to execute either: o DMAWFP to wait for a secure peripheral o DMALDP or DMASTP to notify a secure peripheral o DMAFLUSHP to flush a secure peripheral. This fault is a precise abort
5	RO	0x0	0 = a DMA channel thread in the Non-secure state is not violating the security permissions 1 = a DMA channel thread in the Non-secure state attempted to execute either: o DMAWFE to wait for a secure event o DMASEV to create a secure event or secure interrupt. This fault is a precise abort
4:2	RO	0x0	Reserved
1	RO	0x0	valid for the configuration of the DMAC: 0 = valid operand 1 = invalid operand. This fault is a precise abort
0	RO	0x0	0 = defined instruction 1 = undefined instruction. This fault is a precise abort

DMA_FTR4

Address: Operational Base + offset (0x0050)

Bit	Attr	Reset Value	Description
31	RO	0x0	0 = DMA channel has adequate resources 1 = DMA channel has locked-up because of insufficient resources. This fault is an imprecise abort
30	RO	0x0	memory or from the debug interface: 0 = instruction that generated an abort was read from system memory 1 = instruction that generated an abort was read from the debug interface. This fault is an imprecise abort but the bit is only valid when a precise abort occurs
29:19	RO	0x0	Reserved
18	RO	0x0	thread performs a data read: 0 = OKAY response 1 = EXOKAY, SLVERR, or DECERR response. This fault is an imprecise abort

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
17	RO	0x0	thread performs a data write: 0 = OKAY response 1 = EXOKAY, SLVERR, or DECERR response. This fault is an imprecise abort
16	RO	0x0	thread performs an instruction fetch: 0 = OKAY response 1 = EXOKAY, SLVERR, or DECERR response. This fault is a precise abort
15:14	RO	0x0	Reserved
13	RO	0x0	0 = MFIFO contains all the data to enable the DMAST to complete 1 = previous DMALDs have not put enough data in the MFIFO to enable the DMAST to complete. This fault is a precise abort
12	RO	0x0	DMALD 0 = MFIFO contains sufficient space 1 = MFIFO is too small to hold the data that DMALD requires. DMAST 0 = MFIFO contains sufficient data 1 = MFIFO is too small to store the data to enable DMAST to complete. This fault is an imprecise abort
11:8	RO	0x0	Reserved
7	RO	0x0	to perform a secure read or secure write: 0 = a DMA channel thread in the Non-secure state is not violating the security permissions 1 = a DMA channel thread in the Non-secure state attempted to perform a secure read or secure write. This fault is a precise abort
6	RO	0x0	DMASTP, or DMAFLUSHP with inappropriate security permissions: 0 = a DMA channel thread in the Non-secure state is not violating the security permissions 1 = a DMA channel thread in the Non-secure state attempted to execute either: o DMAWFP to wait for a secure peripheral o DMALDP or DMASTP to notify a secure peripheral o DMAFLUSHP to flush a secure peripheral. This fault is a precise abort
5	RO	0x0	0 = a DMA channel thread in the Non-secure state is not violating the security permissions 1 = a DMA channel thread in the Non-secure state attempted to execute either: o DMAWFE to wait for a secure event o DMASEV to create a secure event or secure interrupt. This fault is a precise abort
4:2	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
1	RO	0x0	valid for the configuration of the DMAC: 0 = valid operand 1 = invalid operand. This fault is a precise abort
0	RO	0x0	0 = defined instruction 1 = undefined instruction. This fault is a precise abort

DMA_FTR5

Address: Operational Base + offset (0x0054)

Bit	Attr	Reset Value	Description
31	RO	0x0	0 = DMA channel has adequate resources 1 = DMA channel has locked-up because of insufficient resources. This fault is an imprecise abort
30	RO	0x0	memory or from the debug interface: 0 = instruction that generated an abort was read from system memory 1 = instruction that generated an abort was read from the debug interface. This fault is an imprecise abort but the bit is only valid when a precise abort occurs
29:19	RO	0x0	Reserved
18	RO	0x0	thread performs a data read: 0 = OKAY response 1 = EXOKAY, SLVERR, or DECERR response. This fault is an imprecise abort
17	RO	0x0	thread performs a data write: 0 = OKAY response 1 = EXOKAY, SLVERR, or DECERR response. This fault is an imprecise abort
16	RO	0x0	thread performs an instruction fetch: 0 = OKAY response 1 = EXOKAY, SLVERR, or DECERR response. This fault is a precise abort
15:14	RO	0x0	Reserved
13	RO	0x0	0 = MFIFO contains all the data to enable the DMAST to complete 1 = previous DMALDs have not put enough data in the MFIFO to enable the DMAST to complete. This fault is a precise abort
12	RO	0x0	DMALD 0 = MFIFO contains sufficient space 1 = MFIFO is too small to hold the data that DMALD requires. DMAST 0 = MFIFO contains sufficient data 1 = MFIFO is too small to store the data to enable DMAST to complete. This fault is an imprecise abort

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
11:8	RO	0x0	Reserved
7	RO	0x0	to perform a secure read or secure write: 0 = a DMA channel thread in the Non-secure state is not violating the security permissions 1 = a DMA channel thread in the Non-secure state attempted to perform a secure read or secure write. This fault is a precise abort
6	RO	0x0	DMASTP, or DMAFLUSHP with inappropriate security permissions: 0 = a DMA channel thread in the Non-secure state is not violating the security permissions 1 = a DMA channel thread in the Non-secure state attempted to execute either: o DMAWFP to wait for a secure peripheral o DMALDP or DMASTP to notify a secure peripheral o DMAFLUSHP to flush a secure peripheral. This fault is a precise abort
5	RO	0x0	0 = a DMA channel thread in the Non-secure state is not violating the security permissions 1 = a DMA channel thread in the Non-secure state attempted to execute either: o DMAWFE to wait for a secure event o DMASEV to create a secure event or secure interrupt. This fault is a precise abort
4:2	RO	0x0	Reserved
1	RO	0x0	valid for the configuration of the DMAC: 0 = valid operand 1 = invalid operand. This fault is a precise abort
0	RO	0x0	0 = defined instruction 1 = undefined instruction. This fault is a precise abort

DMA_CSR0

Address: Operational Base + offset (0x0100)

Bit	Attr	Reset Value	Description
31:22	RO	0x0	Reserved
21	RO	0x0	0 = DMA channel operates in the Secure state 1 = DMA channel operates in the Non-secure state
20:16	RO	0x0	Reserved
15	RO	0x0	0 = DMAWFP executed with the periph operand not set 1 = DMAWFP executed with the periph operand set
14	RO	0x0	0 = DMAWFP executed with the single operand set 1 = DMAWFP executed with the burst operand set
13:9	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
8:4	RO	0x00	indicate the event or peripheral number that the channel is waiting for: b00000 = DMA channel is waiting for event, or peripheral, 0 b00001 = DMA channel is waiting for event, or peripheral, 1 b00010 = DMA channel is waiting for event, or peripheral, 2 . . . b11111 = DMA channel is waiting for event, or peripheral, 31
3:0	RO	0x0	b0000 = Stopped b0001 = Executing b0010 = Cache miss b0011 = Updating PC b0100 = Waiting for event b0101 = At barrier b0110 = Reserved b0111 = Waiting for peripheral b1000 = Killing b1001 = Completing b1010-b1101 = Reserved b1110 = Faulting completing b1111 = Faulting

DMA_CPC0

Address: Operational Base + offset (0x0104)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Program counter for the DMA channel 0 thread

DMA_CSR1

Address: Operational Base + offset (0x0108)

Bit	Attr	Reset Value	Description
31:22	RO	0x0	Reserved
21	RO	0x0	0 = DMA channel operates in the Secure state 1 = DMA channel operates in the Non-secure state
20:16	RO	0x0	Reserved
15	RO	0x0	0 = DMAWFP executed with the periph operand not set 1 = DMAWFP executed with the periph operand set
14	RO	0x0	0 = DMAWFP executed with the single operand set 1 = DMAWFP executed with the burst operand set
13:9	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
8:4	RO	0x00	indicate the event or peripheral number that the channel is waiting for: b00000 = DMA channel is waiting for event, or peripheral, 0 b00001 = DMA channel is waiting for event, or peripheral, 1 b00010 = DMA channel is waiting for event, or peripheral, 2 . . . b11111 = DMA channel is waiting for event, or peripheral, 31
3:0	RO	0x0	b0000 = Stopped b0001 = Executing b0010 = Cache miss b0011 = Updating PC b0100 = Waiting for event b0101 = At barrier b0110 = Reserved b0111 = Waiting for peripheral b1000 = Killing b1001 = Completing b1010-b1101 = Reserved b1110 = Faulting completing b1111 = Faulting

DMA_CPC1

Address: Operational Base + offset (0x010c)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Program counter for the DMA channel 1 thread

DMA_CSR2

Address: Operational Base + offset (0x0110)

Bit	Attr	Reset Value	Description
31:22	RO	0x0	Reserved
21	RO	0x0	0 = DMA channel operates in the Secure state 1 = DMA channel operates in the Non-secure state
20:16	RO	0x0	Reserved
15	RO	0x0	0 = DMAWFP executed with the periph operand not set 1 = DMAWFP executed with the periph operand set
14	RO	0x0	0 = DMAWFP executed with the single operand set 1 = DMAWFP executed with the burst operand set
13:9	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
8:4	RO	0x00	indicate the event or peripheral number that the channel is waiting for: b00000 = DMA channel is waiting for event, or peripheral, 0 b00001 = DMA channel is waiting for event, or peripheral, 1 b00010 = DMA channel is waiting for event, or peripheral, 2 . . . b11111 = DMA channel is waiting for event, or peripheral, 31
3:0	RO	0x0	b0000 = Stopped b0001 = Executing b0010 = Cache miss b0011 = Updating PC b0100 = Waiting for event b0101 = At barrier b0110 = Reserved b0111 = Waiting for peripheral b1000 = Killing b1001 = Completing b1010-b1101 = Reserved b1110 = Faulting completing b1111 = Faulting

DMA_CPC2

Address: Operational Base + offset (0x0114)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Program counter for the DMA channel 2 thread

DMA_CSR3

Address: Operational Base + offset (0x0118)

Bit	Attr	Reset Value	Description
31:22	RO	0x0	Reserved
21	RO	0x0	0 = DMA channel operates in the Secure state 1 = DMA channel operates in the Non-secure state
20:16	RO	0x0	Reserved
15	RO	0x0	0 = DMAWFP executed with the periph operand not set 1 = DMAWFP executed with the periph operand set
14	RO	0x0	0 = DMAWFP executed with the single operand set 1 = DMAWFP executed with the burst operand set
13:9	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
8:4	RO	0x00	indicate the event or peripheral number that the channel is waiting for: b00000 = DMA channel is waiting for event, or peripheral, 0 b00001 = DMA channel is waiting for event, or peripheral, 1 b00010 = DMA channel is waiting for event, or peripheral, 2 . . . b11111 = DMA channel is waiting for event, or peripheral, 31
3:0	RO	0x0	b0000 = Stopped b0001 = Executing b0010 = Cache miss b0011 = Updating PC b0100 = Waiting for event b0101 = At barrier b0110 = Reserved b0111 = Waiting for peripheral b1000 = Killing b1001 = Completing b1010-b1101 = Reserved b1110 = Faulting completing b1111 = Faulting

DMA_CPC3

Address: Operational Base + offset (0x011c)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Program counter for the DMA channel 3 thread

DMA_CSR4

Address: Operational Base + offset (0x0120)

Bit	Attr	Reset Value	Description
31:22	RO	0x0	Reserved
21	RO	0x0	0 = DMA channel operates in the Secure state 1 = DMA channel operates in the Non-secure state
20:16	RO	0x0	Reserved
15	RO	0x0	0 = DMAWFP executed with the periph operand not set 1 = DMAWFP executed with the periph operand set
14	RO	0x0	0 = DMAWFP executed with the single operand set 1 = DMAWFP executed with the burst operand set
13:9	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
8:4	RO	0x00	indicate the event or peripheral number that the channel is waiting for: b00000 = DMA channel is waiting for event, or peripheral, 0 b00001 = DMA channel is waiting for event, or peripheral, 1 b00010 = DMA channel is waiting for event, or peripheral, 2 . . . b11111 = DMA channel is waiting for event, or peripheral, 31
3:0	RO	0x0	b0000 = Stopped b0001 = Executing b0010 = Cache miss b0011 = Updating PC b0100 = Waiting for event b0101 = At barrier b0110 = Reserved b0111 = Waiting for peripheral b1000 = Killing b1001 = Completing b1010-b1101 = Reserved b1110 = Faulting completing b1111 = Faulting

DMA_CPC4

Address: Operational Base + offset (0x0124)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Program counter for the DMA channel 4 thread

DMA_CSR5

Address: Operational Base + offset (0x0128)

Bit	Attr	Reset Value	Description
31:22	RO	0x0	Reserved
21	RO	0x0	0 = DMA channel operates in the Secure state 1 = DMA channel operates in the Non-secure state
20:16	RO	0x0	Reserved
15	RO	0x0	0 = DMAWFP executed with the periph operand not set 1 = DMAWFP executed with the periph operand set
14	RO	0x0	0 = DMAWFP executed with the single operand set 1 = DMAWFP executed with the burst operand set
13:9	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
8:4	RO	0x00	indicate the event or peripheral number that the channel is waiting for: b00000 = DMA channel is waiting for event, or peripheral, 0 b00001 = DMA channel is waiting for event, or peripheral, 1 b00010 = DMA channel is waiting for event, or peripheral, 2 . . . b11111 = DMA channel is waiting for event, or peripheral, 31
3:0	RO	0x0	b0000 = Stopped b0001 = Executing b0010 = Cache miss b0011 = Updating PC b0100 = Waiting for event b0101 = At barrier b0110 = Reserved b0111 = Waiting for peripheral b1000 = Killing b1001 = Completing b1010-b1101 = Reserved b1110 = Faulting completing b1111 = Faulting

DMA_CPC5

Address: Operational Base + offset (0x012c)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Program counter for the DMA channel 5 thread

DMA_SAR0

Address: Operational Base + offset (0x0400)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Address of the source data for DMA channel 0

DMA_DAR0

Address: Operational Base + offset (0x0404)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Address of the Destinationdata for DMA channel 0

DMA_CCR0

Address: Operational Base + offset (0x0408)

Bit	Attr	Reset Value	Description
31:28	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
27:25	RO	0x0	Bit [27] 0 = AWCACHE[3] is LOW 1 = AWCACHE[3] is HIGH. Bit [26] 0 = AWCACHE[1] is LOW 1 = AWCACHE[1] is HIGH. Bit [25] 0 = AWCACHE[0] is LOW 1 = AWCACHE[0] is HIGH
24:22	RO	0x0	Bit [24] 0 = AWPROT[2] is LOW 1 = AWPROT[2] is HIGH. Bit [23] 0 = AWPROT[1] is LOW 1 = AWPROT[1] is HIGH. Bit [22] 0 = AWPROT[0] is LOW 1 = AWPROT[0] is HIGH
21:18	RO	0x0	the destination data: b0000 = 1 data transfer b0001 = 2 data transfers b0010 = 3 data transfers . . . b1111 = 16 data transfers. The total number of bytes that the DMAC writes out of the MFIFO when it executes a DMAST instruction is the product of dst_burst_len and dst_burst_size
17:15	RO	0x0	b000 = writes 1 byte per beat b001 = writes 2 bytes per beat b010 = writes 4 bytes per beat b011 = writes 8 bytes per beat b100 = writes 16 bytes per beat b101-b111 = Reserved. The total number of bytes that the DMAC writes out of the MFIFO when it executes a DMAST instruction is the product of dst_burst_len and dst_burst_size
14	RO	0x0	0 = Fixed-address burst. The DMAC signals AWBURST[0] LOW. 1 = Incrementing-address burst. The DMAC signals AWBURST[0] HIGH
13:11	RO	0x0	Bit [13] 0 = ARCACHE[2] is LOW 1 = ARCACHE[2] is HIGH. Bit [12] 0 = ARCACHE[1] is LOW 1 = ARCACHE[1] is HIGH. Bit [11] 0 = ARCACHE[0] is LOW 1 = ARCACHE[0] is HIGH

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
10:8	RO	0x0	Bit [10] 0 = ARPROT[2] is LOW 1 = ARPROT[2] is HIGH. Bit [9] 0 = ARPROT[1] is LOW 1 = ARPROT[1] is HIGH. Bit [8] 0 = ARPROT[0] is LOW 1 = ARPROT[0] is HIGH
7:4	RO	0x0	b0000 = 1 data transfer b0001 = 2 data transfers b0010 = 3 data transfers . . . b1111 = 16 data transfers. The total number of bytes that the DMAC reads into the MFIFO when it executes a DMALD instruction is the product of src_burst_len and src_burst_size
3:1	RO	0x0	b000 = reads 1 byte per beat b001 = reads 2 bytes per beat b010 = reads 4 bytes per beat b011 = reads 8 bytes per beat b100 = reads 16 bytes per beat b101-b111 = Reserved. The total number of bytes that the DMAC reads into the MFIFO when it executes a DMALD instruction is the product of src_burst_len and src_burst_size
0	RO	0x0	0 = Fixed-address burst. The DMAC signals ARBURST[0] LOW. 1 = Incrementing-address burst. The DMAC signals ARBURST[0] HIGH

DMA_LC0_0

Address: Operational Base + offset (0x040c)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	Reserved
7:0	RO	0x00	Loop counter 0 iterations

DMA_LC1_0

Address: Operational Base + offset (0x0410)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	Reserved
7:0	RO	0x00	Loop counter 1 iterations

DMA_SAR1

Address: Operational Base + offset (0x0420)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Address of the source data for DMA channel 1

DMA_DAR1

Address: Operational Base + offset (0x0424)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Address of the Destinationdata for DMA channel 1

DMA_CCR1

Address: Operational Base + offset (0x0428)

Bit	Attr	Reset Value	Description
31:28	RO	0x0	Reserved
27:25	RO	0x0	Bit [27] 0 = AWCACHE[3] is LOW 1 = AWCACHE[3] is HIGH. Bit [26] 0 = AWCACHE[1] is LOW 1 = AWCACHE[1] is HIGH. Bit [25] 0 = AWCACHE[0] is LOW 1 = AWCACHE[0] is HIGH
24:22	RO	0x0	Bit [24] 0 = AWPROT[2] is LOW 1 = AWPROT[2] is HIGH. Bit [23] 0 = AWPROT[1] is LOW 1 = AWPROT[1] is HIGH. Bit [22] 0 = AWPROT[0] is LOW 1 = AWPROT[0] is HIGH
21:18	RO	0x0	the destination data: b0000 = 1 data transfer b0001 = 2 data transfers b0010 = 3 data transfers . . . b1111 = 16 data transfers. The total number of bytes that the DMAC writes out of the MFIFO when it executes a DMAST instruction is the product of dst_burst_len and dst_burst_size
17:15	RO	0x0	b000 = writes 1 byte per beat b001 = writes 2 bytes per beat b010 = writes 4 bytes per beat b011 = writes 8 bytes per beat b100 = writes 16 bytes per beat b101-b111 = Reserved. The total number of bytes that the DMAC writes out of the MFIFO when it executes a DMAST instruction is the product of dst_burst_len and dst_burst_size
14	RO	0x0	0 = Fixed-address burst. The DMAC signals AWBURST[0] LOW. 1 = Incrementing-address burst. The DMAC signals AWBURST[0] HIGH

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
13:11	RO	0x0	Bit [13] 0 = ARCACHE[2] is LOW 1 = ARCACHE[2] is HIGH. Bit [12] 0 = ARCACHE[1] is LOW 1 = ARCACHE[1] is HIGH. Bit [11] 0 = ARCACHE[0] is LOW 1 = ARCACHE[0] is HIGH
10:8	RO	0x0	Bit [10] 0 = ARPROT[2] is LOW 1 = ARPROT[2] is HIGH. Bit [9] 0 = ARPROT[1] is LOW 1 = ARPROT[1] is HIGH. Bit [8] 0 = ARPROT[0] is LOW 1 = ARPROT[0] is HIGH
7:4	RO	0x0	b0000 = 1 data transfer b0001 = 2 data transfers b0010 = 3 data transfers . . . b1111 = 16 data transfers. The total number of bytes that the DMAC reads into the MFIFO when it executes a DMALD instruction is the product of src_burst_len and src_burst_size
3:1	RO	0x0	b000 = reads 1 byte per beat b001 = reads 2 bytes per beat b010 = reads 4 bytes per beat b011 = reads 8 bytes per beat b100 = reads 16 bytes per beat b101-b111 = Reserved. The total number of bytes that the DMAC reads into the MFIFO when it executes a DMALD instruction is the product of src_burst_len and src_burst_size
0	RO	0x0	0 = Fixed-address burst. The DMAC signals ARBURST[0] LOW. 1 = Incrementing-address burst. The DMAC signals ARBURST[0] HIGH

DMA_LC0_1

Address: Operational Base + offset (0x042c)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	Reserved
7:0	RO	0x00	Loop counter 0 iterations

DMA_LC1_1

Address: Operational Base + offset (0x0430)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:8	RO	0x0	Reserved
7:0	RO	0x00	Loop counter 1 iterations

DMA_SAR2

Address: Operational Base + offset (0x0440)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Address of the source data for DMA channel 2

DMA_DAR2

Address: Operational Base + offset (0x0444)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Address of the Destinationdata for DMA channel 2

DMA_CCR2

Address: Operational Base + offset (0x0448)

Bit	Attr	Reset Value	Description
31:28	RO	0x0	Reserved
27:25	RO	0x0	Bit [27] 0 = AWCACHE[3] is LOW 1 = AWCACHE[3] is HIGH. Bit [26] 0 = AWCACHE[1] is LOW 1 = AWCACHE[1] is HIGH. Bit [25] 0 = AWCACHE[0] is LOW 1 = AWCACHE[0] is HIGH
24:22	RO	0x0	Bit [24] 0 = AWPROT[2] is LOW 1 = AWPROT[2] is HIGH. Bit [23] 0 = AWPROT[1] is LOW 1 = AWPROT[1] is HIGH. Bit [22] 0 = AWPROT[0] is LOW 1 = AWPROT[0] is HIGH
21:18	RO	0x0	the destination data: b0000 = 1 data transfer b0001 = 2 data transfers b0010 = 3 data transfers . . . b1111 = 16 data transfers. The total number of bytes that the DMAC writes out of the MFIFO when it executes a DMAST instruction is the product of dst_burst_len and dst_burst_size

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
17:15	RO	0x0	b000 = writes 1 byte per beat b001 = writes 2 bytes per beat b010 = writes 4 bytes per beat b011 = writes 8 bytes per beat b100 = writes 16 bytes per beat b101-b111 = Reserved. The total number of bytes that the DMAC writes out of the MFIFO when it executes a DMAST instruction is the product of dst_burst_len and dst_burst_size
14	RO	0x0	0 = Fixed-address burst. The DMAC signals AWBURST[0] LOW. 1 = Incrementing-address burst. The DMAC signals AWBURST[0] HIGH
13:11	RO	0x0	Bit [13] 0 = ARCACHE[2] is LOW 1 = ARCACHE[2] is HIGH. Bit [12] 0 = ARCACHE[1] is LOW 1 = ARCACHE[1] is HIGH. Bit [11] 0 = ARCACHE[0] is LOW 1 = ARCACHE[0] is HIGH
10:8	RO	0x0	Bit [10] 0 = ARPROT[2] is LOW 1 = ARPROT[2] is HIGH. Bit [9] 0 = ARPROT[1] is LOW 1 = ARPROT[1] is HIGH. Bit [8] 0 = ARPROT[0] is LOW 1 = ARPROT[0] is HIGH
7:4	RO	0x0	b0000 = 1 data transfer b0001 = 2 data transfers b0010 = 3 data transfers . . . b1111 = 16 data transfers. The total number of bytes that the DMAC reads into the MFIFO when it executes a DMALD instruction is the product of src_burst_len and src_burst_size
3:1	RO	0x0	b000 = reads 1 byte per beat b001 = reads 2 bytes per beat b010 = reads 4 bytes per beat b011 = reads 8 bytes per beat b100 = reads 16 bytes per beat b101-b111 = Reserved. The total number of bytes that the DMAC reads into the MFIFO when it executes a DMALD instruction is the product of src_burst_len and src_burst_size

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
0	RO	0x0	0 = Fixed-address burst. The DMAC signals ARBURST[0] LOW. 1 = Incrementing-address burst. The DMAC signals ARBURST[0] HIGH

DMA_LC0_2

Address: Operational Base + offset (0x044c)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	Reserved
7:0	RO	0x00	Loop counter 0 iterations

DMA_LC1_2

Address: Operational Base + offset (0x0450)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	Reserved
7:0	RO	0x00	Loop counter 1 iterations

DMA_SAR3

Address: Operational Base + offset (0x0460)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Address of the source data for DMA channel 3

DMA_DAR3

Address: Operational Base + offset (0x0464)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Address of the Destinationdata for DMA channel 3

DMA_CCR3

Address: Operational Base + offset (0x0468)

Bit	Attr	Reset Value	Description
31:28	RO	0x0	Reserved
27:25	RO	0x0	Bit [27] 0 = AWCACHE[3] is LOW 1 = AWCACHE[3] is HIGH. Bit [26] 0 = AWCACHE[1] is LOW 1 = AWCACHE[1] is HIGH. Bit [25] 0 = AWCACHE[0] is LOW 1 = AWCACHE[0] is HIGH
24:22	RO	0x0	Bit [24] 0 = AWPROT[2] is LOW 1 = AWPROT[2] is HIGH. Bit [23] 0 = AWPROT[1] is LOW 1 = AWPROT[1] is HIGH. Bit [22] 0 = AWPROT[0] is LOW 1 = AWPROT[0] is HIGH

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
21:18	RO	0x0	the destination data: b0000 = 1 data transfer b0001 = 2 data transfers b0010 = 3 data transfers . . . b1111 = 16 data transfers. The total number of bytes that the DMAC writes out of the MFIFO when it executes a DMAST instruction is the product of dst_burst_len and dst_burst_size
17:15	RO	0x0	b000 = writes 1 byte per beat b001 = writes 2 bytes per beat b010 = writes 4 bytes per beat b011 = writes 8 bytes per beat b100 = writes 16 bytes per beat b101-b111 = Reserved. The total number of bytes that the DMAC writes out of the MFIFO when it executes a DMAST instruction is the product of dst_burst_len and dst_burst_size
14	RO	0x0	0 = Fixed-address burst. The DMAC signals AWBURST[0] LOW. 1 = Incrementing-address burst. The DMAC signals AWBURST[0] HIGH
13:11	RO	0x0	Bit [13] 0 = ARCACHE[2] is LOW 1 = ARCACHE[2] is HIGH. Bit [12] 0 = ARCACHE[1] is LOW 1 = ARCACHE[1] is HIGH. Bit [11] 0 = ARCACHE[0] is LOW 1 = ARCACHE[0] is HIGH
10:8	RO	0x0	Bit [10] 0 = ARPROT[2] is LOW 1 = ARPROT[2] is HIGH. Bit [9] 0 = ARPROT[1] is LOW 1 = ARPROT[1] is HIGH. Bit [8] 0 = ARPROT[0] is LOW 1 = ARPROT[0] is HIGH
7:4	RO	0x0	b0000 = 1 data transfer b0001 = 2 data transfers b0010 = 3 data transfers . . . b1111 = 16 data transfers. The total number of bytes that the DMAC reads into the MFIFO when it executes a DMALD instruction is the product of src_burst_len and src_burst_size

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
3:1	RO	0x0	b000 = reads 1 byte per beat b001 = reads 2 bytes per beat b010 = reads 4 bytes per beat b011 = reads 8 bytes per beat b100 = reads 16 bytes per beat b101-b111 = Reserved. The total number of bytes that the DMAC reads into the MFIFO when it executes a DMALD instruction is the product of src_burst_len and src_burst_size
0	RO	0x0	0 = Fixed-address burst. The DMAC signals ARBURST[0] LOW. 1 = Incrementing-address burst. The DMAC signals ARBURST[0] HIGH

DMA_LC0_3

Address: Operational Base + offset (0x046c)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	Reserved
7:0	RO	0x00	Loop counter 0 iterations

DMA_LC1_3

Address: Operational Base + offset (0x0470)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	Reserved
7:0	RO	0x00	Loop counter 1 iterations

DMA_SAR4

Address: Operational Base + offset (0x0480)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Address of the source data for DMA channel 4

DMA_DAR4

Address: Operational Base + offset (0x0484)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Address of the Destinationdata for DMA channel 4

DMA_CCR4

Address: Operational Base + offset (0x0488)

Bit	Attr	Reset Value	Description
31:28	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
27:25	RO	0x0	Bit [27] 0 = AWCACHE[3] is LOW 1 = AWCACHE[3] is HIGH. Bit [26] 0 = AWCACHE[1] is LOW 1 = AWCACHE[1] is HIGH. Bit [25] 0 = AWCACHE[0] is LOW 1 = AWCACHE[0] is HIGH
24:22	RO	0x0	Bit [24] 0 = AWPROT[2] is LOW 1 = AWPROT[2] is HIGH. Bit [23] 0 = AWPROT[1] is LOW 1 = AWPROT[1] is HIGH. Bit [22] 0 = AWPROT[0] is LOW 1 = AWPROT[0] is HIGH
21:18	RO	0x0	the destination data: b0000 = 1 data transfer b0001 = 2 data transfers b0010 = 3 data transfers . . . b1111 = 16 data transfers. The total number of bytes that the DMAC writes out of the MFIFO when it executes a DMAST instruction is the product of dst_burst_len and dst_burst_size
17:15	RO	0x0	b000 = writes 1 byte per beat b001 = writes 2 bytes per beat b010 = writes 4 bytes per beat b011 = writes 8 bytes per beat b100 = writes 16 bytes per beat b101-b111 = Reserved. The total number of bytes that the DMAC writes out of the MFIFO when it executes a DMAST instruction is the product of dst_burst_len and dst_burst_size
14	RO	0x0	0 = Fixed-address burst. The DMAC signals AWBURST[0] LOW. 1 = Incrementing-address burst. The DMAC signals AWBURST[0] HIGH
13:11	RO	0x0	Bit [13] 0 = ARCACHE[2] is LOW 1 = ARCACHE[2] is HIGH. Bit [12] 0 = ARCACHE[1] is LOW 1 = ARCACHE[1] is HIGH. Bit [11] 0 = ARCACHE[0] is LOW 1 = ARCACHE[0] is HIGH

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
10:8	RO	0x0	Bit [10] 0 = ARPROT[2] is LOW 1 = ARPROT[2] is HIGH. Bit [9] 0 = ARPROT[1] is LOW 1 = ARPROT[1] is HIGH. Bit [8] 0 = ARPROT[0] is LOW 1 = ARPROT[0] is HIGH
7:4	RO	0x0	b0000 = 1 data transfer b0001 = 2 data transfers b0010 = 3 data transfers . . . b1111 = 16 data transfers. The total number of bytes that the DMAC reads into the MFIFO when it executes a DMALD instruction is the product of src_burst_len and src_burst_size
3:1	RO	0x0	b000 = reads 1 byte per beat b001 = reads 2 bytes per beat b010 = reads 4 bytes per beat b011 = reads 8 bytes per beat b100 = reads 16 bytes per beat b101-b111 = Reserved. The total number of bytes that the DMAC reads into the MFIFO when it executes a DMALD instruction is the product of src_burst_len and src_burst_size
0	RO	0x0	0 = Fixed-address burst. The DMAC signals ARBURST[0] LOW. 1 = Incrementing-address burst. The DMAC signals ARBURST[0] HIGH

DMA_LC0_4

Address: Operational Base + offset (0x048c)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	Reserved
7:0	RO	0x00	Loop counter 0 iterations

DMA_LC1_4

Address: Operational Base + offset (0x0490)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	Reserved
7:0	RO	0x00	Loop counter 1 iterations

DMA_SAR5

Address: Operational Base + offset (0x04a0)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Address of the source data for DMA channel 5

DMA_DAR5

Address: Operational Base + offset (0x04a4)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Address of the Destinationdata for DMA channel 5

DMA_CCR5

Address: Operational Base + offset (0x04a8)

Bit	Attr	Reset Value	Description
31:28	RO	0x0	Reserved
27:25	RO	0x0	Bit [27] 0 = AWCACHE[3] is LOW 1 = AWCACHE[3] is HIGH. Bit [26] 0 = AWCACHE[1] is LOW 1 = AWCACHE[1] is HIGH. Bit [25] 0 = AWCACHE[0] is LOW 1 = AWCACHE[0] is HIGH
24:22	RO	0x0	Bit [24] 0 = AWPROT[2] is LOW 1 = AWPROT[2] is HIGH. Bit [23] 0 = AWPROT[1] is LOW 1 = AWPROT[1] is HIGH. Bit [22] 0 = AWPROT[0] is LOW 1 = AWPROT[0] is HIGH
21:18	RO	0x0	the destination data: b0000 = 1 data transfer b0001 = 2 data transfers b0010 = 3 data transfers . . . b1111 = 16 data transfers. The total number of bytes that the DMAC writes out of the MFIFO when it executes a DMAST instruction is the product of dst_burst_len and dst_burst_size
17:15	RO	0x0	b000 = writes 1 byte per beat b001 = writes 2 bytes per beat b010 = writes 4 bytes per beat b011 = writes 8 bytes per beat b100 = writes 16 bytes per beat b101-b111 = Reserved. The total number of bytes that the DMAC writes out of the MFIFO when it executes a DMAST instruction is the product of dst_burst_len and dst_burst_size

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
14	RO	0x0	0 = Fixed-address burst. The DMAC signals AWBURST[0] LOW. 1 = Incrementing-address burst. The DMAC signals AWBURST[0] HIGH
13:11	RO	0x0	Bit [13] 0 = ARCACHE[2] is LOW 1 = ARCACHE[2] is HIGH. Bit [12] 0 = ARCACHE[1] is LOW 1 = ARCACHE[1] is HIGH. Bit [11] 0 = ARCACHE[0] is LOW 1 = ARCACHE[0] is HIGH
10:8	RO	0x0	Bit [10] 0 = ARPROT[2] is LOW 1 = ARPROT[2] is HIGH. Bit [9] 0 = ARPROT[1] is LOW 1 = ARPROT[1] is HIGH. Bit [8] 0 = ARPROT[0] is LOW 1 = ARPROT[0] is HIGH
7:4	RO	0x0	b0000 = 1 data transfer b0001 = 2 data transfers b0010 = 3 data transfers . . . b1111 = 16 data transfers. The total number of bytes that the DMAC reads into the MFIFO when it executes a DMALD instruction is the product of src_burst_len and src_burst_size
3:1	RO	0x0	b000 = reads 1 byte per beat b001 = reads 2 bytes per beat b010 = reads 4 bytes per beat b011 = reads 8 bytes per beat b100 = reads 16 bytes per beat b101-b111 = Reserved. The total number of bytes that the DMAC reads into the MFIFO when it executes a DMALD instruction is the product of src_burst_len and src_burst_size
0	RO	0x0	0 = Fixed-address burst. The DMAC signals ARBURST[0] LOW. 1 = Incrementing-address burst. The DMAC signals ARBURST[0] HIGH

DMA_LC0_5

Address: Operational Base + offset (0x04ac)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	Reserved
7:0	RO	0x00	Loop counter 0 iterations

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

DMA_LC1_5

Address: Operational Base + offset (0x04b0)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	Reserved
7:0	RO	0x00	Loop counter 1 iterations

DMA_DBGSTATUS

Address: Operational Base + offset (0x0d00)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	Reserved
1:0	RO	0x0	b00 = execute the instruction that the DBGINST [1:0] Registers contain b01 = Reserved b10 = Reserved b11 = Reserved

DMA_DBGCMD

Address: Operational Base + offset (0x0d04)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	Reserved
1:0	WO	0x0	b00 = execute the instruction that the DBGINST [1:0] Registers contain b01 = Reserved b10 = Reserved b11 = Reserved

DMA_DBGINST0

Address: Operational Base + offset (0x0d08)

Bit	Attr	Reset Value	Description
31:24	WO	0x00	Instruction byte 1
23:16	WO	0x00	Instruction byte 0
15:11	RO	0x0	Reserved
10:8	WO	0x0	b000 = DMA channel 0 b001 = DMA channel 1 b010 = DMA channel 2 … b111 = DMA channel 7
7:1	RO	0x0	Reserved
0	WO	0x0	0 = DMA manager thread 1 = DMA channel

DMA_DBGINST1

Address: Operational Base + offset (0x0d0c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:24	WO	0x00	Instruction byte 5
23:16	WO	0x00	Instruction byte 4
15:8	WO	0x00	Instruction byte 3
7:0	WO	0x00	Instruction byte 2

DMA_CR0

Address: Operational Base + offset (0x0e00)

Bit	Attr	Reset Value	Description
31:22	RO	0x0	Reserved
21:17	RO	0x02	b00000 = 1 interrupt output, irq[0] b00001 = 2 interrupt outputs, irq[1:0] b00010 = 3 interrupt outputs, irq[2:0] . . . b11111 = 32 interrupt outputs, irq[31:0]
16:12	RO	0x07	b00000 = 1 peripheral request interface b00001 = 2 peripheral request interfaces b00010 = 3 peripheral request interfaces . . . b11111 = 32 peripheral request interfaces
11:7	RO	0x0	Reserved
6:4	RO	0x5	b000 = 1 DMA channel b001 = 2 DMA channels b010 = 3 DMA channels . . . b111 = 8 DMA channels
3	RO	0x0	Reserved
2	RO	0x0	0 = boot_manager_ns was LOW 1 = boot_manager_ns was HIGH
1	RO	0x0	0 = boot_from_pc was LOW 1 = boot_from_pc was HIGH
0	RO	0x1	0 = the DMAC does not provide a peripheral request interface 1 = the DMAC provides the number of peripheral request interfaces that the num_periph_req field specifies

DMA_CR1

Address: Operational Base + offset (0x0e04)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:8	RO	0x0	Reserved
7:4	RO	0x5	b0000 = 1 i-cache line b0001 = 2 i-cache lines b0010 = 3 i-cache lines … b1111 = 16 i-cache lines
3	RO	0x0	Reserved
2:0	RO	0x7	b000-b001 = Reserved b010 = 4 bytes b011 = 8 bytes b100 = 16 bytes b101 = 32 bytes b110-b111 = Reserved

DMA_CR2

Address: Operational Base + offset (0x0e08)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Provides the value of boot_addr[31:0] when the DMAC exited from reset

DMA_CR3

Address: Operational Base + offset (0x0e0c)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Bit [N] = 0 Assigns event<N> or irq[N] to the Secure state. Bit [N] = 1 Assigns event<N> or irq[N] to the Non-secure state

DMA_CR4

Address: Operational Base + offset (0x0e10)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000006	Bit [N] = 0 Assigns peripheral request interface N to the Secure state. Bit [N] = 1 Assigns peripheral request interface N to the Non-secure state

DMA_CRDn

Address: Operational Base + offset (0x0e14)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	Reserved
29:20	RO	0x020	b000000000 = 1 line b000000001 = 2 lines … b111111111 = 1024 lines

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
19:16	RO	0x9	b0000 = 1 line b0001 = 2 lines . . . b1111 = 16 lines
15	RO	0x0	Reserved
14:12	RO	0x4	b000 = 1 b001 = 2 … b111 = 8
11:8	RO	0x7	b0000 = 1 line b0001 = 2 lines … b1111 = 16 lines
7	RO	0x0	Reserved
6:4	RO	0x3	b000 = 1 b001 = 2 … b111 = 8
3	RO	0x0	Reserved
2:0	RO	0x3	b000 = Reserved b001 = Reserved b010 = 32-bit b011 = 64-bit b100 = 128-bit b101-b111 = Reserved

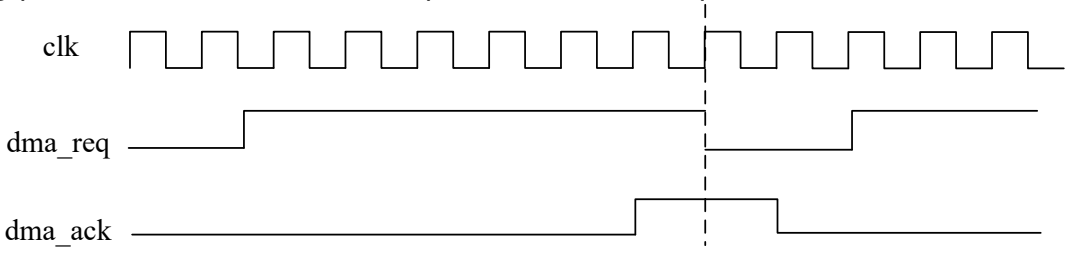
DMA_WD

Address: Operational Base + offset (0x0e80)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	Reserved
0	RW	0x0	0 = the DMAC aborts all of the contributing DMA channels and sets irq_abort HIGH 1 = the DMAC sets irq_abort HIGH

6.5 Timing Diagram

Following picture shows the relationship between dma_req and dma_ack.



Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Fig. 6-3 DMAC request and acknowledge timing

6.6 Interface Description

DMAC has the following tie-off signals.It can be configured by SGRF register. (Please refer to the GRF chapter to find them)

Table 6-2 DMAC boot interface

Interface	Reset value	Control source
boot_irq_ns	0xFFFF	sgrf_dmac_con0[15:0]
boot_periph_ns	0xFFFFFFFF	{sgrf_dmac_con2[15:0],sgrf_dmac_con1[15:0]}
boot_manager_ns	0x1	sgrf_dmac_con3[0]
grf_drtype_ch0	0x1	sgrf_dmac_con4[1:0]
grf_drtype_ch1	0x1	sgrf_dmac_con4[3:2]
grf_drtype_ch2	0x1	sgrf_dmac_con4[5:4]
grf_drtype_ch3	0x1	sgrf_dmac_con4[7:6]
grf_drtype_ch4	0x1	sgrf_dmac_con4[9:8]
grf_drtype_ch5	0x1	sgrf_dmac_con4[11:10]
grf_drtype_ch6	0x1	sgrf_dmac_con4[13:12]
grf_drtype_ch7	0x1	sgrf_dmac_con4[15:14]
grf_drtype_ch8	0x1	sgrf_dmac_con5[1:0]
grf_drtype_ch9	0x1	sgrf_dmac_con5[3:2]
grf_drtype_ch10	0x1	sgrf_dmac_con5[5:4]
grf_drtype_ch11	0x1	sgrf_dmac_con5[7:6]
grf_drtype_ch12	0x1	sgrf_dmac_con5[9:8]
grf_drtype_ch13	0x1	sgrf_dmac_con5[11:10]
grf_drtype_ch14	0x1	sgrf_dmac_con5[13:12]
grf_drtype_ch15	0x1	sgrf_dmac_con5[15:14]
grf_drtype_ch16	0x1	sgrf_dmac_con6[1:0]
grf_drtype_ch17	0x1	sgrf_dmac_con6[3:2]
grf_drtype_ch18	0x1	sgrf_dmac_con6[5:4]
grf_drtype_ch19	0x1	sgrf_dmac_con6[7:6]
grf_drtype_ch20	0x1	sgrf_dmac_con6[9:8]
grf_drtype_ch21	0x1	sgrf_dmac_con6[11:10]
grf_drtype_ch22	0x1	sgrf_dmac_con6[13:12]
grf_drtype_ch23	0x1	sgrf_dmac_con6[15:14]
grf_drtype_ch24	0x1	sgrf_dmac_con7[1:0]
grf_drtype_ch25	0x1	sgrf_dmac_con7[3:2]
grf_drtype_ch26	0x1	sgrf_dmac_con7[5:4]
grf_drtype_ch27	0x1	sgrf_dmac_con7[7:6]
grf_drtype_ch28	0x1	sgrf_dmac_con7[9:8]
grf_drtype_ch29	0x1	sgrf_dmac_con7[11:10]
grf_drtype_ch30	0x1	sgrf_dmac_con7[13:12]
grf_drtype_ch31	0x1	sgrf_dmac_con7[15:14]
grf_modify_dis_ch0	0x0	sgrf_dmac_con8[0]
grf_modify_dis_ch1	0x0	sgrf_dmac_con8[1]
grf_modify_dis_ch2	0x0	sgrf_dmac_con8[2]

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Interface	Reset value	Control source
grf_modify_dis_ch3	0x0	sgrf_dmac_con8[3]
grf_modify_dis_ch4	0x0	sgrf_dmac_con8[4]
grf_modify_dis_ch5	0x0	sgrf_dmac_con8[5]
grf_modify_dis_ch6	0x0	sgrf_dmac_con8[6]
grf_modify_dis_ch7	0x0	sgrf_dmac_con8[7]
grf_modify_dis_ch8	0x0	sgrf_dmac_con8[8]
grf_modify_dis_ch9	0x0	sgrf_dmac_con8[9]
grf_modify_dis_ch10	0x0	sgrf_dmac_con8[10]
grf_modify_dis_ch11	0x0	sgrf_dmac_con8[11]
grf_modify_dis_ch12	0x0	sgrf_dmac_con8[12]
grf_modify_dis_ch13	0x0	sgrf_dmac_con8[13]
grf_modify_dis_ch14	0x0	sgrf_dmac_con8[14]
grf_modify_dis_ch15	0x0	sgrf_dmac_con8[15]
grf_modify_dis_ch16	0x0	sgrf_dmac_con9[0]
grf_modify_dis_ch17	0x0	sgrf_dmac_con9[1]
grf_modify_dis_ch18	0x0	sgrf_dmac_con9[2]
grf_modify_dis_ch19	0x0	sgrf_dmac_con9[3]
grf_modify_dis_ch20	0x0	sgrf_dmac_con9[4]
grf_modify_dis_ch21	0x0	sgrf_dmac_con9[5]
grf_modify_dis_ch22	0x0	sgrf_dmac_con9[6]
grf_modify_dis_ch23	0x0	sgrf_dmac_con9[7]
grf_modify_dis_ch24	0x0	sgrf_dmac_con9[8]
grf_modify_dis_ch25	0x0	sgrf_dmac_con9[9]
grf_modify_dis_ch26	0x0	sgrf_dmac_con9[10]
grf_modify_dis_ch27	0x0	sgrf_dmac_con9[11]
grf_modify_dis_ch28	0x0	sgrf_dmac_con9[12]
grf_modify_dis_ch29	0x0	sgrf_dmac_con9[13]
grf_modify_dis_ch30	0x0	sgrf_dmac_con9[14]
grf_modify_dis_ch31	0x0	sgrf_dmac_con9[15]

boot_manager_ns
When the DMAC exits from reset, this signal controls the security state of the DMA manager thread:
0 = assigns DMA manager to the Secure state
1 = assigns DMA manager to the Non-secure state.
boot_irq_ns
Controls the security state of an event-interrupt resource, when the DMAC exits from reset:
boot_irq_ns[x] is LOW
The DMAC assigns event<x> or irq[x] to the Secure state.
boot_irq_ns[x] is HIGH
The DMAC assigns event<x> or irq[x] to the Non-secure state.
boot_periph_ns
Controls the security state of a peripheral request interface, when the DMAC exits from reset:
boot_periph_ns[x] is LOW
The DMAC assigns peripheral request interface x to the Secure state.
boot_periph_ns[x] is HIGH

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

The DMAC assigns peripheral request interface x to the Non-secure state.
grf_drtype_<x>
The DMAC sets the state of the request_type flag:
grf_drtype_<x>[1:0]=b00: request_type<x> = Single.
grf_drtype_<x>[1:0]=b01: request_type<x> = Burst.

6.7 Application Notes

6.7.1 Using the APB slave interfaces

You must ensure that you use the appropriate APB interface, depending on the security state in which the boot_manager_ns initializes the DMAC to operate. For example, if the DMAC is in the secure state, you must issue the instruction by using the secure APB interface, otherwise the DMAC ignores the instruction. You can use the secure APB interface, or the non-secure APB interface, to start or restart a DMA channel when the DMAC is in the Non-secure state.
The necessary steps to start a DMA channel thread using the debug instruction registers as following:
1. Create a program for the DMA channel.
2. Store the program in a region of system memory.
3. Poll the DBGSTATUS Register to ensure that debug is idle, that is, the dbgstatus bit is 0.
4. Write to the DBGINST0 Register and enter the:
   - Instruction byte 0 encoding for DMAGO.
   - Instruction byte 1 encoding for DMAGO.
   - Debug thread bit to 0. This selects the DMA manager thread.
5. Write to the DBGINST1 Register with the DMAGO instruction byte [5:2] data, see Debug Instruction-1 Register o. You must set these four bytes to the address of the first instruction in the program, that was written to system memory in step 2.
6. Writing zero to the DBGCMD Register. The DMAC starts the DMA channel thread and sets the dbgstatus bit to 1.

6.7.2 Security usage

- **DMA manager thread is in the secure state**

If the DNS bit is 0, the DMA manager thread operates in the secure state and it only performs secure instruction fetches. When a DMA manager thread in the secure state processes:

- **DMAGO**

  It uses the status of the ns bit, to set the security state of the DMA channel thread by writing to the CNS bit for that channel.
- **DMAWFE**

  It halts execution of the thread until the event occurs. When the event occurs, the DMAC continues execution of the thread, irrespective of the security state of the corresponding INS bit.
- **DMASEV**

  It sets the corresponding bit in the INT_EVENT_RIS Register, irrespective of the security state of the corresponding INS bit.

- **DMA manager thread is in the Non-secure state**

If the DNS bit is 1, the DMA manager thread operates in the Non-secure state, and it only performs non-secure instruction fetches. When a DMA manager thread in the Non-secure state processes:

- **DMAGO**

  The DMAC uses the status of the ns bit, to control if it starts a DMA channel thread. If:
  ns = 0 The DMAC does not start a DMA channel thread and instead it:
  1. Executes a NOP.
  2. Sets the FSRD Register, see Fault Status DMA Manager
  3. Sets the dmago_err bit in the FTRD Register, see Fault Type DMA Manager Register.
  4. Moves the DMA manager to the Faulting state.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

ns = 1
The DMAC starts a DMA channel thread in the Non-secure state and programs the CNS bit to be non-secure.

- **DMAWFE**

The DMAC uses the status of the corresponding INS bit, in the CR3 Register, to control if it waits for the event. If:
INS = 0
The event is in the Secure state. The DMAC:
1. Executes a NOP.
2. Sets the FSRD Register, see Fault Status DMA Manager Register.
3. Sets the mgr_evnt_err bit in the FTRD Register, see Fault Type DMA Manager Register.
4. Moves the DMA manager to the Faulting state.
INS = 1
The event is in the Non-secure state. The DMAC halts execution of the thread and waits for the event to occur.

- **DMASEV**

The DMAC uses the status of the corresponding INS bit, in the CR3Register, to control if it creates the event-interrupt. If:
INS = 0
The event-interrupt resource is in the secure state. The DMAC:
1. Executes a NOP.
2. Sets the FSRD Register, see Fault Status DMA Manager Register.
3. Sets the mgr_evnt_err bit in the FTRD Register, see Fault Type DMA Manager Register.
4. Moves the DMA manager to the Faulting state.
INS = 1
The event-interrupt resource is in the Non-secure state. The DMAC creates the event-interrupt.

- **DMA channel thread is in the secure state**

When the CNS bit is 0, the DMA channel thread is programmed to operate in the Secure state and it only performs secure instruction fetches.
When a DMA channel thread in the secure state processes the following instructions:

- **DMAWFE**

The DMAC halts execution of the thread until the event occurs. When the event occurs, the DMAC continues execution of the thread, irrespective of the security state of the corresponding INS bit, in the CR3 Register.

- **DMASEV**

The DMAC creates the event-interrupt, irrespective of the security state of the corresponding INS bit, in the CR3 Register.

- **DMAWFP**

The DMAC halts execution of the thread until the peripheral signals a DMA request. When this occurs, the DMAC continues execution of the thread, irrespective of the security state of the corresponding PNS bit, in the CR4 Register.

- **DMALDP, DMASTP**

The DMAC sends a message to the peripheral to communicate that data transfer is complete, irrespective of the security state of the corresponding PNS bit, in the CR4 Register.

- **DMAFLUSHP**

The DMAC clears the state of the peripheral and sends a message to the peripheral to resend its level status, irrespective of the security state of the corresponding PNS bit, in the CR4 Register.
When a DMA channel thread is in the Secure state, it enables the DMAC to perform secure and non-secure AXI accesses

- **DMA channel thread is in the Non-secure state**

When the CNS bit is 1, the DMA channel thread is programmed to operate in the Non-secure state and it only performs non-secure instruction fetches.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

When a DMA channel thread in the Non-secure state processes the following instructions:

- **DMAWFE**

The DMAC uses the status of the corresponding INS bit, in the CR3 Register, to control if it waits for the event. If:
INS = 0
The event is in the Secure state. The DMAC:
1. Executes a NOP.
2. Sets the appropriate bit in the FSRC Register that corresponds to the DMA channel number. See Fault Status DMA Channel Register.
3. Sets the ch_evnt_err bit in the FTRn Register, see Fault Type DMA Channel Registers.
4. Moves the DMA channel to the Faulting completing state.
INS = 1
The event is in the Non-secure state. The DMAC halts execution of the thread and waits for the event to occur.

- **DMASEV**

The DMAC uses the status of the corresponding INS bit, in the CR3 Register, to control if it creates the event. If:
INS = 0
The event-interrupt resource is in the Secure state. The DMAC:
1. Executes a NOP.
2. Sets the appropriate bit in the FSRC Register that corresponds to the DMA channel number. See Fault Status DMA Channel Register.
3. Sets the ch_evnt_err bit in the FTRn Register, see Fault Type DMA Channel Registers .
4. Moves the DMA channel to the Faulting completing state.
INS = 1
The event-interrupt resource is in the Non-secure state. The DMAC creates the event-interrupt.

- **DMAWFP**

The DMAC uses the status of the corresponding PNS bit, in the CR4 Register, to control if it waits for the peripheral to signal a request. If:
PNS = 0
The peripheral is in the Secure state. The DMAC:
1. Executes a NOP.
2. Sets the appropriate bit in the FSRC Register that corresponds to the DMA channel number. See Fault Status DMA Channel Register.
3. Sets the ch_periph_err bit in the FTRn Register, see Fault Type DMA Channel Registers.
4. Moves the DMA channel to the Faulting completing state.
PNS = 1
The peripheral is in the Non-secure state. The DMAC halts execution of the thread and waits for the peripheral to signal a request.

- **DMALDP, DMASTP**

The DMAC uses the status of the corresponding PNS bit, in the CR4 Register, to control if it sends an acknowledgement to the peripheral. If:
PNS = 0
The peripheral is in the secure state. The DMAC:
1. Executes a NOP.
2. Sets the appropriate bit in the FSRC Register that corresponds to the DMA channel number. See Fault Status DMA Channel Register.
3. Sets the ch_periph_err bit in the FTRn Register, see Fault Type DMA Channel Registers.
4. Moves the DMA channel to the Faulting completing state.
PNS = 1
The peripheral is in the Non-secure state. The DMAC sends a message to the peripheral to communicate when the data transfer is complete.

- **DMAFLUSHP**

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

The DMAC uses the status of the corresponding PNS bit, in the CR4 Register, to control if it sends a flush request to the peripheral. If:
PNS = 0
The peripheral is in the secure state. The DMAC:
1. Executes a NOP.
2. Sets the appropriate bit in the FSRC Register that corresponds to the DMA channel number. See Fault Status DMA Channel Register.
3. Sets the ch_periph_err bit in the FTRn Register, see Fault Type DMA Channel Registers.
4. Moves the DMA channel to the Faulting completing state.
PNS = 1
The peripheral is in the Non-secure state. The DMAC clears the state of the peripheral and sends a message to the peripheral to resend its level status.
When a DMA channel thread is in the Non-secure state, and a DMAMOV CCR instruction attempts to program the channel to perform a secure AXI transaction, the DMAC:
1. Executes a DMANOP.
2. Sets the appropriate bit in the FSRC Register that corresponds to the DMA channel number. See Fault Status DMA Channel Register.
3. Sets the ch_rdwr_err bit in the FTRn Register, see Fault Type DMA Channel Registers.
4. Moves the DMA channel thread to the Faulting completing state.

6.7.3 Programming restrictions

Fixed unaligned bursts

The DMAC does not support fixed unaligned bursts. If you program the following conditions, the DMAC treats this as a programming error:
Unaligned read

- src_inc field is 0 in the CCRn Register
- the SARn Register contains an address that is not aligned to the size of data that the src_burst_size field contain

Unaligned write

- dst_inc field is 0 in the CCRn Register
- the DARn Register contains an address that is not aligned to the size of data that the dst_burst_size field contains

Endian swap size restrictions

If you program the endian_swap_size field in the CCRn Register, to enable a DMA channel to perform an endian swap then you must set the corresponding SARn Register and the corresponding DARn Register to contain an address that is aligned to the value that the endian_swap_size field contains.

Updating DMA channel control registers during a DMA cyclerestrictions

Prior to the DMAC executing a sequence of DMALD and DMAST instructions, the values you program in to the CCRn Register, SARn Register, and DARn Register control the data byte lane manipulation that the DMAC performs when it transfers the data from the source address to the destination address. You'd better not update these registers during a DMA cycle.

Resource sharing between DMA channels

DMA channel programs share the MFIFO data storage resource. You must not start a set ofconcurrently running DMA channel programs with a resource requirement that exceeds theconfigured size of the MFIFO. If you exceed this limit then the DMAC might lock up and generate a Watchdog abort.

6.7.4 Unaligned transfers may be corrupted

For a configuration with more than one channel, if any of channels 1 to 7 is performing transfers between certain types of misaligned source and destination addresses, then the output data may be corrupted by the action of channel 0.
Data corruption might occur if all of the following are true:
1. Two beats of AXI read data are received for one of channels 1 to 7.
2. Source and destination address alignments mean that each read data beat is

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

splited across two lines in the data buffer (see Splitting data, below).
3. There is one idle cycle between the two read data beats.
4. Channel 0 performs an operation that updates channel control information during this idle cycle (see Updates to channel control information, below)

Splitting data

Depending upon the programmed values for the DMA transfer, one beat of read data from the AXI interface need to be splited across two lines in the internal data buffer. This occurs when the read data beat contains data bytes which will be written to addresses that wrap around at the AXI interface data width, so that these bytes could not be transferred by a single AXI write data beat of the full interface width.
Most applications of DMA-330 do not split data in this way, so are NOT vulnerable to data corruption from thisdefect.
The following cases are NOT vulnerable to data corruption because they do not split data:

- Byte lane offset between source and destination addresses is 0 when source and destination addresses have the same byte lane alignment, the offset is 0 and a wrap operation that splits data cannot occur.
- Byte lane offset between source and destination addresses is a multiple of source size

Table 6-3Source size in CCRn

Source size in CCRn	Allowed offset between SARn and DARn
SS8	any offset allowed.
SS16	0,2,4,6,8,10,12,14
SS32	0,4,8,12
SS64	0,8

6.7.5 Interrupt shares between channel

As the DMAC does not record which channel (or list of channels) have asserted an interrupt. So it will depend on your program and whether any of the visible information for that program can be used to determine progress, and help identify the interrupt source. There are 4 likely information sources that can be used to determine the progress made by a program:
- Program counter (PC)
- Source address
- Destination address
- Loop counters (LC)

For example, a program might emit an interrupt each time that it iterates around a loop. In this case, the interrupt service routine (ISR) would need to store the loop value of each channel when it is called, and then compare against the new value when it is next called. A change in value would indicate that the program has progressed.
The ISR must be carefully written to ensure that no interrupts are lost. The sequence of operations is as follows:
1. Disable interrupts
2. Immediately clear the interrupt in DMA-330
3. Check the relevant registers for both channels to determine which must be serviced
4. Take appropriate action for the channels
5. Re-enable interrupts and exit ISR

6.7.6 Instruction sets

Table 6-4 DMAC Instruction sets

Mnemonic	Instruction	Thread usage
DMAADDH	Add Halfword	C
DMAEND	End	M/C
DMAFLUSHP	Flush and notify Peripheral	C
DMAGO	Go	M
DMAKILL	Kill	C
DMALD	Load	C
DMALDP	Load Peripheral	C
DMALP	Loop	C

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Mnemonic	Instruction	Thread usage
DMALPEND	Loop End	C
DMALPFE	Loop Forever	C
DMAMOV	Move	C
DMANOP	No operation	M/C
DMARMB	Read Memory Barrier	C
DMASEV	Send Event	M/C
DMAST	Store	C
DMASTP	Store and notify Peripheral	C
DMASTZ	Store Zero	C
DMAWFE	Wait For Event M	M/C
DMAWFP	Wait For Peripheral	C
DMAWMB	Write Memory Barrier	C
DMAADNH	Add Negative Halfword	C

Notes:Thread usage:C=DMA channel, *M=DMA manager*

6.7.7 Assembler directives

In this document, only DMMADNH instruction is took as an example to show the way the instruction assembled. For the other instructions, please refer to pl330_trm.pdf.

DMAADNH

Add Negative Halfword adds an immediate negative 16-bit value to the SARn Register or DARn Register, for the DMA channel thread. This enables the DMAC to support 2D DMA operations, or reading or writing an area of memory in adifferent order to naturally incrementing addresses. See Source Address Registers and Destination Address Registers. The immediate unsigned 16-bit value is one-extended to 32 bits, to create a value that is the two's complement representation of a negative number between -65536 and -1, before the DMAC adds it to the address using 32-bit addition. The DMAC discards the carry bit so that addresses wrap from 0xFFFFFFFF to 0x00000000. The net effect is to subtract between 65536 and 1 from the current value in the Source or Destination Address Register.
Following table shows the instruction encoding.

Table 6-5 DMAC instruction encoding

Imm[15:8]	Imm[7:0]	0	1	0	1	1	1	ra	0

Assembler syntax

DMAADNH <address_register>, <16-bit immediate>
where:
<address_register>
Selects the address register to use. It must be either:
SAR
SARn Register and sets ra to 0.
DAR
DARn Register and sets ra to 1.
<16-bit immediate>
The immediate value to be added to the <address_register>.
You should specify the 16-bit immediate as the number that is to be represented in the instruction encoding. For example, DMAADNH DAR, 0xFFF0 causes the value 0xFFFFFFF0 to be added to the current value of the Destination Address Register, effectively subtracting 16 from the DAR.
You can only use this instruction in a DMA channel thread.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 7 Generic Interrupt Controller (GIC)

7.1 Overview

The GIC-500 in RK1808 provides registers for managing interrupt sources, interrupt behavior, and interrupt routing to one or more cores.
The configuration of GIC-500 is shown below:

Table 7-1 GIC-500 configuration

Configuration item	Value
num_clusters	1
gic_num_rid_bits	4
num_spis	256
disable_security	false
gic_num_wid_bits	4
lpi_support	true
cpus_per_clsuter_0	2
are_option	false
lpi_size	256
did_size	16

The GIC-500 in RK1808 supports following feature:

- Support 1 cluster
- Support cluster 0 with 2 CPUs
- The following interrupt types:
  - Locality-specific Peripheral Interrupts (LPIs). These interrupts are generated by a peripheral writing to a memory-mapped register in the GIC-500. See Configurable options for the GIC-500 RTL on page 1-9
  - 256 Shared Peripheral Interrupts (SPIs)
  - 16 Private Peripheral Interrupts (PPIs), that are independent for each core and can be programmed to support either edge-triggered or level-sensitive interrupts
  - 16 SGIs, that are generated either by using software to write to GICD_SGIR or through the GIC CPU interface of a core
- Interrupt Translation Service (ITS). This provides device isolation and ID translation for message-based interrupts, which allows virtual machines to program devices directly
- Memory-mapped access to all registers
- Interrupt masking and prioritization
- Programmable interrupt routing that is based on affinity
- Three different interrupt groups, which allow interrupts to target different Exception levels:
  - Group 0
  - Non-secure Group 1
  - Secure Group 1
- A global Disable Security (DS) bit. This allows support for systems with and without security
- 32 priority values, five bits for each interrupt

7.2 Block Diagram

The GIC-500 in the RK1808 is connected with CPU cluster through AXI Stream bus, as shown below:

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

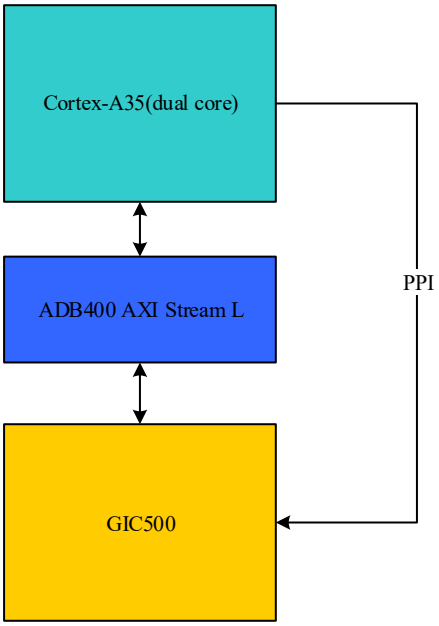


Fig. 7-1 Block Diagram

7.3 Function Description

Please refer to the document ARM_GIC-500_r1p1-00rel0_Technical_Reference_Manual.pdf for the detail function description.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 8 Power Management Unit (PMU)

8.1 Overview

In order to meet low power requirements, a power management unit (PMU) is designed for controlling power resources in RK1808. The RK1808 PMU is dedicated for managing the power of the whole chip.

8.1.1 Features

- Support 4 voltage domains: VD_CORE, VD_NPU, VD_LOGIC, VD_PMU
- Support power off VD_CORE and PD_SCU, VD_NPU, VD_LOGIC
- 2 Power domains in VD_CORE:PD_CPU0/1
- PD_CPU0/1 support cpu auto power down ， support SCU auto power down
- power domains in VD_LOGIC include PD_VPU, PD_VIO, PD_PCIE, PD_DDR
- Support wakeup source timer_int, usb, sdmmc, sdio, gpio0_int, timeout,32bit gpio, uart0, software and arm_int
- Support DDR self-refresh, auto-gating and retention
- Support Power down PLLs(A/D/C/G/N,PPLL), disable OSC, PMU clock switch to 32K(from IO or PVTM or CRU)
- Support BUS/CORE/DDRPHY clock auto gating
- Support Flush L2 by software and hardware
- Support idle request for each power domain
- Support PMU debug through IO or UART interface
- Support wakeup reset

8.2 Block Diagram

8.2.1 Voltage partition

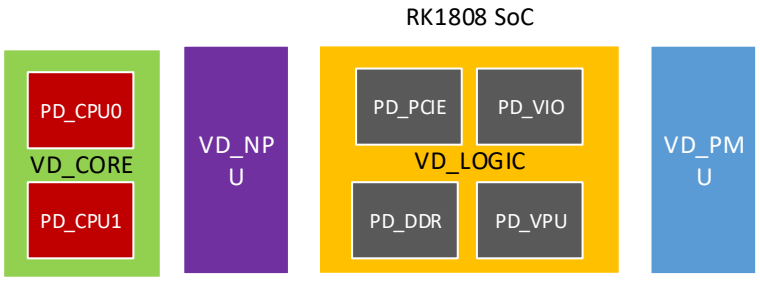


Fig. 8-1 Power Domain Partition

The above diagram describes voltage domain and power domain partition.

Table 8-1 Power Domain and Voltage Domain Summary

Voltage Domain	Blocks	Description
VD_CORE	PD_CPU0	CPU Core 0 with NEON and FPU
	PD_CPU1	CPU Core 1 with NEON and FPU
	PD_SCU	DAP Lite, SCU and 256KB L2
VD_NPU	PD_NPU	NPU
VD_LOGIC	PD_VIO	DSI HOST, CSI2TX, CSI HOST, VOP_LITE, VOP_RAW, RGA
	PD_PCIE	USB3, USB2, PCIE
	PD_VPU	VPU
	PD_GIC(Not real power domain)	GIC500
	PD_DDR	DDR_CTRL, DDR_GRF, DDR_STDBY and DDR_MONITOR
	PD_SYSTEM_	SYSTEM_SRAM

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Voltage Domain	Blocks	Description
	SRAM(Not real power domain)	
	PD_DDR	UPCTL2, DFIMON, Memory Scheduler
	PD_MMC_SFC(Not real power domain)	EMMC, SFC
	PD_SD_GMAC(Not real power domain)	SDMMC, SDIO, GMAC
	PD_AUDIO(Not real power domain)	I2S0,I2S1, PDM ,VAD
	PD_BUS(Not real power domain)	TSADC, SARADC, EFUSE, GRF, I2C1-5, TIMER0-5, STIMER0-1, WDTNS, WDTS, GPIO1-4, SPI0-2, UART1-7, PWM0-2, BootRom, CRYPTO, DMAC, DCF
	ALIVE(Not real power domain)	CRU, USBGRF, PLL(ADCGN)
VD_PMU	PD_PMU	PMU, UART0, GPIO0, PMU_GRF, PMU_SRAM, PVTM, PMUCRU, I2C0

8.2.2 PMU block diagram

The following figure is the PMU block diagram. The PMU includes the 3 following sections:

- APB interface and register, which can accept the system configuration
- Low Power State Control, which generate low power control signals.
- Power Switch Control, which control all power domain switch

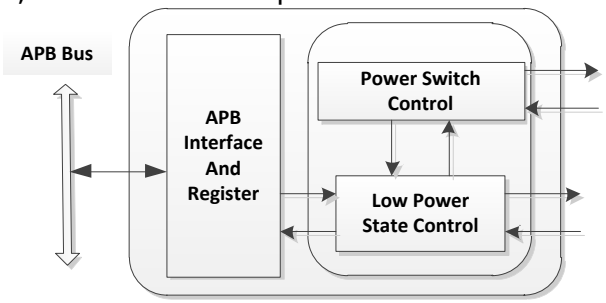


Fig. 8-2 PMU Bock Diagram

8.3 Function Description

First of all, we define two operation modes of PMU, normal mode and low power mode. When operating at normal mode, that means software can manage power sources directly by accessing PMU register.

For example, Cortex-A35 CPU can write PMU_PWRDN_CON register to determine that power off/on which power domain independently.

When operating at low power mode, software manages power sources indirectly through FSM (Finite States Machine) in PMU and those settings always not take effect immediately. That means software also can configure PMU registers to power down/up some power resources, but these setting will not be executed immediately after configuration. They will delay to execute after FSM running in particular phase.

To enter low power mode, after setting some power configurations, the

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

PMU_POWER_MODE[0] bit must be set 1 to enable PMU FSM. Then Cortex-A35 CPU needs to execute a WFI command to perform ready signal. After PMU detects all Cortex-A35 CPUs in WFI status, then the FSM will be fetched. And the specific power sources will be controlled during specific status in FSM. So the low power mode is a "delay affect" way to handle power sources inside the RK1808.

8.4 Register Description

Slave address can be divided into different length for different usage, which is shown as follows.

8.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
PMU_PMU_WAKEUP_CFG0_LO	0x0000	W	0x00000000	Wakeup configure register low 16 bit
PMU_PMU_WAKEUP_CFG0_HI	0x0004	W	0x00000000	Wakeup configure register high 16 bit
PMU_PMU_WAKEUP_CFG1_LO	0x0008	W	0x00000000	Wakeup configure register low 16 bit
PMU_PMU_WAKEUP_CFG1_HI	0x000c	W	0x00000000	Wakeup configure register high 16 bit
PMU_PMU_WAKEUP_CFG2_LO	0x0010	W	0x00000000	Wakeup configure register low 16 bit
PMU_PMU_PWRDN_CON_LO	0x0018	W	0x00000000	Power down control low 16 bit
PMU_PMU_PWRDN_CON_HI	0x001c	W	0x00000000	Power down control high 16 bit
PMU_PMU_PWRDN_ST	0x0020	W	0x00000000	Power Down Status
PMU_PMU_PWRMODE_CORE_CON_LO	0x0024	W	0x00000000	PMU power mode core config low 16 bit
PMU_PMU_PWRMODE_CORE_CON_HI	0x0028	W	0x00000000	PMU power mode core config high 16 bit
PMU_PMU_PWRMODE_COMMON_CON_LO	0x002c	W	0x00000000	Power mode common configure low 16 bit.
PMU_PMU_PWRMODE_COMMON_CON_HI	0x0030	W	0x00000000	Power mode common configure high 16 bit.
PMU_PMU_SFT_CON_LO	0x0034	W	0x00000000	PMU software configure low 16 bit.
PMU_PMU_SFT_CON_HI	0x0038	W	0x00000000	PMU software configure high 16 bit.
PMU_PMU_INT_ST	0x0044	W	0x00000000	PMU interrupt status register.
PMU_PMU_GPIO_POS_INT_ST	0x0058	W	0x00000000	PMU GPIO posedge interrupt status
PMU_PMU_GPIO_NEG_INT_ST	0x005c	W	0x00000000	GPIO0 negedge interrupt status

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
PMU_PMU_CORE_PWR_ST	0x0060	W	0x00000000	PMU Core power status
PMU_PMU_BUS_IDLE_REQ_LO	0x0064	W	0x00000000	Bus idle request.
PMU_PMU_BUS_IDLE_REQ_HI	0x0068	W	0x00000000	Bus idle request.
PMU_PMU_BUS_IDLE_ST	0x006c	W	0x00000000	Bus idle status register.
PMU_PMU_BUS_IDLE_ST1	0x0070	W	0x00000000	Bus idle status register1.
PMU_PMU_OSC_CNT_LO	0x0078	W	0x00000000	PMU OSC count low 16 bit.
PMU_PMU_OSC_CNT_HI	0x007c	W	0x00000000	PMU OSC count high 4 bit.
PMU_PMU_PLLLOCK_CNT_LO	0x0080	W	0x00000000	PMU PLL lock count low 16 bit.
PMU_PMU_PLLLOCK_CNT_HI	0x0084	W	0x00000000	PMU PLL lock count high 4 bit.
PMU_PMU_PLLRST_CNT_LO	0x0088	W	0x00000000	PMU PLL reset count low 16 bit.
PMU_PMU_PLLRST_CNT_HI	0x008c	W	0x00000000	PMU PLL reset count high 4 bit.
PMU_PMU_STABLE_CNT_LO	0x0090	W	0x00000000	PMU PMIC stable count low 16 bit.
PMU_PMU_STABLE_CNT_HI	0x0094	W	0x00000000	PMU PMIC stable count high 4 bit.
PMU_PMU_WAKEUP_RST_CLR_CNT_LO	0x00a0	W	0x00000000	PMU wakeup reset clear count low 16 bit.
PMU_PMU_WAKEUP_RST_CLR_CNT_HI	0x00a4	W	0x00000000	PMU wakeup reset clear count high 4 bit.
PMU_PMU_DDR_SREF_ST	0x00a8	W	0x00000000	DDR self-refresh status
PMU_PMU_SYS_REG0_LO	0x00ac	W	0x00000000	PMU system register0 low 16 bits.
PMU_PMU_SYS_REG0_HI	0x00b0	W	0x00000000	PMU system register0 high 16 bits.
PMU_PMU_SYS_REG1_LO	0x00b4	W	0x00000000	PMU system register1 low 16 bits.
PMU_PMU_SYS_REG1_HI	0x00b8	W	0x00000000	PMU system register high 16 bits.
PMU_PMU_SYS_REG2_LO	0x00bc	W	0x00000000	PMU system register2 low 16 bits.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
PMU_PMU_SYS_REG2_HI	0x00c0	W	0x00000000	PMU system register2 high 16 bits.
PMU_PMU_SYS_REG3_LO	0x00c4	W	0x00000000	PMU system register3 low 16 bits.
PMU_PMU_SYS_REG3_HI	0x00c8	W	0x00000000	PMU system register 3 high 16 bits.
PMU_PMU_SCU_NPU_PWRDN_CNT_LO	0x00cc	W	0x00000000	Determine how much time for waiting scu or npu power down, low 16 bit.
PMU_PMU_SCU_NPU_PWRDN_CNT_HI	0x00d0	W	0x00000000	Determine how much time for waiting scu or npu power down, high 4 bit.
PMU_PMU_SCU_NPU_PWRUP_CNT_LO	0x00d4	W	0x00000000	Determine how much time for waiting scu or npu power up, low 16 bit.
PMU_PMU_SCU_NPU_PWRUP_CNT_HI	0x00d8	W	0x00000000	Determine how much time for waiting scu or npu power up, high 4 bit.
PMU_PMU_TIMEOUT_CNT_LO	0x00dc	W	0x00000000	Determine how much time for waiting PMU timeout, low 16 bit.
PMU_PMU_TIMEOUT_CNT_HI	0x00e0	W	0x00000000	Determine how much time for waiting PMU timeout, high 4 bit.
PMU_PMU_CPU0APM_CON	0x00e4	W	0x00000000	Auto power down register for cpu0.
PMU_PMU_CPU1APM_CON	0x00e8	W	0x00000000	Auto power down register for cpu1.
PMU_PMU_INFO_TX_CON	0x00f4	W	0x00000000	TX information configure.

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

8.4.2 Detail Register Description

PMU_PMU_WAKEUP_CFG0_LO

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0000	wakeup_gpio_pos_en_lo bit 0-7: gpio0a posedge wakeup enable bit 8-16: gpio0b posedge wakeup enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

PMU_PMU_WAKEUP_CFG0_HI

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0000	wakeup_gpio_pos_en_hi bit 0-7: gpio0c posedge wakeup enable bit 8-16: gpio0d posedge wakeup enable

PMU_PMU_WAKEUP_CFG1_LO

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0000	wakeup_gpio_pos_en_lo bit 0-7: gpio0a negedge wakeup enable bit 8-16: gpio0b negedge wakeup enable

PMU_PMU_WAKEUP_CFG1_HI

Address: Operational Base + offset (0x000c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0000	wakeup_gpio_pos_en_hi bit 0-7: gpio0c negedge wakeup enable bit 8-16: gpio0d negedge wakeup enable

PMU_PMU_WAKEUP_CFG2_LO

Address: Operational Base + offset (0x0010)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:11	RO	0x0	Reserved
10	RW	0x0	wakeup_timeout_en Timeout wakeup enable. 1'b0: Disable 1'b1: Enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
9	RW	0x0	wakeup_vad_en VAD wakeup enable. 1'b0: Disable 1'b1: Enable
8	RW	0x0	wakeup_sft_en Software wakeup enable. 1'b0: Disable 1'b1: Enable
7	RW	0x0	wakeup_usbdev_en USB wakeup enable. 1'b0: Disable 1'b1: Enable
6	RW	0x0	wakeup_timer_en Timer wakeup enable. 1'b0: Disable 1'b1: Enable
5	RW	0x0	wakeup_uart0_en UART0 wakeup enable. 1'b0: Disable 1'b1: Enable
4	RW	0x0	wakeup_sdmmc_en SDMMC wakeup enable. 1'b0: Disable 1'b1: Enable
3	RW	0x0	wakeup_sdio_en SDIO wakeup enable. 1'b0: Disable 1'b1: Enable
2	RW	0x0	wakeup_gpio_int_en GPIO0 interrupt wakeup enable. 1'b0: Disable 1'b1: Enable
1	RO	0x0	Reserved
0	RW	0x0	wakeup_int_cluster_en Interrupt wakeup enable. 1'b0: Disable 1'b1: Enable

PMU_PMU_PWRDN_CON_LO

Address: Operational Base + offset (0x0018)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15	RW	0x1	vd_npu_pwrdwn_en vd_vpu power down enable. 1'b0: Disable 1'b1: Enable
14	RW	0x0	pd_vio_pwrdwn_en pd_vio power down enable 1'b0: Disable 1'b1: Enable
13	RW	0x0	pd_vpu_pwrdwn_en pd_vpu power down enable. 1'b0: Disable 1'b1: Enable
12:10	RO	0x0	Reserved
9	RW	0x0	pd_pcie_pwrdwn_en pd_pcie power down enable. 1'b0: Disable 1'b1: Enable
8	RO	0x0	Reserved
7	RW	0x0	vd_core_pwrdwn_en vd_core pwrdwn enable. 1’b0: Disable 1’b1: Enable
6	RW	0x0	pd_ddr_pwrdwn_en pd_ddr power down enable. 1’b0: Disable 1’b1: Enable
5	RO	0x0	Reserved
4	RW	0x0	pd_scu_pwrdwn_en pd_scu power down enable. 1’b0: Disable 1’b1: Enable
3:2	RO	0x0	Reserved
1	RW	0x0	cpu1_pwrdwn_en cpu1 power down enable. 1’b0: Disable 1’b1: Enable
0	RW	0x0	cpu0_pwrdwn_en cpu0 power down enable. 1’b0: Disable 1’b1: Enable

PMU_PMU_PWRDN_CON_HI

Address: Operational Base + offset (0x001c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:3	RW	0x0	Reserved
2	RW	0x0	chip_pwrdwn_en chip power down enable 1'b0: Disable 1'b1: Enable
1:0	RW	0x0	Reserved

PMU_PMU_PWRDN_ST

Address: Operational Base + offset (0x0020)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RO	0x0	Reserved
15	RW	0x1	vd_npu_pwr_stat vd_npu power status. 1'b0: power up 1'b1: power down
14	RW	0x0	pd_vio_pwr_stat pd_vio power status. 1'b0: power up 1'b1: power down
13	RW	0x0	pd_vpu_pwr_stat pd_vpu power status. 1'b0: power up 1'b1: power down
12:10	RO	0x0	Reserved
9	RW	0x0	pd_pcie_pwr_stat pd_pcie power status. 1'b0: power up 1'b1: power down
8	RO	0x0	Reserved
7	RW	0x0	vd_core_pwr_stat vd_core power status. 1'b0: power up 1'b1: power down
6	RW	0x0	pd_ddr_pwr_stat pd_ddr power status. 1'b0: power up 1'b1: power down
5	RO	0x0	Reserved
4	RW	0x0	pd_scu_pwr_stat pd_scu power status. 1'b0: power up 1'b1: power down
3:2	RO	0x0	Reserved
1	RW	0x0	cpu1_pwr_stat cpu1 power status. 1'b0: power up 1'b1: power down
0	RO	0x0	cpu0_pwr_stat cpu0 power status. 1'b0: power up 1'b1: power down

PMU_PMU_PWRMODE_CORE_CON_LO

Address: Operational Base + offset (0x0024)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RO	0x0	Reserved
14	RW	0x0	npll_pd_en Power down npll during low power mode. 1'b0: Disable 1'b1: Enable
13	RW	0x0	gpll_pd_en Power down gpll during low power mode. 1'b0: Disable 1'b1: Enable
12	RW	0x0	cpll_pd_en Power down cpll during low power mode. 1'b0: Disable 1'b1: Enable
11	RW	0x0	dpll_pd_en Power down dpll during low power mode. 1'b0: Disable 1'b1: Enable
10	RW	0x0	apll_pd_en Power down apll during low power mode. 1'b0: Disable 1'b1: Enable
9	RW	0x0	l2_flush_en Flush L2 during low power mode 1'b0: Disable 1'b1: Enable
8	RW	0x0	l2_idel_en L2 idle enable during low power mode. 1'b0: Disable 1'b1: Enable
7	RW	0x0	core_pd_en Power down VD_CORE during low power mode. 1'b0: Disable 1'b1: Enable
6	RW	0x0	scu_pd_en Power down pd_scu during low power mode. 1'b0: Disable 1'b1: Enable
5	RW	0x0	clr_core_niu clear core NIU during low power mode. 1'b0: Disable 1'b1: Enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4	RO	0x0	Reserved
3	RW	0x0	cpu0_pd_en Power down cpu0 enable. 1'b0: Disable 1'b1: Enable
2:0	RO	0x0	Reserved

PMU_PMU_PWRMODE_CORE_CON_HI

Address: Operational Base + offset (0x0028)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RO	0x0	Reserved
14	RW	0x0	ddrphy_clk_src_gate_en Gate ddrphy clock during low power mode. 1'b0: Disable 1'b1: Enable
13	RW	0x0	pwm_switch_en Enable pwm switch to gpio function during low power mode. 1'b0: Disable 1'b1: Enable
12:11	RO	0x0	Reserved
10	RW	0x0	clr_gic2core Clear gic2core axi4 stream bus. 1'b0: Disable 1'b1: Enable
9	RW	0x0	clr_core2gic Clear core2gic AXI stream bus. 1'b0: Disable 1'b1: Enable
8	RW	0x0	clr_system_sram Clear SYSTEM_SRAM NIU during low power mode. 1'b0: Disable 1'b1: Enable
7:4	RO	0x0	Reserved
3	RW	0x0	clr_bus_noc1 Clear bus_noc1 during low power mode. 1'b0: Disable 1'b1: Enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2	RW	0x0	clr_bus_noc2 Clear bus_noc2 during low power mode. 1'b0: Disable 1'b1: Enable
1	RW	0x0	ppll_pd_en Power down ppll during low power mode. 1'b0: Disable 1'b1: Enable
0	RO	0x0	Reserved

PMU_PMU_PWRMODE_COMMON_CON_LO

Address: Operational Base + offset (0x002c)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	clr_php_mid Clear php_mid NIU during low power mode. 1'b0: Disable 1'b1: Enable
14	RW	0x0	clr_pmu Clear PD_PMU NIU during low power mode. 1'b0: Disable 1'b1: Enable
13	RW	0x0	ddr_ret_de_req De-assert DDR retention request. 1'b0: Disable 1'b1: Enable
12	RW	0x0	ddr_ret_en DDR retention enable during low power mode. 1'b0: Disable 1'b1: Enable
11	RW	0x0	ddrc_gating_en Gating DDR clock (controller and phy) during low power mode. 1'b0: Disable 1'b1: Enable
10	RW	0x0	sref_enter_en DDR enter self-refresh during low power mode. 1'b0: Disable 1'b1: Enable
9	RW	0x0	input_clamp_en Clamp PD_PMU input during low power mode. 1'b0: Disable 1'b1: Enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
8	RW	0x0	osc_24m_dis Disable 24M OSC during low power mode. 1'b0: Disable 1'b1: Enable
7	RO	0x0	Reserved
6	RW	0x0	pmu_lf_ena PD_PMU switch to low frequency enable. 1'b0: Disable 1'b1: Enable
5	RO	0x0	Reserved
4	RW	0x0	pll_pd_en Power down pll during low power mode. 1'b0: Disable 1'b1: Enable
3	RW	0x0	wakeup_reset_en Wakeup reset during low power mode. 1'b0: Disable 1'b1: Enable
2	RO	0x0	Reserved
1	RW	0x0	ddr_pd_en Power down pd_ddr during low power mode. 1'b0: Disable 1'b1: Enable
0	RW	0x0	power_mode_en Power mode enable. 1'b0: Disable 1'b1: Enable

PMU_PMU_PWRMODE_COMMON_CON_HI

Address: Operational Base + offset (0x0030)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	scu_apm_en Auto power down SCU enable. 1'b0: Disable 1'b1: Enable
14	RW	0x0	scu_auto_gating_en Gating core clock when SCU auto power down. 1'b0: Disable 1'b1: Enable
13:12	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
11	RW	0x0	wait_wakeup_begin Begin to wait for wakeup. 1'b0: Disable 1'b1: Enable
10	RW	0x0	clr_pcie Clear pcie NIU when in power mode. 1'b0: Disable 1'b1: Enable
9	RW	0x0	clr_audio Clear audio NIU when in power mode. 1'b0: Disable 1'b1: Enable
8	RO	0x0	Reserved
7	RW	0x0	clr_npu Clear npu NIU when in power mode. 1'b0: Disable 1'b1: Enable
6	RW	0x0	clr_vio Clear vio NIU when in power mode. 1'b0: Disable 1'b1: Enable
5	RW	0x0	clr_vpu Clear vpu NIU when in power mode. 1'b0: Disable 1'b1: Enable
4	RW	0x0	clr_sd_gmac Clear sd_gmac NIU when in power mode.
3	RW	0x0	clr_npu_lp Clear npu low power interface when in power mode. 1'b0: Disable 1'b1: Enable
2	RW	0x0	clr_msch Clear msch NIU when in power mode. 1'b0: Disable 1'b1: Enable
1	RW	0x0	clr_mmc Clear mmc NIU when in power mode. 1'b0: Disable 1'b1: Enable
0	RW	0x0	clr_bus Clear bus NIU when power mode. 1'b0: Disable 1'b1: Enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

PMU_PMU_SFT_CON_LO

Address: Operational Base + offset (0x0034)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	osc_disable_cfg Disable OSC configure. 1’b0: Disable 1’b1: Enable
14	RW	0x0	pmu_lf_ena_cfg PD_PMU clock switch to low frequency configure. 1’b0: Disable 1’b1: Enable
13:12	RO	0x0	Reserved
11	RW	0x0	ddr_ret_cfg DDR retention configure. 1’b0: Disable 1’b1: Enable
10	RW	0x0	upctl_c_sysreq_cfg UPCTL sysreq configure. 1’b0: Disable 1’b1: Enable
9	RW	0x0	pmu_24m_ena PD_PMU clock switch to 24M configure. 1’b0: Disable 1’b1: Enable
8:7	RO	0x0	Reserved
6	RW	0x0	l2flushreq_cfg Flush L2 configure. 1’b0: Disable 1’b1: Enable
5	RW	0x0	gpll_pd_cfg Power down gpll configure. 1’b0: Disable 1’b1: Enable
4	RW	0x0	cpll_pd_cfg Power down cpll configure. 1’b0: Disable 1’b1: Enable
3	RW	0x0	dpll_pd_cfg Power down dpll configure. 1’b0: Disable 1’b1: Enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2	RW	0x0	apll_pd_cfg Power down apll configure. 1'b0: Disable 1'b1: Enable
1	RW	0x0	npll_pd_cfg Power down npll configure. 1'b0: Disable 1'b1: Enable
0	RW	0x0	wakeup_sft Software wakeup control. 1'b0: Disable 1'b1: Enable

PMU_PMU_SFT_CON_HI

Address: Operational Base + offset (0x0038)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:6	RO	0x0	Reserved
5	RW	0x0	wakeup_rst_cfg Wake up reset configure. 1'b0: Disable 1'b1: Enable
4:2	RO	0x0	Reserved
1	RW	0x0	ppll_pd_cfg Power down ppll configure. 1'b0: Disable 1'b1: Enable
0	RO	0x0	Reserved

PMU_PMU_INT_ST

Address: Operational Base + offset (0x0044)

Bit	Attr	Reset Value	Description
31	RO	0x0	Reserved
30	RW	0x0	vd_npu_pwr_switch_status Indicate whether a vd_npu power switch event happened. 1'b0: Not happened. 1'b1: Happened.
29	RW	0x0	pd_vio_pwr_switch_status Indicate whether a pd_vio power switch event happened. 1'b0: Not happened. 1'b1: Happened.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
28	RW	0x0	pd_vpu_pwr_switch_status Indicate whether a pd_vpu power switch event happened. 1'b0: Not happened. 1'b1: Happened.
27	RO	0x0	Reserved
26	RW	0x0	pd_pcie_pwr_switch_status Indicate whether a pd_pcie power switch event happened. 1'b0: Not happened. 1'b1: Happened.
25	RW	0x0	vd_core_pwr_switch_status Indicate whether a vd_core power switch event happened. 1'b0: Not happened. 1'b1: Happened.
24:21	RO	0x0	Reserved
20	RW	0x0	pd_ddr_pwr_switch_status Indicate whether a pd_ddr power switch event happened. 1'b0: Not happened. 1'b1: Happened.
19	RW	0x0	pd_scu_pwr_switch_status Indicate whether a pd_scu power switch event happened. 1'b0: Not happened. 1'b1: Happened.
18:17	RO	0x0	Reserved
16	RW	0x0	cpu1_pwr_switch_status Indicate whether a cpu1 power switch event happened. 1'b0: Not happened. 1'b1: Happened.
15	RW	0x0	cpu0_pwr_switch_status Indicate whether a cpu0 power switch event happened. 1'b0: Not happened. 1'b1: Happened.
14	RW	0x0	wakeup_sft_status Indicate whether a software wakeup happened. 1'b0: Not happened. 1'b1: Happened.
13	RW	0x0	wakeup_timeout_status Indicate whether a timeout wakeup happened. 1'b0: Not happened. 1'b1: Happened.
12	RW	0x0	wakeup_vad_status Indicate whether a vad wakeup happened. 1'b0: Not happened. 1'b1: Happened.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
11	RW	0x0	wakeup_sdmmc_status Indicate whether a sdmmc wakeup happened. 1'b0: Not happened. 1'b1: Happened.
10	RW	0x0	wakeup_sdio_status Indicate whether a sdio wakeup happened. 1'b0: Not happened. 1'b1: Happened.
9	RW	0x0	wakeup_gpio_pos_status Indicate whether a gpio posedge wakeup happened. 1'b0: Not happened. 1'b1: Happened.
8	RW	0x0	wakeup_gpio_neg_status Indicate whether a gpio negedge wakeup happened. 1'b0: Not happened. 1'b1: Happened.
7	RW	0x0	wakeup_gpio_int Indicate whether a gpio int wakeup happened. 1'b0: Not happened. 1'b1: Happened.
6	RW	0x0	wakeup_usb_status Indicate whether a usb wakeup happened. 1'b0: Not happened. 1'b1: Happened.
5	RW	0x0	wakeup_timer_status Indicate whether a timer wakeup happened. 1'b0: Not happened. 1'b1: Happened.
4	RW	0x0	wakeup_int_cluster_status Indicate whether a cpu interrupt wakeup happened. 1'b0: Not happened. 1'b1: Happened.
3	RO	0x0	Reserved
2	RW	0x0	wakeup_uart0_status Indicate whether a uart0 wake up happened. 1'b0: Not happened. 1'b1: Happened.
1	RO	0x0	pwrmode_wakeup_status Indicate whether a power mode wakeup happened. 1'b0: Not happened. 1'b1: Happened.
0	RO	0x0	Reserved

PMU_PMU_GPIO_POS_INT_ST

Address: Operational Base + offset (0x0058)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	gpio_pos_int_status GPIO0 posedge interrupt status

PMU_PMU_GPIO_NEG_INT_ST

Address: Operational Base + offset (0x005c)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	gpio_neg_int_status GPIO0 negedge interrupt status

PMU_PMU_CORE_PWR_ST

Address: Operational Base + offset (0x0060)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	Reserved
7:6	RW	0x0	standbywfi Status of WFI.
5:4	RO	0x0	Reserved
3:2	RW	0x0	standbywfe Status of WFE.
1	RW	0x0	standbywfil2 WFI L2 status. 1'b0: L2 is not idle 1'b1: L2 is idle
0	RO	0x0	l2flushdone Status of L2 flush done. 1'b0: Not finish 1'b1: Flush finish

PMU_PMU_BUS_IDLE_REQ_LO

Address: Operational Base + offset (0x0064)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15	RW	0x0	idle_req_bus_noc1_cfg Idle request of bus_noc1 NIU. 1'b0: Disable 1'b1: Enable
14	RW	0x0	idle_req_audio_cfg Idle request of PMU NIU. 1'b0: Disable 1'b1: Enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
13	RW	0x0	idle_req_pmu_cfg Idle request of PMU NIU. 1'b0: Disable 1'b1: Enable
12	RW	0x0	idle_req_peri_php Idle request of peri php NIU. 1'b0: Disable 1'b1: Enable
11	RW	0x0	idle_req_msch_cfg Idle request of msch NIU. 1'b0: Disable 1'b1: Enable
10	RO	0x0	Reserved
9	RW	0x0	idle_req_npu_lp Idle request of npu low power interface. 1'b0: Disable 1'b1: Enable
8	RW	0x0	idle_req_vio_cfg Idle request of vio NIU. 1'b0: Disable 1'b1: Enable
7	RW	0x0	idle_req_vpu_cfg Idle request of vpu NIU. 1'b0: Disable 1'b1: Enable
6	RW	0x0	idle_req_gmac_cfg Idle request of gmac NIU. 1'b0: Disable 1'b1: Enable
5	RW	0x0	idle_req_mmc_cfg Idle request of mmc NIU. 1'b0: Disable 1'b1: Enable
4	RW	0x0	idle_req_pcie_cfg Idle request of pcie NIU. 1'b0: Disable 1'b1: Enable
3	RW	0x0	idle_req_core_cfg Idle request of core NIU. 1'b0: Disable 1'b1: Enable
2	RW	0x0	idle_req_npu_cfg Idle request of npu NIU. 1'b0: Disable 1'b1: Enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
1	RW	0x0	idle_req_bus_noc2_cfg Idle request of bus_noc2 NIU. 1'b0: Disable 1'b1: Enable
0	RW	0x0	idle_req_bus_cfg Idle request of bus NIU. 1'b0: Disable 1'b1: Enable

PMU_PMU_BUS_IDLE_REQ_HI

Address: Operational Base + offset (0x0068)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:7	RO	0x0	Reserved
6	RW	0x0	idle_req_gic2core_cfg Idle request of gic2core stream bus. 1'b0: Disable 1'b1: Enable
5	RW	0x0	idle_req_core2gic_cfg Idle request of core2gic stream bus. 1'b0: Disable 1'b1: Enable
4	RW	0x0	idle_req_system_sram_cfg Idle request of SYSTEM_SRAM NIU. 1'b0: Disable 1'b1: Enable
3:0	RO	0x0	Reserved

PMU_PMU_BUS_IDLE_ST

Address: Operational Base + offset (0x006c)

Bit	Attr	Reset Value	Description
31	RO	0x0	idle_bus_noc1 Idle status of bus_noc1 NIU. 1'b0: Not idle 1'b1: Idle
30	RO	0x0	idle_audio Idle status of audio NIU. 1'b0: Not idle 1'b1: Idle

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
29	RO	0x0	idle_pmu Idle status of PMU NIU. 1'b0: Not idle 1'b1: Idle
28	RO	0x0	idle_peri_php Idle status of peri_php NIU. 1'b0: Not idle 1'b1: Idle
27	RO	0x0	idle_msch Idle status of msch NIU. 1'b0: Not idle 1'b1: Idle
26	RO	0x0	Reserved
25	RO	0x0	idle_npu_lp Idle status of npu low power interface. 1'b0: Not idle 1'b1: Idle
24	RO	0x0	idle_vio Idle status of vio NIU. 1'b0: Not idle 1'b1: Idle
23	RO	0x0	idle_vpu Idle status of vpu NIU. 1'b0: Not idle 1'b1: Idle
22	RO	0x0	idle_gmac Idle status of gmac NIU. 1'b0: Not idle 1'b1: Idle
21	RO	0x0	idle_mmc Idle status of mmc NIU. 1'b0: Not idle 1'b1: Idle
20	RO	0x0	idle_pcie Idle status of pcie NIU. 1'b0: Not idle 1'b1: Idle
19	RO	0x0	idle_core Idle status of core NIU. 1'b0: Not idle 1'b1: Idle
18	RO	0x0	idle_npu Idle status of npu NIU. 1'b0: Not idle 1'b1: Idle

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
17	RO	0x0	idle_bus_noc2 Idle status of bus_noc2 NIU. 1'b0: Not idle 1'b1: Idle
16	RO	0x0	idle_bus Idle status of bus NIU. 1'b0: Not idle 1'b1: Idle
15	RO	0x0	idle_ack_bus_noc1 Idle ack status of bus_noc1 NIU. 1’b0: Idle is not acknowledged 1’b1: Idle is acknowledged
14	RO	0x0	idle_ack_audio Idle ack status of audio NIU. 1’b0: Idle is not acknowledged 1’b1: Idle is acknowledged
13	RO	0x0	idle_ack_pmu Idle ack status of PMU NIU. 1’b0: Idle is not acknowledged 1’b1: Idle is acknowledged
12	RO	0x0	idle_ack_peri_php Idle ack status of peri php NIU. 1’b0: Idle is not acknowledged 1’b1: Idle is acknowledged
11	RO	0x0	idle_ack_msch Idle ack status of msch NIU. 1’b0: Idle is not acknowledged 1’b1: Idle is acknowledged
10	RO	0x0	Reserved
9	RO	0x0	idle_ack_npu_lp Idle ack status of npu low power interface. 1’b0: Idle is not acknowledged 1’b1: Idle is acknowledged
8	RO	0x0	idle_ack_vio Idle ack status of vio NIU. 1’b0: Idle is not acknowledged 1’b1: Idle is acknowledged
7	RO	0x0	idle_ack_vpu Idle ack status of vpu NIU. 1’b0: Idle is not acknowledged 1’b1: Idle is acknowledged
6	RO	0x0	idle_ack_gmac Idle ack status of gmac NIU. 1’b0: Idle is not acknowledged 1’b1: Idle is acknowledged

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5	RO	0x0	idle_ack_mmc Idle ack status of mmc NIU. 1'b0: Idle is not acknowledged 1'b1: Idle is acknowledged
4	RO	0x0	idle_ack_pcie Idle ack status of pcie NIU. 1'b0: Idle is not acknowledged 1'b1: Idle is acknowledged
3	RO	0x0	idle_ack_core Idle ack status of core NIU. 1'b0: Idle is not acknowledged 1'b1: Idle is acknowledged
2	RO	0x0	idle_ack_npu Idle ack status of npu NIU. 1'b0: Idle is not acknowledged 1'b1: Idle is acknowledged
1	RO	0x0	idle_ack_bus_noc2 Idle ack status of bus_noc2 NIU. 1'b0: Idle is not acknowledged 1'b1: Idle is acknowledged
0	RO	0x0	idle_ack_bus Idle ack status of bus NIU. 1'b0: Idle is not acknowledged 1'b1: Idle is acknowledged

PMU_PMU_BUS_IDLE_ST1

Address: Operational Base + offset (0x0070)

Bit	Attr	Reset Value	Description
31:23	RO	0x0	Reserved
22	RW	0x0	idle_gic2core Idle status of gic2core AXI stream interface. 1'b0: Not idle 1'b1: Idle
21	RW	0x0	idle_core2gic Idle status of core2gic AXI stream interface. 1'b0: Not idle 1'b1: Idle
20	RW	0x0	idle_system_sram Idle status of SYSTEM_SRAM NIU. 1'b0: Not idle 1'b1: Idle
19:7	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
6	RW	0x0	idle_ack_gic2core Idle ack status of gic2core AXI stream interface. 1'b0: Not idle acknowledged 1'b1: Idle acknowledged
5	RW	0x0	idle_ack_core2gic Idle ack status of core2gic AXI stream interface. 1'b0: Not idle acknowledged 1'b1: Idle acknowledged
4	RW	0x0	idle_ack_system_sram Idle ack status of SYSTEM_SRAM NIU. 1'b0: Not idle acknowledged 1'b1: Idle acknowledged
3:0	RO	0x0	Reserved

PMU_PMU_OSC_CNT_LO

Address: Operational Base + offset (0x0078)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0010	osc_cnt_lo Low 16 bit of osc count, which determine time for waiting OSC enable.

PMU_PMU_OSC_CNT_HI

Address: Operational Base + offset (0x007c)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	Reserved
19:16	WO	0x0	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:4	RO	0x0	Reserved
3:0	RW	0x0	osc_cnt_hi High 4 bit lof osc count, which determine time for waiting OSC enable.

PMU_PMU_PLLLOCK_CNT_LO

Address: Operational Base + offset (0x0080)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15:0	RW	0x0010	plllock_cnt_lo Low 16 bit of pll lock count, which determine time for waiting PLL lock.

PMU_PMU_PLLLOCK_CNT_HI

Address: Operational Base + offset (0x0084)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	Reserved
19:16	WO	0x0	write_enable Write enable for lower 4 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:4	RO	0x0	Reserved
3:0	RW	0x0	plllock_cnt_hi High 4 bit of pll lock count, which determine time for waiting pll lock.

PMU_PMU_PLLRST_CNT_LO

Address: Operational Base + offset (0x0088)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0010	pllrst_cnt_lo Low 16 bit of pll lock count, which determine time for waiting PLL reset.

PMU_PMU_PLLRST_CNT_HI

Address: Operational Base + offset (0x008c)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	Reserved
19:16	WO	0x0	write_enable Write enable for lower 4 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:4	RO	0x0	Reserved
3:0	RW	0x0	pllrst_cnt_hi High 4 bit lof pll reset count, which determine time for waiting pll reset.

PMU_PMU_STABLE_CNT_LO

Address: Operational Base + offset (0x0090)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0010	stable_cnt_lo Low 16 bit of pmic stable count, which determine time for waiting pmic stable.

PMU_PMU_STABLE_CNT_HI

Address: Operational Base + offset (0x0094)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	Reserved
19:16	WO	0x0	write_enable Write enable for lower 4 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:4	RO	0x0	Reserved
3:0	RW	0x0	stable_cnt_hi High 4 bit lof PMIC stable count, which determine time for waiting PMIC stable.

PMU_PMU_WAKEUP_RST_CLR_CNT_LO

Address: Operational Base + offset (0x00a0)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0010	wakeup_rst_clr_cnt_lo Low 16 bit of wake up reset clear count, which determine time for waiting wake up reset clear.

PMU_PMU_WAKEUP_RST_CLR_CNT_HI

Address: Operational Base + offset (0x00a4)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0000	wakeup_rst_clr_cnt_hi High 4 bit of wake up reset clear count, which determine time for waiting wake up reset clear.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

PMU_PMU_DDR_SREF_ST

Address: Operational Base + offset (0x00a8)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	Reserved
1	RO	0x0	upctl_c_sysack upctl_c_sysack status.
0	RO	0x0	upctl_c_active upctl_active status.

PMU_PMU_SYS_REG0_LO

Address: Operational Base + offset (0x00ac)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0000	sys_reg0_lo Low 16 bit of system register0.

PMU_PMU_SYS_REG0_HI

Address: Operational Base + offset (0x00b0)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0000	sys_reg0_hi High 16 bit of system register0.

PMU_PMU_SYS_REG1_LO

Address: Operational Base + offset (0x00b4)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0000	sys_reg1_lo Low 16 bit of system register0.

PMU_PMU_SYS_REG1_HI

Address: Operational Base + offset (0x00b8)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0000	sys_reg1_hi High 16 bit of system register1.

PMU_PMU_SYS_REG2_LO

Address: Operational Base + offset (0x00bc)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0000	sys_reg2_lo Low 16 bit of system register0.

PMU_PMU_SYS_REG2_HI

Address: Operational Base + offset (0x00c0)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0000	sys_reg2_hi High 16 bit of system register0.

PMU_PMU_SYS_REG3_LO

Address: Operational Base + offset (0x00c4)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0000	sys_reg3_lo Low 16 bit of system register3.

PMU_PMU_SYS_REG3_HI

Address: Operational Base + offset (0x00c8)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0000	sys_reg3_hi High 16 bit of system register0.

PMU_PMU_SCU_NPU_PWRDN_CNT_LO

Address: Operational Base + offset (0x00cc)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0000	cnt_lo Low 16 bit of count, which determine time for waiting .

PMU_PMU_SCU_NPU_PWRDN_CNT_HI

Address: Operational Base + offset (0x00d0)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	Reserved
19:16	WO	0x0	write_enable Write enable for lower 4 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:4	RO	0x0	Reserved
3:0	RW	0x0	cnt_hi High 4 bit of count, which determine time for waiting .

PMU_PMU_SCU_NPU_PWRUP_CNT_LO

Address: Operational Base + offset (0x00d4)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0000	cnt_lo Low 16 bit of count, which determine time for waiting .

PMU_PMU_SCU_NPU_PWRUP_CNT_HI

Address: Operational Base + offset (0x00d8)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:20	RO	0x0	Reserved
19:16	WO	0x0	write_enable Write enable for lower 4 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:4	RO	0x0	Reserved
3:0	RW	0x0	cnt_hi High 4 bit of count, which determine time for waiting .

PMU_PMU_TIMEOUT_CNT_LO

Address: Operational Base + offset (0x00dc)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 16 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:0	RW	0x0000	cnt_lo Low 16 bit of count, which determine time for waiting .

PMU_PMU_TIMEOUT_CNT_HI

Address: Operational Base + offset (0x00e0)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	Reserved
19:16	WO	0x0	write_enable Write enable for lower 4 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:4	RO	0x0	Reserved
3:0	RW	0x0	cnt_hi High 4 bit of count, which determine time for waiting.

PMU_PMU_CPU0APM_CON

Address: Operational Base + offset (0x00e4)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 4 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:4	RO	0x0	Reserved
3	RW	0x0	cpu_sft_wakeup Software wakeup cpu0 configure. 1'b0: Disable 1'b1: Enable
2	RO	0x0	Reserved
1	RW	0x0	int_wakeup_en CPU0 interrupt wakeup enable. 1'b0: Disable 1'b1: Enable
0	RW	0x0	wfi_pwrdn_en Enable cpu0 WFI auto power down function. 1'b0: Disable 1'b1: Enable

PMU_PMU_CPU1APM_CON

Address: Operational Base + offset (0x00e8)

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 4 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:4	RO	0x0	Reserved
3	RW	0x0	cpu_sft_wakeup Software wakeup cpu1 configure. 1'b0: Disable 1'b1: Enable
2	RO	0x0	Reserved
1	RW	0x0	int_wakeup_en CPU1 interrupt wakeup enable. 1'b0: Disable 1'b1: Enable
0	RW	0x0	wfi_pwrdn_en Enable cpu1 wif auto power down function. 1'b0: Disable 1'b1: Enable

PMU_PMU_INFO_TX_CON

Address: Operational Base + offset (0x00f4)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	WO	0x0000	write_enable Write enable for lower 4 bits, each bit is individual. 1'b0: Write access disable 1'b1: Write access enable
15:8	RW	0x0	info_tx_intv_time Control the interval time of pmu tx.
7:1	RO	0x0	Reserved
0	RW	0x0	info_tx_en Enable pmu tx(sout) function. 1'b0: Disable 1'b1: Enable

8.5 Timing Diagram

8.5.1 Each domain power switch timing

The following figure is each domain power down and power up timing.

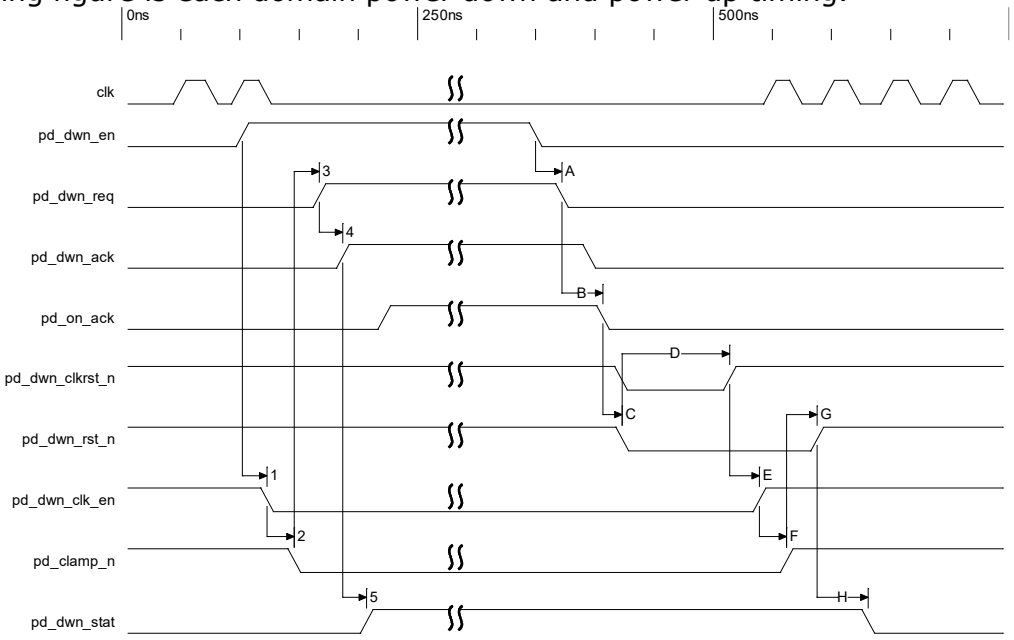


Fig. 8-3 Each Domain Power Switch Timing

8.5.2 External wakeup PAD timing

PMU supports a lot of external wakeup sources, such as SDIO/SDMMC, USBDEV, SIM detect wakeup, GPIO0 wakeup source and so on. All these external wakeup sources must meet the timing requirement (at least 200us) when the wakeup event is asserted. The following figure gives the timing information.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

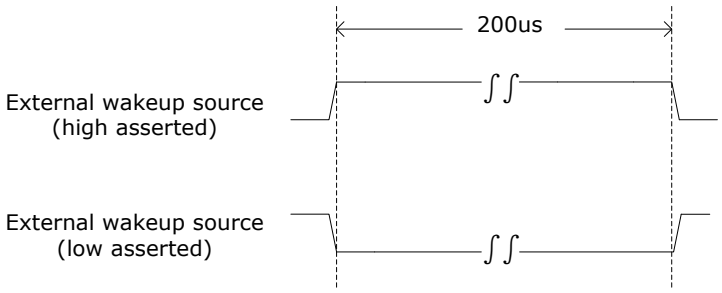


Fig. 8-4 External Wakeup Source PAD Timing

8.6 Application Note

8.6.1 Low power mode

PMU can work in the low power mode by setting bit[0] of PMU_PWRMODE_CON register. After setting this bit and all CPU cores enter WFI states, PMU low power FSM will start to run. In the low power mode, PMU will manage power resources by hardware, such as power on/off the specified power domain, send idle request to specified power domain, shut down/up PLL and so on. All of above are configurable by setting corresponding registers. All FSM power states could be monitored through IO. The following table describes all power states of PMU FSM.

Table 8-2 Low Power State

Num	States	Description
0	ST_NORMAL	Still in normal state
1	ST_CPU0_PWRDN	Power down cpu core 0
2	ST_L2_FLUSH	Flush L2
3	ST_L2_IDLE	L2 Idle
4	ST_TRANS_NO_FIN_CORE	Wait for buses in vd_core idle
5	ST_SCU_PWRDN	Power down pd_scu
6	ST_CORE_PWRDN	Power down vd_core (not really remove power, assert reset only)
7	ST_CORE_CLK_DIS	clock gating core clock
8	ST_TRANS_NO_FIN_BUS	Wait for buses in vd_logic idle
9	ST_SREF_ENTER	DDR enter self-refresh
10	ST_DDR_IO_RET	DDR IO retention
11	ST_DDR_IO_PWROFF	NOT USED
12	ST_DDR_PWRDN	Power down PD_DDR
13	ST_BUS_PWRDN	NOT USED
14	ST_PMU_LF	PD_PMU clock switch to low frequency
15	ST_PLL_PWRDN	Power down PLL
16	ST_INPUT_CLAMP	Camp all input to pd_pmu
17	ST_POWEROFF	Power off vd_logic
18	ST_24M_OSC_DIS	Disable 24M OSC
19	ST_WAIT_WAKEUP	Wait for wakeup
20	ST_WAKEUP_RESET	Assert system reset of vd_logic
21	ST_EXT_PWRUP	Turn on vd_logic power, and wait for PMIC stable
23	ST_RELEASE_CLAMP	Release clamp input to pd_pmu
22	ST_24M_OSC_EN	Enable 24M OSC

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Num	States	Description
24	ST_PMU_HF	PD_PMU clock switch to high frequency
25	ST_WAKEUP_RESET_CLR	De-assert system reset of vd_logic
26	ST_PLL_PWRUP	Power up PLL
27	ST_BUS_PWRUP	NOT USED
28	ST_DDR_PWRUP	Power UP PD_DDR
29	ST_DDR_IO_PWRUP	NOT USED
30	ST_SREF_EXIT	DDR exit self-refresh
31	ST_TRANS_RESTORE_BUS	Release idle request to vd_logic bus
32	ST_CORE_CLK_EN	Enable core clock
33	ST_CORE_PWRUP	Power up vd_core
34	ST_SCU_PWRUP	Power up pd_scu
35	ST_TRANS_RESTORE_CORE	Release idle request to vd_core
36	ST_CPU0_PWRUP	Power up CPU core 0

8.6.2 Debug IO

RK1808 provides PMU Debug IO for FSM observation. Each IO corresponding with a bit of PMU power states [4:0].

Table 8-3 Low Power State

Module Pin	Direction	Pad Name	IOMUX Setting
power_state[0]	O	IO_UART0tx_PMUdebug0_GPIO0B2pmuio2	PMUGRF_GPIO0B_IOMUX[5:4]=2'b10
power_state[1]	O	IO_UART0rx_PMUdebug1_GPIO0B3pmuio2	PMUGRF_GPIO0B_IOMUX[7:6]=2'b10
power_state[2]	O	IO_UART0cts_PMUdebug2_PMUdebug_sout_GPIO0B4pmuio2	PMUGRF_GPIO0B_IOMUX[9:8]=2'b10
power_state[3]	O	IO_PWM1_UART3txm0_PMUdebug3_GPIO0C3pmuio2	PMUGRF_GPIO0C_IOMUX[7:6]=2'b11
power_state[4]	O	IO_PWM3_UART3rxm0_PMUdebug4_GPIO0C4pmuio2	PMUGRF_GPIO0C_IOMUX[9:8]=2'b11
power_state[5]	O	IO_I2C1scl_PMUdebug5_GPIO0C0pmuio2	PMUGRF_GPIO0C_IOMUX[1:0]=2'b11
debug_Sout	O	IO_UART0cts_PMUdebug2_PMUdebug_sout_GPIO0B4pmuio2	PMUGRF_GPIO0B_IOMUX[9:8]=2'b11

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 9 Process Voltage Temperature Monitor (PVTM)

9.1 Overview

The Process-Voltage-Temperature Monitor (PVTM) is used to monitor the chip performance variance caused by chip process, voltage and temperature.
PVTM supports the following features:

- A clock oscillation ring is integrated and used to generate a clock like signal, the frequency of this clock is determined by the cell delay value of clock oscillation ring circuit.
- A frequency counter is used to measure the frequency of the clock oscillation ring.
- Follow PVTM blocks are supported:
  - core_pvtm, used near Cortex-A35
  - npu_pvtm, used near NPU
  - pmu_pvtm, used near PMU

9.2 Block Diagram

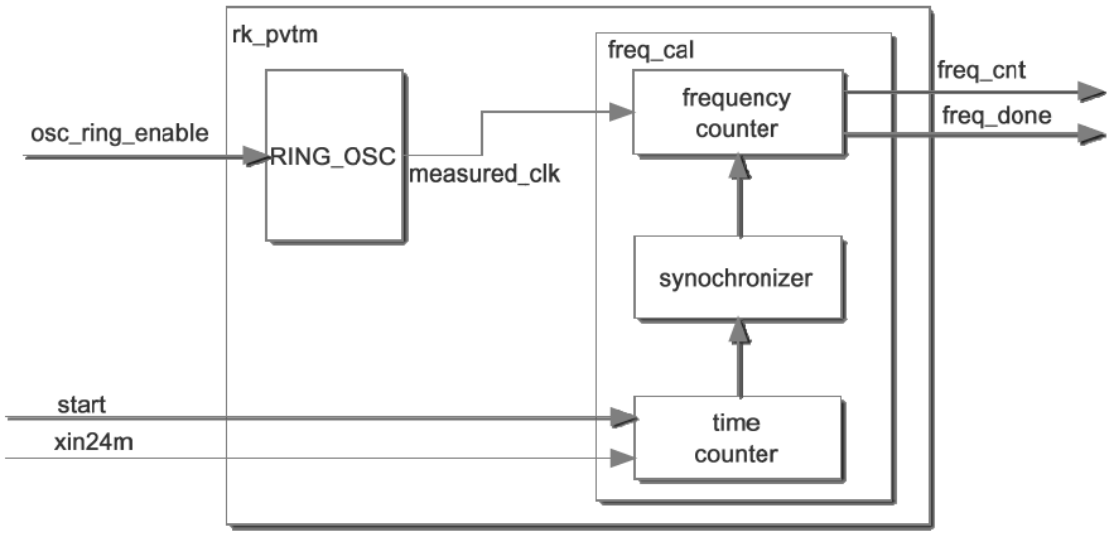


Fig. 9-1 PVTM Block Diagram

The PVTM include two main blocks:

- RING_OSC, it is composed with inverters with odd number, which is used to generate a clock. core_pvtm support 3clock oscillation rings in RING_OSC, and finally select a clock output by signal osc_ring_sel.
- Freq_cal, it is used to measure the frequency of clock which generated from the RING_SOC block.

9.3 Function Description

9.3.1 Frequency Calculation

A clock is generated by the RING_OSC, and a frequency fixed clock (24MHz) is used to calculate the cycles of the clock. Suppose the time period is 1s, then the clock period of RING_OSC clock is T= 1/clock_counter(s), the cell delay value is T/2.

9.3.2 Control Source and Result Destination

The pvtm is controlled by CRU and GRF, and the monitor result is got by GRF. Following tables shows the PVTM control source and result destination.

Table 9-1 Core_pvtm control source and result destination

Interface	Reset value	Control Source/Result Destination
xin24m	0x0	CRU_CLKGATE_CON0[5], clock gating control

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Interface	Reset value	Control Source/Result Destination
resetn	0x1	CRU_SOFTRST_CON2[11], reverse connect to resetn, active high
start	0x0	CORE_GRF_COREPVTM_CON0[0], active high
osc_ring_enable	0x0	CORE_GRF_COREPVTM_CON0[1], active high
osc_ring_sel	0x0	CORE_GRF_COREPVTM_CON0[4:2] 3'b000: Osc_ring 0 3'b001: Osc_ring 1 3'b010: Osc_ring 2 3'b011: Osc_ring 3 3'b100: Osc_ring 4 Others: Reserved
cal_cnt	0x0	CORE_GRF_COREPVTM_CON1[31:0]
freq_done	0x0	CORE_GRF_COREPVTM_STATUS0[0]
freq_cnt	0x0	CORE_GRF_COREPVTM_STATUS1[31:0]

Table 9-2 PMU_pvtm control source and result destination

Interface	Reset value	Control Source/Result Destination
xin24m	0x0	CRU_PMU_CLKGATE_CON1[4], clock gating control
resetn	0x1	CRU_SOFTRST_CON9[8], reverse connect to resetn, active high
start	0x1	PMU_GRF_PVTM_CON0[0], active high
osc_ring_enable	0x1	PMU_GRF_PVTM_CON0[1], active high
cal_cnt	0x0	PMU_GRF_PVTM_CON1[31:0]
freq_done	0x0	PMU_GRF_PVTM_STATUS0[0]
freq_cnt	0x0	PMU_GRF_PVTM_STATUS1[31:0]

Table 9-3 NPU_pvtm control source and result destination

Interface	Reset value	Control Source/Result Destination
xin24m	0x0	CRU_CLKGATE_CON0[15], clock gating control
resetn	0x1	CRU_SOFTRST_CON2[10], reverse connect to resetn, active high
start	0x0	BUS_GRF_NPUPVTM_CON0[0], active high
osc_ring_enable	0x0	BUS_GRF_NPUPVTM_CON0[1], active high
osc_ring_sel	0x0	BUS_GRF_NPUPVTM_CON0[4:2] 3'b000: Osc_ring 0 3'b001: Osc_ring 1 3'b010: Osc_ring 2 3'b011: Osc_ring 3 3'b100: Osc_ring 4 Others: Reserved
cal_cnt	0x0	BUS_GRF_NPUPVTM_CON1[31:0]
freq_done	0x0	BUS_GRF_NPUPVTM_STATUS0[0]
freq_cnt	0x0	BUS_GRF_NPUPVTM_STATUS1[31:0]

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

9.3.3 pmu_pvtm usage

A clock divided from pmu_pvtm oscillation ring is used in low power mode, which can replace the function of 32KHz clock source by configure CRU_CLKSEL_CON2[9:8]. The division factor is configured by GRF_PVTM_CON0[11:2].

9.4 Application Notes

9.4.1 PVTM Usage Flow

1. Enable the frequency fixed clock xin24m.
2. Reset the pvtm.
3. Set osc_ring_enable '1' to enable the generated clock.
4. Set osc_ring_sel to select the clock oscillation ring (Only core_pvtm need)
5. Configure the cal_cnt to an appropriate value.
6. Set start '1' to calculate the cycles of the generated clock.
7. Wait the freq_done is asserted, then get the value of freq_cnt. The period of RING_OSC clock is T= cal_cnt*(Period of 24MHz clock)/freq_cnt, the cell delay value is T/2.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 10 Advanced eXtensible Interface Performance (AXI PERF)

10.1 Overview

IN RK1808 there is one AXI_PERF: CA35_AXI_PERF for Core_A35. AXI PERF finish related statistic by monitoring IP AXI interface. Following is the specific features:

- AXI read statistic
  - Support max read latency of one ID
  - Support the statistic of the average latency of one ID
  - Support count the burst which the read latency is more than A value
  - Support read bandwidth statistic for one or all ID
  - Support read real bandwidth statistic and DDR align bandwidth statistic
- AXI write statistic
  - Support write bandwidth statistic for one or all ID
  - Support write real bandwidth statistic and DDR align bandwidth statistic
- AXI read and write address monitor(just for enhance version)
  - Support monitor read from appointed address area for one or some ID
  - Support monitor write to appointed address area for one or some ID
  - Interrupt generate.

10.2 Block Diagram

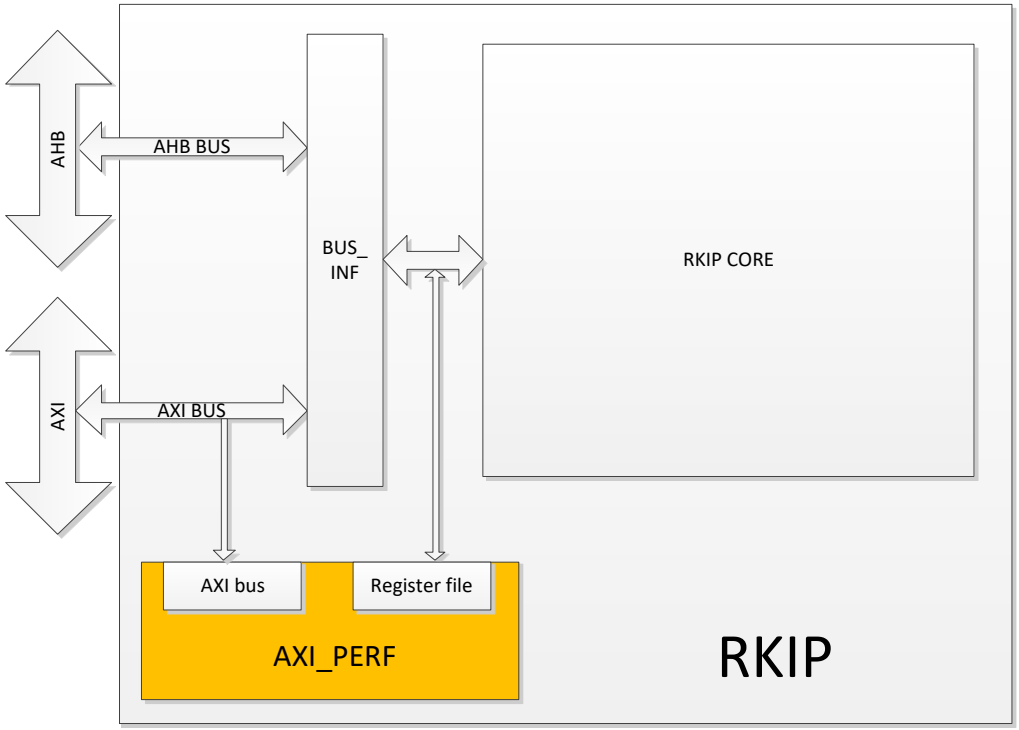


Fig. 10-1 AXI_PERF block diagram

AXI_PERF module implement statistic by grabbing RKIP AXI interface signals. It's configuration signals and static results are configured by RKIP AHB interface.

10.3 Register Description

10.3.1 Registers Summary

Name	Offset	Size	Reset Value	Description
CORE_AXI_PERF_PERF_CON0	0x0000	W	0x00000000	AXI performance monitor control register

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
CORE_AXI_PERF_PERF_CON1	0x0004	W	0x00000000	AXI performance monitor control register
CORE_AXI_PERF_PERF_CON2	0x0008	W	0x00000000	AXI performance monitor control register
CORE_AXI_PERF_PERF_CON3	0x000c	W	0x00000000	AXI performance monitor control register
CORE_AXI_PERF_PERF_CON4	0x0010	W	0x00000000	axi performance monitor control register
CORE_AXI_PERF_PERF_CON5	0x0014	W	0x00000000	AXI performance monitor control register
CORE_AXI_PERF_PERF_CON6	0x0018	W	0x00000000	AXI performance monitor control register
CORE_AXI_PERF_PERF_CON7	0x001c	W	0x00000000	AXI performance monitor control register
CORE_AXI_PERF_PERF_CON8	0x0020	W	0x00000000	AXI performance monitor control register
CORE_AXI_PERF_PERF_RD_MAX_LATENCY_NUM	0x0030	HW	0x0000	Read max latency number.AXI performance monitor status register.
CORE_AXI_PERF_PERF_RD_LATENCY_SAMP_NUM	0x0034	W	0x00000000	The burst number of bigger than configed threshold value.AXI performance monitor status register.
CORE_AXI_PERF_PERF_RD_LATENCY_ACC_SUM	0x0038	W	0x00000000	Total sample number.AXI performance monitor status register.
CORE_AXI_PERF_PERF_RD_AXI_TOTAL_BYTE	0x003c	W	0x00000000	AXI active total read bytes/ddr align read bytes.AXI performance monitor status register.
CORE_AXI_PERF_PERF_WR_AXI_TOTAL_BYTE	0x0040	W	0x00000000	AXI active total write bytes/ddr align write bytes.AXI performance monitor status register.
CORE_AXI_PERF_PERF_WORKING_CNT	0x0044	W	0x00000000	RKIP working counter.AXI performance monitor status register.
CORE_AXI_PERF_PERF_INT_STATUS	0x0048	W	0x00000000	Raw Interrupt Status

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

10.3.2 Detail Register Description

CORE_AXI_PERF_PERF_CON0

Address: Operational Base + offset (0x0000)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Bit0~15 write enable When bit 16=1, bit 0 can be written by software. When bit 16=0, bit 0 cannot be written by software; When bit 17=1, bit 1 can be written by software . When bit 17=0, bit 1 cannot be written by software; ..... When bit 31=1, bit 15 can be written by software . When bit 31=0, bit 15 cannot be written by software;
15	RW	0x0	sw_rd_latency_id_range_e For latency: 0: ID latency collect; 1: ID latency collect in sw_araddr_mon_st, sw_araddr_mon_end
14	RO	0x0	reserved
13:8	RW	0x00	sw_rd_latency_id Axi read channel id for latency AXI_PERFormance test
7:6	RW	0x0	sw_ddr_align_type 2'b00: 16-Byte align 2'b01: 32-Byte align 2'b10: 64-Byte align 2'b11: 128-Byte align
5	RW	0x0	sw_aw_cnt_id_type AXI_PERF counter id control 1'b0: Count all write channel id 1'b1: Count sw_ar_count_id write channel only
4	RW	0x0	sw_ar_cnt_id_type AXI_PERF counter id control 1'b0: Count all read channel id 1'b1: Count sw_ar_count_id read channel only
3	RW	0x0	sw_axi_cnt_type_wrap sw_axi_cnt_type=1: ddr_align_cnt_mode to count wrap data
2	RW	0x0	sw_axi_cnt_type AXI_PERF counter type 1'b0: AXI transfer test 1'b1: DDR align transfer num count
1	RW	0x0	sw_axi_perf_clr AXI_PERF clear bit 1'b0: Disable 1'b1: Enable
0	RW	0x0	sw_axi_perf_work AXI_PERF enable bit 1'b0: Disable 1'b1: Enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

CORE_AXI_PERF_PERF_CON1

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Bit0~15 write enable When bit 16=1, bit 0 can be written by software . When bit 16=0, bit 0 cannot be written by software; When bit 17=1, bit 1 can be written by software . When bit 17=0, bit 1 cannot be written by software; ..... When bit 31=1, bit 15 can be written by software . When bit 31=0, bit 15 cannot be written by software;
15:12	RO	0x0	reserved
11:0	RW	0x000	sw_rd_latency_thr Read channel latency threshold

CORE_AXI_PERF_PERF_CON2

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Bit0~15 write enable When bit 16=1, bit 0 can be written by software . When bit 16=0, bit 0 cannot be written by software; When bit 17=1, bit 1 can be written by software . When bit 17=0, bit 1 cannot be written by software; ..... When bit 31=1, bit 15 can be written by software . When bit 31=0, bit 15 cannot be written by software;
15	RW	0x0	sw_axi_perf_int_clr perf_int_interrupt clear, high avlid
14	RW	0x0	sw_axi_perf_int_e perf_int interrupt enable, high avlid
13	RO	0x0	reserved
12:8	RW	0x00	sw_aw_count_id When sw_aw_cnt_id_type=1, only count the sw_aw_count_id channel
7:6	RO	0x0	reserved
5:0	RW	0x00	sw_ar_count_id When sw_ar_cnt_id_type=1, only count the sw_ar_count_id channel

CORE_AXI_PERF_PERF_CON3

Address: Operational Base + offset (0x000c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	write_enable Bit0~15 write enable When bit 16=1, bit 0 can be written by software . When bit 16=0, bit 0 cannot be written by software; When bit 17=1, bit 1 can be written by software . When bit 17=0, bit 1 cannot be written by software; ..... When bit 31=1, bit 15 can be written by software . When bit 31=0, bit 15 cannot be written by software;
15:10	RW	0x00	sw_ar_mon_id Monitor AXI read id access the range sw_araddr_mon_st, sw_araddr_mon_end.
9:4	RW	0x00	sw_ar_mon_id_bmsk sw_ar_mon_id_type=1 mode: sw_ar_mon_id bit compare mask, high mask
3:2	RO	0x0	reserved
1	RW	0x0	sw_ar_mon_id_type 1'b0: Monitor AXI all ar-channels; 1'b1: Count sw_ar_mon_id channel only.
0	RW	0x0	sw_ar_mon_id_msk perf_int of read AXI access the range, high mask

CORE_AXI_PERF_PERF_CON4

Address: Operational Base + offset (0x0010)

Bit	Attr	Reset Value	Description
31:12	RO	0x0	reserved
11:7	RW	0x00	sw_aw_mon_id Monitor AXI write id access the range sw_awaddr_mon_st, sw_awaddr_mon_end.
6:2	RW	0x00	sw_aw_mon_id_bmsk sw_aw_mon_id_type=1 mode: sw_aw_mon_id bit compare mask, high mask
1	RW	0x0	sw_aw_mon_id_type 1'b0: Monitor AXI all aw-channels; 1'b1: Count sw_aw_mon_id channel only.
0	RW	0x0	sw_aw_mon_id_msk perf_int of write AXI access the range, high mask

CORE_AXI_PERF_PERF_CON5

Address: Operational Base + offset (0x0014)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_araddr_mon_st Performance monitor read start address

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

CORE_AXI_PERF_PERF_CON6

Address: Operational Base + offset (0x0018)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_araddr_mon_end Performance monitor read end address

CORE_AXI_PERF_PERF_CON7

Address: Operational Base + offset (0x001c)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_awaddr_mon_st Performance write monitor start address

CORE_AXI_PERF_PERF_CON8

Address: Operational Base + offset (0x0020)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_awaddr_mon_end Performance monitor write end address

CORE_AXI_PERF_PERF_RD_MAX_LATENCY_NUM

Address: Operational Base + offset (0x0030)

Bit	Attr	Reset Value	Description
15:12	RO	0x0	reserved
11:0	RW	0x000	rd_max_latency_r AXI max read latency(unit: cycles), read max latency value

CORE_AXI_PERF_PERF_RD_LATENCY_SAMP_NUM

Address: Operational Base + offset (0x0034)

Bit	Attr	Reset Value	Description
31:26	RO	0x0	reserved
25:0	RO	0x0000000	rd_latency_samp_cnt_r AXI read latency total sample number, read latency thr number

CORE_AXI_PERF_PERF_RD_LATENCY_ACC_SUM

Address: Operational Base + offset (0x0038)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	rd_latency_acc_cnt_r AXI read latency (>sw_rd_latency_thr) total number

CORE_AXI_PERF_PERF_RD_AXI_TOTAL_BYTE

Address: Operational Base + offset (0x003c)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	rd_axi_dat_cnt_r AXI active total read bytes/ddr align read bytes

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

CORE_AXI_PERF_PERF_WR_AXI_TOTAL_BYTE

Address: Operational Base + offset (0x0040)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	wr_axi_dat_cnt_r AXI active total write bytes/ddr align write bytes

CORE_AXI_PERF_PERF_WORKING_CNT

Address: Operational Base + offset (0x0044)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	perf_working_cnt_r RKIP working counter

CORE_AXI_PERF_PERF_INT_STATUS

Address: Operational Base + offset (0x0048)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:24	RW	0x00	aw_mon_axi_id_status The ID Be monitored write to the specific addr area
23:17	RO	0x0	reserved
16	RW	0x0	aw_mon_axi_hit_flag Write to the specific addr area interrupt status
15:14	RO	0x0	reserved
13:8	RW	0x00	ar_mon_axi_id_status The ID Be monitored read from the specific addr area
7:1	RO	0x0	reserved
0	RW	0x0	ar_mon_axi_hit_flag Read from the specific addr area interrupt status

10.4 Application Notes

10.4.1 Start and finish flow

- Assert sw_axi_perf_clr_e 1，and then assert sw_axi_perf_clr_e 0
- Config all the signals which are need tobe configured (sw_axi_perf_work_e still keep 0)
- Assert sw_axi_perf_work_e 1 after getting A frame end signal to start AXI_PERF
- Read back the statistic values after one or more frames
- Next loop

10.4.2 Read latency statistic case

The Read latency statistic is only fit for one ID specify by sw_rd_latency_id

Case1：the user want to get the average read latency and max read latency of one frame and ID 1 at RGA. Here is the steps:

- Finish RGA AXI PERF clear step
- Assert sw_rd_latency_id 1, sw_rd_latency_thr 0, sw_axi_perf_frm_type 1（PERF_LATENCY_CTRL0=0x14）
- Start RGA AXI PERF
- Start RGA and wait for finish interrupt

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

- Read back PERF_WORKING_CNT. This value means how many cycles are used by this frame.
- Read back PERF_RD_MAX_LATENCY_NUM. This value means the max latency of one frame for ID 1.
- Read back PERF_RD_LATENCY_SAMP_NUM and PERF_RD_LATENCY_ACC_SUM. Use the formulation perf_rd_latency_average=perf_rd_latency_acc_sum/perf_rd_latency_samp_num；perf_rd_latency_average is the average latency of ID 1.
- Clear RGA AXI_PERF and wait for next statistic

Case2：the user want to statistic the read average latency of every ID(assume ID from 0 to 9) and 60 frames at RGA. The burst to be counted need bigger than n. Here is the steps:

- Finish RGA AXI PERF clear step
- Assert sw_rd_latency_id 0, sw_rd_latency_thr n, sw_axi_perf_frm_type 0 （PERF_LATENCY_CTRL0=0xn00）
- Start RGA AXI PERF
- Start RGA and run 60 frames continually
- Read back PERF_WORKING_CNT，this value which is divided by60 means the average cycle uses of running this scenario
- Read back PERF_RD_LATENCY_SAMP_NUM and PERF_RD_LATENCY_ACC_SUM. Use the formulation perf_rd_latency_average=perf_rd_latency_acc_sum/perf_rd_latency_samp_num；perf_rd_latency_average is the average latency of ID 0 by 60 frames .
- Clear RGA AXI_PERF
- Repeat the above steps and finish id1 to id9 statistic

10.4.3 Bandwidth statistic case

Case3：the user want to get the single frame read and write bandwidth of ID 1 at RGA. Here is the steps:

- Finish RGA AXI PERF clear step
- Assert sw_ar_cnt_id_type sw_aw_cnt_id_type 1，sw_ar_count_id sw_aw_count_id 1，sw_axi_cnt_type 0，sw_axi_perf_frm_type 1(PERF_LATENCY_CTRL0=0x40 PERF_LATENCY_CTRL1=0x11c)
- Start RGA AXI PERF
- Start RGA and wait for finish interrupt
- Read back PERF_RD_AXI_TOTAL_BYTE, this value means the total read bytes of ID 1
- Read back PERF_WR_AXI_TOTAL_BYTE, this value means the total write bytes of ID 1
- Clear RGA AXI_PERF and wait for next statistic

Case4：the user want to get the 60 frames read and write average bandwidth of all ID at RGA(assume the DDR type is LPDDR4). Here is the steps:

- Finish RGA AXI PERF clear step
- Assert sw_ar_cnt_id_type sw_aw_cnt_id_type 0，sw_addr_align_type 2 , sw_axi_cnt_type 1, sw_axi_perf_frm_type 0 (PERF_LATENCY_CTRL0=0x00 PERF_LATENCY_CTRL1=0x2)
- Start RGA AXI PERF
- Start RGA and run 60 frames continually
- Read back PERF_RD_AXI_TOTAL_BYTE, this value which is divided by60 means the average total read bytes of all ID

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

- Read back PERF_WR_AXI_TOTAL_BYTE, this value which is divided by60 means the average total write bytes of all ID
- Clear RGA AXI_PERF and wait for next statistic

10.4.4 CPU address read and write monitor case

Case5：the user want to monitor CPU of ID 127 read from or write to the address area(from 0x4000 to 0x8000) action of some scenario. Here is the steps:

- Finish RGA AXI PERF clear step, assert mon_id_msk 0
- Assert mon_id_type 1，mon_id 127，AXI_PERFE_RD_MON_ST AXI_PERFE_WR_MON_ST 0x4000，AXI_PERFE_RD_MON_END，AXI_PERFE_WR_MON_END 0x8000（AXI_PERFE_CON3=0xfffffe02）
- Start CPU AXI PERF
- Start CPU, finish this scenario and wait for the AXI PERF interrupt during this period
- If CPU get AXI PERF interrupt, then read back AXI_PERFE_INT_STATUS, get ar_mon_axi_hit_flag and aw_mon_axi_hit_flag, these bits can make sure whether ID 127 read from the specify address area or write to the specify address area
- If the AXI PERF interrupt is not asserted, means ID 127’s action do not hit the event
- Clear CPU AXI_PERF and wait for next statistic

Case6：the user want to monitor CPU of ID0 - ID127 read from or write to the address area(from 0x14000 to 0x18000) action of some scenario. Here is the steps:

- Finish RGA AXI PERF clear step, assert mon_id_msk 0
- Assert mon_id_type 0，mon_id 0，AXI_PERFE_RD_MON_ST AXI_PERFE_WR_MON_ST 0x14000，AXI_PERFE_RD_MON_END，AXI_PERFE_WR_MON_END 0x18000（AXI_PERFE_CON3=0xffff01fc）
- Start CPU AXI PERF
- Start CPU, finish this scenario and wait for the AXI PERF interrupt during this period
- If CPU get AXI PERF interrupt, then read back AXI_PERFE_INT_STATUS, get ar_mon_axi_hit_flag and aw_mon_axi_hit_flag, these bits can make sure whether these IDs read from the specify address area or write to the specify address area. We can make sure which ID hit the event by aw_mon_axi_id_status or ar_mon_axi_id_status
- If the AXI PERF interrupt is not asserted, means no ID’s action hit the event
- Clear CPU AXI_PERF and wait for next statistic

10.4.5 DDR bandwidth statics with wrap mode

In wrap mode, DDR is always aligned. So, we must solve the deviation of bandwidth statics when the sw_axi_cnt_type assert to 1 by asserting the configured bit sw_axi_cnt_type_wrap to 1.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 11 Mobile Storage Host Controller

11.1 Overview

The Mobile Storage Host Controller is designed to support Secure Digital memory (SD-max version 3.01) with 1 bits or 4 bits data width, Multimedia Card(MMC-max version 4.51) with 1 bits or 4 bits or 8 bits data width.
The Host Controller is instantiated for SDMMC, SDIO and EMMC. The interface difference between these instances is shown in "Interface Description".
The Host Controller supports following features:

- Bus Interface Features:
  - Support AMBA AHB interface for master and slave
  - Supports internal DMA interface(IDMAC)
    - Supports 32-bit data transfers
    - Single-channel; single engine used for Transmit and Receive, which are mutually exclusive
    - Dual-buffer and chained descriptor linked list
    - Each descriptor can transfer up to 4KB of data in chained mode and 8KB of data in dual-buffer mode
    - Programmable burst size for optimal host bus utilization
  - Support combined single FIFO for both transmit and receive operations
  - Support FIFO size of 256x32
  - Support FIFO over-run and under-run prevention by stopping card clock
- Card Interface Features:
  - Support Secure Digital memory protocol commands
  - Support Secure Digital I/O protocol commands
  - Support Multimedia Card protocol commands
  - Support Command Completion Signal and interrupts to host
  - Support CRC generation and error detection
  - Support programmable baud rate
  - Support power management and power switch
  - Support card detection
  - Support write protection
  - Support hardware reset
  - Support SDIO interrupt in 1-bit and 4-bit modes
  - Support 4-bit mode in SDIO3.0
  - Support SDIO suspend and resume operation
  - Support SDIO read wait
  - Support block size of 1 to 65,535 bytes
  - Support 1-bit, 4-bit and 8-bit SDR modes
  - Support boot in 1-bit, 4-bit and 8-bit SDR modes
  - Support Packed Commands, CMD21, CMD49
- Clock Interface Features:
  - Support 0/90/180/270-degree phase shift operation for sample clock (cclk_in_sample) and drive clock(cclk_in_drv) relative to function clock(cclk_in) respectively

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

- Support phase tuning using delay line for sample clock(cclk_in_sample) and drive clock(cclk_in_drv) relative to function clock (cclk_in) respectively. The max number of delay element number is 256.

11.2 Block Diagram

The Host Controller consists of the following main functional blocks.

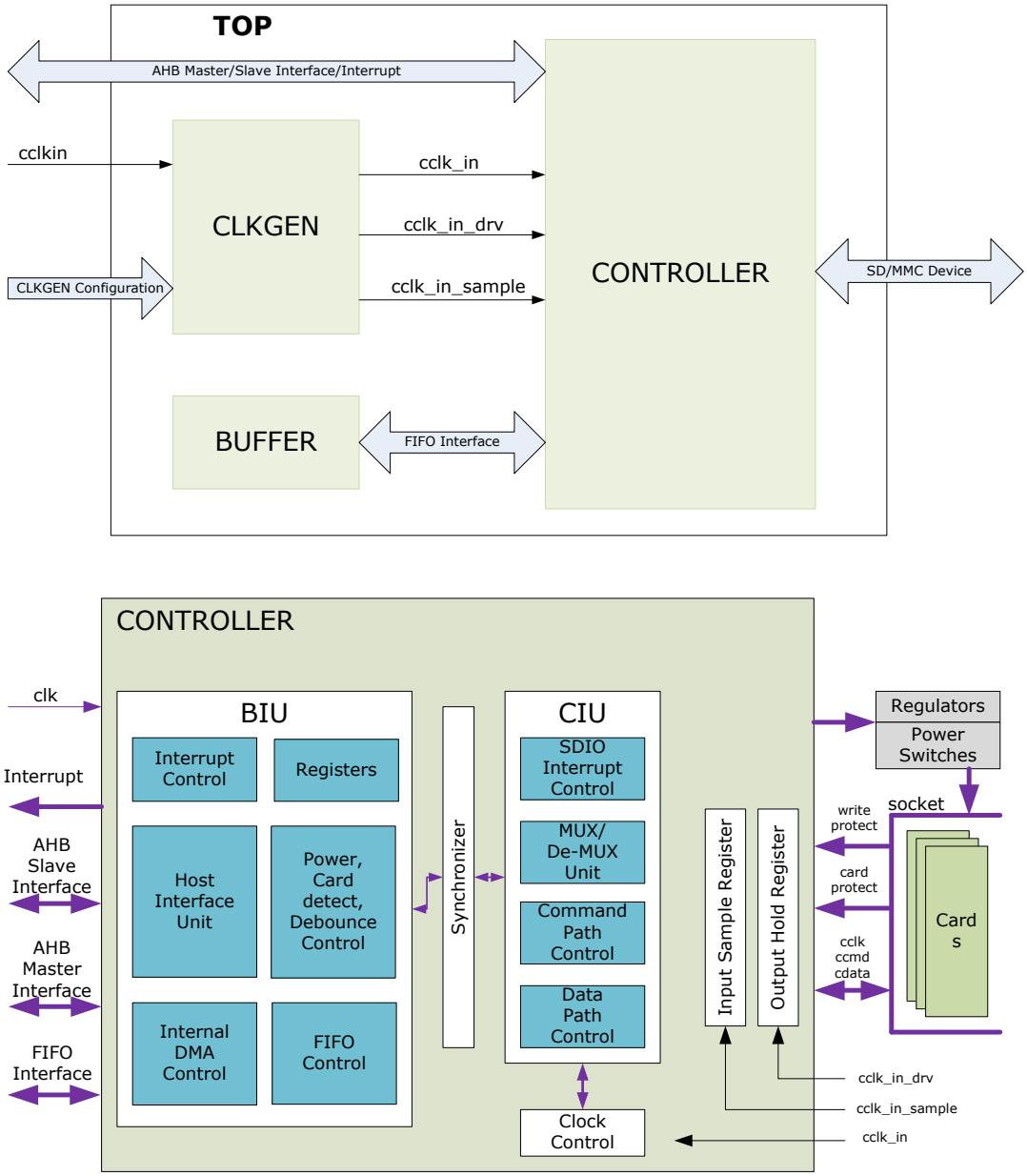


Fig. 11-1 Host Controller Block Diagram

- Clock Generate Unit(CLKGEN): generates card interface clock cclk_in/ cclk_sample/cclk_drv based on cclkin and configuration information.
- Asynchronous dual-port memory(BUFFER/FIFO): Uses a two-clock synchronous read and synchronous write dual-port RAM. One of the ports is connected to the host clock, and the second port is connected to the card clock.
- Bus Interface Unit (BIU): Provides AMBA AHB interfaces for register and data read/writes.
- Card Interface Unit (CIU): Takes care of the SD/MMC protocols and provides clock management.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

11.3 Function Description

11.3.1 Bus Interface Unit

The Bus Interface Unit provides the following functions:

- Host interface
- Interrupt control
- Register access
- External FIFO access
- Power control and card detection

1. Host Interface Unit

The Host Interface Unit is an AHB slave interface, which provides the interface between the SD/MMC card and the host bus.

2. Register Unit

The register unit is part of the bus interface unit; it provides read and write access to the registers.

All registers reside in the Bus Interface Unit clock domain. When a command is sent to a card by setting the start_bit, which is bit[31] of the SDMMC_CMD register, all relevant registers needed for the CIU operation are transferred to the CIU block. During this time, the registers that are transferred from the BIU to the CIU should not be written. The software should wait for the hardware to clear the start bit before writing to these registers again. The register unit has a hardware locking feature to prevent illegal writes to registers. The lock is necessary in order to avoid metastability violations, both because the host and card clock domains are different and to prevent illegal software operations.

Once a command start is issued by setting the start_bit of the SDMMC_CMD register, the following registers cannot be reprogrammed until the command is accepted by the card interface unit:

- SDMMC_CMD – Command
- SDMMC_CMDARG – Command Argument
- SDMMC_BYTCNT – Byte Count
- SDMMC_BLKSIZ – Block Size
- SDMMC_CLKDIV – Clock Divider
- SDMMC_CLKENA – Clock Enable
- SDMMC_CLKSRC – Clock Source
- SDMMC_TMOUT – Timeout
- SDMMC_CTYPE – Card Type

The hardware resets the start_bit once the CIU accepts the command. If a host write to any of these registers is attempted during this locked time, then the write is ignored and the hardware lock error bit is set in the SDMMC_RINTSTS register. Additionally, if the interrupt is enabled and not masked for a hardware lock error, then an interrupt is sent to the host.

When the Card Interface Unit is in an idle state, it typically takes the following number of clocks for the command handshake, where clk is the BIU clock and cclk_in is the CIU clock: 3 (clk) + 3 (cclk_in)

Once a command is accepted, you can send another command to the CIU-which has a one-deep command queue-under the following conditions:

- If the previous command was not a data transfer command, the new command is sent to the SD/MMC card once the previous command completes.
- If the previous command is a data transfer command and if wait_prvdata_complete (bit[13]) of the SDMMC_CMD register is set for the new command, the new command is sent to the SD/MMC card only when the data transfer completes.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

- If the wait_prvdata_complete is 0, then the new command is sent to the SD/MMC card as soon as the previous command is sent. Typically, you should use this only to stop or abort a previous data transfer or query the card status in the middle of a data transfer.

3. Interrupt Controller Unit

The interrupt controller unit generates an interrupt that depends on the controller raw interrupt status, the interrupt-mask register, and the global interrupt-enable register bit. Once an interrupt condition is detected, it sets the corresponding interrupt bit in the SDMMC_RINTSTS register. The raw interrupt status bit stays on until the software clears the bit by writing a 1 to the interrupt bit; a 0 leaves the bit untouched.

The interrupt port, int, is an active-high, level-sensitive interrupt. The interrupt port is active only when any bit in the SDMMC_RINTSTS register is active, the corresponding interrupt mask bit is 1, and the global interrupt enable bit is 1. The interrupt port is registered in order to avoid any combinational glitches.

The int_enable is reset to 0 on power-on, and the interrupt mask bits are set to 32'h0, which masks all the interrupts.

Notes:

Before enabling the interrupt, it is always recommended that you write 32'hffff_ffff to the SDMMC_RINTSTS register in order to clear any pending unserviced interrupts. When clearing interrupts during normal operation, ensure that you clear only the interrupt bits that you serviced.

The SDIO interrupt, Receive FIFO Data Request (RXDR), and Transmit FIFO Data Request (TXDR) are set by level-sensitive interrupt sources. Therefore, the interrupt source should be first cleared before you can clear the interrupt bit of the Raw Interrupt register. For example, on seeing the Receive FIFO Data Request (RXDR) interrupt, the FIFO should be emptied so that the "FIFO count greater than the RX-Watermark" condition, which triggers the interrupt, becomes inactive. The rest of the interrupts are triggered by a single clock-pulse-width source.

Table 11-1 Bits in Interrupt Status Register

Bits	Interrupt	Description
24	sdio_interrupt	Interrupt from SDIO card. In MMC-Ver3.3-only mode, these bits are always 0
16	Card no-busy	If card exit busy status, the interrupt happened
15	End Bit Error (read) /Write no CRC (EBE)	Error in end-bit during read operation, or no data CRC received during write operation. For MMC CMD19, there may be no CRC status returned by the card. Hence, EBE is set for CMD19. The application should not treat this as an error.
14	Auto Command Done (ACD)	Stop/abort commands automatically sent by card unit and not initiated by host; similar to Command Done (CD) interrupt. Recommendation: Software typically need not enable this for non CE-ATA accesses; Data Transfer Over (DTO) interrupt that comes after this interrupt determines whether data transfer has correctly competed.
13	Start Bit Error (SBE)	Error in data start bit when data is read from a card. In 4-bit mode, if DAT[0] line indicates start bit-that is, 0-and any of the other data bits do not have start bit, then this error is set. Busy Complete Interrupt when data is written to the card. This interrupt is generated after completion of busy driven by the card after the last data block is written into the card.
12	Hardware Locked write Error (HLE)	During hardware-lock period, write attempted to one of locked registers. When software sets the start_cmd bit in the SDMMC_CMD register, the Host Controller tries to load the command. If the command buffer is already filled with a command, this error is raised. The software then has to reload the command.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bits	Interrupt	Description
11	FIFO Underrun/ Overrun Error (FRUN)	Host tried to push data when FIFO was full, or host tried to read data when FIFO was empty. Typically this should not happen, except due to error in software. Card unit never pushes data into FIFO when FIFO is full, and pop data when FIFO is empty. If IDMAC is enabled, FIFO underrun/overrun can occur due to a programming error on MSIZE and watermark values in SDMMC_FIFOTH register.
10	Data Starvation by Host Timeout (HTO)	To avoid data loss, card clock out is stopped if FIFO is empty when writing to card, or FIFO is full when reading from card. Whenever card clock is stopped to avoid data loss, data-starvation timeout counter is started with data-timeout value. This interrupt is set if host does not fill data into FIFO during write to card, or does not read from FIFO during read from card before timeout period. Even after timeout, card clock stays in stopped state, with CIU state machines waiting. It is responsibility of host to push or pop data into FIFO upon interrupt, which automatically restarts cclk_out and card state machines. Even if host wants to send stop/abort command, it still needs to ensure it has to push or pop FIFO so that clock starts in order for stop/abort command to send on cmd signal along with data that is sent or received on data line.
9	Data Read Timeout (DRTO)	In Normal functioning mode: Data read timeout (DRTO) Data timeout occurred. Data Transfer Over (DTO) also set if data timeout occurs. In Boot Mode: Boot Data Start (BDS) When set, indicates that Host Controller has started to receive boot data from the card. A write to this register with a value of 1 clears this interrupt.
8	Response Timeout (RTO)	In normal functioning mode: Response timeout (RTO) Response timeout occurred. Command Done (CD) also set if response timeout occurs. If command involves data transfer and when response times out, no data transfer is attempted by Host Controller. In Boot Mode: Boot Ack Received (BAR) When expect_boot_ack is set, on reception of a boot acknowledge pattern—0-1-0—this interrupt is asserted. A write to this register with a value of 1 clears this interrupt.
7	Data CRC Error (DCRC)	Received Data CRC does not match with locally-generated CRC in CIU. Can also occur if the Write CRC status is incorrectly sampled by the Host.
6	Response CRC Error (RCRC)	Response CRC does not match with locally-generated CRC in CIU.
5	Receive FIFO Data Request (RXDR)	Interrupt set during read operation from card when FIFO level is greater than Receive-Threshold level. Recommendation: In DMA modes, this interrupt should not be enabled. In non-DMA mode: pop rx_wmark + 1 data from

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bits	Interrupt	Description
		FIFO.

4	Transmit FIFO Data Request (TXDR)	Interrupt set during write operation to card when FIFO level reaches less than or equal to Transmit-Threshold level.
		Recommendation:
		In DMA modes, this interrupt should not be enabled.
		In non-DMA mode:
		if (pending_bytes > (FIFO_DEPTH - tx_wmark))
		push (FIFO_DEPTH - tx_wmark) data into FIFO
		else
		push pending_bytes data into FIFO

3	Data Transfer Over (DTO)	Indicates Data transfer completed. Though on detection of errors-Start Bit Error, Data CRC error, and so on, DTO may or may not be set; the application must issue CMD12, which ensures that DTO is set. Recommendation: In non-DMA mode, when data is read from card, on seeing interrupt, host should read any pending data from FIFO. In DMA mode, DMA controllers guarantee FIFO is flushed before interrupt. DTO bit is set at the end of the last data block, even if the device asserts MMC busy after the last data block.
2	Command Done(CD)	Command sent to card and got response from card, even if Response Error or CRC error occurs.
1	Response Error (RE)	Error in received response set if one of following occurs: • Transmission bit != 0 • Command index mismatch • End-bit != 1
0	Card-Detect (CDT)	When one or more cards inserted or removed, this interrupt occurs. Software should read card-detect register to determine current card status. Recommendation: After power-on and before enabling interrupts, software should read card detect register and store it in memory. When interrupt occurs, it should read card detect register and compare it with value stored in memory to determine which card(s) were removed/inserted. Before exiting ISR, software should update memory with new card-detect value.

4. FIFO Controller Unit

The FIFO controller interfaces the external FIFO to the host interface and the card controller unit. When FIFO overrun and under-run conditions occur, the card clock stops in order to avoid data loss.

The FIFO uses a two-clock synchronous read and synchronous write dual-port RAM. One of the ports is connected to the host clock, clk, and the second port is connected to the card clock, cclk_in.

Notes: The FIFO controller does not support simultaneous read/write access from the same port. For debugging purposes, the software may try to write into the FIFO and read back the data; results are indeterminate, since the design does not support read/write access from the same port.

5. Power Control and Card Detection Unit

The register unit has registers that control the power. Power to each card can be selectively turned on or off.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

The card detection unit looks for any changes in the card-detect signals for card insertion or card removal. It filters out the debounces associated with mechanical insertion or removal, and generates one interrupt to the host. You can program the debounce filter value.
On power-on, the controller should read in the card_detect port and store the value in the memory. Upon receiving a card-detect interrupt, it should again read the card_detect port and XOR with the previous card-detect status to find out which card has interrupted. If more than one card is simultaneously removed or inserted, there is only one card-detect interrupt; the XOR value indicates which cards have been disturbed. The memory should be updated with the new card-detect value.

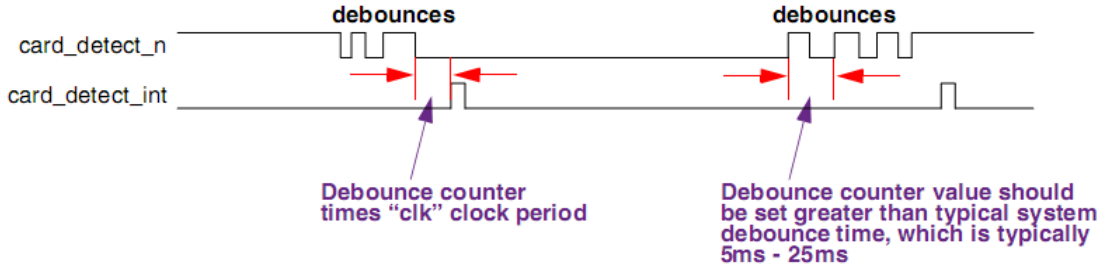


Fig. 11-2 SD/MMC Card-Detect Signal

6. DMA Interface Unit

DMA signals interface the Host Controller to an external DMA controller to reduce the software overhead during FIFO data transfers. The DMA request/acknowledge handshake is used for only data transfers. The DMA interface provides a connection to the DMA Controller.
On seeing the DMA request, the DMA controller initiates accesses through the host interface to read or write into the data FIFO. The Host Controller has FIFO transmit/receive watermark registers that you can set, depending on system latency. The DMA interface asserts the request in the following cases:

- Read from a card when the data FIFO word count exceeds the Rx-Watermark level
- Write to a card when the FIFO word count is less than or equal to the Tx-Watermark level

When the DMA interface is enabled, you can use normal host read/write to access the data FIFO.

11.3.2 Card Interface Unit

The Card Interface Unit (CIU) interfaces with the Bus Interface Unit (BIU) and the devices. The host writes command parameters to the BIU control registers, and these parameters are then passed to the CIU. Depending on control register values, the CIU generates SD/MMC command and data traffic on a selected card bus according to SD/MMC protocol. The Host Controller accordingly controls the command and data path.
The following software restrictions should be met for proper CIU operation:

- Only one data transfer command can be issued at a time.
- During an open-ended card write operation, if the card clock is stopped because the FIFO is empty, the software must first fill the data into the FIFO and start the card clock. It can then issue only a stop/abort command to the card.
- When issuing card reset commands (CMD0, CMD15 or CMD52_reset) while a card data transfer is in progress, the software must set the stop_abort_cmd bit in the SDMMC_CMD register so that the Host Controller can stop the data transfer after issuing the card reset command.
- When the data end bit error is set in the SDMMC_RINTSTS register, the Host Controller does not guarantee SDIO interrupt. The software should ignore the SDIO interrupt and issue the stop/abort command to the card, so that the card stops sending the read data.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

- If the card clock is stopped because the FIFO is full during a card read, the software should read at least two FIFO locations to start the card clock.

The CIU block consists of the following primary functional blocks:

- Command path
- Data path
- SDIO interrupt control
- Clock control
- Mux/demux unit

1. Command Path

The command path performs the following functions:

- Loads clock parameters
- Loads card command parameters
- Sends commands to card bus (ccmd_out line)
- Receives responses from card bus (ccmd_in line)
- Sends responses to BIU
- Drives the P-bit on command line

A new command is issued to the Host Controller by programming the BIU registers and setting the start_cmd bit in the SDMMC_CMD register. The BIU asserts start_cmd, which indicates that a new command is issued to the SD/MMC device. The command path loads this new command (command, command argument, timeout) and sends acknowledge to the BIU by asserting cmd_taken.

Once the new command is loaded, the command path state machine sends a command to the device bus-including the internally generated CRC7-and receives a response, if any. The state machine then sends the received response and signals to the BIU that the command is done, and then waits for eight clocks before loading a new command.

Load Command Parameters

One of the following commands or responses is loaded in the command path:

- New command from BIU – When start_cmd is asserted, then the start_cmd bit is set in the SDMMC_CMD register.
- Internally-generated auto-stop command – When the data path ends, the stop command request is loaded.
- IRQ response with RCA 0x000 – When the command path is waiting for an IRQ response from the MMC card and a "send irq response" request is signaled by the BIU, then the send_irq_response bit is set in the control register.

Loading a new command from the BIU in the command path depends on the following SDMMC_CMD register bit settings:

- update_clock_regs_only – If this bit is set in the SDMMC_CMD register, the command path updates only the clock enable, clock divider, and clock source registers. If this bit is not set, the command path loads the command, command argument, and timeout registers; it then starts processing the new command.
- wait_prvdata_complete – If this bit is set, the command path loads the new command under one of the following conditions:
  - Immediately, if the data path is free (that is, there is no data transfer in progress), or if an open-ended data transfer is in progress (byte_count = 0).
  - After completion of the current data transfer, if a predefined data transfer is in progress.

Send Command and Receive Response

Once a new command is loaded in the command path, update_clock_regs_only bit is unset – the command path state machine sends out a command on the device bus; the command path state machine is illustrated in following figure.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

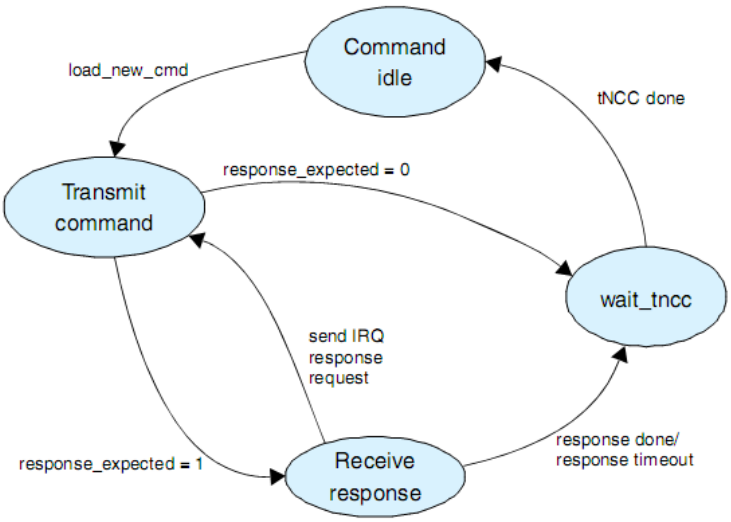


Fig. 11-3 Host Controller Command Path State Machine

The command path state machine performs the following functions, according to SDMMC_CMD register bit values:

- send_initialization – Initialization sequence of 80 clocks is sent before sending the command.
- response_expected – Response is expected for the command. After the command is sent out, the command path state machine receives a 48-bit or 136-bit response and sends it to the BIU. If the start bit of the card response is not received within the number of clocks programmed in the timeout register, then the response timeout and command done bit is set in the SDMMC_RINTSTS register as a signal to the BIU. If the response-expected bit is not set, the command path sends out a command and signals a response done to the BIU; that is, the command done bit is set in the SDMMC_RINTSTS register.
- response_length – If this bit is set, a 136-bit response is received; if it is not set, a 48-bit response is received.
- check_response_crc – If this bit is set, the command path compares CRC7 received in the response with the internally-generated CRC7. If the two do not match, the response CRC error is signaled to the BIU; that is, the response CRC error bit is set in the SDMMC_RINTSTS register.

Send Response to BIU

If the response_expected bit is set in the SDMMC_CMD register, the received response is sent to the BIU. The SDMMC_RESP0 register is updated for a short response, and the SDMMC_RESP3, SDMMC_RESP2, SDMMC_RESP1, and SDMMC_RESP0 registers are updated on a long response, after which the Command Done bit is set. If the response is for an auto_stop command sent by the CIU, the response is saved in the SDMMC_RESP1 register, after which the Auto Command Done bit is set.

Additionally, the command path checks for the following:

- Transmission bit = 0
- Command index matches command index of the sent command
- End bit = 1 in received card response

The command index is not checked for a 136-bit response or if the check_response_crc bit is unset. For a 136-bit response and reserved CRC 48-bit responses, the command index is reserved-that is, 111111.

Polling Command Completion Signal

The device generates the Command Completion Signal in order to notify the host controller of the normal command completion or command termination.

Command Completion Signal Detection and Interrupt to Host Processor

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

If the ccs_expected bit is set in the SDMMC_CMD register, the Command Completion Signal (CCS) from the device is indicated by setting the Data Transfer Over (DTO) bit in the SDMMC_RINTSTS register. The Host Controller generates a Data Transfer Over (DTO) interrupt if this interrupt is not masked.

Command Completion Signal Timeout

If the command expects a CCS from the device—if the ccs_expected bit is set in the SDMMC_CMD register—the command state machine waits for the CCS and remains in a wait_CCSS state. If the device fails to send out the CCS, the host software should implement a timeout mechanism to free the command and data path. The host controller does not implement a hardware timer; it is the responsibility of the host software to maintain a software timer.

In the event of a CCS timeout, the host should issue a CCSD by setting the send_ccsd bit in the SDMMC_CTRL register. The host controller command state machine sends the CCSD to the device and exits to an idle state. After sending the CCSD, the host should also send a CMD12 to the device in order to abort the outstanding command.

Send Command Completion Signal Disable

If the send_ccsd bit is set in the SDMMC_CTRL register, the host sends a Command Completion Signal Disable (CCSD) pattern on the CMD line. The host can send the CCSD while waiting for the CCS or after a CCS timeout happens.

After sending the CCSD pattern, the host sets the Command Done (CD) bit in SDMMC_RINTSTS and also generates an interrupt to the host if the Command Done interrupt is not masked.

2. Data Path

The data path block pops the data FIFO and transmits data on cdata_out during a write data transfer, or it receives data on cdata_in and pushes it into the FIFO during a read data transfer. The data path loads new data parameters—that is, data expected, read/write data transfer, stream/block transfer, block size, byte count, card type, timeout registers—whenever a data transfer command is not in progress.

If the data_expected bit is set in the SDMMC_CMD register, the new command is a data transfer command and the data path starts one of the following:

- Transmit data if the read/write bit = 1
- Data receive if read/write bit = 0

Data Transmit

The data transmit state machine, illustrated in following figure, starts data transmission two clocks after a response for the data write command is received; this occurs even if the command path detects a response error or response CRC error. If a response is not received from the card because of a response timeout, data is not transmitted. Depending upon the value of the transfer_mode bit in the SDMMC_CMD register, the data transmit state machine puts data on the card data bus in a stream or in block(s).

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

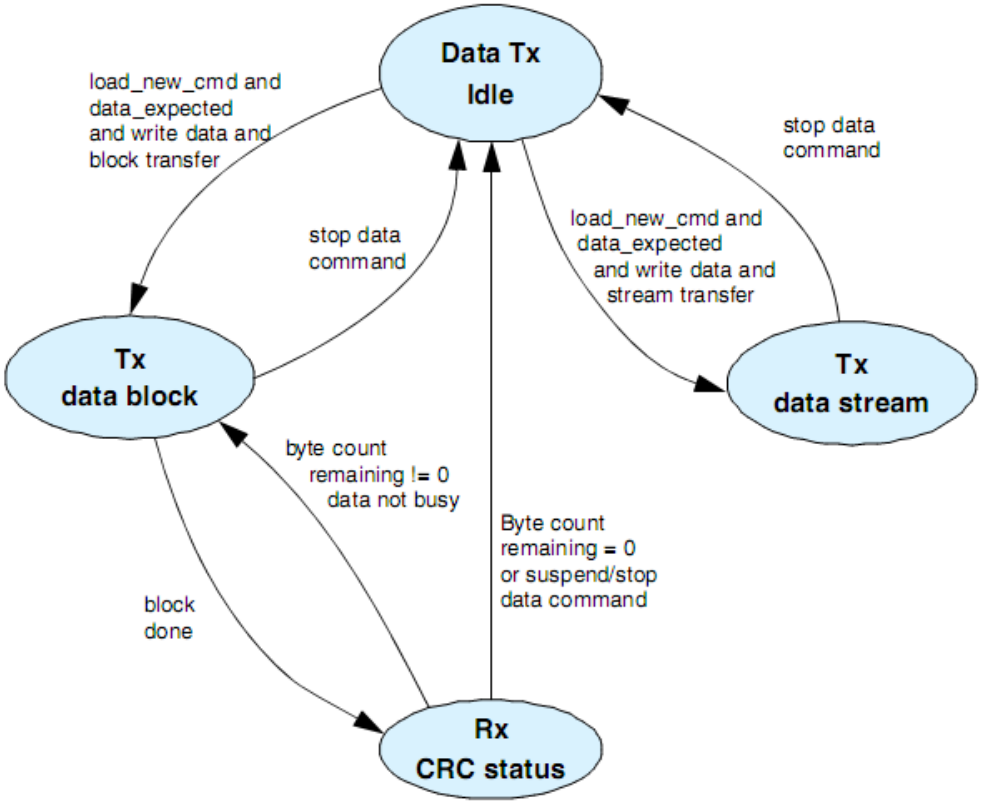


Fig. 11-4 Host Controller Data Transmit State Machine

Stream Data Transmit

If the transfer_mode bit in the SDMMC_CMD register is set to 1, it is a stream-write data transfer. The data path pops the FIFO from the BIU and transmits in a stream to the card data bus. If the FIFO becomes empty, the card clock is stopped and restarted once data is available in the FIFO.

If the SDMMC_BYTCNT register is programmed to 0, it is an open-ended stream-write data transfer. During this data transfer, the data path continuously transmits data in a stream until the host software issues a stop command. A stream data transfer is terminated when the end bit of the stop command and end bit of the data match over two clocks.

If the SDMMC_BYTCNT register is programmed with a non-zero value and the send_auto_stop bit is set in the SDMMC_CMD register, the stop command is internally generated and loaded in the command path when the end bit of the stop command occurs after the last byte of the stream write transfer matches.

This data transfer can also terminate if the host issues a stop command before all the data bytes are transferred to the card bus.

Single Block Data

If the transfer_mode bit in the SDMMC_CMD register is set to 0 and the SDMMC_BYTCNT register value is equal to the value of the SDMMC_BLKSIZ register, a single-block write-data transfer occurs. The data transmit state machine sends data in a single block, where the number of bytes equals the block size, including the internally-generated CRC16.

If the SDMMC_CTYPE register bit for the selected card – indicated by the card_num value in the SDMMC_CMD register – is set for a 1-bit, 4-bit, or 8-bit data transfer, the data is transmitted on 1, 4, or 8 data lines, respectively, and CRC16 is separately generated and transmitted for 1, 4, or 8 data lines, respectively.

After a single data block is transmitted, the data transmit state machine receives the CRC status from the card and signals a data transfer to the BIU; this happens when the data-transfer-over bit is set in the SDMMC_RINTSTS register.

If a negative CRC status is received from the card, the data path signals a data CRC error to the BIU by setting the data CRC error bit in the SDMMC_RINTSTS register.

Additionally, if the start bit of the CRC status is not received by two clocks after the end of the data block, a CRC status start bit error is signaled to the BIU by setting the write-no-CRC bit in the SDMMC_RINTSTS register.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Multiple Block Data

A multiple-block write-data transfer occurs if the transfer_mode bit in the SDMMC_CMD register is set to 0 and the value in the SDMMC_BYTCNT register is not equal to the value of the SDMMC_BLKSIZ register. The data transmit state machine sends data in blocks, where the number of bytes in a block equals the block size, including the internally-generated CRC16.

If the SDMMC_CTYPE register bit for the selected card – indicated by the card_num value in the SDMMC_CMD register – is set to 1-bit, 4-bit, or 8-bit data transfer, the data is transmitted on 1, 4, or 8 data lines, respectively, and CRC16 is separately generated and transmitted on 1, 4, or 8 data lines, respectively.

After one data block is transmitted, the data transmit state machine receives the CRC status from the card. If the remaining byte_count becomes 0, the data path signals to the BIU that the data transfer is done; this happens when the data-transfer-over bit is set in the SDMMC_RINTSTS register.

If the remaining data bytes are greater than 0, the data path state machine starts to transmit another data block.

If a negative CRC status is received from the card, the data path signals a data CRC error to the BIU by setting the data CRC error bit in the SDMMC_RINTSTS register, and continues further data transmission until all the bytes are transmitted.

Additionally, if the CRC status start bit is not received by two clocks after the end of a data block, a CRC status start bit error is signaled to the BIU by setting the write-no-CRC bit in the SDMMC_RINTSTS register; further data transfer is terminated.

If the send_auto_stop bit is set in the SDMMC_CMD register, the stop command is internally generated during the transfer of the last data block, where no extra bytes are transferred to the card. The end bit of the stop command may not exactly match the end bit of the CRC status in the last data block.

If the block size is less than 4, 16, or 32 for card data widths of 1 bit, 4 bits, or 8 bits, respectively, the data transmit state machine terminates the data transfer when all the data is transferred, at which time the internally generated stop command is loaded in the command path.

If the byte_count is 0 – the block size must be greater than 0 – it is an open-ended block transfer. The data transmit state machine for this type of data transfer continues the block-write data transfer until the host software issues a stop or abort command.

Data Receive

The data-receive state machine, illustrated in following figure, receives data two clock cycles after the end bit of a data read command, even if the command path detects a response error or response CRC error. If a response is not received from the card because a response timeout occurs, the BIU does not receive a signal that the data transfer is complete; this happens if the command sent by the Host Controller is an illegal operation for the card, which keeps the card from starting a read data transfer.

If data is not received before the data timeout, the data path signals a data timeout to the BIU and an end to the data transfer done. Based on the value of the transfer_mode bit in the SDMMC_CMD register, the data-receive state machine gets data from the card data bus in a stream or block(s).

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

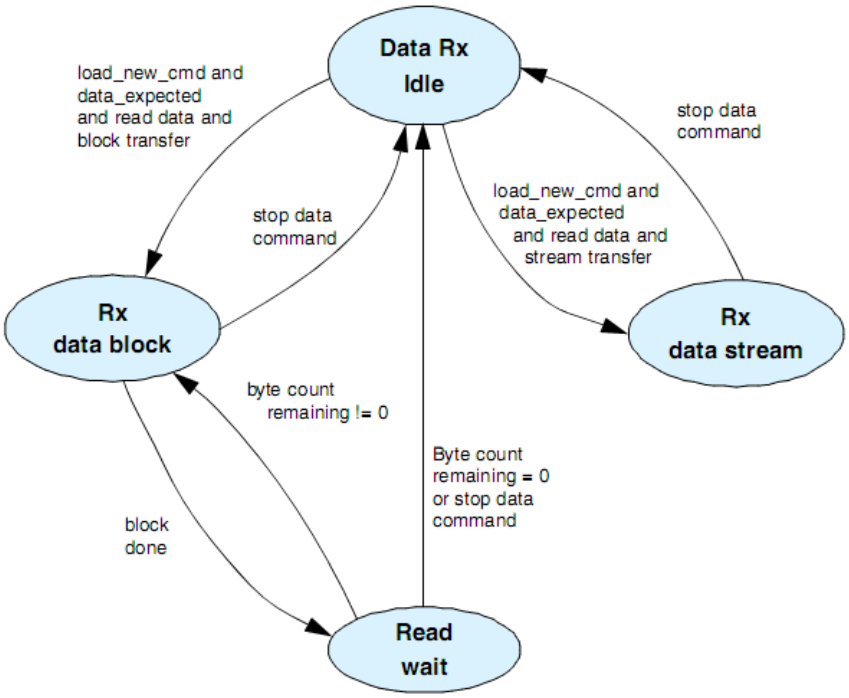


Fig. 11-5 Host Controller Data Receive State Machine

Stream Data Read

A stream-read data transfer occurs if the transfer_mode bit in the SDMMC_CMD register equals 1, at which time the data path receives data from the card and pushes it to the FIFO. If the FIFO becomes full, the card clock stops and restarts once the FIFO is no longer full.

An open-ended stream-read data transfer occurs if the SDMMC_BYTCNT register equals 0. During this type of data transfer, the data path continuously receives data in a stream until the host software issues a stop command. A stream data transfer terminates two clock cycles after the end bit of the stop command.

If the SDMMC_BYTCNT register contains a non-zero value and the send_auto_stop bit is set in the SDMMC_CMD register, a stop command is internally generated and loaded into the command path, where the end bit of the stop command occurs after the last byte of the stream data transfer is received. This data transfer can terminate if the host issues a stop or abort command before all the data bytes are received from the card.

Single-Block Data Read

A single-block read-data transfer occurs if the transfer_mode bit in the SDMMC_CMD register is set to 0 and the value of the SDMMC_BYTCNT register is equal to the value of the SDMMC_BLKSIZ register. When a start bit is received before the data times out, data bytes equal to the block size and CRC16 are received and checked with the internally-generated CRC16.

If the SDMMC_CTYPE register bit for the selected card – indicated by the card_num value in the SDMMC_CMD register – is set to a 1-bit, 4-bit, or 8-bit data transfer, data is received from 1, 4, or 8 data lines, respectively, and CRC16 is separately generated and checked for 1, 4, or 8 data lines, respectively. If there is a CRC16 mismatch, the data path signals a data CRC error to the BIU. If the received end bit is not 1, the BIU receives an end-bit error.

Multiple-Block Data Read

If the transfer_mode bit in the SDMMC_CMD register is set to 0 and the value of the SDMMC_BYTCNT register is not equal to the value of the SDMMC_BLKSIZ register, it is a multiple-block read-data transfer. The data-receive state machine receives data in blocks, where the number of bytes in a block is equal to the block size, including the internally-generated CRC16.

If the SDMMC_CTYPE register bit for the selected card – indicated by the card_num value in the SDMMC_CMD register – is set to a 1-bit, 4-bit, or 8-bit data transfer, data is received from 1, 4, or 8 data lines, respectively, and CRC16 is separately generated and checked for

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

1, 4, or 8 data lines, respectively.
After a data block is received, if the remaining byte_count becomes 0, the data path signals a data transfer to the BIU.
If the remaining data bytes are greater than 0, the data path state machine causes another data block to be received. If CRC16 of a received data block does not match the internally-generated CRC16, a data CRC error to the BIU and data reception continue further data transmission until all bytes are transmitted.
Additionally, if the end of a received data block is not 1, data on the data path signals terminate the bit error to the CIU and the data-receive state machine terminates data reception, waits for data timeout, and signals to the BIU that the data transfer is complete.
If the send_auto_stop bit is set in the SDMMC_CMD register, the stop command is internally generated when the last data block is transferred, where no extra bytes are transferred from the card; the end bit of the stop command may not exactly match the end bit of the last data block.
If the requested block size for data transfers to cards is less than 4, 16, or 32 bytes for 1-bit, 4-bit, or 8-bit data transfer modes, respectively, the data-transmit state machine terminates the data transfer when all data is transferred, at which point the internally-generated stop command is loaded in the command path. Data received from the card after that are then ignored by the data path.
If the byte_count is 0—the block size must be greater than 0—it is an open-ended block transfer. For this type of data transfer, the data-receive state machine continues the block-read data transfer until the host software issues a stop or abort command.

Auto-Stop

The Host Controller internally generates a stop command and is loaded in the command path when the send_auto_stop bit is set in the SDMMC_CMD register.
The software should set the send_auto_stop bit according to details listed in following table.

Table 11-2 Auto-Stop Generation

Card type	Transfer type	Byte Count	send_auto_stop bit set	Comments
MMC	Stream read	0	No	Open-ended stream
MMC	Stream read	>0	Yes	Auto-stop after all bytes transfer
MMC	Stream write	0	No	Open-ended stream
MMC	Stream write	>0	Yes	Auto-stop after all bytes transfer
MMC	Single-block read	>0	No	Byte count =0 is illegal
MMC	Single-block write	>0	No	Byte count =0 is illegal
MMC	Multiple-block read	0	No	Open-ended multiple block
MMC	Multiple-block read	>0	Yes①	Pre-defined multiple block
MMC	Multiple-block write	0	No	Open-ended multiple block
MMC	Multiple-block write	>0	Yes①	Pre-defined multiple block
SDMEM	Single-block read	>0	No	Byte count =0 is illegal
SDMEM	Single-block write	>0	No	Byte count =0 illegal
SDMEM	Multiple-block read	0	No	Open-ended multiple block
SDMEM	Multiple-block read	>0	Yes	Auto-stop after all bytes transfer
SDMEM	Multiple-block write	0	No	Open-ended multiple block
SDMEM	Multiple-block write	>0	Yes	Auto-stop after all bytes transfer
SDIO	Single-block read	>0	No	Byte count =0 is illegal

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Card type	Transfer type	Byte Count	send_auto_stop bit set	Comments
SDIO	Single-block write	>0	No	Byte count =0 illegal
SDIO	Multiple-block read	0	No	Open-ended multiple block
SDIO	Multiple-block read	>0	No	Pre-defined multiple block
SDIO	Multiple-block write	0	No	Open-ended multiple block
SDIO	Multiple-block write	>0	No	Pre-defined multiple block

①:*The condition under which the transfer mode is set to block transfer and byte_count is equal to block size is treated as a single-block data transfer command for both MMC and SD cards. If byte_count = n*block_size (n = 2, 3, …), the condition is treated as a predefined multiple-block data transfer command. In the case of an MMC card, the host software can perform a predefined data transfer in two ways: 1) Issue the CMD23 command before issuing CMD18/CMD25 commands to the card – in this case, issue MD18/CMD25 commands without setting the send_auto_stop bit. 2) Issue CMD18/CMD25 commands without issuing CMD23 command to the card, with the send_auto_stop bit set. In this case, the multiple-block data transfer is terminated by an internally-generated auto-stop command after the programmed byte count.*

The following list conditions for the auto-stop command.

- Stream read for MMC card with byte count greater than 0 – The Host Controller generates an internal stop command and loads it into the command path so that the end bit of the stop command is sent out when the last byte of data is read from the card and no extra data byte is received. If the byte count is less than 6 (48 bits), a few extra data bytes are received from the card before the end bit of the stop command is sent.
- Stream write for MMC card with byte count greater than 0 - The Host Controller generates an internal stop command and loads it into the command path so that the end bit of the stop command is sent when the last byte of data is transmitted on the card bus and no extra data byte is transmitted. If the byte count is less than 6 (48 bits), the data path transmits the data last in order to meet the above condition.
- Multiple-block read memory for SD card with byte count greater than 0 – If the block size is less than 4 (single-bit data bus), 16 (4-bit data bus), or 32 (8-bit data bus), the auto-stop command is loaded in the command path after all the bytes are read. Otherwise, the top command is loaded in the command path so that the end bit of the stop command is sent after the last data block is received.
- Multiple-block write memory for SD card with byte count greater than 0 – If the block size is less than 3 (single-bit data bus), 12 (4-bit data bus), or 24 (8-bit data bus), the auto-stop command is loaded in the command path after all data blocks are transmitted. Otherwise, the stop command is loaded in the command path so that the end bit of the stop command is sent after the end bit of the CRC status is received.
- Precaution for host software during auto-stop – Whenever an auto-stop command is issued, the host software should not issue a new command to the SD/MMC device until the auto-stop is sent by the Host Controller and the data transfer is complete. If the host issues a new command during a data transfer with the auto-stop in progress, an auto-stop command may be sent after the new command is sent and its response is received; this can delay sending the stop command, which transfers extra data bytes. For a stream write, extra data bytes are erroneous data that can corrupt the card data. If the host wants to terminate the data transfer before the data transfer is complete, it

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

can issue a stop or abort command, in which case the Host Controller does not generate an auto-stop command.

3. Non-Data Transfer Commands that Use Data Path

Some non-data transfer commands (non-read/write commands) also use the data path. Following table lists the commands and register programming requirements for them.

Table 11-3 Non-data Transfer Commands and Requirements

Base Address [12:8]	CMD 27	CMD 30	CMD 42	ACMD 13	ACMD 22	ACMD 51
SDMMC_CMD register programming						
cmd_index	6'h1b	6'h1e	6'h2a	6'h0d	6'h16	6'h33
response_expect	1	1	1	1	1	1
rResponse_length	0	0	0	0	0	0
check_response_crc	1	1	1	1	1	1
data_expected	1	1	1	1	1	1
read/write	1	0	1	0	0	0
transfer_mode	0	0	0	0	0	0
send_auto_stop	0	0	0	0	0	0
wait_prevdata_complete	0	0	0	0	0	0
stop_abort_cmd	0	0	0	0	0	0
Command Argument register programming						
	stuff bits	32-bit write protect data address	stuff bits	stuff bits	stuff bits	stuff bits
Block Size register programming						
	16	4	Num_bytes①	64	4	8
SDMMC_BYTCNT register programming						
	16	4	Num_bytes①	64	4	8

①: *Num_bytes = No. of bytes specified as per the lock card data structure (Refer to the SD specification and the MMC specification)*

4. SDIO Interrupt Control

Interrupt for SD card are reported to the BIU by asserting an interrupt signal for two clock cycles. SDIO card signal an interrupt by asserting cdata_in low during the interrupt period; an interrupt period for the selected card is determined by the interrupt control state machine. An interrupt period is always valid for non-active or non-selected cards, and 1-bit data mode for the selected card. An interrupt period for a wide-bus active or selected card is valid for the following conditions:

- Card is idle
- Non-data transfer command in progress
- Third clock after end bit of data block between two data blocks
- From two clocks after end bit of last data until end bit of next data transfer command

Bear in mind that, in the following situations, the controller does not sample the SDIO interrupt of the selected card when the card data width is 4 bits. Since the SDIO interrupt is level-triggered, it is sampled in a further interrupt period and the host does not lose any SDIO interrupt from the card.

- Read/Write Resume – The CIU treats the resume command as a normal data transfer command. SDIO interrupt during the resume command are handled similarly to other data commands. According to the SDIO specification, for the normal data command the interrupt period ends after the command end bit of the data command; for the resume command, it ends after the response end bit. In the case of the resume command, the

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

controller stops the interrupt sampling period after the resume command end bit, instead of stopping after the response end bit of the resume command.

- Suspend during read transfer – If the read data transfer is suspended by the host, the host sets the abort_read_data bit in the controller to reset the data state machine. In the CIU, the SDIO interrupt are handled such that the interrupt sampling starts after the abort_read_data bit is set by the host. In this case the controller does not sample SDIO interrupt between the period from response of the suspend command to setting the abort_read_data bit, and starts sampling after setting the abort_read_data bit.

5. Clock Control

The clock control block provides different clock frequencies required for SD/MMC cards. The cclk_in signal is the source clock (cclk_in >= card max operating frequency) for clock divider of the clock control block. This source clock (cclk_in) is used to generate different card clock frequencies (cclk_out). The card clock can have different clock frequencies, since the card can be a low-speed card or a full-speed card. The Host Controller provides one clock signal (cclk_out).

The clock frequency of a card depends on the following SDMMC_CLKENA registers:

- SDMMC_CLKDIV register– Internal clock divider is used to generate different clock frequencies required for card. The division factor for each clock divider can be programmed by writing to SDMMC_CLKDIV register. The clock divider is an 8-bit value that provides a clock division factor from 1 to 510; a value of 0 represents a clock-divider bypass, a value of 1 represents a divide by 2, a value of 2 represents a divide by 4, and so on. The implementation max divider is 1.
- SDMMC_CLKENA register – cclk_out can be enabled or disabled for each card under the following conditions:
  - clk_enable – cclk_out for a card is enabled if the clk_enable bit for a card in the SDMMC_CLKENA register is programmed (set to 1) or disabled (set to 0).
  - low-power mode – low-power mode of a card can be enabled by setting the low-power mode bit of the SDMMC_CLKENA register to 1. If low-power mode is enabled to save card power, the cclk_out is disabled when the card is idle for at least 8 card clock cycles. It is enabled when a new command is loaded and the command path goes to a non-idle state.

Additionally, cclk_out is disabled when an internal FIFO is full – card read (no more data can be received from card) – or when the FIFO is empty – card write (no data is available for transmission). This helps to avoid FIFO overrun and underrun conditions. It is used by the command and data path to qualify cclk_in for driving outputs and sampling inputs at the programmed clock frequency for the selected card, according to the SDMMC_CLKDIV and SDMMC_CLKSRC register values.

Under the following conditions, the card clock is stopped or disabled, along with the active clk_en, for the selected card:

- Clock can be disabled by writing to SDMMC_CLKENA register (clk_en bit = 1).
- If low-power mode is selected and card is idle, or not selected for 8 clocks.
- FIFO is full and data path cannot accept more data from the card and data transfer is incomplete –to avoid FIFO overrun.
- FIFO is empty and data path cannot transmit more data to the card and data transfer is incomplete – to avoid FIFO underrun.

6. Error Detection

- Response
  - Response timeout – Response expected with response start bit is not received within programmed number of clocks in timeout register.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

  - Response CRC error – Response is expected and check response CRC requested; response CRC7 does not match with the internally-generated CRC7.
  - Response error – Response transmission bit is not 0, command index does not match with the command index of the send command, or response end bit is not 1.
- Data transmit
  - No CRC status – During a write data transfer, if the CRC status start bit is not received two clocks after the end bit of the data block is sent out, the data path does the following:
    - Signals no CRC status error to the BIU
    - Terminates further data transfer
    - Signals data transfer done to the BIU
  - Negative CRC – If the CRC status received after the write data block is negative (that is, not 010), a data CRC error is signaled to the BIU and further data transfer is continued.
  - Data starvation due to empty FIFO – If the FIFO becomes empty during a write data transmission, or if the card clock is stopped and the FIFO remains empty for data timeout clocks, then a data-starvation error is signaled to the BIU and the data path continues to wait for data in the FIFO.
- Data receive
  - Data timeout – During a read-data transfer, if the data start bit is not received before the number of clocks that were programmed in the timeout register, the data path does the following:
    - Signals data-timeout error to the BIU
    - Terminates further data transfer
    - Signals data transfer done to BIU
  - Data start bit error – During a 4-bit or 8-bit read-data transfer, if the all-bit data line does not have a start bit, the data path signals a data start bit error to the BIU and waits for a data timeout, after which it signals that the data transfer is done.
  - Data CRC error – During a read-data-block transfer, if the CRC16 received does not match with the internally generated CRC16, the data path signals a data CRC error to the BIU and continues further data transfer.
  - Data end-bit error – During a read-data transfer, if the end bit of the received data is not 1, the data path signals an end-bit error to the BIU, terminates further data transfer, and signals to the BIU that the data transfer is done.
  - Data starvation due to FIFO full – During a read data transmission and when the FIFO becomes full, the card clock is stopped. If the FIFO remains full for data timeout clocks, a data starvation error is signaled to the BIU (Data Starvation by Host Timeout bit is set in SDMMC_RINTSTS register) and the data path continues to wait for the FIFO to start to empty.

11.3.3 Internal Direct Memory Access Controller (IDMAC)

The Internal Direct Memory Access Controller (IDMAC) has a control and status register (CSR) and a single Transmit/Receive engine, which transfers data from host memory to the device port and vice versa. The controller utilizes a descriptor to efficiently move data from source to destination with minimal host CPU intervention. You can program the controller to interrupt the host CPU in situations such as data Transmit and Receive transfer completion from the card, as well as other normal or error conditions.
The IDMAC and the host driver communicate through a single data structure. CSR

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

addresses 0x80 to 0x98 are reserved for host programming.
The IDMAC transfers the data received from the card to the data buffer in the host memory, and it transfers transmit data from the data buffer in the host memory to the FIFO. Descriptors that reside in the host memory act as pointers to these buffers.
A data buffer resides in physical memory space of the host and consists of complete data or partial data. Buffers contain only data, while buffer status is maintained in the descriptor. Data chaining refers to data that spans multiple data buffers. However, a single descriptor cannot span multiple data.
A single descriptor is used for both reception and transmission. The base address of the list is written into descriptor list base address register (SDMMC_DBADDR @0x88). A descriptor list is forward linked. The last descriptor can point back to the first entry in order to create a ring structure. The descriptor list resides in the physical memory address space of the host. Each descriptor can point to a maximum of two data buffers.

1. IDMAC CSR Access

When an IDMAC is introduced, an additional CSR space resides in the IDMAC that controls the IDMAC functionality. The host accesses the new CSR space in addition to the existing control register set in the BIU. The IDMAC CSR primarily contains descriptor information. For a write operation to the CSR, the respective CSR logic of the IDMAC and BIU decodes the address before accepting. For a read operation from the CSR, the appropriate CSR read path is enabled.
You can enable or disable the IDMAC operation by programming bit[25] in the SDMMC_CTRL register of the BIU. This allows the data transfer by accessing the slave interface on the AMBA bus if the IDMAC is present but disabled. When IDMAC is enabled, the FIFO cannot be accessed through the slave interface.

2. Descriptors

- Descriptor structures

The IDMAC uses these types of descriptor structures:

- Dual-Buffer Structure – The distance between two descriptors is determined by the skip length value programmed in the Descriptor Skip Length (DSL) field of the bus mode register (SDMMC_BMOD @0x80).

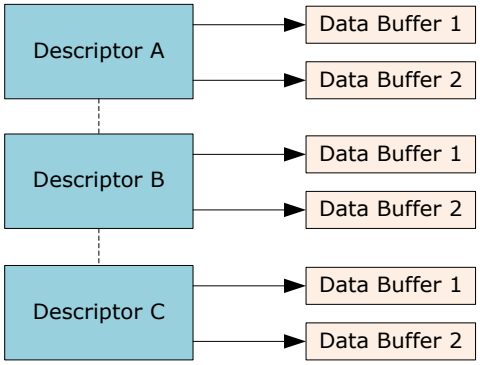
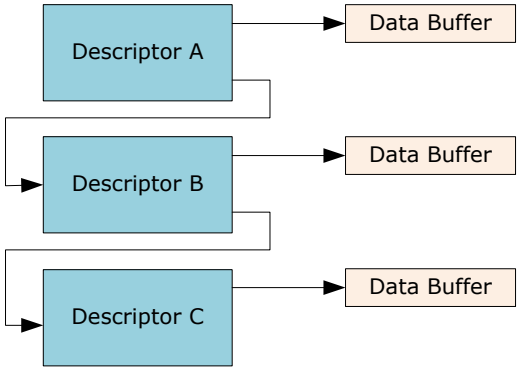


Fig. 11-6 Dual-Buffer Descriptor Structure

- Chain Structure – Each descriptor points to a unique buffer and the next descriptor.



Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Fig. 11-7 Chain Descriptor Structure

- Descriptor formats

Following figure illustrates the internal formats of a descriptor. The descriptor addresses must be aligned to the bus width used for 32-bit AHB data buses. Each descriptor contains 16 bytes of control and status information. DES0 is a notation used to denote the [31:0] bits, DES1 to denote [63:32] bits, DES2 to denote [95:64] bits, DES3 to denote [127:96] bits.

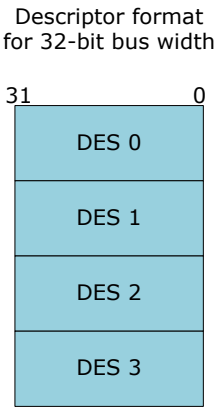


Fig. 11-8 Descriptor Formats for 32-bit AHB Address Bus Width

- The DES0 element in the IDMAC contains control and status information.

Table 11-4 Bits in IDMAC DES0 Element

Bit	Name	Description
31	OWN	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. The IDMAC clears this bit when it completes the data transfer.
30	Card Error Summary (CES)	These error bits indicate the status of the transaction to or from the card. These bits are also present in SDMMC_RINTSTS indicates the logical OR of the following bits: ● EBE: End Bit Error ● RTO: Response Time out ● RCRC: Response CRC ● SBE: Start Bit Error ● DRTO: Data Read Timeout ● DCRC: Data CRC for Receive ● RE: Response Error
29:6	Reserved	-
5	End of Ring (ER)	When set, this bit indicates that the descriptor list reached its final descriptor. The IDMAC returns to the base address of the list, creating a descriptor ring. This is meaningful for only a dual-buffer descriptor structure.
4	Second Address Chained (CH)	When set, this bit indicates that the second address in the descriptor is the Next Descriptor address rather than the second buffer address. When this bit is set, BS2 (DES1[25:13]) should be all zeros.
3	First Descriptor (FS)	When set, this bit indicates that this descriptor contains the first buffer of the data. If the size of the first buffer is 0, next Descriptor contains the beginning of the data.
2	Last	This bit is associated with the last block of a DMA transfer.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Name	Description
	Descriptor (LD)	When set, the bit indicates that the buffers pointed to by this descriptor are the last buffers of the data. After this descriptor is completed, the remaining byte count is 0. In other words, after the descriptor with the LD bit set is completed, the remaining byte count should be 0.
1	Disable Interrupt on Completion (DIC)	When set, this bit will prevent the setting of the TI/RI bit of the IDMAC status register (SDMMC_IDSTS) for the data that ends in the buffer pointed to by this descriptor.
0	Reserved	-

- The DES1 element contains the buffer size.

Table 11-5 Bits in IDMAC DES1 Element

Bit	Name	Description
31:26	Reserved	-
25:13	Buffer 2 Size (BS2)	These bits indicate the second data buffer byte size. The buffer size must be a multiple of 2, 4, or 8, depending upon the bus widths—16, 32, and 64 respectively. In the case where the buffer size is not a multiple of 2, 4, or 8, the resulting behavior is undefined. If this field is 0, the DMA ignores this buffer and proceeds to the next buffer in case of a dual-buffer structure. This field is not valid for chain structure; that is, if DES0[4] is set.
12:0	Buffer 1 Size (BS1)	Indicates the data buffer byte size, which must be a multiple of 2, 4, or 8 bytes, depending upon the bus widths—16, 32, and 64, respectively. In the case where the buffer size is not a multiple of 2, 4, or 8, the resulting behavior is undefined. This field should not be zero. Note: If there is only one buffer to be programmed, you need to use only the Buffer 1, and not Buffer 2.

- The DES2 element contains the address pointer to the data buffer.

Table 11-6 Bits in IDMAC DES2 Element

Bit	Name	Description
31:26	Reserved	
25:13	Buffer 2 Size (BS2)	These bits indicate the second data buffer byte size. The buffer size must be a multiple of 2, 4, or 8, depending upon the bus widths—16, 32, and 64 respectively. In the case where the buffer size is not a multiple of 2, 4, or 8, the resulting behavior is undefined. If this field is 0, the DMA ignores this buffer and proceeds to the next buffer in case of a dual-buffer structure. This field is not valid for chain structure; that is, if DES0[4] is set.
12:0	Buffer 1 Size (BS1)	Indicates the data buffer byte size, which must be a multiple of 2, 4, or 8 bytes, depending upon the bus widths—16, 32, and 64, respectively. In the case where the buffer size is not a multiple of 2, 4, or 8, the resulting behavior is undefined. This field should not be zero. Note: If there is only one buffer to be programmed, you need to use only the Buffer 1, and not Buffer 2.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

- The DES3 element contains the address pointer to the next descriptor if the present descriptor is not the last descriptor in a chained descriptor structure or the second buffer address for a dual-buffer structure.

Table 11-7 Bits in IDMAC DES3 Element

Bit	Name	Description
31:0	Buffer Address Pointer 2/Next Descriptor Address (BAP2)	These bits indicate the physical address of the second buffer when the dual-buffer structure is used. If the Second Address Chained (DES0[4]) bit is set, then this address contains the pointer to the physical memory where the Next Descriptor is present. If this is not the last descriptor, then the Next Descriptor address pointer must be bus-width aligned.

3. Initialization

IDMAC initialization occurs as follows:

1) Write to SDMMC_BMOD register to set host bus access parameters.
2) Write to SDMMC_IDINTEN register to mask unnecessary interrupt causes.
3) The software driver creates either the transmit or the receive descriptor list. Then it writes to IDMAC descriptor list base address register (SDMMC_DBADDR), providing the IDMAC with the starting address of the list.
4) The IDMAC engine attempts to acquire descriptors from the descriptor lists.

- Host Bus Burst Access

The IDMAC attempts to execute fixed-length burst transfers on the AHB Master interface if configured using the FB bit of the SDMMC_BMOD register. The maximum burst length is indicated and limited by the PBL field. The descriptors are always accessed in the maximum possible burst-size for the 16-bytes to be read— 16*8/bus-width.
The IDMAC initiates a data transfer only when sufficient space to accommodate the configured burst is available in the FIFO or the number of bytes to the end of data, when less than the configured burst-length.
The IDMAC indicates the start address and the number of transfers required to the AHB Master Interface. When the AHB Interface is configured for fixed-length bursts, then it transfers data using the best combination of INCR4/8/16 and SINGLE transactions. Otherwise, in no fixed-length bursts, it transfers data using INCR (undefined length) and SINGLE transactions.

- Host data buffer Alignment

The transmit and receive data buffers in host memory must be aligned, depending on the data width.

- Buffer Size Calculations

The driver knows the amount of data to transmit or receive. For transmitting to the card, the IDMAC transfers the exact number of bytes to the FIFO, indicated by the buffer size field of DES1.
If a descriptor is not marked as last-LS bit of DES0-then the corresponding buffer(s) of the descriptor are full, and the amount of valid data in a buffer is accurately indicated by its buffer size field. If a descriptor is marked as last, then the buffer cannot be full, as indicated by the buffer size in DES1. The driver is aware of the number of locations that are valid in this case.

- Transmission

IDMAC transmission occurs as follows:

1) The host sets up the elements (DES0-DES3) for transmission and sets the OWN bit (DES0[31]). The host also prepares the data buffer.
2) The host programs the write data command in the SDMMC_CMD register in BIU.
3) The host will also program the required transmit threshold level (tx_wmark field in SDMMC_FIFOTH register).

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

4) The IDMAC determines that a write data transfer needs to be done as a consequence of step 2.
5) The IDMAC engine fetches the descriptor and checks the OWN bit. If the OWN bit is not set, it means that the host owns the descriptor. In this case the IDMAC enters suspend state and asserts the Descriptor Unable interrupt in the SDMMC_IDSTS register. In such a case, the host needs to release the IDMAC by writing any value to the poll demand register.
6) It will then wait for Command Done (CD) bit and no errors from BIU which indicates that a transfer can be done.
7) The IDMAC engine will now wait for a DMA interface request from BIU. This request will be generated based on the programmed transmit threshold value. For the last bytes of data which can't be accessed using a burst, SINGLE transfers are performed on AHB master interface.
8) The IDMAC fetches the transmit data from the data buffer in the host memory and transfers to the FIFO for transmission to card.
9) When data spans across multiple descriptors, the IDMAC will fetch the next descriptor and continue with its operation with the next descriptor. The Last Descriptor bit in the descriptor indicates whether the data spans multiple descriptors or not.
10) When data transmission is complete, status information is updated in SDMMC_IDSTS register by setting Transmit Interrupt, if enabled. Also, the OWN bit is cleared by the IDMAC by performing a write transaction to DES0.

- Reception

IDMAC reception occurs as follows:

1) The host sets up the element (DES0-DES3) for reception, sets the OWN (DES0[31]).
2) The host programs the read data command in the SDMMC_CMD register in BIU.
3) The host will program the required receive threshold level (rx_wmark field in FIFOTH register).
4) The IDMAC determines that a read data transfer needs to be done as a consequence of step 2.
5) The IDMAC engine fetches the descriptor and checks the OWN bit. If the OWN bit is not set, it means that the host owns the descriptor. In this case the DMA enters suspend state and asserts the Descriptor Unable interrupt in the SDMMC_IDSTS register. In such a case, the host needs to release the IDMAC by writing any value to the poll demand register.
6) It will then wait for Command Done (CD) bit and no errors from BIU which indicates that a transfer can be done.
7) The IDMAC engine will now wait for a DMA interface request from BIU. This request will be generated based on the programmed receive threshold value. For the last bytes of data which can't be accessed using a burst, SINGLE transfers are performed on AHB.
8) The IDMAC fetches the data from the FIFO and transfer to host memory.
9) When data spans across multiple descriptors, the IDMAC will fetch the next descriptor and continue with its operation with the next descriptor. The Last Descriptor bit in the descriptor indicates whether the data spans multiple descriptors or not.
10) When data reception is complete, status information is updated in SDMMC_IDSTS register by setting Receive Interrupt, if enabled. Also, the OWN bit is cleared by the IDMAC by performing a write transaction to DES0.

- Interrupts

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Interrupts can be generated as a result of various events. SDMMC_IDSTS register contains all the bits that might cause an interrupt. SDMMC_IDINTEN register contains an enable bit for each of the events that can cause an interrupt.
There are two groups of summary interrupts-Normal and Abnormal-as outlined in SDMMC_IDSTS register. Interrupts are cleared by writing a 1 to the corresponding bit position. When all the enabled interrupts within a group are cleared, the corresponding summary bit is cleared. When both the summary bits are cleared, the interrupt signal dmac_intr_o is de-asserted.
Interrupts are not queued and if the interrupt event occurs before the driver has responded to it, no additional interrupts are generated. For example, Receive Interrupt—SDMMC_IDSTS[1] indicates that one or more data was transferred to the host buffer.
An interrupt is generated only once for simultaneous, multiple events. The driver must scan SDMMC_IDSTS register for the interrupt cause.

11.4 Register Description

Slave address can be divided into different length for different usage, which is shown as follows.

11.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
SDMMC_CTRL	0x0000	W	0x00000000	Control register
SDMMC_PWREN	0x0004	W	0x00000000	Power-enable register
SDMMC_CLKDIV	0x0008	W	0x00000000	Clock-divider register
SDMMC_CLKSRC	0x000c	W	0x00000000	SD clock source register
SDMMC_CLKENA	0x0010	W	0x00000000	Clock-enable register
SDMMC_TMOUT	0x0014	W	0xffffff40	Time-out register
SDMMC_CTYPE	0x0018	W	0x00000000	Card-type register
SDMMC_BLKSIZ	0x001c	W	0x00000200	Block-size register
SDMMC_BYTCNT	0x0020	W	0x00000200	Byte-count register
SDMMC_INTMASK	0x0024	W	0x00000000	Interrupt-mask register
SDMMC_CMDARG	0x0028	W	0x00000000	Command-argument register
SDMMC_CMD	0x002c	W	0x00000000	Command register
SDMMC_RESP0	0x0030	W	0x00000000	Response-0 register
SDMMC_RESP1	0x0034	W	0x00000000	Response-1 register
SDMMC_RESP2	0x0038	W	0x00000000	Response-2 register
SDMMC_RESP3	0x003c	W	0x00000000	Response-3 register
SDMMC_MINTSTS	0x0040	W	0x00000000	Masked interrupt-status register
SDMMC_RINTSTS	0x0044	W	0x00000000	Raw interrupt-status register
SDMMC_STATUS	0x0048	W	0x00000506	Status register
SDMMC_FIFOTH	0x004c	W	0x00000000	FIFO threshold register
SDMMC_CDETECT	0x0050	W	0x00000000	Card-detect register
SDMMC_WRTPRT	0x0054	W	0x00000000	Write-protect register
SDMMC_TCBCNT	0x005c	W	0x00000000	Transferred CIU card byte count
SDMMC_TBBCNT	0x0060	W	0x00000000	Transferred BIU FIFO byte count
SDMMC_DEBNCE	0x0064	W	0x00ffffff	Card detect debounce register
SDMMC_USRID	0x0068	W	0x07967797	User ID register
SDMMC_VERID	0x006c	W	0x5342270a	Version ID register
SDMMC_HCON	0x0070	W	0x04c47cc1	Hardware configuration register

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
SDMMC_UHSREG	0x0074	W	0x00000000	UHS-1 register
SDMMC_RSTN	0x0078	W	0x00000001	Hardware reset register
SDMMC_BMOD	0x0080	W	0x00000000	Bus mode register
SDMMC_PLDMND	0x0084	W	0x00000000	Poll demand register
SDMMC_DBADDR	0x0088	W	0x00000000	Descriptor list base address register
SDMMC_IDSTS	0x008c	W	0x00000000	Internal DMAC status register
SDMMC_IDINTEN	0x0090	W	0x00000000	Internal DMAC interrupt enable register
SDMMC_DSCADDR	0x0094	W	0x00000000	Current host descriptor address register
SDMMC_BUFADDR	0x0098	W	0x00000000	Current buffer descriptor address register
SDMMC_CARDTHRCTL	0x0100	W	0x00000000	Card read threshold enable
SDMMC_BACKEND_POWER	0x0104	W	0x00000000	Back-end power register
SDMMC_EMMCDDR_REG	0x010c	W	0x00000000	DDR start bit detection control register
SDMMC_RDYINT_GEN	0x0120	W	0x000000ff	Card ready interrupt generation register
SDMMC_FIFO_BASE	0x0200	W	0x00000000	FIFO base address register

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

11.4.2 Detail Register Description

SDMMC_CTRL

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:26	RO	0x0	reserved
25	RW	0x0	use_internal_dmac Present only for the Internal DMAC configuration; else, it is reserved. 1'b0: The host performs data transfers through the slave interface 1'b1: Internal DMAC used for data transfer
24:12	RO	0x0	reserved
11	RW	0x0	ceata_device_interrupt_status 1'b0: Interrupts not enabled in CE-ATA device 1'b1: Interrupts are enabled in CE-ATA device Software should appropriately write to this bit after power-on reset or any other reset to CE-ATA device. After reset, usually CE-ATA device interrupt is disabled. If the host enables CE-ATA device interrupt, then software should set this bit.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
10	RW	0x0	send_auto_stop_ccsd 1'b0: Clear bit if Host Controller does not reset the bit 1'b1: Send internally generated STOP after sending CCSD to CE-ATA device NOTE: Always set send_auto_stop_ccsd and send_ccsd bits together send_auto_stop_ccsd should not be set independent of send_ccsd. When set, the Host Controller automatically sends internally-generated STOP command (CMD12) to CE-ATA device. After sending internally-generated STOP command, Auto Command Done (ACD) bit in SDMMC_RINTSTS is set and generates interrupt to host if Auto Command Done interrupt is not masked. After sending the CCSD, the Host Controller automatically clears send_auto_stop_ccsd bit.
9	RW	0x0	send_ccsd 1'b0: Clear bit if Host Controller does not reset the bit 1'b1: Send Command Completion Signal Disable (CCSD) to CE-ATA device When set, the Host Controller sends CCSD to CE-ATA device. Software sets this bit only if current command is expecting CCS (that is, RW_BLK) and interrupts are enabled in CE-ATA device. Once the CCSD pattern is sent to device, Host Controller automatically clears send_ccsd bit. It also sets Command Done (CD) bit in SDMMC_RINTSTS register and generates interrupt to host if Command Done interrupt is not masked. NOTE: Once send_ccsd bit is set, it takes two card clock cycles to drive the CCSD on the CMD line. Due to this, during the boundary conditions it may happen that CCSD is sent to the CE-ATA device, even if the device signalled CCS.
8	RW	0x0	abort_read_data 1'b0: No change 1'b1: After suspend command is issued during read-transfer, software polls card to find when suspend happened. Once suspend occurs, software sets bit to reset data state-machine, which is waiting for next block of data. Bit automatically clears once data state machine resets to idle. Used in SDIO card suspend sequence.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7	RW	0x0	send_irq_response 1'b0: No change 1'b1: Send auto IRQ response Bit automatically clears once response is sent. To wait for MMC card interrupts, software issues CMD40, and the Host Controller waits for interrupt response from MMC card. In meantime, if software wants Host Controller to exit waiting for interrupt state, it can set this bit, at which time Host Controller command state-machine sends CMD40 response on bus and returns to idle state.
6	RW	0x0	read_wait 1'b0: Clear read wait 1'b1: Assert read wait For sending read-wait to SDIO cards.
5	RW	0x0	dma_enable 1'b0: Disable DMA transfer mode 1'b1: Enable DMA transfer mode Even when DMA mode is enabled, host can still push/pop data into or from FIFO; this should not happen during the normal operation. If there is simultaneous FIFO access from host/DMA, the data coherency is lost. Also, there is no arbitration inside SDMMC Controller to prioritize simultaneous host/DMA access.
4	RW	0x0	int_enable Global interrupt enable/disable bit 1'b0: Disable interrupts 1'b1: Enable interrupts The int port is 1 only when this bit is 1 and one or more unmasked interrupts are set.
3	RO	0x0	reserved
2	W1C	0x0	dma_reset 1'b0: No change 1'b1: Reset internal DMA interface control logic To reset DMA interface, firmware should set bit to 1. This bit is auto-cleared after two AHB clocks.
1	W1C	0x0	fifo_reset 1'b0: No change 1'b1: Reset to data FIFO To reset FIFO pointers To reset FIFO, firmware should set bit to 1. This bit is auto-cleared after completion of reset operation.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
0	W1C	0x0	controller_reset 1'b0: No change 1'b1: Reset Host Controller To reset controller, firmware should set bit to 1. This bit is auto-cleared after two AHB and two cclk_in clock cycles. This resets: a. BIU/CIU interface b. CIU and state machines c. abort_read_data, send_irq_response, and read_wait bits of Control register d. start_cmd bit of Command register Does not affect any registers or DMA interface, or FIFO or host interrupts.

SDMMC_PWREN

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	power_enable Power on/off switch for the card. Once power is turned on, firmware should wait for regulator/switch ramp-up time before trying to initialize card. 1'b0: Power off 1'b1: Power on Bit values output to card_power_en port.

SDMMC_CLKDIV

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	clk_divider0 Clock divider-0 value. Clock division is 2*n. For example, value of 0 means divide by 2*0 = 0 (no division, bypass), value of 1 means divide by 2*1 = 2, and so on. The recommended value is 0 or 1.

SDMMC_CLKSRC

Address: Operational Base + offset (0x000c)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1:0	RW	0x0	clk_source Clock divider source. 2'b00: Clock divider 0 The cclk_out is always from clock divider 0, and this register is not implemented.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

SDMMC_CLKENA

Address: Operational Base + offset (0x0010)

Bit	Attr	Reset Value	Description
31:17	RO	0x0	reserved
16	RW	0x0	cclk_low_power Low-power control for SD card clock and MMC card clock supported. 1'b0: Non-low-power mode 1'b1: Low-power mode. Stop clock when card in IDLE (should be normally set to only MMC and SD memory cards; for SDIO cards, if interrupts must be detected, clock should not be stopped).
15:1	RO	0x0	reserved
0	RW	0x0	cclk_enable Clock-enable control for SD card clock and MMC card clock supported. 1'b0: Clock disabled 1'b1: Clock enabled

SDMMC_TMOUT

Address: Operational Base + offset (0x0014)

Bit	Attr	Reset Value	Description
31:8	RW	0xffffff	data_timeout Value for card data read timeout; same value also used for data starvation by host timeout. Value is in number of card output clock. Note: The software timer should be used if the timeout value is in the order of 100 ms. In this case, read data timeout interrupt needs to be disabled.
7:0	RW	0x40	response_timeout Response timeout value. Value is in number of card output clock cclk_out.

SDMMC_CTYPE

Address: Operational Base + offset (0x0018)

Bit	Attr	Reset Value	Description
31:17	RO	0x0	reserved
16	RW	0x0	card_width_8 Indicates if card is 8-bit. 1'b0: Non 8-bit mode 1'b1: 8-bit mode
15:1	RO	0x0	reserved
0	RW	0x0	card_width Indicates if card is 1-bit or 4-bit. 1'b0: 1-bit mode 1'b1: 4-bit mode

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

SDMMC_BLKSIZ

Address: Operational Base + offset (0x001c)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RW	0x0200	block_size Block size

SDMMC_BYTCNT

Address: Operational Base + offset (0x0020)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000200	byte_count Number of bytes to be transferred; should be integer multiple of block size for block transfers. For undefined number of byte transfers, byte count should be set to 0. When byte count is set to 0, it is responsibility of host to explicitly send stop/abort command to terminate data transfer.

SDMMC_INTMASK

Address: Operational Base + offset (0x0024)

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24	RW	0x0	sdio_int_mask 1'b0: SDIO interrupt not masked 1'b1: SDIO interrupt masked
23:17	RO	0x0	reserved
16	RW	0x0	data_nobusy_int_mask 1'b0: Data no busy interrupt not masked 1'b1: Data no busy interrupt masked

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15:0	RW	0x0000	int_mask Bits used to mask unwanted interrupts. Value of 0 masks interrupt; value of 1 enables interrupt. bit 15: End-bit error (read)/Write no CRC (EBE) bit 14: Auto command done (ACD) bit 13: Start-bit error (SBE) bit 12: Hardware locked write error (HLE) bit 11: FIFO underrun/overrun error (FRUN) bit 10: Data starvation-by-host timeout (HTO) /Volt_switch_int bit 9: Data read timeout (DRTO) bit 8: Response timeout (RTO) bit 7: Data CRC error (DCRC) bit 6: Response CRC error (RCRC) bit 5: Receive FIFO data request (RXDR) bit 4: Transmit FIFO data request (TXDR) bit 3: Data transfer over (DTO) bit 2: Command done (CD) bit 1: Response error (RE) bit 0: Card detect (CD)

SDMMC_CMDARG

Address: Operational Base + offset (0x0028)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	cmd_arg Value indicates command argument to be passed to card

SDMMC_CMD

Address: Operational Base + offset (0x002c)

Bit	Attr	Reset Value	Description
31	RW	0x0	start_cmd Start command. Once command is taken by CIU, bit is cleared. When bit is set, host should not attempt to write to any command registers. If write is attempted, hardware lock error is set in raw interrupt register. Once command is sent and response is received from SD_MMC cards, Command Done bit is set in raw interrupt register.
30	RO	0x0	reserved
29	RW	0x0	use_hold_reg Use hold register. 1'b0: CMD and DATA sent to card bypassing hold register 1'b1: CMD and DATA sent to card through the hold register

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
28	RW	0x0	volt_switch Voltage switch bit. 1'b0: No voltage switching 1'b1: Voltage switching enabled; must be set for CMD11 only
27	RW	0x0	boot_mode Boot mode selection. 1'b0: Mandatory boot operation 1'b1: Alternate boot operation
26	RW	0x0	disable_boot Disable boot. When software sets this bit along with start_cmd, CIU terminates the boot operation. Do not set disable_boot and enable_boot together.
25	RW	0x0	expect_boot_ack Expect boot acknowledge. When software sets this bit along with enable_boot, CIU expects a boot acknowledge start pattern of 0-1-0 from the selected card.
24	RW	0x0	enable_boot Enable boot. This bit should be set only for mandatory boot mode. When Software sets this bit along with start_cmd, CIU starts the boot sequence for the corresponding card by asserting the CMD line low. Do not set disable_boot and enable_boot together.
23	RW	0x0	ccs_expected 1'b0: Interrupts are not enabled in CE-ATA device or command does not expect CCS from device 1'b1: Interrupts are enabled in CE-ATA device and RW_BLK command expects command completion signal from CE-ATA device If the command expects command completion signal (CCS) from the CE-ATA device, the software should set this control bit. The Host Controller sets data transfer over (DTO) bit in SDMMC_RINTSTS register and generates interrupt to host if data transfer over interrupt is not masked.
22	RW	0x0	read_ceata_device 1'b0: Host is not performing read access towards CE-ATA device 1'b1: Host is performing read access towards CE-ATA device Software should set this bit to indicate that CE-ATA device is being accessed for read transfer. This bit is used to disable read data timeout indication while performing CE-ATA read transfers. Maximum value of I/O transmission delay can be no less than 10 seconds. The Host Controller should not indicate read data timeout while waiting for data from CE-ATA device.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
21	RW	0x0	update_clock_regs_only 1'b0: Normal command sequence 1'b1: Do not send commands, just update clock register value into card clock domain Following register values transferred into card clock domain: SDMMC_CLKDIV, SDMMC_CLRSRC, SDMMC_CLKENA. Changes card clocks (change frequency, truncate off or on, and set low-frequency mode); provided in order to change clock frequency or stop clock without having to send command to cards. During normal command sequence, when update_clock_regs_only = 0, following control registers are transferred from BIU to CIU: SDMMC_CMD, SDMMC_CMDARG, SDMMC_TMOUT, SDMMC_CTYPE, SDMMC_BLKSIZ, SDMMC_BYTCNT. CIU uses new register values for new command sequence to card. When bit is set, there are no Command Done interrupts because no command is sent to SD_MMC_CEATA cards.
20:16	RO	0x0	reserved
15	RW	0x0	send_initialization 1'b0: Do not send initialization sequence (80 clocks of 1) before sending this command 1'b1: Send initialization sequence before sending this command After power on, 80 clocks must be sent to card for initialization before sending any commands to card. Bit should be set while sending first command to card so that controller will initialize clocks before sending command to card. This bit should not be set for either of the boot modes (alternate or mandatory).
14	RW	0x0	stop_abort_cmd 1'b0: Neither stop nor abort command to stop current data transfer in progress. If abort is sent to function-number currently selected or not in data-transfer mode, then bit should be set to 0. 1'b1: Stop or abort command intended to stop current data transfer in progress. When open-ended or predefined data transfer is in progress, and host issues stop or abort command to stop data transfer, bit should be set so that command/data state-machines of CIU can return correctly to idle state. This is also applicable for Boot mode transfers. To Abort boot mode, this bit should be set along with SDMMC_CMD[26]=disable_boot.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
13	RW	0x0	wait_prvdata_complete 1'b0: Send command at once, even if previous data transfer has not completed 1'b1: Wait for previous data transfer completion before sending command The wait_prvdata_complete=0 option typically used to query status of card during data transfer or to stop current data transfer; card_number should be same as in previous command.
12	RW	0x0	send_auto_stop 1'b0: No stop command sent at end of data transfer 1'b1: Send stop command at end of data transfer When set, the Host Controller sends stop command to card at end of data transfer. a. When send_auto_stop bit should be set, since some data transfers do not need explicit stop commands b. Open-ended transfers that software should explicitly send to stop command Additionally, when "resume" is sent to resume-suspended memory access of SD-Combo card, bit should be set correctly if suspended data transfer needs send_auto_stop. Don't care if no data expected from card.
11	RW	0x0	transfer_mode 1'b0: Block data transfer command 1'b1: Stream data transfer command Don't care if no data expected.
10	RW	0x0	wr 1'b0: Read from card 1'b1: Write to card Don't care if no data expected from card.
9	RW	0x0	data_expected 1'b0: No data transfer expected (read/write) 1'b1: Data transfer expected (read/write)
8	RW	0x0	check_response_crc 1'b0: Do not check response CRC 1'b1: Check response CRC Some of command responses do not return valid CRC bits. Software should disable CRC checks for those commands in order to disable CRC checking by controller.
7	RW	0x0	response_length 1'b0: Short response expected from card 1'b1: Long response expected from card
6	RW	0x0	response_expect 1'b0: No response expected from card 1'b1: Response expected from card

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5:0	RW	0x00	cmd_index Command index

SDMMC_RESP0

Address: Operational Base + offset (0x0030)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	response0 Bit[31:0] of response

SDMMC_RESP1

Address: Operational Base + offset (0x0034)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	response Register represents bit[63:32] of long response. When CIU sends auto-stop command, then response is saved in register. Response for previous command sent by host is still preserved in response 0 register. Additional auto-stop issued only for data transfer commands, and response type is always "short" for them.

SDMMC_RESP2

Address: Operational Base + offset (0x0038)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	response2 Bit[95:64] of long response

SDMMC_RESP3

Address: Operational Base + offset (0x003c)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	response3 Bit[127:96] of long response

SDMMC_MINTSTS

Address: Operational Base + offset (0x0040)

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24	RO	0x0	sdio_interrupt SDIO interrupt status when sdio_int_mask is set.
23:17	RO	0x0	reserved
16	RW	0x0	data_nobusy_int_status Data no busy interrupt status when data_nobusy_int_mask is set.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15:0	RO	0x0000	int_status Interrupt enabled only if corresponding bit in interrupt mask register is set. bit 15: End-bit error (read)/Write no CRC (EBE) bit 14: Auto command done (ACD) bit 13: Start-bit error (SBE) bit 12: Hardware locked write error (HLE) bit 11: FIFO underrun/overrun error (FRUN) bit 10: Data starvation-by-host timeout (HTO) /Volt_switch_int bit 9: Data read timeout (DRTO) bit 8: Response timeout (RTO) bit 7: Data CRC error (DCRC) bit 6: Response CRC error (RCRC) bit 5: Receive FIFO data request (RXDR) bit 4: Transmit FIFO data request (TXDR) bit 3: Data transfer over (DTO) bit 2: Command done (CD) bit 1: Response error (RE) bit 0: Card detect (CD)

SDMMC_RINTSTS

Address: Operational Base + offset (0x0044)

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24	R/WSC	0x0	sdio_interrupt Raw SDIO interrupt status. Write value of 1 clears this bit, and value of 0 leaves bit intact.
23:17	RO	0x0	reserved
16	R/WSC	0x0	data_nobusy_int_status Raw data no busy interrupt status. Write value of 1 clears this bit, and value of 0 leaves bit intact.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15:0	R/WSC	0x0000	int_status Raw interrupt status. Writes to bits clear status bit. Write value of 1 clears status bit, and value of 0 leaves bit intact. bit 15: End-bit error (read)/Write no CRC (EBE) bit 14: Auto command done (ACD) bit 13: Start-bit error (SBE) bit 12: Hardware locked write error (HLE) bit 11: FIFO underrun/overrun error (FRUN) bit 10: Data starvation-by-host timeout (HTO) /Volt_switch_int bit 9: Data read timeout (DRTO) bit 8: Response timeout (RTO) bit 7: Data CRC error (DCRC) bit 6: Response CRC error (RCRC) bit 5: Receive FIFO data request (RXDR) bit 4: Transmit FIFO data request (TXDR) bit 3: Data transfer over (DTO) bit 2: Command done (CD) bit 1: Response error (RE) bit 0: Card detect (CD)

SDMMC_STATUS

Address: Operational Base + offset (0x0048)

Bit	Attr	Reset Value	Description
31	RO	0x0	dma_req DMA request signal state
30	RO	0x0	dma_ack DMA acknowledge signal state
29:17	RO	0x0000	fifo_count Number of filled locations in FIFO
16:11	RO	0x00	response_index Index of previous response, including any auto-stop sent by core.
10	RO	0x1	data_state_mc_busy Data transmit or receive state-machine is busy
9	RO	0x0	data_busy Inverted version of raw selected card_data[0] 1'b0: Card data not busy 1'b1: Card data busy
8	RO	0x1	data_3_status Raw selected card_data[3]; checks whether card is present 1'b0: Card not present 1'b1: Card present

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7:4	RO	0x0	command_fsm_states Command FSM states: 4'h0: Idle 4'h1: Send init sequence 4'h2: Tx cmd start bit 4'h3: Tx cmd tx bit 4'h4: Tx cmd index + arg 4'h5: Tx cmd crc7 4'h6: Tx cmd end bit 4'h7: Tx resp start bit 4'h8: Rx resp IRQ response 4'h9: Rx resp tx bit 4'ha: Rx resp cmd idx 4'hb: Rx resp data 4'hc: Rx resp crc7 4'hd: Rx resp end bit 4'he: Cmd path wait NCC 4'hf: Wait; CMD-to-response turnaround The command FSM state is represented using 19 bits. The SDMMC_STATUS register[7:4] has 4 bits to represent the command FSM states. Using these 4 bits, only 16 states can be represented. Thus three states cannot be represented in the SDMMC_STATUS[7:4] register. The three states that are not represented in the SDMMC_STATUS Register[7:4] are: Bit 16: Wait for CCS Bit 17: Send CCSD Bit 18: Boot Mode Due to this, while command FSM is in "Wait for CCS state" or "Send CCSD" or "Boot Mode", the SDMMC_STATUS register indicates status as 0 for the bit field [7:4].
3	RO	0x0	fifo_full FIFO is full status
2	RO	0x1	fifo_empty FIFO is empty status
1	RO	0x1	fifo_tx_watermark FIFO reached Transmit watermark level; not qualified with data transfer.
0	RO	0x0	fifo_rx_watermark FIFO reached Receive watermark level; not qualified with data transfer.

SDMMC_FIFOTH

Address: Operational Base + offset (0x004c)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
30:28	RW	0x0	dma_mutiple_transaction_size Burst size of multiple transaction; should be programmed same as DMA controller multiple-transaction-size SRC/DEST_MSIZE. 3'b000: 1 transfers 3'b001: 4 transfers 3'b010: 8 transfers 3'b011: 16 transfers 3'b100: 32 transfers 3'b101: 64 transfers 3'b110: 128 transfers 3'b111: 256 transfers The unit for transfer is 32bits.
27:16	RW	0x000	rx_wmark FIFO threshold watermark level when receiving data to card. When FIFO data count reaches greater than this number, DMA/FIFO request is raised. During end of packet, request is generated regardless of threshold programming in order to complete any remaining data. In non-DMA mode, when receiver FIFO threshold (RXDR) interrupt is enabled, then interrupt is generated instead of DMA request. During end of packet, interrupt is not generated if threshold programming is larger than any remaining data. It is responsibility of host to read remaining bytes on seeing Data Transfer Done interrupt. In DMA mode, at end of packet, even if remaining bytes are less than threshold, DMA request does single transfers to flush out any remaining bytes before Data Transfer Done interrupt is set. 12 bits-1 bit less than FIFO-count of status register, which is 13 bits. Limitation: rx_wmark <= FIFO_DEPTH-2 Recommended: (FIFO_DEPTH/2) - 1; (means greater than (FIFO_DEPTH/2) - 1) NOTE: In DMA mode during CCS time-out, the DMA does not generate the request at the end of packet, even if remaining bytes are less than threshold. In this case, there will be some data left in the FIFO. It is the responsibility of the application to reset the FIFO after the CCS timeout.
15:12	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
11:0	RW	0x000	tx_wmark FIFO threshold watermark level when transmitting data to card. When FIFO data count is less than or equal to this number, DMA/FIFO request is raised. If Interrupt is enabled, then interrupt occurs. During end of packet, request or interrupt is generated, regardless of threshold programming. In non-DMA mode, when transmit FIFO threshold (TXDR) interrupt is enabled, then interrupt is generated instead of DMA request. During end of packet, on last interrupt, host is responsible for filling FIFO with only required remaining bytes (not before FIFO is full or after CIU completes data transfers, because FIFO may not be empty). In DMA mode, at end of packet, if last transfer is less than burst size, DMA controller does single cycles until required bytes are transferred. 12 bits -1 bit less than FIFO-count of status register, which is 13 bits. Limitation: tx_wmark >= 1; Recommended: FIFO_DEPTH/2; (means less than or equal to FIFO_DEPTH/2)

SDMMC_CDETECT

Address: Operational Base + offset (0x0050)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RO	0x0	card_detect_n Value on card_detect_n input port. 1'b0: Represents presence of card 1'b1: Represents absence of card

SDMMC_WRTPRT

Address: Operational Base + offset (0x0054)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	write_protect Value on card_write_prt input port. 1 represents write protection.

SDMMC_TCBCNT

Address: Operational Base + offset (0x005c)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	trans_card_byte_count Number of bytes transferred by CIU unit to card.

SDMMC_TBBCNT

Address: Operational Base + offset (0x0060)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	trans_fifo_byte_count Number of bytes transferred between host/DMA memory and BIU FIFO.

SDMMC_DEBNCE

Address: Operational Base + offset (0x0064)

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:0	RW	0xffffff	debounce_count Number of host clock used by debounce filter logic; typical debounce time is 5-25 ms.

SDMMC_USRID

Address: Operational Base + offset (0x0068)

Bit	Attr	Reset Value	Description
31:0	RW	0x07967797	usrid User identification register; value set by user. Default reset value can be picked by user while configuring core before synthesis. Can also be used as scratch pad register by user. The default value is determined by configuration value.

SDMMC_VERID

Address: Operational Base + offset (0x006c)

Bit	Attr	Reset Value	Description
31:0	RO	0x5342270a	verid Version identification register, register value is hard-wired. Can be read by firmware to support different versions of core.

SDMMC_HCON

Address: Operational Base + offset (0x0070)

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved
26	RO	0x1	area_optimized 1'b0: No area optimization 1'b1: Area optimization
25:24	RO	0x0	num_clk_div divider number-1.
23	RO	0x1	set_clk_false_path 1'b0: No false path 1'b1: False path set
22	RO	0x1	impl_hold_reg 1'b0: No hold register 1'b1: Hold register

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
21	RO	0x0	fifo_ram_inside 1'b0: Outside 1'b1: Inside
20:18	RO	0x1	ge_dma_data_width 3'b000: 16 bits 3'b001: 32 bits 3'b010: 64 bits others: Reserved
17:16	RO	0x0	dma_interface 2'b00: None 2'b01: DW_DMA 2'b10: GENERIC_DMA 2'b11: NON-DW-DMA
15:10	RO	0x1f	h_addr_width 6'h8: 9 bits 6'h9: 10 bits … 6'h1f: 32 bits others: Reserved
9:7	RO	0x1	h_data_width 3'b000: 16 bits 3'b001: 32 bits 3'b010: 64 bits others: Reserved
6	RO	0x1	h_bus_type 1'b0: APB 1'b1: AHB
5:1	RO	0x00	card_num Card number -1.
0	RO	0x1	card_type Card type. 1'b0: MMC_ONLY 1'b1: SD_MMC

SDMMC_UHSREG

Address: Operational Base + offset (0x0074)

Bit	Attr	Reset Value	Description
31:17	RO	0x0	reserved
16	RW	0x0	ddr_reg DDR mode. Determines the voltage fed to the buffers by an external voltage regulator. 1'b0: Non-DDR mode 1'b1: DDR mode
15:0	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

SDMMC_RSTN

Address: Operational Base + offset (0x0078)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x1	card_reset Hardware reset. 1'b0: Active mode 1'b1: Reset These bits cause the cards to enter pre-idle state, which requires them to be re-initialized.

SDMMC_BMOD

Address: Operational Base + offset (0x0080)

Bit	Attr	Reset Value	Description
31:11	RO	0x0	reserved
10:8	RO	0x0	PBL Programmable burst length. These bits indicate the maximum number of beats to be performed in one IDMAC transaction. The IDMAC will always attempt to burst as specified in PBL each time it starts a Burst transfer on the host bus. The permissible values are 1, 4, 8, 16, 32, 64, 128 and 256. This value is the mirror of MSIZE of SDMMC_FIFOTH register. In order to change this value, write the required value to SDMMC_FIFOTH register. This is an encode value as follows. 3'b000: 1 transfers 3'b001: 4 transfers 3'b010: 8 transfers 3'b011: 16 transfers 3'b100: 32 transfers 3'b101: 64 transfers 3'b110: 128 transfers 3'b111: 256 transfers Transfer unit is 32 bits. PBL is a read-only value and is applicable only for data access; it does not apply to descriptor accesses.
7	RW	0x0	DE IDMAC enable. When set, the IDMAC is enabled.
6:2	RW	0x00	DSL Descriptor skip length. Specifies the number of Word to skip between two unchained descriptors. This is applicable only for dual buffer structure.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
1	RW	0x0	fb Fixed burst. Controls whether the AHB Master interface performs fixed burst transfers or not. When set, the AHB will use only SINGLE, INCR4, INCR8 or INCR16 during start of normal burst transfers. When reset, the AHB will use SINGLE and INCR burst transfer operations.
0	RW	0x0	swr Software reset. When set, the DMA Controller resets all its internal registers. It is automatically cleared after 1 clock cycle.

SDMMC_PLDMND

Address: Operational Base + offset (0x0084)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	pd Poll demand. If the OWN bit of a descriptor is not set, the FSM goes to the Suspend state. The host needs to write any value into this register for the IDMAC FSM to resume normal descriptor fetch operation.

SDMMC_DBADDR

Address: Operational Base + offset (0x0088)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sbl Start of descriptor list. Contains the base address of the first descriptor. The LSB bits[1:0] are ignored and taken as all-zero by the IDMAC internally. Hence these LSB bits are read-only.

SDMMC_IDSTS

Address: Operational Base + offset (0x008c)

Bit	Attr	Reset Value	Description
31:17	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
16:13	RO	0x0	fsm DMAC FSM present state. 4'h0: DMA_IDLE 4'h1: DMA_SUSPEND 4'h2: DESC_RD 4'h3: DESC_CHK 4'h4: DMA_RD_REQ_WAI 4'h5: DMA_WR_REQ_WAI 4'h6: DMA_RD 4'h7: DMA_WR 4'h8: DESC_CLOSE Others: Reserved
12:10	RO	0x0	eb Error bits. Indicates the type of error that caused a bus error. Valid only with fatal bus. 3'h1: Host abort received during transmission 3'h2: Host abort received during reception Others: Reserved
9	RW	0x0	ais Abnormal interrupt summary. Logical OR of the following: SDMMC_IDSTS[2] fatal bus interrupt SDMMC_IDSTS[4] du bit 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. Writing a 1 clears this bit.
8	RW	0x0	nis Normal interrupt summary. Logical OR of the following: SDMMC_IDSTS[0] transmit interrupt SDMMC_IDSTS[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. Writing a 1 clears this bit.
7:6	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5	RW	0x0	ces Card error summary. Indicates the status of the transaction to/from the card; also present in SDMMC_RINTSTS. Indicates the logical OR of the following bits: EBE: End Bit Error RTO: Response Timeout/Boot Ack Timeout RCRC: Response CRC SBE: Start Bit Error DRTO: Data Read Timeout/BDS timeout DCRC: Data CRC for Receive RE: Response Error Writing a 1 clears this bit. The abort condition of the IDMAC depends on the setting of this CES bit. If the CES bit is enabled, then the IDMAC aborts on a "response error"; however, it will not abort if the CES bit is cleared.
4	RW	0x0	dui Descriptor unavailable interrupt. This bit is set when the descriptor is unavailable due to OWN bit = 0 (DES0[31] =0). Writing a 1 clears this bit.
3	RO	0x0	reserved
2	RW	0x0	fbe Fatal bus error interrupt. Indicates that a bus error occurred (SDMMC_IDSTS[12:10]). When this bit is set, the DMA disables all its bus accesses. Writing a 1 clears this bit.
1	RW	0x0	ri Receive interrupt. Indicates the completion of data reception for a descriptor. Writing a 1 clears this bit.
0	RW	0x0	ti Transmit interrupt. Indicates that data transmission is finished for a descriptor. Writing 1 clears this bit.

SDMMC_IDINTEN

Address: Operational Base + offset (0x0090)

Bit	Attr	Reset Value	Description
31:10	RO	0x0	reserved
9	RW	0x0	ai Abnormal interrupt summary enable. When set, an abnormal interrupt is enabled. This bit enables the following bits: SDMMC_IDINTEN[2] fatal bus error interrupt SDMMC_IDINTEN[4] du interrupt

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
8	RW	0x0	ni Normal interrupt summary enable. When set, a normal interrupt is enabled. When reset, a normal interrupt is disabled. This bit enables the following bits: SDMMC_IDINTEN[0] transmit interrupt SDMMC_IDINTEN[1] receive interrupt
7:6	RO	0x0	reserved
5	RW	0x0	ces Card error summary interrupt enable. When set, it enables the card interrupt summary.
4	RW	0x0	du Descriptor unavailable interrupt. When set along with abnormal interrupt summary enable, the du interrupt is enabled.
3	RO	0x0	reserved
2	RW	0x0	fbe Fatal bus error enable. When set with abnormal interrupt summary enable, the fatal bus error interrupt is enabled. When reset, fatal bus error enable interrupt is disabled.
1	RW	0x0	ri Receive interrupt enable. When set with normal interrupt summary enable, receive interrupt is enabled. When reset, receive interrupt is disabled.
0	RW	0x0	ti Transmit interrupt enable. When set with normal interrupt summary enable, transmit interrupt is enabled. When reset, transmit interrupt is disabled.

SDMMC_DSCADDR

Address: Operational Base + offset (0x0094)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	hda Host descriptor address pointer. This register points to the start address of the current descriptor read by the IDMAC. Cleared on reset. Pointer updated by IDMAC during operation.

SDMMC_BUFADDR

Address: Operational Base + offset (0x0098)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	hba Host buffer address pointer. This register points to the current data buffer address being accessed by the IDMAC. Cleared on Reset. Pointer updated by IDMAC during operation.

SDMMC_CARDTHRCTL

Address: Operational Base + offset (0x0100)

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:16	RW	0x000	card_rd_thres Card read threshold size
15:2	RO	0x0	reserved
1	RW	0x0	busy_clr_int_en Busy clear interrupt. 1'b0: Busy clear interrupt disabled 1'b1: Busy clear interrupt enabled Note: The application can disable this feature if it does not want to wait for a busy clear interrupt. For example, in a multi-card scenario, the application can switch to the other card without waiting for a busy to be completed. In such cases, the application can use the polling method to determine the status of busy. By default this feature is disabled and backward-compatible to the legacy drivers where polling is used.
0	RW	0x0	card_rd_thres_en Card read threshold enable. 1'b0: Card read threshold disabled 1'b1: Card read threshold enabled. Host Controller initiates read transfer only if card_rd_thres amount of space is available in receive FIFO.

SDMMC_BACKEND_POWER

Address: Operational Base + offset (0x0104)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	back_end_power Back end power 1'b0: Off; Reset 1'b1: Back-end power supplied to card application

SDMMC_EMMCDDR_REG

Address: Operational Base + offset (0x010c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	half_start_bit Control for start bit detection mechanism inside Host Controller based on duration of start bit. For eMMC 4.5, start bit can be: 1'b0: Full cycle (half_start_bit=0) 1'b1: Less than one full cycle (half_start_bit=1) Set half_start_bit=1 for eMMC 4.5 and above; set to 0 for SD applications.

SDMMC_RDYINT_GEN

Address: Operational Base + offset (0x0120)

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24	RO	0x0	rdyint_cnt_finish Counter finish indication. When high, it indicates that the rdyint counter is finished.
23:16	RO	0x00	rdyint_cnt_status Couner status, reflect internal counter value.
15:9	RO	0x0	reserved
8	RW	0x0	rdyint_gen_working Working indication for rdyint generator. When high, Host Controller start to count and generate one rdyint trigger. After the rdyint trigger is generated, this bit will be set to 0 by Host Controller. So software should set it to 1 before detecting next interrupt.
7:0	RW	0xff	rdyint_gen_maxval Max counter value to detect cdata_in0 high value for generating rdyint, based on internal clock frequency.

SDMMC_FIFO_BASE

Address: Operational Base + offset (0x0200)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	fifo_base_addr FIFO base address

11.5 Interface Description

The interface and IOMUX setting for SDMMC, SDIO, EMMC are shown as follows.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Table 11-8 SDMMC Interface Description

Module Pin	Direction	Pad Name	IOMUX Setting
sdmmc_cclk	O	GPIO4_A1/SDMMC0_CLK	GRF_GPIO4A_IOMUX_SEL_L [7:4]=4'h1
sdmmc_ccmd	I/O	GPIO4_A0/SDMMC0_CMD/TEST_CLK0	GRF_GPIO4A_IOMUX_SEL_L [3:0]=4'h1
sdmmc_cdata0	I/O	GPIO4_A2/SDMMC0_D0/UART2_TX_M0	GRF_GPIO4A_IOMUX_SEL_L [11:8]=4'h1
sdmmc_cdata1	I/O	GPIO4_A3/SDMMC0_D1/UART2_RX_M0	GRF_GPIO4A_IOMUX_SEL_L [15:12]=4'h1
sdmmc_cdata2	I/O	GPIO4_A4/SDMMC0_D2/JTAG_TCK	GRF_GPIO4A_IOMUX_SEL_H [3:0]=4'h1
sdmmc_cdata3	I/O	GPIO4_A5/SDMMC0_D3/JTAG_TMS	GRF_GPIO4A_IOMUX_SEL_H [7:4]=4'h1
sdmmc_cdetn	I	GPIO0_A3/PCIE_CLKREQN_M0/SDMMC0_DETN	PMUGRF_GPIO0A_IOMUX [7:6]=2'h1

Notes: I=input, O=output, I/O=input/output, bidirectional

Table 11-9 SDIO Interface Description

Module Pin	Direction	Pad Name	IOMUX Setting
sdio_cclk	O	GPIO4_A7/SDMMC1_CLK	GRF_GPIO4A_IOMUX_SEL_H [15:12]=4'h1
sdio_ccmd	I/O	GPIO4_A6/SDMMC1_CMD	GRF_GPIO4A_IOMUX_SEL_H [11:8]=4'h1
sdio_cdata0	I/O	GPIO4_B0/SDMMC1_D0/UART1_RX_M0	GRF_GPIO4B_IOMUX_SEL_L [3:0]=4'h1
sdio_cdata1	I/O	GPIO4_B1/SDMMC1_D1/UART1_TX_M0	GRF_GPIO4B_IOMUX_SEL_L [7:4]=4'h1
sdio_cdata2	I/O	GPIO4_B2/SDMMC1_D2/UART1_CTS	GRF_GPIO4B_IOMUX_SEL_L [11:8]=4'h1
sdio_cdata3	I/O	GPIO4_B3/SDMMC1_D3/UART1_RTS	GRF_GPIO4B_IOMUX_SEL_L [15:12]=4'h1

Notes: I=input, O=output, I/O=input/output, bidirectional

Table 11-10 EMMC Interface Description

Module Pin	Direction	Pad Name	IOMUX Setting
emmc_cclk	O	GPIO1_B1/EMMC_CLKOUT/SPI2_CSN_M0	GRF_GPIO1B_IOMUX_SEL_L [7:4]=4'h1
emmc_ccmd	I/O	GPIO1_B2/EMMC_CMD	GRF_GPIO1B_IOMUX_SEL_L [11:8]=4'h1
emmc_cdata0	I/O	GPIO1_A0/EMMC_D0/SFC_SIO0	GRF_GPIO1A_IOMUX_SEL_L [3:0]=4'h1
emmc_cdata1	I/O	GPIO1_A1/EMMC_D1/SFC_SIO1	GRF_GPIO1A_IOMUX_SEL_L [7:4]=4'h1
emmc_cdata2	I/O	GPIO1_A2/EMMC_D2/SFC_SIO2	GRF_GPIO1A_IOMUX_SEL_L [11:8]=4'h1
emmc_cdata3	I/O	GPIO1_A3/EMMC_D3/SFC_SIO3	GRF_GPIO1A_IOMUX_SEL_L [15:12]=4'h1
emmc_cdata4	I/O	GPIO1_A4/EMMC_D4/SFC_CSN0	GRF_GPIO1A_IOMUX_SEL_H [3:0]=4'h1
emmc_cdata5	I/O	GPIO1_A5/EMMC_D5/SFC_CLK	GRF_GPIO1A_IOMUX_SEL_H [7:4]=4'h1
emmc_cdata6	I/O	GPIO1_A6/EMMC_D6/SPI2_MISO_M0	GRF_GPIO1A_IOMUX_SEL_H [11:8]=4'h1
emmc_cdata7	I/O	GPIO1_A7/EMMC_D7/SPI2_CLK_M0	GRF_GPIO1A_IOMUX_SEL_H [15:12]=4'h1
emmc_pwren	O	GPIO1_B0/EMMC_PWREN/SPI2_MOSI_M0	GRF_GPIO1B_IOMUX_SEL_L [3:0]=4'h1
emmc_rstn	O	GPIO1_B3/EMMC_RSTN	GRF_GPIO1B_IOMUX_SEL_L [15:12]=4'h1

Notes: I=input, O=output, I/O=input/output, bidirectional

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

11.6 Application Notes

11.6.1 Card-Detect and Write-Protect Mechanism

Following figure illustrates how the SD/MMC card detection and write-protect signals are connected. Most of the SD/MMC sockets have card-detect pins. When no card is present, card_detect_n is 1 due to the pull-up. When the card is inserted, the card-detect pin is shorted to ground, which makes card_detect_n go to 0. Similarly in SD cards, when the write-protect switch is toward the left, it shorts the write_protect port to ground.

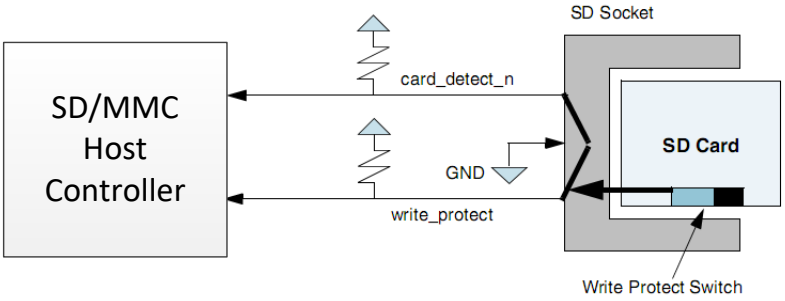


Fig. 11-9 SD/MMC Card-Detect and Write-Protect

11.6.2 SD/MMC Termination Requirement

Following Figure illustrates the SD/MMC termination requirements, which is required to pull up ccmd and cdata lines on the device bus. The recommended specification for pull-up on the ccmd line (Rcmd) is 4.7K-100K for MMC, and 10K-100K for an SD. The recommended pull-up on the cdata line (Rdat) is 50K-100K.

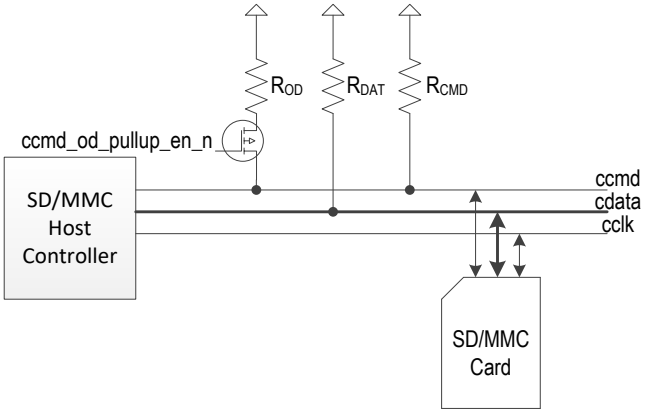


Fig. 11-10 SD/MMC Card Termination

11.6.3 Software/Hardware Restriction

Before issuing a new data transfer command, the software should ensure that the card is not busy due to any previous data transfer command. Before changing the card clock frequency, the software must ensure that there are no data or command transfers in progress.

To avoid glitches in the card clock outputs, the software should use the following steps when changing the card clock frequency:

1) Before disable clock, ensure that the card is not busy due to any previous data command. To determine this, check for 0 in bit 9 of SDMMC_STATUS register.

2) Update the SDMMC_CLKENA register to disable clock. To ensure completion of any previous command before this update, send a command to the CIU to update the clock registers by setting:

- start_cmd bit
- "update clock registers only" bit
- "wait_previous data complete" bit

Wait for the CIU to take the command by polling for 0 on the start_cmd bit.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

3) Set the start_cmd bit to update the SDMMC_CLKDIV and/or SDMMC_CLKSRC registers, and send a command to the CIU in order to update the clock registers; wait for the CIU to take the command.
4) Set start_cmd to update the SDMMC_CLKENA register in order to enable the required clocks and send a command to the CIU to update the clock registers; wait for the CIU to take the command.
In non-DMA mode, while reading from a card, the Data Transfer Over (SDMMC_RINTSTS[3]) interrupt occurs as soon as the data transfer from the card is over. There still could be some data left in the FIFO, and the rx_wmark interrupt may or may not occur, depending on the remaining bytes in the FIFO. Software should read any remaining bytes upon seeing the Data Transfer Over (DTO) interrupt. While using the external DMA interface for reading from a card, the DTO interrupt occurs only after all the data is flushed to memory by the DMA interface unit.
While writing to a card in external DMA mode, if an undefined-length transfer is selected by setting the SDMMC_BYTCNT register to 0, the DMA logic will likely request more data than it will send to the card, since it has no way of knowing at which point the software will stop the transfer. The DMA request stops as soon as the DTO is set by the CIU.
If the software issues a controller_reset command by setting SDMMC_CTRL register bit[0] to 1, all the CIU state machines are reset; the FIFO is not cleared. The DMA sends all remaining bytes to the host. In addition to a card-reset, if a FIFO reset is also issued, then:

- Any pending DMA transfer on the bus completes correctly
- DMA data read is ignored
- Write data is unknown(x)

Additionally, if dma_reset is also issued, any pending DMA transfer is abruptly terminated. When the DMA is used, the DMA controller channel should also be reset and reprogrammed.
If any of the previous data commands do not properly terminate, then the software should issue the FIFO reset in order to remove any residual data, if any, in the FIFO. After asserting the FIFO reset, you should wait until this bit is cleared.
One data-transfer requirement between the FIFO and host is that the number of transfers should be a multiple of the FIFO data width (32bits). For example, you want to write only 15 bytes to an SD/MMC card (SDMMC_BYTCNT), the host should write 16 bytes to the FIFO or program the DMA to do 16-byte transfers. The software can still program the SDMMC_BYTCNT register to only 15, at which point only 15 bytes will be transferred to the card. Similarly, when 15 bytes are read from a card, the host should still read all 16 bytes from the FIFO.
It is recommended that you not change the FIFO threshold register in the middle of data transfers.

11.6.4 Programming Sequence

1. Initialization

Following figure illustrates the initialization flow.

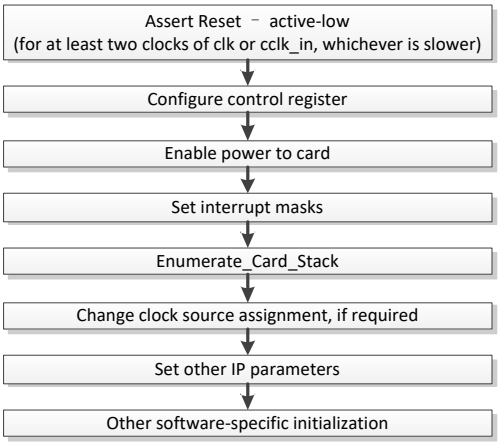


Fig. 11-11 Host Controller Initialization Sequence

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Once the power and clocks are stable, reset_n should be asserted(active-low) for at least two clocks of clk or cclk_in, whichever is slower. The reset initializes the registers, ports, FIFO-pointers, DMA interface controls, and state-machines in the design. After power-on reset, the software should do the following:

1) Configure control register
2) Enable power to cards – Before enabling the power, confirm that the voltage setting to the voltage regulators is correct. Enable power to the connected cards by setting the corresponding bit to 1 in the Power Enable register. Wait for the power ramp-up time.
3) Set masks for interrupts by clearing appropriate bits in the Interrupt Mask register. Set the global int_enable bit of the Control register. It is recommended that you write 0xffff_ffff to the Raw Interrupt register in order to clear any pending interrupts before setting the int_enable bit.
4) Enumerate card stack – Each card is enumerated according to card type; for details, refer to "Enumerated Card Stack". For enumeration, you should restrict the clock frequency to 400KHz.
5) Changing clock source assignment – set the card frequency using the clock-divider and clock-source registers; for details, refer to "Clock Programming". MMC cards operate at a maximum of 20MHz (at maximum of 52MHz in high-speed mode). SD mode operates at a maximum of 25MHz (at maximum of 50MHz in high-speed mode).
6) Set other parameters, which normally do not need to be changed with every command, with a typical value such as timeout values in cclk_out according to SD/MMC specifications.

- ResponseTimeOut = 0x64
- DataTimeOut = highest of one of the following:
- (10*((TAAC*Fop)+(100*NSAC))
- Host FIFO read/write latency from FIFO empty/full
- Set the debounce value to 25ms(default:0x0fffff) in host clock cycle units in the DEBNCE register.
- FIFO threshold value in bytes in the SDMMC_FIFOTH register.

2. Enumerated Card Stack

The card stack does the following:

- Enumerates all connected cards
- Sets the RCA for the connected cards
- Reads card-specific information
- Stores card-specific information locally

Enumeration depends on the operating mode of the SD/MMC card; the card type is first identified and the appropriate card enumeration routine is called.

1) Check if the card is connected.
2) Clear the card type register to set the card width as a single bit. For the given card number, clear the corresponding bits in the card_type register. Clear the register bit for a 1-bit, 4-bit bus width. For example, for card number=1, clear bit 0 and bit 16 of the card_type register.
3) Set clock frequency to FOD=400KHz, maximum – Program clock divider0 (bits 0-7 in the SDMMC_CLKDIV register) value to one-half of the cclk_in frequency divided by 400KHz. For example, if cclk_in is 20MHz, then the value is 20, 000/(2*400)=25.
4) Identify the card type; that is, SD, MMC, or SDIO.
   a. Send CMD5 first. If a response is received, then the card is SDIO
   b. If not, send CMD8 with the following Argument

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit[31:12] = 20'h0 //reserved bits
Bit[11:8] = 4'b0001 //VHS value
Bit[7:0] = 8'b10101010 //Preferred Check Pattern by SD2.0

c. If Response is received the card supports High Capacity SD2.0 then send ACMD41 with the following Argument
Bit[31] = 1'b0; //Reserved bits
Bit[30] = 1'b1; //High Capacity Status
Bit[29:24] = 6'h0; //Reserved bits
Bit[23:0] = Supported Voltage Range

d. If Response is received for ACMD41 then the card is SD. Otherwise the card is MMC.

e. If response is not received for initial CMD8 then card does not support High Capacity SD2.0, then issue CMD0 followed by ACMD41 with the following Argument
Bit[31] = 1'b0; //Reserved bits
Bit[30] = 1'b0; //High Capacity Status
Bit[29:24] = 6'h0; //Reserved bits
Bit[23:0] = Supported Voltage Range

5) Enumerate the card according to the card type.
6) Use a clock source with a frequency = Fod (that is, 400KHz) and use the following enumeration command sequence:

- SD card – Send CMD0, CMD8, ACMD41, CMD2, CMD3.
- MMC – Send CMD0, CMD1, CMD2, CMD3.

3. Power Control

You can implement power control using the following register, along with external circuitry:

- SDMMC_PWREN register – Control power to individual cards.

Programming the register depends on the implemented external circuitry. While turning on or off the power enable, you should confirm that power supply settings are correct. Power to all cards usually should be disabled while switching off the power.

4. Clock Programming

The Host Controller supports one clock source. The clock to an individual card can be enabled or disabled. Registers that support this are:

- SDMMC_CLKDIV – Programs individual clock source frequency. SDMMC_CLKDIV limited to 0 or 1 is recommended.
- SDMMC_CLKSRC – Assign clock source for each card.
- SDMMC_CLKENA – Enables or disables clock for individual card and enables low-power mode, which automatically stops the clock to a card when the card is idle for more than 8 clocks.

The Host Controller loads each of these registers only when the start_cmd bit and the update_clk_regs_only bit in the SDMMC_CMD register are set. When a command is successfully loaded, the Host Controller clears this bit, unless the Host Controller already has another command in the queue, at which point it gives an HLE(Hardware Locked Error). Software should look for the start_cmd and the update_clk_regs_only bits, and should also set the wait_prvdata_complete bit to ensure that clock parameters do not change during data transfer. Note that even though start_cmd is set for updating clock registers, the Host Controller does not raise a command_done signal upon command completion.
The following shows how to program these registers:

1) Confirm that no card is engaged in any transaction; if there is a transaction, wait until it finishes.
2) Stop all clocks by writing 0 to the SDMMC_CLKENA register. Set the start_cmd, update_clk_regs_only, and wait_prvdata_complete bits in the SDMMC_CMD register. Wait until start_cmd is cleared or an HLE is set; in case of an HLE, repeat the command.
3) Program the SDMMC_CLKDIV and SDMMC_CLKSRC registers, as required. Set the start_cmd, update_clk_regs_only, and wait_prvdata_complete bits in the SDMMC_CMD

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

register. Wait until start_cmd is cleared or an HLE is set; in case of an HLE, repeat the command.

4) Re-enable all clocks by programming the SDMMC_CLKENA register. Set the start_cmd, update_clk_regs_only, and wait_prvdata_complete bits in the SDMMC_CMD register. Wait until start_cmd is cleared or an HLE is set; in case of an HLE, repeat the command.

5. No-Data Command With or Without Response Sequence

To send any non-data command, the software needs to program the SDMMC_CMD register @0x2C and the SDMMC_CMDARG register @0x28 with appropriate parameters. Using these two registers, the Host Controller forms the command and sends it to the command bus. The Host Controller reflects the errors in the command response through the error bits of the SDMMC_RINTSTS register.

When a response is received – either erroneous or valid – the Host Controller sets the command_done bit in the SDMMC_RINTSTS register. A short response is copied in SDMMC_RESP0 register, while along response is copied to all four response registers @0x30, 0x34, 0x38, and 0x3C. The SDMMC_RESP3 register bit 31 represents the MSB, and the SDMMC_RESP0 register bit 0 represents the LSB of a long response.

For basic commands or non-data commands, follow these steps:

1) Program the SDMMC_CMDARG register @0x28 with the appropriate command argument parameter.
2) Program the SDMMC_CMD register @0x2C with the settings in following table.

Table 11-11 Command Settings for No-Data Command

Parameter	Value	Description
Default		
start_cmd	1	-
use_hold_reg	1	-
update_clk_regs_only	0	No clock parameters update command
data_expected	0	No data command
card number	0	Actual card number(one controller only connect one card, the num is No. 0)
cmd_index	command-index	-
send_initialization	0	Can be 1, but only for card reset commands, such as CMD0
stop_abort_cmd	0	Can be 1 for commands to stop data transfer, such as CMD12
response_length	0	Can be 1 for R2(long) response
response_expect	1	Can be 0 for commands with no response; for example, CMD0, CMD4, CMD15, and so on
User-selectable		
wait_prvdata_complete	1	Before sending command on command line, host should wait for completion of any data command in process, if any (recommended to always set this bit, unless the current command is to query status or stop data transfer when transfer is in progress)
check_response_crc	1	If host should crosscheck CRC of response received

1) Wait for command acceptance by host. The following happens when the command is loaded into the Host Controller:
   - Host Controller accepts the command for execution and clears the start_cmd bit in the SDMMC_CMD register, unless one command is in process, at which point the Host Controller can load and keep the second command in the buffer.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

- If the Host Controller is unable to load the command – that is, a command is already in progress, a second command is in the buffer, and a third command is attempted – then it generates an HLE (hardware-locked error).

2) Check if there is an HLE.
3) Wait for command execution to complete. After receiving either a response from a card or response timeout, the Host Controller sets the command_done bit in the SDMMC_RINTSTS register. Software can either poll for this bit or respond to a generated interrupt.
4) Check if response_timeout error, response_CRC error, or response error is set. This can be done either by responding to an interrupt raised by these errors or by polling bits 1, 6, and 8 from the SDMMC_RINTSTS register @0x44. If no response error is received, then the response is valid. If required, the software can copy the response from the response registers @0x30-0x3C.

Software should not modify clock parameters while a command is being executed.

6. Data Transfer Commands

Data transfer commands transfer data between the memory card and the Host Controller. To send a data command, the Host Controller needs a command argument, total data size, and block size. Software can receive or send data through the FIFO.

Before a data transfer command, software should confirm that the card is not busy and is in a transfer state, which can be done using the CMD13 and CMD7 commands, respectively. For the data transfer commands, it is important that the same bus width that is programmed in the card should be set in the SDMMC_ CTYPE register @0x18.

The Host Controller generates an interrupt for different conditions during data transfer, which are reflected in the SDMMC_RINTSTS register @0x44 as:

1) Data_Transfer_Over (bit 3) – When data transfer is over or terminated. If there is a response timeout error, then the Host Controller does not attempt any data transfer and the "Data Transfer Over" bit is never set.
2) Transmit_FIFO_Data_request (bit 4) – FIFO threshold for transmitting data was reached; software is expected to write data, if available, in FIFO.
3) Receive_FIFO_Data_request (bit 5) – FIFO threshold for receiving data was reached; software is expected to read data from FIFO.
4) Data starvation by Host timeout (bit 10) – FIFO is empty during transmission or is full during reception. Unless software writes data for empty condition or reads data for full condition, the Host Controller cannot continue with data transfer. The clock to the card has been stopped.
5) Data read timeout error (bit 9) – Card has not sent data within the timeout period.
6) Data CRC error (bit 7) – CRC error occurred during data reception.
7) Start bit error (bit 13) – Start bit was not received during data reception.
8) End bit error (bit 15) – End bit was not received during data reception or for a write operation; a CRC error is indicated by the card.

Conditions 6), 7), and 8) indicate that the received data may have errors. If there was a response timeout, then no data transfer occurred.

7. Single-Block or Multiple-Block Read

Steps involved in a single-block or multiple-block read are:

1) Write the data size in bytes in the SDMMC_BYTCNT register @0x20.
2) Write the block size in bytes in the SDMMC_BLKSIZ register @0x1C. The Host Controller expects data from the card in blocks of size SDMMC_BLKSIZ each.
3) Program the SDMMC_CMDARG register @0x28 with the data address of the beginning of a data read.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

4) Program the SDMMC_CMD register with the parameters listed in following table. For SD and MMC cards, use CMD17 for a single-block read and CMD18 for a multiple-block read. For SDIO cards, use CMD53 for both single-block and multiple-block transfers.

Table 11-12 Command Setting for Single or Multiple-Block Read

Parameter	Value	Description
Default		
start_cmd	1	-
use_hold_reg	1	-
update_clk_regs_only	0	No clock parameters update command
card number	0	Actual card number(one controller only connect one card, the num is No.0)
send_initialization	0	Can be 1, but only for card reset commands, such as CMD0
stop_abort_cmd	0	Can be 1 for commands to stop data transfer, such as CMD12
send_auto_stop	0/1	-
transfer_mode	0	Block transfer
read_write	0	Read from card
data_expected	1	Data command
response_length	0	Can be 1 for R2(long) response
response_expect	1	Can be 0 for commands with no response; for example, CMD0, CMD4, CMD15, and so on
User-selectable		
cmd_index	command-index	-
wait_prvdata_complete	1	0: Sends command immediately 1: Sends command after previous data transfer ends
check_response_crc	1	0: Host Controller should not check response CRC 1: Host Controller should check response CRC

After writing to the SDMMC_CMD register, the Host Controller starts executing the command; when the command is sent to the bus, the command_done interrupt is generated.

- Software should look for data error interrupts; that is, bits 7, 9, 13, and 15 of the SDMMC_RINTSTS register. If required, software can terminate the data transfer by sending a STOP command.
- Software should look for Receive_FIFO_Data_request and/or data starvation by host timeout conditions. In both cases, the software should read data from the FIFO and make space in the FIFO for receiving more data.
- When a Data_Transfer_Over interrupt is received, the software should read the remaining data from the FIFO.

8. Single-Block or Multiple-Block Write

Steps involved in a single-block or multiple-block write are:

1) Write the data size in bytes in the SDMMC_BYTCNT register @0x20.
2) Write the block size in bytes in the SDMMC_BLKSIZ register @0x1C; the Host Controller sends data in blocks of size SDMMC_BLKSIZ each.
3) Program SDMMC_CMDARG register @0x28 with the data address to which data should be written.
4) Write data in the FIFO; it is usually best to start filling data the full depth of the FIFO.
5) Program the SDMMC_CMD register with the parameters listed in following table.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Table 11-13 Command Settings for Single or Multiple-Block Write

Parameter	Value	Description
Default		
start_cmd	1	-
use_hold_reg	1	-
update_clk_regs_only	0	No clock parameters update command
card number	0	Actual card number(one controller only connect one card, the num is No. 0)
send_initialization	0	Can be 1, but only for card reset commands, such as CMD0
stop_abort_cmd	0	Can be 1 for commands to stop data transfer, such as CMD12
send_auto_stop	0/1	-
transfer_mode	0	Block transfer
read_write	1	Write to card
data_expected	1	Data command
response_length	0	Can be 1 for R2(long) response
response_expect	1	Can be 0 for commands with no response; for example, CMD0, CMD4, CMD15, and so on
User-selectable		
cmd_index	command-index	-
wait_prvdata_complete	1	0: Sends command immediately 1: Sends command after previous data transfer ends
check_response_crc	1	0: Host Controller should not check response CRC 1: Host Controller should check response CRC

After writing to the SDMMC_CMD register, Host Controller starts executing a command; when the command is sent to the bus, a command_done interrupt is generated.

- Software should look for data error interrupts; that is, for bits 7, 9, and 15 of the SDMMC_RINTSTS register. If required, software can terminate the data transfer by sending the STOP command.
- Software should look for Transmit_FIFO_Data_Request and/or timeout conditions from data starvation by the host. In both cases, the software should write data into the FIFO.
- When a Data_Transfer_Over interrupt is received, the data command is over. For an open-ended block transfer, if the byte count is 0, the software must send the STOP command. If the byte count is not 0, then upon completion of a transfer of a given number of bytes, the Host Controller should send the STOP command, if necessary. Completion of the AUTO-STOP command is reflected by the Auto_command_done interrupt – bit 14 of the SDMMC_RINTSTS register. A response to AUTO_STOP is stored in SDMMC_RESP1 @0x34.

9. Stream Read

A stream read is like the block read mentioned in "Single-Block or Multiple-Block Read", except for the following bits in the SDMMC_CMD register:

transfer_mode = 1; //Stream transfer
cmd_index = CMD20;

A stream transfer is allowed for only a single-bit bus width.

10. Stream Write

A stream write is exactly like the block write mentioned in "Single-Block or Multiple-Block Write", except for the following bits in the SDMMC_CMD register:

transfer_mode = 1;//Stream transfer

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

cmd_index = CMD11;

In a stream transfer, if the byte count is 0, then the software must send the STOP command. If the byte count is not 0, then when a given number of bytes completes a transfer, the Host Controller sends the STOP command. Completion of this AUTO_STOP command is reflected by the Auto_command_done interrupt. A response to an AUTO_STOP is stored in the SDMMC_RESP1 register@0x34.

A stream transfer is allowed for only a single-bit bus width.

11. Packed Commands

In order to reduce overhead, read and write commands can be packed in groups of commands—either all read or all write—that transfer the data for all commands in the group in one transfer on the bus.

Packed commands can be of two types:

- Packed Write: CMD23 →CMD25
- Packed Read: CMD23 → CMD25 → CMD23 → CMD18

Packed commands are put in packets by the application software and are transparent to the core.

12. Sending Stop or Abort in Middle of Transfer

The STOP command can terminate a data transfer between a memory card and the Host Controller, while the ABORT command can terminate an I/O data transfer for only the SDIO_IOONLY and SDIO_COMBO cards.

- Send STOP command – Can be sent on the command line while a data transfer is in progress; this command can be sent at any time during a data transfer.

You can also use an additional setting for this command in order to set the SDMMC_CMD register bits (5-0) to CMD12 and set bit 14 (stop_abort_cmd) to 1. If stop_abort_cmd is not set to 1, the Controller does not know that the user stopped a data transfer. Reset bit 13 of the SDMMC_CMD register (wait_prvdata_complete) to 0 in order to make the Controller send the command at once, even though there is a data transfer in progress.

- Send ABORT command – Can be used with only an SDIO_IOONLY or SDIO_COMBO card. To abort the function that is transferring data, program the function number in ASx bits (CCCR register of card, address 0x06, bits (0-2) using CMD52.

13. Read_Wait Sequence

Read_wait is used with only the SDIO card and can temporarily stall the data transfer—either from function or memory—and allow the host to send commands to any function within the SDIO device. The host can stall this transfer for as long as required. The Host Controller provides the facility to signal this stall transfer to the card. The steps for doing this are:

1) Check if the card supports the read_wait facility; read SRW (bit 2) of the CCCR register @0x08. If this bit is 1, then all functions in the card support the read_wait facility. Use CMD52 to read this bit.
2) If the card supports the read_wait signal, then assert it by setting the read_wait (bit 6) in the SDMMC_CTRL register @0x00.
3) Clear the read_wait bit in the SDMMC_CTRL register.

14. Controller/DMA/FIFO Reset Usage

- Controller reset – Resets the controller by setting the controller_reset bit (bit 0) in the SDMMC_CTRL register; this resets the CIU and state machines, and also resets the BIU-to-CIU interface. Since this reset bit is self-clearing, after issuing the reset, wait until this bit is cleared.
- FIFO reset - Resets the FIFO by setting the fifo_reset bit (bit 1) in the SDMMC_CTRL register; this resets the FIFO pointers and counters of the FIFO. Since this reset bit is self-clearing, after issuing the reset, wait until this bit is cleared.

In external DMA transfer mode, even when the FIFO pointers are reset, if there is a DMA transfer in progress, it could push or pop data to or from the FIFO; the DMA itself completes correctly. In order to clear the FIFO, the software should issue an additional FIFO

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

reset and clear any FIFO underrun or overrun errors in the SDMMC_RINTSTS register caused by the DMA transfers after the FIFO was reset.

15. Card Read Threshold

When an application needs to perform a Single or Multiple Block Read command, the application must program the SDMMC_CARDTHRCTL register with the appropriate card read threshold size (card_rd_thres) and set the card read threshold enable (card_rd_thres_en) bit to 1'b1. This additional programming ensures that the Host Controller sends a Read Command only if there is space equal to the card_rd_thres available in the RX FIFO. This in turn ensures that the card clock is not stopped in the middle a block of data being transmitted from the card. The card read threshold can be set to the block size of the transfer, which guarantees that there is a minimum of one block size of space in the RX FIFO before the controller enables the card clock. The card read threshold is required when the Round Trip Delay (Delay_R = Delay_O + tODLY + Delay_I) is greater than 0.5cclk_in period.

16. Error Handling

The Host Controller implements error checking; errors are reflected in the SDMMC_RINTSTS register@0x44 and can be communicated to the software through an interrupt, or the software can poll for these bits. Upon power-on, interrupts are disabled (int_enable in the SDMMC_CTRL register is 0), and all the interrupts are masked (bits 0-31 of the SDMMC_INTMASK register; default is 0).

Error handling:

- Response and data timeout errors – For response timeout, software can retry the command. For data timeout, the Host Controller has not received the data start bit – either for the first block or the intermediate block – within the timeout period, so software can either retry the whole data transfer again or retry from a specified block onwards. By reading the contents of the SDMMC_TCBCNT later, the software can decide how many bytes remain to be copied.
- Response errors – Set when an error is received during response reception. In this case, the response that copied in the response registers is invalid. Software can retry the command.
- Data errors – Set when error in data reception are observed; for example, data CRC, start bit not found, end bit not found, and so on. These errors could be set for any block-first block, intermediate block, or last block. On receipt of an error, the software can issue a STOP or ABORT command and retry the command for either whole data or partial data.
- Hardware locked error – Set when the Host Controller cannot load a command issued by software. When software sets the start_cmd bit in the SDMMC_CMD register, the Host Controller tries to load the command. If the command buffer is already filled with a command, this error is raised. The software then has to reload the command.
- FIFO underrun/overrun error – If the FIFO is full and software tries to write data in the FIFO, then an overrun error is set. Conversely, if the FIFO is empty and the software tries to read data from the FIFO, an underrun error is set. Before reading or writing data in the FIFO, the software should read the fifo_empty or fifo_full bits in the SDMMC_STATUS register.
- Data starvation by host timeout – Raised when the Host Controller is waiting for software intervention to transfer the data to or from the FIFO, but the software does not transfer within the stipulated timeout period. Under this condition and when a read transfer is in process, the software should read data from the FIFO and create space for further data reception. When a transmit operation is in process, the software should fill data in the FIFO in order to start transferring data to the card.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

- CRC Error on Command – If a CRC error is detected for a command, the CE-ATA device does not send a response, and a response timeout is expected from the Host Controller. The ATA layer is notified that an MMC transport layer error occurred.

Notes: During a multiple-block data transfer, if a negative CRC status is received from the device, the data path signals a data CRC error to the BIU by setting the data CRC error bit in the SDMMC_RINTSTS register. It then continues further data transmission until all the bytes are transmitted.

11.6.5 Voltage Switching

The Host Controller supports SD 3.0 Ultra High Speed (UHS-1) and is capable of voltage switching in SD-mode, which can be applied to SD High-Capacity (SDHC) and SD Extended Capacity (SDXC) cards. UHS-1 supports only 4-bit mode.
However, whether the IO voltage of 1.8v supported or not is depended on the SoC design.
SD 3.0 UHS-1 supports the following transfer speed modes for UHS-50 and/or UHS-104 cards:

- DS – default-speed up to 25MHz, 3.3V signaling
- HS – high-speed up to 50MHz, 3.3V signaling
- SDR12 – SDR up to SDR 25MHz, 1.8V signaling
- SDR25 – SDR up to 50MHz, 1.8V signaling
- SDR50 – SDR up to 100MHz, 1.8V signaling
- DDR50 – DDR up to 50MHz, 1.8V signaling

Voltage selection can be done in only SD mode. The first CMD0 selects the bus mode-either SD mode or SPI mode. The card must be in SD mode in order for 1.8V signaling mode to apply, during which time the card cannot be switched to SPI mode or 3.3V signaling without a power cycle.
If the System BIOS in an embedded system already knows that it is connected to an SD 3.0 card, then the driver programs the Controller to initiate ACMD41. The software knows from the response of ACMD41 whether or not the card supports voltage switching to 1.8V.

- If bit 32 of ACMD41 response is 1'b1: card supports voltage switching and next command-CMD11-invokes voltage switching sequence. After CMD11 is started, the software must program the IO voltage selection register based on the soc architecture.
- If bit 32 of ACMD41 response is 1'b0: card does not support voltage switching and CMD11 should not be started.

If the card and Host Controller accept voltage switching, then they support UHS-1 modes of data transfer. After the voltage switch to 1.8V, SDR12 is the default speed.
Since the UHS-1 can be used in only 4-bit mode, the software must start ACMD6 and change the card data width to 4-bit mode; ACMD6 is driven in any of the UHS-1 speeds. If the host wants to select the DDR mode of data transfer, then the software must program the SDMMC_DDR_REG register in the CSR space.
To choose from any of the SDR or DDR modes, appropriate values should be programmed in the SDMMC_CLKDIV register.

1. Voltage Switch Operation

The Voltage Switch operation must be performed in SD mode only.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

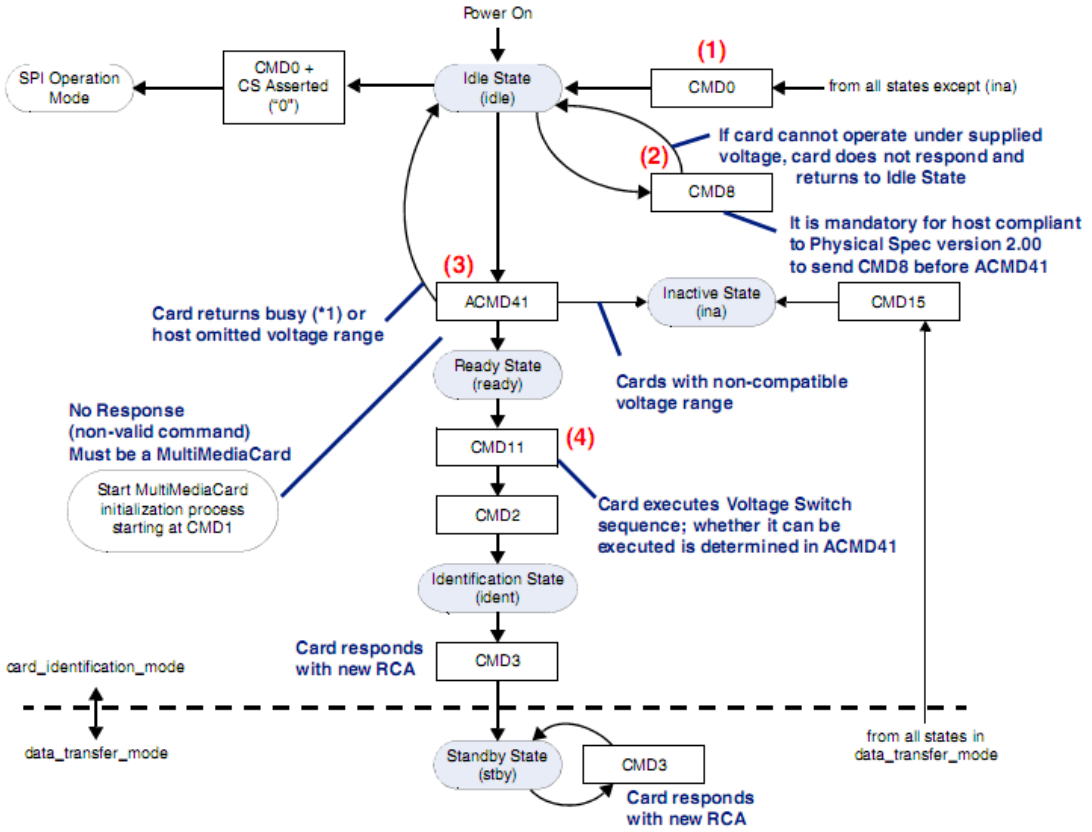


Fig. 11-12 Voltage Switching Command Flow Diagram

The following outlines the steps for the voltage switch programming sequence

1) Software Driver starts CMD0, which selects the bus mode as SD.
2) After the bus is in SD card mode, CMD8 is started in order to verify if the card is compatible with the SD Memory Card Specification, SD 2.0. CMD8 determines if the card is capable of working within the host supply voltage specified in the VHS (19:16) field of the CMD; the card supports the current host voltage if a response to CMD8 is received.
3) ACMD 41 is started. The response to this command informs the software if the card supports voltage switching; bits 38, 36, and 32 are checked by the card argument of ACMD41; refer to following figure.

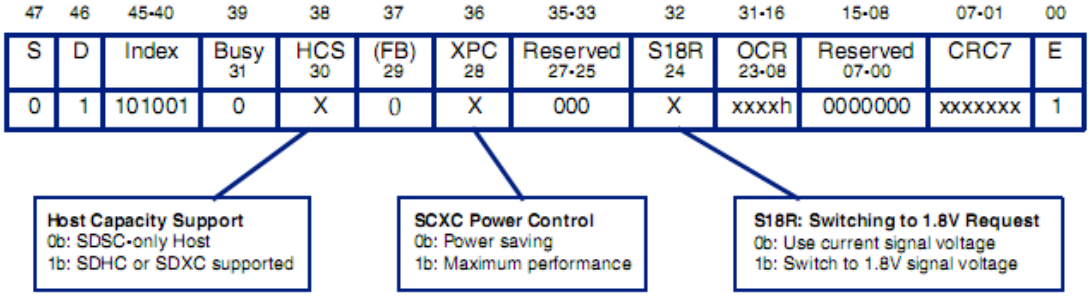


Fig. 11-13 ACMD41 Argument

- Bit 30 informs the card if host supports SDHC/SDXC or not; this bit should be set to 1'b1.
- Bit 28 can be either 1 or 0.
- Bit 24 should be set to 1'b1, indicating that the host is capable of voltage switching; refer to following figure.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

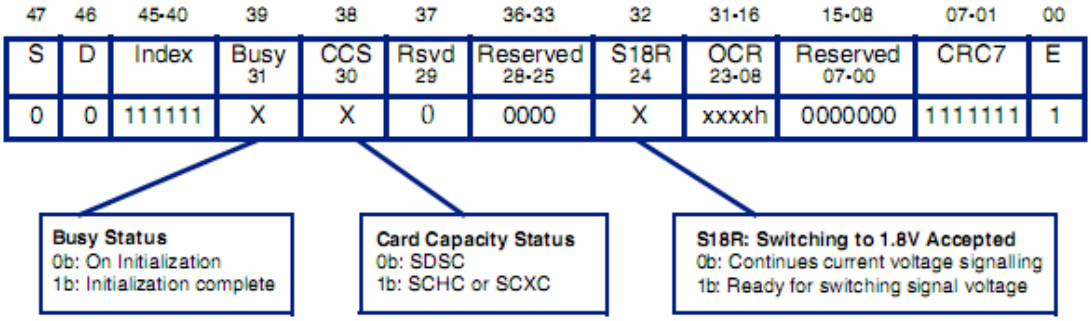


Fig. 11-14 ACMD41 Response(R3)

- Bit 30 – If set to 1'b1, card supports SDHC/SDXC; if set to 1'b0, card supports only SDSC
- Bit 24 – If set to 1'b1, card supports voltage switching and is ready for the switch
- Bit 31 – If set to 1'b1, initialization is over; if set to 1'b0, means initialization in process

4) If the card supports voltage switching, then the software must perform the steps discussed for either the "Voltage Switch Normal Scenario" or the "Voltage Switch Error Scenario".

2. Voltage Switch Normal Scenario

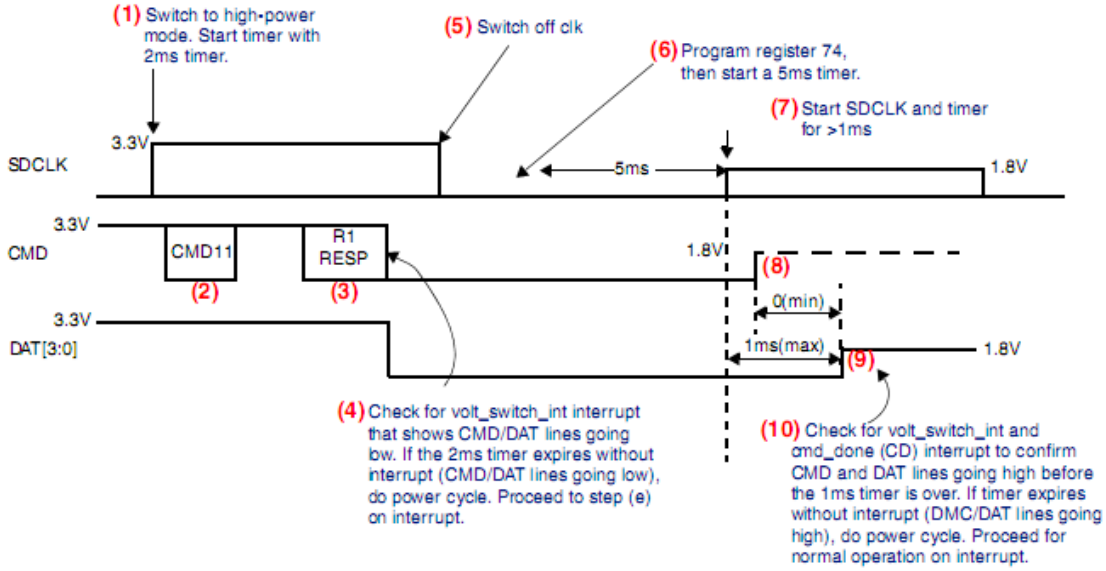


Fig. 11-15 Voltage Switch Normal Scenario

1) The host programs SDMMC_CLKENA register—with zero (0) for the corresponding card, which makes the Host Controller move to high-power mode. The application should start a timer with a recommended value of 2ms; this value of 2 ms is determined as below:Total clk required for CMD11 = 48 clks
   Total clk required for RESP R1 = 48 clks
   Maximum clk delay between MCD11 end to start of RESP1 = 60 clks
   Total = 48+48 + 60 = 160
   Minimum frequency during enumeration is 100 KHz; that is, 10us
   Total time = 160 * 10us = 1600us = 1. 6ms ~ 2ms
2) The host issues CMD11 to start the voltage switch sequence. Set bit 28 to 1'b1 in CMD when setting CMD11; for more information on setting bits, refer to "Boot Operation".
3) The card returns R1 response; the Host Controller does not generate cmd_done interrupt on receiving R1 response.
4) The card drives CMD and DAT [3:0] to low immediately after the response. The Host Controller generates interrupt (VOLT_SWITCH_INT) once the CMD or DAT [3:0] line

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

goes low. The application should wait for this interrupt. If the 2ms timer expires without an interrupt (CMD/DAT lines going low), do a power cycle.

Note: Before doing a power cycle, switch off the card clock by programming SDMMC_CLKENA register

Proceed to step (5) on getting an interrupt (VOLT_SWITCH_INT).

Note: This interrupt must be cleared once this interrupt is received. Additionally, this interrupt should not be masked during the voltage switch sequence.

If the timer expires without interrupt (CMD/DAT lines going low), perform a power cycle. Proceed to step (5) on interrupt.

1) Program the SDMMC_CLKENA register, with 0 for the corresponding card; the host stops supplying SDCLK.
2) Program Voltage register to the required values for the corresponding card. The application should start a timer > 5ms.
3) After the 5ms timer expires, the host voltage regulator is stable. Program SDMMC_CLKENA, cclk_enable register, with 1 for the corresponding card; the host starts providing SDCLK at 1.8V; this can be at zero time after Voltage register has been programmed. When the SDMMC_CLKENA register is programmed, the application should start another timer > 1ms.
4) By detecting SDCLK, the card drives CMD to high at 1.8V for at least one clock and then stops driving (tri-state); CMD is triggered by the rising edge of SDCLK (SDR timing).
5) If switching to 1.8V signaling is completed successfully, the card drives DAT [3:0] to high at 1.8V for at least one clock and then stops driving (tri-state); DAT [3:0] is triggered by the rising edge of SDCLK (SDR timing). DAT[3:0] must be high within 1ms from the start of SDCLK.
6) The Host Controller generates a voltage switch interrupt (VOLT_SWITCH_INT) and a command done (CD) interrupt once the CMD and DAT[3:0] lines go high. The application should wait for this interrupt to confirm CMD and DAT lines going high before the 1ms timer is done.

If the timer expires without the voltage switch interrupt (VOLT_SWITCH_INT), a power cycle should be performed. Program the SDMMC_CLKENA register to stop the clock for the corresponding card number. Wait for the cmd_done (CD) interrupt. Proceed for normal operation on interrupt. After the sequence is completed, the host and the card start communication in SDR12 timing.

3. Voltage Switch Error Scenario

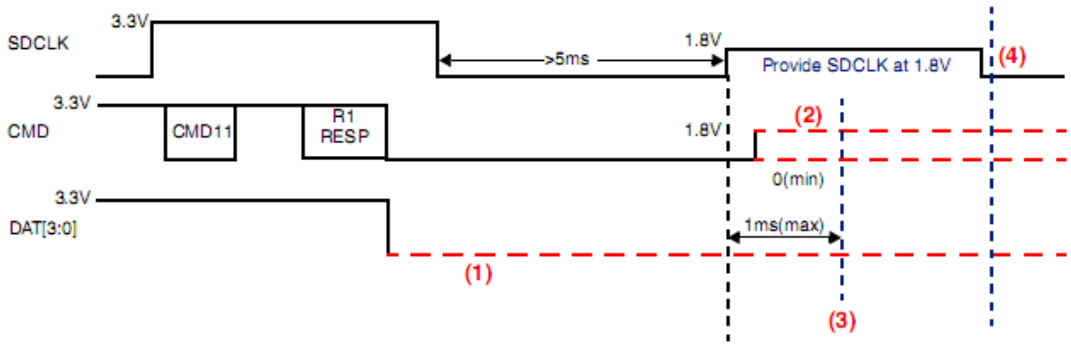


Fig. 11-16 Voltage Switch Error Scenario

1) If the interrupt (VOLT_SWITCH_INT) does not come, then the 2 ms timer should time out and a power cycle should be initiated.

Note: Before performing a power cycle, switch off the card clock by programming SDMMC_CLKENA register; no cmd_done (CD) interrupt is generated.

Additionally, if the card detects a voltage error at any point in between steps (5) and (7) in the card keeps driving DAT[3:0] to low until card power off.

2) CMD can be low or tri-state.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

3) The Host Controller generates a voltage switch interrupt once the CMD and DAT[3:0] lines go high. The application should check for an interrupt to confirm CMD and DAT lines going high before the 1 ms timer is done.

If the 1 ms timer expires without interrupt (VOLT_SWITCH_INT) and cmd_done (CD), a power cycle should be performed. Program the SDMMC_CLKENA register to stop SDCLK of the corresponding card. Wait for the cmd_done interrupt. Proceed for normal operation on interrupt.

4) If DAT[3:0] is low, the host drives SDCLK to low and then stops supplying the card power.

Note: The card checks voltages of its own regulator output and host signals to ensure they are less than 2.5V. Errors are indicated by (1) and (2).

- If voltage switching is accepted by the card, the default speed is SDR12.
- Command Done is given:
  - If voltage switching is properly done, CMD and DAT line goes high.
  - If switching is not complete, the 1ms timer expires, and the card clk is switched off.

Note: No other CMD should be driven before the voltage switching operation is completed and Command Done is received.

- The application should use CMD6 to check and select the particular function; the function appropriate-speed should be selected.

After the function switches, the application should program the correct value in the CLKDIV register, depending on the function chosen. Additionally, if Function 0x4 of the Access mode is chosen—that is, DDR50, then the application should also program 1'b1 in DDR_REG for the card number that has been selected for DDR50 mode.

11.6.6 DDR Operation

1. 4-bit DDR Programming Sequence

DDR programming should be done only after the voltage switch operation has completed. The following outlines the steps for the DDR programming sequence:

1) Once the voltage switch operation is complete, the user must program voltage selection register to the required values for the corresponding card.

- To start a card to work in DDR mode, the application must program a bit of the newly defined SDMMC_UHSREG[16] register with a value of 1'b1.
- The bit that the user programs depends on which card is to be accessed in DDR mode.

2) To move back to SDR mode, a power cycle should be run on the card—putting the card in SDR12 mode—and only then should SDMMC_UHSREG[16] be set back to 1'b0 for the appropriate card.

2. 8-bit DDR Programming Sequence

The following outlines the steps for the 8-bit DDR programming sequence:

1) The cclk_in signal should be twice the speed of the required cclk_out. Thus, if the cclk_out signal is required to be 50 MHz, the cclk_in signal should be 100 MHz.
2) The SDMMC_CLKDIV register should always be programmed with a value higher than zero; that is, a clock divider should always be used for 8-bit DDR mode.
3) The application must program the SDMMC_UHSREG[16] register (ddr_reg bit) by assigning it with a value of 1 for the bit corresponding to the card number; this causes the selected card to start working in DDR mode.
4) Depending on the card number, the SDMMC_CTYPE [16] bits should be set in order to make the host work in the 8-bit mode.

3. eMMC4.5 DDR START Bit

The eMMC4.5 changes the START bit definition in the following manner:

1) Receiver samples the START bit on the rising edge.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

2) On the next rising edge after sampling the START bit, the receiver must sample the data.

3) Removes requirement of the START bit and END bit to be high for one full cycle.

Notes: The Host Controller does not support a START bit duration higher than one clock cycle. START bit durations of one or less than one clock cycle are supported and can be defined at the time of startup by programming the SDMMC_EMMCDDR_REG register.

Following figure illustrates cases for the definition change of the START bit with eMMC4.5; it also illustrates how some of these cases can fail in sampling when higher-value delays are considered for I/O PADs.

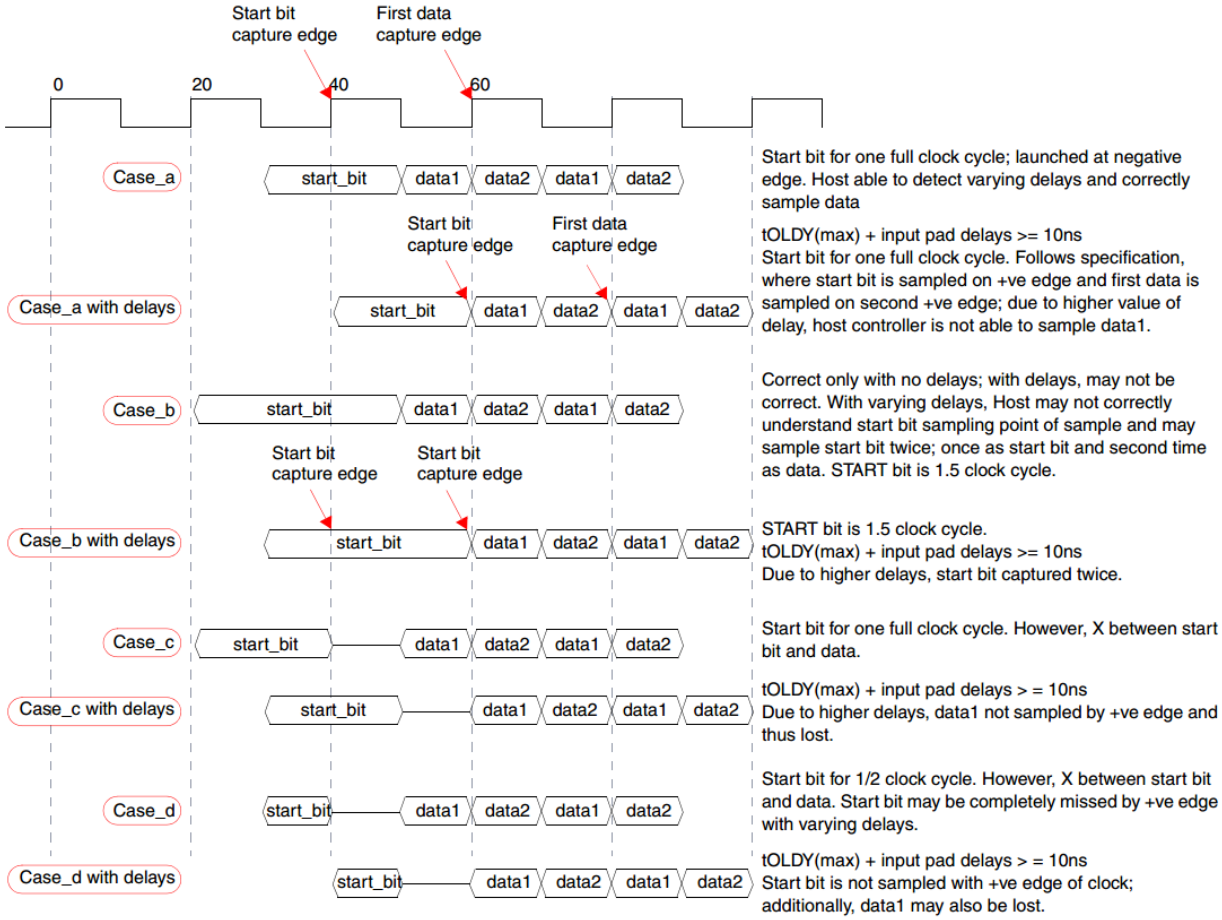


Fig. 11-17 CASES for eMMC 4.5 START bit

4. Reset Command/Moving from DDR50 to SDR12

To reset the mode of operation from DDR50 to SDR12, the following sequence of operations has to be done by the application:

1) Issue CMD0.

When CMD0 is received, the card changes from DDR50 to SDR12.

2) Program the SDMMC_CLKDIV register with an appropriate value.

3) Set ddr_reg to 0.

Note: The voltage register should not be programmed to 0 while switching from DDR50 to SDR12, since the card is still operating in 1.8V mode after receiving CMD0.

11.6.7 H/W Reset Operation

When the RST_N signal goes low, the card enters a pre-idle state from any state other than the inactive state.

H/W Reset Programming Sequence

The following outlines the steps for the H/W reset programming sequence:

11) Program CMD12 to end any transfer in process.

12) Wait for DTO, even if no response is sent back by the card.

13) Set the following resets:

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

- DMA reset–SDMMC_CTRL[2]
- FIFO reset –SDMMC_CTRL[1]

Note: The above steps are required only if a transfer is in process.

14) Program the SDMMC_RSTN register with a value of 0; this can be done at any time when the card is connected to the controller. This programming asserts the RST_N signal and resets the card.
15) Wait for minimum of 1 µs or cclk_in period, whichever is greater
16) After a minimum of 1 µs, the application should program a value of 0 into the SDMMC_RSTN register. This de-asserts the RST_N signal and takes the card out of reset.
17) The application can program a new CMD only after a minimum of 200 µs after the de-assertion of the RST_N signal, as per the MMC 4.41 standard.

Note: For backward compatibility, the RST_N signal is temporarily disabled in the card by default. The host may need to set the signal as either permanently enabled or permanently disabled before it uses the card.

11.6.8 FBE Scenarios

An FBE occurs due to an AHB error response on the AHB bus. This is a system error, so the software driver should not perform any further programming to the host. The only recovery mechanism from such scenarios is to do one of the following:

- Issue a hard reset by asserting the reset_n signal
- Do a program controller reset by writing to the SDMMC_CTRL[0] register

1. FIFO Overflow and Underflow

During normal data transfer conditions, FIFO overflow and underflow will not occur. However if there is a programming error, then FIFO overflow/underflow can result. For example, consider the following scenarios.

- For transmit: PBL=4, tx wmark = 1. For the above programming values, if the FIFO has only one location empty, it issues a dma_req to IDMAC FSM. Due to PBL value=4, the IDMAC FSM performs 4 pushes into the FIFO. This will result in a FIFO overflow interrupt.
- For receive: PBL=4, rx wmark = 1. For the above programming values, if the FIFO has only one location filled, it issues a dma_req to IDMAC FSM. Due to PBL value=4, the IDMAC FSM performs 4 pops to the FIFO. This will result in a FIFO underflow interrupt.

The driver should ensure that the number of bytes to be transferred as indicated in the descriptor should be a multiple of 4bytes with respect to H_DATA_WIDTH=32. For example, if the BYTCNT = 13, the number of bytes indicated in the descriptor should be 16 for H_DATA_WIDTH=32.

2. Programming of PBL and Watermark Levels

The DMAC performs data transfers depending on the programmed PBL and threshold values.

Table 11-14 PBL and Watermark Levels

PBL (Number of transfers)	Tx/Rx Watermark Value
1	Greater than or equal to 1
4	Greater than or equal to 4
8	Greater than or equal to 8
16	Greater than or equal to 16
32	Greater than or equal to 32
64	Greater than or equal to 64
128	Greater than or equal to 128
256	Greater than or equal to 256

11.6.9 Variable Delay Tuning

Tuning is defined by SD and MMC cards to determine the correct sampling point required

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

for the host, especially for the speed modes SDR104 and HS200 where the output delays from the cards can be up to 2 UI. Tuning is required for other speed modes-such as DDR50-even though the output delay from the card is less than one cycle.
Command for tuning is different for different cards.

- SD Memory Card:
  - CMD19 – SD card for SDR50 and SDR104 speed modes. Tuning data is defined by card specifications.
  - CMD6 – SD card for speed modes not supporting CMD19. Tuning data is the 64byte SD status.
- Multimedia Card:
  - CMD21 – MMC card for HS200 speed mode. Tuning data is defined by card specifications.
  - CMD8 – MMC card for speed modes not supporting CMD21. Tuning data is 512 byte ExtCSD data.

The following is the procedure for variable delay tuning:

1) Set a phase shift of 0-degree on cclk_in_sample.
2) Send the tuning command to the card; the card in turn sends an R1 response on the CMD line and tuning data on the DAT line.
3) If the host sees any of the errors—start bit error, data crc error, end bit error, data read time-out, response crc error, response error—then the sampling point is incorrect.
4) Send CMD12 to bring the Host Controller state machines to idle.

- The card may treat CMD12 as an invalid command because the card has successfully sent the tuning data, and it cannot send a response.
- The Host Controller may generate a response time-out interrupt that must be cleared by software.

5) Repeat steps 2) to 4) by increasing the phase shift value or delay element number on cclk_in_sample until the correct sampling point is received such that the host does not see any of the errors.
6) Mark this phase shift value as the starting point of the sampling window.
7) Repeat steps 2) to 4) by increasing the phase shift value or delay element number on cclk_in_sample until the host sees the errors starting to come again or the phase shift value reaches max degree.
8) Mark the last successful phase shift value as the ending point of the sampling window.

A window is established where the tuning block is matched. For example, for a scenario where the tuning block is received correctly for a phase shift window of 90-degree and 180-degree, then an appropriate sampling point is established as 135-degree. Once a sampling point is established, no errors should be visible in the tuning block.

Variable Delay/Clock Generation

Variable delay mechanism for the cclk_in_drv is useful in order to meet a range of hold-time requirements across modes or tuning. Variable delay mechanism for the cclk_in_sample is mandatory and is required to achieve the correct sampling point for data. cclk_in/cclk_in_sample/ cclk_in_drv is generated by Clock Generation Unit (CLKGEN) with variable delay mechanism, which includes Phase Shift Unit and Delay Line Unit selectable. The Phase Shift Unit can shift cclk_in_sample by 0/90/180/270-degree relative to cclk_in, controlled by sample_degree. The Phase Shift Unit can shift cclk_in_drv by 0/90/180/270-degree relative to cclk_in, controlled by drv_degree.
The Delay Line Unit can shift cclk_in_sample in the unit of 40ps~80ps for every delay element. The delay unit number is determined by sample_delaynum, and enabled by sample_sel.
The Delay Line Unit can shift cclk_in_drv in the unit of 40ps~80ps for every delay element. The delay unit number is determined by drv_delaynum, and enabled by drv_sel.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

cclk_in is generated by cclkin divided by 2. cclk_in_drv and cclk_in_sample clocks are phase-shifted with delay of cclk_in. All clocks are recommended to have a 50% duty cycle; DDR modes must have 50% duty cycles.
The architecture is as follows.

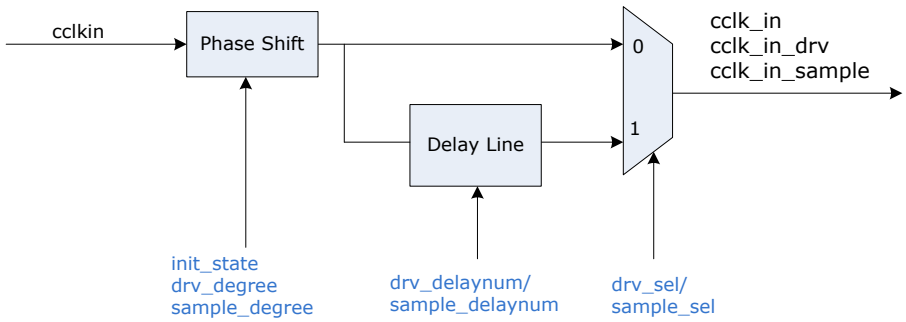


Fig. 11-18 Clock Generation Unit

The control signals for different Host Controller instance are shown as follows:

Table 11-15 Configuration for SDMMC Clock Generation

Signal Name	Source	Default	Description
init_state	CRU_SDMMC_CON0[0]	0	Soft initial state for phase shift.
drv_degree[1:0]	CRU_SDMMC_CON0[2:1]	2	Phase shift for cclk_in_drv. 2'b00: 0-degree 2'b01: 90-degree 2'b10: 180-degree 2'b11: 270-degree
drv_delaynum[7:0]	CRU_SDMMC_CON0[10:3]	0	Element number in delay line for cclk_in_drv
drv_sel	CRU_SDMMC_CON0[11]	0	cclk_in_drv source selection: 1'b0: Use clock after phase_shift 1'b1: Use clock after phase_shift and delay line
sample_degree[1:0]	CRU_SDMMC_CON1[2:1]	0	Phase shift for cclk_in_sample. 2'b00: 0-degree 2'b01: 90-degree 2'b10: 180-degree 2'b11: 270-degree
sample_delaynum[7:0]	CRU_SDMMC_CON1[10:3]	0	Element number in delay line for cclk_in_sample
sample_sel	CRU_SDMMC_CON1[11]	0	cclk_in_sample source selection: 1'b0: Use clock after phase_shift 1'b1: Use clock after phase_shift and delay line

Table 11-16 Configuration for SDIO Clock Generation

Signal Name	Source	Default	Description
init_state	CRU_SDIO_CON0[0]	0	Soft initial state for phase shift.
drv_degree[1:0]	CRU_SDIO_CON0[2:1]	2	Phase shift for cclk_in_drv. 2'b00: 0-degree 2'b01: 90-degree 2'b10: 180-degree 2'b11: 270-degree
drv_delaynum[7:0]	CRU_SDIO_CON0[10:3]	0	Element number in delay line for cclk_in_drv
drv_sel	CRU_SDIO_CON0[11]	0	cclk_in_drv source selection: 1'b0: Use clock after phase_shift 1'b1: Use clock after phase_shift and delay line

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Signal Name	Source	Default	Description
sample_degree [1:0]	CRU_SDIO_CON1[2:1]	0	Phase shift for cclk_in_sample. 2'b00: 0-degree 2'b01: 90-degree 2'b10: 180-degree 2'b11: 270-degree
sample_delaynum[7:0]	CRU_SDIO_CON1[10:3]	0	Element number in delay line for cclk_in_sample
sample_sel	CRU_SDIO_CON1[11]	0	cclk_in_sample source selection: 1'b0: Use clock after phase_shift 1'b1: Use clock after phase_shift and delay line

Table 11-17 Configuration for EMMC Clock Generation

Signal Name	Source	Default	Description
init_state	CRU_EMMC_CON0[0]	0	Soft initial state for phase shift.
drv_degree[1:0]	CRU_EMMC_CON0[2:1]	2	Phase shift for cclk_in_drv. 2'b00: 0-degree 2'b01: 90-degree 2'b10: 180-degree 2'b11: 270-degree
drv_delaynum [7:0]	CRU_EMMC_CON0[10:3]	0	Element number in delay line for cclk_in_drv
drv_sel	CRU_EMMC_CON0[11]	0	cclk_in_drv source selection: 1'b0: Use clock after phase_shift 1'b1: Use clock after phase_shift and delay line
sample_degree[1:0]	CRU_EMMC_CON1[2:1]	0	Phase shift for cclk_in_sample. 2'b00: 0-degree 2'b01: 90-degree 2'b10: 180-degree 2'b11: 270-degree
sample_delaynum[7:0]	CRU_EMMC_CON1[10:3]	0	Element number in delay line for cclk_in_sample
sample_sel	CRU_EMMC_CON1[11]	0	cclk_in_sample source selection: 1'b0: Use clock after phase_shift 1'b1: Use clock after phase_shift and delay line

The following outlines the steps for clock generation sequence:

1) Assert init_state to soft reset the CLKGEN.
2) Configure drv_degree/sample_degree.
3) If fine adjustment required, delay line can be used by configuring drv_delaynum/sample_delaynum and drv_sel/sample_sel.
4) Dis-assert init_state to start CLKGEN.

11.6.10 Card Detection Method

There are many methods for SDMMC/SDIO device detection.

- Method1: Using SDMMC_CDETECT register, which is value on card_detect_n input port. 0 represents presence of card.
- Method2: Using card detection unit in Host Controller, outputting host interrupt. The card detection unit looks for any changes in the card-detect signals for card insertion or card removal. It filters out the debounces associated with mechanical insertion or removal, and generates one interrupt to the host. You can program the debounce filter

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

value in SDMMC_DEBNCE [23:0]. Following figure illustrates the timing for card-detect signals.

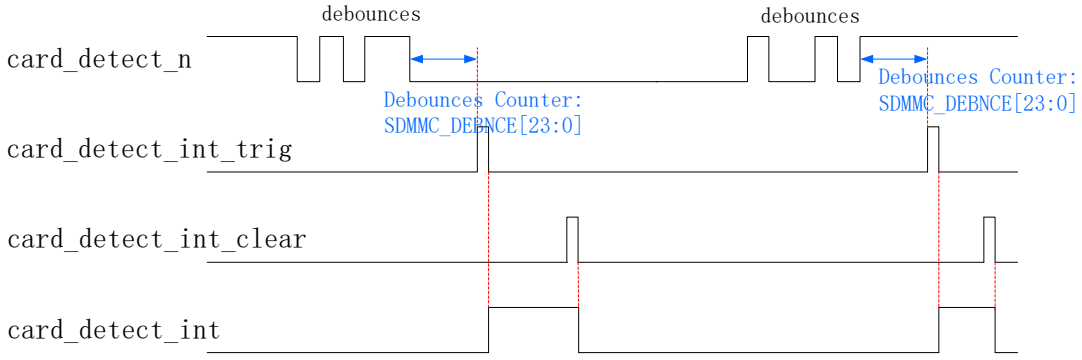


Fig. 11-19 Card Detection Method 2

- Method3: Using card detection unit in SoC, outputting sdmmc_detect_dual_edge_int connecting to IRQ[97]. Similar to Method2, except that the debounce time is configurable by PMUGRF_SDMMC_DET; and the insertion/removal detection interrupt can be enabled or cleared respectively. The detailed register information is:

Table 11-18 Register for SDMMC Card Detection Method 3

Signal Name	Source	Default	Description
sd_detectn_rise_edge_irq_en	PMUGRF_SIG_DETECT_CON[0]	0	sdmmc detect_n signal rise edge interrupt enable. 1'b0: Disable 1'b1: Enable
sd_detect_fall_edge_detect_en	PMUGRF_SIG_DETECT_CON[1]	0	sd_detect_falling_edge enable 1'b0: Disable 1'b1: Enable
sd_detect_rising_edge_dectect_status	PMUGRF_SIG_DETECT_STATUS[0]	0	sd_detect_rising_edge status 1'b0: Disable 1'b1: Enable
sd_detect_fall_edge_detect_status	PMUGRF_SIG_DETECT_STATUS[1]	0	sd_detect_falling_edge status 1'b0: Disable 1'b1: Enable
sd_detect_rising_edge_dectect_clr	PMUGRF_SIG_DETECT_CLR[0]	0	sd_detect_rising_edge clear 1'b0: Disable 1'b1: Enable
sd_detect_fall_edge_detect_clr	PMUGRF_SIG_DETECT_CLR[1]	0	sd_detect_falling_edge clear 1'b0: Disable 1'b1: Enable

- Method4: Using filtered card_detect_n with the debounce time PMUGRF_SDMMC_DET for interrupt source, connecting to IRQ [92] directly.

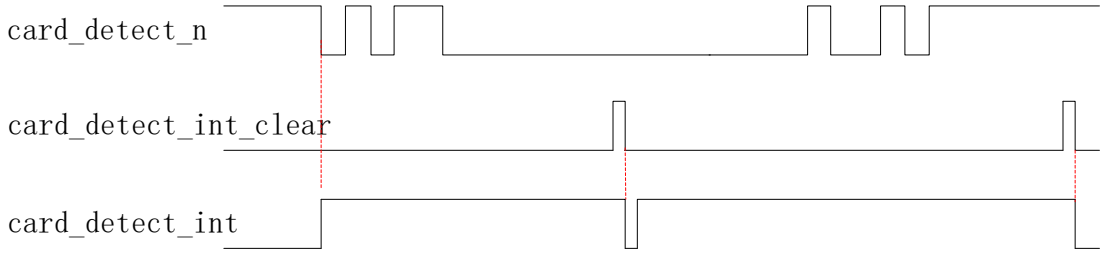


Fig. 11-20 Card Detection Method 4

11.6.11 SDMMC IOMUX With JTAG

The IO for sdmmc_cdata2/sdmmc_cdata3 is shared with jtag_tck/jtag_tms. The condition

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

of usage for SDMMC or JTAG usage is as follows.

- If GRF_CPU_CON1[7](grf_force_jtag) is equal to 1 and sdmmc card is not re-detected after "Card Detection Method 3"within re-detection time(detection time defined in PMUGRF_SDMMC_DET, re-detection time defined in GRF_SOC_CON5), the GPIOs are used for JTAG.
- Otherwise, the GPIOs' usage is defined by IOMUX configuration.

11.6.12 data_nobusy_int Genetation

Software can use data_nobusy_int to detect whether the card is busy or not. The following outlines the steps for data_nobusy_int generation sequence:

1) Set SDMMC_RINTSTS[16] = 1, clear the data_nobusy raw interrupt;
2) Set SDMMC_CTRL[4] = 1, enable global interrupt;
3) Set SDMMC_ INTMASK[16] = 1, enable the data_nobusy interrupt;
4) Set SDMMC_RDYINT_CTRL[7:0] with required value, acts as filtered time;
5) Set SDMMC_RDYINT_CTRL[8] = 1, enable the data_nobusy_int generator;
6) Wait until the Host Controller global interrupt active. Then check SDMMC_RINTSTS[16] and write 1 to clear.
7) Check SDMMC_RDYINT_CTRL[8]. If SDMMC_RDYINT_CTRL[8] equals to 0, then we can say the Host Controller generates data_nobusy_int successfully; otherwise, it is failed.
8) Repeat 4)~7) if data_nobusy_int is needed anytime.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 12 Serial Flash Controller (SFC)

12.1 Overview

The serial flash controller (SFC) is used to control the data transfer between the SoC system and the serial NOR/NAND flash device.
The SFC supports the following features:

- Support AHB slave interface to configure register and read/write serial flash
- Support AHB master interface to transfer data from/to SPI flash device
- Support AHB burst with INCR4 x32bits, or INCR x32bits
- Support two independent clock domain: AHB clock and SPI clock
- Support x1, x2, x4 data bits mode
- Support up to 1 chip selections
- Support interrupt output, interrupt maskable

12.2 Block Diagram

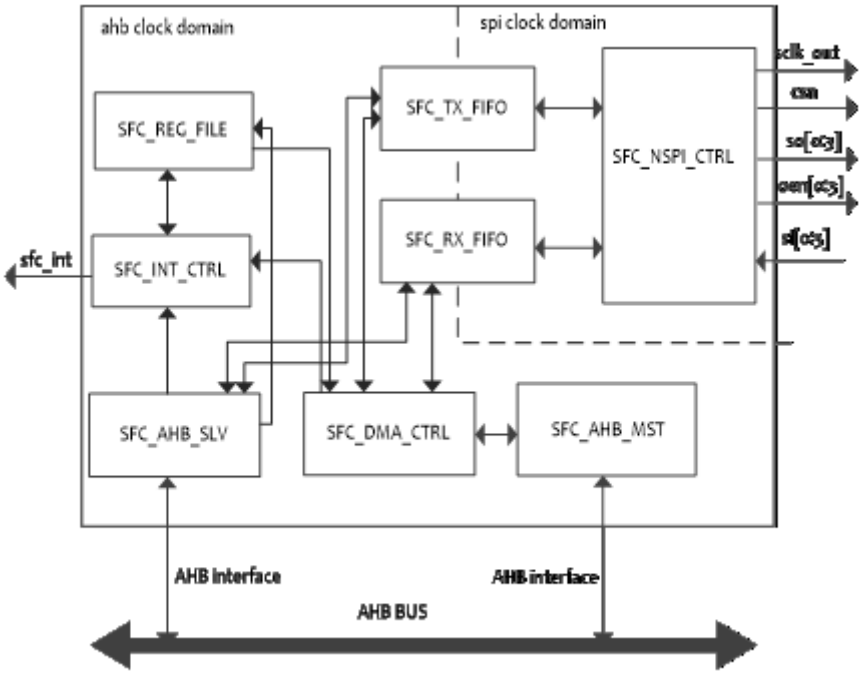


Fig. 12-1 SFC Architecture

12.3 Function Description

12.3.1 SFC AHB slave

The AHB slave is used to configure the register, and also write to/read from the serial NOR/NAND flash device.
The SFC_CTRL register is a global control register, when the controller is in busy state (SFC_SR), SFC_CTRL cannot be set. The field sclk_idle_level_cycles(SFC_CTRL[7:4]) of this register are used to configure the idle level cycles of SFC core clock (sfc_sclk) before reading the first bit of the read command.
Like the following picture shows: the red line of the sclk is the idle cycles, during these cycles, the chip pad is switched to output. When sclk_idle_level_cycles=0, it means there will be not such idle level.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

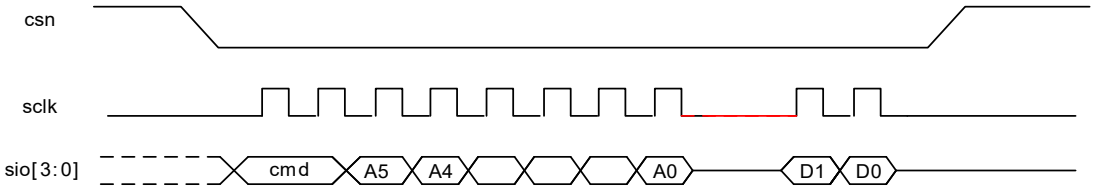


Fig. 12-2 Idle cycles

When the field spi mode is set, the transfer waveform will like following, and switch to mode3.

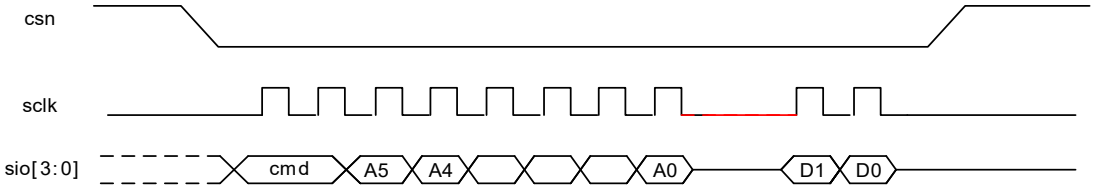


Fig. 12-3 SPI mode

12.4 Register Description

12.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
SFC_CTRL	0x0000	W	0x00000000	Control Register
SFC_IMR	0x0004	W	0x00000000	Interrupt Mask
SFC_ICLR	0x0008	W	0x00000000	Interrupt Clear
SFC_FTLR	0x000c	W	0x00000000	FIFO Threshold Level
SFC_RCVR	0x0010	W	0x00000000	SFC Recover
SFC_AX	0x0014	W	0x00000000	SFC AX Value
SFC_ABIT	0x0018	W	0x00000000	Flash Address bits
SFC_ISR	0x001c	W	0x00000000	Interrupt Status
SFC_FSR	0x0020	W	0x00000001	FIFO Status
SFC_SR	0x0024	W	0x00000000	SFC Status
SFC_RISR	0x0028	W	0x00000000	Raw Interrupt Status
SFC_VER	0x002c	W	0x0a340003	Version Register
SFC_QOP	0x0030	W	0x00000000	Quad line operation io level preset
SFC_DMATR	0x0080	W	0x00000000	DMA Trigger
SFC_DMAADDR	0x0084	W	0x00000000	DMA Address
SFC_CMD	0x0100	W	0x00000000	SFC CMD
SFC_ADDR	0x0104	W	0x00000000	Address of data to read or write.
SFC_DATA	0x0108	W	0x00000000	DATA that write to or read from the flash.

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

12.4.2 Detail Register Description

SFC_CTRL

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:14	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
13:12	RW	0x0	DATB Data bus width. 2'b00: 1bit, x1 mode 2'b01: 2bits, x2 mode 2'b10: 4bits, x4 mode 2'b11: reserved
11:10	RW	0x0	ADRB Address bus width. 2'b00: 1bit, x1 mode 2'b01: 2bits, x2 mode 2'b10: 4bits, x4 mode 2'b11: reserved
9:8	RW	0x0	CMDB Command bus width. 2'b00: 1bit, x1 mode 2'b01: 2bits, x2 mode 2'b10: 4bits, x4 mode 2'b11: reserved
7:4	RW	0x0	IDLE_CYCLE 4'd0: idle hold is disable 4'd1: hold the sclk_out in idle for two cycles when switch to shift in ....
3:2	RO	0x0	reserved
1	RW	0x0	SHIFTPHASE 1'b0: shift in the data at posedge sclk_out 1'b1: shift in the data at negedge sclk_out
0	RW	0x0	SPIM SPI MODE Select. 1'b0: mode 0 1'b1: mode 3

SFC_IMR

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7	RW	0x0	DMAM DMA finish interrupt mask 1'b0: dma_intr interrupt is not masked 1'b1: dma_intr interrupt is masked
6	RW	0x0	NSPIM SPI error interrupt mask 1'b0: nspi_intr interrupt is not masked 1'b1: nspi_intr interrupt is masked

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5	RW	0x0	AHBM AHB error interrupt mask 1'b0: ahb_intr interrupt is not masked 1'b1: ahb_intr interrupt is masked
4	RW	0x0	TRANSM Transfer finish interrupt mask 1'b0: transf_intr interrupt is not masked 1'b1: transf_intr interrupt is masked
3	RW	0x0	TXEM Transmit FIFO empty interrupt 1'b0: txe_intr interrupt is not masked 1'b1: txe_intr interrupt is masked
2	RW	0x0	TXOM Transmit FIFO overflow interrupt mask 1'b0: txo_intr interrupt is not masked 1'b1: txo_intr interrupt is masked
1	RW	0x0	RXUM Receive FIFO underflow interrupt mask 1'b0: rxu_intr interrupt is not masked 1'b1: rxu_intr interrupt is masked
0	RW	0x0	RXFM Receive FIFO full interrupt mask 1'b0: rxf_intr interrupt is not masked 1'b1: rxf_intr interrupt is masked

SFC_ICLR

Address: Operational Base + offset (0x0008)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7	W1C	0x0	DMAC DMA finish Interrupt Clear.
6	W1C	0x0	NSPIC SPI Error Interrupt Clear.
5	W1C	0x0	AHBC AHB Error Interrupt Clear.
4	W1C	0x0	TRANSC Transfer finish Interrupt Clear.
3	W1C	0x0	TXEC Transmit FIFO Empty Interrupt Clear.
2	W1C	0x0	TXOC Transmit FIFO Overflow Interrupt Clear.
1	W1C	0x0	RXUC Receive FIFO Underflow Interrupt Clear.
0	W1C	0x0	RXFC Receive FIFO Full Interrupt Clear.

SFC_FTLR

Address: Operational Base + offset (0x000c)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:8	RW	0x00	RXFTLR When the number of receive FIFO entries is bigger than or equal to this value, the receive FIFO full interrupt is triggered.
7:0	RW	0x00	TXFTLR When the number of transmit FIFO entries is less than or equal to this value, the transmit FIFO empty interrupt is triggered.

SFC_RCVR

Address: Operational Base + offset (0x0010)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	RCVR SFC Recover. Write 1 to recover the SFC State Machine, FIFO state and other logic state.

SFC_AX

Address: Operational Base + offset (0x0014)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	AX The AX Value when doing the continuous read (enhance mode). That is M7-M0 in "Continuous Read Mode". For more information, please refer to specific flash datasheet.

SFC_ABIT

Address: Operational Base + offset (0x0018)

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4:0	RW	0x00	ABIT Flash Address bits.

SFC_ISR

Address: Operational Base + offset (0x001c)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7	RO	0x0	DMAS DMA Finish Interrupt Status. 1'b0: not active 1'b1: active
6	RO	0x0	NSPIS SPI Error Interrupt Status. 1'b0: not active 1'b1: active
5	RO	0x0	AHBS AHB Error Interrupt Status. 1'b0: not active 1'b1: active
4	RO	0x0	TRANSS Transfer finish Interrupt Status. 1'b0: not active 1'b1: active
3	RO	0x0	TXES Transmit FIFO Empty Interrupt Status. 1'b0: not active 1'b1: active
2	RO	0x0	TXOS Transmit FIFO Overflow Interrupt Status. 1'b1: active
1	RO	0x0	RXUS Receive FIFO Underflow Interrupt Status. 1'b0: not active 1'b1: active

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
0	RW	0x0	RXFS Receive FIFO Full Interrupt Status. 1'b0: not active 1'b1: active

SFC_FSR

Address: Operational Base + offset (0x0020)

Bit	Attr	Reset Value	Description
31:21	RO	0x0	reserved
20:16	RW	0x00	RXWLVL RX FIFO Water Level. 5'h0: FIFO is empty 5'h1: 1 entry is taken ... 5'h10: 16 entry is taken, FIFO is full
15:13	RO	0x0	reserved
12:8	RO	0x00	TXWLVL TX FIFO Water Level. 5'h0: FIFO is full 5'h1: left 1 entry ... 5'h10: left 16 entry, FIFO is empty
7:4	RO	0x0	reserved
3	RO	0x0	RXFS Receive FIFO Full Status. 1'b0: rx FIFO is not full 1'b1: rx FIFO is full
2	RO	0x0	RXES Receive FIFO Empty Status. 1'b0: rx FIFO is not empty 1'b1: rx FIFO is empty
1	RO	0x0	TXES Transmit FIFO Empty Status. 1'b0: tx FIFO is not empty 1'b1: tx FIFO is empty
0	RO	0x1	TXFS Transmit FIFO Full Status. 1'b0: tx FIFO is not full 1'b1: tx FIFO is full

SFC_SR

Address: Operational Base + offset (0x0024)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	SR 1'b0: SFC is idle 1'b1: SFC is busy When busy, don't set the control register. When idle, the rx FIFO and tx FIFO are all empty.

SFC_RISR

Address: Operational Base + offset (0x0028)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7	RO	0x0	DMAS DMA Finish Interrupt Status. 1'b0: not active 1'b1: active
6	RO	0x0	NSPIS SPI Error Interrupt Status. 1'b0: not active 1'b1: active
5	RO	0x0	AHBS AHB Error Interrupt Status. 1'b0: not active 1'b1: active
4	RO	0x0	TRANSS Transfer finish Interrupt Status. 1'b0: not active 1'b1: active
3	RO	0x0	TXES Transmit FIFO Empty Interrupt Status. 1'b0: not active 1'b1: active
2	RO	0x0	TXOS Transmit FIFO Overflow Interrupt Status. 1'b0: not active 1'b1: active
1	RO	0x0	RXUS Receive FIFO Underflow Interrupt Status. 1'b0: not active 1'b1: active
0	RO	0x0	RXFS Receive FIFO Full Interrupt Status. 1'b0: not active 1'b1: active

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

SFC_VER

Address: Operational Base + offset (0x002c)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	Reserved. The value is undefined.
15:0	RO	0x3	VER The version id of sfc. 16'h1: this SFC does not have QOP register 16'h3: the SFC has QOP register. Others: reserved

SFC_QOP

Address: Operational Base + offset (0x0030)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	SO123 The value of SIO1, SIO2 and SIO3 during command and address bits output.

SFC_DMATR

Address: Operational Base + offset (0x0080)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	W1C	0x0	DMATR Write 1 to start the dma transfer.

SFC_DMAADDR

Address: Operational Base + offset (0x0084)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	DMAADDR DMA Address.

SFC_CMD

Address: Operational Base + offset (0x0100)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:30	WO	0x0	CS Flash chip select. 2'b00: chip select 0 Others: reserved
29:16	WO	0x0000	TRB Total Data Bytes number that will write to /read from the flash. In DMA mode, this register must be aligned to 2 bytes.
15:14	WO	0x0	ADDRB Address bits number select, if there is not address command to send, set to zero. 2'b00: 0bits 2'b01: 24bits 2'b10: 32bits 2'b11: From the ABIT register
13	WO	0x0	CONT Continuous read mode. 1'b0: disable continuous read mode 1'b1: enable continuous read mode Please refer to specific flash datasheet for more information.
12	WO	0x0	WR Flash Write or Read. 1'b0: read 1'b1: write
11:8	WO	0x0	DUMM Dummy Bits Number.
7:0	WO	0x00	CMD Flash Command.

SFC_ADDR

Address: Operational Base + offset (0x0104)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	ADDR Flash's address.

SFC_DATA

Address: Operational Base + offset (0x0108)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	DATA Flash's Data. The LSB of this data will be sent first.

12.5 Interface Description

Table 12-1 SFC interface description

Module Pin	Direction	Pin Name	IOMUX Setting
sfc_clk	O	GPIO1_A5/EMMC_D5/SFC_CLK	GRF_GPIO1A_IOMUX_H[7:4]=2'b10
sfc_csn0	O	GPIO1_A4/EMMC_D4/SFC_CSN0	GRF_GPIO1A_IOMUX_H[3:0]=2'b10

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Module Pin	Direction	Pin Name	IOMUX Setting
sfc_sio0	I/O	GPIO1_A0/EMMC_D0/SFC_SIO0	GRF_GPIO1A_IOMUX_L[3:0]=2'b10
sfc_sio1	I/O	GPIO1_A1/EMMC_D1/SFC_SIO1	GRF_GPIO1A_IOMUX_L[7:4]=2'b10
sfc_sio2	I/O	GPIO1_A2/EMMC_D2/SFC_WP_SIO2	GRF_GPIO1A_IOMUX_L[11:8]=2'b10
sfc_sio3	I/O	GPIO1_A3/EMMC_D3/SFC_HOLD_SIO3	GRF_GPIO1A_IOMUX_L[15:12]=2'b10

Notes: I=input, O=output, I/O=input/output, bidirectional.

12.6 Application Notes

12.6.1 AHB Slave write flash flow (non-DMA mode)

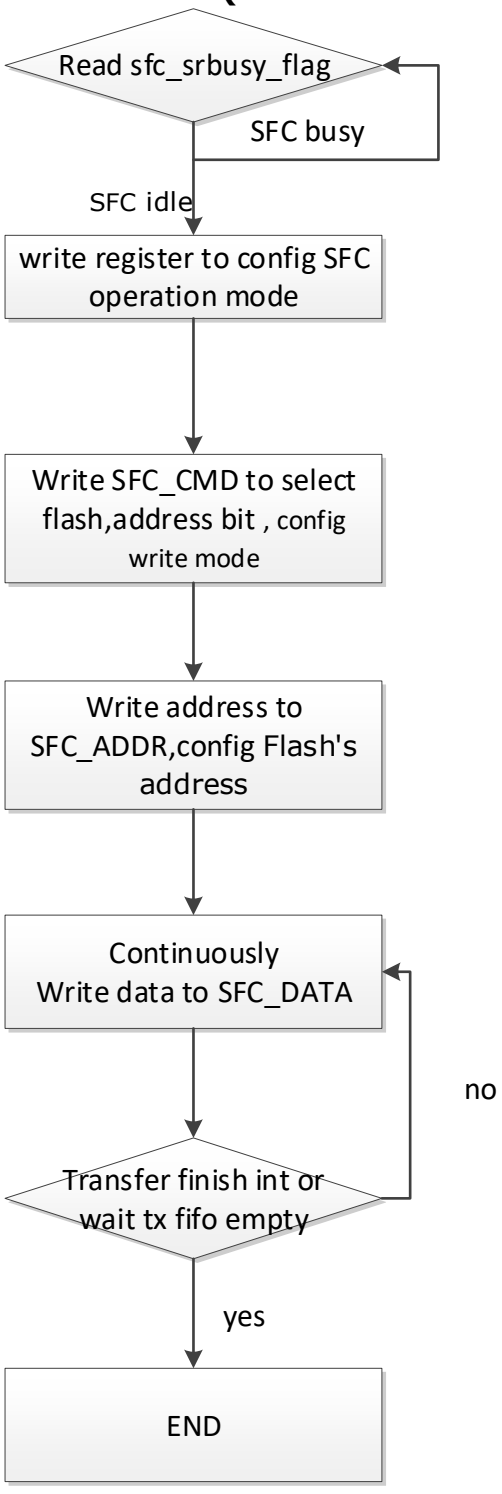


Fig. 12-4 Slave mode write

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

All the AHB bus write data to SFC_CMD, SFC_ADDR and SFC_DATA will be marked with different header and then pushed into transmit FIFO by writing order.

12.6.2 AHB Slave read flash flow (non-DMA mode)

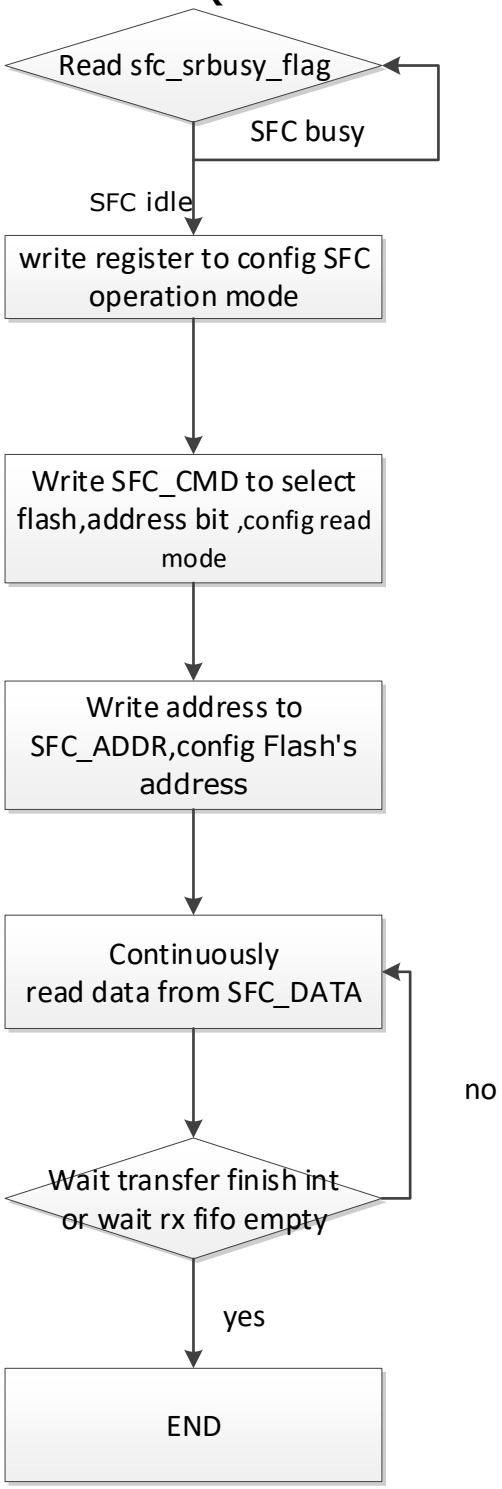


Fig. 12-5 Slave mode read

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

12.6.3 AHB DMA transfer flow (DMA mode)

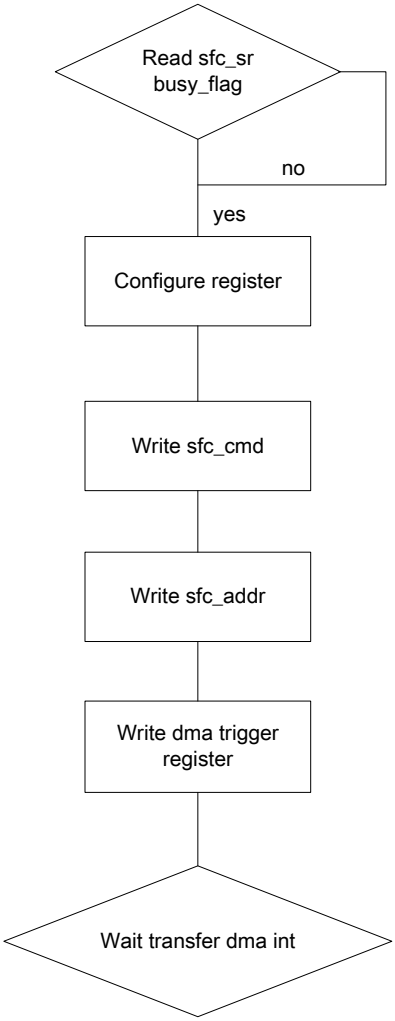


Fig. 12-6 Master mode flow

The total transfer bytes is decided by TRB register in SFC_CMD and must be aligned to 2 bytes.

12.6.4 SPI Mode and Shift Phase

The register SPIM in SFC_CTRL will decide the default value of sclk_out. When SPIM=0, the default value is 0. When SPIM=1, the default value is 1. The register SHIFTPAHSE in SFC_CTRL will decide when to sample the SIO data. If SHIFTPAHSE=0, it will sample the data at the posedge of sclk_out. If SHIFTPHASE=1, it will sample the data at the negedge of sclk_out.

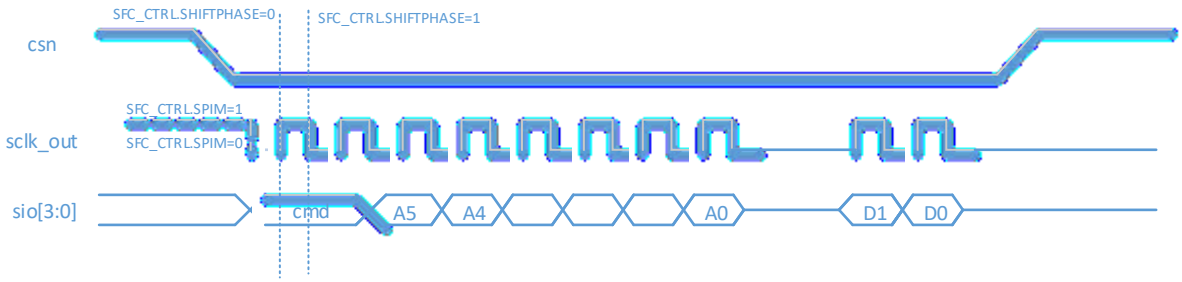


Fig. 12-7 SPI mode

12.6.5 Other Notes

The SFC core clock(SFC_sclk)need to be kept under 100MHZ and the SFC interface clock(sfc_clk) is a half of the SFC_sclk. It's better to soft reset the SFC before data transfer.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 13 Embedded SRAM

13.1 Overview

There are two embedded SRAMs, SYSTEM_SRAM and PMU_SRAM.

13.1.1 Features supported

- SYSTEM_SRAM
  - Provide 2MB access space
  - Support security and non-security access
  - Secure or non-secure space is software programmable, the lower N*4KB is secure and the other space is non-secure, where N is defined by SGRF_SOC_CON1[9:0]

- PMU_SRAM
  - Provide 8KB access space
  - Support secure access only

13.2 Block Diagram

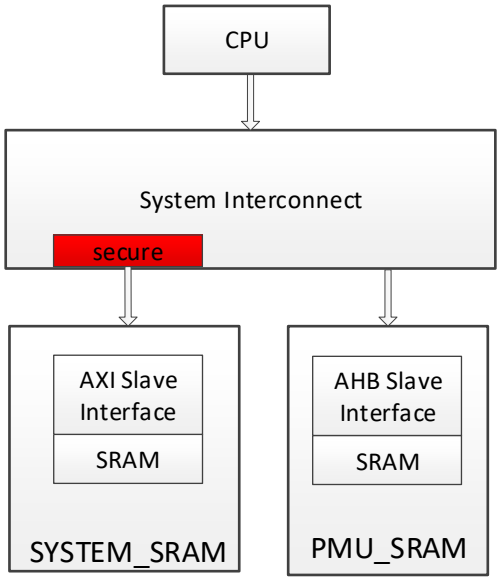


Fig. 13-1 Embedded SRAM block diagram

13.3 Function Description

13.3.1 AXI slave interface of SYSTEM_SRAM

The AXI slave interface is bridge which translate AXI bus access to SRAM interface of SYSTEM_SRAM.

13.3.2 AXI slave interface of PMU_SRAM

The AHB slave interface is bridge which translate AHB bus access to SRAM interface of PMU_SRAM.

13.3.3 Embedded SRAM access path

The SYSTEM_SRAM can be accessed by all IP masters (SPI2APB cannot access SYSTEM_SRAM if sgrf_soc_con1[12] is 1, the default value is 0). The PMU_SRAM can only be accessed by CPU and DMAC.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 14 Spinlock

14.1 Overview

The hardware spinlock used for spinlock status storage. All the CPU can access spinlock to get the lock status.

14.2 Block Diagram

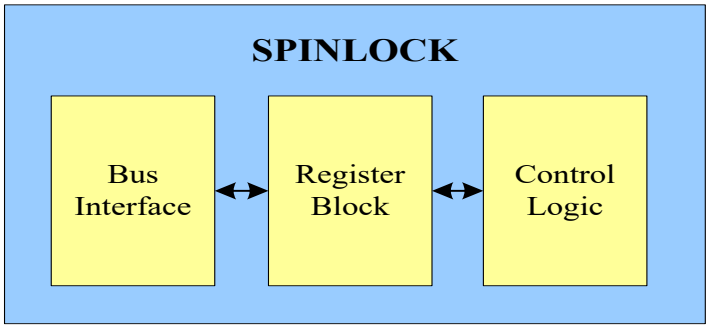


Fig. 14-1 Spinlock Block Diagram

- Bus Interface

The Bus Interface implements the bus slave operation.

- Register Block

The Register block includes the lock status registers.

- Control Logic

The control logic implemented the spinlock function.

14.3 Function Description

The SPINLOCK includes 64 lock-status register groups. Each lock-status register has 4bits, which corresponding to the lock status.
When 4bits spinlock_status is not zero, this lock-status register cannot be written.
When write data is 4'b0, 4bits spinlock_status registers clean to all zero.
If write data is 4'b0, that means unlock process.
If write data is 4bits ID, that means lock process. If read back value is equal to write data ID, that means lock successfully.

14.4 Register Description

14.4.1 Internal Address Mapping

Slave address can be divided into different length for different usage, which is shown as follows.

14.4.2 Registers Summary

Name	Offset	Size	Reset Value	Description
SPINLOCK_status_n (range of n is 0~63)	0x0+4*n	W	0x00000000	spinlock status controller register_n

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

14.4.3 Detail Register Description

SPINLOCK_status_n

Address: Operational Base + offset (0x0000+4*n)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:4	RO	0x0000000	reserved reserved
3:0	RW	0x0	spinlock_status when 4bits is 0, 4bits can be written with new value. when 4bits is not 0, 4bits cannot be written. when write data is 0x0000, 4bits clean to 0.

14.5 Register Description

Please refer to the "Function Description" section.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 15 Neural Process Unit (NPU)

15.1 Overview

NPU is the process unit which is dedicated to neural network. It is designed to accelerate the neural network arithmetic in field of AI (artificial intelligence) such as machine vision and natural language processing. The variety of applications for AI is expanding, and currently provides functionality in a variety of areas, including face tracking as well as gesture and body tracking, image classification, video surveillance, automatic speech recognition (ASR) and advanced driver assistance systems (ADAS).

NPU supports the following features:

- Host interface
  - 32bit AHB interface used for configuration only support single
  - 128bit AXI interface used to fetch data from memory
- Neural Network
  - Support integer 8, integer 16, float 16 convolution operation
  - 1920 MAD (multiply-add units) per cycle (int 8)
  - 192 MAD per cycle (int 16)
  - 64 MAD per cycle (float 16)
  - Support Liner, MIMO, Fully Connected, Fully Convolution
  - Unlimited network size (bound by system resource)
  - Inference Engine : TensorFlow backend, OpenCl, OpenVX, Android NN backend
  - Support network sparse coefficient decompression
  - Support Max, average pooling
  - Max pooling support 2x2, 3x3, stride <= min(input width, input height)
  - Local average pooling size <= 11x11
  - Support unpooling
  - Support batch normalize, l2 normalize, l2 normalize scale, local response normalize
  - Support region proposal
  - Support permute, reshape, concat, depth to space, space to depth, flatten, reorg, squeeze and split
  - Support priorbox layer
  - Support Non-max Suppression
  - Support ROI pooling
  - Convolution size NXN , N <=11*stride, stride <= min(input width, input height)
  - Support dilate convolution, N <=11*stride, stride <= min(input width, input height), dilation <1024
  - Support de-convolution, N <=11*stride, stride <= min(input width, input height)
  - Support Elementwise addition, div, floor, max, mul, scale, sub
  - Support elu, leaky_relu, prelu, relu, relu1, relu6, sigmod, softmax, tanh
  - Support LSTM, RNN
  - Support channel shuffle
  - Support dequantize, dropout.
  - Include embedded lookup table
  - Support hashtable lookup
  - Support lsh projection
  - Support svdf
  - Support reserve

15.2 Block Diagram

NPU comprises with:

- HIF: Host Interface
- PM : Power Management
- NN Engine : Neural Network Engine
- VPU : Vector Procession Unit

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

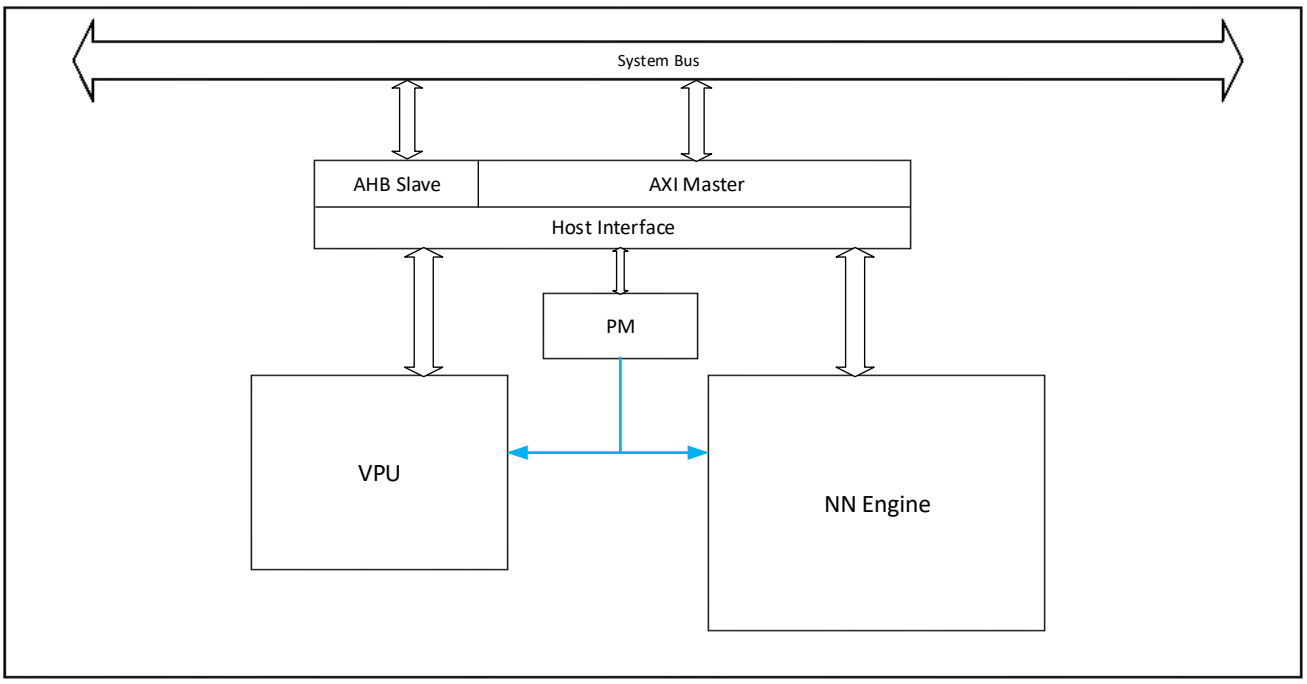


Fig. 15-1 NPU Block Diagram

15.3 Function Description

15.3.1 Host Interface

Allows the NPU to communicate with external memory and the CPU through the AXI or AHB bus. In this block data crosses clock domain boundaries. It includes a Front End (FE) to insert high level primitives and commands into the calculation pipeline. There is an AHB slave and AXI master in Host inter face unit. The AXI master interface is used to fetch data from memory that is attached to the Soc AXI interconnect. The AHB slave interface is used to access the graphics registers for configuration, debug and test.

15.3.2 Power Management

Power management in NPU is used to provide top level controls for clock, reset and power management. AHB clock, AXI clock, clock for VPU and NN Engine are all asynchronization, and clock pins are connected to Power Management. The reset of all models are also controlled by Power Management. Global clock gating can be controlled by Power Management to reduce power.

15.3.3 Neural Network Engine

As the unit name, NN Engine is the main process unit for Neural Network arithmetic. This unit Provides parallel convolution MAC for recognition functions and int8, int16 and fp16 are supported. Active functions and pooling such as leaky_relu, relu, relu1, relu6, sigmod, tanh are also processed in NN Engine. So NN Engine is mainly serve for convolution neural network and fully connected network.

15.3.4 Vector Processing Unit

Vector Processing Unit can be the supplement for NN Engine. The programmable SIMD processor unit is included which perform as a Compute Unit for OpenCL. VPU provides advanced image processing functions. For example, in one cycle, VPU can perform one MUL/ADD instruction or a dot product of two 16-component values. Most element wise operations and matrix operations are processed in VPU.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

15.4 Register Description

15.4.1 Internal Address Mapping

Slave address can be divided into different length for different usage, which is shown as follows.

Table 15-1 NPU Address Mapping

Base Address[12:8]	Device	Address Length	Offset Address Range
0xffbc0000	NPU	512K BYTE	0x00000 ~ 0x7ffff

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 16 Video Input Processor(VIP)

16.1 Overview

The Video Input Processor, receives the data from Camera via DVP/MIPI, and transfers the data into system main memory by AXI bus.
The features of VIP are as follow:

- Support BT601 YCbCr 422 8bit input
- Support BT656 YCbCr 422 8bit input
- Support UYVY/VYUY/YUYV/YVYU configurable
- Support RAW 8/10/12 bit input
- Support JPEG input
- Support BT1120 16bit input,single/dual-edge sampling
- Support receiving CSI2 protocol data(up to four IDs)
- Support receiving DSI protocol data(Video mode/Command mode)
- Support window cropping
- Support virtual stride when write to DDR
- Support different stored address for Y and UV
- Support 422/420 output
- Support the polarity of pixel_clk、hsync、vsync configurable

16.2 Block Diagram

VIP comprises with:

- AHB Slave

Host configure the registers via the AHB Slave

- AXI Master

Transmit the data to chip memory via the AXI Master

- MMU

Map the virtual address to physical address

- INTERFACE

Translate the input video data into the requisite data format

- CROP

Bypass or crop the source video data to a smaller size destination

- DMA

Control the operation of AXI Master

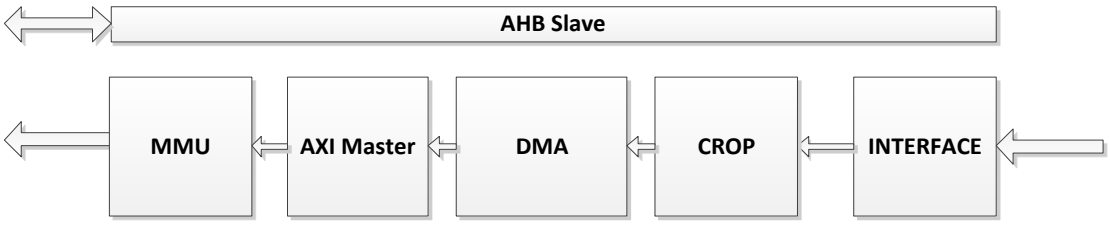


Fig. 16-1 VIP Block Diagram

16.3 Function Description

16.3.1 Interface

Interface module is designed to receive DVP/MIPI data and transfer to pixel data.

16.3.2 Crop

Crop module is used to crop the received image.

16.3.3 DMA

The DMA is used to transfer the data from crop module to the AXI master block which will send the data to the AXI bus.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

16.3.4 AXI Master

There is an AXI master in VIP,it is responsible for transferring the DMA output data to AXI bus.

16.3.5 MMU

There is a MMU in VIP, it is responsible for mapping the virtual address to physical address.

16.4 Register Description

16.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
VIP_DVP_CTRL	0x0000	W	0x00007000	DVP path control
VIP_DVP_INTEN	0x0004	W	0x00000000	DVP path interrupt status
VIP_DVP_INTSTAT	0x0008	W	0x00000000	DVP path interrupt status
VIP_DVP_FOR	0x000c	W	0x00000000	DVP path format
VIP_DVP_DMA_IDLE_REQ	0x0010	W	0x00000000	DVP path dma module idle requeset
VIP_DVP_FRM0_ADDR_Y	0x0014	W	0x00000000	DVP path frame0 y address
VIP_DVP_FRM0_ADDR_UV	0x0018	W	0x00000000	DVP path frame0 uv address
VIP_DVP_FRM1_ADDR_Y	0x001c	W	0x00000000	DVP path frame1 y address
VIP_DVP_FRM1_ADDR_UV	0x0020	W	0x00000000	DVP path frame1 uv address
VIP_DVP_VIR_LINE_WIDTH	0x0024	W	0x00000000	DVP path virtual line width
VIP_DVP_SET_SIZE	0x0028	W	0x01e002d0	The expected width and height of received image
VIP_DVP_LINE_INT_NUM	0x002c	W	0x00000040	DVP path line interrupt number
VIP_DVP_LINE_CNT	0x0030	W	0x00000000	DVP path line count
VIP_DVP_CROP	0x0044	W	0x00000000	The start point of DVP path cropping
VIP_DVP_PATH_SEL	0x0048	W	0x00000000	DVP path selection
VIP_DVP_FIFO_ENTRY	0x0054	W	0x00000000	DVP path FIFO entry
VIP_DVP_FRAME_STATUS	0x0060	W	0x00000000	DVP path frame status
VIP_DVP_CUR_DST	0x0064	W	0x00000000	DVP path current destination address
VIP_DVP_LAST_LINE	0x0068	W	0x00000000	DVP path last frame line number
VIP_DVP_LAST_PIX	0x006c	W	0x00000000	DVP path last line pixel number
VIP_MIPI_ID0_CTRL0	0x0080	W	0x00000000	MIPI path id0 control0
VIP_MIPI_ID0_CTRL1	0x0084	W	0x00000000	MIPI path id0 control1
VIP_MIPI_ID1_CTRL0	0x0088	W	0x00000000	MIPI path id1 control0
VIP_MIPI_ID1_CTRL1	0x008c	W	0x00000000	MIPI path id1 control1
VIP_MIPI_ID2_CTRL0	0x0090	W	0x00000000	MIPI path id2 control0
VIP_MIPI_ID2_CTRL1	0x0094	W	0x00000000	MIPI path id2 control1
VIP_MIPI_ID3_CTRL0	0x0098	W	0x00000000	MIPI path id3 control0
VIP_MIPI_ID3_CTRL1	0x009c	W	0x00000000	MIPI path id3 control1
VIP_MIPI_WATER_LINE	0x00a0	W	0x00000010	DMA fifo water line for MIPI data path

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
VIP_MIPI_FRAME0_ADDR_Y_ID0	0x00a4	W	0x00000000	First address of even frame for ID0 Y/RAW/RGB path
VIP_MIPI_FRAME1_ADDR_Y_ID0	0x00a8	W	0x00000000	First address of odd frame for ID0 Y path
VIP_MIPI_FRAME0_ADDR_UV_ID0	0x00ac	W	0x00000000	First address of even frame for ID0 UV path
VIP_MIPI_FRAME1_ADDR_UV_ID0	0x00b0	W	0x00000000	First address of odd frame for ID0 UV path
VIP_MIPI_FRAME0_VLW_Y_ID0	0x00b4	W	0x00000000	Virtual line width of even frame for ID0 Y/RAW/RGB path
VIP_MIPI_FRAME1_VLW_Y_ID0	0x00b8	W	0x00000000	Virtual line width of odd frame for ID0 Y/RAW/RGB path
VIP_MIPI_FRAME0_VLW_UV_ID0	0x00bc	W	0x00000000	Virtual line width of even frame for ID0 UV path
VIP_MIPI_FRAME1_VLW_UV_ID0	0x00c0	W	0x00000000	Virtual line width of odd frame for ID0 UV path
VIP_MIPI_FRAME0_ADDR_Y_ID1	0x00c4	W	0x00000000	First address of even frame for ID1 Y/RAW/RGB path
VIP_MIPI_FRAME1_ADDR_Y_ID1	0x00c8	W	0x00000000	First address of odd frame for ID1 Y/RAW/RGB path
VIP_MIPI_FRAME0_ADDR_UV_ID1	0x00cc	W	0x00000000	First address of even frame for ID1 UV path
VIP_MIPI_FRAME1_ADDR_UV_ID1	0x00d0	W	0x00000000	First address of odd frame for ID1 UV path
VIP_MIPI_FRAME0_VLW_Y_ID1	0x00d4	W	0x00000000	Virtual line width of even frame for ID1 Y/RAW/RGB path
VIP_MIPI_FRAME1_VLW_Y_ID1	0x00d8	W	0x00000000	Virtual line width of odd frame for ID1 Y path
VIP_MIPI_FRAME0_VLW_UV_ID1	0x00dc	W	0x00000000	Virtual line width of even frame for ID1 UV path
VIP_MIPI_FRAME1_VLW_UV_ID1	0x00e0	W	0x00000000	Virtual line width of odd frame for ID1 UV path
VIP_MIPI_FRAME0_ADDR_Y_ID2	0x00e4	W	0x00000000	First address of even frame for ID2 Y/RAW/RGB path
VIP_MIPI_FRAME1_ADDR_Y_ID2	0x00e8	W	0x00000000	First address of odd frame for ID2 Y/RAW/RGB path
VIP_MIPI_FRAME0_ADDR_UV_ID2	0x00ec	W	0x00000000	First address of even frame for ID2 UV path
VIP_MIPI_FRAME1_ADDR_UV_ID2	0x00f0	W	0x00000000	First address of odd frame for ID2 UV path
VIP_MIPI_FRAME0_VLW_Y_ID2	0x00f4	W	0x00000000	Virtual line width of even frame for ID2 Y/RAW/RGB path

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
VIP_MIPI_FRAME1_VLW_Y_ID2	0x00f8	W	0x00000000	Virtual line width of odd frame for ID2 Y/RAW/RGB path
VIP_MIPI_FRAME0_VLW_UV_ID2	0x00fc	W	0x00000000	Virtual line width of even frame for ID2 UV path
VIP_MIPI_FRAME1_VLW_UV_ID2	0x0100	W	0x00000000	Virtual line width of odd frame for ID2 UV path
VIP_MIPI_FRAME0_ADDR_Y_ID3	0x0104	W	0x00000000	First address of even frame for ID3 Y/RAW/RGB path
VIP_MIPI_FRAME1_ADDR_Y_ID3	0x0108	W	0x00000000	First address of odd frame for ID3 Y/RAW/RGB path
VIP_MIPI_FRAME0_ADDR_UV_ID3	0x010c	W	0x00000000	First address of even frame for ID3 UV path
VIP_MIPI_FRAME1_ADDR_UV_ID3	0x0110	W	0x00000000	First address of odd frame for ID3 UV path
VIP_MIPI_FRAME0_VLW_Y_ID3	0x0114	W	0x00000000	Virtual line width of even frame for ID3 Y/RAW/RGB path
VIP_MIPI_FRAME1_VLW_Y_ID3	0x0118	W	0x00000000	Virtual line width of odd frame for ID3 Y/RAW/RGB path
VIP_MIPI_FRAME0_VLW_UV_ID3	0x011c	W	0x00000000	Virtual line width of even frame for ID3 UV path
VIP_MIPI_FRAME1_VLW_UV_ID3	0x0120	W	0x00000000	Virtual line width of odd frame for ID3 UV path
VIP_MIPI_INTEN	0x0124	W	0x00000000	MIPI path interrupt enable
VIP_MIPI_INTSTAT	0x0128	W	0x00000000	MIPI path interrupt status
VIP_MIPI_LINE_INT_NUM_ID0_1	0x012c	W	0x00400040	Line number of the MIPI path ID0/1 line interrupt
VIP_MIPI_LINE_INT_NUM_ID2_3	0x0130	W	0x00400040	Line number of the MIPI path ID2/3 line interrupt
VIP_MIPI_LINE_CNT_ID0_1	0x0134	W	0x00000000	Line count of the MIPI path ID0/1
VIP_MIPI_LINE_CNT_ID2_3	0x0138	W	0x00000000	Line count of the MIPI path ID2/3
VIP_MIPI_ID0_CROP_START	0x013c	W	0x00000000	The start point of MIPI ID0 cropping
VIP_MIPI_ID1_CROP_START	0x0140	W	0x00000000	The start point of MIPI ID1 cropping
VIP_MIPI_ID2_CROP_START	0x0144	W	0x00000000	The start point of MIPI ID2 cropping
VIP_MIPI_ID3_CROP_START	0x0148	W	0x00000000	The start point of MIPI ID3 cropping
VIP_MMU_DTE_ADDR	0x0800	W	0x00000000	MMU current page table address
VIP_MMU_STATUS	0x0804	W	0x00000000	MMU status register
VIP_MMU_COMMAND	0x0808	W	0x00000000	MMU command register
VIP_MMU_PAGE_FAULT_ADDR	0x080c	W	0x00000000	MMU logical address of last page fault

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
VIP_MMU_ZAP_ONE_LINE	0x0810	W	0x00000000	MMU Zap cache line register
VIP_MMU_INT_RAWSTAT	0x0814	W	0x00000000	MMU raw interrupt status register
VIP_MMU_INT_CLEAR	0x0818	W	0x00000000	MMU interrupt status clear
VIP_MMU_INT_MASK	0x081c	W	0x00000000	MMU interrupt mask
VIP_MMU_INT_STATUS	0x0820	W	0x00000000	MMU interrupt status
VIP_MMU_AUTO_GATING	0x0824	W	0x00000000	MMU auto gating

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

16.4.2 Detail Register Description

VIP_DVP_CTRL

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:12	RW	0x7	axi_burst_type 0-15 : Burst1~16
11:3	RO	0x0	reserved
2:1	RW	0x0	work_mode 00 : One frame stop mode 01 : Ping-pong mode 02 : Reserved 03 : Reserved Note: BT1120 only support ping-pong mode
0	RW	0x0	cap_en 0 : Disable 1 : Enable

VIP_DVP_INTEN

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:11	RO	0x0	reserved
10	RW	0x0	line_int_en The specified line end interrupt enable 1'b0 : Disable 1'b1 : Enable
9	RW	0x0	pst_inf_frame_end_en Frame end after interface FIFO interrupt enable 0 : Disable 1 : Enable
8	RW	0x0	pre_inf_frame_end_en Frame end before interface FIFO interrupt enable 0 : Disable 1 : Enable
7	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
6	W1C	0x0	bus_err_en Axi master or ahb slave response error interrupt enable 0 : Disable 1 : Enable
5	RW	0x0	dfifo_of_en DMA FIFO overflow interrupt enable 0 : Disable 1 : Enable
4	RW	0x0	ififo_of_en Interface FIFO overflow interrupt enable 0 : Disable 1 : Enable
3	W1C	0x0	pix_err_en The pixel number of last line not equal to the set height interrupt enable 0 : Disable 1 : Enable
2	W1C	0x0	line_err_en The line number of last frame not equal to the set height interrupt enable 0 : Disable 1 : Enable
1	W1C	0x0	line_end_en Line end interrupt enable 0 : Disable 1 : Enable
0	W1C	0x0	dma_frame_end_en Dma frame end interrupt enable 0 : Disable 1 : Enable

VIP_DVP_INTSTAT

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:11	RO	0x0	reserved
10	RW	0x0	line_int The specified line end interrupt 0 : No interrupt 1 : Interrupt
9	RW	0x0	pst_inf_frame_end Frame end after interface FIFO interrupt 0 : No interrupt 1 : Interrupt

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
8	RW	0x0	pre_inf_frame_end Frame end before interface FIFO interrupt 0 : No interrupt 1 : Interrupt
7	RO	0x0	reserved
6	W1C	0x0	bus_err Axi master or ahb slave response error interrupt 0 : No interrupt 1 : Interrupt
5	RW	0x0	dfifo_of DMA FIFO overflow interrupt 0 : No interrupt 1 : Interrupt
4	RW	0x0	ififo_of Interface FIFO overflow interrupt 0 : No interrupt 1 : Interrupt
3	W1C	0x0	pix_err The pixel number of last line not equal to the set height interrupt 0 : No interrupt 1 : Interrupt
2	W1C	0x0	line_err The line number of last frame not equal to the set height interrupt 0 : No interrupt 1 : Interrupt
1	W1C	0x0	line_end Line end interrupt 0 : No interrupt 1 : Interrupt
0	W1C	0x0	dma_frame_end Dma frame end interrupt 0 : No interrupt 1 : Interrupt

VIP_DVP_FOR

Address: Operational Base + offset (0x000c)

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved
26	RW	0x0	sw_yc_swap 1'b0 : No swap for y and c 1'b1 : Swap for y and c Only for BT1120 mode

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
25	RW	0x0	sw_progress_en 1'b0 : Interlace 1'b1 : Progress Only for BT1120 mode
24	RW	0x0	sw_dualedge_en 1'b0 : Only use single edge of clock 1'b1 : Use double edges of clock Only for BT1120 mode
23:20	RO	0x0	reserved
19	RW	0x0	uv_store_order 0 : UVUV 1 : VUVU
18	RW	0x0	raw_end 0 : Little end 1 : Big end
17	RW	0x0	out_420_order 0 : UV in the even line 1 : UV in the odd line Note: The first line is even line(line 0)
16	RW	0x0	output_420 0 : Output is 422 1 : Output is 420
15:13	RO	0x0	reserved
12:11	RW	0x0	raw_width 00 : 8bit raw data; 01 : 10bit raw data; 10 : 12bit raw data; 11 : Reserve;
10	RW	0x0	jpeg_mode 0 : Other mode 1 : Mode1
9	RW	0x0	field_order 0 : Odd field first 1 : Even field first
8:7	RO	0x0	reserved
6:5	RW	0x0	yuv_in_order 00 : UYVY 01 : YVYU 10 : VYUY 11 : YUYV

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4:2	RW	0x0	input_mode 000 : YUV 010 : PAL 011 : NTSC 100 : RAW 101 : JPEG 110 : Reserved 111：BT1120
1	RW	0x0	href_pol 0 : High active 1 : Low active
0	RW	0x0	vsync_pol 0 : Low active 1 : High active

VIP_DVP_DMA_IDLE_REQ

Address: Operational Base + offset (0x0010)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	dma_idle_req Write 1 will stop VIP dma. When this bit change to 0,dma is stopped really

VIP_DVP_FRM0_ADDR_Y

Address: Operational Base + offset (0x0014)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	frm0_addr_y DVP path frame0 y address

VIP_DVP_FRM0_ADDR_UV

Address: Operational Base + offset (0x0018)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	frm0_addr_uv DVP path frame0 uv address

VIP_DVP_FRM1_ADDR_Y

Address: Operational Base + offset (0x001c)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	frm1_addr_y DVP path frame1 y address

VIP_DVP_FRM1_ADDR_UV

Address: Operational Base + offset (0x0020)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	frm1_addr_uv DVP path frame1 uv address

VIP_DVP_VIR_LINE_WIDTH

Address: Operational Base + offset (0x0024)

Bit	Attr	Reset Value	Description
31:15	RO	0x0	reserved
14:0	RW	0x0000	vir_line_width DVP path virtual line width

VIP_DVP_SET_SIZE

Address: Operational Base + offset (0x0028)

Bit	Attr	Reset Value	Description
31:29	RO	0x0	reserved
28:16	RW	0x01e0	set_height The expected height of received image
15:13	RO	0x0	reserved
12:0	RW	0x02d0	set_width The expected width of received image

VIP_DVP_LINE_INT_NUM

Address: Operational Base + offset (0x002c)

Bit	Attr	Reset Value	Description
31:14	RO	0x0	reserved
13:0	RW	0x0040	line_int_num If line_int_num=100,then vip receive 100th line菔th line蔔th line.....the line_int will be 1

VIP_DVP_LINE_CNT

Address: Operational Base + offset (0x0030)

Bit	Attr	Reset Value	Description
31:14	RO	0x0	reserved
13:0	RO	0x0000	line_cnt Current line count

VIP_DVP_CROP

Address: Operational Base + offset (0x0044)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:29	RO	0x0	reserved
28:16	RW	0x0000	start_y The vertical ordinate of the start point
15:13	RO	0x0	reserved
12:0	RW	0x0000	start_x The horizontal ordinate of the start point

VIP_DVP_PATH_SEL

Address: Operational Base + offset (0x0048)

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5	RW	0x0	raw_sel 0 : Not select 1 : Select
4	RW	0x0	yuv_sel 0 : Not select 1 : Select
3:0	RO	0x0	reserved

VIP_DVP_FIFO_ENTRY

Address: Operational Base + offset (0x0054)

Bit	Attr	Reset Value	Description
31:18	RO	0x0	reserved
17:9	RW	0x000	uv_fifo_entry Write 0 clear
8:0	RO	0x000	y_fifo_entry Write 0 clear

VIP_DVP_FRAME_STATUS

Address: Operational Base + offset (0x0060)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RO	0x0000	frame_num Completed frame number Write 0 to clear
15:3	RO	0x0	reserved
2	RO	0x0	idle 1'b0: Work 1'b1: Idle
1	RO	0x0	f1_sts 0 : Frame 1 not ready 1 : Frame 1 ready Write 0 clear
0	RO	0x0	f0_sts 0 : Frame 0 not ready 1 : Frame 0 ready Write 0 clear

VIP_DVP_CUR_DST

Address: Operational Base + offset (0x0064)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	cur_dst DVP path current destination address

VIP_DVP_LAST_LINE

Address: Operational Base + offset (0x0068)

Bit	Attr	Reset Value	Description
31:29	RO	0x0	reserved
28:16	RW	0x0000	last_uv_num UV line number of last frame,only for bt1120 mode
15:13	RO	0x0	reserved
12:0	RO	0x0000	last_y_num Y line number of last frame

VIP_DVP_LAST_PIX

Address: Operational Base + offset (0x006c)

Bit	Attr	Reset Value	Description
31:29	RO	0x0	reserved
28:16	RW	0x0000	last_uv_num DVP path last line uv number
15:13	RO	0x0	reserved
12:0	RO	0x0000	last_y_num DVP path last line y number

VIP_MIPI_ID0_CTRL0

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x0080)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:10	RW	0x00	sw_dt_id0 Data type for id0
9:8	RW	0x0	sw_vc_id0 Virtual channel for id0
7:6	RO	0x0	reserved
5	RW	0x0	sw_crop_en_id0 Enable to crop for id0 1'b0 : Disable 1'b1 : Enable
4	RW	0x0	sw_command_mode_en_id0 Select command mode for id0 1'b0 : Not command mode 1'b1 : Command mode
3:1	RW	0x0	sw_wrddr_type_id0 The type of id0 3'b000 : For raw8(occupy 8bit) 3'b001 : For raw10(occupy 16bit) 3'b010 : For raw12(occupy 16bit) 3'b011 : For rgb888(occupy 8bit) 3'b100 : For yuv422(occupy 8bit,then y and uv will write to different address)
0	RW	0x0	sw_cap_en_id0 Enable to capture id0 1'b0 : Disable 1'b1 : Enable

VIP_MIPI_ID0_CTRL1

Address: Operational Base + offset (0x0084)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:16	RW	0x0000	sw_height_id0 Height for id0
15:14	RO	0x0	reserved
13:0	RW	0x0000	sw_width_id0 Width for id0，if sw_wrddr_type is rgb888,then the width is equal to the number of bytes(not pixel) If sw_crop_en_id0 is enable the width value must be 8 aligned when sw_wrddr_type is raw8/yuv422, must be 4 aligned when sw_wrddr_type is raw10/raw12,and must be 24 aligned when sw_wrddy_type is rgb888

VIP_MIPI_ID1_CTRL0

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x0088)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:10	RW	0x00	sw_dt_id1 Data type for id1
9:8	RW	0x0	sw_vc_id1 Virtual channel for id1
7:6	RO	0x0	reserved
5	RW	0x0	sw_crop_en_id1 Enable to crop for id1 1'b0 : Disable 1'b1 : Enable
4	RW	0x0	sw_command_mode_en_id1 Select command mode for id1 1'b0 : Not command mode 1'b1 : Command mode
3:1	RW	0x0	sw_wrddr_type_id1 The type of id1 3'b000 : For raw8(occupy 8bit) 3'b001 : For raw10(occupy 16bit) 3'b010 : For raw12(occupy 16bit) 3'b011 : For rgb888(occupy 8bit) 3'b100 : For yuv422(occupy 8bit,then y and uv will write to different address)
0	RW	0x0	sw_cap_en_id1 Enable to capture id1 1'b0 : Disable 1'b1 : Enable

VIP_MIPI_ID1_CTRL1

Address: Operational Base + offset (0x008c)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:16	RW	0x0000	sw_height_id1 Height for id1
15:14	RO	0x0	reserved
13:0	RW	0x0000	sw_width_id1 Width for id1，if sw_wrddr_type is rgb888,then the width is equal to the number of bytes (not pixel) If sw_crop_en_id1 is enable the width value must be 8 aligned when sw_wrddr_type is raw8/yuv422, must be 4 aligned when sw_wrddr_type is raw10/raw12,and must be 24 aligned when sw_wrddy_type is rgb888

VIP_MIPI_ID2_CTRL0

Address: Operational Base + offset (0x0090)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:10	RW	0x00	sw_dt_id2 Data type for id2
9:8	RW	0x0	sw_vc_id2 Virtual channel for id2
7:6	RO	0x0	reserved
5	RW	0x0	sw_crop_en_id2 Enable to crop for id2 1'b0 : Disable 1'b1 : Enable
4	RW	0x0	sw_command_mode_en_id2 Select command mode for id2 1'b0 : Not command mode 1'b1 : Command mode
3:1	RW	0x0	sw_wrddr_type_id2 The type of id2 3'b000 : For raw8(occupy 8bit) 3'b001 : For raw10(occupy 16bit) 3'b010 : For raw12(occupy 16bit) 3'b011 : For rgb888(occupy 8bit) 3'b100 : For yuv422(occupy 8bit,then y and uv will write to different address)
0	RW	0x0	sw_cap_en_id2 Enable to capture id2 1'b0 : Disable 1'b1 : Enable

VIP_MIPI_ID2_CTRL1

Address: Operational Base + offset (0x0094)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:16	RW	0x0000	sw_height_id2 Height for id2
15:14	RO	0x0	reserved
13:0	RW	0x0000	sw_width_id2 Width for id2，if sw_wrddr_type is rgb888,then the width is equal to the number of bytes(not pixel) If sw_crop_en_id2 is enable the width value must be 8 aligned when sw_wrddr_type is raw8/yuv422, must be 4 aligned when sw_wrddr_type is raw10/raw12,and must be 24 aligned when sw_wrddy_type is rgb888

VIP_MIPI_ID3_CTRL0

Address: Operational Base + offset (0x0098)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:10	RW	0x00	sw_dt_id3 Data type for id3
9:8	RW	0x0	sw_vc_id3 Virtual channel for id3
7:6	RO	0x0	reserved
5	RW	0x0	sw_crop_en_id3 Enable to crop for id3 1'b0 : Disable 1'b1 : Enable
4	RW	0x0	sw_command_mode_en_id3 Select command mode for id3 1'b0 : Not command mode 1'b1 : Command mode
3:1	RW	0x0	sw_wrddr_type_id3 The type of id3 3'b000 : For raw8(occupy 8bit) 3'b001 : For raw10(occupy 16bit) 3'b010 : For raw12(occupy 16bit) 3'b011 : For rgb888(occupy 8bit) 3'b100 : For yuv422(occupy 8bit,then y and uv will write to different address)
0	RW	0x0	sw_cap_en_id3 Enable to capture id3 1'b0 : Disable 1'b1 : Enable

VIP_MIPI_ID3_CTRL1

Address: Operational Base + offset (0x009c)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:16	RW	0x0000	sw_height_id3 Height for id3
15:14	RO	0x0	reserved
13:0	RW	0x0000	sw_width_id3 Width for id3，if sw_wrddr_type is rgb888,then the width is equal to the number of bytes(not pixel) If sw_crop_en_id3 is enable the width value must be 8 aligned when sw_wrddr_type is raw8/yuv422, must be 4 aligned when sw_wrddr_type is raw10/raw12,and must be 24 aligned when sw_wrddy_type is rgb888

VIP_MIPI_WATER_LINE

Address: Operational Base + offset (0x00a0)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24	RW	0x0	sw_dma_idle 1'b0 : Not idle 1'b1 : Idle MIPI path dma transport
23:19	RO	0x0	reserved
18:17	RW	0x0	sw_hurry_value hurry value
16	RW	0x0	sw_hurry_en 1'b0 : disable hurry 1'b1 : enable hurry
15:5	RO	0x0	reserved
4	RW	0x1	sw_water_line_en 1'b0: disable water line 1'b1: enable water line
3:2	RO	0x0	reserved
1:0	RW	0x0	sw_water_line 2'b00 : 75% 2'b01 : 50% 2'b10 : 25% 2'b11 : 0%

VIP_MIPI_FRAME0_ADDR_Y_ID0

Address: Operational Base + offset (0x00a4)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_frame0_addr_y_id0 First address of even frame for ID0 Y/RAW/RGB path(must be aligned to double word)

VIP_MIPI_FRAME1_ADDR_Y_ID0

Address: Operational Base + offset (0x00a8)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_frame1_addr_y_id0 First address of odd frame for ID0 Y path(must be aligned to double word)

VIP_MIPI_FRAME0_ADDR_UV_ID0

Address: Operational Base + offset (0x00ac)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_frame0_addr_uv_id0 First address of even frame for ID0 UV path(must be aligned to double word)

VIP_MIPI_FRAME1_ADDR_UV_ID0

Address: Operational Base + offset (0x00b0)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_frame1_addr_uv_id0 First address of odd frame for ID0 UV path(must be aligned to double word)

VIP_MIPI_FRAME0_VLW_Y_ID0

Address: Operational Base + offset (0x00b4)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x00000	sw_frame0_vlw_y_id0 Virtual line width of even frame for ID0 Y/RAW/RGB path(must be aligned to double word)

VIP_MIPI_FRAME1_VLW_Y_ID0

Address: Operational Base + offset (0x00b8)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x00000	sw_frame1_vlw_y_id0 Virtual line width of odd frame for ID0 Y/RAW/RGB path(must be aligned to double word)

VIP_MIPI_FRAME0_VLW_UV_ID0

Address: Operational Base + offset (0x00bc)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x00000	sw_frame0_vlw_uv_id0 Virtual line width of even frame for ID0 UV path(must be aligned to double word)

VIP_MIPI_FRAME1_VLW_UV_ID0

Address: Operational Base + offset (0x00c0)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x00000	sw_frame1_vlw_uv_id0 Virtual line width of odd frame for ID0 UV path(must be aligned to double word)

VIP_MIPI_FRAME0_ADDR_Y_ID1

Address: Operational Base + offset (0x00c4)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_frame0_addr_y_id1 First address of even frame for ID1 Y/RAW/RGB path(must be aligned to double word)

VIP_MIPI_FRAME1_ADDR_Y_ID1

Address: Operational Base + offset (0x00c8)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_frame1_addr_y_id1 First address of odd frame for ID1 Y/RAW/RGB path(must be aligned to double word)

VIP_MIPI_FRAME0_ADDR_UV_ID1

Address: Operational Base + offset (0x00cc)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_frame0_addr_uv_id1 First address of even frame for ID1 UV path(must be aligned to double word)

VIP_MIPI_FRAME1_ADDR_UV_ID1

Address: Operational Base + offset (0x00d0)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_frame1_addr_uv_id1 First address of odd frame for ID1 UV path(must be aligned to double word)

VIP_MIPI_FRAME0_VLW_Y_ID1

Address: Operational Base + offset (0x00d4)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x00000	sw_frame0_vlw_y_id1 Virtual line width of even frame for ID1 Y/RAW/RGB path(must be aligned to double word)

VIP_MIPI_FRAME1_VLW_Y_ID1

Address: Operational Base + offset (0x00d8)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x00000	sw_frame1_vlw_y_id1 Virtual line width of odd frame for ID1 Y path(must be aligned to double word)

VIP_MIPI_FRAME0_VLW_UV_ID1

Address: Operational Base + offset (0x00dc)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x00000	sw_frame0_vlw_uv_id1 Virtual line width of even frame for ID1 UV path(must be aligned to double word)

VIP_MIPI_FRAME1_VLW_UV_ID1

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x00e0)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x00000	sw_frame1_vlw_uv_id1 Virtual line width of odd frame for ID1 UV path(must be aligned to double word)

VIP_MIPI_FRAME0_ADDR_Y_ID2

Address: Operational Base + offset (0x00e4)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_frame0_addr_y_id2 First address of even frame for ID2 Y/RAW/RGB path(must be aligned to double word)

VIP_MIPI_FRAME1_ADDR_Y_ID2

Address: Operational Base + offset (0x00e8)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_frame1_addr_y_id2 First address of odd frame for ID2 Y/RAW/RGB path(must be aligned to double word)

VIP_MIPI_FRAME0_ADDR_UV_ID2

Address: Operational Base + offset (0x00ec)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_frame0_addr_uv_id2 First address of even frame for ID2 UV path(must be aligned to double word)

VIP_MIPI_FRAME1_ADDR_UV_ID2

Address: Operational Base + offset (0x00f0)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_frame1_addr_uv_id0 First address of odd frame for ID2 UV path(must be aligned to double word)

VIP_MIPI_FRAME0_VLW_Y_ID2

Address: Operational Base + offset (0x00f4)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x00000	sw_frame0_vlw_y_id2 Virtual line width of even frame for ID2 Y/RAW/RGB path(must be aligned to double word)

VIP_MIPI_FRAME1_VLW_Y_ID2

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x00f8)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x00000	sw_frame1_vlw_y_id2 Virtual line width of odd frame for ID2 Y/RAW/RGB path(must be aligned to double word)

VIP_MIPI_FRAME0_VLW_UV_ID2

Address: Operational Base + offset (0x00fc)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x00000	sw_frame0_vlw_uv_id2 Virtual line width of even frame for ID2 UV path(must be aligned to double word)

VIP_MIPI_FRAME1_VLW_UV_ID2

Address: Operational Base + offset (0x0100)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x00000	sw_frame1_vlw_uv_id2 Virtual line width of odd frame for ID2 UV path(must be aligned to double word)

VIP_MIPI_FRAME0_ADDR_Y_ID3

Address: Operational Base + offset (0x0104)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_frame0_addr_y_id3 First address of even frame for ID3 Y/RAW/RGB path(must be aligned to double word)

VIP_MIPI_FRAME1_ADDR_Y_ID3

Address: Operational Base + offset (0x0108)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_frame1_addr_y_id3 First address of odd frame for ID3 Y/RAW/RGB path(must be aligned to double word)

VIP_MIPI_FRAME0_ADDR_UV_ID3

Address: Operational Base + offset (0x010c)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_frame0_addr_uv_id3 First address of even frame for ID3 UV path(must be aligned to double word)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

VIP_MIPI_FRAME1_ADDR_UV_ID3

Address: Operational Base + offset (0x0110)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_frame1_addr_uv_id3 First address of odd frame for ID3 UV path(must be aligned to double word)

VIP_MIPI_FRAME0_VLW_Y_ID3

Address: Operational Base + offset (0x0114)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x00000	sw_frame0_vlw_y_id3 Virtual line width of even frame for ID3 Y/RAW/RGB path(must be aligned to double word)

VIP_MIPI_FRAME1_VLW_Y_ID3

Address: Operational Base + offset (0x0118)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x00000	sw_frame1_vlw_y_id0 Virtual line width of odd frame for ID3 Y/RAW/RGB path(must be aligned to double word)

VIP_MIPI_FRAME0_VLW_UV_ID3

Address: Operational Base + offset (0x011c)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x00000	sw_frame0_vlw_uv_id3 Virtual line width of even frame for ID3 UV path(must be aligned to double word)

VIP_MIPI_FRAME1_VLW_UV_ID3

Address: Operational Base + offset (0x0120)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x00000	sw_frame1_vlw_uv_id3 Virtual line width of odd frame for ID3 UV path(must be aligned to double word)

VIP_MIPI_INTEN

Address: Operational Base + offset (0x0124)

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24	RW	0x0	line_id3_inten 1'b0 : disable 1'b1 : enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
23	RW	0x0	line_id2_inten 1'b0 : disable 1'b1 : enable
22	RW	0x0	line_id1_inten 1'b0 : disable 1'b1 : enable
21	RW	0x0	line_id0_inten 1'b0 : disable 1'b1 : enable
20	RW	0x0	csi2rx_fifo_overflow_inten 1'b0 : disable 1'b1 : enable
19	RW	0x0	bandwidth_lack_inten 1'b0 : disable 1'b1 : enable
18	RW	0x0	config_fifo_overflow_inten 1'b0 : disable 1'b1 : enable
17	RW	0x0	dma_uv_fifo_overflow_inten 1'b0 : disable 1'b1 : enable
16	RW	0x0	dma_y_fifo_overflow_inten Enable the interrupt of dma fifo overflow of y path 1'b0 : Disable 1'b1 : Enable
15	RW	0x0	frame1_dma_end_id3_inten Enable the interrupt of end of odd frame for ID1 1'b0 : Disable 1'b1 : Enable
14	RW	0x0	frame0_dma_end_id3_inten Enable the interrupt of end of even frame for ID3 1'b0 : Disable 1'b1 : Enable
13	RW	0x0	frame1_dma_end_id2_inten Enable the interrupt of end of odd frame for ID1 1'b0 : Disable 1'b1 : Enable
12	RW	0x0	frame0_dma_end_id2_inten Enable the interrupt of end of even frame for ID2 1'b0 : Disable 1'b1 : Enable
11	RW	0x0	frame1_dma_end_id1_inten Enable the interrupt of end of odd frame for ID1 1'b0 : Disable 1'b1 : Enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
10	RW	0x0	frame0_dma_end_id1_inten Enable the interrupt of end of even frame for ID1 1'b0 : Disable 1'b1 : Enable
9	RW	0x0	frame1_dma_end_id0_inten Enable the interrupt of end of odd frame for ID0 1'b0 : Disable 1'b1 : Enable
8	RW	0x0	frame0_dma_end_id0_inten Enable the interrupt of end of even frame for ID0 1'b0 : Disable 1'b1 : Enable
7	RW	0x0	frame1_start_id3_inten Enable the interrupt of start of odd frame for ID3 1'b0 : Disable 1'b1 : Enable
6	RW	0x0	frame0_start_id3_inten Enable the interrupt of start of even frame for ID3 1'b0 : Disable 1'b1 : Enable
5	RW	0x0	frame1_start_id2_inten Enable the interrupt of start of odd frame for ID2 1'b0 : Disable 1'b1 : Enable
4	RW	0x0	frame0_start_id2_inten Enable the interrupt of start of even frame for ID2 1'b0 : Disable 1'b1 : Enable
3	RW	0x0	frame1_start_id1_inten Enable the interrupt of start of odd frame for ID1 1'b0 : Disable 1'b1 : Enable
2	RW	0x0	frame0_start_id1_inten Enable the interrupt of start of even frame for ID1 1'b0 : Disable 1'b1 : Enable
1	RW	0x0	frame1_start_id0_inten Enable the interrupt of start of odd frame for ID0 1'b0 : Disable 1'b1 : Enable
0	RW	0x0	frame0_start_id0_inten Enable the interrupt of start of even frame for ID0 1'b0 : Disable 1'b1 : Enable

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

VIP_MIPI_INTSTAT

Address: Operational Base + offset (0x0128)

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24	RW	0x0	line_id3_intst 0 : No interrupt 1 : Interrupt
23	RW	0x0	line_id2_intst 0 : No interrupt 1 : Interrupt
22	RW	0x0	line_id1_intst 0 : No interrupt 1 : Interrupt
21	RW	0x0	line_id0_intst 0 : No interrupt 1 : Interrupt
20	W1C	0x0	csi2rx_fifo_overflow_intst 0 : No interrupt 1 : Interrupt
19	W1C	0x0	bandwidth_lack_intst 0 : No interrupt 1 : Interrupt
18	W1C	0x0	config_fifo_overflow_intst 0 : No interrupt 1 : Interrupt
17	W1C	0x0	dma_uv_fifo_overflow_intst 0 : No interrupt 1 : Interrupt
16	W1C	0x0	dma_y_fifo_overflow_intst 0 : No interrupt 1 : Interrupt
15	W1C	0x0	frame1_dma_end_id3_intst 0 : No interrupt 1 : Interrupt
14	W1C	0x0	frame0_dma_end_id3_intst 0 : No interrupt 1 : Interrupt
13	W1C	0x0	frame1_dma_end_id2_intst 0 : No interrupt 1 : Interrupt
12	W1C	0x0	frame0_dma_end_id2_intst 0 : No interrupt 1 : Interrupt

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
11	W1C	0x0	frame1_dma_end_id1_intst 0 : No interrupt 1 : Interrupt
10	W1C	0x0	frame0_dma_end_id1_intst 0 : No interrupt 1 : Interrupt
9	W1C	0x0	frame1_dma_end_id0_intst 0 : No interrupt 1 : Interrupt
8	W1C	0x0	frame0_dma_end_id0_intst 0 : No interrupt 1 : Interrupt
7	W1C	0x0	frame1_start_id3_intst 0 : No interrupt 1 : Interrupt
6	W1C	0x0	frame0_start_id3_intst 0 : No interrupt 1 : Interrupt
5	W1C	0x0	frame1_start_id2_intst 0 : No interrupt 1 : Interrupt
4	W1C	0x0	frame0_start_id2_intst 0 : No interrupt 1 : Interrupt
3	W1C	0x0	frame1_start_id1_intst 0 : No interrupt 1 : Interrupt
2	W1C	0x0	frame0_start_id1_intst 0 : No interrupt 1 : Interrupt
1	W1C	0x0	frame1_start_id0_intst 0 : No interrupt 1 : Interrupt
0	W1C	0x0	frame0_start_id0_intst 0 : No interrupt 1 : Interrupt

VIP_MIPI_LINE_INT_NUM_ID0_1

Address: Operational Base + offset (0x012c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:16	RW	0x0040	line_int_num_id1 if line_int_num_id1=100,then channel 1 receive 100th line蔽th line蔔th line.....the line_id1_intst will be 1
15:14	RO	0x0	reserved
13:0	RW	0x0040	line_int_num_id0 if line_int_num_id0=100,then channel 0 receive 100th line蔽th line蔔th line.....the line_id0_intst will be 1

VIP_MIPI_LINE_INT_NUM_ID2_3

Address: Operational Base + offset (0x0130)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:16	RW	0x0040	line_int_num_id3 if line_int_num_id3=100,then channel 3 receive 100th line蔽th line蔔th line.....the line_id3_intst will be 1
15:14	RO	0x0	reserved
13:0	RW	0x0040	line_int_num_id2 if line_int_num_id2=100,then channel 2 receive 100th line蔽th line蔔th line.....the line_id2_intst will be 1

VIP_MIPI_LINE_CNT_ID0_1

Address: Operational Base + offset (0x0134)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:16	RO	0x0000	line_cnt_id1 current line count for id1
15:14	RO	0x0	reserved
13:0	RO	0x0000	line_cnt_id0 current line count for id0

VIP_MIPI_LINE_CNT_ID2_3

Address: Operational Base + offset (0x0138)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:16	RO	0x0000	line_cnt_id3 current line count for id3
15:14	RO	0x0	reserved
13:0	RO	0x0000	line_cnt_id2 current line count for id2

VIP_MIPI_ID0_CROP_START

Address: Operational Base + offset (0x013c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:16	RW	0x0000	sw_start_y_id0 The start y coordinate for id0
15:14	RO	0x0	reserved
13:0	RW	0x0000	sw_start_x_id0 The start x coordinate for id0，if sw_wrddr_type is rgb888,then the start x is measured in byte. The start x value must be 8 aligned when sw_wrddr_type is raw8/yuv422, must be 4 aligned when sw_wrddr_type is raw10/raw12,and must be 24 aligned when sw_wrddy_type is rgb888

VIP_MIPI_ID1_CROP_START

Address: Operational Base + offset (0x0140)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:16	RW	0x0000	sw_start_y_id1 The start y coordinate for id1
15:14	RO	0x0	reserved
13:0	RW	0x0000	sw_start_x_id1 the start x coordinate for id1，if sw_wrddr_type is rgb888,then the start x is measured in byte The start x value must be 8 aligned when sw_wrddr_type is raw8/yuv422, must be 4 aligned when sw_wrddr_type is raw10/raw12,and must be 24 aligned when sw_wrddy_type is rgb888

VIP_MIPI_ID2_CROP_START

Address: Operational Base + offset (0x0144)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:16	RW	0x0000	sw_start_y_id2 The start y coordinate for id2
15:14	RO	0x0	reserved
13:0	RW	0x0000	sw_start_x_id2 the start x coordinate for id2，if sw_wrddr_type is rgb888,then the start x is measured in byte The start x value must be 8 aligned when sw_wrddr_type is raw8/yuv422, must be 4 aligned when sw_wrddr_type is raw10/raw12,and must be 24 aligned when sw_wrddy_type is rgb888

VIP_MIPI_ID3_CROP_START

Address: Operational Base + offset (0x0148)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:16	RW	0x0000	sw_start_y_id3 The start y coordinate for id3
15:14	RO	0x0	reserved
13:0	RW	0x0000	sw_start_x_id3 the start x coordinate for id3，if sw_wrddr_type is rgb888,then the start x is measured in byte The start x value must be 8 aligned when sw_wrddr_type is raw8/yuv422, must be 4 aligned when sw_wrddr_type is raw10/raw12,and must be 24 aligned when sw_wrddy_type is rgb888

VIP_MMU_DTE_ADDR

Address: Operational Base + offset (0x0800)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	mmu_dte_addr MMU current page Table address

VIP_MMU_STATUS

Address: Operational Base + offset (0x0804)

Bit	Attr	Reset Value	Description
31:11	RO	0x0	reserved
10:6	RO	0x00	page_fault_bus_id Index of master reponsible for last page fault
5	RO	0x0	page_fault_is_write The direction of access for last page fault: 0 = Read 1 = Write
4	RO	0x0	replay_buffer_empty The MMU replay buffer is empty
3	RO	0x0	mmu_idle The MMU is idle when accesses are being translated and there are no unfinished translated accesses
2	RO	0x0	stail_active MMU stall mode currently enabled. The mode is enabled by command
1	RO	0x0	page_fault_active MMU page fault mode currently enabled . The mode is enabled by command
0	RO	0x0	paging_enabled Paging is enabled

VIP_MMU_COMMAND

Address: Operational Base + offset (0x0808)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved
2:0	WO	0x0	mmu_cmd MMU_CMD. This can be: 3'b000: MMU_ENABLE_PAGING 3'b001: MMU_DISABLE_PAGING 3'b010: MMU_ENABLE_STALL 3'b011: MMU_DISABLE_STALL 3'b100: MMU_ZAP_CACHE 3'b101: MMU_PAGE_FAULT_DONE

VIP_MMU_PAGE_FAULT_ADDR

Address: Operational Base + offset (0x080c)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	page_fault_addr address of last page fault

VIP_MMU_ZAP_ONE_LINE

Address: Operational Base + offset (0x0810)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	mmu_zap_one_line address to be invalidated from the page table cache

VIP_MMU_INT_RAWSTAT

Address: Operational Base + offset (0x0814)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x0	read_bus_error read bus error
0	RW	0x0	page_fault page fault

VIP_MMU_INT_CLEAR

Address: Operational Base + offset (0x0818)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	WO	0x0	read_bus_error read bus error
0	WO	0x0	page_fault page fault

VIP_MMU_INT_MASK

Address: Operational Base + offset (0x081c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x0	read_bus_error read bus error
0	RW	0x0	page_fault page fault

VIP_MMU_INT_STATUS

Address: Operational Base + offset (0x0820)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RO	0x0	read_bus_error read bus error
0	RO	0x0	page_fault page fault

VIP_MMU_AUTO_GATING

Address: Operational Base + offset (0x0824)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	mmu_auto_gating when it is 1'b1, the mmu will auto gating it self

16.5 Interface Description

Table 16-1 VIP Interface Description

Module Pin	Direction	Pad Name	IOMUX Setting
vip_clkout	O	IO_CIFclkoutm0_RGMIIclk_GPIO2B7vccio2	GRF_GPIO2B_IOMUX_H[15:12]=4'h1
vip_clkin	I	IO_CIFclkinm0_RGMIIrxd3_GPIO2B6vccio2	GRF_GPIO2B_IOMUX_H[11:8]=4'h1
vip_href	I	IO_CIFhrefm0_RGMIIrxd2_GPIO2B5vccio2	GRF_GPIO2B_IOMUX_H[7:4]=4'h1
vip_vsync	I	IO_CIFvsyncm0_RGMIItxd2_GPIO2B4vccio2	GRF_GPIO2B_IOMUX_H[3:0]=4'h1
vip_data0	I	IO_CIFd0m0_CLKout_ethernet_GPIO2C1vccio2	GRF_GPIO2C_IOMUX_L[3:0]=4'h1
vip_data1	I	IO_CIFd1m0_RGMIItxclk_GPIO2C1vccio2	GRF_GPIO2C_IOMUX_L[7:4]=4'h1
vip_data2	I	IO_CIFd2m0_RGMIIrxd0_SPI2M1miso_GPIO2A4vccio2	GRF_GPIO2A_IOMUX_H[3:0]=4'h1
vip_data3	I	IO_CIFd3m0_RGMIIrxd1_SPI2M1clk_GPIO2A5vccio2	GRF_GPIO2A_IOMUX_H[7:4]=4'h1
vip_data4	I	IO_CIFd4m0_RGMIIrxer_SPI2M1mosi_GPIO2A6vccio2	GRF_GPIO2A_IOMUX_H[11:8]=4'h1
vip_data5	I	IO_CIFd5m0_RGMIIrxdv_SPI2M1csn_GPIO2A7vcci	GRF_GPIO2A_IOMUX_H[15:12]=4'h1

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Module Pin	Direction	Pad Name	IOMUX Setting
		o2	
vip_data6	I	IO_CIFd6m0_RGMIImdio_GPIO2B0vccio2	GRF_GPIO2B_IOMUX_L[3:0]=4'h1
vip_data7	I	IO_CIFd7m0_RGMIIcol_GPIO2B1vccio2	GRF_GPIO2B_IOMUX_L[7:4]=4'h1
vip_data8	I	IO_CIFd8m0_RGMIImdc_LCDChsyncm0_GPIO2B2vccio2	GRF_GPIO2B_IOMUX_L[11:8]=4'h1
vip_data9	I	IO_CIFd9m0_RGMIItxd3_LCDCvsyncm0_GPIO2B3vccio2	GRF_GPIO2B_IOMUX_L[15:12]=4'h1
vip_data10	I	IO_CIFd10m0_RGMIIrxclk_LCDCd2m0_GPIO2C2vccio2	GRF_GPIO2C_IOMUX_L[11:8]=4'h1
vip_data11	I	IO_CIFd11m0_LCDCd3m0_GPIO2C3vccio2	GRF_GPIO2C_IOMUX_L[15:12]=4'h1
vip_data12	I	IO_CIFd12m0_RGMIIcrs_LCDCd6m0_GPIO2A0vccio2	GRF_GPIO2A_IOMUX_L[3:0]=4'h1
vip_data13	I	IO_CIFd13m0_RGMIItxen_LCDCd7m0_GPIO2A1vccio2	GRF_GPIO2A_IOMUX_L[7:4]=4'h1
vip_data14	I	IO_CIFd14m0_RGMIItxd1_LCDCd0m0_GPIO2A2vccio2	GRF_GPIO2A_IOMUX_L[11:8]=4'h1
vip_data15	I	IO_CIFd15m0_RGMIItxd0_LCDCd1m0_GPIO2A3vccio2	GRF_GPIO2A_IOMUX_L[15:12]=4'h1

16.6 Application Notes

16.6.1 DVP Application

The most important configuration requirement of all operations is the DVP_CAP_EN bit must be set after all the mode selection is ready.The configuration order of the input/output data format, YUV order, the address,frame size/width, AXI burst length and other options do not need to care.

There are many debug registers to make it easy to read the internal operation information of VIP. The valid pixel number of scale result in FIFO can be known by read VIP_DVP_FIFO_ENTRY.The line number of last frame and the pixel number of last line can be also known by read the VIP_DVP_LAST_LINE and VIP_DVP_LAST_PIX.

If bt1120 is received,the bus_grf_vi_con1[9] must be configured as 1.

16.6.2 MIPI Application

VIP support receiving up to four MIPI IDs at the same time.VIP could receive all ID value via configuring the sw_vc_id0/1/2/3 and sw_dt_id0/1/2/3,but VIP could only transform MIPI data to RAW8/10/12,RGB888,YUV422 via configuring the sw_wrddr_type. The register sw_height must be same as the received image,and the register sw_width is useless when sw_crop_en is disable.If sw_wrddr_type is rgb888,then the sw_width and sw_start_x are equal to the number of bytes.

If AXI bus is too busy to transfer sensor data,you can enable sw_hurry_en and sw_water_line_en via configuring MIPI_WATER_LINE

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 17 GMAC Ethernet Interface

17.1 Overview

The GMAC Ethernet Controller provides a complete Ethernet interface from processor to a Reduced Media Independent Interface (RMII) and Reduced Gigabit Media Independent Interface (RGMII) compliant Ethernet PHY.
The GMAC includes a DMA controller. The DMA controller efficiently moves packet data from microprocessor's RAM, formats the data for an IEEE 802.3-2002 compliant packet and transmits the data to an Ethernet Physical Interface (PHY). It also efficiently moves packet data from RXFIFO to microprocessor's RAM.

17.1.1 Feature

- Supports 10/100/1000-Mbps data transfer rates with the RGMII interfaces
- Supports 10/100-Mbps data transfer rates with the RMII interfaces
- Supports both full-duplex and half-duplex operation
  - Supports CSMA/CD Protocol for half-duplex operation
  - Supports packet bursting and frame extension in 1000 Mbps half-duplex operation
  - Supports IEEE 802.3x flow control for full-duplex operation
  - Optional forwarding of received pause control frames to the user application in full-duplex operation
  - Back-pressure support for half-duplex operation
  - Automatic transmission of zero-quanta pause frame on de-assertion of flow control input in full-duplex operation
- Preamble and start-of-frame data (SFD) insertion in Transmit, and deletion in Receive paths
- Automatic CRC and pad generation controllable on a per-frame basis
- Options for Automatic Pad/CRC Stripping on receive frames
- Programmable frame length to support Standard Ethernet frames
- Programmable InterFrameGap (40-96 bit times in steps of 8)
- Supports a variety of flexible address filtering modes:
  - 64-bit Hash filter (optional) for multicast and uni-cast (DA) addresses
  - Option to pass all multicast addressed frames
  - Promiscuous mode support to pass all frames without any filtering for network monitoring
  - Passes all incoming packets (as per filter) with a status report
- Separate 32-bit status returned for transmission and reception packets
- Supports IEEE 802.1Q VLAN tag detection for reception frames
- MDIO Master interface for PHY device configuration and management
- Support detection of LAN wake-up frames and AMD Magic Packet frames
- Support checksum off-load for received IPv4 and TCP packets encapsulated by the Ethernet frame
- Support checking IPv4 header checksum and TCP, UDP, or ICMP checksum encapsulated in IPv4 or IPv6 datagrams
- Comprehensive status reporting for normal operation and transfers with errors
- Support per-frame Transmit/Receive complete interrupt control
- Supports 4-KB receive FIFO depths on reception.
- Supports 2-KB FIFO depth on transmission
- Automatic generation of PAUSE frame control or backpressure signal to the GMAC core based on Receive FIFO-fill (threshold configurable) level
- Handles automatic retransmission of Collision frames for transmission
- Discards frames on late collision, excessive collisions, excessive deferral and underrun conditions
- AXI interface to any CPU or memory
- Software can select the type of AXI burst (fixed and variable length burst) in the AXI Master interface

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

- Supports internal loopback on the RGMII/RMII for debugging
- Debug status register that gives status of FSMs in Transmit and Receive data-paths and FIFO fill-levels.

17.2 Block Diagram

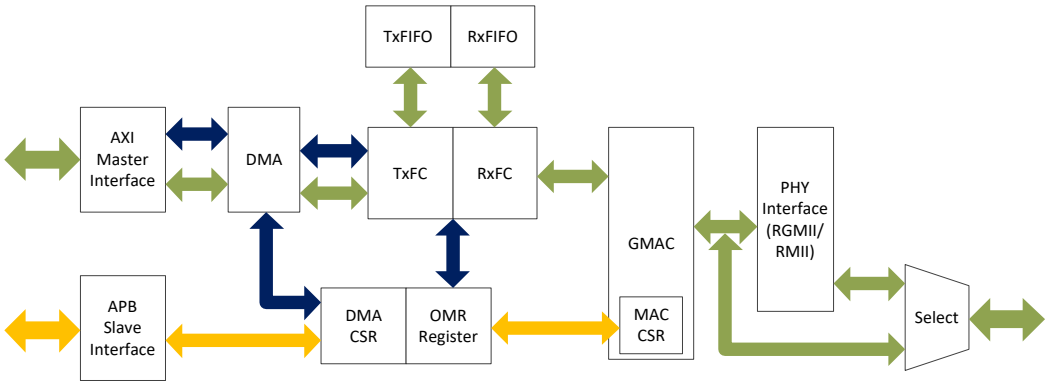


Fig. 17-1 GMACArchitecture

The GMAC is broken up into multiple separate functional units. These blocks are interconnected in the MAC module. The block diagram shows the general flow of data and control signals between these blocks.

The GMAC transfers data to system memory through the AXI master interface. The host CPU uses the APB Slave interface to access the GMAC subsystem's control and status registers (CSRs).

The GMAC supports the PHY interfaces of reduced GMII (RGMII) and reduced MII (RMII).

The Transmit FIFO (Tx FIFO) buffers data read from system memory by the DMA before transmission by the GMAC Core. Similarly, the Receive FIFO (Rx FIFO) stores the Ethernet frames received from the line until they are transferred to system memory by the DMA. These are asynchronous FIFOs, as they also transfer the data between the application clock and the GMAC line clocks.

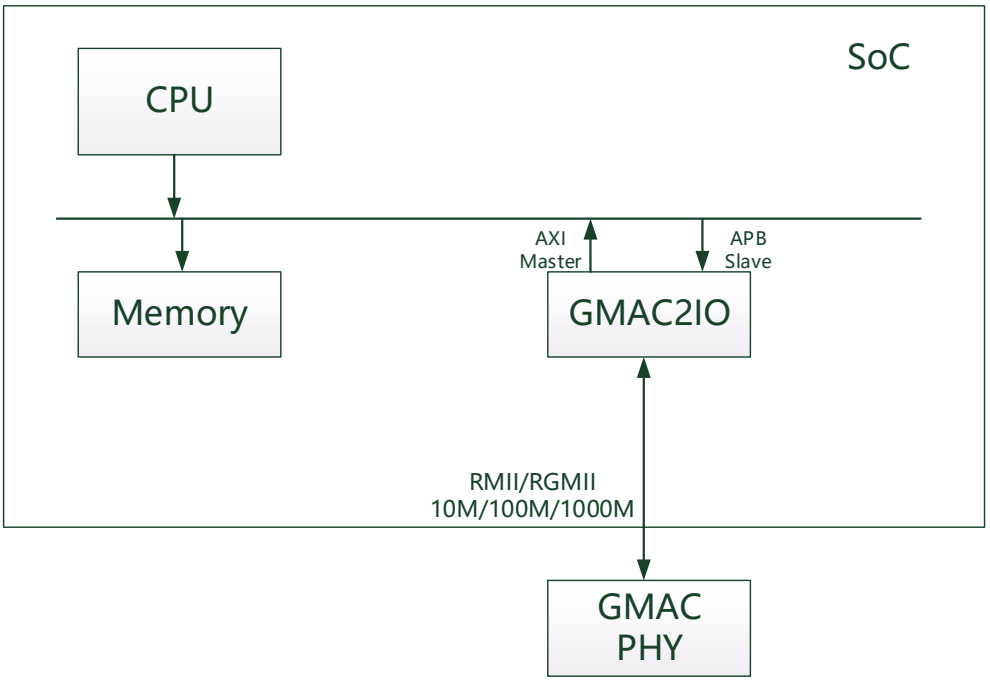


Fig. 17-2 GMAC Architecture

There are two independent GMAC controllers named GMAC2IO and GMAC2PHY:

- GMAC2IO Supports 10/100/1000-Mbps data transfer rates with the RGMII interfaces and Supports 10/100-Mbps data transfer rates with the RMII interfaces

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

17.3 Function Description

17.3.1 Frame Structure

Data frames transmitted shall have the frame format shown in Fig. 1-3.

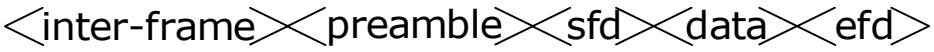


Fig. 17-3 MAC Block Diagram

The preamble <preamble> begins a frame transmission. The bit value of the preamble field consists of 7 octets with the following bit values:
10101010 10101010 10101010 10101010 10101010 10101010 10101010
The SFD (start frame delimiter) <sfd> indicates the start of a frame and follows the preamble.The bit value is 10101011.
The data in a well formed frame shall consist of N octet's data.

17.3.2 RMII Interface timing diagram

The Reduced Media Independent Interface (RMII) specification reduces the pin count between Ethernet PHYs and Switch ASICs (only in 10/100 mode). According to the IEEE 802.3u standard, an MII contains 16 pins for data and control. In devices incorporating multiple MAC or PHY interfaces (such as switches), the number of pins adds significant cost with increase in port count. The RMII specification addresses this problem by reducing the pin count to 7 for each port - a 62.5% decrease in pin count.
The RMII module is instantiated between the GMAC and the PHY. This helps translation of the MAC's MII into the RMII. The RMII block has the following characteristics:

- Supports 10-Mbps and 100-Mbps operating rates. It does not support 1000-Mbps operation.
- Two clock references are sourced externally or CRU, providing independent, 2-bit wide transmit and receive paths.

Transmit Bit Ordering

Each nibble from the MII must be transmitted on the RMII a di-bit at a time with the order of di-bit transmission shown in Fig.1-4. The lower order bits (D1 and D0) are transmitted first followed by higher order bits (D2 and D3).

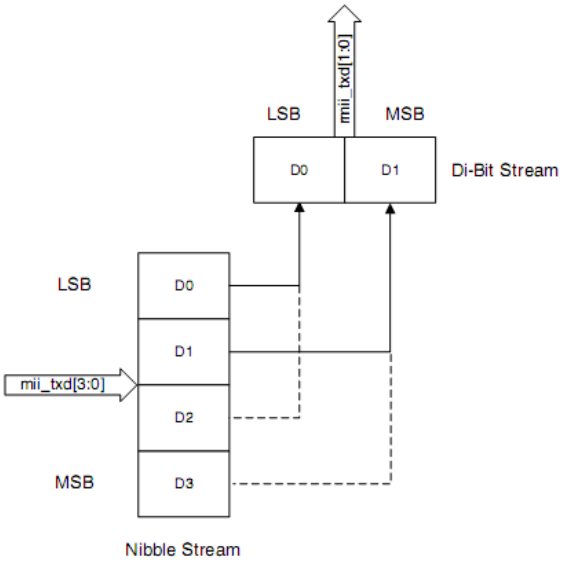


Fig. 17-4 RMII transmission bit ordering

RMII Transmit Timing Diagrams

Fig.1-5 through 1-8 show MII-to-RMII transaction timing.The clk_rmii_i (REF_CLK) frequency is 50MHz in RMII interface.In 10Mb/s mode, as the REF_CLK frequency is 10 times as the data rate, the value on rmii_txd_o[1:0] (TXD[1:0]) shall be valid such that TXD[1:0] may be sampled every 10th cycle,regard-less of the starting cycle within the group and yield the correct frame data.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

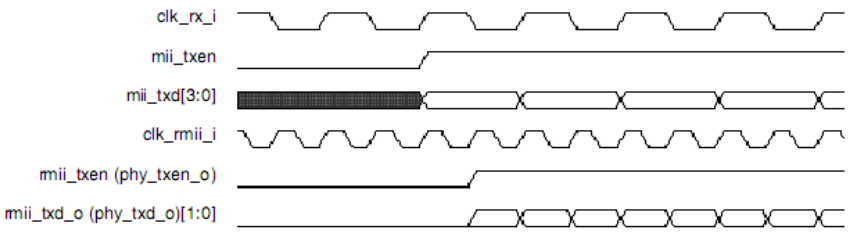


Fig. 17-5 Start of MII and RMII transmission in 100-Mbps mode

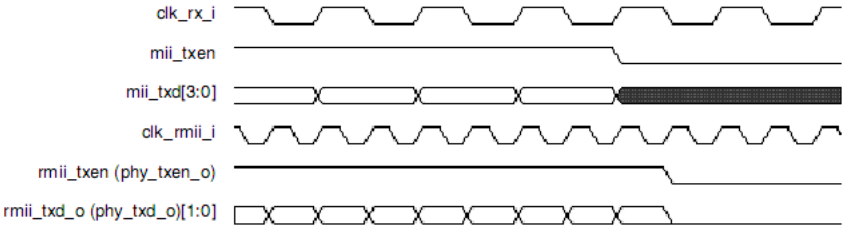


Fig. 17-6 End of MII and RMII Transmission in 100-Mbps Mode

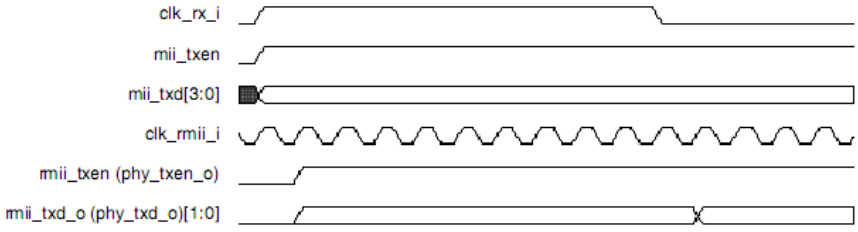


Fig. 17-7 Start of MII and RMII Transmission in 10-Mbps Mode

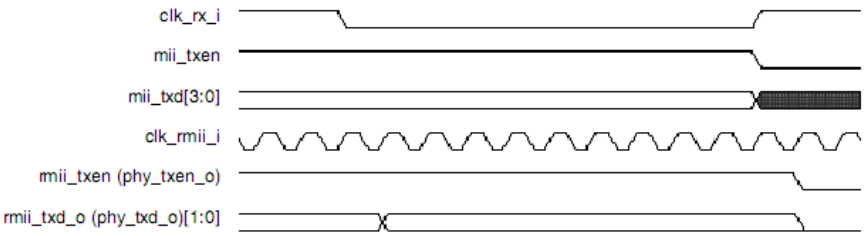


Fig. 17-5 End of MII and RMII Transmission in 10-Mbps Mode

Receive Bit Ordering

Each nibble is transmitted to the MII from the di-bit received from the RMII in the nibble transmission order shown in Fig.1-9. The lower order bits (D0 and D1) are received first, followed by the higher order bits (D2 and D3).

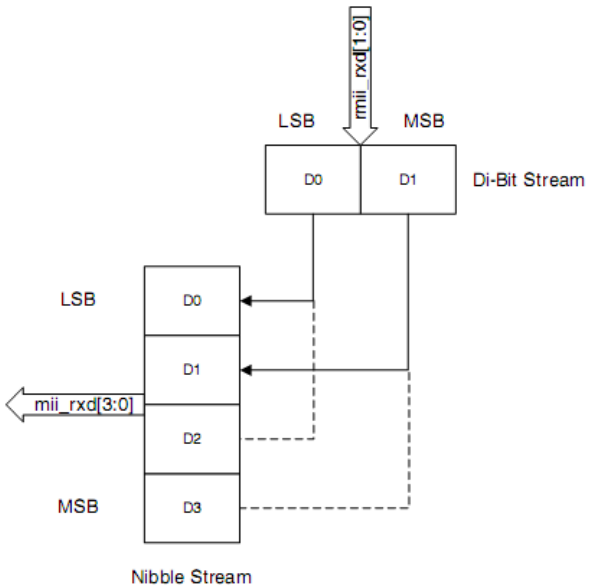


Fig. 17-6 RMII receive bit ordering

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

17.3.3 RGMII interface

The Reduced Gigabit Media Independent Interface (RGMII) specification reduces the pin count of the interconnection between the GMAC 10/100/1000 controller and the PHY for GMII and MII interfaces. To achieve this, the data path and control signals are reduced and multiplexed together with both the edges of the transmission and receive clocks. For gigabit operation the clocks operate at 125 MHz; for 10/100 operation, the clock rates are 2.5 MHz/25 MHz.

In the GMAC 10/100/1000 controller, the RGMII module is instantiated between the GMAC core's GMII and the PHY to translate the control and data signals between the GMII and RGMII protocols.

The RGMII block has the following characteristics:

- Supports 10-Mbps, 100-Mbps, and 1000-Mbps operation rates.
- For the RGMII block, no extra clock is required because both the edges of the incoming clocks are used.
- The RGMII block extracts the in-band (link speed, duplex mode and link status) status signals from the PHY and provides them to the GMAC core logic for link detection.

17.3.4 Management Interface

The MAC management interface provides a simple, two-wire, serial interface to connect the GMAC and a managed PHY, for the purposes of controlling the PHY and gathering status from the PHY. The management interface consists of a pair of signals that transport the management information across the MII bus: MDIO and MDC.

The GMAC initiates the management write/read operation. The clock gmii_mdc_o(MDC) is a divided clock from the application clock pclk_gmac. The divide factor depends on the clock range setting in the GMII address register. Clock range is set as follows:

Selection	pclk_gmac	MDC Clock
0000	60-100 MHz	pclk_gmac/42
0001	100-150 MHz	pclk_gmac/62
0010	20-35 MHz	pclk_gmac/16
0011	35-60 MHz	pclk_gmac/26
0100	150-250 MHz	pclk_gmac/102
0101	250-300 MHz	pclk_gmac/124
0110, 0111	Reserved	

The MDC is the derivative of the application clock pclk_gmac. The management operation is performed through the gmii_mdi_i, gmii_mdo_o and gmii_mdo_o_e signals. A three-state buffer is implemented in the PAD.

The frame structure on the MDIO line is shown below.

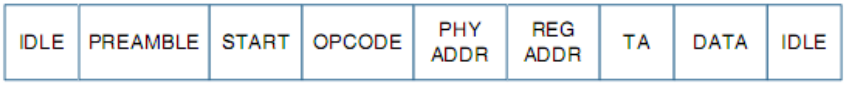


Fig. 17-7 MDIO frame structure

IDLE: The mdio line is three-state; there is no clock on gmii_mdc_o
PREAMBLE: 32 continuous bits of value 1
START: Start-of-frame is 2'b01
OPCODE: 2'b10 for read and 2'b01 for write
PHY ADDR: 5-bit address select for one of 32 PHYs
REG ADDR: Register address in the selected PHY
TA: Turnaround is 2'bZ0 for read and 2'b10 for Write
DATA: Any 16-bit value. In a write operation, the GMAC drives mdio; in a read operation, PHY drives it.

17.3.5 Power Management Block

Power management (PMT) supports the reception of network (remote) wake-up frames and Magic Packet frames. PMT does not perform the clock gate function, but generates interrupts for wake-up frames and Magic Packets received by the GMAC. The PMT block sits on the receiver path of the GMAC and is enabled with remote wake-up frame enable and Magic Packet enable. These enables are in the PMT control and status register and are programmed by the application.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

When the power down mode is enabled in the PMT, then all received frames are dropped by the core and they are not forwarded to the application. The core comes out of the power down mode only when either a Magic Packet or a Remote Wake-up frame is received and the corresponding detection is enabled.

Remote Wake-Up Frame Detection

When the GMAC is in sleep mode and the remote wake-up bit is enabled in register GMAC_PMT_CTRL_STA (0x002C), normal operation is resumed after receiving a remote wake-up frame. The application writes all eight wake-up filter registers, by performing a sequential write to address (0028). The application enables remote wake-up by writing a 1 to bit 2 of the register GMAC_PMT_CTRL_STA.

PMT supports four programmable filters that allow support of different receive frame patterns. If the incoming frame passes the address filtering of Filter Command, and if Filter CRC-16 matches the incoming examined pattern, then the wake-up frame is received. Filter_offset (minimum value 12, which refers to the 13th byte of the frame) determines the offset from which the frame is to be examined. Filter Byte Mask determines which bytes of the frame must be examined. The thirty-first bit of Byte Mask must be set to zero.

The remote wake-up CRC block determines the CRC value that is compared with Filter CRC-16. The wake-up frame is checked only for length error, FCS error, dribble bit error, GMII error, collision, and to ensure that it is not a runt frame. Even if the wake-up frame is more than 512 bytes long, if the frame has a valid CRC value, it is considered valid. Wake-up frame detection is updated in the register GMAC_PMT_CTRL_STA for every remote Wake-up frame received. A PMT interrupt to the application triggers a read to the GMAC_PMT_CTRL_STA register to determine reception of a wake-up frame.

Magic Packet Detection

The Magic Packet frame is based on a method that uses Advanced Micro Device's Magic Packet technology to power up the sleeping device on the network. The GMAC receives a specific packet of information, called a Magic Packet, addressed to the node on the network.

Only Magic Packets that are addressed to the device or a broadcast address will be checked to determine whether they meet the wake-up requirements. Magic Packets that pass the address filtering (unicast or broadcast) will be checked to determine whether they meet the remote Wake-on-LAN data format of 6 bytes of all ones followed by a GMAC Address appearing 16 times.

The application enables Magic Packet wake-up by writing a 1 to Bit 1 of the register GMAC_PMT_CTRL_STA. The PMT block constantly monitors each frame addressed to the node for a specific Magic Packet pattern. Each frame received is checked for a 48'hFF_FF_FF_FF_FF_FF pattern following the destination and source address field. The PMT block then checks the frame for 16 repetitions of the GMAC address without any breaks or interruptions. In case of a break in the 16 repetitions of the address, the 48'hFF_FF_FF_FF_FF_FF pattern is scanned for again in the incoming frame. The 16 repetitions can be anywhere in the frame, but must be preceded by the synchronization stream (48'hFF_FF_FF_FF_FF_FF). The device will also accept a multicast frame, as long as the 16 duplications of the GMAC address are detected.

If the MAC address of a node is 48'h00_11_22_33_44_55, then the GMAC scans for the data sequence:

Destination Address Source Address ……. FF FFFFFFFFF
00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55
00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55
00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55
00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55
…CRC

Magic Packet detection is updated in the PMT Control and Status register for Magic Packet received. A PMT interrupt to the Application triggers a read to the PMT CSR to determine whether a Magic Packet frame has been received.

17.3.6 MAC Management Counters

The counters in the MAC Management Counters (MMC) module can be viewed as an

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

extension of the register address space of the CSR module. The MMC module maintains a set of registers for gathering statistics on the received and transmitted frames. These include a control register for controlling the behavior of the registers, two 32-bit registers containing interrupts generated (receive and transmit), and two 32-bit registers containing masks for the Interrupt register (receive and transmit). These registers are accessible from the Application through the MAC Control Interface (MCI). Non-32-bit accesses are allowed as long as the address is word-aligned.
The organization of these registers is shown in Register Description. The MMCs are accessed using transactions, in the same way the CSR address space is accessed. The Register Description in this chapter describe the various counters and list the address for each of the statistics counters. This address will be used for Read/Write accesses to the desired transmit/receive counter.
The MMC module gathers statistics on encapsulated IPv4, IPv6, TCP, UDP, or ICMP payloads in received Ethernet frames.

17.4 Register Description

17.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
GMAC_MAC_CONF	0x0000	W	0x00000000	MAC Configuration Register This is the operation mode register for the MAC
GMAC_MAC_FRM_FILT	0x0004	W	0x00000000	MAC Frame Filter Contains the frame filtering controls
GMAC_HASH_TAB_HI	0x0008	W	0x00000000	Hash Table High Register Contains the higher 32 bits of the Multicast Hash table. This register is present only when the Hash filter function is selected in core Consultant
GMAC_HASH_TAB_LO	0x000c	W	0x00000000	Hash Table Low Register Contains the lower 32 bits of the Multicast Hash table. This register is present only when the Hash filter function is selected in core Consultant
GMAC_GMII_ADDR	0x0010	W	0x00000000	GMII Address Register Controls the management cycles to an external PHY
GMAC_GMII_DATA	0x0014	W	0x00000000	GMII Data Register Contains the data to be written to or read from the PHY register
GMAC_FLOW_CTRL	0x0018	W	0x00000000	Flow Control Register Controls the generation of control frames
GMAC_VLAN_TAG	0x001c	W	0x00000000	VLAN Tag Register Identifies IEEE 802.1Q VLAN type frames

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
GMAC_DEBUG	0x0024	W	0x00000000	Debug register This debug register gives the status of all the main modules of the transmit and receive data-paths and the FIFOs. An all-zero status indicates that the MAC core is in idle state (and FIFOs are empty) and no activity is going on in the data-paths
GMAC_PMT_CTRL_STA	0x002c	W	0x00000000	PMT Control and Status Register PMT Control and Status
GMAC_INT_STATUS	0x0038	W	0x00000000	Interrupt Status Register Contains the interrupt status
GMAC_INT_MASK	0x003c	W	0x00000000	Interrupt Mask Register Contains the masks for generating the interrupts
GMAC_MAC_ADDR0_HI	0x0040	W	0x0000ffff	MAC Address0 High Register Contains the higher 16 bits of the first MAC address
GMAC_MAC_ADDR0_LO	0x0044	W	0xffffffff	MAC Address0 Low Register Contains the lower 32 bits of the first MAC address
GMAC_AN_CTRL	0x00c0	W	0x00000000	AN Control Register Enables and/or restarts auto-negotiation. It also enables PCS loopback
GMAC_AN_STATUS	0x00c4	W	0x00000008	AN Status Register Indicates the link and auto-negotiation status
GMAC_AN_ADV	0x00c8	W	0x000001e0	Auto Negotiation Advertisement Register This register is configured before auto-negotiation begins. It contains the advertised ability of the GMAC
GMAC_AN_LINK_PART_AB	0x00cc	W	0x00000000	Auto Negotiation Link Partner Ability Register Contains the advertised ability of the link partner. Its value is valid after successful completion of auto-negotiation or when a new base page has been received (indicated in the Auto-Negotiation Expansion Register)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
GMAC_AN_EXP	0x00d0	W	0x00000000	Auto Negotiation Expansion Register Indicates whether a new base page has been received from the link partner
GMAC_INTF_MODE_STA	0x00d8	W	0x00000000	RGMII Status Register Indicates the status signals received from the PHY through the RGMII interface
GMAC_MMC_CTRL	0x0100	W	0x00000000	MMC Control Register The MMC Control register establishes the operating mode of the management counters
GMAC_MMC_RX_INTR	0x0104	W	0x00000000	MMC Receive Interrupt Register The MMC Receive Interrupt register maintains the interrupts generated when the receive statistic counters reach half their maximum values (0x8000_0000), and when they cross their maximum values (0xFFFF_FFFF). When Counter Stop Rollover is set, then interrupts are set but the counter remains at all-ones. The MMC Receive Interrupt register is a 32-bit wide register. An interrupt bit is cleared when the respective MMC counter that caused the interrupt is read. The least significant byte lane (bits[7:0]) of the respective counter must be read in order to clear the interrupt bit

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
GMAC_MMC_TX_INTR	0x0108	W	0x00000000	MMC Transmit Interrupt Register The MMC Transmit Interrupt register maintains the interrupts generated when transmit statistic counters reach half their maximum values (0x8000_0000), and when they cross their maximum values (0xFFFF_FFFF). When Counter Stop Rollover is set, then interrupts are set but the counter remains at all-ones. The MMC Transmit Interrupt register is a 32-bit wide register. An interrupt bit is cleared when the respective MMC counter that caused the interrupt is read. The least significant byte lane (bits[7:0]) of the respective counter must be read in order to clear the interrupt bit
GMAC_MMC_RX_INT_MSK	0x010c	W	0x00000000	MMC Receive Interrupt Mask Register The MMC Receive Interrupt Mask register maintains the masks for the interrupts generated when receive statistic counters reach half their maximum value, and when they reach their maximum values
GMAC_MMC_TX_INT_MSK	0x0110	W	0x00000000	MMC Transmit Interrupt Mask Register The MMC Transmit Interrupt Mask register maintains the masks for the interrupts generated when transmit statistic counters reach half their maximum value, and when they reach their maximum values
GMAC_MMC_TXOCTETCNT_GB	0x0114	W	0x00000000	MMC TX OCTET Good and Bad Counter
GMAC_MMC_TXFRMCNT_GB	0x0118	W	0x00000000	MMC TX OCTET Good and Bad Counter
GMAC_MMC_TXUNDFLWERR	0x0148	W	0x00000000	MMC TX Underflow Error

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
GMAC_MMC_TXCARERR	0x0160	W	0x00000000	MMC TX Carrier Error
GMAC_MMC_TXOCTETCNT_G	0x0164	W	0x00000000	MMC TX OCTET Good Counter
GMAC_MMC_TXFRMCNT_G	0x0168	W	0x00000000	MMC TX Frame Good Counter
GMAC_MMC_RXFRMCNT_GB	0x0180	W	0x00000000	MMC RX Frame Good and Bad Counter
GMAC_MMC_RXOCTETCNT_GB	0x0184	W	0x00000000	MMC RX OCTET Good and Bad Counter
GMAC_MMC_RXOCTETCNT_G	0x0188	W	0x00000000	MMC RX OCTET Good Counter
GMAC_MMC_RXMCFRMCNT_G	0x0190	W	0x00000000	MMC RX Multicast Frame Good Counter
GMAC_MMC_RXCRCERR	0x0194	W	0x00000000	MMC RX Carrier
GMAC_MMC_RXLENERR	0x01c8	W	0x00000000	MMC RX Length Error
GMAC_MMC_RXFIFOOVRFLW	0x01d4	W	0x00000000	MMC RX FIFO Overflow
GMAC_MMC_IPC_INT_MSK	0x0200	W	0x00000000	MMC Receive Checksum Offload Interrupt Mask Register The MMC Receive Checksum Offload Interrupt Mask register maintains the masks for the interrupts generated when the receive IPC (Checksum Offload) statistic counters reach half their maximum value , and when they reach their maximum values

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
GMAC_MMC_IPC_INTR	0x0208	W	0x00000000	MMC Receive Checksum Offload Interrupt Register The MMC Receive Checksum Offload Interrupt register maintains the interrupts generated when receive IPC statistic counters reach half their maximum values (0x8000_0000), and when they cross their maximum values (0xFFFF_FFFF). When Counter Stop Rollover is set, then interrupts are set but the counter remains at all-ones. When the MMC IPC counter that caused the interrupt is read, its corresponding interrupt bit is cleared. The counterís least-significant byte lane (bits[7:0]) must be read to clear the interrupt bit
GMAC_MMC_RXIPV4GFRM	0x0210	W	0x00000000	MMC RX IPV4 Good Frame
GMAC_MMC_RXIPV4HDERRFRM	0x0214	W	0x00000000	MMC RX IPV4 Head Error Frame
GMAC_MMC_RXIPV6GFRM	0x0224	W	0x00000000	MMC RX IPV6 Good Frame
GMAC_MMC_RXIPV6HDERRFRM	0x0228	W	0x00000000	MMC RX IPV6 Head Error Frame
GMAC_MMC_RXUDPERRFRM	0x0234	W	0x00000000	MMC RX UDP Error Frame
GMAC_MMC_RXTCPERRFRM	0x023c	W	0x00000000	MMC RX TCP Error Frame
GMAC_MMC_RXICMPERRFRM	0x0244	W	0x00000000	MMC RX ICMP Error Frame
GMAC_MMC_RXIPV4HDERROCT	0x0254	W	0x00000000	MMC RX OCTET IPV4 Head Error
GMAC_MMC_RXIPV6HDERROCT	0x0268	W	0x00000000	MMC RX OCTET IPV6 Head Error
GMAC_MMC_RXUDPERROCT	0x0274	W	0x00000000	MMC RX OCTET UDP Error
GMAC_MMC_RXTCPERROCT	0x027c	W	0x00000000	MMC RX OCTET TCP Error
GMAC_MMC_RXICMPERROCT	0x0284	W	0x00000000	MMC RX OCTET ICMP Error
GMAC_BUS_MODE	0x1000	W	0x00020101	Bus Mode Register

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
GMAC_TX_POLL_DEMAND	0x1004	W	0x00000000	Transmit Poll Demand Register Used by the host to instruct the DMA to poll the Transmit Descriptor List
GMAC_RX_POLL_DEMAND	0x1008	W	0x00000000	Receive Poll Demand Register Used by the Host to instruct the DMA to poll the Receive Descriptor list
GMAC_RX_DESC_LIST_ADDR	0x100c	W	0x00000000	Receive Descriptor List Address Register Points the DMA to the start of the Receive Descriptor list
GMAC_TX_DESC_LIST_ADDR	0x1010	W	0x00000000	Transmit Descriptor List Address Register Points the DMA to the start of the Transmit Descriptor List
GMAC_STATUS	0x1014	W	0x00000000	Status Register The Software driver (application) reads this register during interrupt service routine or polling to determine the status of the DMA
GMAC_OP_MODE	0x1018	W	0x00000000	Operation Mode Register Establishes the Receive and Transmit operating modes and command
GMAC_INT_ENA	0x101c	W	0x00000000	Interrupt Enable Register Enables the interrupts reported by the Status Register
GMAC_OVERFLOW_CNT	0x1020	W	0x00000000	Missed Frame and Buffer Overflow Counter Register Contains the counters for discarded frames because no host Receive Descriptor was available, and discarded frames because of Receive FIFO Overflow
GMAC_REC_INT_WDT_TIMER	0x1024	W	0x00000000	Receive Interrupt Watchdog Timer Register Watchdog time-out for Receive Interrupt (RI) from DMA

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
GMAC_AXI_BUS_MODE	0x1028	W	0x00110001	AXI Bus Mode Register Controls AXI Master behavior (mainly controls burst splitting and number of outstanding requests)
GMAC_AXI_STATUS	0x102c	W	0x00000000	AXI Status Register Gives the idle status of the AXI master's read/write channels
GMAC_CUR_HOST_TX_DESC	0x1048	W	0x00000000	Current Host Transmit Descriptor Register Points to the start of current Transmit Descriptor read by the DMA
GMAC_CUR_HOST_RX_DESC	0x104c	W	0x00000000	Current Host Receive Descriptor Register Points to the start of current Receive Descriptor read by the DMA
GMAC_CUR_HOST_TX_BUF_ADDR	0x1050	W	0x00000000	Current Host Transmit Buffer Address Register Points to the current Transmit Buffer address read by the DMA
GMAC_CUR_HOST_RX_BUF_ADDR	0x1054	W	0x00000000	Current Host Receive Buffer Address Register Points to the current Receive Buffer address read by the DMA

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

17.4.2 Detail Register Description

GMAC_MAC_CONF

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24	RW	0x0	TC Transmit Configuration in RGMII When set, this bit enables the transmission of duplex mode, link speed, and link up/down information to the PHY in the RGMII ports. When this bit is reset, no such information is driven to the PHY

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
23	RW	0x0	WD Watchdog Disable When this bit is set, the GMAC disables the watchdog timer on the receiver, and can receive frames of up to 16,384 bytes. When this bit is reset, the GMAC allows no more than 2,048 bytes (10,240 if JE is set high) of the frame being received and cuts off any bytes received after that
22	RW	0x0	JD Jabber Disable When this bit is set, the GMAC disables the jabber timer on the transmitter, and can transfer frames of up to 16,384 bytes. When this bit is reset, the GMAC cuts off the transmitter if the application sends out more than 2,048 bytes of data (10,240 if JE is set high) during transmission
21	RW	0x0	BE Frame Burst Enable When this bit is set, the GMAC allows frame bursting during transmission in GMII Half-Duplex mode
20	RO	0x0	reserved
19:17	RW	0x0	IFG Inter-Frame Gap These bits control the minimum IFG between frames during transmission. 3'b000: 96 bit times 3'b001: 88 bit times 3'b010: 80 bit times ... 3'b111: 40 bit times
16	RW	0x0	DCRS Disable Carrier Sense During Transmission When set high, this bit makes the MAC transmitter ignore the (G)MII CRS signal during frame transmission in Half-Duplex mode. This request results in no errors generated due to Loss of Carrier or No Carrier during such transmission. When this bit is low, the MAC transmitter generates such errors due to Carrier Sense and will even abort the transmissions
15	RW	0x0	PS Port Select Selects between GMII and MII: 1'b0: GMII (1000 Mbps) 1'b1: MII (10/100 Mbps)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
14	RW	0x0	FES Speed Indicates the speed in Fast Ethernet (MII) mode: 1'b0: 10 Mbps 1'b1: 100 Mbps
13	RW	0x0	DO Disable Receive Own When this bit is set, the GMAC disables the reception of frames when the gmii_txen_o is asserted in Half-Duplex mode. When this bit is reset, the GMAC receives all packets that are given by the PHY while transmitting
12	RW	0x0	LM Loopback Mode When this bit is set, the GMAC operates in loopback mode at GMII/MII. The (G)MII Receive clock input (clk_rx_i) is required for the loopback to work properly, as the Transmit clock is not looped-back internally
11	RW	0x0	DM Duplex Mode When this bit is set, the GMAC operates in a Full-Duplex mode where it can transmit and receive simultaneously. This bit is RO with default value of 1'b1 in Full-Duplex-only configuration
10	RW	0x0	IPC Checksum Offload When this bit is set, the GMAC calculates the 16-bit one's complement of the one's complement sum of all received Ethernet frame payloads. It also checks whether the IPv4 Header checksum (assumed to be bytes 25-26 or 29-30 (VLAN-tagged) of the received Ethernet frame) is correct for the received frame and gives the status in the receive status word. The GMAC core also appends the 16-bit checksum calculated for the IP header datagram payload (bytes after the IPv4 header) and appends it to the Ethernet frame transferred to the application (when Type 2 COE is deselected). When this bit is reset, this function is disabled. When Type 2 COE is selected, this bit, when set, enables IPv4 checksum checking for received frame payloads TCP/UDP/ICMP headers. When this bit is reset, the COE function in the receiver is disabled and the corresponding PCE and IP HCE status bits are always cleared

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
9	RW	0x0	DR Disable Retry When this bit is set, the GMAC will attempt only 1 transmission. When a collision occurs on the GMII/MII, the GMAC will ignore the current frame transmission and report a Frame Abort with excessive collision error in the transmit frame status. When this bit is reset, the GMAC will attempt retries based on the settings of BL
8	RW	0x0	LUD Link Up/Down Indicates whether the link is up or down during the transmission of configuration in RGMII interface: 1'b0: Link Down 1'b1: Link Up
7	RW	0x0	ACS Automatic Pad/CRC Stripping When this bit is set, the GMAC strips the Pad/FCS field on incoming frames only if the length's field value is less than or equal to 1,500 bytes. All received frames with length field greater than or equal to 1,501 bytes are passed to the application without stripping the Pad/FCS field. When this bit is reset, the GMAC will pass all incoming frames to the Host unmodified
6:5	RW	0x0	BL Back-Off Limit The Back-Off limit determines the random integer number (r) of slot time delays (4,096 bit times for 1000 Mbps and 512 bit times for 10/100 Mbps) the GMAC waits before rescheduling a transmission attempt during retries after a collision. This bit is applicable only to Half-Duplex mode and is reserved (RO) in Full-Duplex-only configuration. 2'b00: k = min (n, 10) 2'b01: k = min (n, 8) 2'b10: k = min (n, 4) 2'b11: k = min (n, 1), Where n = retransmission attempt. The random integer r takes the value in the range 0 = r < 2^k

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4	RW	0x0	DC Deferral Check When this bit is set, the deferral check function is enabled in the GMAC. The GMAC will issue a Frame Abort status, along with the excessive deferral error bit set in the transmit frame status when the transmit state machine is deferred for more than 24,288 bit times in 10/100-Mbps mode. If the Core is configured for 1000 Mbps operation, the threshold for deferral is 155,680 bits times. Deferral begins when the transmitter is ready to transmit, but is prevented because of an active CRS (carrier sense) signal on the GMII/MII. Defer time is not cumulative. If the transmitter defers for 10,000 bit times, then transmits, collides, backs off, and then has to defer again after completion of back-off, the deferral timer resets to 0 and restarts. When this bit is reset, the deferral check function is disabled and the GMAC defers until the CRS signal goes inactive
3	RW	0x0	TE Transmitter Enable When this bit is set, the transmit state machine of the GMAC is enabled for transmission on the GMII/MII. When this bit is reset, the GMAC transmit state machine is disabled after the completion of the transmission of the current frame, and will not transmit any further frames
2	RW	0x0	RE Receiver Enable When this bit is set, the receiver state machine of the GMAC is enabled for receiving frames from the GMII/MII. When this bit is reset, the GMAC receive state machine is disabled after the completion of the reception of the current frame, and will not receive any further frames from the GMII/MII
1:0	RO	0x0	reserved

GMAC_MAC_FRM_FILT

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31	RW	0x0	RA Receive All When this bit is set, the GMAC Receiver module passes to the Application all frames received irrespective of whether they pass the address filter. The result of the SA/DA filtering is updated (pass or fail) in the corresponding bits in the Receive Status Word. When this bit is reset, the Receiver module passes to the Application only those frames that pass the SA/DA address filter
30:11	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
10	RW	0x0	HPF Hash or Perfect Filter When set, this bit configures the address filter to pass a frame if it matches either the perfect filtering or the hash filtering as set by HMC or HUC bits. When low and if the HUC/HMC bit is set, the frame is passed only if it matches the Hash filter
9	RW	0x0	SAF Source Address Filter Enable The GMAC core compares the SA field of the received frames with the values programmed in the enabled SA registers. If the comparison matches, then the SAMatch bit of RxStatus Word is set high. When this bit is set high and the SA filter fails, the GMAC drops the frame. When this bit is reset, then the GMAC Core forwards the received frame to the application and with the updated SA Match bit of the RxStatus depending on the SA address comparison
8	RW	0x0	SAIF SA Inverse Filtering When this bit is set, the Address Check block operates in inverse filtering mode for the SA address comparison. The frames whose SA matches the SA registers will be marked as failing the SA Address filter. When this bit is reset, frames whose SA does not match the SA registers will be marked as failing the SA Address filter
7:6	RW	0x0	PCF Pass Control Frames These bits control the forwarding of all control frames (including unicast and multicast PAUSE frames). Note that the processing of PAUSE control frames depends only on RFE of Register GMAC_FLOW_CTRL[2]. 2'b00: GMAC filters all control frames from reaching the application. 2'b01: GMAC forwards all control frames except PAUSE control frames to application even if they fail the Address filter. 2'b10: GMAC forwards all control frames to application even if they fail the Address Filter. 2'b11: GMAC forwards control frames that pass the Address Filter
5	RW	0x0	DBF Disable Broadcast Frames When this bit is set, the AFM module filters all incoming broadcast frames. When this bit is reset, the AFM module passes all received broadcast frames

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4	RW	0x0	PM Pass All Multicast When set, this bit indicates that all received frames with a multicast destination address (first bit in the destination address field is '1') are passed. When reset, filtering of multicast frame depends on HMC bit
3	RW	0x0	DAIF DA Inverse Filtering When this bit is set, the Address Check block operates in inverse filtering mode for the DA address comparison for both unicast and multicast frames. When reset, normal filtering of frames is performed
2	RW	0x0	HMC Hash Multicast When set, MAC performs destination address filtering of received multicast frames according to the hash table. When reset, the MAC performs a perfect destination address filtering for multicast frames, that is, it compares the DA field with the values programmed in DA registers
1	RW	0x0	HUC Hash Unicast When set, MAC performs destination address filtering of unicast frames according to the hash table. When reset, the MAC performs a perfect destination address filtering for unicast frames, that is, it compares the DA field with the values programmed in DA registers
0	RW	0x0	PR Promiscuous Mode When this bit is set, the Address Filter module passes all incoming frames regardless of its destination or source address. The SA/DA Filter Fails status bits of the Receive Status Word will always be cleared when PR is set

GMAC_HASH_TAB_HI

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	HTH Hash Table High This field contains the upper 32 bits of Hash table

GMAC_HASH_TAB_LO

Address: Operational Base + offset (0x000c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	HTL Hash Table Low This field contains the lower 32 bits of Hash table

GMAC_GMII_ADDR

Address: Operational Base + offset (0x0010)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:11	RW	0x00	PA Physical Layer Address This field tells which of the 32 possible PHY devices are being accessed
10:6	RW	0x00	GR GMII Register These bits select the desired GMII register in the selected PHY device

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5:2	RW	0x0	CR APB Clock Range The APB Clock Range selection determines the frequency of the MDC clock as per the pclk_gmac frequency used in your design. The suggested range of pclk_gmac frequency applicable for each value below (when Bit[5] = 0) ensures that the MDC clock is approximately between the frequency range 1.0 MHz - 2.5 MHz. Selection pclk_gmac MDC Clock 0000 60-100 MHz pclk_gmac/42 0001 100-150 MHz pclk_gmac/62 0010 20-35 MHz pclk_gmac/16 0011 35-60 MHz pclk_gmac/26 0100 150-250 MHz pclk_gmac/102 0101 250-300 MHz pclk_gmac/124 0110, 0111 Reserved When bit 5 is set, you can achieve MDC clock of frequency higher than the IEEE 802.3 specified frequency limit of 2.5 MHz and program a clock divider of lower value. For example, when pclk_gmac is of frequency 100 MHz and you program these bits as "1010", then the resultant MDC clock will be of 12.5 MHz which is outside the limit of IEEE 802.3 specified range. Please program the values given below only if the interfacing chips supports faster MDC clocks. Selection MDC Clock 1000 pclk_gmac/4 1001 pclk_gmac/6 1010 pclk_gmac/8 1011 pclk_gmac/10 1100 pclk_gmac/12 1101 pclk_gmac/14 1110 pclk_gmac/16 1111 pclk_gmac/18
1	RW	0x0	GW GMII Write When set, this bit tells the PHY that this will be a Write operation using register GMAC_GMII_DATA. If this bit is not set, this will be a Read operation, placing the data in register GMAC_GMII_DATA

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
0	W1C	0x0	GB GMII Busy This bit should read a logic 0 before writing to Register GMII_ADDR and Register GMII_DATA. This bit must also be set to 0 during a Write to Register GMII_ADDR. During a PHY register access, this bit will be set to 1'b1 by the Application to indicate that a Read or Write access is in progress. Register GMII_DATA (GMII Data) should be kept valid until this bit is cleared by the GMAC during a PHY Write operation. The Register GMII_DATA is invalid until this bit is cleared by the GMAC during a PHY Read operation. The Register GMII_ADDR (GMII Address) should not be written to until this bit is cleared

GMAC_GMII_DATA

Address: Operational Base + offset (0x0014)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RW	0x0000	GD GMII Data This contains the 16-bit data value read from the PHY after a Management Read operation or the 16-bit data value to be written to the PHY before a Management Write operation

GMAC_FLOW_CTRL

Address: Operational Base + offset (0x0018)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	PT Pause Time This field holds the value to be used in the Pause Time field in the transmit control frame. If the Pause Time bits is configured to be double-synchronized to the (G)MII clock domain, then consecutive writes to this register should be performed only after at least 4 clock cycles in the destination clock domain
15:8	RO	0x0	reserved
7	RW	0x0	DZPQ Disable Zero-Quanta Pause When set, this bit disables the automatic generation of Zero-Quanta Pause Control frames on the de-assertion of the flow-control signal from the FIFO layer (MTL or external sideband flow control signal sbd_flowctrl_i/mti_flowctrl_i). When this bit is reset, normal operation with automatic Zero-Quanta Pause Control frame generation is enabled
6	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5:4	RW	0x0	PLT Pause Low Threshold This field configures the threshold of the PAUSE timer at which the input flow control signal mti_flowctrl_i (or sbd_flowctrl_i) is checked for automatic retransmission of PAUSE Frame. The threshold values should be always less than the Pause Time configured in Bits[31:16]. For example, if PT = 100H (256 slot-times), and PLT = 01, then a second PAUSE frame is automatically transmitted if the mti_flowctrl_i signal is asserted at 228 (256-28) slot-times after the first PAUSE frame is transmitted. Selection Threshold 00 Pause time minus 4 slot times 01 Pause time minus 28 slot times 10 Pause time minus 144 slot times 11 Pause time minus 256 slot times Slot time is defined as time taken to transmit 512 bits (64 bytes) on the GMII/MII interface
3	RW	0x0	UP Unicast Pause Frame Detect When this bit is set, the GMAC will detect the Pause frames with the station's unicast address specified in MAC Address0 High Register and MAC Address0 Low Register, in addition to the detecting Pause frames with the unique multicast address. When this bit is reset, the GMAC will detect only a Pause frame with the unique multicast address specified in the 802.3x standard
2	RW	0x0	RFE Receive Flow Control Enable When this bit is set, the GMAC will decode the received Pause frame and disable its transmitter for a specified (Pause Time) time. When this bit is reset, the decode function of the Pause frame is disabled
1	RW	0x0	TFE Transmit Flow Control Enable In Full-Duplex mode, when this bit is set, the GMAC enables the flow control operation to transmit Pause frames. When this bit is reset, the flow control operation in the GMAC is disabled, and the GMAC will not transmit any Pause frames. In Half-Duplex mode, when this bit is set, the GMAC enables the back-pressure operation. When this bit is reset, the backpressure feature is disabled

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
0	RW	0x0	FCB_BPA Flow Control Busy/Backpressure Activate This bit initiates a Pause Control frame in Full-Duplex mode and activates the backpressure function in Half-Duplex mode if TFE bit is set. In Full-Duplex mode, this bit should be read as 1'b0 before writing to the register GMAC_FLOW_CTRL. To initiate a pause control frame, the application must set this bit to 1'b1. During a transfer of the control frame, this bit will continue to be set to signify that a frame transmission is in progress. After the completion of Pause control frame transmission, the GMAC will reset this bit to 1'b0. The register GMAC_FLOW_CTRL should not be written to until this bit is cleared. In Half-Duplex mode, when this bit is set (and TFE is set), then backpressure is asserted by the GMAC Core. During backpressure, when the GMAC receives a new frame, the transmitter starts sending a JAM pattern resulting in a collision. This control register bit is logically OR'ed with the mti_flowctrl_i input signal for the backpressure function

GMAC_VLAN_TAG

Address: Operational Base + offset (0x001c)

Bit	Attr	Reset Value	Description
31:17	RO	0x0	reserved
16	RW	0x0	ETV Enable 12-Bit VLAN Tag Comparison When this bit is set, a 12-bit VLAN identifier, rather than the complete 16-bit VLAN tag, is used for comparison and filtering. Bits[11:0] of the VLAN tag are compared with the corresponding field in the received VLAN-tagged frame. When this bit is reset, all 16 bits of the received VLAN frame's fifteenth and sixteenth bytes are used for comparison
15:0	RW	0x0000	VL VLAN Tag Identifier for Receive Frames This contains the 802.1Q VLAN tag to identify VLAN frames, and is compared to the fifteenth and sixteenth bytes of the frames being received for VLAN frames. Bits[15:13] are the User Priority, Bit[12] is the Canonical Format Indicator (CFI) and bits[11:0] are the VLAN tag's VLAN Identifier (VID) field. When the ETV bit is set, only the VID (Bits[11:0]) is used for comparison. If VL (VL[11:0] if ETV is set) is all zeros, the GMAC does not check the fifteenth and sixteenth bytes for VLAN tag comparison, and declares all frames with a Type field value of 0x8100 to be VLAN frames

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

GMAC_DEBUG

Address: Operational Base + offset (0x0024)

Bit	Attr	Reset Value	Description
31:26	RO	0x0	reserved
25	RW	0x0	TFIFO3 When high, it indicates that the MTL TxStatus FIFO is full and hence the MTL will not be accepting any more frames for transmission
24	RW	0x0	TFIFO2 When high, it indicates that the MTL TxFIFO is not empty and has some data left for transmission
23	RO	0x0	reserved
22	RW	0x0	TFIFO1 When high, it indicates that the MTL TxFIFO Write Controller is active and transferring data to the TxFIFO
21:20	RW	0x0	TFIFOSTA This indicates the state of the TxFIFO read Controller: 2'b00: IDLE state 2'b01: READ state (transferring data to MAC transmitter) 2'b10: Waiting for TxStatus from MAC transmitter 2'b11: Writing the received TxStatus or flushing the TxFIFO
19	RW	0x0	PAUSE When high, it indicates that the MAC transmitter is in PAUSE condition (in full-duplex only) and hence will not schedule any frame for transmission
18:17	RW	0x0	TSAT This indicates the state of the MAC Transmit Frame Controller module: 2'b00: IDLE 2'b01: Waiting for Status of previous frame or IFG/backoff period to be over 2'b10: Generating and transmitting a PAUSE control frame (in full duplex mode) 2'b11: Transferring input frame for transmission
16	RW	0x0	TACT When high, it indicates that the MAC GMII/MII transmit protocol engine is actively transmitting data and not in IDLE state
15:10	RO	0x0	reserved
9:8	RW	0x0	RFIFO This gives the status of the RxFIFO Fill-level: 2'b00: RxFIFO Empty 2'b01: RxFIFO fill-level below flow-control de-activate threshold 2'b10: RxFIFO fill-level above flow-control activate threshold 2'b11: RxFIFO Full
7	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
6:5	RW	0x0	RFIFORD It gives the state of the RxFIFO read Controller: 2'b00: IDLE state 2'b01: Reading frame data 2'b10: Reading frame status (or time-stamp) 2'b11: Flushing the frame data and Status
4	RW	0x0	RFIFOWR When high, it indicates that the MTL RxFIFO Write Controller is active and transferring a received frame to the FIFO
3	RO	0x0	reserved
2:1	RW	0x0	ACT When high, it indicates the active state of the small FIFO Read and Write controllers respectively of the MAC receive Frame Controller module
0	RW	0x0	RDB When high, it indicates that the MAC GMII/MII receive protocol engine is actively receiving data and not in IDLE state

GMAC_PMT_CTRL_STA

Address: Operational Base + offset (0x002c)

Bit	Attr	Reset Value	Description
31	W1C	0x0	WFFRPR Wake-Up Frame Filter Register Pointer Reset When set, resets the Remote Wake-up Frame Filter register pointer to 3'b000. It is automatically cleared after 1 clock cycle
30:10	RO	0x0	reserved
9	RW	0x0	GU Global Unicast When set, enables any unicast packet filtered by the GMAC (DAF) address recognition to be a wake-up frame
8:7	RO	0x0	reserved
6	RC	0x0	WFR Wake-Up Frame Received When set, this bit indicates the power management event was generated due to reception of a wake-up frame. This bit is cleared by a read into this register
5	RC	0x0	MPR Magic Packet Received When set, this bit indicates the power management event was generated by the reception of a Magic Packet. This bit is cleared by a read into this register
4:3	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2	RW	0x0	WFE Wake-Up Frame Enable When set, enables generation of a power management event due to wake-up frame reception
1	RW	0x0	MPE Magic Packet Enable When set, enables generation of a power management event due to Magic Packet reception
0	R/WSC	0x0	PD Power Down When set, all received frames will be dropped. This bit is cleared automatically when a magic packet or Wake-Up frame is received, and Power-Down mode is disabled. Frames received after this bit is cleared are forwarded to the application. This bit must only be set when either the Magic Packet Enable or Wake-Up Frame Enable bit is set high

GMAC_INT_STATUS

Address: Operational Base + offset (0x0038)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7	RO	0x0	MRCOIS MMC Receive Checksum Offload Interrupt Status This bit is set high whenever an interrupt is generated in the MMC Receive Checksum Offload Interrupt Register. This bit is cleared when all the bits in this interrupt register are cleared
6	RO	0x0	MTIS MMC Transmit Interrupt Status This bit is set high whenever an interrupt is generated in the MMC Transmit Interrupt Register. This bit is cleared when all the bits in this interrupt register are cleared. This bit is only valid when the optional MMC module is selected during configuration
5	RO	0x0	MRIS MMC Receive Interrupt Status This bit is set high whenever an interrupt is generated in the MMC Receive Interrupt Register. This bit is cleared when all the bits in this interrupt register are cleared. This bit is only valid when the optional MMC module is selected during configuration
4	RO	0x0	MIS MMC Interrupt Status This bit is set high whenever any of bits 7:5 is set high and cleared only when all of these bits are low. This bit is valid only when the optional MMC module is selected during configuration

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
3	RO	0x0	PIS PMT Interrupt Status This bit is set whenever a Magic packet or Wake-on-LAN frame is received in Power-Down mode). This bit is cleared when both bits[6:5] are cleared due to a read operation to the register GMAC_PMT_CTRL_STA
2:1	RO	0x0	reserved
0	RO	0x0	RIS RGMII Interrupt Status This bit is set due to any change in value of the Link Status of RGMII interface. This bit is cleared when the user makes a read operation the RGMII Status register

GMAC_INT_MASK

Address: Operational Base + offset (0x003c)

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3	RW	0x0	PIM PMT Interrupt Mask This bit when set, will disable the assertion of the interrupt signal due to the setting of PMT Interrupt Status bit in Register GMAC_INT_STATUS
2:1	RO	0x0	reserved
0	RW	0x0	RIM RGMII Interrupt Mask This bit when set, will disable the assertion of the interrupt signal due to the setting of RGMII Interrupt Status bit in Register GMAC_INT_STATUS

GMAC_MAC_ADDR0_HI

Address: Operational Base + offset (0x0040)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RW	0xffff	A47_A32 MAC Address0 [47:32] This field contains the upper 16 bits (47:32) of the 6-byte first MAC address. This is used by the MAC for filtering for received frames and for inserting the MAC address in the Transmit Flow Control (PAUSE) Frames

GMAC_MAC_ADDR0_LO

Address: Operational Base + offset (0x0044)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RW	0xffffffff	A31_A0 MAC Address0 [31:0] This field contains the lower 32 bits of the 6-byte first MAC address. This is used by the MAC for filtering for received frames and for inserting the MAC address in the Transmit Flow Control (PAUSE) Frames

GMAC_AN_CTRL

Address: Operational Base + offset (0x00c0)

Bit	Attr	Reset Value	Description
31:13	RO	0x0	reserved
12	RW	0x0	ANE Auto-Negotiation Enable When set, will enable the GMAC to perform auto-negotiation with the link partner. Clearing this bit will disable auto-negotiation
11:10	RO	0x0	reserved
9	R/WSC	0x0	RAN Restart Auto-Negotiation When set, will cause auto-negotiation to restart if the ANE is set. This bit is self-clearing after auto-negotiation starts. This bit should be cleared for normal operation
8:0	RO	0x0	reserved

GMAC_AN_STATUS

Address: Operational Base + offset (0x00c4)

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5	RO	0x0	ANC Auto-Negotiation Complete When set, this bit indicates that the auto-negotiation process is completed. This bit is cleared when auto-negotiation is reinitiated
4	RO	0x0	reserved
3	RO	0x1	ANA Auto-Negotiation Ability This bit is always high, because the GMAC supports auto-negotiation
2	R/WSC	0x0	LS Link Status When set, this bit indicates that the link is up. When cleared, this bit indicates that the link is down
1:0	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

GMAC_AN_ADV

Address: Operational Base + offset (0x00c8)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15	RO	0x0	NP Next Page Support This bit is tied to low, because the GMAC does not support the next page
14	RO	0x0	reserved
13:12	RW	0x0	RFE Remote Fault Encoding These 2 bits provide a remote fault encoding, indicating to a link partner that a fault or error condition has occurred
11:9	RO	0x0	reserved
8:7	RW	0x3	PSE Pause Encoding These 2 bits provide an encoding for the PAUSE bits, indicating that the GMAC is capable of configuring the PAUSE function as defined in IEEE 802.3x
6	RW	0x1	HD Half-Duplex This bit, when set high, indicates that the GMAC supports Half-Duplex. This bit is tied to low (and RO) when the GMAC is configured for Full-Duplex-only operation
5	RW	0x1	FD Full-Duplex This bit, when set high, indicates that the GMAC supports Full-Duplex
4:0	RO	0x0	reserved

GMAC_AN_LINK_PART_AB

Address: Operational Base + offset (0x00cc)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15	RO	0x0	NP Next Page Support When set, this bit indicates that more next page information is available. When cleared, this bit indicates that next page exchange is not desired
14	RO	0x0	ACK Acknowledge When set, this bit is used by the auto-negotiation function to indicate that the link partner has successfully received the GMAC's base page. When cleared, it indicates that a successful receipt of the base page has not been achieved
13:12	RO	0x0	RFE Remote Fault Encoding These 2 bits provide a remote fault encoding, indicating a fault or error condition of the link partner
11:9	RO	0x0	reserved
8:7	RO	0x0	PSE Pause Encoding These 2 bits provide an encoding for the PAUSE bits, indicating that the link partner's capability of configuring the PAUSE function as defined in IEEE 802.3x
6	RO	0x0	HD Half-Duplex When set, this bit indicates that the link partner has the ability to operate in Half-Duplex mode. When cleared, the link partner does not have the ability to operate in Half-Duplex mode
5	RO	0x0	FD Full-Duplex When set, this bit indicates that the link partner has the ability to operate in Full-Duplex mode. When cleared, the link partner does not have the ability to operate in Full-Duplex mode
4:0	RO	0x0	reserved

GMAC_AN_EXP

Address: Operational Base + offset (0x00d0)

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved
2	RO	0x0	NPA Next Page Ability This bit is tied to low, because the GMAC does not support next page function

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
1	RO	0x0	NPR New Page Received When set, this bit indicates that a new page has been received by the GMAC. This bit will be cleared when read
0	RO	0x0	reserved

GMAC_INTF_MODE_STA

Address: Operational Base + offset (0x00d8)

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3	RO	0x0	LST Link Status Indicates whether the link is up (1'b1) or down (1'b0)
2:1	RO	0x0	LSD Link Speed Indicates the current speed of the link: 2'b00: 2.5 MHz 2'b01: 25 MHz 2'b10: 125 MHz
0	RW	0x0	LM Link Mode Indicates the current mode of operation of the link: 1'b0: Half-Duplex mode 1'b1: Full-Duplex mode

GMAC_MMC_CTRL

Address: Operational Base + offset (0x0100)

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5	RW	0x0	FHP Full-Half preset When low and bit4 is set, all MMC counters get preset to almost-half value. All octet counters get preset to 0x7FFF_F800 (half - 2K Bytes) and all frame-counters gets preset to 0x7FFF_FFF0 (half - 16) When high and bit4 is set, all MMC counters get preset to almost-full value. All octet counters get preset to 0xFFFF_F800 (full - 2K Bytes) and all frame-counters gets preset to 0xFFFF_FFF0 (full - 16)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4	R/W SC	0x0	CP Counters Preset When set, all counters will be initialized or preset to almost full or almost half as per Bit5 above. This bit will be cleared automatically after 1 clock cycle. This bit along with bit5 is useful for debugging and testing the assertion of interrupts due to MMC counter becoming half-full or full
3	RW	0x0	MCF MMC Counter Freeze When set, this bit freezes all the MMC counters to their current value. (None of the MMC counters are updated due to any transmitted or received frame until this bit is reset to 0. If any MMC counter is read with the Reset on Read bit set, then that counter is also cleared in this mode.)
2	RW	0x0	ROR Reset on Read When set, the MMC counters will be reset to zero after Read (self-clearing after reset). The counters are cleared when the least significant byte lane (bits[7:0]) is read
1	RW	0x0	CSR Counter Stop Rollover When set, counter after reaching maximum value will not roll over to zero
0	R/W SC	0x0	CR Counters Reset When set, all counters will be reset. This bit will be cleared automatically after 1 clock cycle

GMAC_MMC_RX_INTR

Address: Operational Base + offset (0x0104)

Bit	Attr	Reset Value	Description
31:22	RO	0x0	reserved
21	RW	0x0	INT21 The bit is set when the rxfifooverflow counter reaches half the maximum value, and also when it reaches the maximum value
20:19	RO	0x0	reserved
18	RC	0x0	INT18 The bit is set when the rxlengtherror counter reaches half the maximum value, and also when it reaches the maximum value
17:6	RO	0x0	reserved
5	RW	0x0	INT5 The bit is set when the rxcrcerror counter reaches half the maximum value, and also when it reaches the maximum value

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4	RC	0x0	INT4 The bit is set when the rxmulticastframes_g counter reaches half the maximum value, and also when it reaches the maximum value
3	RO	0x0	reserved
2	RC	0x0	INT2 The bit is set when the rxoctetcount_g counter reaches half the maximum value, and also when it reaches the maximum value
1	RC	0x0	INT1 The bit is set when the rxoctetcount_gb counter reaches half the maximum value, and also when it reaches the maximum value
0	RC	0x0	INT0 The bit is set when the rxframecount_gb counter reaches half the maximum value, and also when it reaches the maximum value

GMAC_MMC_TX_INTR

Address: Operational Base + offset (0x0108)

Bit	Attr	Reset Value	Description
31:22	RO	0x0	reserved
21	RC	0x0	INT21 The bit is set when the txframecount_g counter reaches half the maximum value, and also when it reaches the maximum value
20	RC	0x0	INT20 The bit is set when the txoctetcount_g counter reaches half the maximum value, and also when it reaches the maximum value
19	RC	0x0	INT19 The bit is set when the txcarriererror counter reaches half the maximum value, and also when it reaches the maximum value
18:14	RO	0x0	reserved
13	RC	0x0	INT13 The bit is set when the txunderflowerror counter reaches half the maximum value, and also when it reaches the maximum value
12:2	RO	0x0	reserved
1	RC	0x0	INT1 The bit is set when the txframecount_gb counter reaches half the maximum value, and also when it reaches the maximum value
0	RC	0x0	INT0 The bit is set when the txoctetcount_gb counter reaches half the maximum value, and also when it reaches the maximum value

GMAC_MMC_RX_INT_MSK

Address: Operational Base + offset (0x010c)

Bit	Attr	Reset Value	Description
31:22	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
21	RW	0x0	INT21 Setting this bit masks the interrupt when the rxfifooverflow counter reaches half the maximum value, and also when it reaches the maximum value
20:19	RO	0x0	reserved
18	RW	0x0	INT18 Setting this bit masks the interrupt when the rxlengtherror counter reaches half the maximum value, and also when it reaches the maximum value
17:6	RO	0x0	reserved
5	RW	0x0	INT5 Setting this bit masks the interrupt when the rxcrcerror counter reaches half the maximum value, and also when it reaches the maximum value
4	RW	0x0	INT4 Setting this bit masks the interrupt when the rxmulticastframes_g counter reaches half the maximum value, and also when it reaches the maximum value
3	RO	0x0	reserved
2	RW	0x0	INT2 Setting this bit masks the interrupt when the rxoctetcount_g counter reaches half the maximum value, and also when it reaches the maximum value
1	RW	0x0	INT1 Setting this bit masks the interrupt when the rxoctetcount_gb counter reaches half the maximum value, and also when it reaches the maximum value
0	RW	0x0	INT0 Setting this bit masks the interrupt when the rxframecount_gb counter reaches half the maximum value, and also when it reaches the maximum value

GMAC_MMC_TX_INT_MSK

Address: Operational Base + offset (0x0110)

Bit	Attr	Reset Value	Description
31:22	RO	0x0	reserved
21	RW	0x0	INT21 Setting this bit masks the interrupt when the txframecount_g counter reaches half the maximum value, and also when it reaches the maximum value
20	RW	0x0	INT20 Setting this bit masks the interrupt when the txoctetcount_g counter reaches half the maximum value, and also when it reaches the maximum value

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
19	RW	0x0	INT19 Setting this bit masks the interrupt when the txcarriererror counter reaches half the maximum value, and also when it reaches the maximum value
18:14	RO	0x0	reserved
13	RW	0x0	INT13 Setting this bit masks the interrupt when the txunderflowerror counter reaches half the maximum value, and also when it reaches the maximum value
12:2	RO	0x0	reserved
1	RW	0x0	INT1 Setting this bit masks the interrupt when the txframecount_gb counter reaches half the maximum value, and also when it reaches the maximum value
0	RW	0x0	INT0 Setting this bit masks the interrupt when the txoctetcount_gb counter reaches half the maximum value, and also when it reaches the maximum value

GMAC_MMC_TXOCTETCNT_GB

Address: Operational Base + offset (0x0114)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	txoctetcount_gb Number of bytes transmitted, exclusive of preamble and retried bytes, in good and bad frames

GMAC_MMC_TXFRMCNT_GB

Address: Operational Base + offset (0x0118)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	txframecount_gb Number of good and bad frames transmitted, exclusive of retried frames

GMAC_MMC_TXUNDFLWERR

Address: Operational Base + offset (0x0148)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	txunderflowerror Number of frames aborted due to frame underflow error

GMAC_MMC_TXCARERR

Address: Operational Base + offset (0x0160)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	txcarriererror Number of frames aborted due to carrier sense error (no carrier or loss of carrier)

GMAC_MMC_TXOCTETCNT_G

Address: Operational Base + offset (0x0164)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	txoctetcount_g Number of bytes transmitted, exclusive of preamble, in good frames only

GMAC_MMC_TXFRMCNT_G

Address: Operational Base + offset (0x0168)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	txframecount_g Number of good frames transmitted

GMAC_MMC_RXFRMCNT_GB

Address: Operational Base + offset (0x0180)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxframecount_gb Number of good and bad frames received

GMAC_MMC_RXOCTETCNT_GB

Address: Operational Base + offset (0x0184)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxoctetcount_gb Number of bytes received, exclusive of preamble, in good and bad frames

GMAC_MMC_RXOCTETCNT_G

Address: Operational Base + offset (0x0188)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxoctetcount_g Number of bytes received, exclusive of preamble, only in good frames

GMAC_MMC_RXMCFRMCNT_G

Address: Operational Base + offset (0x0190)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxmulticastframes_g Number of good multicast frames received

GMAC_MMC_RXCRCERR

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x0194)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxcrcerror Number of frames received with CRC error

GMAC_MMC_RXLENERR

Address: Operational Base + offset (0x01c8)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxlengtherror Number of frames received with length error (Length type field ≠frame size), for all frames with valid length field

GMAC_MMC_RXFIFOOVRFLW

Address: Operational Base + offset (0x01d4)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxfifooverflow Number of missed received frames due to FIFO overflow

GMAC_MMC_IPC_INT_MSK

Address: Operational Base + offset (0x0200)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29	RW	0x0	INT29 Setting this bit masks the interrupt when the rxicmp_err_octets counter reaches half the maximum value, and also when it reaches the maximum value
28	RO	0x0	reserved
27	RW	0x0	INT27 Setting this bit masks the interrupt when the rxtcp_err_octets counter reaches half the maximum value, and also when it reaches the maximum value
26	RO	0x0	reserved
25	RW	0x0	INT25 Setting this bit masks the interrupt when the rxudp_err_octets counter reaches half the maximum value, and also when it reaches the maximum value
24:23	RO	0x0	reserved
22	RW	0x0	INT22 Setting this bit masks the interrupt when the rxipv6_hdrerr_octets counter reaches half the maximum value, and also when it reaches the maximum value
21:18	RO	0x0	reserved
17	RW	0x0	INT17 Setting this bit masks the interrupt when the rxipv4_hdrerr_octets counter reaches half the maximum value, and also when it reaches the maximum value

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
16:14	RO	0x0	reserved
13	RW	0x0	INT13 Setting this bit masks the interrupt when the rxicmp_err_frms counter reaches half the maximum value, and also when it reaches the maximum value
12	RO	0x0	reserved
11	RW	0x0	INT11 Setting this bit masks the interrupt when the rxtcp_err_frms counter reaches half the maximum value, and also when it reaches the maximum value
10	RO	0x0	reserved
9	RW	0x0	INT9 Setting this bit masks the interrupt when the rxudp_err_frms counter reaches half the maximum value, and also when it reaches the maximum value
8:7	RO	0x0	reserved
6	RW	0x0	INT6 Setting this bit masks the interrupt when the rxipv6_hdrerr_frms counter reaches half the maximum value, and also when it reaches the maximum value
5	RW	0x0	INT5 Setting this bit masks the interrupt when the rxipv6_gd_frms counter reaches half the maximum value, and also when it reaches the maximum value
4:2	RO	0x0	reserved
1	RW	0x0	INT1 Setting this bit masks the interrupt when the rxipv4_hdrerr_frms counter reaches half the maximum value, and also when it reaches the maximum value
0	RW	0x0	INT0 Setting this bit masks the interrupt when the rxipv4_gd_frms counter reaches half the maximum value, and also when it reaches the maximum value

GMAC_MMC_IPC_INTR

Address: Operational Base + offset (0x0208)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29	RC	0x0	INT29 The bit is set when the rxicmp_err_octets counter reaches half the maximum value, and also when it reaches the maximum value
28	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
27	RC	0x0	INT27 The bit is set when the rxtcp_err_octets counter reaches half the maximum value, and also when it reaches the maximum value
26	RO	0x0	reserved
25	RC	0x0	INT25 The bit is set when the rxudp_err_octets counter reaches half the maximum value, and also when it reaches the maximum value
24:23	RO	0x0	reserved
22	RC	0x0	INT22 The bit is set when the rxipv6_hdrerr_octets counter reaches half the maximum value, and also when it reaches the maximum value
21:18	RO	0x0	reserved
17	RC	0x0	INT17 The bit is set when the rxipv4_hdrerr_octets counter reaches half the maximum value, and also when it reaches the maximum value
16:14	RO	0x0	reserved
13	RC	0x0	INT13 The bit is set when the rxicmp_err_frms counter reaches half the maximum value, and also when it reaches the maximum value
12	RO	0x0	reserved
11	RC	0x0	INT11 The bit is set when the rxtcp_err_frms counter reaches half the maximum value, and also when it reaches the maximum value
10	RO	0x0	reserved
9	RC	0x0	INT9 The bit is set when the rxudp_err_frms counter reaches half the maximum value, and also when it reaches the maximum value
8:7	RO	0x0	reserved
6	RC	0x0	INT6 The bit is set when the rxipv6_hdrerr_frms counter reaches half the maximum value, and also when it reaches the maximum value
5	RC	0x0	INT5 The bit is set when the rxipv6_gd_frms counter reaches half the maximum value, and also when it reaches the maximum value
4:2	RO	0x0	reserved
1	RC	0x0	INT1 The bit is set when the rxipv4_hdrerr_frms counter reaches half the maximum value, and also when it reaches the maximum value
0	RC	0x0	INT0 The bit is set when the rxipv4_gd_frms counter reaches half the maximum value, and also when it reaches the maximum value

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

GMAC_MMC_RXIPV4GFRM

Address: Operational Base + offset (0x0210)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxipv4_gd_frms Number of good IPv4 datagrams received with the TCP, UDP, or ICMP payload

GMAC_MMC_RXIPV4HDERRFRM

Address: Operational Base + offset (0x0214)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxipv4_hdrerr_frms Number of IPv4 datagrams received with header (checksum, length, or version mismatch) errors

GMAC_MMC_RXIPV6GFRM

Address: Operational Base + offset (0x0224)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxipv6_gd_frms Number of good IPv6 datagrams received with TCP, UDP, or ICMP payloads

GMAC_MMC_RXIPV6HDERRFRM

Address: Operational Base + offset (0x0228)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxipv6_hdrerr_frms Number of IPv6 datagrams received with header errors (length or version mismatch)

GMAC_MMC_RXUDPERRFRM

Address: Operational Base + offset (0x0234)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxudp_err_frms Number of good IP datagrams whose UDP payload has a checksum error

GMAC_MMC_RXTCPERRFRM

Address: Operational Base + offset (0x023c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxtcp_err_frms Number of good IP datagrams whose TCP payload has a checksum error

GMAC_MMC_RXICMPERRFRM

Address: Operational Base + offset (0x0244)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxicmp_err_frms Number of good IP datagrams whose ICMP payload has a checksum error

GMAC_MMC_RXIPV4HDERROCT

Address: Operational Base + offset (0x0254)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxipv4_hdrerr_octets Number of bytes received in IPv4 datagrams with header errors (checksum, length, version mismatch). The value in the Length field of IPv4 header is used to update this counter

GMAC_MMC_RXIPV6HDERROCT

Address: Operational Base + offset (0x0268)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxipv6_hdrerr_octets Number of bytes received in IPv6 datagrams with header errors (length, version mismatch). The value in the IPv6 header's Length field is used to update this counter

GMAC_MMC_RXUDPERROCT

Address: Operational Base + offset (0x0274)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxudp_err_octets Number of bytes received in a UDP segment that had checksum errors

GMAC_MMC_RXTCPERROCT

Address: Operational Base + offset (0x027c)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxtcp_err_octets Number of bytes received in a TCP segment with checksum errors

GMAC_MMC_RXICMPERROCT

Address: Operational Base + offset (0x0284)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxicmp_err_octets Number of bytes received in an ICMP segment with checksum errors

GMAC_BUS_MODE

Address: Operational Base + offset (0x1000)

Bit	Attr	Reset Value	Description
31:26	RO	0x0	reserved
25	RW	0x0	AAL Address-Aligned Beats When this bit is set high and the FB bit equals 1, the AXI interface generates all bursts aligned to the start address LS bits. If the FB bit equals 0, the first burst (accessing the data buffer's start address) is not aligned, but subsequent bursts are aligned to the address
24	RW	0x0	PBL_Mode 8xPBL Mode When set high, this bit multiplies the PBL value programmed (bits [22:17] and bits [13:8]) eight times. Thus the DMA will transfer data in to a maximum of 8, 16, 32, 64, 128, and 256 beats depending on the PBL value
23	RW	0x0	USP Use Separate PBL When set high, it configures the RxDMA to use the value configured in bits [22:17] as PBL while the PBL value in bits [13:8] is applicable to TxDMA operations only. When reset to low, the PBL value in bits [13:8] is applicable for both DMA engines
22:17	RW	0x01	RPBL RxDMA PBL These bits indicate the maximum number of beats to be transferred in one RxDMA transaction. This will be the maximum value that is used in a single block Read/Write. The RxDMA will always attempt to burst as specified in RPBL each time it starts a Burst transfer on the host bus. RPBL can be programmed with permissible values of 1, 2, 4, 8, 16, and 32. Any other value will result in undefined behavior. These bits are valid and applicable only when USP is set high
16	RW	0x0	FB Fixed Burst This bit controls whether the AXI Master interface performs fixed burst transfers or not. When set, the AHB will use only SINGLE, INCR4, INCR8 or INCR16 during start of normal burst transfers. When reset, the AXI will use SINGLE and INCR burst transfer operations
15:14	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
13:8	RW	0x01	PBL Programmable Burst Length These bits indicate the maximum number of beats to be transferred in one DMA transaction. This will be the maximum value that is used in a single block Read/Write. The DMA will always attempt to burst as specified in PBL each time it starts a Burst transfer on the host bus. PBL can be programmed with permissible values of 1, 2, 4, 8, 16, and 32. Any other value will result in undefined behavior. When USP is set high, this PBL value is applicable for TxDMA transactions only. The PBL values have the following limitations. The maximum number of beats (PBL) possible is limited by the size of the Tx FIFO and Rx FIFO in the MTL layer and the data bus width on the DMA. The FIFO has a constraint that the maximum beat supported is half the depth of the FIFO, except when specified (as given below). For different data bus widths and FIFO sizes, the valid PBL range (including x8 mode) is provided in the following table. If the PBL is common for both transmit and receive DMA, the minimum Rx FIFO and Tx FIFO depths must be considered. Do not program out-of-range PBL values, because the system may not behave properly. For TxFIFO, valid PBL range in full duplex mode and duplex mode is 128 or less. For RxFIFO, valid PBL range in full duplex mode is all
7	RO	0x0	reserved
6:2	RW	0x00	DSL Descriptor Skip Length This bit specifies the number of dword to skip between two unchained descriptors. The address skipping starts from the end of current descriptor to the start of next descriptor. When DSL value equals zero, then the descriptor table is taken as contiguous by the DMA, in Ring mode
1	RO	0x0	reserved
0	R/WSC	0x1	SWR Software Reset When this bit is set, the MAC DMA Controller resets all GMAC Subsystem internal registers and logic. It is cleared automatically after the reset operation has completed in all of the core clock domains. Read a 0 value in this bit before re-programming any register of the core. Note: The reset operation is completed only when all the resets in all the active clock domains are de-asserted. Hence it is essential that all the PHY inputs clocks (applicable for the selected PHY interface) are present for software reset completion

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

GMAC_TX_POLL_DEMAND

Address: Operational Base + offset (0x1004)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	TPD Transmit Poll Demand When these bits are written with any value, the DMA reads the current descriptor pointed to by Register GMAC_CUR_HOST_TX_DESC. If that descriptor is not available (owned by Host), transmission returns to the Suspend state and DMA Register GMAC_STATUS[2] is asserted. If the descriptor is available, transmission resumes

GMAC_RX_POLL_DEMAND

Address: Operational Base + offset (0x1008)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	RPD Receive Poll Demand When these bits are written with any value, the DMA reads the current descriptor pointed to by Register GMAC_CUR_HOST_RX_DESC. If that descriptor is not available (owned by Host), reception returns to the Suspended state and Register GMAC_STATUS[7] is not asserted. If the descriptor is available, the Receive DMA returns to active state

GMAC_RX_DESC_LIST_ADDR

Address: Operational Base + offset (0x100c)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	SRL Start of Receive List This field contains the base address of the First Descriptor in the Receive Descriptor list. The LSB bits [1/2/3:0] for 32/64/128-bit bus width) will be ignored and taken as all-zero by the DMA internally. Hence these LSB bits are Read Only

GMAC_TX_DESC_LIST_ADDR

Address: Operational Base + offset (0x1010)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	STL Start of Transmit List This field contains the base address of the First Descriptor in the Transmit Descriptor list. The LSB bits [1/2/3:0] for 32/64/128-bit bus width) will be ignored and taken as all-zero by the DMA internally. Hence these LSB bits are Read Only

GMAC_STATUS

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x1014)

Bit	Attr	Reset Value	Description
31:29	RO	0x0	reserved
28	RO	0x0	GPI GMAC PMT Interrupt This bit indicates an interrupt event in the GMAC core's PMT module. The software must read the corresponding registers in the GMAC core to get the exact cause of interrupt and clear its source to reset this bit to 1'b0. The interrupt signal from the GMAC subsystem (sbd_intr_o) is high when this bit is high
27	RO	0x0	GMI GMAC MMC Interrupt This bit reflects an interrupt event in the MMC module of the GMAC core. The software must read the corresponding registers in the GMAC core to get the exact cause of interrupt and clear the source of interrupt to make this bit as 1'b0. The interrupt signal from the GMAC subsystem (sbd_intr_o) is high when this bit is high
26	RO	0x0	GLI GMAC Line interface Interrupt This bit reflects an interrupt event in the GMAC Core's PCS or RGMII interface block. The software must read the corresponding registers in the GMAC core to get the exact cause of interrupt and clear the source of interrupt to make this bit as 1'b0. The interrupt signal from the GMAC subsystem (sbd_intr_o) is high when this bit is high
25:23	RO	0x0	EB Error Bits These bits indicate the type of error that caused a Bus Error (e.g., error response on the AXI interface). Valid only with Fatal Bus Error bit (Register GMAC_STATUS[13]) set. This field does not generate an interrupt. Bit 23: 1'b1 Error during data transfer by TxDMA 1'b0 Error during data transfer by RxDMA Bit 24: 1'b1 Error during read transfer 1'b0 Error during write transfer Bit 25: 1'b1 Error during descriptor access 1'b0 Error during data buffer access

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
22:20	RO	0x0	TS Transmit Process State These bits indicate the Transmit DMA FSM state. This field does not generate an interrupt. 3'b000: Stopped; Reset or Stop Transmit Command issued. 3'b001: Running; Fetching Transmit Transfer Descriptor. 3'b010: Running; Waiting for status. 3'b011: Running; Reading Data from host memory buffer and queuing it to transmit buffer (Tx FIFO). 3'b100: TIME_STAMP write state. 3'b101: Reserved for future use. 3'b110: Suspended; Transmit Descriptor Unavailable or Transmit Buffer Underflow. 3'b111: Running; Closing Transmit Descriptor
19:17	RO	0x0	RS Receive Process State These bits indicate the Receive DMA FSM state. This field does not generate an interrupt. 3'b000: Stopped: Reset or Stop Receive Command issued. 3'b001: Running: Fetching Receive Transfer Descriptor. 3'b010: Reserved for future use. 3'b011: Running: Waiting for receive packet. 3'b100: Suspended: Receive Descriptor Unavailable. 3'b101: Running: Closing Receive Descriptor. 3'b110: TIME_STAMP write state. 3'b111: Running: Transferring the receive packet data from receive buffer to host memory
16	W1C	0x0	NIS Normal Interrupt Summary Normal Interrupt Summary bit value is the logical OR of the following when the corresponding interrupt bits are enabled in Register OP_MODE: Register GMAC_STATUS[0]: Transmit Interrupt Register GMAC_STATUS[2]: Transmit Buffer Unavailable Register GMAC_STATUS[6]: Receive Interrupt Register GMAC_STATUS[14]: Early Receive Interrupt Only unmasked bits affect the Normal Interrupt Summary bit. This is a sticky bit and must be cleared (by writing a 1 to this bit) each time a corresponding bit that causes NIS to be set is cleared

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15	W1C	0x0	AIS Abnormal Interrupt Summary Abnormal Interrupt Summary bit value is the logical OR of the following when the corresponding interrupt bits are enabled in Register OP_MODE: Register GMAC_STATUS[1]: Transmit Process Stopped Register GMAC_STATUS[3]: Transmit Jabber Timeout Register GMAC_STATUS[4]: Receive FIFO Overflow Register GMAC_STATUS[5]: Transmit Underflow Register GMAC_STATUS[7]: Receive Buffer Unavailable Register GMAC_STATUS[8]: Receive Process Stopped Register GMAC_STATUS[9]: Receive Watchdog Timeout Register GMAC_STATUS[10]: Early Transmit Interrupt Register GMAC_STATUS[13]: Fatal Bus Error Only unmasked bits affect the Abnormal Interrupt Summary bit. This is a sticky bit and must be cleared each time a corresponding bit that causes AIS to be set is cleared
14	W1C	0x0	ERI Early Receive Interrupt This bit indicates that the DMA had filled the first data buffer of the packet. Receive Interrupt Register GMAC_STATUS[6] automatically clears this bit
13	W1C	0x0	FBI Fatal Bus Error Interrupt This bit indicates that a bus error occurred, as detailed in [25:23]. When this bit is set, the corresponding DMA engine disables all its bus accesses
12:11	RO	0x0	reserved
10	W1C	0x0	ETI Early Transmit Interrupt This bit indicates that the frame to be transmitted was fully transferred to the MTL Transmit FIFO
9	W1C	0x0	RWT Receive Watchdog Timeout This bit is asserted when a frame with a length greater than 2,048 bytes is received
8	W1C	0x0	RPS Receive Process Stopped This bit is asserted when the Receive Process enters the Stopped state

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7	W1C	0x0	RU Receive Buffer Unavailable This bit indicates that the Next Descriptor in the Receive List is owned by the host and cannot be acquired by the DMA. Receive Process is suspended. To resume processing Receive descriptors, the host should change the ownership of the descriptor and issue a Receive Poll Demand command. If no Receive Poll Demand is issued, Receive Process resumes when the next recognized incoming frame is received. Register GMAC_STATUS[7] is set only when the previous Receive Descriptor was owned by the DMA
6	W1C	0x0	RI Receive Interrupt This bit indicates the completion of frame reception. Specific frame status information has been posted in the descriptor. Reception remains in the Running state
5	W1C	0x0	UNF Transmit Underflow This bit indicates that the Transmit Buffer had an Underflow during frame transmission. Transmission is suspended and an Underflow Error TDES0[1] is set
4	W1C	0x0	OVF Receive Overflow This bit indicates that the Receive Buffer had an Overflow during frame reception. If the partial frame is transferred to application, the overflow status is set in RDES0[11]
3	W1C	0x0	TJT Transmit Jabber Timeout This bit indicates that the Transmit Jabber Timer expired, meaning that the transmitter had been excessively active. The transmission process is aborted and placed in the Stopped state. This causes the Transmit Jabber Timeout TDES0[14] flag to assert
2	W1C	0x0	TU Transmit Buffer Unavailable This bit indicates that the Next Descriptor in the Transmit List is owned by the host and cannot be acquired by the DMA. Transmission is suspended. Bits[22:20] explain the Transmit Process state transitions. To resume processing transmit descriptors, the host should change the ownership of the bit of the descriptor and then issue a Transmit Poll Demand command
1	W1C	0x0	TPS Transmit Process Stopped This bit is set when the transmission is stopped

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
0	W1C	0x0	TI Transmit Interrupt This bit indicates that frame transmission is finished and TDES1[31] is set in the First Descriptor

GMAC_OP_MODE

Address: Operational Base + offset (0x1018)

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved
26	RW	0x0	DT Disable Dropping of TCP/IP Checksum Error Frames When this bit is set, the core does not drop frames that only have errors detected by the Receive Checksum Offload engine. Such frames do not have any errors (including FCS error) in the Ethernet frame received by the MAC but have errors in the encapsulated payload only. When this bit is reset, all error frames are dropped if the FEF bit is reset
25	RW	0x0	RSF Receive Store and Forward When this bit is set, the MTL only reads a frame from the Rx FIFO after the complete frame has been written to it, ignoring RTC bits. When this bit is reset, the Rx FIFO operates in Cut-Through mode, subject to the threshold specified by the RTC bits
24	RW	0x0	DFF Disable Flushing of Received Frames When this bit is set, the RxDMA does not flush any frames due to the unavailability of receive descriptors/buffers as it does normally when this bit is reset
23:22	RO	0x0	reserved
21	RW	0x0	TSF Transmit Store and Forward When this bit is set, transmission starts when a full frame resides in the MTL Transmit FIFO. When this bit is set, the TTC values specified in Register GMAC_OP_MODE[16:14] are ignored. This bit should be changed only when transmission is stopped

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
20	W1C	0x0	FTF Flush Transmit FIFO When this bit is set, the transmit FIFO controller logic is reset to its default values and thus all data in the Tx FIFO is lost/flushed. This bit is cleared internally when the flushing operation is completed fully. The Operation Mode register should not be written to until this bit is cleared. The data which is already accepted by the MAC transmitter will not be flushed. It will be scheduled for transmission and will result in underflow and runt frame transmission. Note: The flush operation completes only after emptying the TxFIFO of its contents and all the pending Transmit Status of the transmitted frames are accepted by the host. In order to complete this flush operation, the PHY transmit clock (clk_tx_i) is required to be active
19:17	RO	0x0	reserved
16:14	RW	0x0	TTC Transmit Threshold Control These three bits control the threshold level of the MTL Transmit FIFO. Transmission starts when the frame size within the MTL Transmit FIFO is larger than the threshold. In addition, full frames with a length less than the threshold are also transmitted. These bits are used only when the TSF bit (Bit 21) is reset. 3'b000: 64 3'b001: 128 3'b010: 192 3'b011: 256 3'b100: 40 3'b101: 32 3'b110: 24 3'b111: 16

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
13	RW	0x0	ST Start/Stop Transmission Command When this bit is set, transmission is placed in the Running state, and the DMA checks the Transmit List at the current position for a frame to be transmitted. Descriptor acquisition is attempted either from the current position in the list, which is the Transmit List Base Address set by Register GMAC_TX_DESC_LIST_ADDR, or from the position retained when transmission was stopped previously. If the current descriptor is not owned by the DMA, transmission enters the Suspended state and Transmit Buffer Unavailable (Register GMAC_STATUS[2]) is set. The Start Transmission command is effective only when transmission is stopped. If the command is issued before setting DMA Register TX_DESC_LIST_ADDR, then the DMA behavior is unpredictable. When this bit is reset, the transmission process is placed in the Stopped state after completing the transmission of the current frame. The Next Descriptor position in the Transmit List is saved, and becomes the current position when transmission is restarted. The stop transmission command is effective only the transmission of the current frame is complete or when the transmission is in the Suspended state
12:11	RW	0x0	RFD Threshold for deactivating flow control (in both HD and FD) These bits control the threshold (Fill-level of Rx FIFO) at which the flow-control is de-asserted after activation. 2'b00: Full minus 1 KB 2'b01: Full minus 2 KB 2'b10: Full minus 3 KB 2'b11: Full minus 4 KB Note that the de-assertion is effective only after flow control is asserted
10:9	RW	0x0	RFA Threshold for activating flow control (in both HD and FD) These bits control the threshold (Fill level of Rx FIFO) at which flow control is activated. 2'b00: Full minus 1 KB 2'b01: Full minus 2 KB 2'b10: Full minus 3 KB 2'b11: Full minus 4 KB Note that the above only applies to Rx FIFOs of 4 KB or more when the EFC bit is set high

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
8	RW	0x0	EFC Enable HW flow control When this bit is set, the flow control signal operation based on fill-level of Rx FIFO is enabled. When reset, the flow control operation is disabled
7	RW	0x0	FEF Forward Error Frames When this bit is reset, the Rx FIFO drops frames with error status (CRC error, collision error, GMII_ER, giant frame, watchdog timeout, overflow). However, if the frame's start byte (write) pointer is already transferred to the read controller side (in Threshold mode), then the frames are not dropped. When FEF is set, all frames except runt error frames are forwarded to the DMA. But when RxFIFO overflows when a partial frame is written, then such frames are dropped even when FEF is set
6	RW	0x0	FUF Forward Undersized Good Frames When set, the Rx FIFO will forward Undersized frames (frames with no Error and length less than 64 bytes) including pad-bytes and CRC). When reset, the Rx FIFO will drop all frames of less than 64 bytes, unless it is already transferred due to lower value of Receive Threshold (e.g., RTC = 01)
5	RO	0x0	reserved
4:3	RW	0x0	RTC Receive Threshold Control These two bits control the threshold level of the MTL Receive FIFO. Transfer (request) to DMA starts when the frame size within the MTL Receive FIFO is larger than the threshold. In addition, full frames with a length less than the threshold are transferred automatically. Note that value of 11 is not applicable if the configured Receive FIFO size is 128 bytes. These bits are valid only when the RSF bit is zero, and are ignored when the RSF bit is set to 1. 2'b00: 64 2'b01: 32 2'b10: 96 2'b11: 128
2	RW	0x0	OSF Operate on Second Frame When this bit is set, this bit instructs the DMA to process a second frame of Transmit data even before status for first frame is obtained

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
1	RW	0x0	SR Start/Stop Receive When this bit is set, the Receive process is placed in the Running state. The DMA attempts to acquire the descriptor from the Receive list and processes incoming frames. Descriptor acquisition is attempted from the current position in the list, which is the address set by register GMAC_RX_DESC_LIST_ADDR or the position retained when the Receive process was previously stopped. If no descriptor is owned by the DMA, reception is suspended and Receive Buffer Unavailable (Register GMAC_STATUS[7]) is set. The Start Receive command is effective only when reception has stopped. If the command was issued before setting register GMAC_RX_DESC_LIST_ADDR, DMA behavior is unpredictable. When this bit is cleared, RxDMA operation is stopped after the transfer of the current frame. The next descriptor position in the Receive list is saved and becomes the current position after the Receive process is restarted. The Stop Receive command is effective only when the Receive process is in either the Running (waiting for receive packet) or in the Suspended state
0	RO	0x0	reserved

GMAC_INT_ENA

Address: Operational Base + offset (0x101c)

Bit	Attr	Reset Value	Description
31:17	RO	0x0	reserved
16	RW	0x0	NIE Normal Interrupt Summary Enable When this bit is set, a normal interrupt is enabled. When this bit is reset, a normal interrupt is disabled. This bit enables the following bits: Register GMAC_STATUS[0]: Transmit Interrupt Register GMAC_STATUS[2]: Transmit Buffer Unavailable Register GMAC_STATUS[6]: Receive Interrupt Register GMAC_STATUS[14]: Early Receive Interrupt

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15	RW	0x0	AIE Abnormal Interrupt Summary Enable When this bit is set, an Abnormal Interrupt is enabled. When this bit is reset, an Abnormal Interrupt is disabled. This bit enables the following bits Register GMAC_STATUS[1]: Transmit Process Stopped Register GMAC_STATUS[3]: Transmit Jabber Timeout Register GMAC_STATUS[4]: Receive Overflow Register GMAC_STATUS[5]: Transmit Underflow Register GMAC_STATUS[7]: Receive Buffer Unavailable Register GMAC_STATUS[8]: Receive Process Stopped Register GMAC_STATUS[9]: Receive Watchdog Timeout Register GMAC_STATUS[10]: Early Transmit Interrupt Register GMAC_STATUS[13]: Fatal Bus Error
14	RW	0x0	ERE Early Receive Interrupt Enable When this bit is set with Normal Interrupt Summary Enable (BIT 16), Early Receive Interrupt is enabled. When this bit is reset, Early Receive Interrupt is disabled
13	RW	0x0	FBE Fatal Bus Error Enable When this bit is set with Abnormal Interrupt Summary Enable (BIT 15), the Fatal Bus Error Interrupt is enabled. When this bit is reset, Fatal Bus Error Enable Interrupt is disabled
12:11	RO	0x0	reserved
10	RW	0x0	ETE Early Transmit Interrupt Enable When this bit is set with an Abnormal Interrupt Summary Enable (BIT 15), Early Transmit Interrupt is enabled. When this bit is reset, Early Transmit Interrupt is disabled
9	RW	0x0	RWE Receive Watchdog Timeout Enable When this bit is set with Abnormal Interrupt Summary Enable (BIT 15), the Receive Watchdog Timeout Interrupt is enabled. When this bit is reset, Receive Watchdog Timeout Interrupt is disabled
8	RW	0x0	RSE Receive Stopped Enable When this bit is set with Abnormal Interrupt Summary Enable (BIT 15), Receive Stopped Interrupt is enabled. When this bit is reset, Receive Stopped Interrupt is disabled

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7	RW	0x0	RUE Receive Buffer Unavailable Enable When this bit is set with Abnormal Interrupt Summary Enable (BIT 15), Receive Buffer Unavailable Interrupt is enabled. When this bit is reset, the Receive Buffer Unavailable Interrupt is disabled
6	RW	0x0	RIE Receive Interrupt Enable When this bit is set with Normal Interrupt Summary Enable (BIT 16), Receive Interrupt is enabled. When this bit is reset, Receive Interrupt is disabled
5	RW	0x0	UNE Underflow Interrupt Enable When this bit is set with Abnormal Interrupt Summary Enable (BIT 15), Transmit Underflow Interrupt is enabled. When this bit is reset, Underflow Interrupt is disabled
4	RW	0x0	OVE Overflow Interrupt Enable When this bit is set with Abnormal Interrupt Summary Enable (BIT 15), Receive Overflow Interrupt is enabled. When this bit is reset, Overflow Interrupt is disabled
3	RW	0x0	TJE Transmit Jabber Timeout Enable When this bit is set with Abnormal Interrupt Summary Enable (BIT 15), Transmit Jabber Timeout Interrupt is enabled. When this bit is reset, Transmit Jabber Timeout Interrupt is disabled
2	RW	0x0	TUE Transmit Buffer Unavailable Enable When this bit is set with Normal Interrupt Summary Enable (BIT 16), Transmit Buffer Unavailable Interrupt is enabled. When this bit is reset, Transmit Buffer Unavailable Interrupt is disabled
1	RW	0x0	TSE Transmit Stopped Enable When this bit is set with Abnormal Interrupt Summary Enable (BIT 15), Transmission Stopped Interrupt is enabled. When this bit is reset, Transmission Stopped Interrupt is disabled
0	RW	0x0	TIE Transmit Interrupt Enable When this bit is set with Normal Interrupt Summary Enable (BIT 16), Transmit Interrupt is enabled. When this bit is reset, Transmit Interrupt is disabled

GMAC_OVERFLOW_CNT

Address: Operational Base + offset (0x1020)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:29	RO	0x0	reserved
28	RC	0x0	FIFO_overflow_bit Overflow bit for FIFO Overflow Counter
27:17	RC	0x000	Frame_miss_number Indicates the number of frames missed by the application This counter is incremented each time the MTL asserts the sideband signal mtl_rxoverflow_o. The counter is cleared when this register is read with mci_be_i[2] at 1'b1
16	RC	0x0	Miss_frame_overflow_bit Overflow bit for Missed Frame Counter
15:0	RC	0x0000	Frame_miss_number_2 Indicates the number of frames missed by the controller due to the Host Receive Buffer being unavailable. This counter is incremented each time the DMA discards an incoming frame. The counter is cleared when this register is read with mci_be_i[0] at 1'b1

GMAC_REC_INT_WDT_TIMER

Address: Operational Base + offset (0x1024)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	RIWT RI Watchdog Timer count Indicates the number of system clock cycles multiplied by 256 for which the watchdog timer is set. The watchdog timer gets triggered with the programmed value after the RxDMA completes the transfer of a frame for which the RI status bit is not set due to the setting in the corresponding descriptor RDES1[31]. When the watch-dog timer runs out, the RI bit is set and the timer is stopped. The watchdog timer is reset when RI bit is set high due to automatic setting of RI as per RDES1[31] of any received frame

GMAC_AXI_BUS_MODE

Address: Operational Base + offset (0x1028)

Bit	Attr	Reset Value	Description
31	RW	0x0	EN_LPI Enable LPI (Low Power Interface) When set to 1, enable the LPI (Low Power Interface) supported by the GMAC and accepts the LPI request from the AXI System Clock controller. When set to 0, disables the Low Power Mode and always denies the LPI request from the AXI System Clock controller

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
30	RW	0x0	UNLCK_ON_MGK_RWK Unlock on Magic Packet or Remote Wake Up When set to 1, enables it to request coming out of Low Power mode only when Magic Packet or Remote Wake Up Packet is received. When set to 0, enables it requests to come out of Low Power mode when any frame is received
29:22	RO	0x0	reserved
21:20	RW	0x1	WR_OSR_LMT AXI Maximum Write Out Standing Request Limit This value limits the maximum outstanding request on the AXI write interface. Maximum outstanding requests = WR_OSR_LMT+1
19:18	RO	0x0	reserved
17:16	RW	0x1	RD_OSR_LMT AXI Maximum Read Out Standing Request Limit This value limits the maximum outstanding request on the AXI read interface. Maximum outstanding requests = RD_OSR_LMT+1
15:13	RO	0x0	reserved
12	RO	0x0	AXI_AAL Address-Aligned Beats This bit is read-only bit and reflects the AAL bit Register0 (register GMAC_BUS_MODE[25]). When this bit set to 1, it performs address-aligned burst transfers on both read and write channels
11:4	RO	0x0	reserved
3	RW	0x0	BLEN16 AXI Burst Length 16 When this bit is set to 1, or when UNDEF is set to 1, it is allowed to select a burst length of 16
2	RW	0x0	BLEN8 AXI Burst Length 8 When this bit is set to 1, or when UNDEF is set to 1, it is allowed to select a burst length of 8
1	RW	0x0	BLEN4 AXI Burst Length 4 When this bit is set to 1, or when UNDEF is set to 1, it is allowed to select a burst length of 4

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
0	RO	0x1	UNDEF AXI Undefined Burst Length This bit is read-only bit and indicates the complement (invert) value of FB bit in register GMAC_BUS_MODE[16]. When this bit is set to 1, it is allowed to perform any burst length equal to or below the maximum allowed burst length as programmed in bits[7:1]; When this bit is set to 0, it is allowed to perform only fixed burst lengths as indicated by BLEN256/128/64/32/16/8/4, or a burst length of 1

GMAC_AXI_STATUS

Address: Operational Base + offset (0x102c)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RO	0x0	RD_CH_STA When high, it indicates that AXI Master's read channel is active and transferring data
0	RO	0x0	WR_CH_STA When high, it indicates that AXI Master's write channel is active and transferring data

GMAC_CUR_HOST_TX_DESC

Address: Operational Base + offset (0x1048)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	HTDAP Host Transmit Descriptor Address Pointer Cleared on Reset. Pointer updated by DMA during operation

GMAC_CUR_HOST_RX_DESC

Address: Operational Base + offset (0x104c)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	HRDAP Host Receive Descriptor Address Pointer Cleared on Reset. Pointer updated by DMA during operation

GMAC_CUR_HOST_TX_BUF_ADDR

Address: Operational Base + offset (0x1050)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	HTBAP Host Transmit Buffer Address Pointer Cleared on Reset. Pointer updated by DMA during operation

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

GMAC_CUR_HOST_RX_BUF_ADDR

Address: Operational Base + offset (0x1054)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	HRBAP Host Receive Buffer Address Pointer Cleared on Reset. Pointer updated by DMA during operation

17.5 Interface Description

Table 17-1 RGMII Interface Description

Module pin	Direction	Pad name	IOMUX setting
RMII interface			
mac_clk	I/O	GPIO2_B7/CIF_CLKOUT/RGMII_CLK	GRF_GPIO2B_IOMUX_SEL_H=[14:12]=3'b2
mac_txen	O	GPIO2_A1/CIF_D13/RGMII_TXEN/LCDC_D7	GRF_GPIO2A_IOMUX_SEL_L=[6:4]=3'b2
mac_txd3	O	GPIO2_B3/CIF_D9/RGMII_TXD3/LCDC_VSYNC_M0	GRF_GPIO2B_IOMUX_SEL_L=[14:12]=3'b2
mac_txd2	O	GPIO2_B4/CIF_VSYNC/RGMII_TXD2	GRF_GPIO2B_IOMUX_SEL_H=[2:0]=3'b2
mac_txd1	O	GPIO2_A2/CIF_D14/RGMII_TXD1/LCDC_D0	GRF_GPIO2A_IOMUX_SEL_L=[10:8]=3'b2
mac_txd0	O	GPIO2_A3/CIF_D15/RGMII_TXD0/LCDC_D1	GRF_GPIO2A_IOMUX_SEL_L=[14:12]=3'b2
mac_txclk	O	GPIO2_C1/CIF_D1/RGMII_TXCLK	GRF_GPIO2C_IOMUX_SEL_L=[6:4]=3'b2
mac_rxdv	I	GPIO2_A7/CIF_D5/RGMII_RXDV/SPI2_CSN_M1	GRF_GPIO2A_IOMUX_SEL_H=[14:12]=3'b2
mac_rxer	I	GPIO2_A6/CIF_D4/RGMII_RXER/SPI2_MOSI_M1	GRF_GPIO2A_IOMUX_SEL_H=[10:8]=3'b2
mac_rxd3	I	GPIO2_B6/CIF_CLKIN/RGMII_RXD3	GRF_GPIO2B_IOMUX_SEL_H=[10:8]=3'b2
mac_rxd2	I	GPIO2_B5/CIF_HREF/RGMII_RXD2	GRF_GPIO2B_IOMUX_SEL_H=[6:4]=3'b2
mac_rxd1	I	GPIO2_A5/CIF_D3/RGMII_RXD1/SPI2_CLK_M1	GRF_GPIO2A_IOMUX_SEL_H=[6:4]=3'b2
mac_rxd0	I	GPIO2_A4/CIF_D2/RGMII_RXD0/SPI2_MISO_M1	GRF_GPIO2A_IOMUX_SEL_H=[2:0]=3'b2
mac_rxclk	I	GPIO2_C2/CIF_D10/RGMII_RXCLK/LCDC_D2	GRF_GPIO2C_IOMUX_SEL_L=[10:8]=3'b2
mac_crs	I	GPIO2_A0/CIF_D12/RGMII_CRS/LCDC_D6	GRF_GPIO2A_IOMUX_SEL_L=[2:0]=3'b2
mac_col	I	GPIO2_B1/CIF_D7/RGMII_COL	GRF_GPIO2B_IOMUX_SEL_L=[6:4]=3'b2
Management interface			
mac_mdio	I/O	GPIO2_B0/CIF_D6/RGMII_MDIO	GRF_GPIO2B_IOMUX_SEL_L=[2:0]=3'b2
mac_mdc	O	GPIO2_B2/CIF_D8/RGMII_MDC/LCDC_HSYNC_M0	GRF_GPIO2B_IOMUX_SEL_L=[10:8]=3'b2

Notes: I=input, O=output, I/O=input/output, bidirectional

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

17.6 Application Notes

17.6.1 Descriptors

The DMA in GMAC can communicate with Host driver through descriptor lists and data buffers. The DMA transfers data frames received by the core to the Receive Buffer in the Host memory, and Transmit data frames from the Transmit Buffer in the Host memory. Descriptors that reside in the Host memory act as pointers to these buffers.
There are two descriptor lists; one for reception, and one for transmission. The base address of each list is written into DMA Registers RX_DESC_LIST_ADDR and TX_DESC_LIST_ADDR, respectively. A descriptor list is forward linked (either implicitly or explicitly). The last descriptor may point back to the first entry to create a ring structure. Explicit chaining of descriptors is accomplished by setting the second address chained in both Receive and Transmit descriptors (RDES1[24] and TDES1[24]). The descriptor lists resides in the Host physical memory address space. Each descriptor can point to a maximum of two buffers. This enables two buffers to be used, physically addressed, rather than contiguous buffers in memory.
A data buffer resides in the Host physical memory space, and consists of an entire frame or part of a frame, but cannot exceed a single frame. Buffers contain only data, buffer status is maintained in the descriptor. Data chaining refers to frames that span multiple data buffers. However, a single descriptor cannot span multiple frames. The DMA will skip to the next frame buffer when end-of-frame is detected. Data chaining can be enabled or disabled
The descriptor ring and chain structure is shown in following figure.

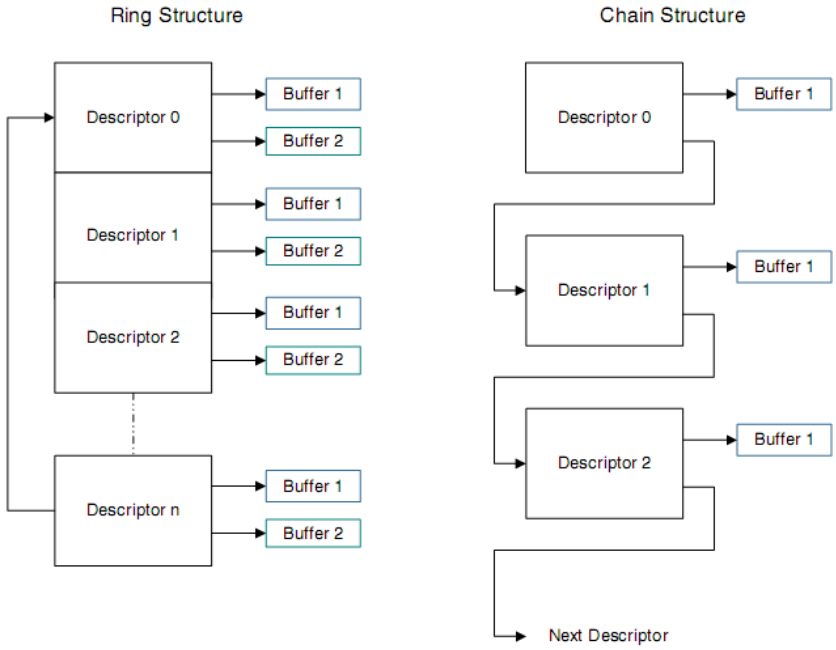


Fig. 17-8 Descriptor Ring and Chain Structure

Each descriptor contains two buffers, two byte-count buffers, and two address pointers, which enable the adapter port to be compatible with various types of memory management schemes. The descriptor addresses must be aligned to the bus width used (Word/Dword/Lword for 32/64/128-bit buses).

	63 55	47	39	31	23 15 7 0
DES1-DES0	Control Bits [9:0]	Byte Count Buffer2 [10:0]	Byte Count Buffer1[10:0]	OWN	Status [30:0]
DES3-DES2	Buffer2 Address [31:0] / Next Descriptor Address [31:0]				Buffer1 Address[31:0]

Fig. 17-9 Rx/Tx Descriptors definition

17.6.2 Receive Descriptor

The GMAC Subsystem requires at least two descriptors when receiving a frame. The Receive state machine of the DMAalways attempts to acquire an extra descriptor in

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

anticipation of an incoming frame. (The size of the incoming frame is unknown). Before the RxDMA closes a descriptor, it will attempt to acquire the next descriptor even if no frames are received.
In a single descriptor (receive) system, the subsystem will generate a descriptor error if the receive buffer is unable to accommodate the incoming frame and the next descriptor is not owned by the DMA. Thus, the Host is forced to increase either its descriptor pool or the buffer size. Otherwise, the subsystem starts dropping all incoming frames.

Receive Descriptor 0 (RDES0)

RDES0 contains the received frame status, the frame length, and the descriptor ownership information.

Table 17-2 Receive Descriptor 0

Bit	Description
31	OWN: Own Bit When set, this bit indicates that the descriptor is owned by the DMA of the GMAC Subsystem. When this bit is reset, this bit indicates that the descriptor is owned by the Host. The DMA clears this bit either when it completes the frame reception or when the buffers that are associated with this descriptor are full.
30	AFM: Destination Address Filter Fail When set, this bit indicates a frame that failed in the DA Filter in the GMAC Core.
29:16	FL: Frame Length These bits indicate the byte length of the received frame that was transferred to host memory (including CRC). This field is valid when Last Descriptor (RDES0[8]) is set and either the Descriptor Error (RDES0[14]) or Overflow Error bits are reset. The frame length also includes the two bytes appended to the Ethernet frame when IP checksum calculation (Type 1) is enabled and the received frame is not a MAC control frame. This field is valid when Last Descriptor (RDES0[8]) is set. When the Last Descriptor and Error Summary bits are not set, this field indicates the accumulated number of bytes that have been transferred for the current frame.
15	ES: Error Summary Indicates the logical OR of the following bits: • RDES0[0]: Payload Checksum Error • RDES0[1]: CRC Error • RDES0[3]: Receive Error • RDES0[4]: Watchdog Timeout • RDES0[6]: Late Collision • RDES0[7]: IPC Checksum • RDES0[11]: Overflow Error • RDES0[14]: Descriptor Error This field is valid only when the Last Descriptor (RDES0[8]) is set.
14	DE: Descriptor Error When set, this bit indicates a frame truncation caused by a frame that does not fit within the current descriptor buffers, and that the DMA does not own the Next Descriptor. The frame is truncated. This field is valid only when the Last Descriptor (RDES0[8]) is set
13	SAF: Source Address Filter Fail When set, this bit indicates that the SA field of frame failed the SA Filter in the GMAC Core.
12	LE: Length Error When set, this bit indicates that the actual length of the frame received and that the Length/ Type field does not match. This bit is valid only when the Frame Type (RDES0[5]) bit is reset. Length error status is not valid when CRC error is present.
11	OE: Overflow Error When set, this bit indicates that the received frame was damaged due to buffer overflow.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Description
10	VLAN: VLAN Tag When set, this bit indicates that the frame pointed to by this descriptor is a VLAN frame tagged by the GMAC Core.
9	FS: First Descriptor When set, this bit indicates that this descriptor contains the first buffer of the frame. If the size of the first buffer is 0, the second buffer contains the beginning of the frame. If the size of the second buffer is also 0, the next Descriptor contains the beginning of the frame.
8	LS: Last Descriptor When set, this bit indicates that the buffers pointed to by this descriptor are the last buffers of the frame.
7	IPC Checksum Error/Giant Frame When IP Checksum Engine is enabled, this bit, when set, indicates that the 16-bit IPv4 Header checksum calculated by the core did not match the received checksum bytes. The Error Summary bit[15] is NOT set when this bit is set in this mode.
6	LC: Late Collision When set, this bit indicates that a late collision has occurred while receiving the frame in Half-Duplex mode.
5	FT: Frame Type When set, this bit indicates that the Receive Frame is an Ethernet-type frame (the LT field is greater than or equal to 16'h0600). When this bit is reset, it indicates that the received frame is an IEEE802.3 frame. This bit is not valid for Runt frames less than 14 bytes.
4	RWT: Receive Watchdog Timeout When set, this bit indicates that the Receive Watchdog Timer has expired while receiving the current frame and the current frame is truncated after the Watchdog Timeout.
3	RE: Receive Error When set, this bit indicates that the gmii_rxer_i signal is asserted while gmii_rxdv_i is asserted during frame reception. This error also includes carrier extension error in GMII and Half-duplex mode. Error can be of less/no extension, or error (rxd≠ 0f) during extension.
2	DE: Dribble Bit Error When set, this bit indicates that the received frame has a non-integer multiple of bytes (odd nibbles). This bit is valid only in MII Mode.
1	CE: CRC Error When set, this bit indicates that a Cyclic Redundancy Check (CRC) Error occurred on the received frame. This field is valid only when the Last Descriptor (RDES0[8]) is set.
0	Rx MAC Address/Payload Checksum Error When set, this bit indicates that the Rx MAC Address registers value (1 to 15) matched the frame's DA field. When reset, this bit indicates that the Rx MAC Address Register 0 value matched the DA field. If Full Checksum Offload Engine is enabled, this bit, when set, indicates the TCP, UDP, or ICMP checksum the core calculated does not match the received encapsulated TCP, UDP, or ICMP segment's Checksum field. This bit is also set when the received number of payload bytes does not match the value indicated in the Length field of the encapsulated IPv4 or IPv6 datagram in the received Ethernet frame.

Receive Descriptor 1 (RDES1)

RDES1 contains the buffer sizes and other bits that control the descriptor chain/ring.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Table 17-3 Receive Descriptor 1

Bit	Description
31	Disable Interrupt on Completion When set, this bit will prevent the setting of the RI (CSR5[6]) bit of the GMAC_STATUS Register for the received frame that ends in the buffer pointed to by this descriptor. This, in turn, will disable the assertion of the interrupt to Host due to RI for that frame.
30:26	Reserved.
25	RER: Receive End of Ring When set, this bit indicates that the descriptor list reached its final descriptor. The DMA returns to the base address of the list, creating a Descriptor Ring.
24	RCH: Second Address Chained When set, this bit indicates that the second address in the descriptor is the Next Descriptor address rather than the second buffer address. When RDES1[24] is set, RBS2 (RDES1[21-11]) is a "don't care" value. RDES1[25] takes precedence over RDES1[24].
23:22	Reserved.
21:11	RBS2: Receive Buffer 2 Size These bits indicate the second data buffer size in bytes. The buffer size must be a multiple of 8 depending upon the bus widths (64), even if the value of RDES3 (buffer2 address pointer) is not aligned to bus width. In the case where the buffer size is not a multiple of 8, the resulting behavior is undefined. This field is not valid if RDES1[24] is set.
10:0	RBS1: Receive Buffer 1 Size Indicates the first data buffer size in bytes. The buffer size must be a multiple of 8 depending upon the bus widths (64), even if the value of RDES2 (buffer1 address pointer) is not aligned. In the case where the buffer size is not a multiple of 8, the resulting behavior is undefined. If this field is 0, the DMA ignores this buffer and uses Buffer 2 or next descriptor depending on the value of RCH (Bit 24).

Receive Descriptor 2 (RDES2)

RDES2 contains the address pointer to the first data buffer in the descriptor.

Table 17-4 Receive Descriptor 2

Bit	Description
31:0	Buffer 1 Address Pointer These bits indicate the physical address of Buffer 1. There are no limitations on the buffer address alignment except for the following condition: The DMA uses the configured value for its address generation when the RDES2 value is used to store the start of frame. Note that the DMA performs a write operation with the RDES2[2:0] bits as 0 during the transfer of the start of frame but the frame data is shifted as per the actual Buffer address pointer. The DMA ignores RDES2[2:0] (corresponding to bus width of 64) if the address pointer is to a buffer where the middle or last part of the frame is stored.

Receive Descriptor 3 (RDES3)

RDES3 contains the address pointer either to the second data buffer in the descriptor or to the next descriptor.

Table 17-5 Receive Descriptor 3

Bit	Description
31:0	Buffer 2 Address Pointer (Next Descriptor Address) These bits indicate the physical address of Buffer 2 when a descriptor ring structure is used. If the Second Address Chained (RDES1[24]) bit is set, this address contains the pointer to the physical memory where the Next Descriptor is present.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

	If RDES1[24] is set, the buffer (Next Descriptor) address pointer must be bus width-aligned (RDES3[2:0] = 0, corresponding to a bus width of 64. LSBs are ignored internally.) However, when RDES1[24] is reset, there are no limitations on the RDES3 value, except for the following condition: The DMA uses the configured value for its buffer address generation when the RDES3 value is used to store the start of frame. The DMA ignores RDES3[2:0] (corresponding to a bus width of 64) if the address pointer is to a buffer where the middle or last part of the frame is stored.

17.6.3 Transmit Descriptor

The descriptor addresses must be aligned to the bus width used (64). Each descriptor is provided with two buffers, two byte-count buffers, and two address pointers, which enable the adapter port to be compatible with various types of memory-management schemes.

Transmit Descriptor 0 (TDES0)

TDES0 contains the transmitted frame status and the descriptor ownership information.

Table 17-6 Transmit Descriptor 0

Bit	Description
31	OWN: Own Bit When set, this bit indicates that the descriptor is owned by the DMA. When this bit is reset, this bit indicates that the descriptor is owned by the Host. The DMA clears this bit either when it completes the frame transmission or when the buffers allocated in the descriptor are empty. The ownership bit of the First Descriptor of the frame should be set after all subsequent descriptors belonging to the same frame have been set. This avoids a possible race condition between fetching a descriptor and the driver setting an ownership bit.
30:17	Reserved.
16	IHE: IP Header Error When set, this bit indicates that the Checksum Offload engine detected an IP header error and consequently did not modify the transmitted frame for any checksum insertion.
15	ES: Error Summary Indicates the logical OR of the following bits: • TDES0[14]: Jabber Timeout • TDES0[13]: Frame Flush • TDES0[11]: Loss of Carrier • TDES0[10]: No Carrier • TDES0[9]: Late Collision • TDES0[8]: Excessive Collision • TDES0[2]: Excessive Deferral • TDES0[1]: Underflow Error
14	JT: Jabber Timeout When set, this bit indicates the GMAC transmitter has experienced a jabber time-out.
13	FF: Frame Flushed When set, this bit indicates that the DMA/MTL flushed the frame due to a SW flush command given by the CPU.
12	PCE: Payload Checksum Error This bit, when set, indicates that the Checksum Offload engine had a failure and did not insert any checksum into the encapsulated TCP, UDP, or ICMP payload. This failure can be either due to insufficient bytes, as indicated by the IP Header's Payload Length field, or the MTL starting to forward the frame to the MAC transmitter in Store-and-Forward mode without the checksum having been calculated yet. This second error condition only occurs when the Transmit FIFO depth is less than the length of the Ethernet frame being transmitted: to avoid deadlock, the MTL starts forwarding the frame when the FIFO is full, even in

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Description
	Store-and-Forward mode.
11	LC: Loss of Carrier When set, this bit indicates that Loss of Carrier occurred during frame transmission. This is valid only for the frames transmitted without collision and when the GMAC operates in Half-Duplex Mode.
10	NC: No Carrier When set, this bit indicates that the carrier sense signal form the PHY was not asserted during transmission.
9	LC: Late Collision When set, this bit indicates that frame transmission was aborted due to a collision occurring after the collision window (64 byte times including Preamble in RMII Mode and 512 byte times including Preamble and Carrier Extension in RGMII Mode). Not valid if Underflow Error is set.
8	EC: Excessive Collision When set, this bit indicates that the transmission was aborted after 16 successive collisions while attempting to transmit the current frame. If the DR (Disable Retry) bit in the GMAC Configuration Register is set, this bit is set after the first collision and the transmission of the frame is aborted.
7	VF: VLAN Frame When set, this bit indicates that the transmitted frame was a VLAN-type frame.
6:3	CC: Collision Count This 4-bit counter value indicates the number of collisions occurring before the frame was transmitted. The count is not valid when the Excessive Collisions bit (TDES0[8]) is set.
2	ED: Excessive Deferral When set, this bit indicates that the transmission has ended because of excessive deferral of over 24,288 bit times (155,680 bits times in 1000-Mbps mode) if the Deferral Check (DC) bit is set high in the GMAC Control Register.
1	UF: Underflow Error When set, this bit indicates that the GMAC aborted the frame because data arrived late from the Host memory. Underflow Error indicates that the DMA encountered an empty Transmit Buffer while transmitting the frame. The transmission process enters the suspended state and sets both Transmit Underflow (Register GMAC_STATUS[5]) and Transmit Interrupt (Register GMAC_STATUS [0]).
0	DB: Deferred Bit When set, this bit indicates that the GMAC defers before transmission because of the presence of carrier. This bit is valid only in Half-Duplex mode.

Transmit Descriptor 1 (TDES1)

TDES1 contains the buffer sizes and other bits which control the descriptor chain/ring and the frame being transferred.

Table 17-7 Transmit Descriptor 1

Bit	Description
31	IC: Interrupt on Completion When set, this bit sets Transmit Interrupt (Register 5[0]) after the present frame has been transmitted.
30	LS: Last Segment When set, this bit indicates that the buffer contains the last segment of the frame.
29	FS: First Segment When set, this bit indicates that the buffer contains the first segment of a frame.
28:27	CIC: Checksum Insertion Control These bits control the insertion of checksums in Ethernet frames that encapsulate

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Description
	TCP, UDP, or ICMP over IPv4 or IPv6 as described below. • 2'b00: Do nothing. Checksum Engine is bypassed • 2'b01: Insert IPv4 header checksum. Use this value to insert IPv4 header checksum when the frame encapsulates an IPv4 datagram. • 2'b10: Insert TCP/UDP/ICMP checksum. The checksum is calculated over the TCP, UDP, or ICMP segment only and the TCP, UDP, or ICMP pseudo-header checksum is assumed to be present in the corresponding input frame's Checksum field. An IPv4 header checksum is also inserted if the encapsulated datagram conforms to IPv4. • 2'b11: Insert a TCP/UDP/ICMP checksum that is fully calculated in this engine. In other words, the TCP, UDP, or ICMP pseudo-header is included in the checksum calculation, and the input frame's corresponding Checksum field has an all-zero value. An IPv4 Header checksum is also inserted if the encapsulated datagram conforms to IPv4. The Checksum engine detects whether the TCP, UDP, or ICMP segment is encapsulated in IPv4 or IPv6 and processes its data accordingly.
26	DC: Disable CRC When set, the GMAC does not append the Cyclic Redundancy Check (CRC) to the end of the transmitted frame. This is valid only when the first segment (TDES1[29]).
25	TER: Transmit End of Ring When set, this bit indicates that the descriptor list reached its final descriptor. The returns to the base address of the list, creating a descriptor ring.
24	TCH: Second Address Chained When set, this bit indicates that the second address in the descriptor is the Next Descriptor address rather than the second buffer address. When TDES1[24] is set, TBS2 (TDES1[21–11]) are "don't care" values. TDES1[25] takes precedence over TDES1[24].
23	DP: Disable Padding When set, the GMAC does not automatically add padding to a frame shorter than 64 bytes. When this bit is reset, the DMA automatically adds padding and CRC to a frame shorter than 64 bytes and the CRC field is added despite the state of the DC (TDES1[26]) bit. This is valid only when the first segment (TDES1[29]) is set.
22	Reserved.
21:11	TBS2: Transmit Buffer 2 Size These bits indicate the Second Data Buffer in bytes. This field is not valid if TDES1[24] is set.
10:0	TBS1: Transmit Buffer 1 Size These bits indicate the First Data Buffer byte size. If this field is 0, the DMA ignores this buffer and uses Buffer 2 or next descriptor depending on the value of TCH (Bit 24).

Transmit Descriptor 2 (TDES2)

TDES2 contains the address pointer to the first buffer of the descriptor.

Table 17-8 Transmit Descriptor 2

Bit	Description
31:0	Buffer 1 Address Pointer These bits indicate the physical address of Buffer 1. There is no limitation on the buffer address alignment.

Transmit Descriptor 3 (TDES3)

TDES3 contains the address pointer either to the second buffer of the descriptor or the next descriptor.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Table 17-9 Transmit Descriptor 3

Bit	Description
31:0	Buffer 2 Address Pointer (Next Descriptor Address) Indicates the physical address of Buffer 2 when a descriptor ring structure is used. If the Second Address Chained (TDES1[24]) bit is set, this address contains the pointer to the physical memory where the Next Descriptor is present. The buffer address pointer must be aligned to the bus width only when TDES1[24] is set. (LSBs are ignored internally.)

17.6.4 Programming Guide

DMA Initialization – Descriptors

The following operations must be performed to initialize the DMA.

1. Provide a software reset. This will reset all of the GMAC internal registers and logic. (GMAC_OP_MODE[0]).
2. Wait for the completion of the reset process (poll GMAC_OP_MODE[0], which is only cleared after the reset operation is completed).
3. Program the following fields to initialize the Bus Mode Register by setting values in register GMAC_BUS_MODE
   a. Mixed Burst and AAL
   b. Fixed burst or undefined burst
   c. Burst length values and burst mode values.
   d. Descriptor Length (only valid if Ring Mode is used)
   e. Tx and Rx DMA Arbitration scheme
4. Program the AXI Interface options in the register GMAC_BUS_MODE
   a. If fixed burst-length is enabled, then select the maximum burst-length possible on the AXI bus (Bits[7:1])
5. A proper descriptor chain for transmit and receive must be created. It should also ensure that the receive descriptors are owned by DMA (bit 31 of descriptor should be set). When OSF mode is used, at least two descriptors are required.
6. Software should create three or more different transmit or receive descriptors in the chain before reusing any of the descriptors.
7. Initialize receive and transmit descriptor list address with the base address of transmit and receive descriptor (register GMAC_RX_DESC_LIST_ADDR and GMAC_TX_DESC_LIST_ADDR).
8. Program the following fields to initialize the mode of operation by setting values in register GMAC_OP_MODE
   a. Receive and Transmit Store And Forward
   b. Receive and Transmit Threshold Control (RTC and TTC)
   c. Hardware Flow Control enable
   d. Flow Control Activation and De-activation thresholds for MTL Receive and Transmit FIFO (RFA and RFD)
   e. Error Frame and undersized good frame forwarding enable
   f. OSF Mode
9. Clear the interrupt requests, by writing to those bits of the status register (interrupt bits only) which are set. For example, by writing 1 into bit 16 - normal interrupt summary will clear this bit (register GMAC_STATUS).
10. Enable the interrupts by programming the interrupt enable register GMAC_INT_ENA.
11. Start the Receive and Transmit DMA by setting SR (bit 1) and ST (bit 13) of the control register GMAC_OP_MODE.

MAC Initialization

The following MAC Initialization operations can be performed after the DMA initialization sequence. If the MAC Initialization is done before the DMA is set-up, then enable the MAC receiver (last step below) only after the DMA is active. Otherwise, received frames will fill the RxFIFO and overflow.

1. Program the register GMAC_GMII_ADDR for controlling the management cycles for

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

external PHY, for example, Physical Layer Address PA (bits 15-11). Also set bit 0 (GMII Busy) for writing into PHY and reading from PHY.
2. Read the 16-bit data of (GMAC_GMII_DATA) from the PHY for link up, speed of operation, and mode of operation, by specifying the appropriate address value in registerGMAC_GMII_ADDR (bits 15-11).
3. Provide the MAC address registers (GMAC_MAC_ADDR0_HI and GMAC_MAC_ADDR0_LO).
4. If Hash filtering is enabled in your configuration, program the Hash filter register (GMAC_HASH_TAB_HI and GMAC_HASH_TAB_LO).
5. Program the following fields to set the appropriate filters for the incoming frames in register GMAC_MAC_FRM_FILT
   a. Receive All
   b. Promiscuous mode
   c. Hash or Perfect Filter
   d. Unicast, Multicast, broad cast and control frames filter settings etc.
6. Program the following fields for proper flow control in register GMAC_FLOW_CTRL.
   a. Pause time and other pause frame control bits
   b. Receive and Transmit Flow control bits
   c. Flow Control Busy/Backpressure Activate
7. Program the Interrupt Mask register bits, as required, and if applicable, for your configuration.
8. Program the appropriate fields in register GMAC_MAC_CONF for example, Inter-frame gap while transmission, jabber disable, etc. Based on the Auto-negotiation you can set the Duplex mode (bit 11), port select (bit 15), etc.
9. Set the bits Transmit enable (TE bit-3) and Receive Enable (RE bit-2) in register GMAC_MAC_CONF.

Normal Receive and Transmit Operation

For normal operation, the following steps can be followed.

- For normal transmit and receive interrupts, read the interrupt status. Then poll the descriptors, reading the status of the descriptor owned by the Host (either transmit or receive).
- On completion of the above step, set appropriate values for the descriptors, ensuring that transmit and receive descriptors are owned by the DMA to resume the transmission and reception of data.
- If the descriptors were not owned by the DMA (or no descriptor is available), the DMA will go into SUSPEND state. The transmission or reception can be resumed by freeing the descriptors and issuing a poll demand by writing 0 into the Tx/Rx poll demand register (GMAC_TX_POLL_DEMAND and GMAC_RX_POLL_DEMAND).
- The values of the current host transmitter or receiver descriptor address pointer can be read for the debug process (GMAC_CUR_HOST_TX_DESC and GMAC_CUR_HOST_RX_DESC).
- The values of the current host transmit buffer address pointer and receive buffer address pointer can be read for the debug process (GMAC_CUR_HOST_TX_Buf_ADDR and GMAC_CUR_HOST_RX_BUF_ADDR).

Stop and Start Operation

When the transmission is required to be paused for some time then the following steps can be followed.
1. Disable the Transmit DMA (if applicable), by clearing ST (bit 13) of the control register GMAC_OP_MODE.
2. Wait for any previous frame transmissions to complete. This can be checked by reading the appropriate bits of MAC Debug register.
3. Disable the MAC transmitter and MAC receiver by clearing the bits Transmit enable (TE bit-3) and Receive Enable (RE bit-2) in register GMAC_MAC_CONF.
4. Disable the Receive DMA (if applicable), after making sure the data in the RX FIFO is transferred to the system memory (by reading the register GMAC_DEBUG).

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

5. Make sure both the TX FIFO and RX FIFO are empty.
6. To re-start the operation, start the DMAs first, before enabling the MAC Transmitter and Receiver.

17.6.5 Clock Architecture

In RMII mode, reference clock and TX/RX clock can be from CRU or external OSC as following figure.
The mux select rmii_speed is CRU_CLKSEL27_CON[1].

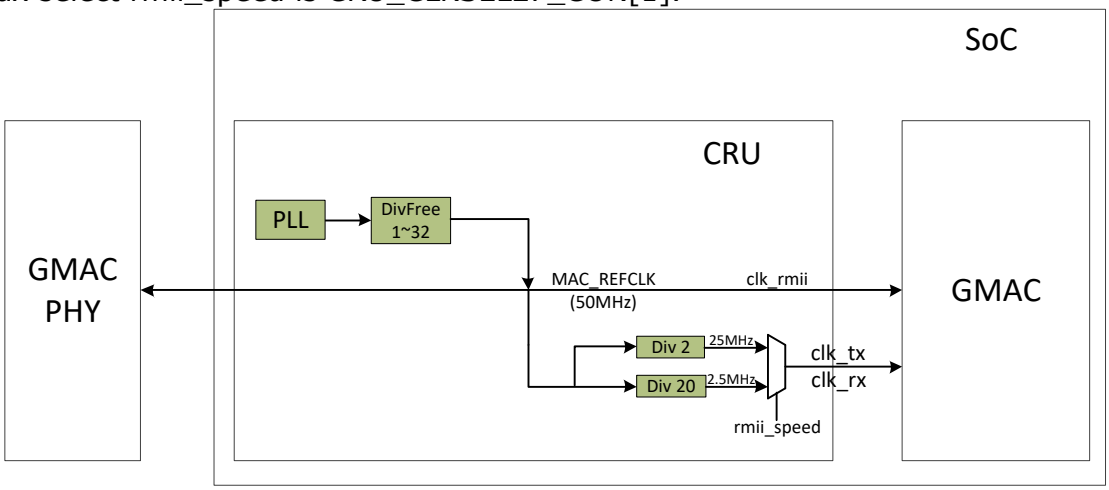


Fig. 17-10 RMII clock architecture when clock source from CRU

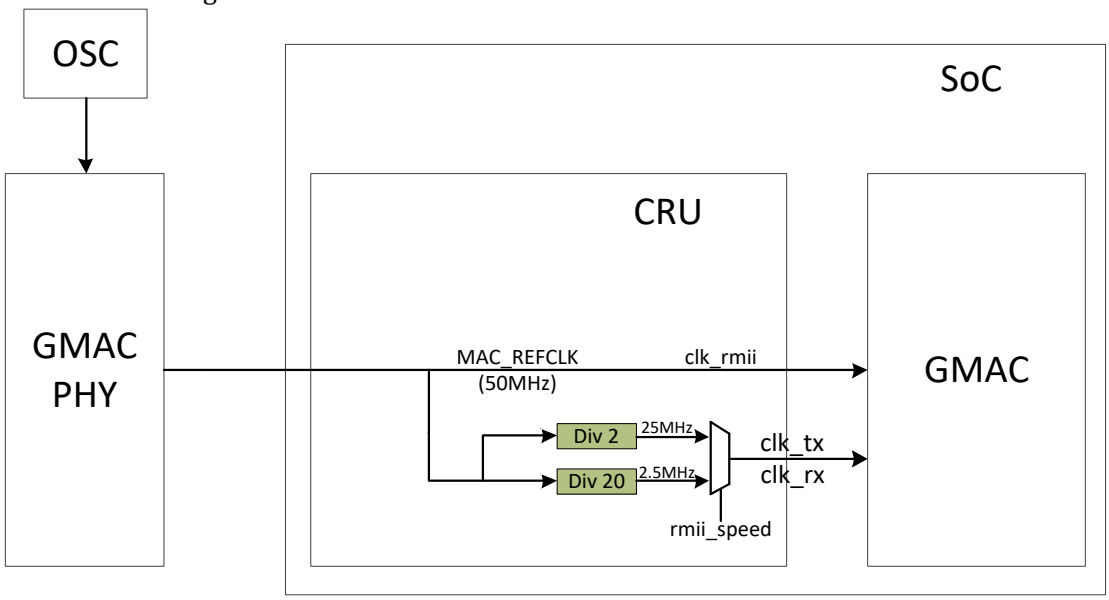


Fig. 17-11 RMII clock architecture when clock source from external OSC

In RGMII mode, clock architecture only supports that TX clock source is from CRU as following figure.
In order to dynamically adjust the timing between TX/RX clocks with data, deleyline is integrated in TX and RX clock path. Register GRF_MAC_CON1[1:0] can enable the deleylines, and GRF_MAC_CON0[15:0] is used to determine the delay length. There are 100 deley elements in each delayline.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

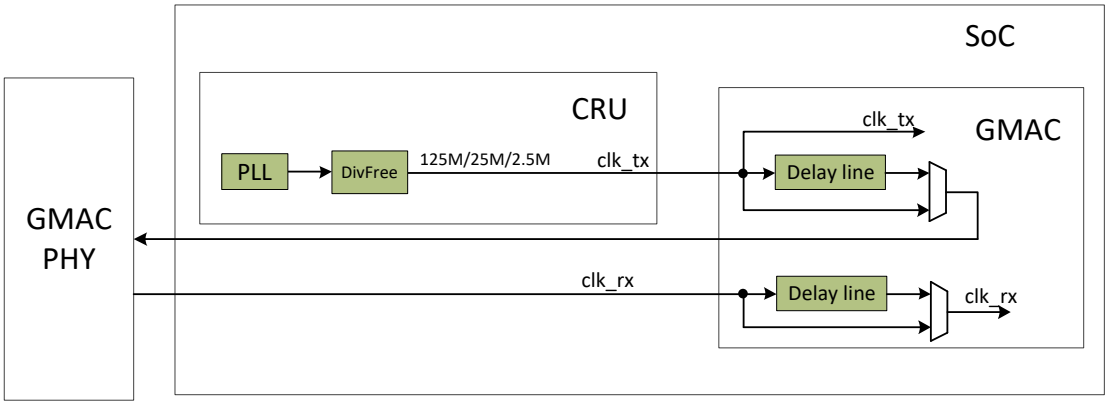


Fig. 17-12 RGMII clock architecture when clock source from CRU

17.6.6 Remote Wake-Up Frame Filter Register

The register wkupfmfilter_reg, address (028H), loads the Wake-up Frame Filter register. To load values in a Wake-up Frame Filter register, the entire register (wkupfmfilter_reg) must be written. The wkupfmfilter_reg register is loaded by sequentially loading the eight register values in address (028) for wkupfmfilter_reg0, wkupfmfilter_reg1, ..., wkupfmfilter_reg7, respectively. Wkupfmfilter_reg is read in the same way.
The internal counter to access the appropriate wkupfmfilter_reg is incremented when lane3 (or lane 0 in big-endian) is accessed by the CPU. This should be kept in mind if you are accessing these registers in byte or half-word mode.

wkupfmfilter_reg0	Filter 0 Byte Mask							
wkupfmfilter_reg1	Filter 1 Byte Mask							
wkupfmfilter_reg2	Filter 2 Byte Mask							
wkupfmfilter_reg3	Filter 3 Byte Mask							
wkupfmfilter_reg4	RSVD	Filter 3 Command	RSVD	Filter 2 Command	RSVD	Filter 1 Command	RSVD	Filter 0 Command
wkupfmfilter_reg5	Filter 3 Offset		Filter 2 Offset		Filter 1 Offset		Filter 0 Offset	
wkupfmfilter_reg6	Filter 1 CRC - 16				Filter 0 CRC - 16			
wkupfmfilter_reg7	Filter 3 CRC - 16				Filter 2 CRC - 16			

Fig. 17-13 Wake-Up Frame Filter Register

Filter i Byte Mask

This register defines which bytes of the frame are examined by filter i (0, 1, 2, and 3) in order to determine whether or not the frame is a wake-up frame. The MSB (thirty-first bit) must be zero. Bit j [30:0] is the Byte Mask. If bit j (byte number) of the Byte Mask is set, then Filter i Offset + j of the incoming frame is processed by the CRC block; otherwise Filter i Offset + j is ignored.

Filter i Command

This 4-bit command controls the filter i operation. Bit 3 specifies the address type, defining the pattern's destination address type. When the bit is set, the pattern applies to only multicast frames; when the bit is reset, the pattern applies only to unicast frame. Bit 2 and Bit 1 are reserved. Bit 0 is the enable for filter i; if Bit 0 is not set, filter i is disabled.

Filter i Offset

This register defines the offset (within the frame) from which the frames are examined by filter i. This 8-bit pattern-offset is the offset for the filter i first byte to examined. The minimum allowed is 12, which refers to the 13th byte of the frame (offset value 0 refers to the first byte of the frame).

Filter i CRC-16

This register contains the CRC_16 value calculated from the pattern, as well as the byte mask programmed to the wake-up filter register block.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

17.6.7 System Consideration During Power-Down

GMAC neither gates nor stops clocks when Power-Down mode is enabled. Power saving by clock gating must be done outside the core by the CRU. The receive data path must be clocked with clk_rx_i during Power-Down mode, because it is involved in magic packet/wake-on-LAN frame detection. However, the transmit path and the APB path clocks can be gated off during Power-Down mode.

The PMT interrupt is asserted when a valid wake-up frame is received. This interrupt is generated in the clk_rx domain.

The recommended power-down and wake-up sequence is as follows.

1. Disable the Transmit DMA (if applicable) and wait for any previous frame transmissions to complete. These transmissions can be detected when Transmit Interrupt (TI - Register GMAC_STATUS[0]) is received.
2. Disable the MAC transmitter and MAC receiver by clearing the appropriate bits in the MAC Configuration register.
3. Wait until the Receive DMA empties all the frames from the Rx FIFO (a software timer may be required).
4. Enable Power-Down mode by appropriately configuring the PMT registers.
5. Enable the MAC Receiver and enter Power-Down mode.
6. Gate the APB and transmit clock inputs to the core (and other relevant clocks in the system) to reduce power and enter Sleep mode.
7. On receiving a valid wake-up frame, the GMAC asserts the PMT interrupt signal and exits Power-Down mode.
8. On receiving the interrupt, the system must enable the APB and transmit clock inputs to the core.
9. Read the register GMAC_PMT_CTRL_STA to clear the interrupt, then enable the other modules in the system and resume normal operation.

17.6.8 GRF Register Summary

GMAC2IO	
GRF Register	**Register Description**
GRF_MAC_CON0[7:0]	RGMII TX clock delayline value
GRF_MAC_CON0[15:8]	RGMII RX clock delayline value
GRF_MAC_CON1[0]	RGMII TX clock delayline enable 1'b1: enable 1'b0: disable
GRF_MAC_CON1[1]	RGMII RX clock delayline enable 1'b1: enable 1'b0: disable
GRF_MAC_CON1[2]	GMACspeed 1'b1: 100-Mbps 1'b0: 10-Mbps
GRF_MAC_CON1[3]	GMAC transmit flow control When set high, instructs the GMAC to transmit PAUSE Control frames in Full-duplex mode. In Half-duplex mode, the GMAC enables the Back-pressure function until this signal is made low again
GRF_MAC_CON1[6:4]	PHY interface select 3'b001: RGMII 3'b100: RMII All others: Reserved
CRU_CLKSEL27_CON[0]	RMII external clock select 1’b1: Internal 1’b0: External
CRU_CLKSEL27_CON[1]	RMII speed select 1’b1: 100M 1’b0: 10M
CRU_CLKSEL27_CON[3:2]	RGMII speed select

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

	2'b0x: 125M 2'b10: 2.5M 2'b11: 25M
CRU_CLKSEL27_CON[4]	RMII or RGMII mode 1'b1: RMII 1'b0: RGMII

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 18 Pulse Density Modulation Interface Controller

18.1 Overview

The PDM interface controller and decoder support mono PDM format. It integrates a clock generator driving the PDM microphone and embeds filters which decimate the incoming bit stream to obtain most common audio rates.
PDM supports the following features:

- Support one internal 32-bit wide and 128-location deep FIFOs for receiving audio data
- Support receive FIFO full, overflow interrupt and all interrupts can be masked
- Support configurable water level of receive FIFO full interrupt
- Support combined interrupt output
- Support AHB bus slave interface
- Support DMA handshaking interface and configurable DMA water level
- Support PDM master receive mode
- Support 4 paths. Each path is composed of two digital microphone channels, the PDM can be used with four stereo or eight mono microphones. Each path is enabled or disabled independently
- Support 16 ~24 bit sample resolution
- Support sample rate: 8khz,16khz,32kHz,64kHz,128khz,11.025khz,22.05khz,44.1khz,88.2khz,176.4khz,12khz,24khz,48khz,96khz,192khz
- Support two 16-bit audio data store together in one 32-bit wide location
- Support 16 to 31 bit audio data left or right justified in 32-bit wide FIFO
- Support programmable data sampling sensibility (rising or falling edge)

18.2 Block Diagram

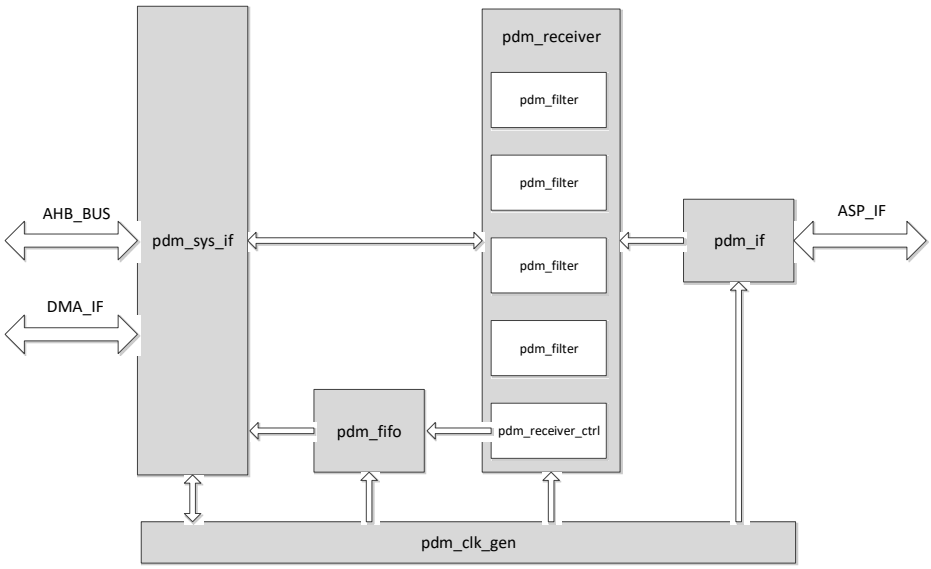


Fig. 18-1 PDM Block Diagram

System Interface
The system interface implements the APB slave operation. It contains not only control registers of receiver inside but also interrupt and DMA handshaking interface.
Clock Generator
The Clock Generator implements clock generation function. The input source clock to the module is MCLK, and by the divider of the module, the clock generator generates CLK_PDM to receiver.
Receiver
The receiver can act as a decimation filter of PDM. And export PCM format data.
Receive FIFO
The Receive FIFO is the buffer to store received audio data. The size of the FIFO is 32bits x

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

128.

ASP interface

The ASP interface implements PDM bit streams receive operation.

18.3 Function Description

18.3.1 AHB Interface

There is an AHB slave interface in PDM. It is responsible for accessing registers and SYSTEM_RAMs. The addresses of these registers and memories are listed in 错误!未找到引用源。.

18.3.2 PDM Interface

The PDM interface is a 5-wire interface. The PDM module can support up to four external stereo and eight digital microphones.

错误!未找到引用源。 and 错误!未找到引用源。 show two cases of use of the PDM, but all configurations are possible with stereo and mono digital microphones.

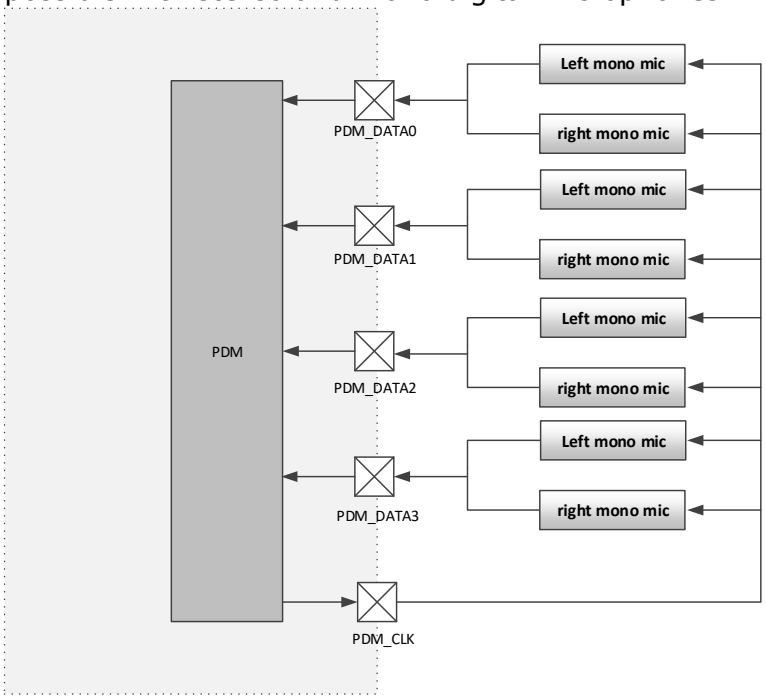


Fig. 18-2 PDM with Eight Mono MIC

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

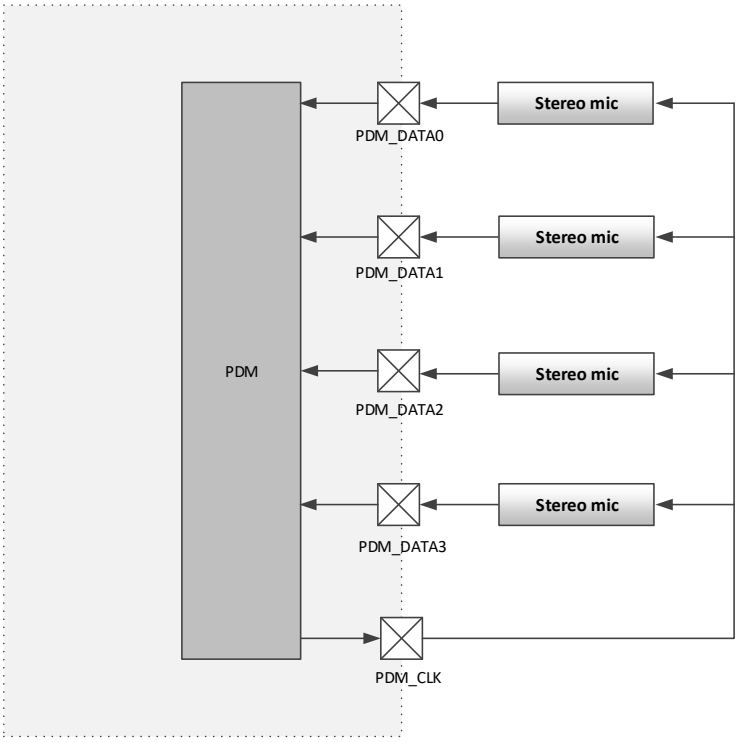


Fig. 18-3 PDM with Four Stereo MIC

The PDM interface consists of a serial-data shift clock output (PDM_CLK) and a serial data input (PDM_DATA). The clock is fanned out to both digital mics, and both digital mics' data (left channel and right channel) outputs share a single signal line. To share a single line, the digital mics tristate their output during one phase of the clock(high or low part of cycle, depending on how they are configured via their L/R input).

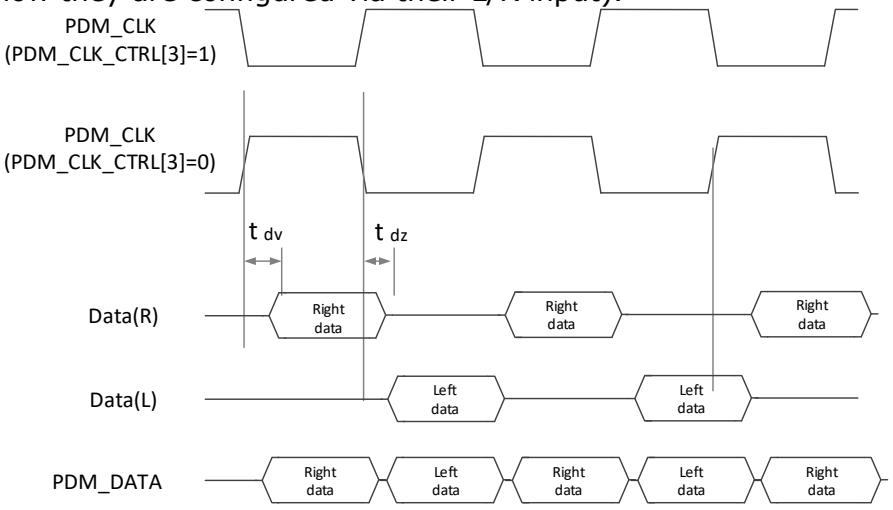


Fig. 18-4 PDM interface diagram with external MIC

18.3.3 Digital Filter

The external PDMIC generates a PDM stream of bits and transfers it in one period or one half-period of the clock provided by the PDM. The aim of the PDM is to process data from the PDM interface, decimate and filter the data, and store the processed data in the FIFO. The four paths are identical. Each path is composed of a left and a right channel. The PDM interface delivers eight parallel data of 1bit. Each bit goes to a filter. The aim of the filter is to limit the noise and export PCM format audio data.

18.3.4 Frequency Configuration

MCLK is the source clock signal. PDM_CLK (refers to O_asp_clk in the Fig.1-5) is the output clocks generated in the PDM and is fed to the external microphones. They are also the internal clock of the external microphones. User must take care about the value of PDM_CLK when selecting the source clock (MCLK).

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

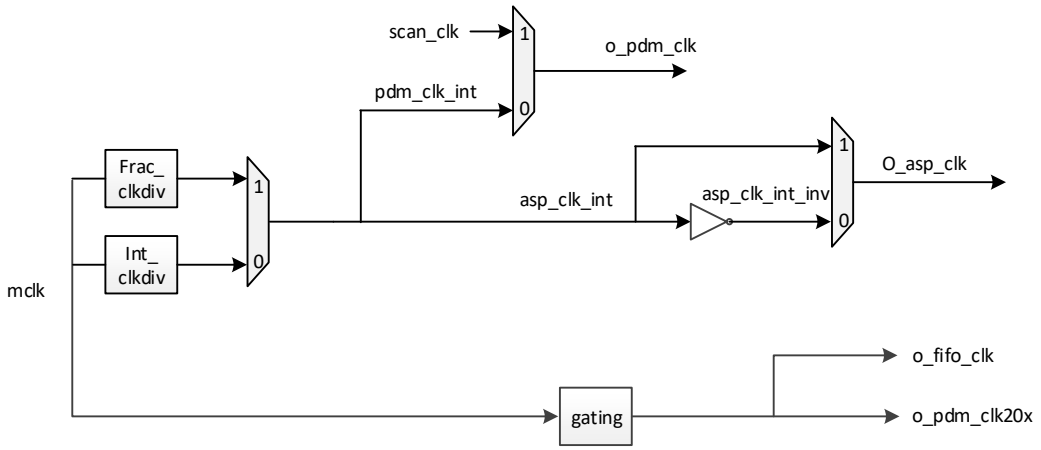


Fig. 18-5 PDM Clock Structure

Table 18-1 Relation between PDM_CLK and sample rate

PDM_CLK	Sample rate
3.072Mhz	12khz,24khz,48khz,96khz,192khz
2.8224Mhz	11.025khz,22.05khz,44.1khz,88.2khz,176.4khz
2.048Mhz	8khz,16khz,32kHz,64kHz,128khz

User must configure the div_con depended on the frequency of MCLK. If MCLK/PDM_CLK is more than 40, PDM_CLK_CTRL[6] should set to 0; if MCLK/PDM_CLK is less than 35, PDM_CLK_CTRL[6] should set to 1.

18.4 Register Description

18.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
PDM_SYSCONFIG	0x0000	W	0x00000000	PDM system config register
PDM_CTRL0	0x0004	W	0x78000017	PDM control register 0
PDM_CTRL1	0x0008	W	0x0bb8ea60	PDM control register 1
PDM_CLK_CTRL	0x000c	W	0x00000000	PDM clock control register
PDM_HPF_CTRL	0x0010	W	0x00000000	PDM high pass filter control register
PDM_FIFO_CTRL	0x0014	W	0x00000000	PDM fifo control register
PDM_DMA_CTRL	0x0018	W	0x0000001f	PDM dma control register
PDM_INT_EN	0x001c	W	0x00000000	PDM interrupt enable register
PDM_INT_CLR	0x0020	W	0x00000000	PDM interrupt clear register
PDM_INT_ST	0x0024	W	0x00000000	PDM interrupt status register
PDM_RXFIFO_DATA_REG	0x0030	W	0x00000000	PDM receive fifo data register
PDM_DATA0R_REG	0x0034	W	0x00000000	PDM path0 right channel data register
PDM_DATA0L_REG	0x0038	W	0x00000000	PDM path0 left channel data register
PDM_DATA1R_REG	0x003c	W	0x00000000	PDM path1 right channel data register
PDM_DATA1L_REG	0x0040	W	0x00000000	PDM path1 left channel data register
PDM_DATA2R_REG	0x0044	W	0x00000000	PDM path2 right channel data register

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
PDM_DATA2L_REG	0x0048	W	0x00000000	PDM path2 left channel data register
PDM_DATA3R_REG	0x004c	W	0x00000000	PDM path3 right channel data register
PDM_DATA3L_REG	0x0050	W	0x00000000	PDM path3 left channel data register
PDM_DATA_VALID	0x0054	W	0x00000000	PDM path data valid register
PDM_VERSION	0x0058	W	0x59313030	PDM version register
PDM_INCR_RXDR	0x0400	W	0x00000000	Increment address receive FIFO data register

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

18.4.2 Detail Register Description

PDM_SYSCONFIG

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:3	RO	0x0	Reserved
2	RW	0x0	rx_start RX Transfer start bit 1'b0: Stop RX transfer. 1'b1: Start RX transfer
1	RO	0x0	Reserved
0	RW	0x0	rx_clr PDM RX logic clear; This is a self cleard bit. High active. Write 0x1: Clear RX logic Write 0x0: No action Read 0x1: Clear ongoing Read 0x0: Clear done

PDM_CTRL0

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31	RW	0x0	sjm_sel Store justified mode: (Can be written only when SYSCONFIG[2] is 0.) 16bit~31bit DATA stored in 32 bits width fifo. If VDW select 16bit data, this bit is valid only when HWT select 1.Because if HWT is 0, every fifo unit contain two 16bit data and 32 bit space is full, it is impossible to choose justified mode. 1'b0: Right justified 1'b1: Left justified

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
30	RW	0x1	path3_en Path 3 enable; 1'b0: Disable 1'b1: Enable
29	RW	0x1	path2_en Path 2 enable; 1'b0: Disable 1'b1: Enable
28	RW	0x1	path1_en Path 1 enable; 1'b0: Disable 1'b1: Enable
27	RW	0x1	path0_en Path 0 enable; 1'b0: Disable 1'b1: Enable
26	RW	0x0	hwt_en HWT Halfword word transform Only valid when VDW select 16bit data. 1'b0: 32 bit data valid to AHB/APB bus. Low 16 bit for left channel and high 16 bit for right channel. 1'b1: Low 16bit data valid to AHB/APB bus, high 16 bit data invalid
25:16	RO	0x0	Reserved
15:8	RW	0x00	int_div_con integer divider; (Can be written only when SYSCONFIG[2] is 0.)
7:5	RO	0x0	Reserved
4:0	RW	0x17	data_vld_width (Can be written only when SYSCONFIG[2] is 0.) Valid Data width 0~14: Reserved 15: 16bit 16: 17bit 17: 18bit 18: 19bit …… n: (n+1)bit …… 23: 24bit

PDM_CTRL1

Address: Operational Base + offset (0x0008)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x0bb8	frac_div_numerator fraction divider numerator; (Can be written only when SYSCONFIG[2] is 0.)
15:0	RW	0xea60	frac_div_denomonator fraction divider denominator; (Can be written only when SYSCONFIG[2] is 0.)

PDM_CLK_CTRL

Address: Operational Base + offset (0x000c)

Bit	Attr	Reset Value	Description
31:7	RO	0x0	Reserved
6	RW	0x0	div_ratio_sel clk divider ratio select: (Can be written only when SYSCONFIG[2] is 0.) 1'b0: Ratio is more than 40 1'b1: Ratio is less than 35
5	RW	0x0	pdm_clk_en Pdm clk enable, working at PDM mode (Can be written only when SYSCONFIG[2] is 0.) 1'b0: Pdm clk disable 1'b1: Pdm clk enable
4	RW	0x0	div_type_sel divider type select signal (Can be written only when SYSCONFIG[2] is 0.) 1'b0: Fraction divider 1'b1: Integer divider
3	RW	0x0	clk_polar PDM_CLK polarity selection (Can be written only when SYSCONFIG[2] is 0.) 1'b0: No inverted 1'b1: Inverted
2:0	RW	0x0	pdm_ds_ratio DS_RATIO,working at PDM mode (Can be written only when SYSCONFIG[2] is 0.) 3'b000: Sample rate 192k/176.5k/128k 3'b001: Sample rate 96kk/88.2k/64k 3'b010: Sample rate 48kk/44.1k/32k 3'b011: Sample rate 24kk/22.05k/16k 3'b100: Sample rate 12kk/11.025k/8k

PDM_HPF_CTRL

Address: Operational Base + offset (0x0010)

Bit	Attr	Reset Value	Description
31:4	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
3	RW	0x0	hpfle HPFLE high pass filter enable for left channel 1'b0: High pass filter for right channel is disabled. 1'b1: High pass filter for right channel is enabled
2	RW	0x0	hpfre HPFRE high pass filter enable for right channel 1'b0: High pass filter for right channel is disabled. 1'b1: High pass filter for right channel is enabled
1:0	RW	0x0	hpf_cf HPF_CF high pass filter configure register high pass filter configure register 2'b00: 3.79Hz 2'b01: 60Hz 2'b10: 243Hz 2'b11: 493Hz

PDM_FIFO_CTRL

Address: Operational Base + offset (0x0014)

Bit	Attr	Reset Value	Description
31:15	RO	0x0	Reserved
14:8	RW	0x00	rft Receive FIFO Threshold When the number of receive FIFO entries is more than or equal to this threshold plus 1, the receive FIFO threshold interrupt is triggered
7:0	RO	0x00	rfl RFL Receive FIFO Level Contains the number of valid data entries in the receive FIFO

PDM_DMA_CTRL

Address: Operational Base + offset (0x0018)

Bit	Attr	Reset Value	Description
31:9	RO	0x0	Reserved
8	RW	0x0	rde Receive DMA Enable 1'b0: Receive DMA disabled 1'b1: Receive DMA enabled
7	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
6:0	RW	0x1f	rdl Receive Data Level This bit field controls the level at which a DMA request is made by the receive logic. The watermark level = DMARDL+1; that is, dma_rx_req is generated when the number of valid data entries in the receive FIFO is equal to or above this field value + 1

PDM_INT_EN

Address: Operational Base + offset (0x001c)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	Reserved
1	RW	0x0	rxoie RX overflow interrupt enable 1'b0: Disable 1'b1: Enable
0	RW	0x0	rxtie RX threshold interrupt enable 1'b0: Disable 1'b1: Enable

PDM_INT_CLR

Address: Operational Base + offset (0x0020)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	Reserved
1	W1C	0x0	rxoic RX overflow interrupt clear, high active, auto clear
0	RO	0x0	Reserved

PDM_INT_ST

Address: Operational Base + offset (0x0024)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	Reserved
1	RO	0x0	rxoi RX overflow interrupt 1'b0: Inactive 1'b1: Active
0	RO	0x0	rxfi RX full interrupt 1'b0: Inactive 1'b1: Active

PDM_RXFIFO_DATA_REG

Address: Operational Base + offset (0x0030)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	rxdr Receive FIFO shadow Register When the register is read, data in the receive FIFO is accessed

PDM_DATA0R_REG

Address: Operational Base + offset (0x0034)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	data0r Data of the path 0 right channel

PDM_DATA0L_REG

Address: Operational Base + offset (0x0038)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	data0l Data of the path 0 left channel

PDM_DATA1R_REG

Address: Operational Base + offset (0x003c)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	Reserved
0	RO	0x0	data1r Data of the path 1 right channel

PDM_DATA1L_REG

Address: Operational Base + offset (0x0040)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	data1l Data of the path 1 left channel

PDM_DATA2R_REG

Address: Operational Base + offset (0x0044)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	data2r Data of the path 2 right channel

PDM_DATA2L_REG

Address: Operational Base + offset (0x0048)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	data2l Data of the path 2 left channel

PDM_DATA3R_REG

Address: Operational Base + offset (0x004c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	data3r Data of the path 3 right channel

PDM_DATA3L_REG

Address: Operational Base + offset (0x0050)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	data3l Data of the path 3 left channel

PDM_DATA_VALID

Address: Operational Base + offset (0x0054)

Bit	Attr	Reset Value	Description
31:4	RO	0x0	Reserved
3	RC	0x0	path0_vld 1'b0: DATA0R_REG, DATA0L_REG value is invalid; 1'b1: DATA0R_REG, DATA0L_REG value is valid;
2	RC	0x0	path1_vld 1'b0: DATA1R_REG, DATA1L_REG value is invalid; 1'b1: DAT1R_REG, DATA1L_REG value is valid;
1	RC	0x0	path2_vld 1'b0: DATA2R_REG, DATA2L_REG value is invalid; 1'b1: DATA2R_REG, DATA2L_REG value is valid;
0	RC	0x0	path3_vld 1'b0: DATA3R_REG, DATA3L_REG value is invalid; 1'b1: DATA3R_REG, DATA3L_REG value is valid;

PDM_VERSION

Address: Operational Base + offset (0x0058)

Bit	Attr	Reset Value	Description
31:0	RO	0x59313030	version PDM version

PDM_INCR_RXDR

Address: Operational Base + offset (0x0400)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	receive_fifo_data FIFO data can be read from these registers, This register is used when the access address is Increment

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

18.5 Interface Description

Table 18-2 PDM Interface Description

Module Pin	Direction	Pin Name	IOMUX Setting
O_asp_clk	O	GPIO3_B0/I2S0_SCLK_RX/PDM_CLK0	GRF_GPIO3B_IOMUX_SEL_L[2:0]=3'b2
		GPIO3_B1/I2S0_LRCK_RX/PDM_CLK1	GRF_GPIO3B_IOMUX_SEL_L[6:4]=3'b2
I_asp_data0	I	GPIO3_C1/I2S0_SDI0/PDM_SDI0	GRF_GPIO3C_IOMUX_SEL_L[6:4]=3'b2
I_asp_data1	I	GPIO3_A7/I2S0_SDI1/PDM_SDI1	GRF_GPIO3A_IOMUX_SEL_H[14:12]=3'b2
I_asp_data2	I	GPIO3_A6/I2S0_SDI2/PDM_SDI2	GRF_GPIO3A_IOMUX_SEL_H[10:8]=3'b2
I_asp_data3	I	GPIO3_A5/I2S0_SDI3/PDM_SDI3	GRF_GPIO3A_IOMUX_SEL_H[6:4]=3'b2

Notes: I=input, O=output, I/O=input/output, bidirectional
When use IO_I2S08CHmclk_VCCIO2gpio2a4, the output enable is control by pmic_sleep

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

18.6 Application Notes

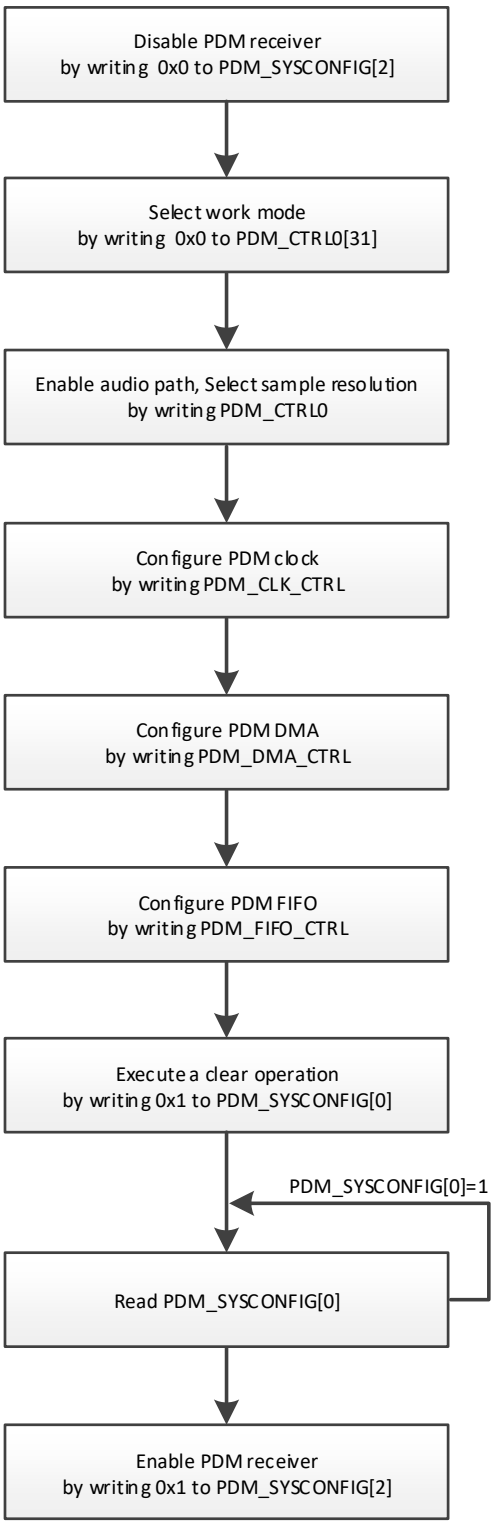


Fig. 18-6 PDM operation flow

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 19 I2S 2-Chanel

19.1 Overview

The I2S/PCM controller is designed for interfacing between the AHB bus and the I2S bus. The I2S bus (Inter-IC sound bus) is a serial link for digital audio data transfer between devices in the system and be invented by Philips Semiconductor. Now it is widely used by many semiconductor manufacturers.
Devices often use the I2S bus are ADC, DAC, DSP, CPU, etc. With the I2S interface, we can connect audio devices and the embedded SoC platform together and provide an audio interface solution for the system.
Not only I2S but also PCM mode surround audio output and stereo input are supported in I2S/PCM controller.
There are two 2 channel I2S/PCM controllers embedded in the design, I2S1 and I2S2.
Common features for I2S1 and I2S2 are as follows.

- Support AHB bus interface
- Support 16 ~ 32 bits audio data transfer
- Support master and slave mode
- Support DMA handshake interface and configurable DMA water level
- Support transmit FIFO empty, underflow, receive FIFO full, overflow interrupt and all interrupts can be masked
- Support configurable water level of transmit FIFO empty and receive FIFO full interrupt
- Support combine interrupt output
- Support 2 channels audio receiving in PCM mode
- Support I2S normal, left and right justified mode serial audio data transfer
- Support PCM early, late1, late2, late3 mode serial audio data transfer
- Support MSB or LSB first serial audio data transfer
- Support 16 to 31 bit audio data left or right justified in 32-bit wide FIFO
- Support two 16-bit audio data store together in one 32-bit wide location
- Support single LRCK for transmitting and receiving data if the sample rate are the same
- Support configurable SCLK and LRCK polarity

19.2 Block Diagram

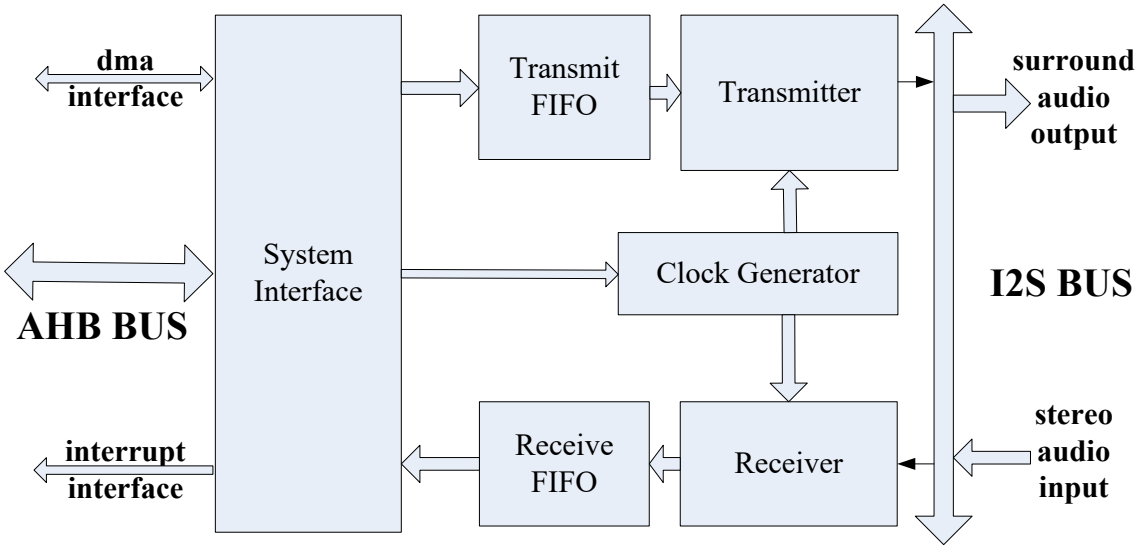


Fig. 19-1 I2S/PCM controller (2 channel) Block Diagram

System Interface
The system interface implements the AHB slave operation. It contains not only control registers of transmitter and receiver inside but also interrupt and DMA handshake interface.
Clock Generator
The Clock Generator implements clock generation function. The input source clock to the module is MCLK_I2S, and by the divider of the module, the clock generator generates SCLK and LRCK to transmitter and receiver.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Transmitter
The Transmitter implements transmission operation. The transmitter can act as either master or slave, with I2S or PCM mode surround serial audio interface.
Receiver
The Receiver implements receive operation. The receiver can act as either master or slave, with I2S or PCM mode stereo serial audio interface.
Transmit FIFO
The Transmit FIFO is the buffer to store transmitted audio data. The size of the FIFO is 32bits x 32.
Receive FIFO
The Receive FIFO is the buffer to store received audio data. The size of the FIFO is 32bits x 32.

19.3 Function description

In the I2S/PCM controller, there are four conditions: transmitter-master & receiver-master; transmitter-master & receiver-slave; transmitter-slave & receiver-master; transmitter-slave & receiver-slave.

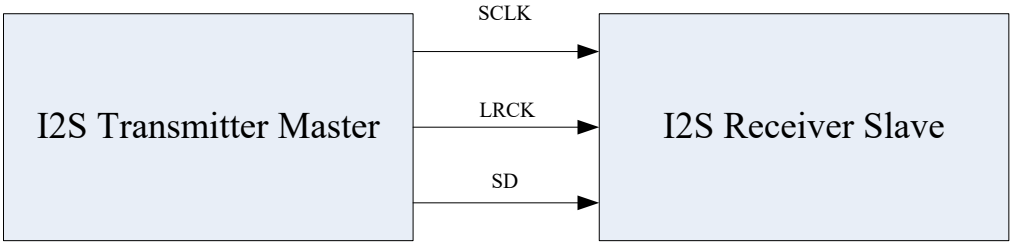


Fig. 19-2 I2S transmitter-master & receiver-slave condition

When transmitter acts as a master, it sends all signals to receiver (slave), and CPU control when to send clock and data to the receiver. When acting as a slave, SD signal still goes from transmitter to receiver, but SCLK and LRCK signals are from receiver (master) to transmitter. Based on three interface specifications, transmitting data should be ready before transmitter receives SCLK and LRCK signals. CPU should know when the receiver to initialize a transaction and when to send data.

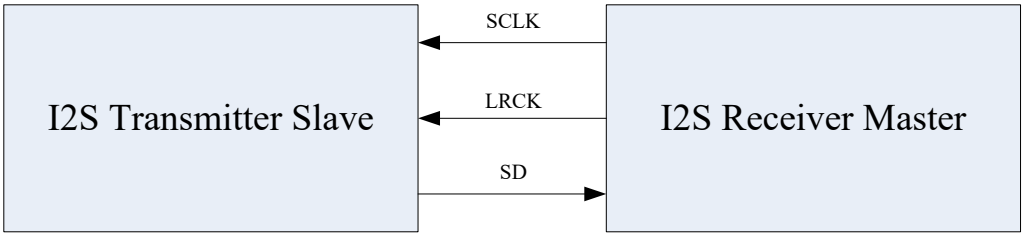


Fig. 19-3 I2S transmitter-slave& receiver-master condition

When the receiver acts as a master, it sends SCLK and LRCK signals to the transmitter (slave) and receives serial data. So CPU must tell the transmitter when to start a transaction for it to prepare transmitting data then the receiver start a transfer and send clock and channel-select signals. When the receiver acts as a slave, CPU should only do initial setting and wait for all signals and then start reading data.
Before transmitting or receiving data, CPU need do initial setting to the I2S register. These includes CPU settings, I2S interface registers settings, and maybe the embedded SoC platform settings. These registers must be set before starting data transfer.

19.3.1 I2S normal mode

This is the waveform of I2S normal mode. For LRCK (i2s_lrck_rx/i2s_lrck_tx) signal, it goes low to indicate left channel and high to right channel. For SD (i2s_sdo, i2s_sdi) signal, it transfers MSB or LSB first and sends the first bit one SCLK clock cycle after LRCK changes. The range of SD signal width is from 16 to 32bits.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

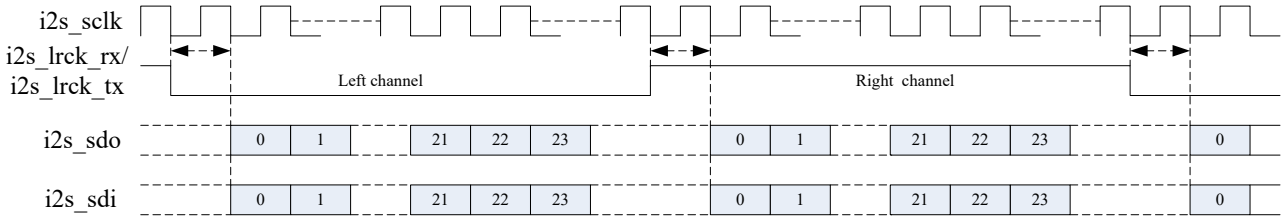


Fig. 19-4 I2S normal mode timing format

19.3.2 I2S left justified mode

This is the waveform of I2S left justified mode. For LRCK (i2s_lrck_rx / i2s_lrck_tx) signal, it goes high to indicate left channel and low to right channel. For SD (i2s_sdo, i2s_sdi) signal, it transfers MSB or LSB first and sends the first bit at the same time when LRCK changes. The range of SD signal width is from 16 to 32bits.

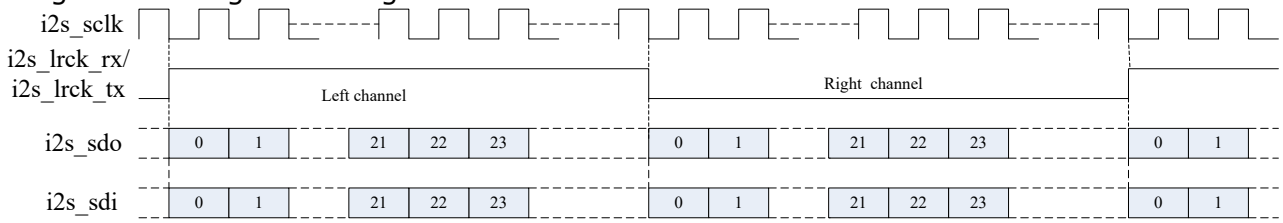


Fig. 19-5 I2S left justified mode timing format

19.3.3 I2S right justified mode

This is the waveform of I2S right justified mode. For LRCK (i2s_lrck_rx / i2s_lrck_tx) signal, it goes high to indicate left channel and low to right channel. For SD (i2s_sdo, i2s_sdi) signal, it transfers MSB or LSB first; but different from I2S normal or left justified mode, its data is aligned to last bit at the edge of the LRCK signal. The range of SD signal width is from 16 to 32bits.

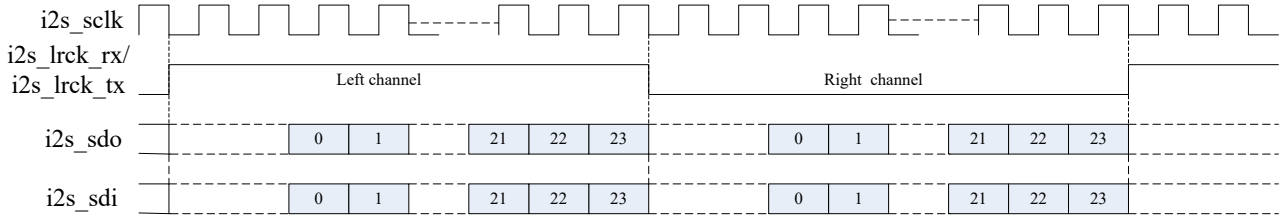


Fig. 19-6 I2S right justified mode timing format

19.3.4 PCM early mode

This is the waveform of PCM early mode. For LRCK (i2s_lrck_rx / i2s_lrck_tx) signal, it goes high to indicate the start of a group of audio channels. For SD (i2s_sdo, i2s_sdi) signal, it transfers MSB or LSB first and sends the first bit at the same time when LRCK goes high. The range of SD signal width is from 16 to 32bits.

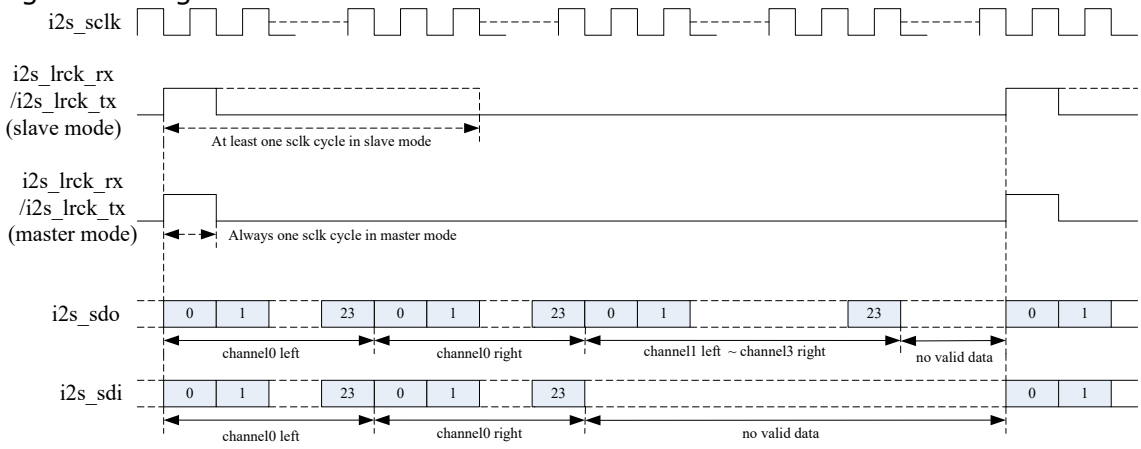


Fig. 19-7 PCM early mode timing format

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

19.3.5 PCM late1 mode

This is the waveform of PCM late1 mode. For LRCK (i2s_lrck_rx / i2s_lrck_tx) signal, it goes high to indicate the start of a group of audio channels. For SD (i2s_sdo, i2s_sdi) signal, it transfers MSB or LSB first and sends the first bit one SCLK clock cycle after LRCK goes high. The range of SD signal width is from 16 to 32bits.

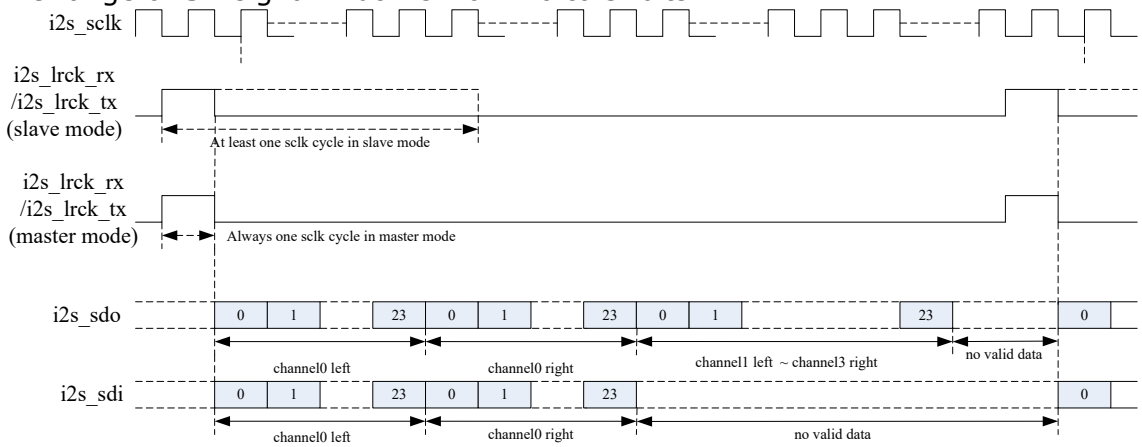


Fig. 19-8 PCM late1 mode timing format

19.3.6 PCM late2 mode

This is the waveform of PCM late2 mode. For LRCK (i2s_lrck_rx / i2s_lrck_tx) signal, it goes high to indicate the start of a group of audio channels. For SD (i2s_sdo, i2s_sdi) signal, it transfers MSB or LSB first and sends the first bit two SCLK clock cycles after LRCK goes high. The range of SD signal width is from 16 to 32bits.

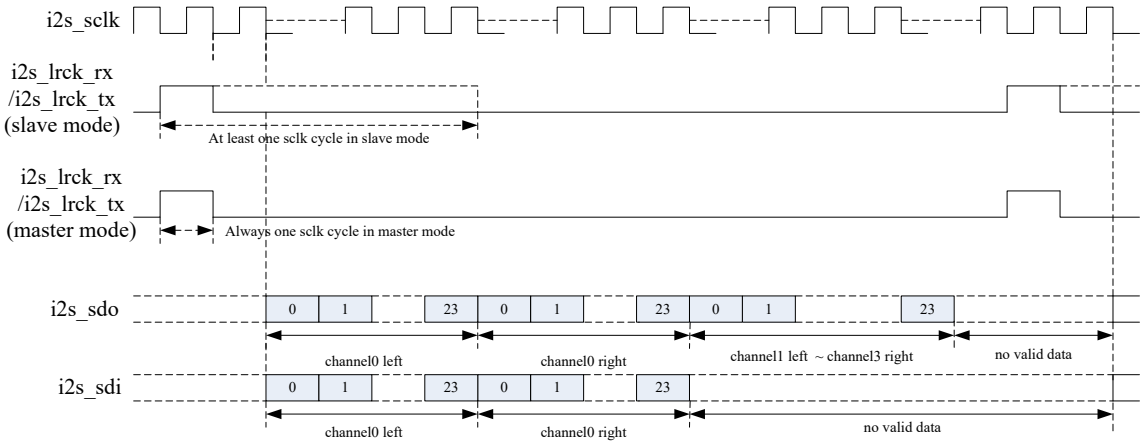


Fig. 19-9 PCM late2 mode timing format

19.3.7 PCM late3 mode

This is the waveform of PCM late3 mode. For LRCK (i2s_lrck_rx / i2s_lrck_tx) signal, it goes high to indicate the start of a group of audio channels. For SD (i2s_sdo, i2s_sdi) signal, it transfers MSB or LSB first and sends the first bit three SCLK clock cycles after LRCK goes high. The range of SD signal width is from 16 to 32bits.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

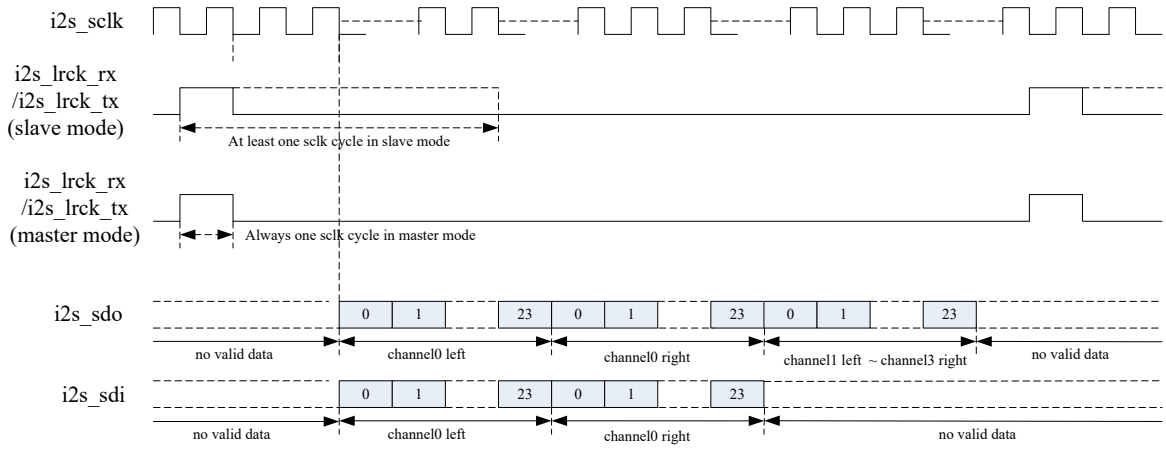


Fig. 19-10 PCM late3 mode timing format

19.4 Register Description

This section describes the control/status registers of the design.

19.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
I2S_TXCR	0x0000	W	0x00000000	Transmit Operation Control Register
I2S_RXCR	0x0004	W	0x00000000	Receive Operation Control Register
I2S_CKR	0x0008	W	0x00001f00	Clock Generation Register
I2S_FIFOLR	0x000c	W	0x00000000	FIFO Level Register
I2S_DMACR	0x0010	W	0x001f0000	DMA Control Register
I2S_INTCR	0x0014	W	0x00000000	Interrupt Control Register
I2S_INTSR	0x0018	W	0x00000000	Interrupt Status Register
I2S_XFER	0x001c	W	0x00000000	Transfer Start Register
I2S_CLR	0x0020	W	0x00000000	SCLK Domain Logic Clear Register
I2S_TXDR	0x0024	W	0x00000000	Transmit FIFO Data Register
I2S_RXDR	0x0028	W	0x00000000	Receive FIFO Data Register

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

19.4.2 Detail Register Description

I2S_TXCR

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:23	RO	0x0	Reserved
22:17	RW	0x00	rcnt Right Justified Counter (Can be written only when XFER[0] bit is 0.) Only valid in I2S Right justified format and slave tx mode is selected. Start to transmit data RCNT sclk cycles after left channel valid.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
16:15	RW	0x0	csr Channel Select Register 2'b00: 2 channel 2'b01~2'b11: Reserved
14	RW	0x0	hwt Halfword Word Transform (Can be written only when XFER[0] bit is 0.) Only valid when VDW select 16bit data. 1'b0: 32 bit data valid from AHB/APB bus. Low 16 bit for left channel and high 16 bit for right channel. 1'b1: Low 16bit data valid from AHB/APB bus, high 16 bit data invalid.
13	RO	0x0	Reserved
12	RW	0x0	sjm Store Justified Mode (Can be written only when XFER[0] bit is 0.) 16bit~31bit DATA stored in 32 bits width fifo. If VDW select 16bit data, this bit is valid only when HWT select 0.Because if HWT is 1, every fifo unit contain two 16bit data and 32 bit space is full, it is impossible to choose justified mode. 1'b0: Right justified 1'b1: Left justified
11	RW	0x0	fbm First Bit Mode (Can be written only when XFER[0] bit is 0.) 1'b0: MSB 1'b1: LSB
10:9	RW	0x0	ibm I2S Bus Mode (Can be written only when XFER[0] bit is 0.) 2'b00: I2S normal 2'b01: I2S Left justified 2'b10: I2S Right justified 2'b11: Reserved
8:7	RW	0x0	pbm PCM Bus Mode (Can be written only when XFER[0] bit is 0.) 2'b00: PCM no delay mode 2'b01: PCM delay 1 mode 2'b10: PCM delay 2 mode 2'b11: PCM delay 3 mode
6	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5	RW	0x0	tfs Transfer Format Select (Can be written only when XFER[0] bit is 0.) 1'b0: I2S format 1'b1: PCM format
4:0	RW	0x00	vdw Valid Data Width (Can be written only when XFER[0] bit is 0.) 5'h0~5'he: Reserved 5'h0f: 16bit 5'h10: 17bit 5'h11: 18bit 5'h12: 19bit …… 5'h1c: 29bit 5'h1d: 30bit 5'h1e: 31bit 5'h1f: 32bit

I2S_RXCR

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:15	RO	0x0	Reserved
14	RW	0x0	hwt Halfword Word Transform (Can be written only when XFER[1] bit is 0.) Only valid when VDW select 16bit data. 1'b0: 32 bit data valid to AHB/APB bus. Low 16 bit for left channel and high 16 bit for right channel. 1'b1: Low 16bit data valid to AHB/APB bus, high 16 bit data invalid.
13	RO	0x0	Reserved
12	RW	0x0	sjm Store Justified Mode (Can be written only when XFER[1] bit is 0.) 16bit~31bit DATA stored in 32 bits width fifo. If VDW select 16bit data, this bit is valid only when HWT select 0.Because if HWT is 1, every fifo unit contain two 16bit data and 32 bit space is full, it is impossible to choose justified mode. 1'b0: Right justified 1'b1: Left justified

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
11	RW	0x0	fbm First Bit Mode (Can be written only when XFER[1] bit is 0.) 1'b0: MSB 1'b1: LSB
10:9	RW	0x0	ibm I2S Bus Mode (Can be written only when XFER[1] bit is 0.) 2'b00: I2S normal 2'b01: I2S Left justified 2'b10: I2S Right justified 2'b11: Reserved
8:7	RW	0x0	pbm PCM Bus Mode (Can be written only when XFER[1] bit is 0.) 2'b00: PCM no delay mode 2'b01: PCM delay 1 mode 2'b10: PCM delay 2 mode 2'b11: PCM delay 3 mode
6	RO	0x0	Reserved
5	RW	0x0	tfs Transfer Format Select (Can be written only when XFER[1] bit is 0.) 1'b0: I2S 1'b1: PCM
4:0	RW	0x00	vdw Valid Data Width (Can be written only when XFER[1] bit is 0.) 5'h0~5'h0e: Reserved 5'h0f: 16bit 5'h10: 17bit 5'h11: 18bit 5'h12: 19bit …… 5'h1c: 29bit 5'h1d: 30bit 5'h1e: 31bit 5'h1f: 32bit

I2S_CKR

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
29:28	RW	0x0	trcm Tx and Rx Common Use 2'b00/2'b11: tx_lrck/rx_lrck are used as synchronous signal for TX /RX respectively. 2'b01: Only tx_lrck is used as synchronous signal for TX and RX. 2'b10: Only rx_lrck is used as synchronous signal for TX and RX.
27	RW	0x0	mss Master/Slave Mode Select (Can be written only when XFER[1] or XFER[0] bit is 0.) 1'b0: Master mode(sclk output) 1'b1: Slave mode(sclk input)
26	RW	0x0	ckp Sclk Polarity (Can be written only when XFER[1] or XFER[0] bit is 0.) 1'b0: Sample data at posedge sclk and drive data at negedge sclk 1'b1: Sample data at negedge sclk and drive data at posedge sclk
25	RW	0x0	rlp Receive Lrck Polarity (Can be written only when XFER[1] or XFER[0] bit is 0.) 1'b0: Normal polarity (I2S normal: Low for left channel, high for right channel I2S left/right just: High for left channel, low for right channel PCM start signal: High valid) 1'b1: Opposite polarity (I2S normal: High for left channel, low for right channel I2S left/right just: Low for left channel, high for right channel PCM start signal: Low valid)
24	RW	0x0	tlp Transmit Lrck Polarity (Can be written only when XFER[1] or XFER[0] bit is 0.) 1'b0: Normal polarity (I2S normal: Low for left channel, high for right channel I2S left/right just: High for left channel, low for right channel PCM start signal: High valid) 1'b1: Opposite polarity (I2S normal: High for left channel, low for right channel I2S left/right just: Low for left channel, high for right channel PCM start signal: Low valid)
23:16	RW	0x00	mdiv Mclk Divider (Can be written only when XFER[1] or XFER[0] bit is 0.) mclk divider = (mclk/sclk)-1. For example, if mclk divider is 5, then the frequency of sclk is mclk/6

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15:8	RW	0x1f	rsd Receive Sclk Divider (Can be written only when XFER[1] or XFER[0] bit is 0.) 8'h00~8'h1e: Reserved 8'h1f~8'hff: Frequency of sclk = (receive sclk divider/2)*2*frequency of rx_lrck
7:0	RW	0x00	tsd Transmit Sclk Divider (Can be written only when XFER[1] or XFER[0] bit is 0.) 8'h00~8'h1e: Reserved 8'h1f~8'hff: Frequency of sclk = (Transmit sclk divider/2)*2*frequency of tx_lrck

I2S_FIFOLR

Address: Operational Base + offset (0x000c)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	Reserved
29:24	RO	0x00	rfl Receive FIFO Level Contains the number of valid data entries in the receive FIFO.
23:6	RO	0x0	Reserved
5:0	RW	0x00	tfl Transmit FIFO0 Level Contains the number of valid data entries in the transmit FIFO.

I2S_DMACR

Address: Operational Base + offset (0x0010)

Bit	Attr	Reset Value	Description
31:25	RO	0x0	Reserved
24	RW	0x0	rde Receive DMA Enable 1'b0: Receive DMA disabled 1'b1: Receive DMA enabled
23:21	RO	0x0	Reserved
20:16	RW	0x1f	rdl Receive Data Level This bit field controls the level at which a DMA request is made by the receive logic. The watermark level = DMARDL+1; that is, dma_rx_req is generated when the number of valid data entries in the receive FIFO is equal to or above this field value + 1.
15:9	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
8	RW	0x0	tde Transmit DMA Enable 1'b0: Transmit DMA disabled 1'b1: Transmit DMA enabled
7:5	RO	0x0	Reserved
4:0	RW	0x00	tdl Transmit Data Level This bit field controls the level at which a DMA request is made by the transmit logic. It is equal to the watermark level; that is, the dma_tx_req signal is generated when the number of valid data entries in the TXFIFO is equal to or below this field value.

I2S_INTCR

Address: Operational Base + offset (0x0014)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:25	RO	0x0	Reserved
24:20	RW	0x00	rft Receive FIFO Threshold When the number of receive FIFO entries is more than or equal to this threshold plus 1, the receive FIFO full interrupt is triggered.
19	RO	0x0	Reserved
18	WO	0x0	rxoic RX Overrun Interrupt Clear Write 1 to clear RX overrun interrupt.
17	RW	0x0	rxoie RX Overrun Interrupt Enable 1'b0: Disable 1'b1: Enable
16	RW	0x0	rxfie RX Full Interrupt Enable 1'b0: Disable 1'b1: Enable
15:9	RO	0x0	Reserved
8:4	RW	0x00	tft Transmit FIFO Threshold When the number of transmit FIFO entries is less than or equal to this threshold, the transmit FIFO empty interrupt is triggered.
3	RO	0x0	Reserved
2	WO	0x0	txuic TX Underrun Interrupt Clear Write 1 to clear TX underrun interrupt.
1	RW	0x0	txuie TX Underrun Interrupt Enable 1'b0: Disable 1'b1: Enable
0	RW	0x0	txele TX empty Interrupt Enable 1'b0: Disable 1'b1: Enable

I2S_INTSR

Address: Operational Base + offset (0x0018)

Bit	Attr	Reset Value	Description
31:18	RO	0x0	Reserved
17	RO	0x0	rxoi RX Overrun Interrupt 1'b0: Inactive 1'b1: Active

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
16	RO	0x0	rxfi RX Full Interrupt 1'b0: Inactive 1'b1: Active
15:2	RO	0x0	Reserved
1	RO	0x0	txui TX Underrun Interrupt 1'b0: Inactive 1'b1: Active
0	RO	0x0	txei TX Empty Interrupt 1'b0: Inactive 1'b1: Active

I2S_XFER

Address: Operational Base + offset (0x001c)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	Reserved
1	RW	0x0	rxs RX Transfer Start Bit 1'b0: Stop RX transfer 1'b1: Start RX transfer
0	RW	0x0	txs TX Transfer Start Bit 1'b0: Stop TX transfer 1'b1: Start TX transfer

I2S_CLR

Address: Operational Base + offset (0x0020)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	Reserved
1	RW	0x0	rxc RX Logic Clear This is a self-cleared bit. Write 1 to clear all receive logic.
0	RW	0x0	txc TX Logic Clear This is a self-cleared bit. Write 1 to clear all transmit logic.

I2S_TXDR

Address: Operational Base + offset (0x0024)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	txdr Transimt FIFO Data Register When it is written to, data are moved into the transmit FIFO.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

I2S_RXDR

Address: Operational Base + offset (0x0028)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxdr Receive FIFO Data Register When the register is read, data in the receive FIFO is accessed.

19.5 Interface Description

Table 19-1 I2S_2CH Interface Description

Module Pin	Direction	Pad Name	IOMUX Setting
Interface for i2s1			
i2s1_mclk	I/O	GPIO3_A2/I2S1_MCLK/PWM7	GRF_GPIO3A_IOMUX_L[10:8]= 3'b001
i2s1_sclk	I/O	GPIO3_A1/I2S1_SCLK/PWM6	GRF_ GPIO3A_IOMUX_L[6:4] =3'b001
i2s1_lrck	I/O	GPIO3_A0/I2S1_LRCK	GRF_GPIO3A_IOMUX_L[2:0] =3'b001
i2s1_sdo	O	GPIO3_A3/I2S1_SDO/UART2_TX_M2	GRF_GPIO3A_IOMUX_L[14:12] =3'b001
i2s1_sdi	I	GPIO3_A4/I2S1_SDI/UART2_RX_M2	GRF_GPIO3A_IOMUX_H[2:0] =3'b001

Notes: I=input, O=output, I/O=input/output, bidirectional

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

19.6 Application Notes

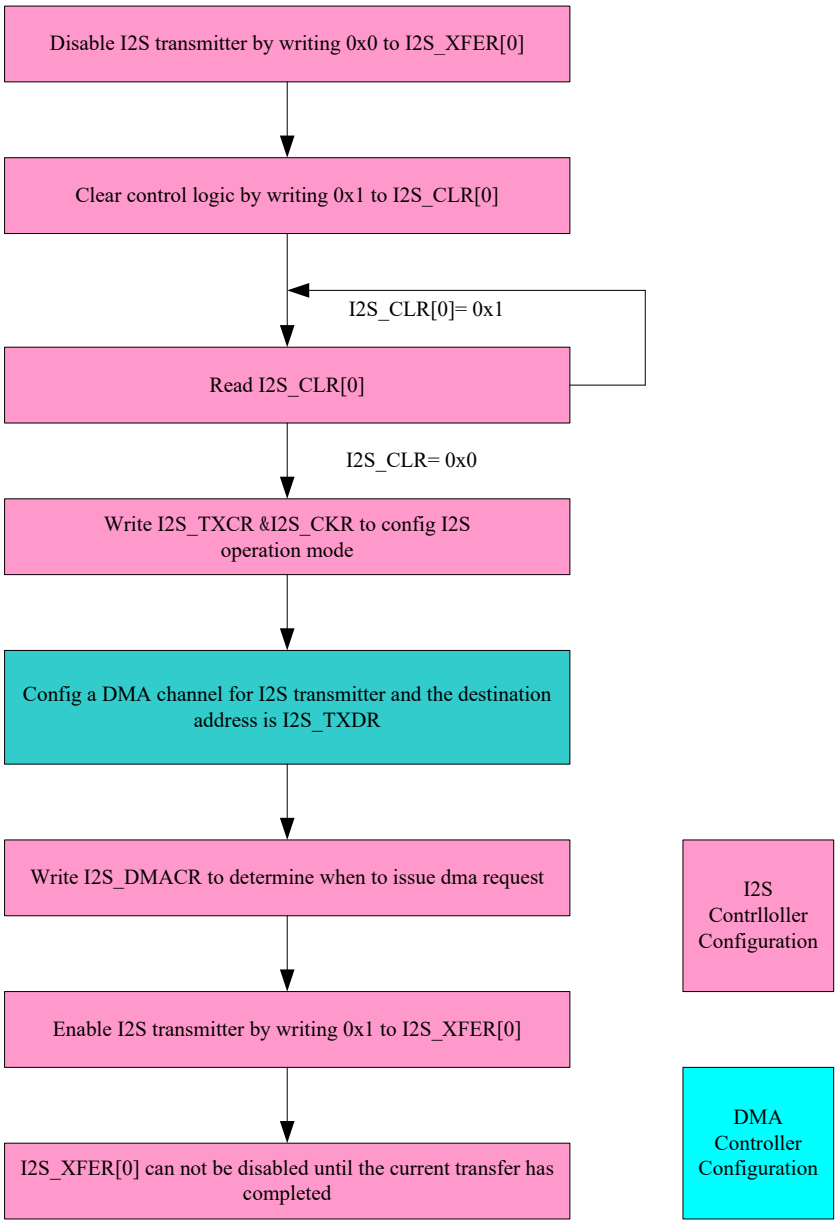


Fig. 19-11 I2S/PCM controller transmit operation flow chart

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

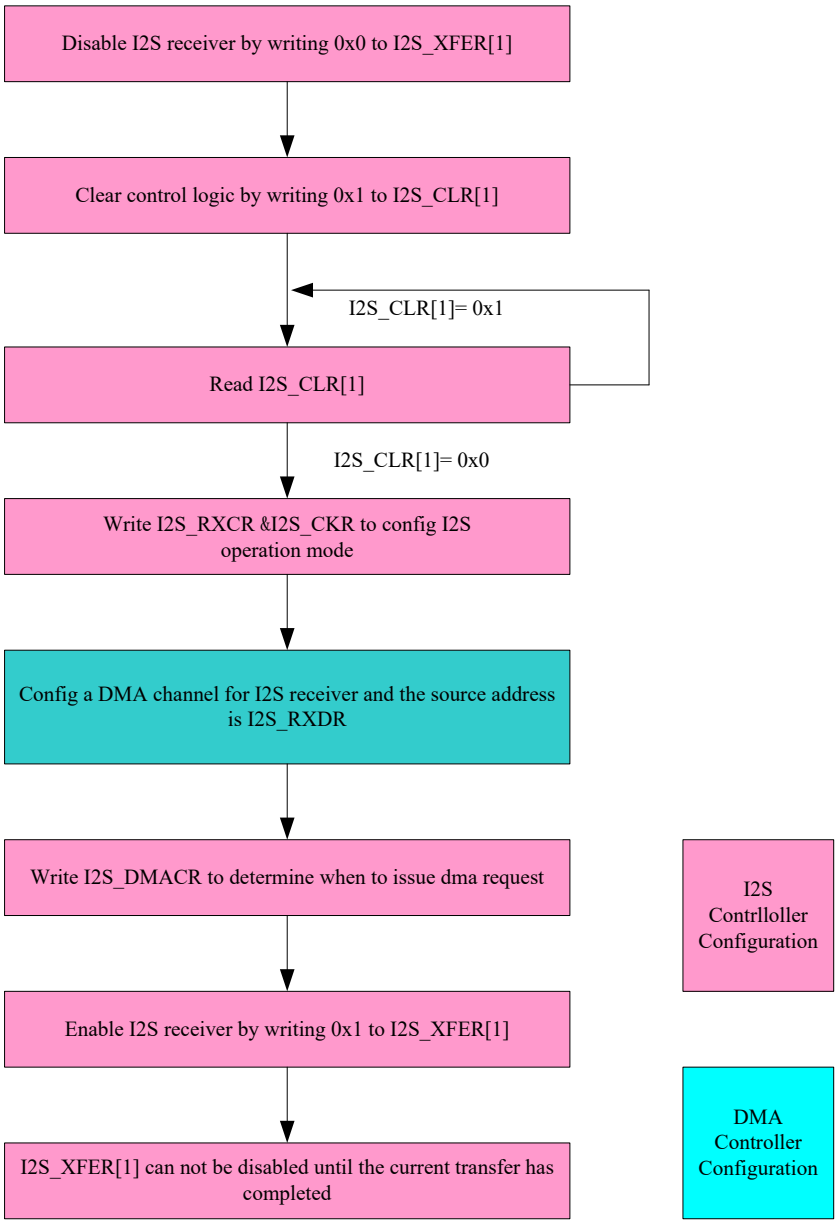


Fig. 19-12 I2S/PCM controller receive operation flow chart

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 20 I2S 8-Channel

20.1 Overview

The I2S/PCM/TDM controller is designed for interfacing between the AHB bus and the I2S bus.
The I2S bus (Inter-IC sound bus) is a serial link for digital audio data transfer between devices in the system and is invented by Philips Semiconductor. Now it is widely used by many semiconductor manufacturers.
I2S bus is widely used in the devices such as ADC, DAC, DSP, CPU, etc. With the I2S interface, we can connect audio devices and the embedded SoC platform together and provide an audio interface solution for the system.

20.1.1 Features

The I2S/PCM/TDM controller supports I2S, PCM and TDM mode stereo audio output and input.

- Support eight internal 32-bit wide and 32-location deep FIFOs, four for transmitting and the other for receiving audio data
- Support AHB bus interface
- Support 16 ~ 32 bits audio data transfer
- Support master and slave mode
- Support DMA handshaking interface and configurable DMA water level
- Support transmit FIFO empty, underflow, receive FIFO full, overflow interrupt and all interrupts can be masked
- Support configurable water level of transmit FIFO empty and receive FIFO full interrupt
- Support combined interrupt output
- Support 8-channel audio transmitting in I2S/TDM mode and 2-channel in PCM mode.
- Support 8-channel audio receiving in I2S/TDM mode and 2 channel in PCM mode
- Support up to 192kHz sample rate
- Support I2S normal, left and right justified mode serial audio data transfer
- Support PCM early, late1, late2, late3 mode serial audio data transfer
- Support TDM normal,1/2 cycle left shift ,1 cycle left shift,2 cycle left shift, right shift mode serial audio data transfer.
- Support MSB or LSB first serial audio data transfer
- Support 16 to 31 bit audio data left or right justified in 32-bit wide FIFO
- Support two 16-bit audio data store together in one 32-bit wide location
- Support 2 independent LRCK signals, one for receiving and the other for transmitting audio data. Single LRCK can be used for transmitting and receiving data if the sample rate are the same
- Support configurable SCLK and LRCK polarity
- Support TDM programmable slot bit width: 16~32bits
- Support TDM programmable frame width: 32~512bits
- Support TDM programmable FSYNC width
- Support SDI,SDO IOMUX.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

20.2 Block Diagram

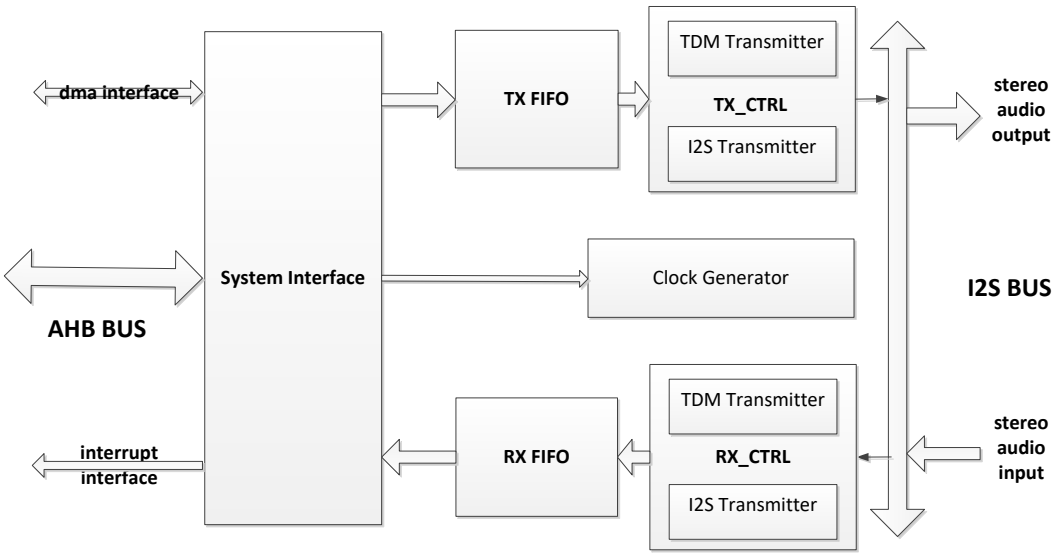


Fig. 20-1 I2S/PCM/TDM controller (8 channel) Block Diagram

System Interface

The system interface implements the AHB slave operation. It contains not only control registers of transmitters and receiver inside but also interrupt and DMA handshaking interface.

Clock Generator

The Clock Generator implements clock generation function. The input source clock to the module is MCLK_I2S, and by the divider of the module, the clock generator generates SCLK and LRCK to transmitter and receiver.

Transmitters

The Transmitters implement transmission operation. The transmitters can act as either a master or a slave, with I2S, PCM or TDM mode surround serial audio interface.

Receiver

The Receiver implements receive operation. The receiver can act as either a master or a slave, with I2S, PCM or TDM mode stereo serial audio interface.

Transmit FIFO

The Transmit FIFO is the buffer to store transmitted audio data. The size of the FIFO is 32bits x 32.

Receive FIFO

The Receive FIFO is the buffer to store received audio data. The size of the FIFO is 32bits x 32.

20.3 Function description

In the I2S/PCM/TDM controller, there are four types: transmitter-master &receiver-master; transmitter-master & receiver-slave; transmitter-slave & receiver-master; transmitter-slave & receiver-slave.

In broadcasting application, the I2S/PCM/TDM controller is used as a transmitter and external or internal audio CODEC is used as a receiver. In recording application, the I2S/PCM/TDM controller is used as a receiver and external or internal audio CODEC is used as a transmitter. Either the I2S/PCM/TDM controller or the audio CODEC can act as a master or a slave, but if one is master, the other must be slave.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

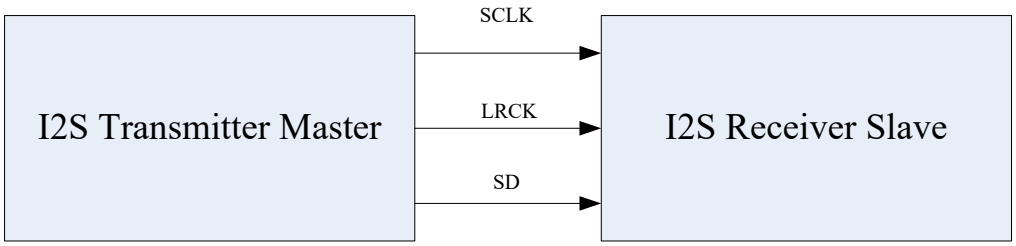


Fig. 20-2 I2S transmitter-master & receiver-slave condition

When the transmitter acts as a master, it sends all signals to the receiver (the slave), and CPU controls when to send clock and data to the receiver. When acts as a slave, SD signal still goes from transmitter to receiver, but SCLK and LRCK signals are from the receiver (the master) to the transmitter. Based on three interface specifications, transmitting data should be ready before transmitter receives SCLK and LRCK signals. CPU should know when the receiver to initialize a transaction and when the transmitter to send data.

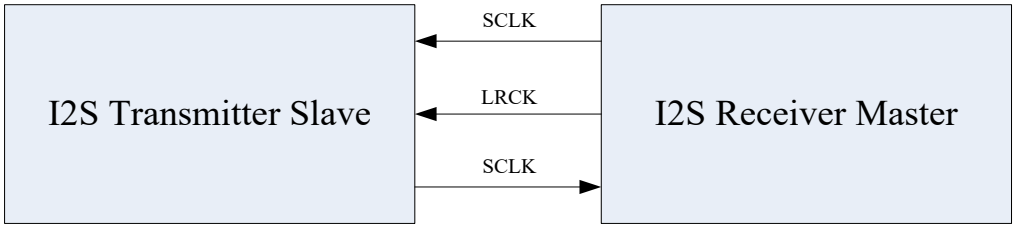


Fig. 20-3 I2S transmitter-slave & receiver-master condition

When the receiver acts as a master, it sends SCLK and LRCK signals to the transmitter (the slave) and receives serial data. So CPU must tell the transmitter when to start a transaction for it to prepare transmitting data then start a transfer and send clock and channel-select signals. When the receiver acts as a slave, CPU should only do initial setting and wait for all signals and then start reading data.

Before transmitting or receiving data, CPU need do initial setting to the I2S register. These includes CPU settings, I2S interface registers settings, and maybe the embedded SoC platform settings. These registers must be set before starting data transfer.

20.3.1 I2S normal mode

This is the waveform of I2S normal mode. For LRCK (i2s1_lrck_rx/i2s1_lrck_tx) signal, it goes low to indicate left channel and high to right channel. For SD (i2s1_sdo, i2s1_sdi) signal, it starts sending the first bit (MSB or LSB) one SCLK clock cycle after LRCK changes. The range of SD signal width is from 16 to 32bits.

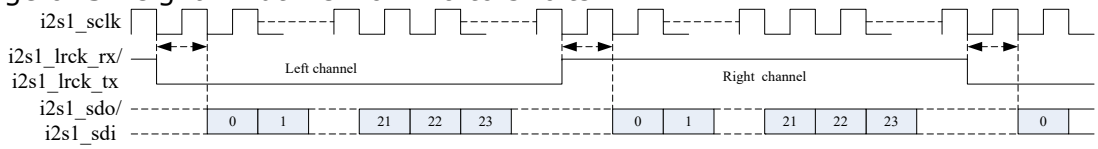


Fig. 20-4 I2S normal mode timing format

20.3.2 I2S left justified mode

This is the waveform of I2S left justified mode. For LRCK (i2s1_lrck_rx / i2s1_lrck_tx) signal, it goes high to indicate left channel and low to right channel. For SD (i2s1_sdo, i2s1_sdi) signal, it starts sending the first bit (MSB or LSB) at the same time when LRCK changes. The range of SD signal width is from 16 to 32bits.

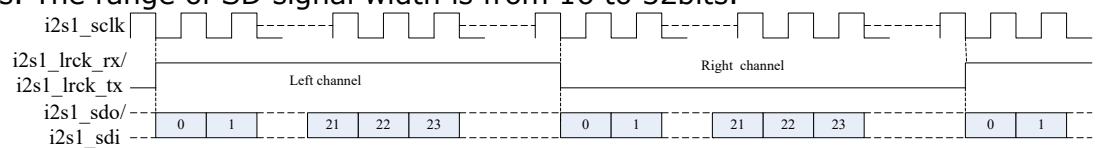


Fig. 20-5 I2S left justified mode timing format

20.3.3 I2S right justified mode

This is the waveform of I2S right justified mode. For LRCK (i2s1_lrck_rx/ i2s1_lrck_tx) signal, it goes high to indicate left channel and low to right channel. For SD (i2s1_sdo,

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

i2s1_sdi) signal, it transfers MSB or LSB first; but what is different from I2S normal or left justified mode, the last bit of the transferred data is aligned to the transition edge of the LRCK signal while one bit is transferred at one SCLK cycle. The range of SD signal width is from 16 to 32bits.

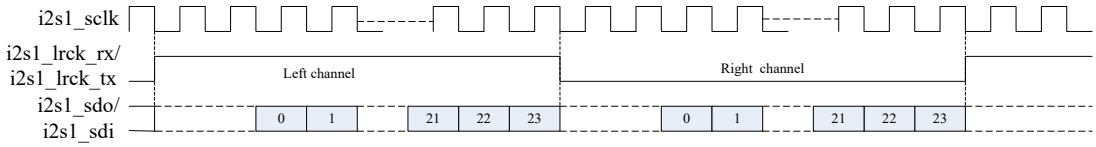


Fig. 20-6 I2S right justified mode timing format

20.3.4 PCM early mode

This is the waveform of PCM early mode. For LRCK (i2s1_lrck_rx/i2s1_lrck_tx) signal, it goes high to indicate the start of a group of audio channels. For SD (i2s1_sdo, i2s1_sdi) signal, it sends the first bit (MSB or LSB) at the same time when LRCK goes high. The range of SD signal width is from 16 to 32bits.

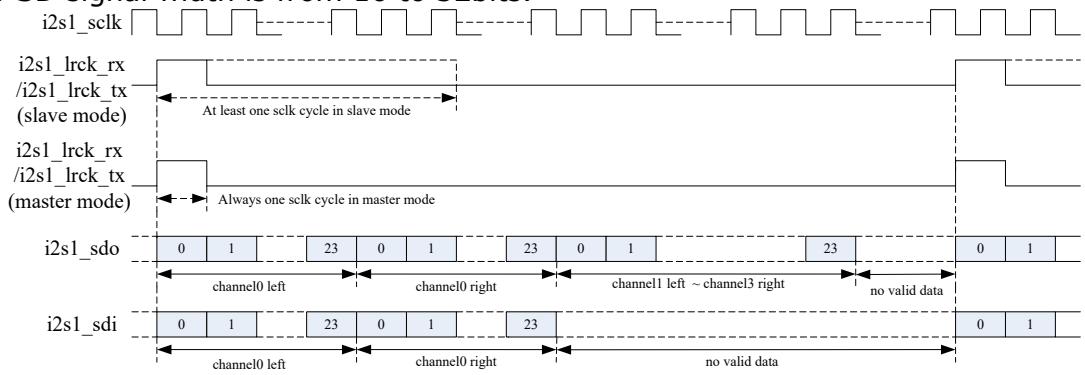


Fig. 20-7 PCM early mode timing format

20.3.5 PCM late1 mode

This is the waveform of PCM early mode. For LRCK (i2s1_lrck_rx/i2s1_lrck_tx) signal, it goes high to indicate the start of a group of audio channels. For SD (i2s1_sdo, i2s1_sdi) signal, it sends the first bit (MSB or LSB) one SCLK clock cycle after LRCK goes high. The range of SD signal width is from 16 to 32bits.

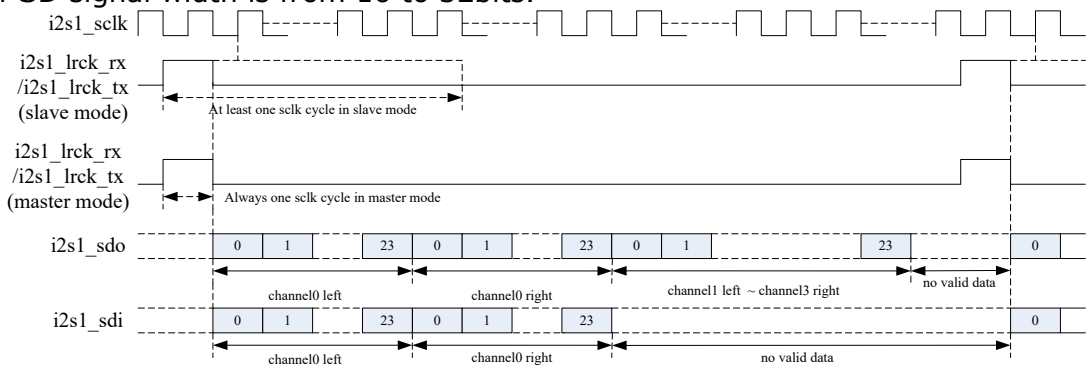


Fig. 20-8 PCM late1 mode timing format

20.3.6 PCM late2 mode

This is the waveform of PCM early mode. For LRCK (i2s1_lrck_rx/i2s1_lrck_tx) signal, it goes high to indicate the start of a group of audio channels. For SD (i2s1_sdo, i2s1_sdi) signal, it sends the first bit (MSB or LSB)two SCLK clock cycles after LRCK goes high. The range of SD signal width is from 16 to 32bits.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

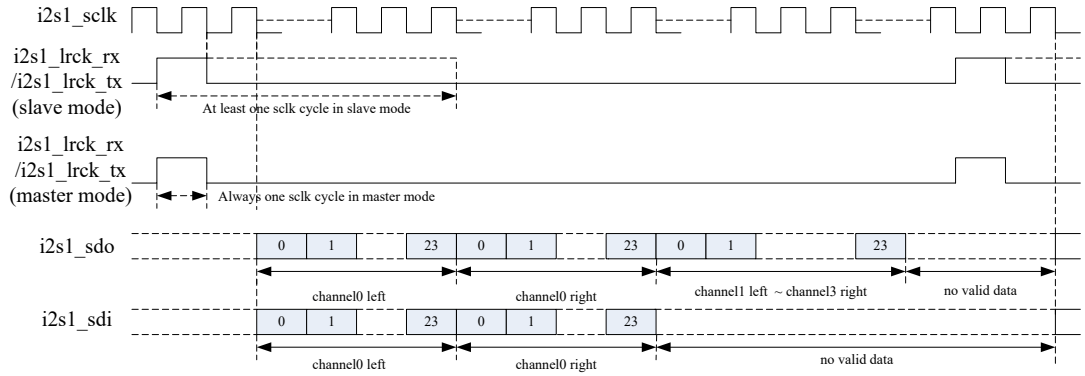


Fig. 20-9 PCM late2 mode timing format

20.3.7 PCM late3 mode

This is the waveform of PCM early mode. For LRCK (i2s1_lrck_rx/i2s1_lrck_tx) signal, it goes high to indicate the start of a group of audio channels. For SD (i2s1_sdo, i2s1_sdi) signal, it sends the first bit (MSB or LSB) three SCLK clock cycles after LRCK goes high. The range of SD signal width is from 16 to 32bits.

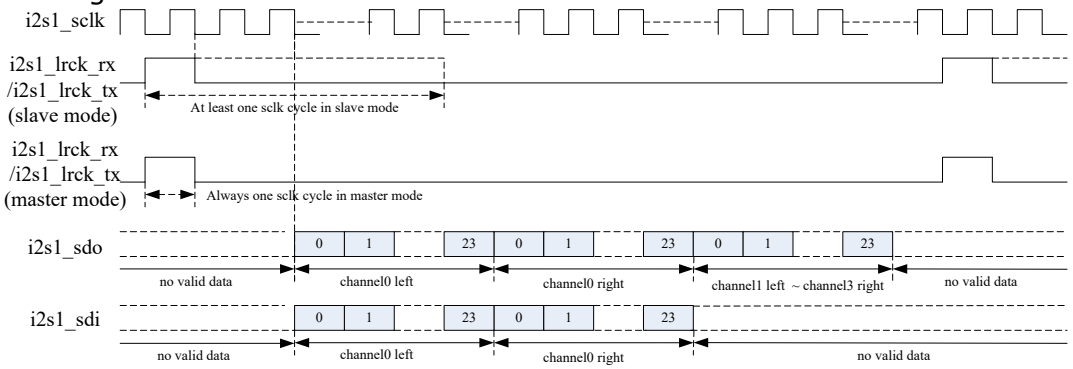
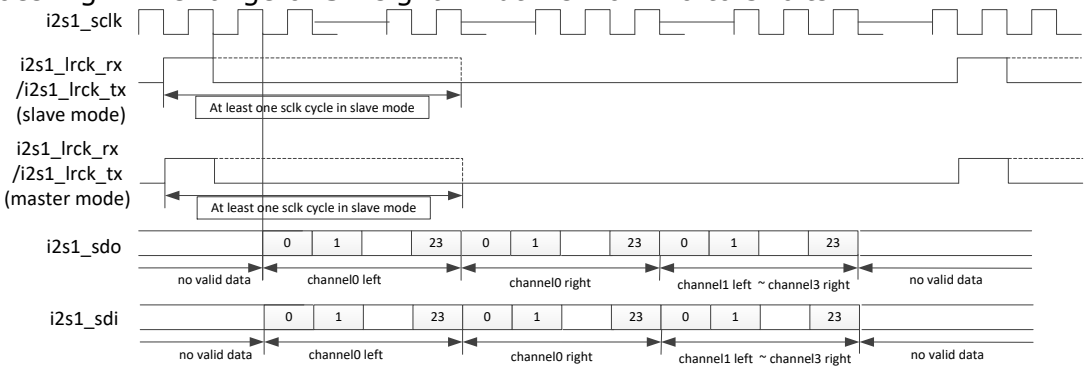


Fig. 20-10 PCM late3 mode timing format

20.3.8 TDM normal mode (PCM format)

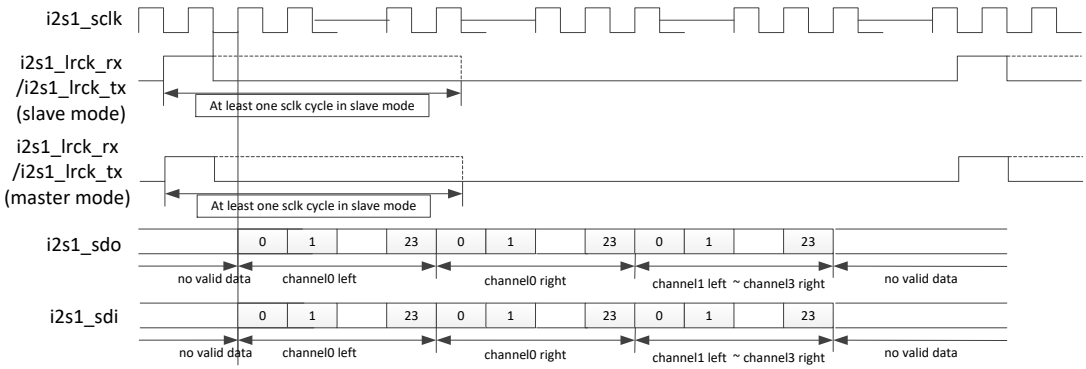
This is the waveform of TDM normal mode. For LRCK (i2s1_lrck_rx/i2s1_lrck_tx) signal, it goes high to indicate the start of a group of audio channels. For SD (i2s1_sdo, i2s1_sdi) signal, it sends the first bit (MSB or LSB) on the second falling edge of SCLK after LRCK goes high. The range of SD signal width is from 16 to 32bits.



20.3.9 TDM left shift mode0 (PCM format)

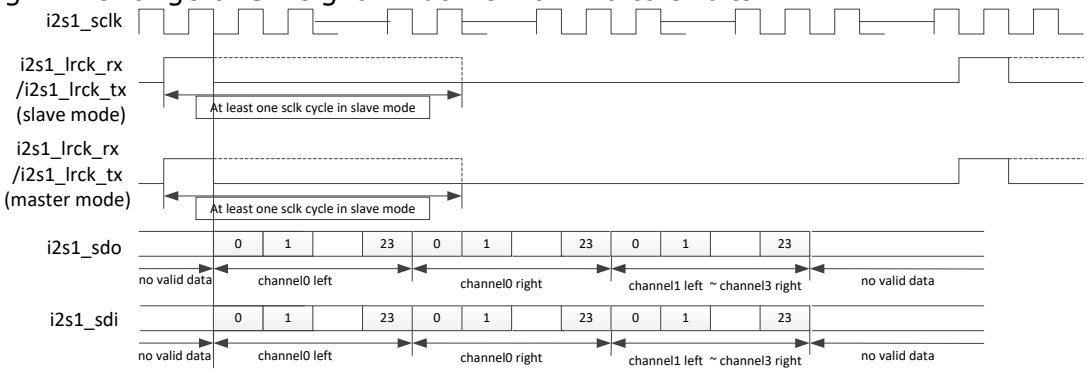
This is the waveform of PCM early mode. For LRCK (i2s1_lrck_rx/i2s1_lrck_tx) signal, it goes high to indicate the start of a group of audio channels. For SD (i2s1_sdo, i2s1_sdi) signal, it sends the first bit (MSB or LSB) on the second rising edge of SCLK after LRCK goes high. The range of SD signal width is from 16 to 32bits.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.



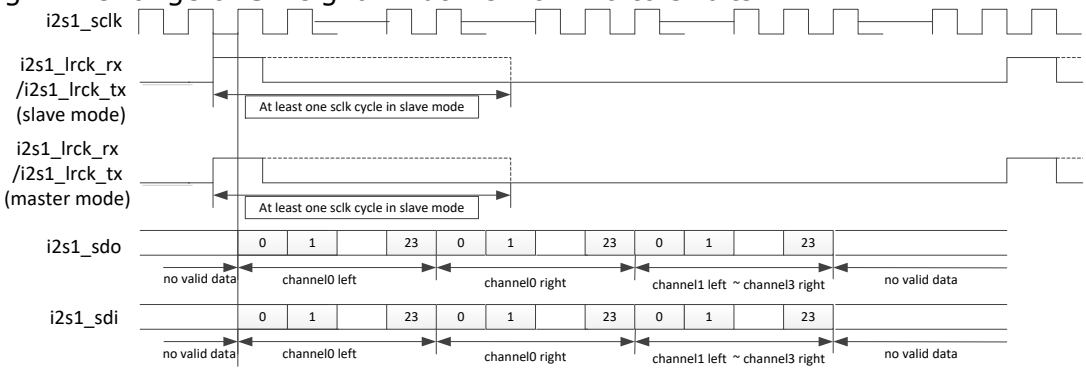
20.3.10 TDM left shift mode1 (PCM format)

This is the waveform of PCM early mode. For LRCK (i2s1_lrck_rx/i2s1_lrck_tx) signal, it goes high to indicate the start of a group of audio channels. For SD (i2s1_sdo, i2s1_sdi) signal, it sends the first bit (MSB or LSB) on the first falling edge of SCLK after LRCK goes high. The range of SD signal width is from 16 to 32bits.



20.3.11 TDM left shift mode2 (PCM format)

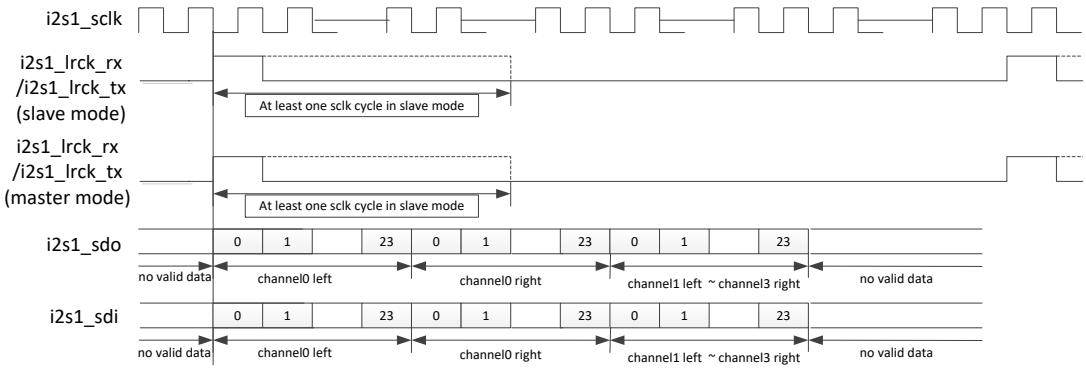
This is the waveform of PCM early mode. For LRCK (i2s1_lrck_rx/i2s1_lrck_tx) signal, it goes high to indicate the start of a group of audio channels. For SD (i2s1_sdo, i2s1_sdi) signal, it sends the first bit (MSB or LSB) on the first rising edge of SCLK after LRCK goes high. The range of SD signal width is from 16 to 32bits.



20.3.12 TDM left shift mode3 (PCM format)

This is the waveform of PCM early mode. For LRCK (i2s1_lrck_rx/i2s1_lrck_tx) signal, it goes high to indicate the start of a group of audio channels. For SD (i2s1_sdo, i2s1_sdi) signal, it sends the first bit (MSB or LSB) at the same time when LRCK goes high. The range of SD signal width is from 16 to 32bits.

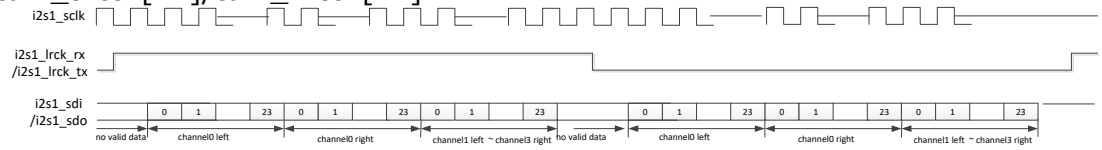
Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.



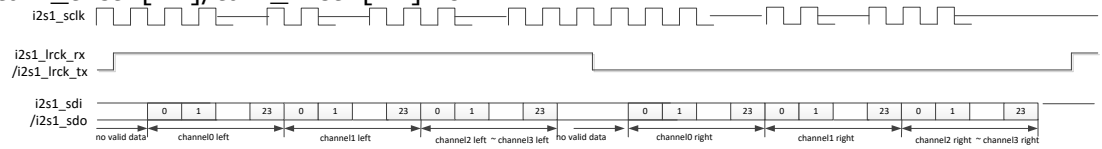
20.3.13 TDM normal mode (I2S format)

This is the waveform of I2S normal mode. For SD (i2s1_sdo, i2s1_sdi) signal, it starts sending the first bit (MSB or LSB) on the first falling edge of SCLK after LRCK changes. The range of SD signal width is from 16 to 32bits.

tdm_txctrl[17]/tdm_rxctrl[17]=1:



tdm_txctrl[17]/tdm_rxctrl[17]=0:

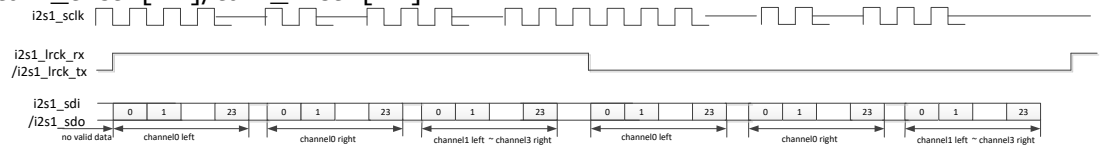


20.3.14 TDM left justified mode (I2S format)

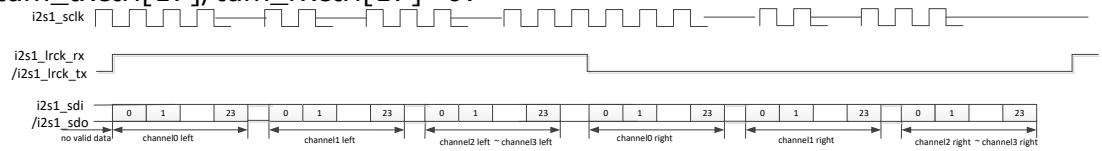
This is the waveform of I2S left justified mode. For SD (i2s1_sdo, i2s1_sdi) signal, it starts sending the first bit (MSB or LSB) at the same time when LRCK changes. The range of SD signal width is from 16 to 32bits.



tdm_txctrl[17]/tdm_rxctrl[17]=1:



tdm_txctrl[17]/tdm_rxctrl[17]=0:

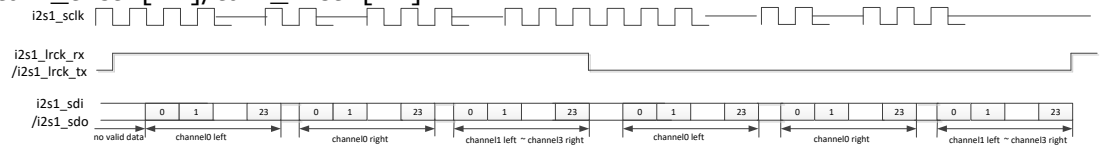


20.3.15 TDM right justified mode (I2S format)

This is the waveform of I2S right justified mode. For SD (i2s1_sdo, i2s1_sdi) signal, it transfers MSB or LSB first; but what is different from I2S normal or left justified mode.The range of SD signal width is from 16 to 32bits.

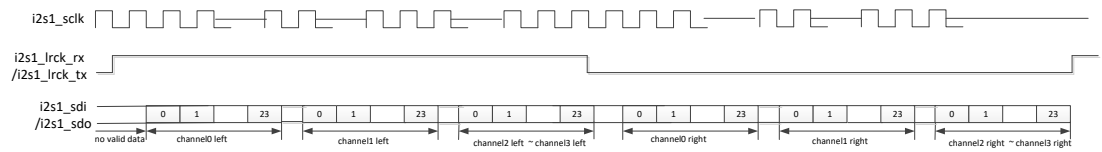


tdm_txctrl[17]/tdm_rxctrl[17]=1:



tdm_txctrl[17]/tdm_rxctrl[17]=0:

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.



20.4 Register description

20.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
I2S_TDM_8CH_TXCR	0x0000	W	0x72000000	Transmit Operation Control Register
I2S_TDM_8CH_RXCR	0x0004	W	0x00480000	Receive Operation Control Register
I2S_TDM_8CH_CKR	0x0008	W	0x00000000	Clock Generation Register
I2S_TDM_8CH_TXFIFOLR	0x000c	W	0x00000000	TX FIFO Level Register
I2S_TDM_8CH_DMACR	0x0010	W	0x00000000	DMA Control Register
I2S_TDM_8CH_INTCR	0x0014	W	0x00000000	Interrupt Control Register
I2S_TDM_8CH_INTSR	0x0018	W	0x00000000	Interrupt Status Register
I2S_TDM_8CH_XFER	0x001c	W	0x00000000	Transfer Start Register
I2S_TDM_8CH_CLR	0x0020	W	0x00000000	Sclk Domain Logic Clear Register
I2S_TDM_8CH_TXDR	0x0024	W	0x00000000	Transimt FIFO Data Register
I2S_TDM_8CH_RXDR	0x0028	W	0x00000000	Receive FIFO Data Register
I2S_TDM_8CH_RXFIFOLR	0x002c	W	0x00000000	RX FIFO Level Register
I2S_TDM_8CH_TDM_TXCTRL	0x0030	W	0x00000000	TDM Mode Transmit Operation Control Register
I2S_TDM_8CH_TDM_RXCTRL	0x0034	W	0x00003eff	TDM Mode Receive Operation Control Register
I2S_TDM_8CH_CLKDIV	0x0038	W	0x00000000	Clock Divider Register
I2S_TDM_8CH_VERSION	0x003c	W	0x00000000	Version Register

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

20.4.2 Detail Register Description

I2S_TDM_8CH_TXCR

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31	RO	0x0	Reserved
30:29	RW	0x3	tx_path_select3 Tx Path Select 2'b00: SDO3 output data from path0 2'b01: SDO3 output data from path1 2'b10: SDO3 output data from path2 2'b11: SDO3 output data from path3 Note: When TDM mode, only path0 enable.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
28:27	RW	0x2	tx_path_select2 Tx Path Select 2'b00: SDO2 output data from path0 2'b01: SDO2 output data from path1 2'b10: SDO2 output data from path2 2'b11: SDO2 output data from path3 Note: When TDM mode, only path0 enable.
26:25	RW	0x1	tx_path_select1 Tx Path Select 2'b00: SDO1 output data from path0 2'b01: SDO1 output data from path1 2'b10: SDO1 output data from path2 2'b11: SDO1 output data from path3 Note: When TDM mode, only path0 enable.
24:23	RW	0x0	tx_path_select0 Tx Path Select 2'b00: SDO0 output data from path0 2'b01: SDO0 output data from path1 2'b10: SDO0 output data from path2 2'b11: SDO0 output data from path3 Note: When TDM mode, only path0 enable.
22:17	RW	0x00	rcnt (Can be written only when XFER[0] bit is 0.) Only vailid in I2S Right justified format and slave tx mode is selected. Start to transmit data RCNT sclk cycles after left channel valid. Note: Only function when TX TFS[1]=0
16:15	RW	0x0	tcsr Transmit Channel Select Register 2'b00: Two channel 2'b01: Four channel 2'b10: Six channel 2'b11: Eight channel
14	RW	0x0	hwt Halfword Word Transform (Can be written only when XFER[0] bit is 0.) Only valid when VDW select 16bit data. 1'b0: 32 bit data valid from AHB/APB bus. Low 16 bit for left channel and high 16 bit for right channel. 1'b1: Low 16bit data valid from AHB/APB bus, high 16 bit data invalid.
13	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
12	RW	0x0	sjm Store Justified Mode (Can be written only when XFER[0] bit is 0.) 16bit~31bit DATA stored in 32 bits width fifo. If VDW select 16bit data, this bit is valid only when HWT select 0.Because if HWT is 1, every fifo unit contain two 16bit data and 32 bit space is full, it is impossible to choose justified mode. 1'b0: Right justified 1'b1: Left justified
11	RW	0x0	fbm First Bit Mode (Can be written only when XFER[0] bit is 0.) 1'b0: MSB 1'b1: LSB
10:9	RW	0x0	ibm I2S Bus Mode (Can be written only when XFER[0] bit is 0.) 2'b00: I2S normal 2'b01: I2S Left justified 2'b10: I2S Right justified 2'b11: Reserved
8:7	RW	0x0	pbm (Can be written only when XFER[0] bit is 0.) 2'b00: PCM no delay mode 2'b01: PCM delay 1 mode 2'b10: PCM delay 2 mode 2'b11: PCM delay 3 mode Note: Function when TX TFS[1:0] is 1
6:5	RW	0x0	tfs (Can be written only when XFER[0] bit is 0.) 2'b00: I2S format 2'b01: PCM format 2'b10: TDM format 0 (PCM mode) 2'b11: TDM format 1 (I2S mode)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4:0	RW	0x00	vdw Valid Data Width (Can be written only when XFER[0] bit is 0.) 5'h0~5'he: Reserved 5'h0f: 16bit 5'h10: 17bit 5'h11: 18bit 5'h12: 19bit …… 5'f1c: 29bit 5'f1d: 30bit 5'f1e: 31bit 5'f1f: 32bit

I2S_TDM_8CH_RXCR

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:25	RO	0x0	Reserved
24:23	RW	0x0	rx_path_select3 Rx Path Select 2'b00: Path3 data from SDI0 2'b01: Path3 data from SDI1 2'b10: Path3 data from SDI2 2'b11: Path3 data from SDI3 Note: Inoperative at TDM mode.
22:21	RW	0x2	rx_path_select2 Rx Path Select 2'b00: Path2 data from SDI0 2'b01: Path2 data from SDI1 2'b10: Path2 data from SDI2 2'b11: Path2 data from SDI3 Note: Inoperative at TDM mode.
20:19	RW	0x1	rx_path_select1 Rx Path Select 2'b00: Path1 data from SDI0 2'b01: Path1 data from SDI1 2'b10: Path1 data from SDI2 2'b11: Path1 data from SDI3 Note: Inoperative at TDM mode.
18:17	RW	0x0	rx_path_select0 Rx Path Select 2'b00: Path0 data from SDI0 2'b01: Path0 data from SDI1 2'b10: Path0 data from SDI2 2'b11: Path0 data from SDI3

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
16:15	RW	0x0	rcsr Receive Channel Select Register 2'b00: Two channel 2'b01: Four channel 2'b10: Six channel 2'b11: Eight channel
14	RW	0x0	hwt Halfword Word Transform (Can be written only when XFER[1] bit is 0.) Only valid when VDW select 16bit data. 1'b0: 32 bit data valid to AHB/APB bus. Low 16 bit for left channel and high 16 bit for right channel. 1'b1: Low 16bit data valid to AHB/APB bus, high 16 bit data invalid.
13	RO	0x0	Reserved
12	RW	0x0	sjm Store Justified Mode (Can be written only when XFER[1] bit is 0.) 16bit~31bit DATA stored in 32 bits width fifo. If VDW select 16bit data, this bit is valid only when HWT select 0.Because if HWT is 1, every fifo unit contain two 16bit data and 32 bit space is full, it is impossible to choose justified mode. 1'b0: Right justified 1'b1: Left justified
11	RW	0x0	fbm First Bit Mode (Can be written only when XFER[1] bit is 0.) 1'b0: MSB 1'b1: LSB
10:9	RW	0x0	ibm I2S Bus Mode (Can be written only when XFER[1] bit is 0.) 2'b00: I2S normal 2'b01: I2S Left justified 2'b10: I2S Right justified 2'b11: Reserved
8:7	RW	0x0	pbm PCM Bus Mode (Can be written only when XFER[1] bit is 0.) 2'b00: PCM no delay mode 2'b01: PCM delay 1 mode 2'b10: PCM delay 2 mode 2'b11: PCM delay 3 mode

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
6:5	RW	0x0	tfs Transfer Format Select (Can be written only when XFER[1] bit is 0.) 2'b00: I2S format 2'b01: PCM format 2'b10: TDM format 0 (PCM mode) 2'b11: TDM format 1 (I2S mode)
4:0	RW	0x00	vdw Valid Data Width (Can be written only when XFER[1] bit is 0.) 5'b00000~5'b01110: Reserved 5'b01111: 16bit 5'b10000: 17bit 5'b10001: 18bit 5'b10010: 19bit …… 5'b11100: 29bit 5'b11101: 30bit 5'b11110: 31bit 5'b11111: 32bit

I2S_TDM_8CH_CKR

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	Reserved
29:28	RW	0x0	lrck_common Tx and Rx Common Use 2'b00/2'b11: tx_lrck/rx_lrck are used as synchronous signal for TX /RX respectively. 2'b01: Only tx_lrck is used as synchronous signal for TX and RX. 2'b10: Only rx_lrck is used as synchronous signal for TX and RX.
27	RW	0x0	mss Master/Slave Mode Select (Can be written only when XFER[1] or XFER[0] bit is 0.) 1'b0: Master mode(sclk output) 1'b1: Slave mode(sclk input)
26	RW	0x0	ckp Sclk Polarity (Can be written only when XFER[1] or XFER[0] bit is 0.) 1'b0: Sample data at posedge sclk and drive data at negedge sclk 1'b1: Sample data at negedge sclk and drive data at posedge sclk

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
25	RW	0x0	rlp Receive Lrck Polarity (Can be written only when XFER[1] or XFER[0] bit is 0.) 1'b0: Normal polartiy (I2S normal: Low for left channel, high for right channel I2S left/right just: High for left channel, low for right channel PCM start signal: High valid) 1'b1: Oppsite polarity (I2S normal: High for left channel, low for right channel I2S left/right just: Low for left channel, high for right channel PCM start signal: Low valid)
24	RW	0x0	tlp Transmit Lrck Polarity (Can be written only when XFER[1] or XFER[0] bit is 0.) 1'b0: Normal polartiy (I2S normal: Low for left channel, high for right channel I2S left/right just: High for left channel, low for right channel PCM start signal: High valid) 1'b1: Oppsite polarity (I2S normal: High for left channel, low for right channel I2S left/right just: Low for left channel, high for right channel PCM start signal: Low valid)
23:16	RO	0x0	Reserved
15:8	RW	0x00	rsd Receive Sclk Divider (Can be written only when XFER[1] or XFER[0] bit is 0.) 8'h00~8'h1e: Reserved 8'h1f~8'hff: Frequency of sclk = (receive sclk divider/2)*2*frequency of rx_lrck Note: Function when RX TFS[1:0] is 2'b00 or 2'b01;
7:0	RW	0x00	tsd Transmit Sclk Divider (Can be written only when XFER[1] or XFER[0] bit is 0.) 8'h00~8'h1e: Reserved 8'h1f~8'hff: Frequency of sclk = (Transmit sclk divider/2)*2*frequency of tx_lrck Note: Function when TX TFS[1:0] is 2'b00 or 2'b01;

I2S_TDM_8CH_TXFIFOLR

Address: Operational Base + offset (0x000c)

Bit	Attr	Reset Value	Description
31:24	RO	0x0	Reserved
23:18	RW	0x00	tfl3 Transmit FIFO3 Level Contains the number of valid data entries in the transmit FIFO3.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
17:12	RW	0x00	tfl2 Transmit FIFO2 Level Contains the number of valid data entries in the transmit FIFO2.
11:6	RW	0x00	tfl1 Transmit FIFO1 Level Contains the number of valid data entries in the transmit FIFO1.
5:0	RO	0x00	tfl0 Transmit FIFO0 Level Contains the number of valid data entries in the transmit FIFO0.

I2S_TDM_8CH_DMACR

Address: Operational Base + offset (0x0010)

Bit	Attr	Reset Value	Description
31:25	RO	0x0	Reserved
24	RW	0x0	rde Receive DMA Enable 1'b0: Receive DMA disabled 1'b1: Receive DMA enabled
23:21	RO	0x0	Reserved
20:16	RW	0x00	rdl Receive Data Level This bit field controls the level at which a DMA request is made by the receive logic. The watermark level = DMARDL+1; that is, dma_rx_req is generated when the number of valid data entries in the receive FIFO is equal to or above this field value + 1.
15:9	RO	0x0	Reserved
8	RW	0x0	tde Transmit DMA Enable 1'b0: Transmit DMA disabled 1'b1: Transmit DMA enabled
7:5	RO	0x0	Reserved
4:0	RW	0x00	tdl Transmit Data Level This bit field controls the level at which a DMA request is made by the transmit logic. It is equal to the watermark level; that is, the dma_tx_req signal is generated when the number of valid data entries in the TXFIFO(TXFIFO0 if CSR=00;TXFIFO1 if CSR=01,TXFIFO2 if CSR=10,TXFIFO3 if CSR=11)is equal to or below this field value.

I2S_TDM_8CH_INTCR

Address: Operational Base + offset (0x0014)

Bit	Attr	Reset Value	Description
31:25	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
24:20	RW	0x00	rft Receive FIFO Threshold When the number of receive FIFO entries is more than or equal to this threshold plus 1, the receive FIFO full interrupt is triggered.
19	RO	0x0	Reserved
18	WO	0x0	rxoic RX Overrun Interrupt Clear Write 1 to clear RX overrun interrupt.
17	RW	0x0	rxoie RX Overrun Interrupt Enable 1'b0: Disable 1'b1: Enable
16	RW	0x0	rxfie RX Full Interrupt Enable 1'b0: Disable 1'b1: Enable
15:9	RO	0x0	Reserved
8:4	RW	0x00	tft Transmit FIFO Threshold When the number of transmit FIFO entries is less than or equal to this threshold, the transmit FIFO empty interrupt is triggered.
3	RO	0x0	Reserved
2	RW	0x0	txuic TX Underrun Interrupt Clear Write 1 to clear TX underrun interrupt.
1	RW	0x0	txuie TX Underrun Interrupt Enable 1'b0: Disable 1'b1: Enable
0	RW	0x0	txeie TX empty Interrupt Enable 1'b0: Disable 1'b1: Enable

I2S_TDM_8CH_INTSR

Address: Operational Base + offset (0x0018)

Bit	Attr	Reset Value	Description
31:18	RO	0x0	Reserved
17	RW	0x0	rxoi RX Overrun Interrupt 1'b0: Inactive 1'b1: Active

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
16	RO	0x0	rxfi RX Full Interrupt 1'b0: Inactive 1'b1: Active
15:2	RO	0x0	Reserved
1	RO	0x0	txui TX Underrun Interrupt 1'b0: Inactive 1'b1: Active
0	RO	0x0	txei TX Empty Interrupt 1'b0: Inactive 1'b1: Active

I2S_TDM_8CH_XFER

Address: Operational Base + offset (0x001c)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	Reserved
1	RW	0x0	rxs RX Start Bit 1'b0: Stop RX transfer 1'b1: Start RX transfer
0	RW	0x0	txs TX Transfer Start Bit 1'b0: Stop TX transfer 1'b1: Start TX transfer

I2S_TDM_8CH_CLR

Address: Operational Base + offset (0x0020)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	Reserved
1	RW	0x0	rxc RX Logic Clear This is a self-cleared bit. Write 1 to clear all receive logic.
0	RW	0x0	txc TX Logic Clear This is a self-cleared bit. Write 1 to clear all transmit logic.

I2S_TDM_8CH_TXDR

Address: Operational Base + offset (0x0024)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	txdr Transimt FIFO Data Register When it is written to, data are moved into the transmit FIFO.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

I2S_TDM_8CH_RXDR

Address: Operational Base + offset (0x0028)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	rxdr Receive FIFO Data Register When the register is read, data in the receive FIFO is accessed.

I2S_TDM_8CH_RXFIFOLR

Address: Operational Base + offset (0x002c)

Bit	Attr	Reset Value	Description
31:24	RO	0x0	Reserved
23:18	RW	0x00	rfl3 Receive FIFO3 Level Contains the number of valid data entries in the Receive FIFO3.
17:12	RW	0x00	rfl2 Receive FIFO2 Level Contains the number of valid data entries in the Receive FIFO2.
11:6	RW	0x00	rfl1 Receive FIFO1 Level Contains the number of valid data entries in the Receive FIFO1.
5:0	RW	0x00	rfl0 Receive FIFO0 Level Contains the number of valid data entries in the Receive FIFO0.

I2S_TDM_8CH_TDM_TXCTRL

Address: Operational Base + offset (0x0030)

Bit	Attr	Reset Value	Description
31:21	RO	0x0	Reserved
20:18	RW	0x0	tx_tdm_fsync_width_sel1 TDM Transfer Fsync Width Sel1 (Can be written only when XFER[0] is 0.) 3'b000: Single period of the sclk_tx. 3'b001: 2 period of the sclk_tx. n: n+1 period of the sclk_tx. 3'b110: 7 period of the sclk_tx. 3'b111: The width is equivalent to a channel block. Note: Function when TX TFS[1:0] is 2 or 3.
17	RW	0x0	tx_tdm_fsync_width_sel0 TDM Transfer Fsync Width Sel0 (Can be written only when XFER[0] is 0.) 1'b0: 1/2 frame width. It should be set to an even number. 1'b1: Frame width

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
16:14	RW	0x0	tdm_tx_shift_ctrl TDM Transfer Shift Ctrl (Can be written only when XFER[0] is 0.) 3'b000: PCM format 0: Normal mode, drive data on the second negedge of sclk_tx after rising edge of TX LRCK. I2S format 0: Normal mode 3'b001: PCM format 1: 1/2 cycle shift left, drive data on second posedge of sclk_tx after rising edge of TX LRCK. I2S format 1: Left justified mode 3'b010: PCM format 2: 1 cycle shift left, drive data on first negedge of sclk_tx after rising edge of TX LRCK. I2S format 2: right justified mode 3'b011: PCM format 3: 3/2 cycle shift left, drive data on first posedge of sclk_tx after rising edge of TX LRCK. I2S format: Not support 3'b100: PCM format 4: 2 cycle shift left, drive data aligned to the posedge of TX LRCK. I2S format: Not support 3'b101~3'b111: Not support Note: Function when TX TFS[1:0] is 2 or 3.
13:9	RW	0x00	tdm_tx_slto_bit TDM Transfer Slot Bits (Can be written only when XFER[0] is 0.) 5'h00~5'h0e: Reserved 5'h0f: 16bit 5'h10: 17bit 5'h11: 18bit 5'h12: 19bit …… 5'h1f: 32bit Note: Function when TX TFS[1:0] is 2 or 3.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
8:0	RW	0x000	tdm_tx_frame_width TDM Transfer Frame Width (Can be written only when XFER[0] is 0.) 9'h000~9'h01e:Reserved 9'h01f: 32bit 9'h020: 33bit 9'h021: 34bit 9'h022: 35bit …… 9'h1ff: 512bit Note: Functional when TX TFS[1:0] is 2 or 3.

I2S_TDM_8CH_TDM_RXCTRL

Address: Operational Base + offset (0x0034)

Bit	Attr	Reset Value	Description
31:21	RO	0x0	Reserved
20:18	RW	0x0	rx_tdm_fsync_width_sel1 TDM Receive Fsync Width Sel1 (Can be written only when XFER[1] is 0.) 3'b000: Single period of the sclk_rx 3'b001: 2 period of the sclk_rx n: n+1 period of the sclk_rx 3'b110: 7 period of the sclk_rx 3'b111: The width is equivalent to a channel block Note: Function when RX TFS[1:0] is 2 or 3.
17	RW	0x0	rx_tdm_fsync_width_sel0 TDM Receive Fsync Width Sel0 (Can be written only when XFER[1] is 0.) 1'b0: 1/2 frame width. It should be set to an even number 1'b1: Frame width

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
16:14	RW	0x0	tdm_rx_shift_ctrl TDM Receive Shift Ctrl (Can be written only when XFER[1] is 0.) 3'b000: PCM format 0: Normal mode, sample data on the third posedge of sclk_rx after rising edge of RX LRCK. I2S format 0: Normal mode 3'b001: PCM format 1: 1/2 cycle shift left, sample data on second negedge of sclk_rx after rising edge of RX LRCK. I2S format 1: Left justified mode 3'b010: PCM format 2: 1 cycle shift left, sample data on second posedge of sclk_rx after rising edge of RX LRCK. I2S format 2: Right justified mode 3'b011: PCM format 3: 3/2 cycle shift left, sample data on first negedge of sclk_rx after rising edge of RX LRCK. I2S format: Not support 3'b100: PCM format 4: 2 cycle shift left, sample data on the first posedge of sclk_rx after rising edge of RX LRCK. I2S format: Not support 3'b101~3'b111: Not support Note: Function when RX TFS[1:0] is 2 or 3.
13:9	RW	0x1f	tdm_rx_slot_bit_width TDM Receive Slot Bits (Can be written only when XFER[1] is 0.) 5'h00~5'h0e: Reserved 5'h0f: 16bit 5'h10: 17bit 5'h11: 18bit 5'h12: 19bit …… 5'h1f: 32bit Note: Function when RX TFS[1:0] is 2 or 3.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
8:0	RW	0x0ff	tdm_rx_frame_width TDM Receive Frame Width (Can be written only when XFER[1] is 0.) 9'h000~9'h01e:Reserved 9'h01f: 32bit 9'h020: 33bit 9'h021: 34bit 9'h022: 35bit …… 9'h1ff: 512bit Note: Functional when RX TFS[1:0] is 2 or 3.

I2S_TDM_8CH_CLKDIV

Address: Operational Base + offset (0x0038)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	Reserved
15:8	RW	0x00	rx_mdiv RX Mclk Divider (Can be written only when XFER[1] bit is 0.) mclk_rx divider = (mclk_rx/sclk_rx)-1. For example, if mclk_rx divider is 5, then the frequency of sclk_rx is mclk_rx/6.
7:0	RW	0x00	tx_mdiv TX Mclk Divider (Can be written only when XFER[0] bit is 0.) mclk_tx divider = (mclk_tx/sclk_tx)-1. For example, if mclk_tx divider is 5, then the frequency of sclk_tx is mclk_tx/6.

I2S_TDM_8CH_VERSION

Address: Operational Base + offset (0x003c)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	i2s_version I2s Version

20.5 Interface Description

Table 20-1 I2S_8CH_TDM Interface Description

Module Pin	Direction	Pad Name	IOMUX Setting
i2s0_8ch_mclk	I/O	GPIO3_B5/I2S0_MCLK/ISP_SHUTTEREN	GRF_GPIO3B_IOMUX_H[6:4]=3'b1
i2s0_8ch_sclk_rx	I/O	GPIO3_B0/I2S0_SCLK_RX/PDM_CLK0	GRF_GPIO3B_IOMUX_L[2:0]=3'b1
i2s0_8ch_sclk_tx	I/O	GPIO3_B7/I2S0_SCLK_TX/ISP_PRELIGHTTRIG	GRF_GPIO3B_IOMUX_H[14:12]=3'b1
i2s0_8ch_lrck_rx	I/O	GPIO3_B1/I2S0_LRCK_RX/PDM_CLK1	GRF_GPIO3B_IOMUX_L[6:4]=3'b1

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Module Pin	Direction	Pad Name	IOMUX Setting
i2s0_8ch_lrck_tx	I/O	GPIO3_B6/I2S0_LRCK_TX/ISP_FLASHTRIGOUT	GRF_GPIO3B_IOMUX_H[10:8] =3'b1
i2s_8ch_sdo0	O	GPIO3_C0/I2S0_SDO0/ISP_SHUTTERTRIG	GRF_GPIO3C_IOMUX_L[2:0] =3'b1
i2s_8ch_sdo1	O	GPIO3_B4/I2S0_SDO1/I2C2_SDA_M0	GRF_GPIO3B_IOMUX_H[2:0] =3'b1
i2s_8ch_sdo2	O	GPIO3_B3/I2S0_SDO2/I2C2_SCL_M0/LCDC_VSYNC_M1	GRF_GPIO3B_IOMUX_L[14:12] =3'b1
i2s_8ch_sdo3	O	GPIO3_B2/I2S0_SDO3/ISP_FLASHTRIGIN/LCDC_HSYNC_M1	GRF_GPIO3B_IOMUX_L[10:8] =3'b1
i2s2_8ch_sdi0	I	GPIO3_C1/I2S0_SDI0/PDM_SDI0	GRF_GPIO3C_IOMUX_H[6:4] =3'b1
i2s2_8ch_sdi1	I	GPIO3_A7/I2S0_SDI1/PDM_SDI1	GRF_GPIO3A_IOMUX_H[14:12] =3'b1
i2s2_8ch_sdi2	I	GPIO3_A6/I2S0_SDI2/PDM_SDI2	GRF_GPIO3A_IOMUX_H[10:8] =3'b1
i2s2_8ch_sdi3	I	GPIO3_A5/I2S0_SDI3/PDM_SDI3	GRF_GPIO3A_IOMUX_H[6:4] =3'b1

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

20.6 Application Notes

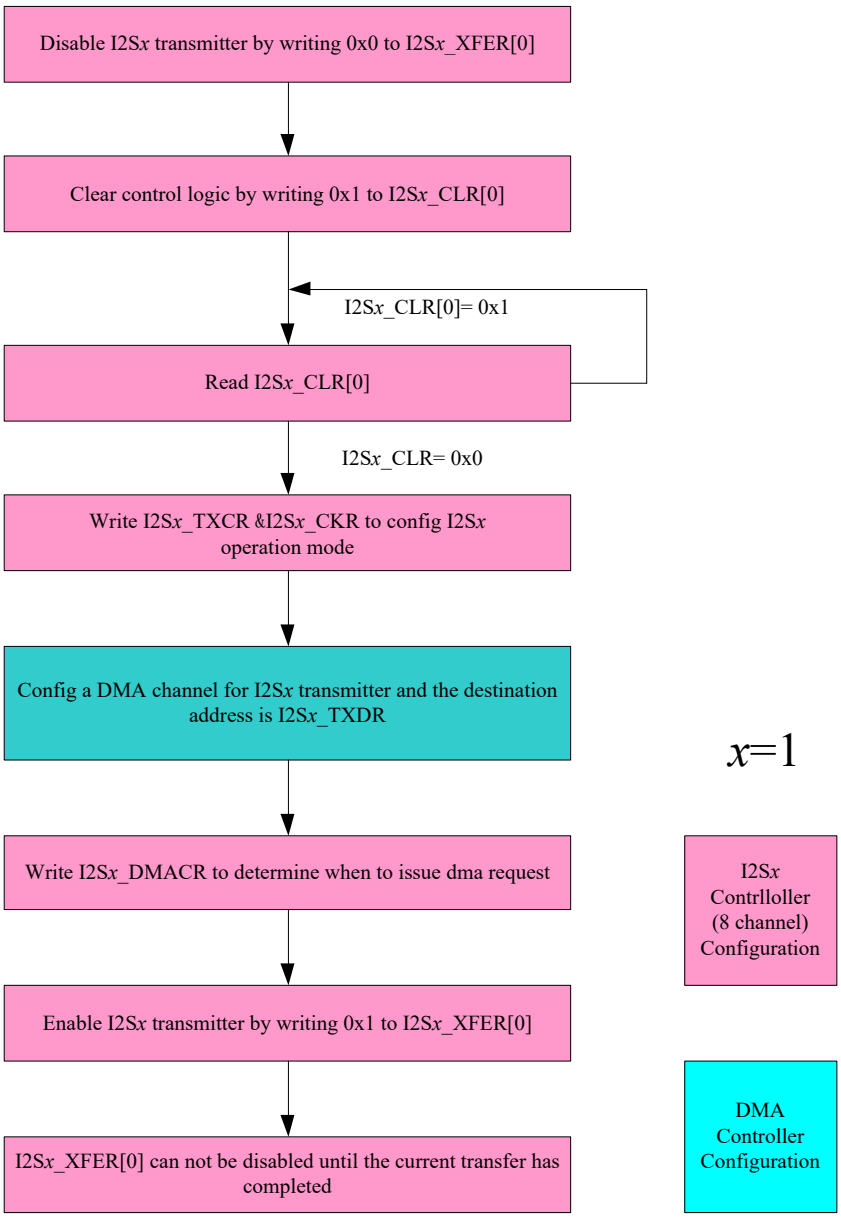


Fig. 20-11 I2S/PCM/TDM controller transmit operation flow chart

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

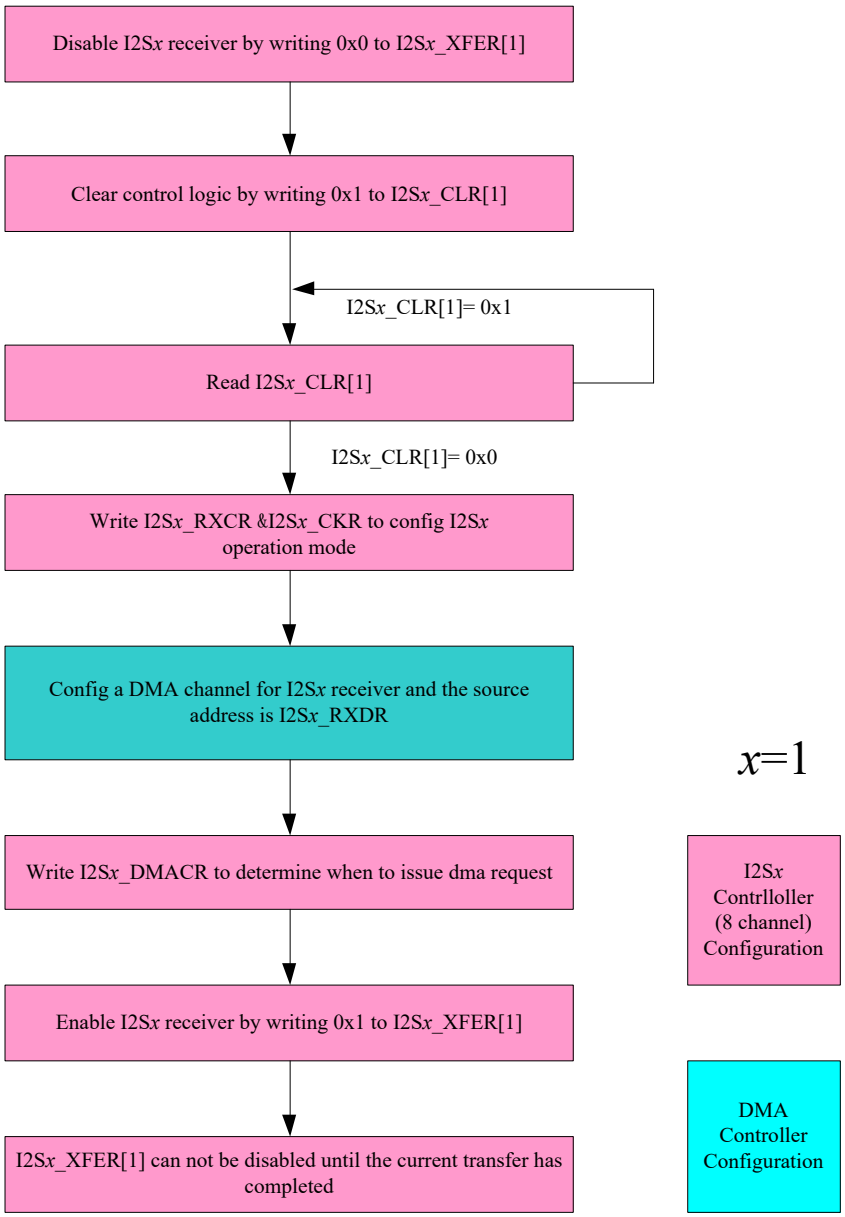


Fig. 20-12 I2S/PCM/TDM controller receive operation flow chart

Note: User should clear TX/RX logical by CLR[0]/CLR[1] and wait clear operation done before configure the other registers.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 21 SPI2APB

21.1 Overview

The SPI2APB module supports all the features of SPI (reference to chapter SPI), and add the following features:

- External SPI master can access the whole address space in the chip through SPI2APB convertor
- A transaction states register and a message register can be read by external SPI master
- Two 32 bit register scan be written by external SPI master, and one of them can generate an interrupt after been written

In RK1808, SPI2APB is also refer as SPI0.

21.2 Block Diagram

The figure below shows SPI2APB diagram.

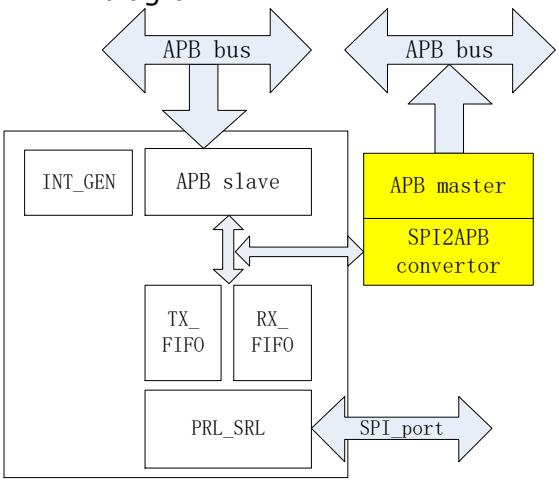


Fig. 21-1 SPI2APBBlock Diagram

21.3 Function Description

21.3.1 Overview

SPI2APB can work in two modes: Normal mode and Convertor mode. When in Normal mode, it work as a normal SPI. When in Convertor mode, the convertor block reads data from RX_FIFO, analysis the data, and operate according to the protocol.

After reset, SPI2APB works in Convertor mode. User can change work mode by modifying the value of WORK_MODE register.

21.3.2 Protocol Description

SPI2APB supports four types of operation: Write, Read, Query and Write Message. The command, address and data are all transferred in 32 bits packets, and the first bit to be transferred is the LSB. The operations begin at the negative edge of CS and end at the positive edge of CS.

Write

External SPI master should dessert CS firstly. The Write command, address and data packets should be transferred according to the order of the figure. The first data (data0 in the figure) will be written to the address ADDR, the second data will be written to the address ADDR+ 4, and so on. External SPI master can terminate the write operation by setting the CS to high level.

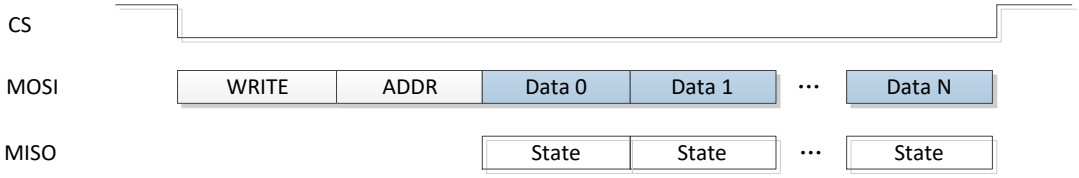


Fig. 21-2 Write operation

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Read

After receiving Read command and address, the convertor will read N data from the address and push the data into the RD_FIFO. Dummy packets can be insert to ensure there is available data in the RD_FIFO. When SPI master is ready to receive data, it should send a RD_BEGIN command packet. The first packets following RD_BEGIN packets (data0 in the figure) is the data read from address ADDR, the second packets is the data read from address ADDR+ 4, and so on. After N data packets has been transferred, SPI2APB will begin to transfer state information.

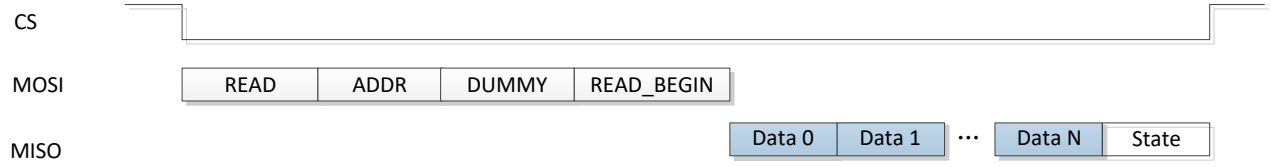


Fig. 21-3 Read operation

Query

There are two registers can be queried: State Register and Message Register 2. State register contains transfer state information such as whether the transfer is complete or not, detail is showed in the table. Message Register 2 can be written by DSP. The query result is immediately following the query command without delay.

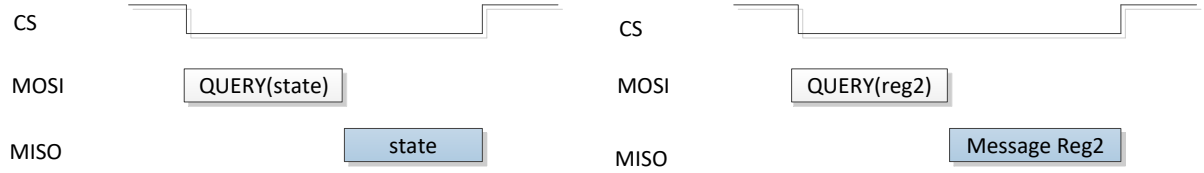


Fig. 21-4 Query operation

Bit	Attr	Reset Value	Description
[31:16]	RW	0x1608	Chip ID
[15:11]	RW	0x0	Reserved
10	RW	0x0	pre_err Error response is received when APB master is doing pre-read transaction
9	RW	0x0	rd_udflw Read FIFO is underflow
8	RW	0x0	rd_err Data with error response has been transmitted
[7:4]	RW	0x0	Reserved
3	RW	0x0	trans_cfl Transaction conflict. Previous transaction is unfinished and new transaction is launched by SPI master
2	RW	0x0	trans_unfi Transaction is unfinished
1	RW	0x0	wr_ovfl Write FIFO is overflow
0	RW	0x0	wr_err Error response is received when APB master is writing

WriteMessage

There are two message registers can be written by external SPI master: Message Register

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

0 and Message Register 1.These two registers can be read by DSP. After Message Register 1 has been written, an interrupt will be generated.



Fig. 21-5 Write message operation

Command encoding
All command are 32bits.

command	format	description
READ	0x0077+(PRE_NUM <<16)	PRE_NUM is pre_read number. For example, 0x00100077 indicates pre_read number is 16. If PRE_NUM is 0, convertor will read data when RD_FIFO is not full and read transaction is not terminated.
READ_BEGIN	0x000000aa	
WRITE	0x00000011	
WRITE MESSAGE (register 0)	0x00010011	
WRITE MESSAGE (register 1)	0x00020011	Write message register 1 will generate an interrupt
QUERY (state)	0x000000ff	
QUERY (Message register 2)	0x000001ff	

21.4 Register Description

21.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
SPI2APB_IMR	0x002c	W	0x00000000	Interrupt Mask
SPI2APB_ISR	0x0030	W	0x00000000	Interrupt Status
SPI2APB_RISR	0x0034	W	0x00000001	Raw Interrupt Status
SPI2APB_ICR	0x0038	W	0x00000000	Interrupt Clear
SPI2APB_WORK_MODE	0x004c	W	0x00000001	SPI to APB convertor enable
SPI2APB_MESSAGE_REG0	0x0050	W	0x00000000	Message register0
SPI2APB_MESSAGE_REG1	0x0054	W	0x00000000	Message register1
SPI2APB_MESSAGE_REG2	0x0058	W	0x00000000	Message register2DSP write to this register . AP read it .

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

21.4.2 Detail Register Description

SPI2APB_IMR
Address: Operational Base + offset (0x002c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:6	RO	0x0	Reserved
5	RW	0x0	QWIM Quick Write Interrupt Mask 1'b0: spi2apb message register 1 quick write interrupt is masked 1'b1: spi2apb message register 1 quick write interrupt is not masked
4	RW	0x0	RFFIM Receive FIFO Full Interrupt Mask 1'b0: spi_rxf_intr interrupt is masked 1'b1: spi_rxf_intr interrupt is not masked
3	RW	0x0	RFOIM Receive FIFO Overflow Interrupt Mask 1'b0: spi_rxo_intr interrupt is masked 1'b1: spi_rxo_intr interrupt is not masked
2	RW	0x0	RFUIM Receive FIFO Underflow Interrupt Mask 1'b0: spi_rxu_intr interrupt is masked 1'b1: spi_rxu_intr interrupt is not masked
1	RW	0x0	TFOIM Transmit FIFO Overflow Interrupt Mask 1'b0: spi_txo_intr interrupt is masked 1'b1: spi_txo_intr interrupt is not masked
0	RW	0x0	TFEIM Transmit FIFO Empty Interrupt Mask 1'b0: spi_txe_intr interrupt is masked 1'b1: spi_txe_intr interrupt is not masked

SPI2APB_ISR

Address: Operational Base + offset (0x0030)

Bit	Attr	Reset Value	Description
31:6	RO	0x0	Reserved
5	RW	0x0	QWIS Quick Write Raw Interrupt Status 1'b0: spi2apb message register 1 quick write interrupt is not active after masking 1'b1: spi2apb message register 1 quick write interrupt is full after masking
4	RO	0x0	RFFIS Receive FIFO Full Interrupt Status 1'b0: spi_rxf_intr interrupt is not active after masking 1'b1: spi_rxf_intr interrupt is full after masking
3	RO	0x0	RFOIS Receive FIFO Overflow Interrupt Status 1'b0: spi_rxo_intr interrupt is not active after masking 1'b1: spi_rxo_intr interrupt is active after masking

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2	RO	0x0	RFUIS Receive FIFO Underflow Interrupt Status 1'b0: spi_rxu_intr interrupt is not active after masking 1'b1: spi_rxu_intr interrupt is active after masking
1	RO	0x0	TFOIS Transmit FIFO Overflow Interrupt Status 1'b0: spi_txo_intr interrupt is not active after masking 1'b1: spi_txo_intr interrupt is active after masking
0	RO	0x0	TFEIS Transmit FIFO Empty Interrupt Status 1'b0: spi_txe_intr interrupt is not active after masking 1'b1: spi_txe_intr interrupt is active after masking

SPI2APB_RISR

Address: Operational Base + offset (0x0034)

Bit	Attr	Reset Value	Description
31:6	RO	0x0	Reserved
5	RW	0x0	QWRIS Quick Write Raw Interrupt Status 1'b0: spi2apb message register 1 quick write interrupt is not active before masked 1'b1: spi2apb message register 1 quick write interrupt is active before masked
4	RO	0x0	RFFRIS Receive FIFO Full Raw Interrupt Status 1'b0: spi_rxf_intr interrupt is not active prior to masking 1'b1: spi_rxf_intr interrupt is full prior to masking
3	RO	0x0	RFORIS Receive FIFO Overflow Raw Interrupt Status 1'b0: spi_rxo_intr interrupt is not active prior to masking 1'b1: spi_rxo_intr interrupt is active prior to masking
2	RO	0x0	RFURIS Receive FIFO Underflow Raw Interrupt Status 1'b0: spi_rxu_intr interrupt is not active prior to masking 1'b1: spi_rxu_intr interrupt is active prior to masking
1	RO	0x0	TFORIS Transmit FIFO Overflow Raw Interrupt Status 1'b0: spi_txo_intr interrupt is not active prior to masking 1'b1: spi_txo_intr interrupt is active prior to masking
0	RO	0x1	TFERIS Transmit FIFO Empty Raw Interrupt Status 1'b0: spi_txe_intr interrupt is not active prior to masking 1'b1: spi_txe_intr interrupt is active prior to masking

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

SPI2APB_ICR

Address: Operational Base + offset (0x0038)

Bit	Attr	Reset Value	Description
31:5	RO	0x0	Reserved
4	RW	0x0	CQWI Clear Quick Write Interrupt Write 1 to Clear SPI2APB message register 1 Interrupt
3	WO	0x0	CTFOI Clear Transmit FIFO Overflow Interrupt Write 1 to Clear Transmit FIFO Overflow Interrupt
2	WO	0x0	CRFOI Clear Receive FIFO Overflow Interrupt Write 1 to Clear Receive FIFO Overflow Interrupt
1	WO	0x0	CRFUI Clear Receive FIFO Underflow Interrupt Write 1 to Clear Receive FIFO Underflow Interrupt
0	WO	0x0	CCI Clear Combined Interrupt Write 1 to Clear Combined Interrupt

SPI2APB_WORK_MODE

Address: Operational Base + offset (0x004c)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	Reserved
0	RW	0x1	SPI2APB SPI to APB convetor enable 1'b0: Work in normal SPI mode 1'b1: Work in SPI2APB convertor mode

SPI2APB_MESSAGE_REG0

Address: Operational Base + offset (0x0050)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	QUICK_REG0 SPI master write to this register, and DSP read it.

SPI2APB_MESSAGE_REG1

Address: Operational Base + offset (0x0054)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	QUICK_REG1 External SPI master write to this register, generate an interrupt ,and DSP read it.

SPI2APB_MESSAGE_REG2

Address: Operational Base + offset (0x0058)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	QUICK_REG2 DSP write this register . External SPI master read it .

21.5 Interface Description

Table 21-1 SPI2APB interface description

Module Pin	Direction	Pad Name	IOMUX Setting
spi0_clk	I/O	GPIO1_B7/SPI0_CLK/PWM5	GRF_GPIO1BH_IOMUX[14:12]=3'h1
spi0_mosi	I	GPIO1_B4/SPI0_MOSI/I2C2_SCL_M1/ UART1_RX_M1	GRF_GPIO1BH_IOMUX[2:0]=3'h1
spi0_miso	O	GPIO1_B5/SPI0_MISO/I2C2_SDA_M1/ UART1_TX_M1	GRF_GPIO1BH_IOMUX [6:4]=3'h1
spi0_csn	I/O	GPIO1_B6/SPI0_CSN	GRF_GPIO1BH_IOMUX[10:8]=3'h1

21.6 Application Notes

External SPI master should query the transaction state after every write and read operation to ensure the operation is completed correctly.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 22 Serial Peripheral Interface (SPI)

22.1 Overview

The serial peripheral interface is an APB slave device. A four wire full duplex serial protocol from Motorola. There are four possible combinations for the serial clock phase and polarity. The clock phase (SCPH) determines whether the serial transfer begins with the falling edge of slave select signals or the first edge of the serial clock. The slave select line is held high when the SPI is idle or disabled. This SPI controller can work as either master or slave mode.

SPI Controller supports the following features:

- Support Motorola SPI,TI Synchronous Serial Protocol and National Semiconductor Micro wire interface
- Support 32-bit APB bus
- Support two internal 16-bit wide and 32-location deep FIFOs, one for transmitting and the other for receiving serial data
- Support two chip select signals in master mode
- Support 4,8,16 bit serial data transfer
- Support configurable interrupt polarity
- Support asynchronous APB bus and SPI clock
- Support master and slave mode
- Support DMA handshake interface and configurable DMA water level
- Support transmit FIFO empty, underflow, receive FIFO full, overflow, interrupt and all interrupts can be masked
- Support configurable water level of transmit FIFO empty and receive FIFO full interrupt
- Support combine interrupt output
- Support up to half of SPI clock frequency transfer in master mode and one sixth of SPI clock frequency transfer in slave mode
- Support full and half duplex mode transfer
- Stop transmitting SCLK if transmit FIFO is empty or receive FIFO is full in master mode
- Support configurable delay from chip select active to SCLK active in master mode
- Support configurable period of chip select inactive between two parallel data in master mode
- Support big and little endian, MSB and LSB first transfer
- Support two 8-bit audio data store together in one 16-bit wide location
- Support sample RXD 0~3 SPI clock cycles later
- Support configurable SCLK polarity and phase
- Support fix and incremental address access to transmit and receive FIFO
- In this chip, SPI1/2 using SPI module, SPI0 using SPI2APB module.

22.2 Block Diagram

The SPI Controller comprises with:

- AMBA APB interface and DMA Controller Interface
- Transmit and receive FIFO controllers and an FSM controller
- Register block
- Shift control and interrupt

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

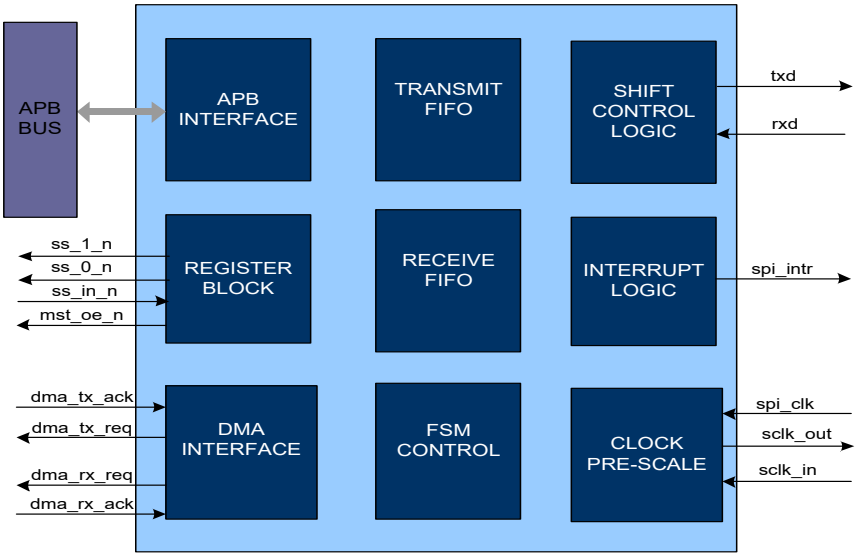


Fig. 22-1 SPI Controller Block diagram

APB INTERFACE

The host processor accesses data, control, and status information on the SPI through the APB interface. The SPI supports APB data bus widths of 32 bits and 8 or 16 bits when reading or writing internal FIFO if data frame size(SPI_CTRL0[1:0]) is set to 8 bits.

DMA INTERFACE

This block has a handshaking interface to a DMA Controller to request and control transfers. The APB bus is used to perform the data transfer to or from the DMA Controller.

FIFO LOGIC

For transmit and receive transfers, data transmitted from the SPI to the external serial device is written into the transmit FIFO. Data received from the external serial device into the SPI is pushed into the receive FIFO. Both fifos are 32x16bits.

FSM CONTROL

Control the state's transformation of the design.

REGISTER BLOCK

All registers in the SPI are addressed at 32-bit boundaries to remain consistent with the APB bus. Where the physical size of any register is less than 32-bits wide, the upper unused bits of the 32-bit boundary are reserved. Writing to these bits has no effect; reading from these bits returns 0.

SHIFT CONTROL

Shift control logic shift the data from the transmit fifo or to the receive fifo. This logic automatically right-justifies receive data in the receive FIFO buffer.

INTERRUPT CONTROL

The SPI supports combined and individual interrupt requests, each of which can be masked. The combined interrupt request is the ORed result of all other SPI interrupts after masking.

22.3 Function Description

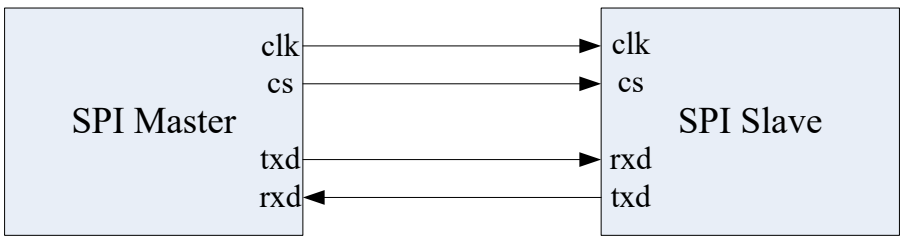


Fig. 22-2 SPI Master and Slave Interconnection

The SPI controller support dynamic switching between master and slave in a system. The diagram show how the SPI controller connects with other SPI devices.

Operation Modes

The SPI can be configured in the following two fundamental modes of operation: Master

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Mode when SPI_CTRLR0 [20] is 1'b0, Slave Mode when SPI_CTRLR0 [20] is 1'b1.

Transfer Modes

The SPI operates in the following three modes when transferring data on the serial bus.

1). Transmit and Receive

When SPI_CTRLR0 [19:18]== 2'b00, both transmit and receive logic are valid.

2).Transmit Only

When SPI_CTRLR0 [19:18] == 2'b01, the receive data are invalid and should not be stored in the receive FIFO.

3).Receive Only

When SPI_CTRLR0 [19:18]== 2'b10, the transmit data are invalid.

Clock Ratios

A summary of the frequency ratio restrictions between the bit-rate clock (sclk_out/sclk_in) and the SPI peripheral clock (spi_clk) are described as,

When SPI Controller works as master, the F_{spi_clk}>= 2 × (maximum F_{sclk_out})

When SPI Controller works as slave, the F_{spi_clk}>= 6 × (maximum F_{sclk_in})

With the SPI, the clock polarity (SCPOL) configuration parameter determines whether the inactive state of the serial clock is high or low. To transmit data, both SPI peripherals must have identical serial clock phase (SCPH) and clock polarity (SCPOL) values. The data frame can be 4/8/16 bits in length.

When the configuration parameter SCPH = 0, data transmission begins on the falling edge of the slave select signal. The first data bit is captured by the master and slave peripherals on the first edge of the serial clock; therefore, valid data must be present on the txd and rxd lines prior to the first serial clock edge. The following two figures show a timing diagram for a single SPI data transfer with SCPH = 0. The serial clock is shown for configuration parameters SCPOL = 0 and SCPOL = 1.

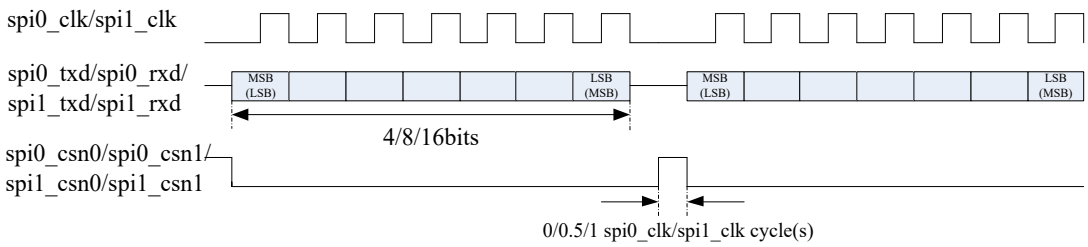


Fig. 22-3 SPI Format (SCPH=0 SCPOL=0)

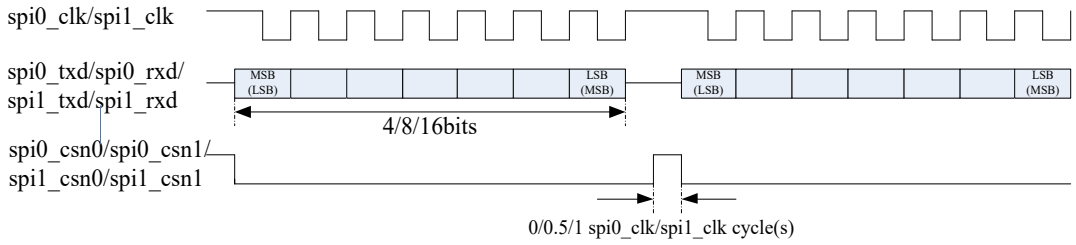


Fig. 22-4 SPI Format (SCPH=0 SCPOL=1)

When the configuration parameter SCPH = 1, both master and slave peripherals begin transmitting data on the first serial clock edge after the slave select line is activated. The first data bit is captured on the second (trailing) serial clock edge. Data are propagated by the master and slave peripherals on the leading edge of the serial clock. During continuous data frame transfers, the slave select line may be held active-low until the last bit of the last frame has been captured. The following two figures show the timing diagram for the SPI format when the configuration parameter SCPH = 1.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

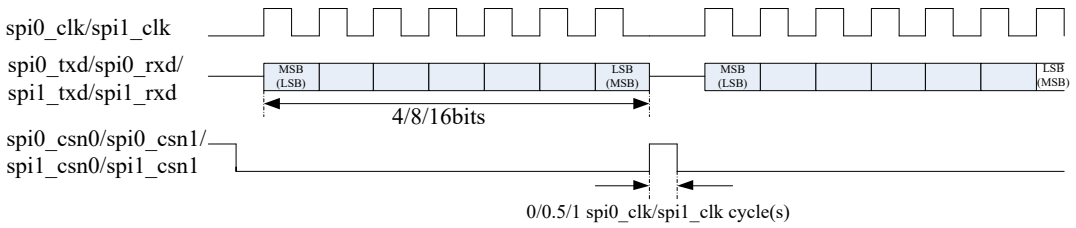


Fig. 22-5 SPI Format (SCPH=1 SCPOL=0)

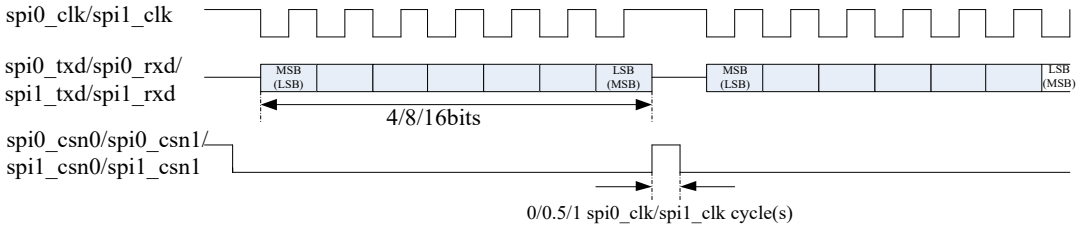


Fig. 22-6 SPI Format (SCPH=1 SCPOL=1)

22.4 Register Description

22.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
SPI_CTRLR0	0x0000	W	0x02000002	Control Register 0
SPI_CTRLR1	0x0004	W	0x00000000	Control Register 1
SPI_ENR	0x0008	W	0x00000000	SPI Enable Register
SPI_SER	0x000c	W	0x00000000	Slave Enable Register
SPI_BAUDR	0x0010	W	0x00000000	Baud Rate Select
SPI_TXFTLR	0x0014	W	0x00000000	Transmit FIFO Threshold Level
SPI_RXFTLR	0x0018	W	0x00000000	Receive FIFO Threshold Level
SPI_TXFLR	0x001c	W	0x00000000	Transmit FIFO Level
SPI_RXFLR	0x0020	W	0x00000000	Receive FIFO Level
SPI_SR	0x0024	W	0x0000000c	SPI Status
SPI_IPR	0x0028	W	0x00000000	Interrupt Polarity
SPI_IMR	0x002c	W	0x00000000	Interrupt Mask
SPI_ISR	0x0030	W	0x00000000	Interrupt Status
SPI_RISR	0x0034	W	0x00000001	Raw Interrupt Status
SPI_ICR	0x0038	W	0x00000000	Interrupt Clear
SPI_DMACR	0x003c	W	0x00000000	DMA Control
SPI_DMATDLR	0x0040	W	0x00000000	DMA Transmit Data Level
SPI_DMARDLR	0x0044	W	0x00000000	DMA Receive Data Level
SPI_SPI2APB	0x004c	W	0x00000001	SPI to APB convetor enable.
SPI_QUICK_REG0	0x0050	W	0x00000000	Register0000 Description
SPI_QUICK_REG1	0x0054	W	0x00000000	Register0000 Description
SPI_QUICK_REG2	0x0058	W	0x00000000	DSP write to this register . AP read it .
SPI_TXDR	0x0400	W	0x00000000	Transmit FIFO Data
SPI_RXDR	0x0800	W	0x00000000	Receive FIFO Data

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

22.4.2 Detail Register Description

SPI_CTRLR0

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:26	RO	0x0	Reserved
25	RW	0x1	sclk_in_mask 1'b1: sclk_in is masked bu ss_in_n 1'b0: sclk_in is not masked
24:23	RW	0x0	txd_delay when txd_drive is 1'b0 , this field control the delay cycle numbers of txd . 2'b00: SCLK posedge pusle. 2'b00: SCLK posedge pusle delay one cycle. 2'b00: SCLK posedge pusle delay two cycles. 2'b00: SCLK posedge pusle delay three cycles.
22	RW	0x0	txd_drive 1'b0: TXD is drived at posedge of SCLK 1'b1: TXD is drived at negedge of SCLK
21	RW	0x0	MTM Microwire Transfer Mode. Valid when frame format is set to National Semiconductors Microwire. 1'b0: Non-sequential transfer 1'b1: Sequential transfer
20	RW	0x0	OPM Operation Mode. 1'b0: Master Mode 1'b1: Slave Mode
19:18	RW	0x0	XFM Transfer Mode. 2'b00: Transmit & Receive 2'b01: Transmit Only 2'b10: Receive Only 2'b11: Reserved
17:16	RW	0x0	FRF Frame Format. 2'b00: Motorola SPI 2'b01: Texas Instruments SSP 2'b10: National Semiconductors Microwire 2'b11: Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
15:14	RW	0x0	RSD Rxd Sample Delay. When SPI is configured as a master, if the rxd data cannot be sampled by the sclk_out edge at the right time, this register should be configured to define the number of the spi_clk cycles after the active sclk_out edge to sample rxd data later when SPI works at high frequency. 2'b00: Do not delay 2'b01: 1 cycle delay 2'b10: 2 cycles delay 2'b11: 3 cycles delay
13	RW	0x0	BHT Byte and Halfword Transform. Valid when data frame size is 8bit. 1'b0: APB 16bit write/read, spi 8bit write/read 1'b1: APB 8bit write/read, spi 8bit write/read
12	RW	0x0	FBM First Bit Mode. 1'b0: First bit is MSB 1'b1: First bit is LSB
11	RW	0x0	EM Endian Mode. Serial endian mode can be configured by this bit. Apb endian mode is always little endian. 1'b0: Little endian 1'b1: Big endian
10	RW	0x0	SSD ss_n to sclk_out delay. Valid when the frame format is set to Motorola SPI and SPI used as a master. 0: The period between ss_n active and sclk_out active is half sclk_out cycles. 1: The period between ss_n active and sclk_out active is one sclk_out cycle.
9:8	RW	0x0	CSM Chip Select Mode. Valid when the frame format is set to Motorola SPI and SPI used as a master. 2'b00: ss_n keep low after every frame data is transferred. 2'b01: ss_n be high for half sclk_out cycles after every frame data is transferred. 2'b10: ss_n be high for one sclk_out cycle after every frame data is transferred. 2'b11: Reserved
7	RW	0x0	SCPOL Serial Clock Polarity. Valid when the frame format is set to Motorola SPI. 1'b0: Inactive state of serial clock is low 1'b1: Inactive state of serial clock is high

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
6	RW	0x0	SCPH Serial Clock Phase. Valid when the frame format is set to Motorola SPI. 1'b0: Serial clock toggles in middle of first data bit 1'b1: Serial clock toggles at start of first data bit
5:2	RW	0x0	CFS Control Frame Size. Selects the length of the control word for the Microwire frame format. 4'b0000~0010:Reserved 4'b0011:4-bit serial data transfer 4'b0100:5-bit serial data transfer 4'b0101:6-bit serial data transfer 4'b0110:7-bit serial data transfer 4'b0111:8-bit serial data transfer 4'b1000:9-bit serial data transfer 4'b1001:10-bit serial data transfer 4'b1010:11-bit serial data transfer 4'b1011:12-bit serial data transfer 4'b1100:13-bit serial data transfer 4'b1101:14-bit serial data transfer 4'b1110:15-bit serial data transfer 4'b1111:16-bit serial data transfer
1:0	RW	0x2	DFS Data Frame Size. Selects the data frame length. 2'b00:4bit data 2'b01:8bit data 2'b10:16bit data 2'b11:Reserved

SPI_CTRLR1

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	Reserved
15:0	RW	0x0000	NDM Number of Data Frames. When Transfer Mode is receive only, this register field sets the number of data frames to be continuously received by the SPI. The SPI continues to receive serial data until the number of data frames received is equal to this register value plus 1, which enables you to receive up to 64 KB of data in a continuous transfer.

SPI_ENR

Address: Operational Base + offset (0x0008)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:1	RO	0x0	Reserved
0	RW	0x0	ENR Enables and disables all SPI operations. Transmit and receive FIFO buffers are cleared when the device is disabled.

SPI_SER

Address: Operational Base + offset (0x000c)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	Reserved
1:0	RW	0x0	SER Slave Select Enable. This register is valid only when SPI is configured as a master device.

SPI_BAUDR

Address: Operational Base + offset (0x0010)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	Reserved
15:0	RW	0x0000	BAUDR SPI Clock Divider. This register is valid only when the SPI is configured as a master device. The LSB for this field is always set to 0 and is unaffected by a write operation, which ensures an even value is held in this register. If the value is 0, the serial output clock (sclk_out) is disabled. The frequency of the sclk_out is derived from the following equation: Fsclk_out = Fspi_clk/ SCKDV Where SCKDV is any even value between 2 and 65534. For example: for Fspi_clk = 3.6864MHz and SCKDV =2 Fsclk_out = 3.6864/2= 1.8432MHz

SPI_TXFTLR

Address: Operational Base + offset (0x0014)

Bit	Attr	Reset Value	Description
31:5	RO	0x0	Reserved
4:0	RW	0x00	TXFTLR When the number of transmit FIFO entries is less than or equal to this value, the transmit FIFO empty interrupt is triggered.

SPI_RXFTLR

Address: Operational Base + offset (0x0018)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:5	RO	0x0	Reserved
4:0	RW	0x00	RXFTLR When the number of receive FIFO entries is greater than or equal to this value + 1, the receive FIFO full interrupt is triggered.

SPI_TXFLR

Address: Operational Base + offset (0x001c)

Bit	Attr	Reset Value	Description
31:6	RO	0x0	Reserved
5:0	RO	0x00	TXFLR Contains the number of valid data entries in the transmit FIFO.

SPI_RXFLR

Address: Operational Base + offset (0x0020)

Bit	Attr	Reset Value	Description
31:6	RO	0x0	Reserved
5:0	RO	0x00	RXFLR Contains the number of valid data entries in the receive FIFO.

SPI_SR

Address: Operational Base + offset (0x0024)

Bit	Attr	Reset Value	Description
31:5	RO	0x0	Reserved
4	RO	0x0	RFF Receive FIFO Full. 1'b0: Receive FIFO is not full 1'b1: Receive FIFO is full
3	RO	0x1	RFE Receive FIFO Empty. 1'b0: Receive FIFO is not empty 1'b1: Receive FIFO is empty
2	RO	0x1	TFE Transmit FIFO Empty. 1'b0: Transmit FIFO is not empty 1'b1: Transmit FIFO is empty
1	RO	0x0	TFF Transmit FIFO Full. 1'b0: Transmit FIFO is not full 1'b1: Transmit FIFO is full
0	RO	0x0	BSF SPI Busy Flag. When set, indicates that a serial transfer is in progress; when cleared indicates that the SPI is idle or disabled. 1'b0: SPI is idle or disabled 1'b1: SPI is actively transferring data

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

SPI_IPR

Address: Operational Base + offset (0x0028)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	Reserved
0	RW	0x0	IPR Interrupt Polarity Register 1'b0: Active Interrupt Polarity Level is HIGH 1'b1: Active Interrupt Polarity Level is LOW

SPI_IMR

Address: Operational Base + offset (0x002c)

Bit	Attr	Reset Value	Description
31:6	RO	0x0	Reserved
5	RW	0x0	QWIM Quick Write Interrupt Mask. 1'b0: SPI2APB reg1 quick write interrupt is masked 1'b1: SPI2APB reg1 quick write interrupt is not masked
4	RW	0x0	RFFIM Receive FIFO Full Interrupt Mask. 1'b0: spi_rxf_intr interrupt is masked 1'b1: spi_rxf_intr interrupt is not masked
3	RW	0x0	RFOIM Receive FIFO Overflow Interrupt Mask. 1'b0: spi_rxo_intr interrupt is masked 1'b1: spi_rxo_intr interrupt is not masked
2	RW	0x0	RFUIM Receive FIFO Underflow Interrupt Mask. 1'b0: spi_rxu_intr interrupt is masked 1'b1: spi_rxu_intr interrupt is not masked
1	RW	0x0	TFOIM Transmit FIFO Overflow Interrupt Mask. 1'b0: spi_txo_intr interrupt is masked 1'b1: spi_txo_intr interrupt is not masked
0	RW	0x0	TFEIM Transmit FIFO Empty Interrupt Mask. 1'b0: spi_txe_intr interrupt is masked 1'b1: spi_txe_intr interrupt is not masked

SPI_ISR

Address: Operational Base + offset (0x0030)

Bit	Attr	Reset Value	Description
31:6	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5	RW	0x0	QWIS Quick Write Raw Interrupt Status. 1'b0: SPI2APB reg1 quick write interrupt is not active after masking 1'b1: SPI2APB reg1 quick write interrupt is full after masking
4	RO	0x0	RFFIS Receive FIFO Full Interrupt Status. 1'b0: spi_rxf_intr interrupt is not active after masking 1'b1: spi_rxf_intr interrupt is full after masking
3	RO	0x0	RFOIS Receive FIFO Overflow Interrupt Status 1'b0: spi_rxo_intr interrupt is not active after masking 1'b1: spi_rxo_intr interrupt is active after masking
2	RO	0x0	RFUIS Receive FIFO Underflow Interrupt Status. 1'b0: spi_rxu_intr interrupt is not active after masking 1'b1: spi_rxu_intr interrupt is active after masking
1	RO	0x0	TFOIS Transmit FIFO Overflow Interrupt Status. 1'b0: spi_txo_intr interrupt is not active after masking 1'b1: spi_txo_intr interrupt is active after masking
0	RO	0x0	TFEIS Transmit FIFO Empty Interrupt Status. 1'b0: spi_txe_intr interrupt is not active after masking 1'b1: spi_txe_intr interrupt is active after masking

SPI_RISR

Address: Operational Base + offset (0x0034)

Bit	Attr	Reset Value	Description
31:6	RO	0x0	Reserved
5	RW	0x0	QWRIS Quick Write Raw Interrupt Status. 1'b0: SPI2APB reg1 quick write interrupt is not active prior to masking 1'b1: SPI2APB reg1 quick write interrupt is full prior to masking
4	RO	0x0	RFFRIS Receive FIFO Full Raw Interrupt Status. 1'b0: spi_rxf_intr interrupt is not active prior to masking 1'b1: spi_rxf_intr interrupt is full prior to masking
3	RO	0x0	RFORIS Receive FIFO Overflow Raw Interrupt Status . 1'b0: spi_rxo_intr interrupt is not active prior to masking 1'b1: spi_rxo_intr interrupt is active prior to masking

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2	RO	0x0	RFURIS Receive FIFO Underflow Raw Interrupt Status. 1'b0: spi_rxu_intr interrupt is not active prior to masking 1'b1: spi_rxu_intr interrupt is active prior to masking
1	RO	0x0	TFORIS Transmit FIFO Overflow Raw Interrupt Status. 1'b0: spi_txo_intr interrupt is not active prior to masking 1'b1: spi_txo_intr interrupt is active prior to masking
0	RO	0x1	TFERIS Transmit FIFO Empty Raw Interrupt Status. 1'b0: spi_txe_intr interrupt is not active prior to masking 1'b1: spi_txe_intr interrupt is active prior to masking

SPI_ICR

Address: Operational Base + offset (0x0038)

Bit	Attr	Reset Value	Description
31:5	RO	0x0	Reserved
4	RW	0x0	CQWI Clear Quick Write Interrupt. Write 1 to Clear SPI2APB quick write reg1 Interrupt
3	WO	0x0	CTFOI Clear Transmit FIFO Overflow Interrupt. Write 1 to Clear Transmit FIFO Overflow Interrupt
2	WO	0x0	CRFOI Clear Receive FIFO Overflow Interrupt. Write 1 to Clear Receive FIFO Overflow Interrupt
1	WO	0x0	CRFUI Clear Receive FIFO Underflow Interrupt. Write 1 to Clear Receive FIFO Underflow Interrupt
0	WO	0x0	CCI Clear Combined Interrupt. Write 1 to Clear Combined Interrupt

SPI_DMACR

Address: Operational Base + offset (0x003c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:2	RO	0x0	Reserved
1	RW	0x0	TDE Transmit DMA Enable. 1'b0: Transmit DMA disabled 1'b1: Transmit DMA enabled
0	RW	0x0	RDE Receive DMA Enable. 1'b0: Receive DMA disabled 1'b1: Receive DMA enabled

SPI_DMATDLR

Address: Operational Base + offset (0x0040)

Bit	Attr	Reset Value	Description
31:5	RO	0x0	Reserved
4:0	RW	0x00	TDL Transmit Data Level. This bit field controls the level at which a DMA request is made by the transmit logic. It is equal to the watermark level; that is, the dma_tx_req signal is generated when the number of valid data entries in the transmit FIFO is equal to or below this field value, and Transmit DMA Enable (DMACR[1]) = 1.

SPI_DMARDLR

Address: Operational Base + offset (0x0044)

Bit	Attr	Reset Value	Description
31:5	RO	0x0	Reserved
4:0	RW	0x00	RDL Receive Data Level. This bit field controls the level at which a DMA request is made by the receive logic. The watermark level = DMARDL+1; that is, dma_rx_req is generated when the number of valid data entries in the receive FIFO is equal to or above this field value + 1, and Receive DMA Enable(DMACR[0])=1.

SPI_SPI2APB

Address: Operational Base + offset (0x004c)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	Reserved
0	RW	0x1	SPI2APB SPI to APB convetor enable.

SPI_QUICK_REG0

Address: Operational Base + offset (0x0050)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	QUICK_REG0 SPI master write to this register, and DSP read it.

SPI_QUICK_REG1

Address: Operational Base + offset (0x0054)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	QUICK_REG1 SPI master write to this register, generate a interrupt .and DSP read it.

SPI_QUICK_REG2

Address: Operational Base + offset (0x0058)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	QUICK_REG2 Field0000 Description

SPI_TXDR

Address: Operational Base + offset (0x0400)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	Reserved
15:0	WO	0x0000	TXDR Transimt FIFO Data Register. When it is written to, data are moved into the transmit FIFO.

SPI_RXDR

Address: Operational Base + offset (0x0800)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	Reserved
15:0	RW	0x0000	RXDR Receive FIFO Data Register. When the register is read, data in the receive FIFO is accessed.

22.5 Interface Description

Table 22-1 SPI1/SPI2 interface description

Module Pin	Direction	Pad Name	IOMUX Setting
spi1m0_clk	I/O	GPIO4_B4/UART4_RX/SPI1_CLK_M0	GRF_GPIO4BH_IOMUX[2:0]=3'h2
spi1m0_mosi	I	GPIO4_B5/UART4_TX/SPI1_MOSI_M0	GRF_GPIO4BH_IOMUX[6:4]=3'h2
spi1m0_miso	O	GPIO4_B7/UART4_RTS/SPI1_MISO_M0	GRF_GPIO4BH_IOMUX[14:12]=3'h2
spi1m0_csn0	O	GPIO4_B6/UART4_CTS/SPI1_CSN0_M0	GRF_GPIO4BH_IOMUX[10:8]=3'h2
spi1m0_csn1	O	GPIO4_C0/SPI1_CSN1_M0	GRF_GPIO4CL_IOMUX[2:0]=3'h2
spi1m1_clk	I/O	GPIO3_C7/LCDC_D13/UART7_RX/SPI1_CLK_M1	GRF_GPIO3CH_IOMUX[14:12]=3'h3
spi1m1_mosi	O	GPIO3_D0/LCDC_D14/PWM8/SPI1_MOSI_M1	GRF_GPIO3DL_IOMUX[2:0]=3'h3

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Module Pin	Direction	Pad Name	IOMUX Setting
spi1m1_miso	O	GPIO3_D2/LCDC_D16/PWM10/SPI1_MISO_M1	GRF_GPIO3DL_IOMUX[10:8]=3'h3
spi1m1_csn0	O	GPIO3_D1/LCDC_D15/PWM9/SPI1_CSN0_M1	GRF_GPIO3DL_IOMUX[6:4]=3'h3
spi1m1_csn1	O	GPIO3_D3/LCDC_D17/PWM11/SPI1_CSN1_M1	GRF_GPIO3DL_IOMUX[14:12]=3'h3
spi2m0_clk	I/O	GPIO1_A7/EMMC_D7/SPI2_CLK_M0	GRF_GPIO1AH_IOMUX[14:12]=3'h2
spi2m0_mosi	O	GPIO1_B0/EMMC_PWREN/SPI2_MOSI_M0	GRF_GPIO1AH_IOMUX[2:0]=3'h2
spi2m0_miso	O	GPIO1_A6/EMMC_D6/SPI2_MISO_M0	GRF_GPIO1AH_IOMUX[10:8]=3'h2
spi2m0_csn	O	GPIO1_B1/EMMC_CLKOUT/SPI2_CSN_M0	GRF_GPIO1BL_IOMUX[6:4]=3'h2
spi2m1_clk	I/O	GPIO2_A5/CIF_D3/RGMII_RXD1/SPI2_CLK_M1	GRF_GPIO2AH_IOMUX[6:4]=3'h3
spi2m1_mosi	O	GPIO2_A4/CIF_D2/RGMII_RXD0/SPI2_MISO_M1	GRF_GPIO2AH_IOMUX[10:8]=3'h3
spi2m1_miso	O	GPIO2_A4/CIF_D2/RGMII_RXD0/SPI2_MISO_M1	GRF_GPIO2AH_IOMUX[2:0]=3'h3
spi2m1_csn	O	GPIO2_A7/CIF_D5/RGMII_RXDV/SPI2_CSN_M1	GRF_GPIO2AH_IOMUX[14:12]=3'h3

Notes: I=input, O=output, I/O=input/output, bidirectional. spi_csn1 can only be used in master mode

22.6 Application Notes

Clock Ratios

A summary of the frequency ratio restrictions between the bit-rate clock (sclk_out/sclk_in) and the SPI peripheral clock (spi_clk) are described as,

When SPI Controller works as master, the Fspi_clk>= 2 × (maximum Fsclk_out)

When SPI Controller works as slave, the Fspi_clk>= 6 × (maximum Fsclk_in)

Master Transfer Flow

When configured as a serial-master device, the SPI initiates and controls all serial transfers. The serial bit-rate clock, generated and controlled by the SPI, is driven out on the sclk_out line. When the SPI is disabled (SPI_ENR = 0), no serial transfers can occur and sclk_out is held in "inactive" state, as defined by the serial protocol under which it operates.

Slave Transfer Flow

When the SPI is configured as a slave device, all serial transfers are initiated and controlled by the serial bus master.

When the SPI serial slave is selected during configuration, it enables its txd data onto the serial bus. All data transfers to and from the serial slave are regulated on the serial clock line (sclk_in), driven from the serial-master device. Data are propagated from the serial slave on one edge of the serial clock line and sampled on the opposite edge.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

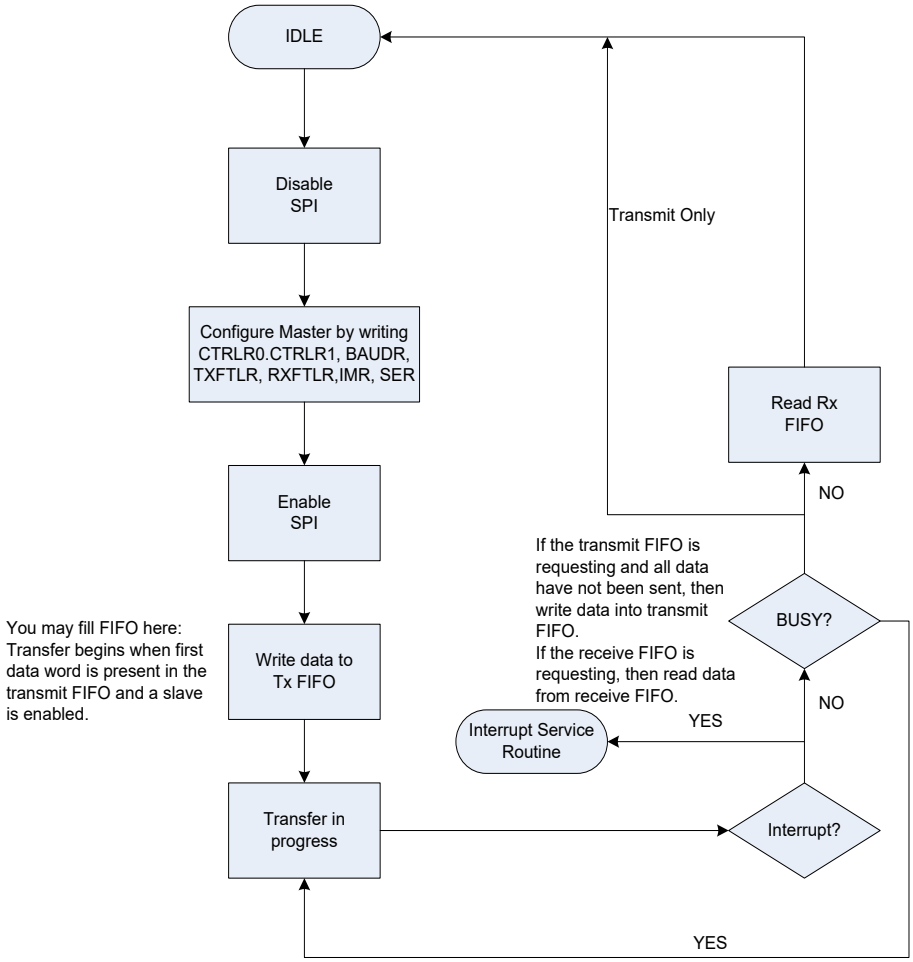


Fig. 22-7 SPI Master transfer flow diagram

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

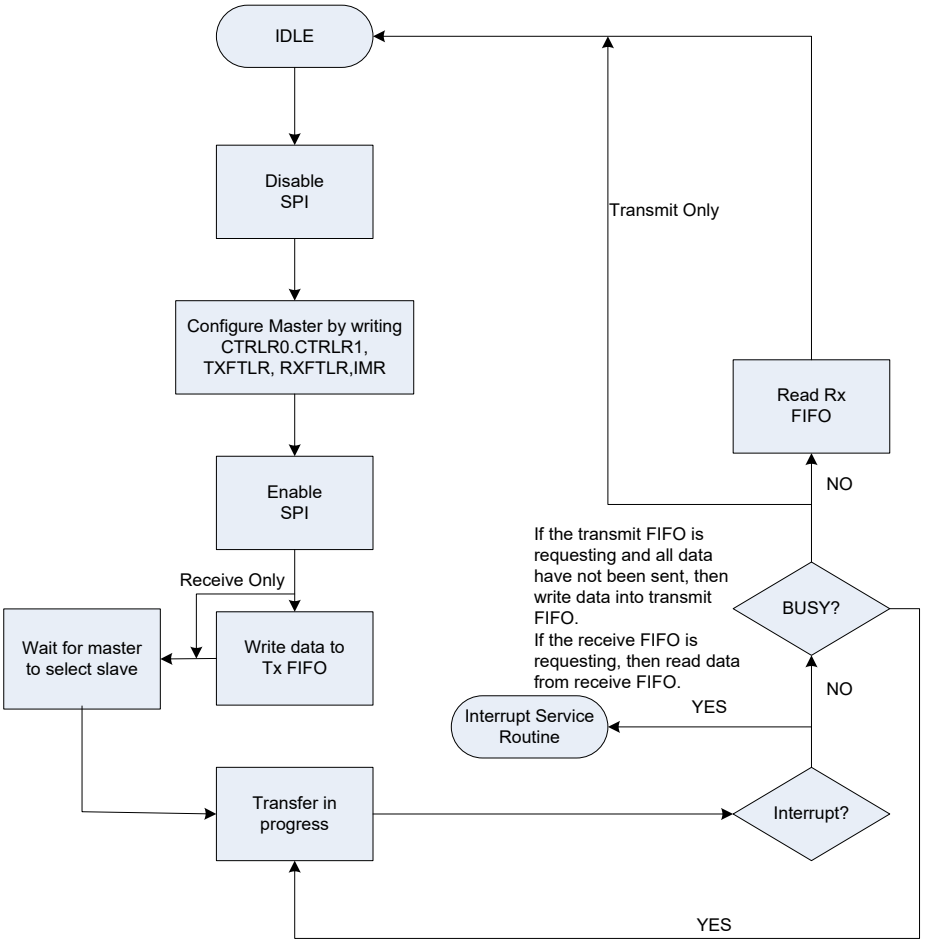


Fig. 22-8 SPI Slave transfer flow diagram

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 23 Universal Asynchronous Receiver/Transmitter (UART)

23.1 Overview

The Universal Asynchronous Receiver/Transmitter (UART) is used for serial communication with a peripheral, modem (data carrier equipment, DCE) or data set. Data is written from a master (CPU) over the APB bus to the UART and it is converted to serial form and transmitted to the destination device. Serial data is also received by the UART and stored for the master (CPU) to read back.
UART Controller supports the following features:

- Support 6 independent UART controller: UART0, UART1, UART2, UART3, UART4, UART5, UART6, UART7
- UART0/UART1/UART2/UART3/UART4/UART5/UART6/UART7 all contain two 64Bytes FIFOs for data receive and transmit
- UART0/UART1/UART3/UART4/UART5/UART6/UART7 all support auto flow-control
- Support bit rates 115.2Kbps,460.8Kbps,921.6Kbps,1.5Mbps,3Mbps, 4Mbps
- Support programmable baud rates, even with non-integer clock divider
- Standard asynchronous communication bits (start, stop and parity)
- Support interrupt-based or DMA-based mode
- Support 5-8 bits width transfer

23.2 Block Diagram

This section provides a description about the functions and behavior under various conditions. The UART Controller comprises with:

- AMBA APB interface
- FIFO controllers
- Register block
- Modem synchronization block and baud clock generation block
- Serial receiver and serial transmitter

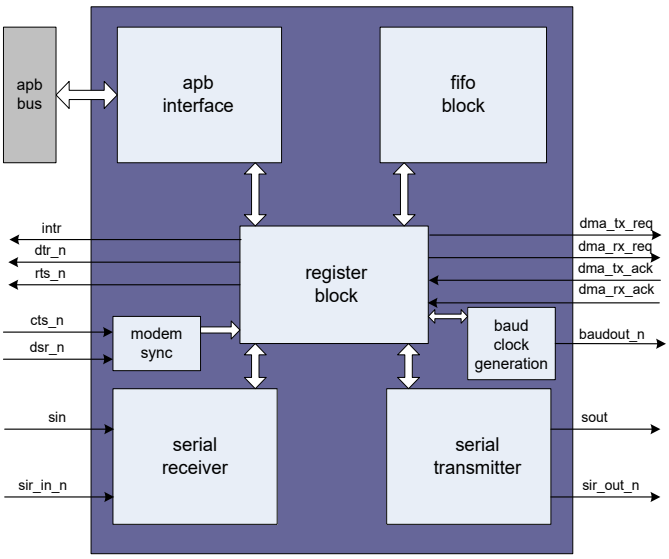


Fig. 23-1 UART Architecture

APB INTERFACE
The host processor accesses data, control, and status information on the UART through the APB interface. The UART supports APB data bus widths of 8, 16, and 32 bits.
Register block
Be responsible for the main UART functionality including control, status and interrupt generation.
Modem Synchronization block
Synchronizes the modem input signal.
FIFO block

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Be responsible for FIFO control and storage (when using internal RAM) or signaling to control external RAM (when used).

Baud Clock Generator

Generates the transmitter and receiver baud clockalong with the output reference clock signal (baudout_n).

Serial Transmitter

Converts the parallel data, written to the UART, into serial form and adds all additional bits, as specified by the control register, for transmission. This makeup of serial data, referred to as a character can exit the block in two forms, either serial UART format or IrDA 1.0 SIR format.

Serial Receiver

Converts the serial data character (as specified by the control register) received in either the UART or IrDA 1.0 SIR format to parallel form. Parity error detection, framing error detection and line break detection is carried out in this block.

23.3 Function Description

UART (RS232) Serial Protocol

Because the serial communication is asynchronous, additional bits (start and stop) are added to the serial data to indicate the beginning and end. An additional parity bit may be added to the serial character. This bit appears after the last data bit and before the stop bit(s) in the character structure to perform simple error checking on the received data, as shown in Figure.

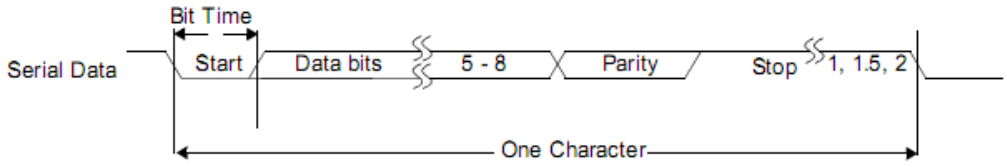


Fig. 23-2 UART Serial protocol

IrDA 1.0 SIR Protocol

The Infrared Data Association (IrDA) 1.0 Serial Infrared (SIR) mode supports bi-directional datacommunications with remote devices using infrared radiation as the transmission medium. IrDA 1.0 SIR mode specifies a maximum baud rate of 115.2 Kbaud.

Transmitting a single infrared pulse signals a logic zero, while a logic one is represented by not sending a pulse. The width of each pulse is 3/16ths of a normal serial bit time. Data transfers can only occur in half-duplex fashion when IrDA SIR mode is enabled.

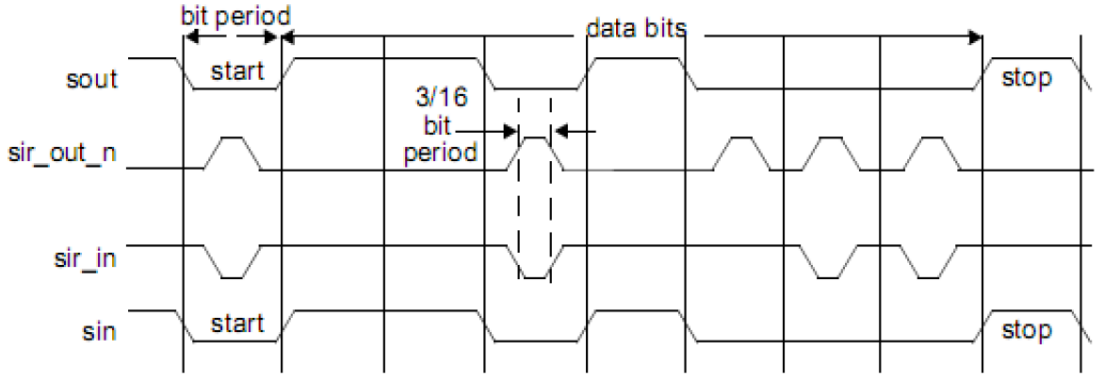


Fig. 23-3 IrDA 1.0

Baud Clock

The baud rate is controlled by the serial clock (sclk or pclk in a single clock implementation) and the Divisor Latch Register (DLH and DLL). As the exact number of baud clocks that each bit was transmitted for is known, calculating the mid-point for sampling is not difficult, that is every 16 baud clocks after the mid-point sample of the start bit.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

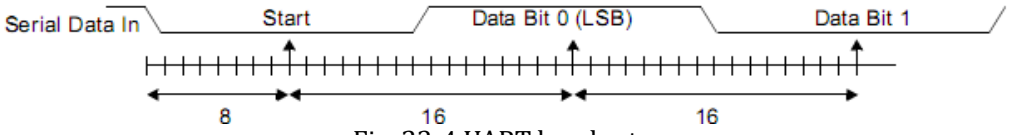


Fig. 23-4 UART baud rate

FIFO Support

1. NONE FIFO MODE

If FIFO support is not selected, then no FIFOs are implemented and only a single receive data byte and transmit data byte can be stored at a time in the RBR and THR.

2. FIFO MODE

The FIFO depth of UART0/UART1/UART2/UART3/UART4/UART5/UART6/UART7 is 64bytes. The FIFO mode of all the UART is enabled by register FCR[0].

Interrupts

The following interrupt types can be enabled with the IER register.

- Receiver Error
- Receiver Data Available
- Character Timeout (in FIFO mode only)
- Transmitter Holding Register Empty at/below threshold (in Programmable THRE Interrupt mode)
- Modem Status

DMA Support

The UART supports DMA signaling with the use of two output signals (dma_tx_req_n and dma_rx_req_n) to indicate when data is ready to be read or when the transmit FIFO is empty.

The dma_tx_req_n signal is asserted under the following conditions:

- When the Transmitter Holding Register is empty in non-FIFO mode.
- When the transmitter FIFO is empty in FIFO mode with Programmable THRE interrupt mode disabled.
- When the transmitter FIFO is at, or below the programmed threshold with Programmable THRE interrupt mode enabled.

The dma_rx_req_n signal is asserted under the following conditions:

- When there is a single character available in the Receive Buffer Register in non-FIFO mode.
- When the Receiver FIFO is at or above the programmed trigger level in FIFO mode.

Auto Flow Control

The UART can be configured to have a 16750-compatible Auto RTS and Auto CTS serial data flow control mode available. If FIFOs are not implemented, then this mode cannot be selected. When Auto Flow Control mode has been selected, it can be enabled with the Modem Control Register (MCR[5]). Following figure shows a block diagram of the Auto Flow Control functionality.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

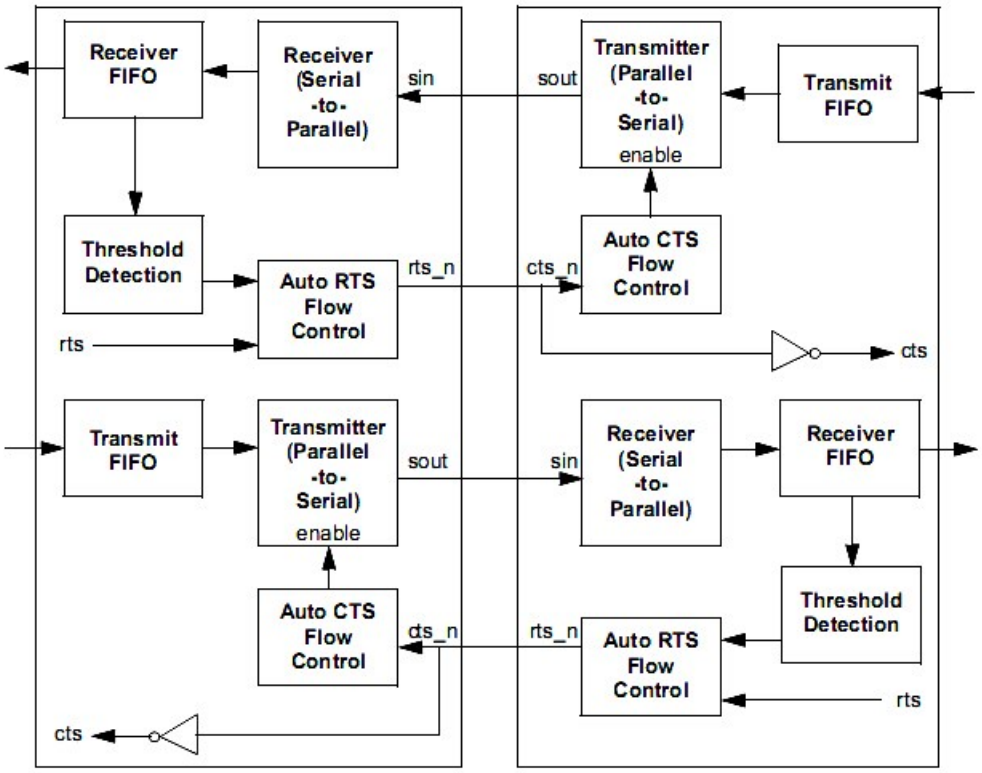


Fig. 23-5 UART Auto flow control block diagram

Auto RTS – Becomes active when the following occurs:

- Auto Flow Control is selected during configuration
- FIFOs are implemented
- RTS (MCR[1] bit and MCR[5]bit are both set)
- FIFOs are enabled (FCR[0]) bit is set)
- SIR mode is disabled (MCR[6] bit is not set)

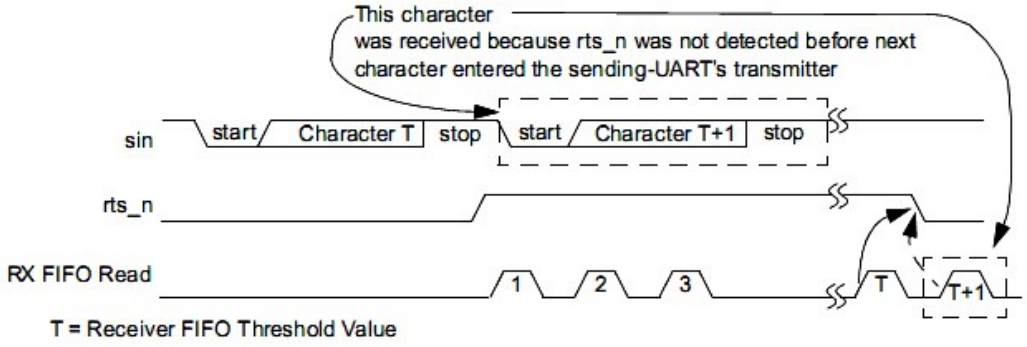


Fig. 23-6 UART AUTO RTS TIMING

Auto CTS – becomes active when the following occurs:

- Auto Flow Control is selected during configuration
- FIFOs are implemented
- AFCE (MCR[5] bit is set)
- FIFOs are enabled through FIFO Control Register FCR[0] bit
- SIR mode is disabled (MCR[6] bit is not set)

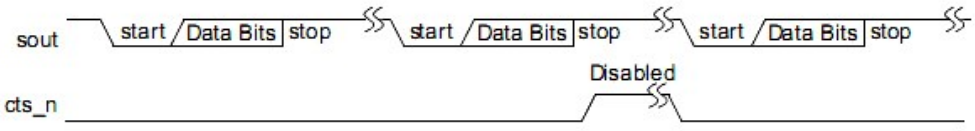


Fig. 23-7 UART AUTO CTS TIMING

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

23.4 Register Description

23.4.1 Internal Address Mapping

This section describes the control/status registers of the design. There are 8 UARTs in RK1808, and each one has its own base address.

23.4.2 Registers Summary

Name	Offset	Size	Reset Value	Description
UART_RBR	0x0000	W	0x00000000	Receive Buffer Register
UART_THR	0x0000	W	0x00000000	Transmit Holding Register
UART_DLL	0x0000	W	0x00000000	Divisor Latch (Low)
UART_DLH	0x0004	W	0x00000000	Divisor Latch (High)
UART_IER	0x0004	W	0x00000000	Interrupt Enable Register
UART_IIR	0x0008	W	0x00000001	Interrupt Identification Register
UART_FCR	0x0008	W	0x00000000	FIFO Control Register
UART_LCR	0x000c	W	0x00000000	Line Control Register
UART_MCR	0x0010	W	0x00000000	Modem Control Register
UART_LSR	0x0014	W	0x00000060	Line Status Register
UART_MSR	0x0018	W	0x00000000	Modem Status Register
UART_SCR	0x001c	W	0x00000000	Scratchpad Register
UART_SRBR	0x0030	W	0x00000000	Shadow Receive Buffer Register
UART_STHR	0x006c	W	0x00000000	Shadow Transmit Holding Register
UART_FAR	0x0070	W	0x00000000	FIFO Access Register
UART_TFR	0x0074	W	0x00000000	Transmit FIFO Read
UART_RFW	0x0078	W	0x00000000	Receive FIFO Write
UART_USR	0x007c	W	0x00000006	UART Status Register
UART_TFL	0x0080	W	0x00000000	Transmit FIFO Level
UART_RFL	0x0084	W	0x00000000	Receive FIFO Level
UART_SRR	0x0088	W	0x00000000	Software Reset Register
UART_SRTS	0x008c	W	0x00000000	Shadow Request to Send
UART_SBCR	0x0090	W	0x00000000	Shadow Break Control Register
UART_SDMAM	0x0094	W	0x00000000	Shadow DMA Mode
UART_SFE	0x0098	W	0x00000000	Shadow FIFO Enable
UART_SRT	0x009c	W	0x00000000	Shadow RCVR Trigger
UART_STET	0x00a0	W	0x00000000	Shadow TX Empty Trigger
UART_HTX	0x00a4	W	0x00000000	Halt TX
UART_DMASA	0x00a8	W	0x00000000	DMA Software Acknowledge
UART_CPR	0x00f4	W	0x00000000	Component Parameter Register
UART_UCV	0x00f8	W	0x3330382a	UART Component Version
UART_CTR	0x00fc	W	0x44570110	Component Type Register

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

23.4.3 Detail Register Description

UART_RBR

Address: Operational Base + offset (0x0000)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	data_input 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 non-FIFO mode (FIFO_MODE == NONE) or FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it is overwritten, resulting in an over-run error. If in FIFO mode (FIFO_MODE != NONE) 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 over-run error occurs.

UART_THR

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	data_output 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. If in non-FIFO mode or FIFOs are disabled (FCR[0] = 0) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If in FIFO mode and FIFOs are enabled (FCR[0] = 1) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.

UART_DLL

Address: Operational Base + offset (0x0000)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	baud_rate_divisor_L 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 DLH is set, at least 8 clock cycles of the slowest Uart clock should be allowed to pass before transmitting or receiving data.

UART_DLH

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	baud_rate_divisor_H Upper 8 bits of a 16-bit, read/write, Divisor Latch register that contains the baud rate divisor for the UART.

UART_IER

Address: Operational Base + offset (0x0004)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7	RW	0x0	prog_thre_int_en Programmable THRE Interrupt Mode Enable. This is used to enable/disable the generation of THRE Interrupt. 1'b0: Disabled 1'b1: Enabled
6:4	RO	0x0	reserved
3	RW	0x0	modem_status_int_en Enable Modem Status Interrupt. This is used to enable/disable the generation of Modem Status Interrupt. This is the fourth highest priority interrupt. 1'b0: Disabled 1'b1: Enabled
2	RW	0x0	receive_line_status_int_en Enable Receiver Line Status Interrupt. This is used to enable/disable the generation of Receiver Line Status Interrupt. This is the highest priority interrupt. 1'b0: Disabled 1'b1: Enabled
1	RW	0x0	trans_hold_empty_int_en Enable Transmit Holding Register Empty Interrupt.
0	RW	0x0	receive_data_available_int_en 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. 1'b0: Disabled 1'b1: Enabled

UART_IIR

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:6	RO	0x0	fifos_en FIFOs Enabled. This is used to indicate whether the FIFOs are enabled or disabled. 2'b00: Disabled 2'b11: Enabled
5:4	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
3:0	RO	0x1	int_id Interrupt ID. This indicates the highest priority pending interrupt which can be one of the following types: 4'b0000: Modem status 4'b0001: No interrupt pending 4'b0010: THR empty 4'b0100: Received data available 4'b0110: Receiver line status 4'b0111: Busy detect 4'b1100: Character timeout

UART_FCR

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:6	WO	0x0	rcvr_trigger 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. The following trigger levels are supported: 2'b00: 1 character in the FIFO 2'b01: FIFO 1/4 full 2'b10: FIFO 1/2 full 2'b11: FIFO 2 less than ful
5:4	WO	0x0	tx_empty_trigger 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. The following trigger levels are supported: 2'b00: FIFO empty 2'b01: 2 characters in the FIFO 2'b10: FIFO 1/4 full 2'b11: FIFO 1/2 full

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
3	WO	0x0	dma_mode DMA Mode. This determines the DMA signalling mode used for the dma_tx_req_n and dma_rx_req_n output signals when additional DMA handshaking signals are not selected . 1'b0: Mode 0 1'b1: Mode 1
2	WO	0x0	xmit_fifo_reset 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 and single signals when additional DMA handshaking signals are selected . Note that this bit is 'self-clearing'. It is not necessary to clear this bit.
1	WO	0x0	rcvr_fifo_reset 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 and single signals when additional DMA handshaking signals are selected . Note that this bit is 'self-clearing'. It is not necessary to clear this bit.
0	WO	0x0	fifo_en FIFO Enable. 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.

UART_LCR

Address: Operational Base + offset (0x000c)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7	RW	0x0	div_lat_access Divisor Latch Access Bit. Writeable only when UART is not busy (USR[0] is zero), 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.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
6	RW	0x0	break_ctrl 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 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.
5	RO	0x0	reserved
4	RW	0x0	even_parity_sel Even Parity Select. Writeable only when UART is not busy (USR[0] is zero), always readable. This is used to select between even and odd parity, when parity is enabled (PEN set to one). If set to one, an even number of logic 1s is transmitted or checked. If set to zero, an odd number of logic 1s is transmitted or checked.
3	RW	0x0	parity_en Parity Enable. Writeable only when UART is not busy (USR[0] is zero), always readable. This bit is used to enable and disable parity generation and detection in transmitted and received serial character respectively. 1'b0: Parity disabled 1'b1: Parity enabled
2	RW	0x0	stop_bits_num Number of stop bits. Writeable only when UART is not busy (USR[0] is zero), 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, twostop bits are transmitted. Note that regardless of the number of stop bits selected, the receiver checks only the first stop bit. 1'b0: 1 stop bit 1'b1: 1.5 stop bits when DLS (LCR[1:0]) is zero, else 2 stop bit.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
1:0	RW	0x0	data_length_sel Data Length Select. Writeable only when UART is not busy (USR[0] is zero), 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: 2'b00: 5 bits 2'b01: 6 bits 2'b10: 7 bits 2'b11: 8 bits

UART_MCR

Address: Operational Base + offset (0x0010)

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6	RW	0x0	sir_mode_en SIR Mode Enable. This is used to enable/disable the IrDA SIR Mode . 1'b0: IrDA SIR Mode disabled 1'b1: IrDA SIR Mode enabled
5	RW	0x0	auto_flow_ctrl_en Auto Flow Control Enable. 1'b0: Auto Flow Control Mode disabled 1'b1: Auto Flow Control Mode enabled
4	RW	0x0	loopback LoopBack Bit. This is used to put the UART into a diagnostic mode for test purposes.
3	RW	0x0	out2 OUT2. This is used to directly control the user-designated Output2 (out2_n) output. The value written to this location is inverted and driven out on out2_n, that is: 1'b0: out2_n de-asserted (logic 1) 1'b1: out2_n asserted (logic 0)
2	RW	0x0	out1 OUT1
1	RW	0x0	req_to_send 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.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
0	RW	0x0	data_terminal_ready 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, that is: 1'b0: dtr_n de-asserted (logic 1) 1'b1: dtr_n asserted (logic 0)

UART_LSR

Address: Operational Base + offset (0x0014)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7	RO	0x0	receiver_fifo_error Receiver FIFO Error bit. This bit is relevant FIFOs are enabled (FCR[0] set to one). This is used to indicate if there is at least one parity error, framing error, or break indication in the FIFO. 1'b0: No error in RX FIFO 1'b1: Error in RX FIFO
6	RO	0x1	trans_empty Transmitter Empty bit. If FIFOs enabled (FCR[0] set to one), this bit is set whenever the Transmitter Shift Register and the FIFO are both empty. If FIFOs are disabled, this bit is set whenever the Transmitter Holding Register and the Transmitter Shift Register are both empty.
5	RO	0x1	trans_hold_reg_empty Transmit Holding Register Empty bit. If THRE mode is disabled (IER[7] set to zero) and regardless of FIFO's being implemented/enabled or not, this bit indicates that the THR or TX FIFO is empty. This bit is set whenever data is transferred from the THR or TX FIFO to the transmitter shift register and no new data has been written to the THR or TX FIFO. This also causes a THRE Interrupt to occur, if the THRE Interrupt is enabled. If IER[7] set to one and FCR[0] set to one respectively, the functionality is switched to indicate the transmitter FIFO is full, and no longer controls THRE interrupts, which are then controlled by the FCR[5:4] threshold setting.
4	RO	0x0	break_int Break Interrupt bit. This is used to indicate the detection of a break sequence on the serial input data.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
3	RO	0x0	framing_error Framing Error bit. 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.
2	RO	0x0	parity_eror Parity Error bit. This is used to indicate the occurrence of a parity error in the receiver if the Parity Enable (PEN) bit (LCR[3]) is set.
1	RO	0x0	overrun_error Overrun error bit. This is used to indicate the occurrence of an overrun error. This occurs if a new data character was received before the previous data was read.
0	RO	0x0	data_ready Data Ready bit. This is used to indicate that the receiver contains at least one character in the RBR or the receiver FIFO. 1'b0: No data ready 1'b1: Data ready

UART_MSR

Address: Operational Base + offset (0x0018)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7	RO	0x0	data_carrior_detect Data Carrier Detect. This is used to indicate the current state of the modem control line dcd_n.
6	RO	0x0	ring_indicator Ring Indicator. This is used to indicate the current state of the modem control line ri_n.
5	RO	0x0	data_set_ready Data Set Ready. This is used to indicate the current state of the modem control line dsr_n.
4	RO	0x0	clear_to_send Clear to Send. This is used to indicate the current state of the modem control line cts_n.
3	RO	0x0	delta_data_carrier_detect 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.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2	RO	0x0	trailing_edge_ring_indicator Trailing Edge of 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.
1	RO	0x0	delta_data_set_ready 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	RO	0x0	delta_clear_to_send 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.

UART_SCR

Address: Operational Base + offset (0x001c)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	temp_store_space This register is for programmers to use as a temporary storage space.

UART_SRBR

Address: Operational Base + offset (0x0030)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RO	0x00	shadow_rbr This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the 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 (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it is overwritten, resulting in an overrun error. If 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 are preserved, but any incoming data is lost. An overrun error also occurs.

UART_STHR

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x006c)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RO	0x00	shadow_thr This is a shadow register for the THR.

UART_FAR

Address: Operational Base + offset (0x0070)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	fifo_access_test_en This register is use to enable a FIFO access mode for testing, so that the receive FIFO can be written by the master and the transmit FIFO can be read by the master when FIFOs are implemented and enabled. When FIFOs are not enabled it allows the RBR to be written by the master and the THR to be read by the master. 1'b0: FIFO access mode disabled 1'b1: FIFO access mode enabled

UART_TFR

Address: Operational Base + offset (0x0074)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RO	0x00	trans_fifo_read Transmit FIFO Read. These bits are only valid when FIFO access mode is enabled (FAR[0] is set to one).When FIFOs are implemented and enabled, reading this register gives the data at the top of the transmit FIFO. Each consecutive read pops the transmit FIFO and gives the next data value that is currently at the top of the FIFO.

UART_RFW

Address: Operational Base + offset (0x0078)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:10	RO	0x0	reserved
9	WO	0x0	receive_fifo_framing_error Receive FIFO Framing Error. These bits are only valid when FIFO access mode is enabled (FAR[0] is set to one).
8	WO	0x0	receive_fifo_parity_error Receive FIFO Parity Error. These bits are only valid when FIFO access mode is enabled (FAR[0] is set to one).
7:0	WO	0x00	receive_fifo_write Receive FIFO Write Data. These bits are only valid when FIFO access mode is enabled (FAR[0] is set to one). When FIFOs are enabled, the data that is written to the RFWD is pushed into the receive FIFO. Each consecutive write pushes the new data to the next write location in the receive FIFO. When FIFOs not enabled, the data that is written to the RFWD is pushed into the RBR.

UART_USR

Address: Operational Base + offset (0x007c)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4	RO	0x0	receive_fifo_full Receive FIFO Full. This is used to indicate that the receive FIFO is completely full. 1'b0: Receive FIFO not full 1'b1: Receive FIFO Full This bit is cleared when the RX FIFO is no longer full.
3	RO	0x0	receive_fifo_not_empty Receive FIFO Not Empty. This is used to indicate that the receive FIFO contains one or more entries. 1'b0: Receive FIFO is empty 1'b1: Receive FIFO is not empty This bit is cleared when the RX FIFO is empty.
2	RO	0x1	trasn_fifo_empty Transmit FIFO Empty. This is used to indicate that the transmit FIFO is completely empty. 1'b0: Transmit FIFO is not empty 1'b1: Transmit FIFO is empty This bit is cleared when the TX FIFO is no longer empty
1	RO	0x1	trans_fifo_not_full Transmit FIFO Not Full. This is used to indicate that the transmit FIFO in not full. 1'b0: Transmit FIFO is full 1'b1: Transmit FIFO is not full This bit is cleared when the TX FIFO is full.
0	RO	0x0	uart_busy UART Busy. This is indicates that a serial transfer is in progress, when cleared indicates that the uart is idle or inactive. 1'b0: Uart is idle or inactive 1'b1: Uart is busy (actively transferring data)

UART_TFL

Address: Operational Base + offset (0x0080)

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4:0	RW	0x00	trans_fifo_level Transmit FIFO Level. This is indicates the number of data entries in the transmit FIFO.

UART_RFL

Address: Operational Base + offset (0x0084)

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4:0	RO	0x00	receive_fifo_level Receive FIFO Level. This is indicates the number of data entries in the receive FIFO.

UART_SRR

Address: Operational Base + offset (0x0088)

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved
2	WO	0x0	xmit_fifo_reset XMIT FIFO Reset. This is a shadow register for the XMIT FIFO Reset bit (FCR[2]).
1	WO	0x0	rcvr_fifo_reset RCVR FIFO Reset. This is a shadow register for the RCVR FIFO Reset bit (FCR[1]).
0	WO	0x0	uart_reset UART Reset. This asynchronously resets the Uart and synchronously removes the reset assertion. For a two clock implementation both pclk and sclk domains are reset.

UART_SRTS

Address: Operational Base + offset (0x008c)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	shadow_req_to_send Shadow Request to Send. This is a shadow register for the RTS bit (MCR[1]), this can be used to remove the burden of having to performing a read-modify-write on the MCR.

UART_SBCR

Address: Operational Base + offset (0x0090)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	shadow_break_ctrl Shadow Break Control Bit. This is a shadow register for the Break bit (LCR[6]), this can be used to remove the burden of having to performing a read modify write on the LCR.

UART_SDMAM

Address: Operational Base + offset (0x0094)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
0	RW	0x0	shadow_dma_mode Shadow DMA Mode. This is a shadow register for the DMA mode bit (FCR[3]).

UART_SFE

Address: Operational Base + offset (0x0098)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	shadow_fifo_en Shadow FIFO Enable. Shadow FIFO Enable. This is a shadow register for the FIFO enable bit (FCR[0]).

UART_SRT

Address: Operational Base + offset (0x009c)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1:0	RW	0x0	shadow_rcvr_trigger Shadow RCVR Trigger. This is a shadow register for the RCVR trigger bits (FCR[7:6]).

UART_STET

Address: Operational Base + offset (0x00a0)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1:0	RW	0x0	shadow_tx_empty_trigger Shadow TX Empty Trigger. This is a shadow register for the TX empty trigger bits (FCR[5:4]).

UART_HTX

Address: Operational Base + offset (0x00a4)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	halt_tx_en 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. 1'b0: Halt TX disabled 1'b1: Halt TX enabled

UART_DMASA

Address: Operational Base + offset (0x00a8)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
0	WO	0x0	dma_software_ack This register is use to perform a DMA software acknowledge if a transfer needs to be terminated due to an error condition.

UART_CPR

Address: Operational Base + offset (0x00f4)

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:16	RO	0x00	FIFO_MODE 0x00 = 0 0x01 = 16 0x02 = 32 to 0x80 = 2048 0x81- 0xff = reserved
15:14	RO	0x0	reserved
13	RO	0x0	DMA_EXTRA 1'b0: FALSE 1'b1: TRUE
12	RO	0x0	UART_ADD_ENCODED_PARAMS 1'b0: FALSE 1'b1: TRUE
11	RO	0x0	SHADOW 1'b0: FALSE 1'b1: TRUE
10	RO	0x0	FIFO_STAT 1'b0: FALSE 1'b1: TRUE
9	RO	0x0	FIFO_ACCESS 1'b0: FALSE 1'b1: TRUE
8	RO	0x0	NEW_FEAT 1'b0: FALSE 1'b1: TRUE
7	RO	0x0	SIR_LP_MODE 1'b0: FALSE 1'b1: TRUE
6	RO	0x0	SIR_MODE 1'b0: FALSE 1'b1: TRUE
5	RO	0x0	THRE_MODE 1'b0: FALSE 1'b1: TRUE

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
4	RO	0x0	AFCE_MODE 1'b0: FALSE 1'b1: TRUE
3:2	RO	0x0	reserved
1:0	RO	0x0	APB_DATA_WIDTH 2'b00: 8 bits 2'b01: 16 bits 2'b10: 32 bits 2'b11: Reserved

UART_UCV

Address: Operational Base + offset (0x00f8)

Bit	Attr	Reset Value	Description
31:0	RO	0x3330382a	ver ASCII value for each number in the version

UART_CTR

Address: Operational Base + offset (0x00fc)

Bit	Attr	Reset Value	Description
31:0	RO	0x44570110	peripheral_id This register contains the peripherals identification code.

23.5 Interface Description

Table 23-1 UART Interface Description

Modulepin	Dir	Pad name	IOMUX
UART0 Interface			
uart0_sin	I	GPIO0_B3/UART0_RX	GRF_GPIO0B_IOMUX[7:6]=2'b01
uart0_sout	O	GPIO0_B2/UART0_TX	GRF_GPIO0B_IOMUX[5:4]=2'b01
uart0_cts_n	I	GPIO0_B4/UART0_CTS	GRF_GPIO0B_IOMUX[9:8]=2'b01
uart0_rts_n	O	GPIO0_B5/UART0_RTS/TEST_CLK1	GRF_GPIO0B_IOMUX[11:10]=2'b01
UART1m0 Interface			
uart1_sin	I	GPIO4_B0/SDMMC1_D0/UART1_RX_M0	GRF_GPIO4BL_IOMUX[2:0]= 3'b010
uart1_sout	O	GPIO4_B1/SDMMC1_D1/UART1_TX_M0	GRF_GPIO4BL_IOMUX[6:4]= 3'b010
uart1_cts_n	I	GPIO4_B2/SDMMC1_D2/UART1_CTS	GRF_GPIO1BL_IOMUX[10:8]= 3'b010
uart1_rts_n	O	GPIO4_B3/SDMMC1_D3/UART1_RTS	GRF_GPIO1BL_IOMUX[14:12]=3'b010
UART1m1 Interface			
uart2m0_sin	I	GPIO1_B4/SPI0_MOSI/I2C2_SCL_M1/UART1_RX_M1	GRF_GPIO1BH_IOMUX[2:0]=3'b011
uart2m0_sout	O	GPIO1_B5/SPI0_MISO/I2C2_SDA_M1/UART1_TX_M1	GRF_GPIO1BH_IOMUX[6:4]= 3'b011

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Modulepin	Dir	Pad name	IOMUX
UART2m0 Interface			
uart2m0_sin	I	GPIO4_A3/SDMMC0_D1/UART2_RX_M0	GRF_GPIO4AL_IOMUX[10:8]=3'b010
uart2m0_sout	O	GPIO4_A2/SDMMC0_D0/UART2_TX_M0	GRF_GPIO4AL_IOMUX[14:12]=3'b010
UART2m1 Interface			
uart2m1_sin	I	GPIO2_D1/I2C3_SDA/UART2_RX_M1	GRF_GPIO2DL_IOMUX[6:4]=3'b010
uart2m1_sout	O	GPIO2_D0/I2C3_SCL/UART2_TX_M1	GRF_GPIO2DL_IOMUX[2:0]=3'b010
UART2m2 Interface			
uart2m2_sin	I	GPIO3_A4/I2S1_SDI/UART2_RX_M2	GRF_GPIO3AH_IOMUX[14:12]=3'b010
uart2m2_sout	O	GPIO3_A3/I2S1_SDO/UART2_TX_M2	GRF_GPIO3AH_IOMUX[2:0]=3'b010
UART3m0 Interface			
uart3_sin	I	GPIO0_C4/PWM3/UART3_RX	GRF_GPIO0C_IOMUX[9:8]= 2'b10
uart3_sout	O	GPIO0_C3/PWM1/UART3_TX	GRF_GPIO0C_IOMUX[7:6]= 2'b10
uart3_cts_n	I	GPIO0_C6/PCIE_CLKREQN_M1/UART3_CTS	GRF_GPIO0C_IOMUX[13:12]= 2'b10
uart3_rts_n	O	GPIO0_C7/UART3_RTS	GRF_GPIO0C_IOMUX[15:14]= 2'b10
UART4 Interface			
uart4_sin	I	GPIO4_B4/UART4_RX/SPI1_CLK_M0	GRF_GPIO4BH_IOMUX[2:0]= 3'b001
uart4_sout	O	GPIO4_B5/UART4_TX/SPI1_MOSI_M0	GRF_GPIO4BH_IOMUX[6:4]= 3'b001
uart4_cts_n	I	GPIO4_B6/UART4_CTS/SPI1_CSN0_M0	GRF_GPIO4BH_IOMUX[10:8]= 3'b001
uart4_rts_n	O	GPIO4_B7/UART4_RTS/SPI1_MISO_M0	GRF_GPIO4BH_IOMUX[14:12]= 3'b001
UART5 Interface			
uart5_sin	I	GPIO3_C3/LCDC_D9/UART5_RX/I2C4_SDA	GRF_GPIO3CL_IOMUX[14:12]= 3'b010
uart5_sout	O	GPIO3_C2/LCDC_D8/UART5_TX/I2C4_SCL	GRF_GPIO3CL_IOMUX[10:8]= 3'b010
UART6 Interface			
uart6_sin	I	GPIO3_C5/LCDC_D11/UART6_RX	GRF_GPIO3CH_IOMUX[6:4]= 3'b010
uart6_sout	O	GPIO3_C4/LCDC_D10/UART6_TX	GRF_GPIO3CH_IOMUX[2:0]= 3'b010
UART7 Interface			
uart7_sin	I	GPIO3_C7/LCDC_D13/UART7_RX/SPI1_CLK_M1	GRF_GPIO3CH_IOMUX[14:12]= 3'b010
uart7_sout	O	GPIO3_C6/LCDC_D12/UART7_TX	GRF_GPIO3CH_IOMUX[10:8]= 3'b010

The I/O interface of UART2 can be chosen by setting GRF_IOFUNC_SEL0[15:14]bit (0: UART2_M0; 1:UART2_M1; 2: UART2_M2. The I/O interface of UART1 can be chosen by setting GRF_IOFUNC_SEL0[13]bit, if this bit is set to 1, UART1 uses the UART1m1 I/O interface.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

23.6 Application Notes

23.6.1 None FIFO Mode Transfer Flow

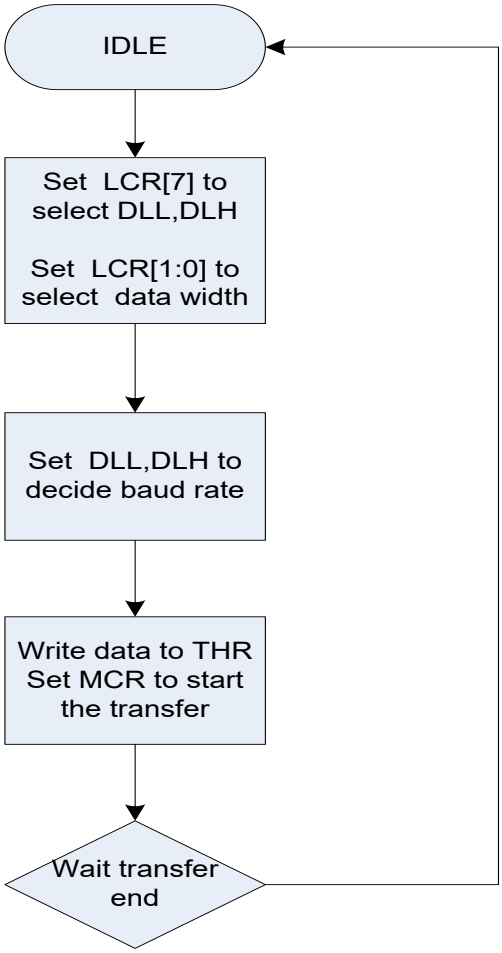


Fig. 23-8 UART none fifo mode

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

23.6.2 FIFO Mode Transfer Flow

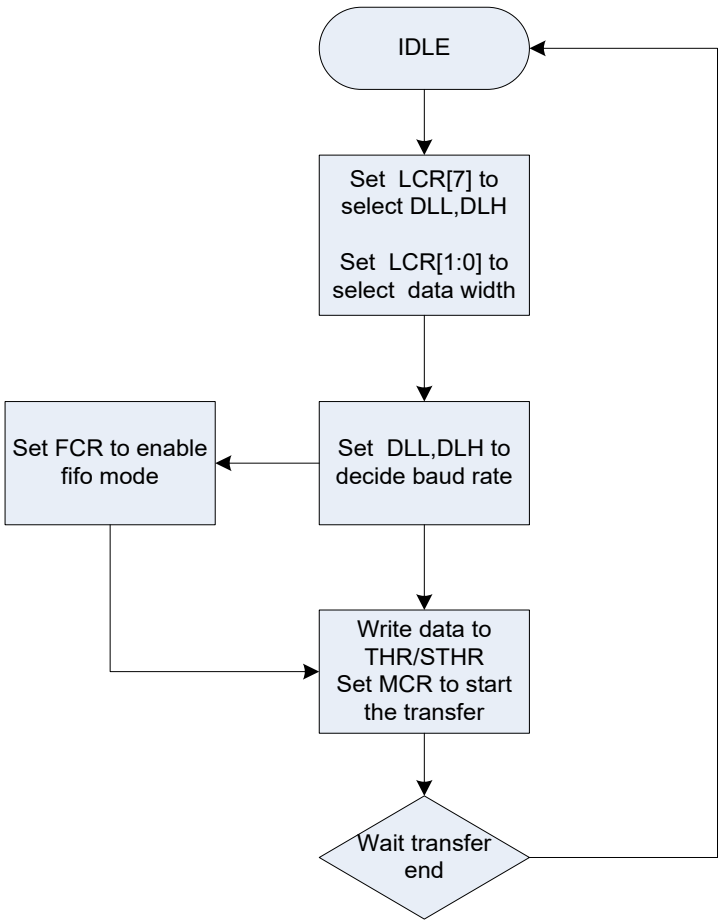


Fig. 23-9 UART fifo mode

The UART is an APB slave performing:
Serial-to-parallel conversion on data received from a peripheral device.
Parallel-to-serial conversion on data transmitted to the peripheral device.
The CPU reads and writes data and control/status information through the APB interface.
The transmitting and receiving paths are buffered with internal FIFO memories enabling up to 64-bytes to be stored independently in both transmit and receive modes. A baud rate generator can generate a common transmit and receive internal clock input. The baud rates will depend on the internal clock frequency. The UART will also provide transmit, receive and exception interrupts to system. A DMA interface is implemented for improving the system performance.

23.6.3 Baud Rate Calculation

UART clock generation

The following figures shows the UART clock generation. UART0~7 source clocks can be selected from four PLL outputs.
UART clocks can be generated by 1 to 32 division of its source clock, or can be fractionally divided again.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

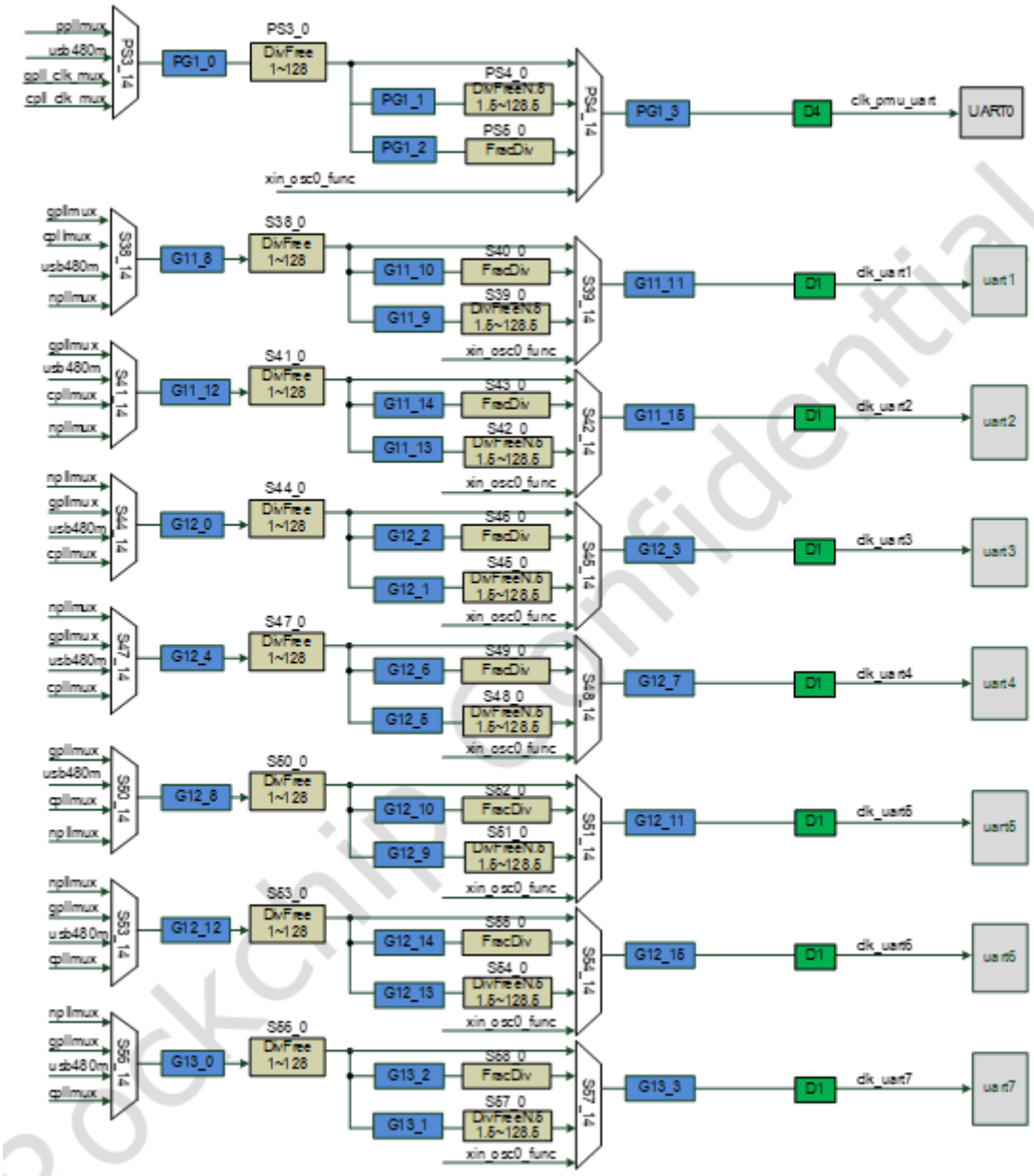


Fig. 23-10 UART clock generation

UART baud rate configuration

The following table provides some reference configuration for different UART baud rates.

Table 23-2 UART baud rate configuration

Baud Rate	Reference Configuration
115.2 Kbps	Configure GENERAL PLL to get 1200MHz clock output; Divide 1200MHz clock by 46875/72 to get 1.8432MHz clock; Configure UART_DLL to 1.
460.8 Kbps	Configure GENERAL PLL to get 1200MHz clock output; Divide 1200MHz clock by 46875/288 to get 7.3728MHz clock; Configure UART_DLL to 1.
921.6 Kbps	Configure GENERAL PLL to get 1200MHz clock output; Divide 1200MHz clock by 46875/576 to get 14.7456MHz clock; Configure UART_DLL to 1.
1.5 Mbps	Choose GENERAL PLL to get 1200MHz clock output; Divide 1200MHz clock by 50 to get 24MHz clock; Configure UART_DLL to 1.
3 Mbps	Choose GENERAL PLL to get 1200MHz clock output; Divide 1200MHz clock by 1200/48 to get 48MHz clock; Configure UART_DLL to 1.

Rockchip Confidential

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Baud Rate	Reference Configuration
4 Mbps	Configure GENERAL PLL to get 1200MHz clock output; Divide 1200MHz clock by 480/7.5 to get 64MHz clock; Configure UART_DLL to 1.

23.6.4 CTS_n and RTS_n Polarity Configurable

The polarity of cts_n and rts_n ports can be configured by GRF registers.

- When grf_uart_cts_sel[*] is configured as 1'b1, cts_n is high active. Otherwise, lowactive.
- When grf_uart_rts_sel[*] is configured as 1'b1, rts_n is high active. Otherwise, lowactive.

Table 23-3 UART cts_n and rts_n polarity configuration

UART	GRF_UART_CTS_SEL	GRF_UART_RTS_SEL
UART0	PMUGRF_SOC_CON0[6]	PMUGRF_SOC_CON0[5]
UART1	GRF_SOC_CON2[2]	GRF_SOC_CON2[3]
UART2	GRF_SOC_CON2[4]	GRF_SOC_CON2[5]
UART3	GRF_SOC_CON2[6]	GRF_SOC_CON2[7]
UART4	GRF_SOC_CON2[8]	GRF_SOC_CON2[9]
UART5	GRF_SOC_CON2[10]	GRF_SOC_CON2[11]
UART6	GRF_SOC_CON1[8]	GRF_SOC_CON1[9]
UART7	GRF_SOC_CON1[10]	GRF_SOC_CON1[11]

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 24 I2C Interface

24.1 Overview

The Inter-Integrated Circuit (I2C) is a two wired (SCL and SDA), bi-directional serial bus that provides an efficient and simple method of information exchange between devices. This I2C bus controller supports master mode acting as a bridge between AMBA protocol and generic I2C bus system.
I2C Controller supports the following features:

- Support 6 independent I2C: I2C0, I2C1, I2C2, I2C3,I2C4,I2C5
- Item Compatible with I2C-bus
- AMBA APB slave interface
- Supports master mode of I2C bus
- Software programmable clock frequency and transfer rate up to 400Kbit/sec
- Supports 7 bits and 10 bits addressing modes
- Interrupt or polling driven multiple bytes data transfer
- Clock stretching and wait state generation
- Filter out glitch on SCL and SDA

24.2 Block Diagram

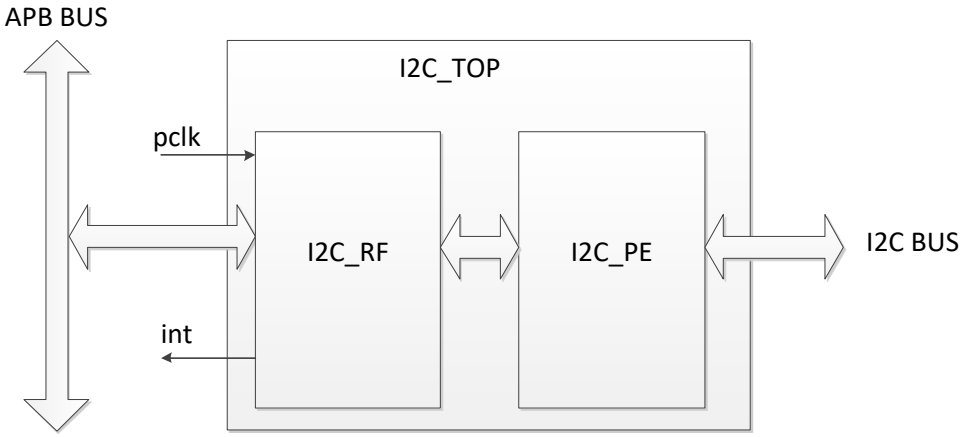


Fig. 24-1 I2C architecture

24.2.1 I2C_RF

I2C_RF module is used to control the I2C controller operation by the host with APB interface. It implements the register set and the interrupt functionality. The CSR component operates synchronously with the pclk clock.

24.2.2 I2C_PE

I2C_PE module implements the I2C master operation for transmit data to and receive data from other I2C devices. The I2C master controller operates synchronously with the pclk.

24.2.3 I2C_TOP

I2C_TOP module is the top module of the I2C controller.

24.3 Function Description

This chapter provides a description about the functions and behavior under various conditions.
The I2C controller supports only Masterfunction. Itsupports the 7-bits/10-bits addressing mode and support general call address. The maximum clock frequency and transfer rate can be up to 400Kbit/sec.
The operations of I2C controller is divided to 2 parts and described separately: initialization

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

and master mode programming.

24.3.1 Initialization

The I2C controller is based on AMBA APB bus architecture and usually is part of a SOC. So before I2C operates, some system setting and configuration must be conformed, which includes:

- I2C interrupt connection type: CPU interrupt scheme should be considered. If the I2C interrupt is connected to extra Interrupt Controller module, we need decide the INTC vector.
- I2C Clock Rate: The I2C controller uses the APB clock as the working clock so the APB clock will determine the I2C bus clock. The correct register setting is subject to the system requirement.

24.3.2 Master Mode Programming

- SCL Clock
  When the I2C controller is programmed in Master mode, the SCL frequency is determined by I2C_CLKDIV register. The SCL frequency is calculated by the following formula:
  SCL Divisor = 8*(CLKDIVL + 1 + CLKDIVH + 1)
  SCL = PCLK/ SCLK Divisor

- Data Receiver Register Access
  When the I2C controller received MRXCNT bytes data， CPU can get the data through register RXDATA0 ~ RXDATA7. The controller can receive up to 32 bytes' data in one transaction.
  When MRXCNT register is written, the I2C controller will start to drive SCL to receive data.

- Transmit Transmitter Register
  Data to transmit are written to TXDATA0~7 by CPU. The controller can transmit up to 32 bytes' data in one transaction. The lower byte will be transmitted first.
  When MTXCNT register is written, the I2C controller will start to transmit data.

- Start Command
  Write 1 to I2C_CON[3], the controller will send I2C start command.

- Stop Command
  Write 1 to I2C_CON[4], the controller will send I2C stop command

- I2C Operation mode
  There are four i2c operation modes.
  - When I2C_CON[2:1] is 2'b00, the controller transmit all valid data in TXDATA0~TXDATA7 byte by byte. The controller will transmit lower byte first.
  - When I2C_CON[2:1] is 2'b01,the controller will transmit device address in MRXADDR first (Write/Read bit = 0) and then transmit device register address in MRXRADDR. After that, the controller will assert restart signal and resend MRXADDR (Write/Read bit = 1). At last, the controller enter receive mode.
  - When I2C_CON[2:1] is 2'b10, the controller is in receive mode, it will trigger clock to read MRXCNT byte data.
  - When I2C_CON[2:1] is 2'b11, the controller will transmit device address in MRXADDR first (Write/Read bit = 1) and then transmit device register address in MRXRADDR . After that, the controller will assert restart signal and resend MRXADDR (Write/Read bit = 1). At last, the controller enter receive mode.

- Read/Write Command
  - When I2C_OPMODE(I2C_CON[2:1]) is 2'b01 or 2'b11, the Read/Write command bit is decided by controller itself.
  - In RX only mode (I2C_CON[2:1] is 2'b10), the Read/Write command bit is decided

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

by MRXADDR[0].
- In TX only mode (I2C_CON[[2:1] is 2'b00), the Read/Write command bit is decided by TXDATA[0].

- Master Interrupt Condition
  There are 7 interrupt bits in I2C_ISR register related to master mode.
  - Byte transmitted finish interrupt (Bit 0): The bit is asserted when Master completed transmitting a byte.
  - Byte received finish interrupt (Bit 1): The bit is asserted when Master completed receiving a byte.
  - MTXCNT bytes data transmitted finish interrupt (Bit 2): The bit is asserted when Master completed transmitting MTXCNT bytes.
  - MRXCNT bytes data received finish interrupt (Bit 3): The bit is asserted when Master completed receiving MRXCNT bytes.
  - Start interrupt (Bit 4): The bit is asserted when Master finished asserting start command to I2C bus.
  - Stop interrupt (Bit 5): The bit is asserted when Master finished asserting stop command to I2C bus.
  - NAK received interrupt (Bit 6): The bit is asserted when Master received a NAK handshake.

- Last byte acknowledge control
  - If I2C_CON[5] is 1, the I2C controller will transmit NAK handshake to slave when the last byte received in RX only mode.
  - If I2C_CON[5] is 0, the I2C controller will transmit ACK handshake to slave when the last byte received in RX only mode.

- How to handle NAK handshake received
  - If I2C_CON[6] is 1, the I2C controller will stop all transactions when NAK handshake received. And the software should take responsibility to handle the problem.
  - If I2C_CON[6] is 0, the I2C controller will ignore all NAK handshake received.

- I2C controller data transfer waveform
  - Bit transferring
    - Data Validity
      The SDA line must be stable during the high period of SCL, and the data on SDA line can only be changed when SCL is in low state.

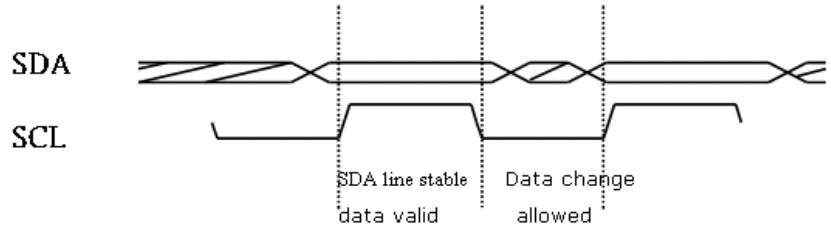


Fig. 24-2 I2C DATA Validity

    - START and STOP conditions
      START condition occurs when SDA goes low while SCL is in high period. STOP condition is generated when SDA line goes high while SCL is in high state.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

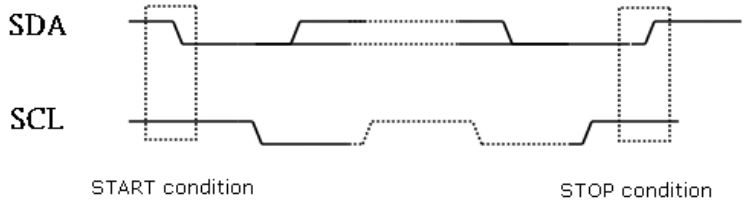


Fig. 24-3 I2C Start and stop conditions

- Data transfer
  - Acknowledge
    After a byte of data transferring (clocks labeled as 1~8), in 9th clock the receiver must assert an ACK signal on SDA line, if the receiver pulls SDA line to low, it means "ACK", on the contrary, it's "NOT ACK".

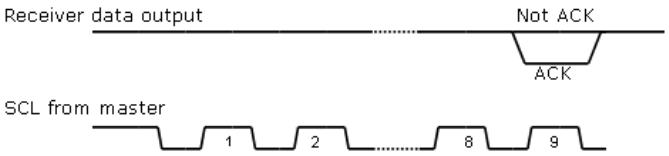


Fig. 24-4 I2C Acknowledge

  - Byte transfer
    The master own I2C bus might initiate multi byte to transfer to a slave.The transfer starts from a "START" command and ends in a "STOP"command. After every byte transfer, the receiver must reply an ACK to transmitter.

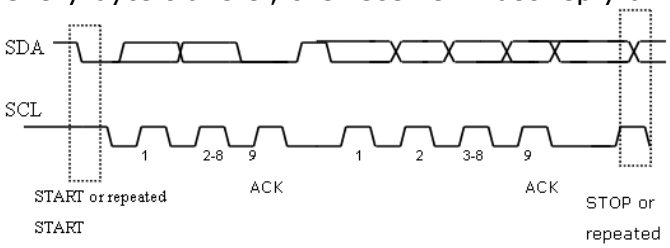


Fig. 24-5 I2C byte transfer

24.4 Register Description

24.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
RKI2C_CON	0x0000	W	0x00030000	control register
RKI2C_CLKDIV	0x0004	W	0x00000001	clock divider register，I2C CLK = PCLK / (16*CLKDIV)
RKI2C_MRXADDR	0x0008	W	0x00000000	the slave address accessed for master rx mode
RKI2C_MRXRADDR	0x000c	W	0x00000000	the slave register address accessed for master rx mode
RKI2C_MTXCNT	0x0010	W	0x00000000	master transmit count.specify the total bytes to be transmit (0~32)
RKI2C_MRXCNT	0x0014	W	0x00000000	master rx count.specify the total bytes to be recieved(0~32)
RKI2C_IEN	0x0018	W	0x00000000	interrupt enable register
RKI2C_IPD	0x001c	W	0x00000000	interrupt pending register

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
RKI2C_FCNT	0x0020	W	0x00000000	finished count: the count of data which has been transmitted or receivedfor debug purpose
RKI2C_SCL_OE_DB	0x0024	W	0x00000020	slave hold debounce configure register
RKI2C_TXDATA0	0x0100	W	0x00000000	I2C tx data register 0
RKI2C_TXDATA1	0x0104	W	0x00000000	I2C tx data register 1
RKI2C_TXDATA2	0x0108	W	0x00000000	I2C tx data register 2
RKI2C_TXDATA3	0x010c	W	0x00000000	I2C tx data register 3
RKI2C_TXDATA4	0x0110	W	0x00000000	I2C tx data register 4
RKI2C_TXDATA5	0x0114	W	0x00000000	I2C tx data register 5
RKI2C_TXDATA6	0x0118	W	0x00000000	I2C tx data register 6
RKI2C_TXDATA7	0x011c	W	0x00000000	I2C tx data register 7
RKI2C_RXDATA0	0x0200	W	0x00000000	I2C rx data register 0
RKI2C_RXDATA1	0x0204	W	0x00000000	I2C rx data register 1
RKI2C_RXDATA2	0x0208	W	0x00000000	I2C rx data register 2
RKI2C_RXDATA3	0x020c	W	0x00000000	I2C rx data register 3
RKI2C_RXDATA4	0x0210	W	0x00000000	I2C rx data register 4
RKI2C_RXDATA5	0x0214	W	0x00000000	I2C rx data register 5
RKI2C_RXDATA6	0x0218	W	0x00000000	I2C rx data register 6
RKI2C_RXDATA7	0x021c	W	0x00000000	I2C rx data register 7
RKI2C_ST	0x0220	W	0x00000000	status debug register
RKI2C_DBGCTRL	0x0224	W	0x00000000	Debug config register

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

24.4.2 Detail Register Description

RKI2C_CON

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:16	RO	0x0003	version rki2c version information
15:14	RW	0x0	stop_setup stop setup config: TSU;sto = (stop_setup + 1) * T(SCL_HIGH) + Tclk_i2c
13:12	RW	0x0	start_setup start setup config: TSU;sta = (start_setup + 1) * T(SCL_HIGH) + Tclk_i2c THD;sta = (start_setup + 2) * T(SCL_HIGH) - Tclk_i2c
11	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
10:8	RW	0x0	data_upd_st SDA update point config: Used to config sda change state when scl is low, used to adjust setup/hold time 4'bn: Thold = (n + 1) * Tclk_i2c Note: 0 <= n <= 5
7	RO	0x0	reserved
6	RW	0x0	act2nak operation when NAK handshake is received: 1'b0: ignored 1'b1: stop transaction
5	RW	0x0	ack last byte acknowledge control in master receive mode: 1'b0: ACK 1'b1: NAK
4	RW	0x0	stop stop enable, when this bit is written to 1, I2C will generate stop signal.
3	RW	0x0	start start enable, when this bit is written to 1, I2C will generate start signal.
2:1	RW	0x0	i2c_mode i2c mode select: 2'b00: transmit only 2'b01: transmit address (device + register address) --> restart --> transmit address -> receive only 2'b10: receive only 2'b11: transmit address (device + register address, write/read bit is 1) --> restart --> transmit address (device address) --> receive data
0	RW	0x0	i2c_en i2c module enable: 1'b0: not enable 1'b1: enable

RKI2C_CLKDIV

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	CLKDIVH scl high level clock count: T(SCL_HIGH) = Tclk_i2c * (CLKDIVH + 1) * 8
15:0	RW	0x0001	CLKDIVL scl low level clock count: T(SCL_LOW) = Tclk_i2c * (CLKDIVL + 1) * 8

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

RKI2C_MRXADDR

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved
26	RW	0x0	addhvld address high byte valid: 1'b0: invalid 1'b1: valid
25	RW	0x0	addmvld address middle byte valid: 1'b0: invalid 1'b1: valid
24	RW	0x0	addlvld address low byte valid: 1'b0: invalid 1'b1: valid
23:0	RW	0x000000	saddr master address register. the lowest bit indicate write or read 24 bits address register

RKI2C_MRXRADDR

Address: Operational Base + offset (0x000c)

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved
26	RW	0x0	sraddhvld address high byte valid: 1'b0: invalid 1'b1: valid
25	RW	0x0	sraddmvld address middle byte valid: 1'b0: invalid 1'b1: valid
24	RW	0x0	sraddlvld address low byte valid: 1'b0: invalid 1'b1: valid
23:0	RW	0x000000	sraddr slave register address accessed. 24 bits register address

RKI2C_MTXCNT

Address: Operational Base + offset (0x0010)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x00	mtxcnt master transmit count. 6 bits counter

RKI2C_MRXCNT

Address: Operational Base + offset (0x0014)

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x00	mrxcnt master rx count. 6 bits counter

RKI2C_IEN

Address: Operational Base + offset (0x0018)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7	RW	0x0	slavehdsclen slave hold scl interrupt enable: 1'b0: disable 1'b1: enable
6	RW	0x0	nakrcvien NAK handshake received interrupt enable: 1'b0: disable 1'b1: enable
5	RW	0x0	stopien stop operation finished interrupt enable: 1'b0: disable 1'b1: enable
4	RW	0x0	startien start operation finished interrupt enable: 1'b0: disable 1'b1: enable
3	RW	0x0	mbrfien MRXCNT data received finished interrupt enable: 1'b0: disable 1'b1: enable
2	RW	0x0	mbtfien MTXCNT data transfer finished interrupt enable: 1'b0: disable 1'b1: enable
1	RW	0x0	brfien byte rx finished interrupt enable: 1'b0: disable 1'b1: enable
0	RW	0x0	btfien byte tx finished interrupt enable: 1'b0: disable 1'b1: enable

RKI2C_IPD

Address: Operational Base + offset (0x001c)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7	RW	0x0	slavehdsclipd slave hold scl interrupt pending bit: 1'b0: no interrupt available 1'b1: slave hold scl interrupt appear, write 1 to clear

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
6	W1C	0x0	nakrcvipd NAK handshake received interrupt pending bit: 1'b0: no interrupt available 1'b1: NAK handshake received interrupt appear, write 1 to clear
5	W1C	0x0	stopipd stop operation finished interrupt pending bit: 1'b0: no interrupt available 1'b1: stop operation finished interrupt appear, write 1 to clear
4	W1C	0x0	startipd start operation finished interrupt pending bit: 1'b0: no interrupt available 1'b1: start operation finished interrupt appear, write 1 to clear
3	W1C	0x0	mbrfipd MRXCNT data received finished interrupt pending bit: 1'b0: no interrupt available 1'b1: MRXCNT data received finished interrupt appear, write 1 to clear
2	W1C	0x0	mbtfipd MTXCNT data transfer finished interrupt pending bit: 1'b0: no interrupt available 1'b1: MTXCNT data transfer finished interrupt appear, write 1 to clear
1	W1C	0x0	brfipd byte rx finished interrupt pending bit: 1'b0: no interrupt available 1'b1: byte rx finished interrupt appear, write 1 to clear
0	W1C	0x0	btfipd byte tx finished interrupt pending bit: 1'b0: no interrupt available 1'b1: byte tx finished interrupt appear, write 1 to clear

RKI2C_FCNT

Address: Operational Base + offset (0x0020)

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RO	0x00	fcnt the count of data which has been transmitted or received for debug purpose

RKI2C_SCL_OE_DB

Address: Operational Base + offset (0x0024)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
7:0	RW	0x20	scl_oe_db slave hold scl debounce. cycles for debounce (unit: Tclk_i2c)

RKI2C_TXDATA0

Address: Operational Base + offset (0x0100)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	txdata0 data0 to be transmitted. 32 bits data

RKI2C_TXDATA1

Address: Operational Base + offset (0x0104)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	txdata1 data1 to be transmitted. 32 bits data

RKI2C_TXDATA2

Address: Operational Base + offset (0x0108)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	txdata2 data2 to be transmitted. 32 bits data

RKI2C_TXDATA3

Address: Operational Base + offset (0x010c)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	txdata3 data3 to be transmitted. 32 bits data

RKI2C_TXDATA4

Address: Operational Base + offset (0x0110)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	txdata4 data4 to be transmitted. 32 bits data

RKI2C_TXDATA5

Address: Operational Base + offset (0x0114)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	txdata5 data5 to be transmitted. 32 bits data

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

RKI2C_TXDATA6

Address: Operational Base + offset (0x0118)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	txdata6 data6 to be transmitted. 32 bits data

RKI2C_TXDATA7

Address: Operational Base + offset (0x011c)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	txdata7 data7 to be transmitted. 32 bits data

RKI2C_RXDATA0

Address: Operational Base + offset (0x0200)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	rxdata0 data0 received. 32 bits data

RKI2C_RXDATA1

Address: Operational Base + offset (0x0204)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	rxdata1 data1 received. 32 bits data

RKI2C_RXDATA2

Address: Operational Base + offset (0x0208)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	rxdata2 data2 received. 32 bits data

RKI2C_RXDATA3

Address: Operational Base + offset (0x020c)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	rxdata3 data3 received. 32 bits data

RKI2C_RXDATA4

Address: Operational Base + offset (0x0210)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	rxdata4 data4 received. 32 bits data

RKI2C_RXDATA5

Address: Operational Base + offset (0x0214)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	rxdata5 data5 received. 32 bits data

RKI2C_RXDATA6

Address: Operational Base + offset (0x0218)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	rxdata6 data6 received. 32 bits data

RKI2C_RXDATA7

Address: Operational Base + offset (0x021c)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	rxdata7 data7 received. 32 bits data

RKI2C_ST

Address: Operational Base + offset (0x0220)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RO	0x0	scl_st scl status: 1'b0: scl status low 1'b0: scl status high
0	RO	0x0	sda_st sda status: 1'b0: sda status low 1'b0: sda status high

RKI2C_DBGCTRL

Address: Operational Base + offset (0x0224)

Bit	Attr	Reset Value	Description
31:15	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
14	RW	0x0	h0_check_scl 1'b0: Check if scl been pull down by slave at the whole SCL_HIGH. 1'b1: Check if scl been pull down by slave only at the h0 of SCL_HIGH(SCL_HIGH including h0~h7).
13	RW	0x0	nak_release_scl 1'b0: Hold scl as low when recieved nack 1'b1: Release scl as high when recieved nack
12	RW	0x0	flt_en SCL edage glitch filter enable 1'b0: disable 1'b1: enable
11:8	RW	0x0	slv_hold_scl_th Slave hold scl threshold = slv_hold_scl_db * Tclk_i2c
7:4	RW	0x0	flt_r Filter scl rising edge glitches of width less than flt_r * Tclk_i2c
3:0	RW	0x0	flt_f Filter scl falling edge glitches of width less than flt_f * Tclk_i2c

24.5 Interface Description

Table 24-1 I2C Interface Description

Module pin	Direction	Pin name	IOMUX
I2C0 Interface			
i2c0_sda	I/O	GPIO1_D0/UART1_RX/I2C0_SDA/SPI2_CLK	GRF_GPIO1D_IOMUX[1:0]=2'b10
i2c0_scl	I/O	GPIO1_D1/UART1_TX/I2C0_SCL/SPI2_CSN0	GRF_GPIO1D_IOMUX[3:2]=2'b10
I2C1 Interface			
i2c1_sda	I/O	GPIO0_B3/I2C1_SDA	GRF_GPIO0B_IOMUX[7:6]=2'b01
i2c1_scl	I/O	GPIO0_B4/I2C1_SCL	GRF_GPIO0B_IOMUX[9:8]=2'b01
I2C2 Interface			
i2c2_sda	I/O	GPIO2_A2/UART0_CTSN/SPI0_CLK/I2C2_SDA	GRF_GPIO2A_IOMUX[5:4]=2'b11
i2c2_scl	I/O	GPIO2_A3/UART0_RTSN/SPI0_CSN0/I2C2_SCL	GRF_GPIO2A_IOMUX[7:6]=2'b11
I2C3 M0 Interface			
i2c3m0_sda	I/O	GPIO0_B7/PWM2/I2C3_SDA_M0	GRF_GPIO0B_IOMUX[15:14]=2'b10
i2c3m0_scl	I/O	GPIO0_C0/PWM3/I2C3_SCL_M0	GRF_GPIO0C_IOMUX[1:0]=2'b10
I2C3 M1 Interface			

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

i2c3m1_sda	I/O	GPIO3_B4/FLASH_RDY/I2C3_SDA_M1/SPI1_MOSI/UART3_RX	GRF_GPIO3B_IOMUX[11:8]=4'b0010
i2c3m1_scl	I/O	GPIO3_B5/FLASH_CSN0/I2C3_SCL_M1/SPI1_CSN0/UART3_TX 3scl_GPIO1B5vccio0	GRF_GPIO3B_IOMUX[15:12]=4'b0010

The I/O interface of I2C3 can be chosen by setting GRF_SOC_CON5[4] bit, if this bit is set to 1, I2C3 uses the I2C3m1 I/O interface, if those bit is set to 0, I2C3 uses the I2C3m0 I/O interface.

24.6 Application Notes

The I2C controller core operation flow chart below is to describe how the software configures and performs an I2C transaction through this I2C controller core. Descriptions are divided into 3 sections, transmit only mode, receive only mode, and mix mode. Users are strongly advised to follow

- Transmit only mode (I2C_CON[1:0]=2'b00)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

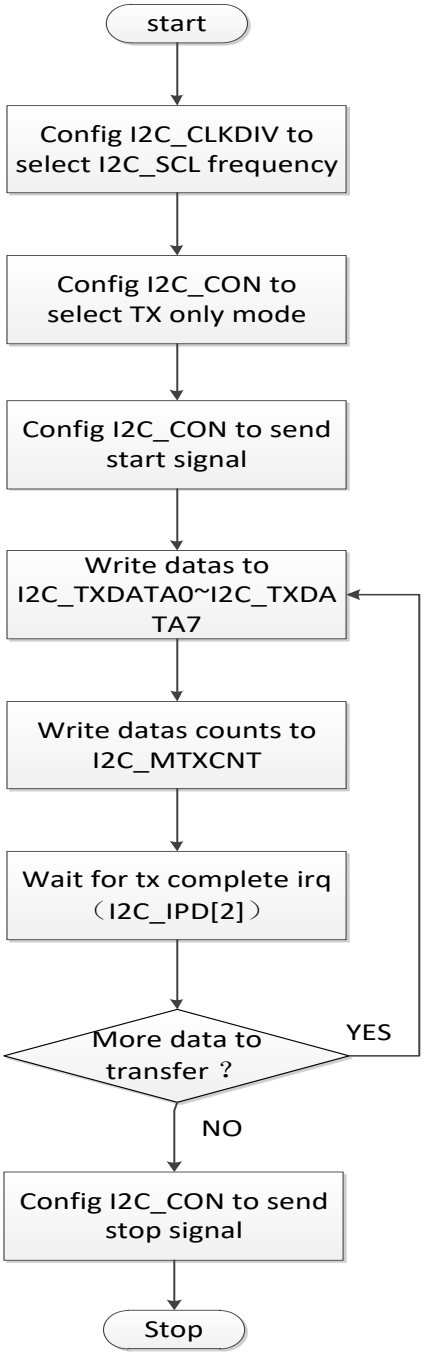


Fig. 24-6 I2C Flow chat for transmit only mode

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

- Receive only mode (I2C_CON[1:0]=2'b10)

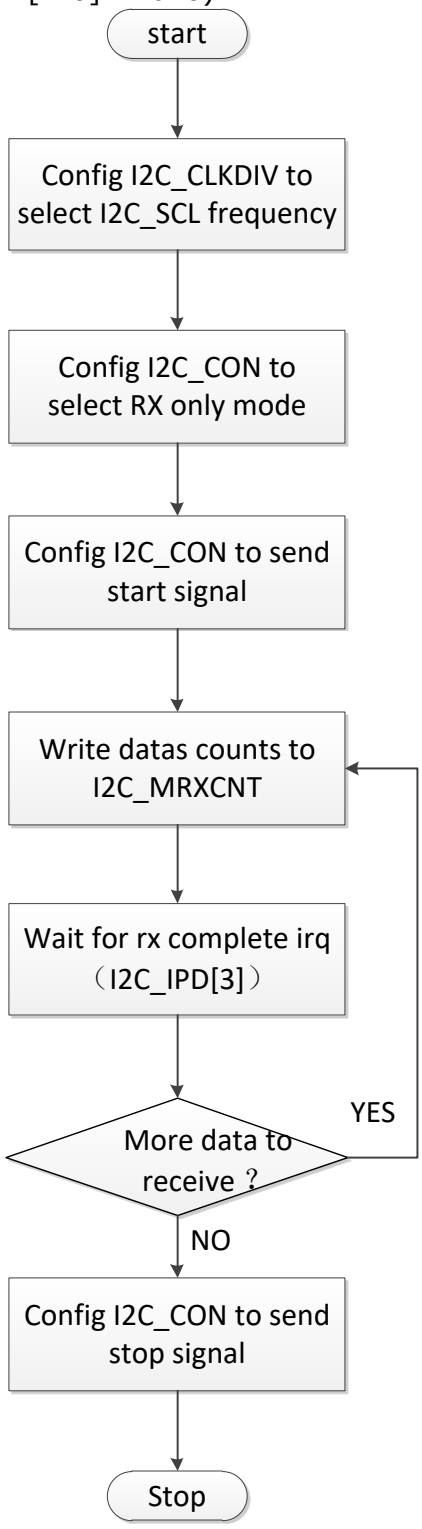


Fig. 24-7 I2C Flow chat for receive only mode

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

- Mix mode (I2C_CON[1:0]=2'b01 or I2C_CON[1:0]=2'b11)

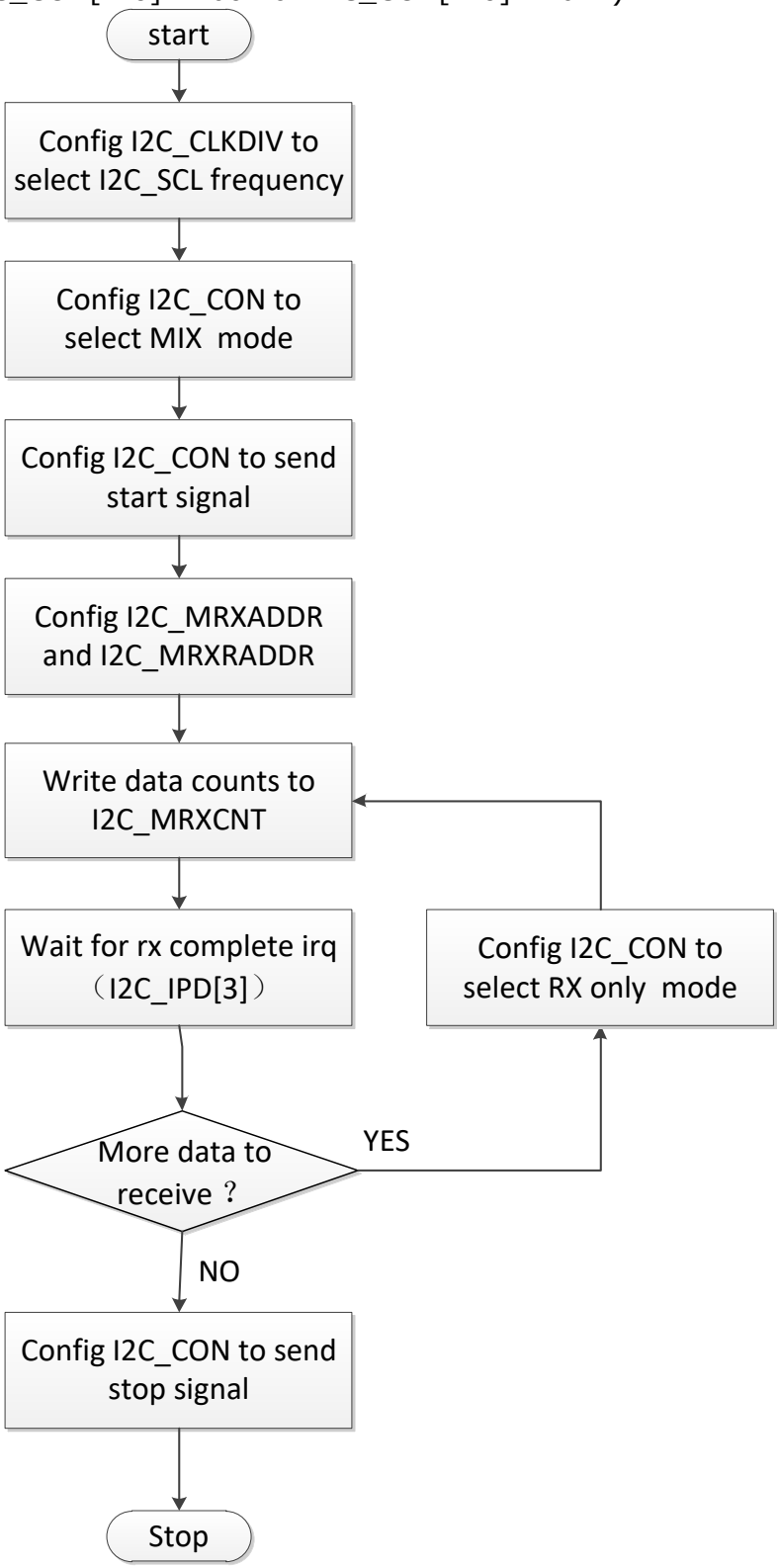


Fig. 24-8 I2C Flow chat for mix mode

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 25 GPIO

25.1 Overview

GPIO is a programmable General Purpose Programming I/O peripheral. This component is an APB slave device. GPIO controls the output data and direction of external I/O pads. It also can read back the data on external pads using memory-mapped registers.
GPIO supports the following features:

- 32 bits APB bus width
- 32 independently configurable signals
- Separate data registers and data direction registers for each signal
- Software control for each signal, or for each bit of each signal
- Configurable interrupt mode

25.2 Block Diagram

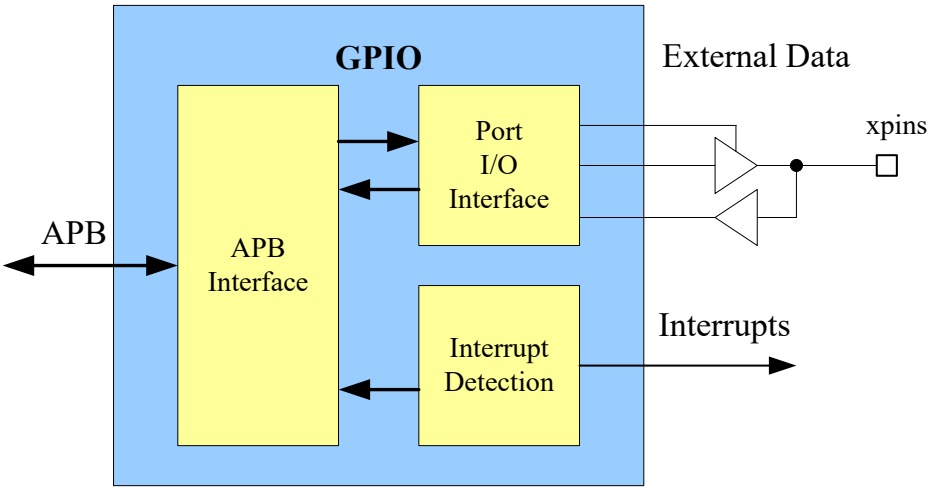


Fig. 25-1 GPIO block diagram

Block descriptions:

APB Interface

The APB Interface implements the APB slave operation. Its data bus width is 32 bits.

Port I/O Interface

External data Interface to or from I/O pads.

Interrupt Detection

Interrupt interface to or from interrupt controller.

25.3 Function Description

25.3.1 Operation

Control Mode (software)

Under software control, the data and direction control for the signal are sourced from the data register (GPIO_SWPORTA_DR) and direction control register (GPIO_SWPORTA_DDR). The direction of the external I/O pad is controlled by a write to the Porta data direction register (GPIO_SWPORTA_DDR). The data written to this memory-mapped register gets mapped onto an output signal, GPIO_PORTA_DDR, of the GPIO peripheral. This output signal controls the direction of an external I/O pad.
The data written to the Porta data register (GPIO_SWPORTA_DR) drives the output buffer of the I/O pad. External data are input on the external data signal, GPIO_EXT_PORTA. Reading the external signal register(GPIO_EXT_PORTA) shows the value on the signal, regardless of the direction. This register is read-only, meaning that it cannot be written from the APB software interface.

Reading External Signals

The data on the GPIO_EXT_PORTA external signal can always be read. The data on the external GPIO signal is read by an APB read of the memory-mapped register,

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

GPIO_EXT_PORTA.
An APB read to the GPIO_EXT_PORTA register yields a value equal to that which is on the GPIO_EXT_PORTA signal.

Interrupts

Port A can be programmed to accept external signals as interrupt sources on any of the bits of the signal. The type of interrupt is programmable with one of the following settings:

- Active-high and level
- Active-low and level
- Rising edge
- Falling edge
- Both the rising edge and the falling edge

The interrupts can be masked by programming the GPIO_INTMASK register. The interrupt status can be read before masking (called raw status) and after masking.
The interrupts are combined into a single interrupt output signal, which has the same polarity as the individual interrupts. In order to mask the combined interrupt, all individual interrupts have to be masked. The single combined interrupt does not have its own mask bit.
Whenever Port A is configured for interrupts, the data direction must be set to Input. If the data direction register is reprogrammed to Output, then any pending interrupts are not lost. However, no new interrupts are generated.
For edge-detected interrupts, the ISR can clear the interrupt by writing a 1 to the GPIO_PORTA_EOI register for the corresponding bit to disable the interrupt. This write also clears the interrupt status and raw status registers. Writing to the GPIO_PORTA_EOI register has no effect on level-sensitive interrupts. If level-sensitive interrupts cause the processor to interrupt, then the ISR can poll the GPIO_INT_RAWSTATUS register until the interrupt source disappears, or it can write to the GPIO_INTMASK register to mask the interrupt before exiting the ISR. If the ISR exits without masking or disabling the interrupt prior to exiting, then the level-sensitive interrupt repeatedly requests an interrupt until the interrupt is cleared at the source.

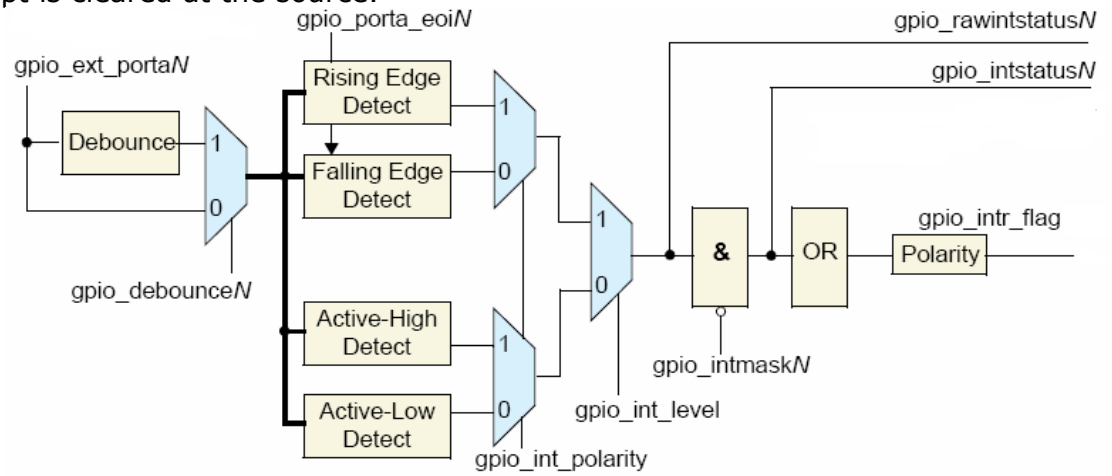


Fig. 25-2 GPIO Interrupt RTL Block Diagram

Debounce operation

Port A has been configured to include the debounce capability interrupt feature. The external signal can be debounced to remove any spurious glitches that are less than one period of the external debouncing clock.
When input interrupt signals are debounced using a debounce clock (pclk), the signals must be active for a minimum of two cycles of the debounce clock to guarantee that they are registered. Any input pulse widths less than a debounce clock period are bounced. A pulse width between one and two debounce clock widths may or may not propagate, depending on its phase relationship to the debounce clock. If the input pulse spans two rising edges of the debounce clock, it is registered. If it spans only one rising edge, it is not registered.

Synchronization of Interrupt Signals to the System Clock

Interrupt signals are internally synchronized to pclk. Synchronization to pclk must occur for edge-detect signals. With level-sensitive interrupts, synchronization is optional and under

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

software control (GPIO_LS_SYNC).

25.3.2 Programming

Programming Considerations

- Reading from an unused location or unused bits in a particular register always returns zeros. There is no error mechanism in the APB.
- Programming the GPIO registers for interrupt capability, edge-sensitive or level-sensitive interrupts, and interrupt polarity should be completed prior to enabling the interrupts on Port A in order to prevent spurious glitches on the interrupt lines to the interrupt controller.
- Writing to the interrupt clear register clears an edge-detected interrupt and has no effect on a level-sensitive interrupt.

GPIOs' hierarchy in the chip

GPIO0 is in PD_PMU subsystem, GPIO1/GPIO2/GPIO3/GPIO4 are in PD_BUS subsystem.

25.4 Register Description

This section describes the control/status registers of the design. Software should read and write these registers using 32-bits accesses. There are 4 GPIOs (GPIO0 ~ GPIO4), and each of them has same register group. Therefore, 5 GPIOs' register groups have 5 different base addresses.

25.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
GPIO_SWPORTA_DR	0x0000	W	0x00000000	Port A data register
GPIO_SWPORTA_DDR	0x0004	W	0x00000000	Port A data direction register
GPIO_INTEN	0x0030	W	0x00000000	Interrupt enable register
GPIO_INTMASK	0x0034	W	0x00000000	Interrupt mask register
GPIO_INTTYPE_LEVEL	0x0038	W	0x00000000	Interrupt level register
GPIO_INT_POLARITY	0x003c	W	0x00000000	Interrupt polarity register
GPIO_INT_STATUS	0x0040	W	0x00000000	Interrupt status of port A
GPIO_INT_RAWSTATUS	0x0044	W	0x00000000	Raw Interrupt status of port A
GPIO_DEBOUNCE	0x0048	W	0x00000000	Debounce enable register
GPIO_PORTA_EOI	0x004c	W	0x00000000	Port A clear interrupt register
GPIO_EXT_PORTA	0x0050	W	0x00000000	Port A external port register
GPIO_LS_SYNC	0x0060	W	0x00000000	Level_sensitive synchronization enable register
GPIO_INT_BOTHEDGE	0x0068	W	0x00000000	Interrupt both edge type

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

25.4.2 Detail Register Description

GPIO_SWPORTA_DR

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	swporta_dr Values written to this register are output on the I/O signals for Port A if the corresponding data direction bits for Port A are set to Output mode. The value read back is equal to the last value written to this register.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

GPIO_SWPORTA_DDR

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	swporta_ddr Values written to this register independently control the direction of the corresponding data bit in Port A. 1'b0: Input (default) 1'b1: Output

GPIO_INTEN

Address: Operational Base + offset (0x0030)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	int_en Allows each bit of Port A to be configured for interrupts. Whenever a 1'b1 is written to a bit of this register, it configures the corresponding bit on Port A to become an interrupt; otherwise, Port A operates as a normal GPIO signal. Interrupts are disabled on the corresponding bits of Port A if the corresponding data direction register is set to Output. 1'b0: Configure Port A bit as normal GPIO signal (default) 1'b1: Configure Port A bit as interrupt

GPIO_INTMASK

Address: Operational Base + offset (0x0034)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	Reserved
0	RW	0x0	int_mask Controls whether an interrupt on Port A can create an interrupt for the interrupt controller by not masking it. Whenever a 1'b1 is written to a bit in this register, it masks the interrupt generation capability for this signal; otherwise interrupts are allowed through. 1'b0: Interrupt bits are unmasked (default) 1'b1: Mask interrupt

GPIO_INTTYPE_LEVEL

Address: Operational Base + offset (0x0038)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	Reserved
0	RW	0x0	inttype_level Controls the type of interrupt that can occur on Port A. 1'b0: Level-sensitive (default) 1'b1: Edge-sensitive

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

GPIO_INT_POLARITY
Address: Operational Base + offset (0x003c)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	int_polarity Controls the polarity of edge or level sensitivity that can occur on input of Port A. 1'b0: Active-low (default) 1'b1: Active-high

GPIO_INT_STATUS
Address: Operational Base + offset (0x0040)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	int_status Interrupt status of Port A

GPIO_INT_RAWSTATUS
Address: Operational Base + offset (0x0044)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	int_rawstatus Raw interrupt of status of Port A (premasking bits)

GPIO_DEBOUNCE
Address: Operational Base + offset (0x0048)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	debounce Controls whether an external signal that is the source of an interrupt needs to be debounced to remove any spurious glitches. Writing a 1'b1 to a bit in this register enables the debouncing circuitry. A signal must be valid for two periods of an external clock before it is internally processed. 1'b0: No debounce (default) 1'b1: Enable debounce

GPIO_PORTA_EOI
Address: Operational Base + offset (0x004c)

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	porta_eoi Controls the clearing of edge type interrupts from Port A. When a 1'b1 is written into a corresponding bit of this register, the interrupt is cleared. All interrupts are cleared when Port A is not configured for interrupts. 1'b0: No interrupt clear (default) 1'b1: Clear interrupt

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

GPIO_EXT_PORTA

Address: Operational Base + offset (0x0050)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	ext_porta When Port A is configured as Input, then reading this location reads the values on the signal. When the data direction of Port A is set as Output, reading this location reads the data register for Port A.

GPIO_LS_SYNC

Address: Operational Base + offset (0x0060)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	ls_sync Writing a 1'b1 to this register results in all level-sensitive interrupts being synchronized to pclk_intr. 1'b0: No synchronization to pclk_intr (default) 1'b1: Synchronize to pclk_intr

GPIO_INT_BOTHEDGE

Address: Operational Base + offset (0x0068)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	interrupt_both_edge_type Controls the edge type of interrupt that can occur on Port A. Whenever a particular bit is programmed to 1'b1, it enables the generation of interrupts on both the rising edge and the falling edge of an external input signal corresponding to that bit on port A.The values programmed in the registers gpio_intype_level and gpio_int_polarity for this particular bit are not considered when the corresponding bit of this register is set to 1'b1. Whenever a particular bit is programmed to 0, the interrupt type depends on the value of the corresponding bits in the gpio_inttype_level and gpio_int_polarity registers.

25.5 Interface Description

Table 25-1 GPIO interface description

Module Pin	Dir	Pad Name	IOMUX Setting
GPIO0 Interface			
gpio0_porta[7:0]	I/O	GPIO0_A[7:0]	PMUGRF_GPIO0A_IOMUX[15:0]=16'h0
gpio0_porta[15:8]	I/O	GPIO0_B[7:0]	PMUGRF_GPIO0B_IOMUX[15:0]=16'h0
gpio0_porta[23:16]	I/O	GPIO0_C[7:0]	PMUGRF_GPIO0C_IOMUX[15:0]=16'h0
GPIO1 Interface			
gpio1_porta[7:0]	I/O	GPIO1_A[3:0]	GRF_GPIO1A_IOMUX_L[15:0]=16'h0

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Module Pin	Dir	Pad Name	IOMUX Setting
		GPIO1_A[7:4]	GRF_GPIO1A_IOMUX_H[15:0]=16'h0
gpio1_porta[15:8]	I/O	GPIO1_B[3:0] GPIO1_B[7:4]	GRF_GPIO1B_IOMUX_L[15:0]=16'h0 GRF_GPIO1B_IOMUX_H[15:0]=16'h0
gpio1_porta[23:16]	I/O	GPIO1_C[3:0] GPIO1_C[7:4]	GRF_GPIO1C_IOMUX_L[15:0]=16'h0 GRF_GPIO1C_IOMUX_H[15:0]=16'h0
gpio1_porta[31:24]	I/O	GPIO1_D[3:0] GPIO1_D[7:4]	GRF_GPIO1D_IOMUX_L[15:0]=16'h0 GRF_GPIO1D_IOMUX_H[15:0]=16'h0
GPIO2 Interface			
gpio2_porta[7:0]	I/O	GPIO2_A[3:0] GPIO2_A[7:4]	GRF_GPIO2A_IOMUX_L[15:0]=16'h0 GRF_GPIO2A_IOMUX_H[15:0]=16'h
gpio2_porta[15:8]	I/O	GPIO2_B[3:0] GPIO2_B[7:4]	GRF_GPIO2B_IOMUX_L[15:0]=16'h0 GRF_GPIO2B_IOMUX_H[15:0]=16'h0
gpio2_porta[23:16]	I/O	GPIO2_C[3:0] GPIO2_C[7:4]	GRF_GPIO2C_IOMUX_L[15:0]=16'h0 GRF_GPIO2C_IOMUX_H[15:0]=16'h0
gpio2_porta[31:24]	I/O	GPIO2_D[3:0] GPIO2_D[7:4]	GRF_GPIO2D_IOMUX_L[15:0]=16'h0 GRF_GPIO2D_IOMUX_H[15:0]=16'h0
GPIO3 Interface			
gpio3_porta[7:0]	I/O	GPIO3_A[3:0] GPIO3_A[7:4]	GRF_GPIO3A_IOMUX_L[15:0]=16'h0 GRF_GPIO3A_IOMUX_H[15:0]=16'h0
gpio3_porta[15:8]	I/O	GPIO3_B[3:0] GPIO3_B[7:4]	GRF_GPIO3B_IOMUX_L[15:0]=16'h0 GRF_GPIO3B_IOMUX_H[15:0]=16'h0
gpio3_porta[23:16]	I/O	GPIO3_C[3:0] GPIO3_C[7:4]	GRF_GPIO3C_IOMUX_L[15:0]=16'h0 GRF_GPIO3C_IOMUX_H[15:0]=16'h0
gpio3_porta[31:24]	I/O	GPIO3_D[3:0] GPIO3_D[7:4]	GRF_GPIO3D_IOMUX_L[15:0]=16'h0 GRF_GPIO3D_IOMUX_H[15:0]=16'h0
GPIO4 Interface			
gpio4_porta[7:0]	I/O	GPIO4_A[3:0] GPIO4_A[7:4]	GRF_GPIO4A_IOMUX_L[15:0]=16'h0 GRF_GPIO4A_IOMUX_H[15:0]=16'h0
gpio4_porta[15:8]	I/O	GPIO4_B[3:0] GPIO4_B[7:4]	GRF_GPIO4B_IOMUX_L[15:0]=16'h0 GRF_GPIO4B_IOMUX_H[15:0]=16'h0
gpio4_porta[23:16]	I/O	GPIO4_C[3:0] GPIO4_C[7:4]	GRF_GPIO4C_IOMUX_L[15:0]=16'h0 GRF_GPIO4C_IOMUX_H[15:0]=16'h0
gpio4_porta[31:24]	I/O	GPIO4_D[3:0] GPIO4_D[7:4]	GRF_GPIO4D_IOMUX_L[15:0]=16'h0 GRF_GPIO4D_IOMUX_H[15:0]=16'h0

25.6 Application Notes

Steps to set GPIO's direction

- Write GPIO_SWPORT_DDR[x] as 1 to set this gpio as output direction and Write GPIO_SWPORT_DDR[x] as 0 to set this gpio as input direction.
- Default GPIO's direction is input direction.

Steps to set GPIO's level

- Write GPIO_SWPORT_DDR[x] as 1 to set this gpio as output direction.
- Write GPIO_SWPORT_DR[x] as v to set this GPIO's value.

Steps to get GPIO's level

- Write GPIO_SWPORT_DDR[x] as 0 to set this gpio as input direction.
- Read from GPIO_EXT_PORT[x] to get GPIO's value

Steps to set GPIO as interrupt source

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

- Write GPIO_SWPORT_DDR[x] as 0 to set this gpio as input direction.
- Write GPIO_INTTYPE_LEVEL[x] as v1 and write GPIO_INT_POLARITY[x] as v2 to set interrupt type
- Write GPIO_INTEN[x] as 1 to enable GPIO's interrupt

Note: Please switch iomux to GPIO mode first!

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 26 Timer

26.1 Overview

Timer is a programmable timer peripheral. This component is an APB slave device. In *RK1808* there are 6 timers(Timer0~5) and 2 Secure timers(STimer0~2). CNTPCT_EL0 is provided by STimer1.

Timer5 and STimer0~1 count up from zero to a programmed value and generate an interrupt when the counter reaches the programmed value.

Timer0~4 count down from a programmed value to zero and generate an interrupt when the counter reaches zero.

Timer supports the following features:

- Timer0~5 is used for no-secure, STimer0~1 is used for secure.
- Two operation modes: free-running and user-defined count.
- Maskable for each individual interrupt.

26.2 Block Diagram

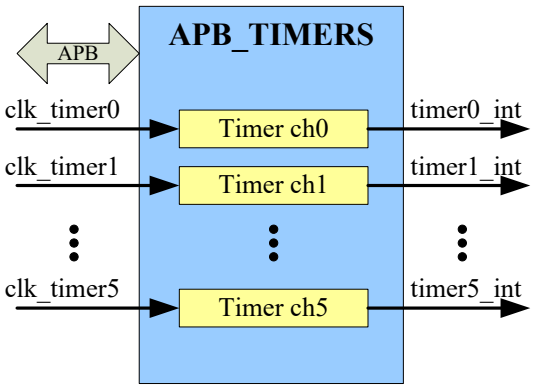


Fig. 26-1 Timer Block Diagram

The above figure shows the architecture of the APB timers (include six programmable timer channels). The Stimers that in the bus subsystem only include two programmable timer channels.

26.3 Function Description

26.3.1 Timer clock

TIMER0~ TIMER5 and STIMER0~1 are in the pd_bus subsystem. The clock source is 24MHz.

26.3.2 Programming sequence

1. Initialize the timer by the TIMERn_CONTROLREG (0≤n≤5) register:

- Disable the timer by writing a "0" to the timer enable bit (bit 0). Accordingly, the timer_en output signal is de-asserted.
- Program the timer mode—user-defined or free-running—by writing a "0" or "1" respectively, to the timer mode bit (bit 1).
- Set the interrupt mask as either masked or not masked by writing a "0" or "1" respectively, to the timer interrupt mask bit (bit 2).

2. Load the timer count value into the TIMERn_LOAD_COUNT1 (0≤n≤5) and TIMERn_ LOAD_COUNT0 (0≤n≤5) register.

3. Enable the timer by writing a "1" to bit 0 of TIMERn_CONTROLREG (0≤n≤5).

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

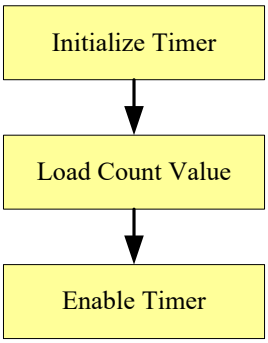


Fig. 26-2 Timer Usage Flow

26.3.3 Loading a timer count value

For the descending Timers(Timer0~4).The initial value for each timer—that is, the value from which it counts down—is loaded into the timer using the load count register (TIMERn_LOAD_COUNT1 and TIMERn_ LOAD_COUNT0). Two events can cause a timer to load the initial value from its load count register:

- Timer is enabled after reset or disabled.
- Timer counts down to 0, when timer is configured into free-running mode.

For the incremental Timers(Timer5 and STimer0~1).The initial value for each timer is zero. The count register will count up to the value loaded in the register TIMERn_LOAD_COUNT1 and TIMERn_ LOAD_COUNT0. Two events can cause a timer to load zero:

- Timer is enabled after reset or disabled.
- Timer counts up to the value stored in TIMERn_LOAD_COUNT1 and TIMERn_ LOAD_COUNT0, when timer is configured into free-running mode.

26.3.4 Timer mode selection

- User-defined count mode – Timer loads TIMERn_LOAD_COUNT1 and TIMERn_ LOAD_ COUNT0 registers (for descending timers) or zero (for incremental timers) as initial value. When the timer counts down to 0 (for descending timers) or counts up to the value in TIMERn_LOAD_COUNT1 and TIMERn_ LOAD_ COUNT0 (for incremental timers),it will not automatically reload the count register. User need to disable timer firstly and follow the programming sequence to make timer work again.
- Free-running mode – Timer loads the TIMERn_LOAD_COUNT1 and TIMERn_LOAD_COUNT0(for descending timers) or zero (for incremental timers)register as initial value. Timer will automatically reload the count register, when timer counts down to 0 (for descending timers) or counts up to the value in TIMERn_LOAD_COUNT1 and TIMERn_LOAD_COUNT0 (for incremental timers).

26.4 Register Description

26.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
TIMER_TIMERn_LOAD_COUNT0	0x0000	W	0x00000000	Timern Load Count Register 0
TIMER_TIMERn_LOAD_COUNT1	0x0004	W	0x00000000	Timern Load Count Register 1.Higher 32 bits Value to be loaded into Timer n. This is the value from which counting commences
TIMER_TIMERn_CURRENT_VALUE0	0x0008	W	0x00000000	Timern Current Value Register 0

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
TIMER_TIMERn_CURRENT_VALUE1	0x000c	W	0x00000000	Timern Current Value Register 1.High 32 bits of Current Value of Timer n
TIMER_TIMERn_CONTROLREG	0x0010	W	0x00000000	Timern Control Register
TIMER_TIMERn_INTSTATUS	0x0018	W	0x00000000	Timern Interrupt Status Register

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

26.4.2 Detail Register Description

TIMER_TIMERn_LOAD_COUNT0

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	load_count_0 Lower 32 bits Value to be loaded into Timer n. This is the value from which counting commences.

TIMER_TIMERn_LOAD_COUNT1

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	load_count_1 Higher 32 bits Value to be loaded into Timer n. This is the value from which counting commences.

TIMER_TIMERn_CURRENT_VALUE0

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	timern_current_value0 Lower 32 bits of Current Value of Timer n

TIMER_TIMERn_CURRENT_VALUE1

Address: Operational Base + offset (0x000c)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	timern_current_value1 Higher 32 bits of Current Value of Timer n

TIMER_TIMERn_CONTROLREG

Address: Operational Base + offset (0x0010)

Bit	Attr	Reset Value	Description
31:3	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2	RW	0x0	timer_int_mask Timer interrupt mask. 1'b0: Mask 1'b1: Not mask
1	RW	0x0	timer_mode Timer mode. 1'b0: Free-running mode 1'b1: User-defined count mode
0	RW	0x0	timer_en Timer enable. 1'b0: Disable 1'b1: Enable

TIMER_TIMERn_INTSTATUS

Address: Operational Base + offset (0x0018)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	Reserved
0	RO	0x0	timern_int This register contains the interrupt status for timern.

26.5 Application Notes

26.6 Register Base Address

Table 26-1 Register Base Address

Module	CNT	Base Addr
TIMER	TIMER0_BASE	0xFF700000
	TIMER1_BASE	0xFF700020
	TIMER2_BASE	0xFF700040
	TIMER3_BASE	0xFF700060
	TIMER4_BASE	0xFF700080
	TIMER5_BASE	0xFF7000a0
STIMER	STIMER0_BASE	0xFF710000
	STIMER1_BASE	0xFF710020

26.7 Clock and Enable

In the chip, the timer_clk is from 24MHz XIN_OSC, asynchronous to the pclk. When user disables the timer enables bit (bit 0 of TIMERn_CONTROLREG (0≤n≤5)), the timer en output signal is de-asserted, and timer_clk will stop. When user enables the timer, the timer_en signal is asserted and timer_clk will start running.
The application is only allowed to re-config registers when timer_en is low.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

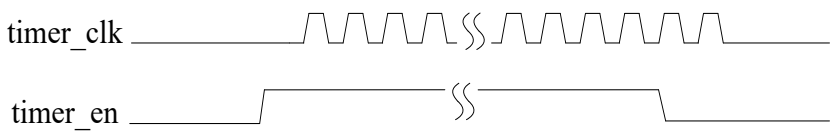


Fig. 26-3 Timing between timer_en and timer_clk

Please refer to function description section for the timer usage flow.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 27 Pulse Width Modulation (PWM)

27.1 Overview

The pulse-width modulator (PWM) feature is very common in embedded systems. It provides a way to generate a pulse periodic waveform for motor control or can act as a digital-to-analog converter with some external components.
The PWM Module supports the following features:

- 4-built-in PWM channels
- Support capture mode
  - Measures the high/low polarity effective cycles of this input waveform
  - Generates a single interrupt at the transition of input waveform polarity
  - 32-bit high polarity capture register
  - 32-bit low polarity capture register
  - 32-bit current value register
  - The capture result can be stored in a FIFO, and the depth of FIFO is 8. The data of FIFO can be read by CPU or DMA
  - Channel 3 support 32-bits power key capture mode
  - Support a input filter to remove glitch
- Support continuous mode or one-shot mode
  - 32-bit period counter
  - 32-bit duty register
  - 32-bit current value register
  - PWM output polarity in inactive state and duty cycle polarity can be configured
  - Period and duty cycle are shadow buffered. Change takes effect when the end of the effective period is reached or when the channel is disabled
  - Programmable center or left aligned outputs, and change takes effect when the end of the effective period is reached or when the channel is disabled
  - 8-bit repeat counter for one-shot operation. One-shot operation will produce N + 1 periods of the waveform, where N is the repeat counter value, and generates a single interrupt at the end of operation
  - Continuous mode generates the waveform continuously, and does not generates any interrupts
- pre-scaled operation to clk_pwm and then further scaled
- Available low-power mode to reduce power consumption when the channel is inactive.

27.2 Block Diagram

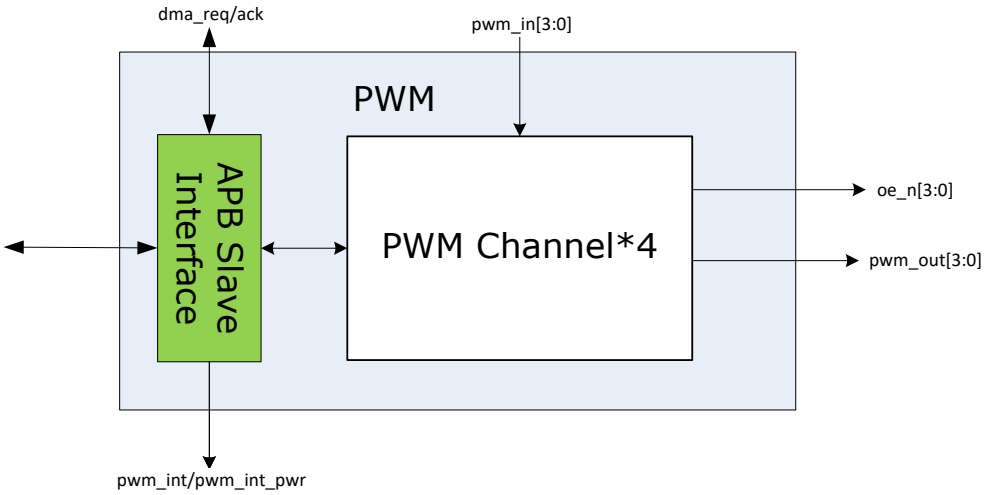


Fig. 27-1 PWM Block Diagram

The host processor gets access to PWM Register Block through the APB slave interface with 32-bit bus width, and asserts the active-high level interrupt. PWM only supports one interrupt output, please refer to interrupt register to know the raw interrupt status when an

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

interrupt is asserted.
PWM Channel is the control logic of PWM module, and controls the operation of PWM module according to the configured working mode.

27.3 Function Description

The PWM supports three operation modes: capture mode, one-shot mode and continuous mode. For the one-shot mode and the continuous mode, the PWM output can be configured as the left-aligned mode or the center-aligned mode.

27.3.1 Capture mode

The capture mode is used to measure the PWM channel input waveform high/low effective cycles with the PWM channel clock, and asserts an interrupt when the polarity of the input waveform changes. The number of the high effective cycles is recorded in the PWMx_PERIOD_HPC register, while the number of the low effective cycles is recorded in the PWMx_DUTY_LPC register.

Notes: the PWM input waveform is doubled buffered when the PWM channel is working in order to filter unexpected shot-time polarity transition, and therefore the interrupt is asserted several cycles after the input waveform polarity changes, and so does the change of the values of PWMx_PERIOD_HPC and PWMx_DUTY_LPC.

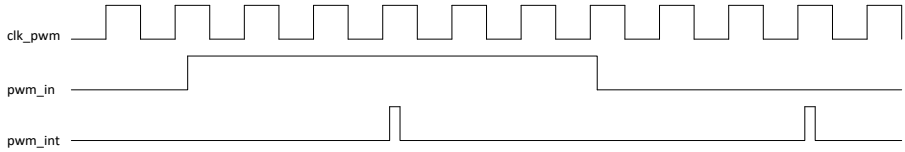


Fig. 27-2 PWM Capture Mode

The capture result also can be stored in a FIFO. The FIFO has an almost full indicator. The indicator can chose to use as an interrupt or DMA request. When it is used as an interrupt, the data in FIFO can be read by CPU. When it is used as a DMA request, the data in FIFO can be read through DMA. It also supports timeout interrupt when the data in FIFO has not been read in a time threshold.

The PWM (only channel 3) support 32-bits power key capture mode. User can configure 10 power key to match, the interrupt will be asserted when the capture value match any one.

27.3.2 Continuous mode

The PWM channel generates a series of the pulses continuously as expected once the channel is enabled with continuous mode.

In the continuous mode, the PWM output waveforms can be in one form of the two output mode: left-aligned mode or center-aligned mode.

For the left-aligned output mode, the PWM channel firstly starts the duty cycle with the configured duty polarity (PWMx_CTRL.duty_pol). Once duty cycle number (PWMx_DUTY_LPC) is reached, the output is switched to the opposite polarity. After the period number (PWMx_PERIOD_HPC) is reached, the output is again switched to the opposite polarity to start another period of desired pulse.

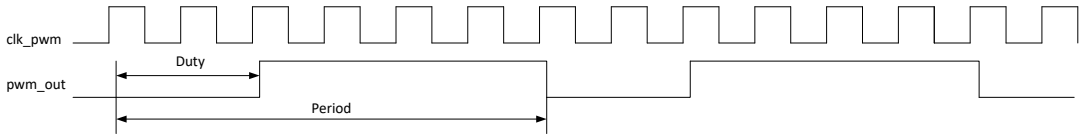
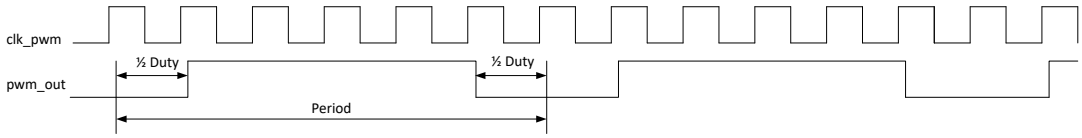


Fig. 27-3 PWM Continuous Left-aligned Output Mode

For the center-aligned output mode, the PWM channel firstly starts the duty cycle with the configured duty polarity (PWMx_CTRL.duty_pol). Once one half of duty cycle number (PWMx_DUTY_LPC) is reached, the output is switched to the opposite polarity. Then if there is one half of duty cycle left for the whole period, the output is again switched to the opposite polarity. Finally after the period number (PWMx_PERIOD_HPC) is reached, the output starts another period of desired pulse.



Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Fig. 27-4 PWM Continuous Center-aligned Output Mode

Once disable the PWM channel, the channel stops generating the output waveforms and output polarity is fixed as the configured inactive polarity (PWMx_CTRL.inactive_pol).

27.3.3 One-shot mode

Unlike the continuous mode, the PWM channel generates the output waveforms within the configured periods (PWM_CTRL.rpt + 1), and then stops. At the same times, an interrupt is asserted to inform that the operation has been finished.

There are also two output modes for the one-shot mode: the left-aligned mode and the center-aligned mode.

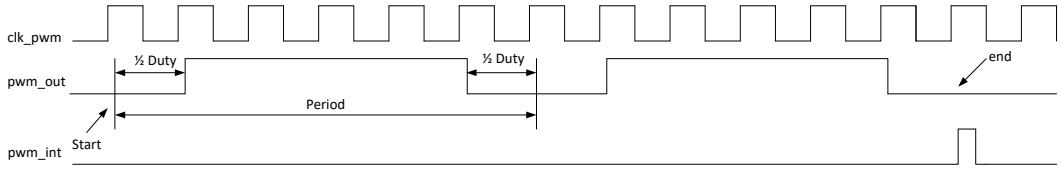


Fig. 27-5 PWM One-shot Center-aligned Output Mode

27.4 Register Description

27.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
PWM_PWM0_CNT	0x0000	W	0x00000000	PWM Channel 0 Counter Register
PWM_PWM0_PERIOD_HPR	0x0004	W	0x00000000	PWM Channel 0 Period Register/High Polarity Capture Register
PWM_PWM0_DUTY_LPR	0x0008	W	0x00000000	PWM Channel 0 Duty Register/Low Polarity Capture Register
PWM_PWM0_CTRL	0x000c	W	0x00000000	PWM Channel 0 Control Register
PWM_PWM1_CNT	0x0010	W	0x00000000	PWM Channel 1 Counter Register
PWM_PWM1_PERIOD_HPR	0x0014	W	0x00000000	PWM Channel 1 Period Register/High Polarity Capture Register
PWM_PWM1_DUTY_LPR	0x0018	W	0x00000000	PWM Channel 1 Duty Register/Low Polarity Capture Register
PWM_PWM1_CTRL	0x001c	W	0x00000000	PWM Channel 1 Control Register
PWM_PWM2_CNT	0x0020	W	0x00000000	PWM Channel 2 Counter Register
PWM_PWM2_PERIOD_HPR	0x0024	W	0x00000000	PWM Channel 2 Period Register/High Polarity Capture Register
PWM_PWM2_DUTY_LPR	0x0028	W	0x00000000	PWM Channel 2 Duty Register/Low Polarity Capture Register
PWM_PWM2_CTRL	0x002c	W	0x00000000	PWM Channel 2 Control Register
PWM_PWM3_CNT	0x0030	W	0x00000000	PWM Channel 3 Counter Register
PWM_PWM3_PERIOD_HPR	0x0034	W	0x00000000	PWM Channel 3 Period Register/High Polarity Capture Register

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
PWM_PWM3_DUTY_LPR	0x0038	W	0x00000000	PWM Channel 3 Duty Register/Low Polarity Capture Register
PWM_PWM3_CTRL	0x003c	W	0x00000000	PWM Channel 3 Control Register
PWM_INTSTS	0x0040	W	0x00000000	Interrupt Status Register
PWM_INT_EN	0x0044	W	0x00000000	Interrupt Enable Register
PWM_FIFO_CTRL	0x0050	W	0x00000000	PWM Channel 3 FIFO Mode Control Register
PWM_FIFO_INTSTS	0x0054	W	0x00000010	FIFO Interrupts Status Register
PWM_FIFO_TOUTTHR	0x0058	W	0x00000000	FIFO Timeout Threshold Register
PWM_VERSION_ID	0x005c	W	0x02120b34	PWM Version ID Register
PWM_FIFO	0x0060	W	0x00000000	FIFO Register
PWM_PWRMATCH_CTRL	0x0080	W	0x00000000	Power Key Match Control Register
PWM_PWRMATCH_LPRE	0x0084	W	0x238c22c4	Power Key Match Of Low Preload Register
PWM_PWRMATCH_HPRE	0x0088	W	0x11f81130	Power Key Match Of High Preload Register
PWM_PWRMATCH_LD	0x008c	W	0x029401cc	Power Key Match Of Low Data Register
PWM_PWRMATCH_HD_ZERO	0x0090	W	0x029401cc	Power Key Match Of High Data For Zero Register
PWM_PWRMATCH_HD_ONE	0x0094	W	0x06fe0636	Power Key Match Of High Data For One Register
PWM_PWRMATCH_VALUE0	0x0098	W	0x00000000	Power Key Match Value 0 Register
PWM_PWRMATCH_VALUE1	0x009c	W	0x00000000	Power Key Match Value 1 Register
PWM_PWRMATCH_VALUE2	0x00a0	W	0x00000000	Power Key Match Value 2 Register
PWM_PWRMATCH_VALUE3	0x00a4	W	0x00000000	Power Key Match Value 3 Register
PWM_PWRMATCH_VALUE4	0x00a8	W	0x00000000	Power Key Match Value 4 Register
PWM_PWRMATCH_VALUE5	0x00ac	W	0x00000000	Power Key Match Value 5 Register
PWM_PWRMATCH_VALUE6	0x00b0	W	0x00000000	Power Key Match Value 6 Register
PWM_PWRMATCH_VALUE7	0x00b4	W	0x00000000	Power Key Match Value 7 Register
PWM_PWRMATCH_VALUE8	0x00b8	W	0x00000000	Power Key Match Value 8 Register
PWM_PWRMATCH_VALUE9	0x00bc	W	0x00000000	Power Key Match Value 9 Register

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
PWM_PWM3_PWRCAPTURE_VALUE	0x00cc	W	0x00000000	Channel 3 Power Key Capture Value Register
PWM_FILTER_CTRL	0x00d0	W	0x00000000	Filter Control Register

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

27.4.2 Detail Register Description

PWM_PWM0_CNT

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	CNT The 32-bit indicates current value of PWM Channel 0 counter. The counter runs at the rate of PWM clock. The value ranges from 0 to (2^32-1)

PWM_PWM0_PERIOD_HPR

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	PERIOD_HPR If PWM is operated at the continuous mode or one-shot mode, this value defines the period of the output waveform. Note that, if the PWM is operated at the center-aligned mode, the period should be an even one, and therefore only the bit [31:1] is taken into account and bit [0] always considered as 0. If PWM is operated at the capture mode, this value indicates the effective high polarity cycles of input waveform. This value is based on the PWM clock. The value ranges from 0 to (2^32-1)

PWM_PWM0_DUTY_LPR

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	DUTY_LPR If PWM is operated at the continuous mode or one-shot mode, this value defines the duty cycle of the output waveform. The PWM starts the output waveform with duty cycle. Note that, if the PWM is operated at the center-aligned mode, the period should be an even one, and therefore only the [31:1] is taken into account. If PWM is operated at the capture mode, this value indicates the effective low polarity cycles of input waveform. This value is based on the PWM clock. The value ranges from 0 to (2^32-1)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

PWM_PWM0_CTRL

Address: Operational Base + offset (0x000c)

Bit	Attr	Reset Value	Description
31:24	RW	0x00	rpt This field defines the repeated effective periods of output waveform in one-shot mode. The value N means N+1 repeated effective periods
23:16	RW	0x00	scale This field defines the scale factor applied to prescaled clock. The value N means the clock is divided by 2*N. If N is 0, it means that the clock is divided by 512(2*256)
15	RO	0x0	Reserved
14:12	RW	0x0	prescale This field defines the prescale factor applied to input clock. The value N means that the input clock is divided by 2^N
11:10	RO	0x0	Reserved
9	RW	0x0	clk_sel 1'b0: non-scaled clock is selected as PWM clock source. It means that the prescale clock is directly used as the PWM clock source 1'b1: scaled clock is selected as PWM clock source
8	RW	0x0	force_clk_en 1'b0: disabled, when PWM channel is inactive state, the clk_pwm to PWM Clock prescale module is blocked to reduce power consumption. 1'b1: enabled, the clk_pwm to PWM Clock prescale module is always enable
7	RW	0x0	ch_cnt_en 1'b0: disabled 1'b1: enabled
6	RW	0x0	conlock pwm period and duty lock to previous configuration 1'b0: disable lock 1'b1: enable lock
5	RW	0x0	output_mode 1'b0: left aligned mode 1'b1: center aligned mode
4	RW	0x0	inactive_pol This defines the output waveform polarity when PWM channel is in inactive state. The inactive state means that PWM finishes the complete waveform in one-shot mode or PWM channel is disabled. 1'b0: negative 1'b1: positive

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
3	RW	0x0	duty_pol This defines the polarity for duty cycle. PWM starts the output waveform with duty cycle. 1'b0: negative 1'b1: positive
2:1	RW	0x0	pwm_mode 2'b00: One shot mode. PWM produces the waveform within the repeated times defined by PWMx_CTRL_rpt . 2'b01: Continuous mode. PWM produces the waveform continuously 2'b10: Capture mode. PWM measures the cycles of high/low polarity of input waveform. 2'b11: Reserved
0	RW	0x0	pwm_en 1'b0: disabled 1'b1: enabled. If the PWM is worked in the one-shot mode, this bit will be cleared at the end of operation

PWM_PWM1_CNT

Address: Operational Base + offset (0x0010)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	CNT The 32-bit indicates current value of PWM Channel 1 counter. The counter runs at the rate of PWM clock. The value ranges from 0 to (2^32-1)

PWM_PWM1_PERIOD_HPR

Address: Operational Base + offset (0x0014)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	PERIOD_HPR If PWM is operated at the continuous mode or one-shot mode, this value defines the period of the output waveform. Note that, if the PWM is operated at the center-aligned mode, the period should be an even one, and therefore only the bit [31:1] is taken into account and bit [0] always considered as 0. If PWM is operated at the capture mode, this value indicates the effective high polarity cycles of input waveform. This value is based on the PWM clock. The value ranges from 0 to (2^32-1)

PWM_PWM1_DUTY_LPR

Address: Operational Base + offset (0x0018)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	DUTY_LPR If PWM is operated at the continuous mode or one-shot mode, this value defines the duty cycle of the output waveform. The PWM starts the output waveform with duty cycle. Note that, if the PWM is operated at the center-aligned mode, the period should be an even one, and therefore only the [31:1] is taken into account. If PWM is operated at the capture mode, this value indicates the effective low polarity cycles of input waveform. This value is based on the PWM clock. The value ranges from 0 to (2^32-1)

PWM_PWM1_CTRL

Address: Operational Base + offset (0x001c)

Bit	Attr	Reset Value	Description
31:24	RW	0x00	rpt This field defines the repeated effective periods of output waveform in one-shot mode. The value N means N+1 repeated effective periods
23:16	RW	0x00	scale This field defines the scale factor applied to prescaled clock. The value N means the clock is divided by 2*N. If N is 0, it means that the clock is divided by 512(2*256)
15	RO	0x0	Reserved
14:12	RW	0x0	prescale This field defines the prescale factor applied to input clock. The value N means that the input clock is divided by 2^N
11:10	RO	0x0	Reserved
9	RW	0x0	clk_sel 1'b0: non-scaled clock is selected as PWM clock source. It means that the prescale clock is directly used as the PWM clock source 1'b1: scaled clock is selected as PWM clock source
8	RW	0x0	force_clk_en 1'b0: disabled, when PWM channel is inactive state, the clk_pwm to PWM Clock prescale module is blocked to reduce power consumption. 1'b1: enabled, the clk_pwm to PWM Clock prescale module is always enable
7	RW	0x0	ch_cnt_en 1'b0: disabled 1'b1: enabled
6	RW	0x0	conlock pwm period and duty lock to previous configuration 1'b0: disable lock 1'b1: enable lock

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5	RW	0x0	output_mode 1'b0: left aligned mode 1'b1: center aligned mode
4	RW	0x0	inactive_pol This defines the output waveform polarity when PWM channel is in inactive state. The inactive state means that PWM finishes the complete waveform in one-shot mode or PWM channel is disabled. 1'b0: negative 1'b1: positive
3	RW	0x0	duty_pol This defines the polarity for duty cycle. PWM starts the output waveform with duty cycle. 1'b0: negative 1'b1: positive
2:1	RW	0x0	pwm_mode 2'b00: One shot mode. PWM produces the waveform within the repeated times defined by PWMx_CTRL_rpt 2'b01: Continuous mode. PWM produces the waveform continuously 2'b10: Capture mode. PWM measures the cycles of high/low polarity of input waveform. 2'b11: Reserved
0	RW	0x0	pwm_en 1'b0: disabled 1'b1: enabled. If the PWM is worked in the one-shot mode, this bit will be cleared at the end of operation

PWM_PWM2_CNT

Address: Operational Base + offset (0x0020)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	CNT The 32-bit indicates current value of PWM Channel 2 counter. The counter runs at the rate of PWM clock. The value ranges from 0 to (2^32-1)

PWM_PWM2_PERIOD_HPR

Address: Operational Base + offset (0x0024)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	PERIOD_HPR If PWM is operated at the continuous mode or one-shot mode, this value defines the period of the output waveform. Note that, if the PWM is operated at the center-aligned mode, the period should be an even one, and therefore only the bit [31:1] is taken into account and bit [0] always considered as 0. If PWM is operated at the capture mode, this value indicates the effective high polarity cycles of input waveform. This value is based on the PWM clock. The value ranges from 0 to (2^32-1)

PWM_PWM2_DUTY_LPR

Address: Operational Base + offset (0x0028)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	DUTY_LPR If PWM is operated at the continuous mode or one-shot mode, this value defines the duty cycle of the output waveform. The PWM starts the output waveform with duty cycle. Note that, if the PWM is operated at the center-aligned mode, the period should be an even one, and therefore only the [31:1] is taken into account. If PWM is operated at the capture mode, this value indicates the effective low polarity cycles of input waveform. This value is based on the PWM clock. The value ranges from 0 to (2^32-1)

PWM_PWM2_CTRL

Address: Operational Base + offset (0x002c)

Bit	Attr	Reset Value	Description
31:24	RW	0x00	rpt This field defines the repeated effective periods of output waveform in one-shot mode. The value N means N+1 repeated effective periods
23:16	RW	0x00	scale This fields defines the scale factor applied to prescaled clock. The value N means the clock is divided by 2*N. If N is 0, it means that the clock is divided by 512(2*256)
15	RO	0x0	Reserved
14:12	RW	0x0	prescale This field defines the prescale factor applied to input clock. The value N means that the input clock is divided by 2^N
11:10	RO	0x0	Reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
9	RW	0x0	clk_sel 1'b0: non-scaled clock is selected as PWM clock source. It means that the prescale clock is directly used as the PWM clock source 1'b1: scaled clock is selected as PWM clock source
8	RW	0x0	force_clk_en 1'b0: disabled, when PWM channel is inactive state, the clk_pwm to PWM Clock prescale module is blocked to reduce power consumption. 1'b1: enabled, the clk_pwm to PWM Clock prescale module is always enable
7	RW	0x0	ch_cnt_en 1'b0: disabled 1'b1: enabled
6	RW	0x0	conlock pwm period and duty lock to previous configuration 1'b0: disable lock 1'b1: enable lock
5	RW	0x0	output_mode 1'b0: left aligned mode 1'b1: center aligned mode
4	RW	0x0	inactive_pol This defines the output waveform polarity when PWM channel is in inactive state. The inactive state means that PWM finishes the complete waveform in one-shot mode or PWM channel is disabled. 1'b0: negative 1'b1: positive
3	RW	0x0	duty_pol This defines the polarity for duty cycle. PWM starts the output waveform with duty cycle. 1'b0: negative 1'b1: positive
2:1	RW	0x0	pwm_mode 2'b00: One shot mode. PWM produces the waveform within the repeated times defined by PWMx_CTRL_rpt. 2'b01: Continuous mode. PWM produces the waveform continuously 2'b10: Capture mode. PWM measures the cycles of high/low polarity of input waveform. 2'b11: Reserved
0	RW	0x0	pwm_en 1'b0: disabled 1'b1: enabled. If the PWM is worked in the one-shot mode, this bit will be cleared at the end of operation

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

PWM_PWM3_CNT

Address: Operational Base + offset (0x0030)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	CNT The 32-bit indicates current value of PWM Channel 3 counter. The counter runs at the rate of PWM clock. The value ranges from 0 to (2^32-1)

PWM_PWM3_PERIOD_HPR

Address: Operational Base + offset (0x0034)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	PERIOD_HPR If PWM is operated at the continuous mode or one-shot mode, this value defines the period of the output waveform. Note that, if the PWM is operated at the center-aligned mode, the period should be an even one, and therefore only the bit [31:1] is taken into account and bit [0] always considered as 0. If PWM is operated at the capture mode, this value indicates the effective high polarity cycles of input waveform. This value is based on the PWM clock. The value ranges from 0 to (2^32-1)

PWM_PWM3_DUTY_LPR

Address: Operational Base + offset (0x0038)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	DUTY_LPR If PWM is operated at the continuous mode or one-shot mode, this value defines the duty cycle of the output waveform. The PWM starts the output waveform with duty cycle. Note that, if the PWM is operated at the center-aligned mode, the period should be an even one, and therefore only the [31:1] is taken into account. If PWM is operated at the capture mode, this value indicates the effective low polarity cycles of input waveform. This value is based on the PWM clock. The value ranges from 0 to (2^32-1)

PWM_PWM3_CTRL

Address: Operational Base + offset (0x003c)

Bit	Attr	Reset Value	Description
31:24	RW	0x00	rpt This field defines the repeated effective periods of output waveform in one-shot mode. The value N means N+1 repeated effective periods

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
23:16	RW	0x00	scale This field defines the scale factor applied to prescaled clock. The value N means the clock is divided by 2*N. If N is 0, it means that the clock is divided by 512(2*256)
15	RO	0x0	Reserved
14:12	RW	0x0	prescale This field defines the prescale factor applied to input clock. The value N means that the input clock is divided by 2^N
11:10	RO	0x0	Reserved
9	RW	0x0	clk_sel 1'b0: non-scaled clock is selected as PWM clock source. It means that the prescale clock is directly used as the PWM clock source 1'b1: scaled clock is selected as PWM clock source
8	RW	0x0	force_clk_en 1'b0: disabled, when PWM channel is inactive state, the clk_pwm to PWM Clock prescale module is blocked to reduce power consumption. 1'b1: enabled, the clk_pwm to PWM Clock prescale module is always enable
7	RW	0x0	ch_cnt_en 1'b0: disabled 1'b1: enabled
6	RW	0x0	conlock pwm period and duty lock to previous configuration 1'b0: disable lock 1'b1: enable lock
5	RW	0x0	output_mode 1'b0: left aligned mode 1'b1: center aligned mode
4	RW	0x0	inactive_pol This defines the output waveform polarity when PWM channel is in inactive state. The inactive state means that PWM finishes the complete waveform in one-shot mode or PWM channel is disabled. 1'b0: negative 1'b1: positive
3	RW	0x0	duty_pol This defines the polarity for duty cycle. PWM starts the output waveform with duty cycle. 1'b0: negative 1'b1: positive

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2:1	RW	0x0	pwm_mode 2'b00: One shot mode. PWM produces the waveform within the repeated times defined by PWMx_CTRL_rpt 2'b01: Continuous mode. PWM produces the waveform continuously 2'b10: Capture mode. PWM measures the cycles of high/low polarity of input waveform. 2'b11: Reserved
0	RW	0x0	pwm_en 1'b0: disabled 1'b1: enabled. If the PWM is worked in the one-shot mode, this bit will be cleared at the end of operation

PWM_INTSTS

Address: Operational Base + offset (0x0040)

Bit	Attr	Reset Value	Description
31:12	RO	0x0	Reserved
11	RO	0x0	CH3_Pol This bit is used in capture mode in order to identify the transition of the input waveform when interrupt is generated. When bit is 1, please refer to PWM3_PERIOD_HPR to know the effective high cycle of Channel 3 input waveform. Otherwise, please refer to PWM3_PERIOD_LPR to know the effective low cycle of Channel 3 input waveform. Write 1 to CH3_IntSts will clear this bit
10	RO	0x0	CH2_Pol This bit is used in capture mode in order to identify the transition of the input waveform when interrupt is generated. When bit is 1, please refer to PWM2_PERIOD_HPR to know the effective high cycle of Channel 2 input waveform. Otherwise, please refer to PWM2_PERIOD_LPR to know the effective low cycle of Channel 2 input waveform. Write 1 to CH2_IntSts will clear this bit
9	RO	0x0	CH1_Pol This bit is used in capture mode in order to identify the transition of the input waveform when interrupt is generated. When bit is 1, please refer to PWM1_PERIOD_HPR to know the effective high cycle of Channel 1 input waveform. Otherwise, please refer to PWM1_PERIOD_LPR to know the effective low cycle of Channel 1 input waveform. Write 1 to CH1_IntSts will clear this bit

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
8	RO	0x0	CH0_Pol This bit is used in capture mode in order to identify the transition of the input waveform when interrupt is generated. When bit is 1, please refer to PWM0_PERIOD_HPR to know the effective high cycle of Channel 0 input waveform. Otherwise, please refer to PWM0_PERIOD_LPR to know the effective low cycle of Channel 0 input waveform. Write 1 to CH0_IntSts will clear this bit
7	W1C	0x0	CH3_pwr_IntSts 1'b0: Channel 3 power key Interrupt not generated 1'b1: Channel 3 power key Interrupt generated
6	W1C	0x0	CH2_pwr_IntSts 1'b0: Channel 2 power key Interrupt not generated 1'b1: Channel 2 power key Interrupt generated
5	W1C	0x0	CH1_pwr_IntSts 1'b0: Channel 1 power keyInterrupt not generated 1'b1: Channel 1 power key Interrupt generated
4	W1C	0x0	CH0_pwr_IntSts 1'b0: Channel 0 power key Interrupt not generated 1'b1: Channel 0 power key Interrupt generated
3	W1C	0x0	CH3_IntSts 1'b0: Channel 3 Interrupt not generated 1'b1: Channel 3 Interrupt generated
2	W1C	0x0	CH2_IntSts 1'b0: Channel 2 Interrupt not generated 1'b1: Channel 2 Interrupt generated
1	W1C	0x0	CH1_IntSts 1'b0: Channel 1 Interrupt not generated 1'b1: Channel 1 Interrupt generated
0	W1C	0x0	CH0_IntSts 1'b0: Channel 0 Interrupt not generated 1'b1: Channel 0 Interrupt generated

PWM_INT_EN

Address: Operational Base + offset (0x0044)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	Reserved
7	RW	0x0	CH3_pwr_Int_en 1'b0: Channel 3 power key Interrupt disabled 1'b1: Channel 3 power key Interrupt enabled
6	RW	0x0	CH2_pwr_Int_en 1'b0: Channel 2 power key Interrupt disabled 1'b1: Channel 2 power key Interrupt enabled

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
5	RW	0x0	CH1_pwr_Int_en 1'b0: Channel 1 power key Interrupt disabled 1'b1: Channel 1 power key Interrupt enabled
4	RW	0x0	CH0_pwr_Int_en 1'b0: Channel 0 power key Interrupt disabled 1'b1: Channel 0 power key Interrupt enabled
3	RW	0x0	CH3_Int_en 1'b0: Channel 3 Interrupt disabled 1'b1: Channel 3 Interrupt enabled
2	RW	0x0	CH2_Int_en 1'b0: Channel 2 Interrupt disabled 1'b1: Channel 2 Interrupt enabled
1	RW	0x0	CH1_Int_en 1'b0: Channel 1 Interrupt disabled 1'b1: Channel 1 Interrupt enabled
0	RW	0x0	CH0_Int_en 1'b0: Channel 0 Interrupt disabled 1'b1: Channel 0 Interrupt enabled

PWM_FIFO_CTRL

Address: Operational Base + offset (0x0050)

Bit	Attr	Reset Value	Description
31:14	RO	0x0	Reserved
13:12	RW	0x0	dma_ch_sel 2'b00: Select PWM0 2'b01: Select PWM1 2'b10: Select PWM2 2'b11: Select PWM3
11	RO	0x0	Reserved
10	RW	0x0	dma_ch_sel_en 1'b1: Enable, use dma_ch_sel to select the channel to FIFO mode and DMA mode. 1'b0: Disable, select the channel PWM3 to FIFO mode and DMA mode
9	RW	0x0	timeout_en fifo timeout enable
8	RW	0x0	dma_mode_en 1'b1: enable 1'b0: disable
7	RO	0x0	Reserved
6:4	RW	0x0	almost_full_watermark Almost full Watermark level
3	RW	0x0	watermark_int_en Watermark full interrupt

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2	RW	0x0	overflow_int_en When high, an interrupt asserts when the fifo overflow
1	RW	0x0	full_int_en When high, an interrupt asserts when the FIFO is full
0	RW	0x0	fifo_mode_sel When high, PWM FIFO mode is activated

PWM_FIFO_INTSTS

Address: Operational Base + offset (0x0054)

Bit	Attr	Reset Value	Description
31:5	RO	0x0	Reserved
4	RO	0x1	fifo_empty_status This bit indicates the FIFO is empty
3	W1C	0x0	timieout_intsts Timeout interrupt
2	W1C	0x0	fifo_watermark_full_intsts This bit indicates the FIFO is Watermark Full
1	W1C	0x0	fifo_overflow_intsts This bit indicates the FIFO is overflow
0	W1C	0x0	fifo_full_intsts This bit indicates the FIFO is full

PWM_FIFO_TOUTTHR

Address: Operational Base + offset (0x0058)

Bit	Attr	Reset Value	Description
31:20	RO	0x0	Reserved
19:0	RW	0x00000	timeout_threshold FIFO Timeout value(unit pwm clock)

PWM_VERSION_ID

Address: Operational Base + offset (0x005c)

Bit	Attr	Reset Value	Description
31:24	RW	0x02	main_version Main version 8'h0:Base version 8'h1:Support DMA mode 8'h2:Support DMA mode and Power key mode
23:16	RW	0x12	minor_version Minor version
15:0	RW	0x0b34	svn_version SVN version

PWM_FIFO

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x0060)

Bit	Attr	Reset Value	Description
31	RO	0x0	pol This bit indicates the polarity of the lower 31-bit counter. 1'b0: Low 1'b1: High
30:0	RO	0x00000000	cycle_cnt This 31-bit counter indicates the effective cycles of high/low waveform

PWM_PWRMATCH_CTRL

Address: Operational Base + offset (0x0080)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	Reserved
15	RW	0x0	CH3_pwrkey_int_ctrl 1'b0: Assert interrupt after key capture with power key match 1'b1: Assert interrupt after key capture without power key match
14:12	RO	0x0	Reserved
11	RW	0x0	CH3_pwrkey_capture_ctrl 1'b0: Capture the value after interrupt 1'b1: Capture the value directly
10:8	RO	0x0	Reserved
7	RW	0x0	CH3_pwrkey_polarity 1'b0: pwm in polarity is positive 1'b1: pwm in polarity is negative
6:4	RO	0x0	Reserved
3	RW	0x0	CH3_pwrkey_enable 1'b0: Disabled 1'b1: Enabled
2:0	RO	0x0	Reserved

PWM_PWRMATCH_LPRE

Address: Operational Base + offset (0x0084)

Bit	Attr	Reset Value	Description
31:16	RW	0x238c	cnt_max The maximum counter value
15:0	RW	0x22c4	cnt_min The minimum counter value

PWM_PWRMATCH_HPRE

Address: Operational Base + offset (0x0088)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:16	RW	0x11f8	cnt_max The maximum counter value
15:0	RW	0x1130	cnt_min The minimum counter value

PWM_PWRMATCH_LD

Address: Operational Base + offset (0x008c)

Bit	Attr	Reset Value	Description
31:16	RW	0x0294	cnt_max The maximum counter value
15:0	RW	0x01cc	cnt_min The minimum counter value

PWM_PWRMATCH_HD_ZERO

Address: Operational Base + offset (0x0090)

Bit	Attr	Reset Value	Description
31:16	RW	0x0294	cnt_max The maximum counter value
15:0	RW	0x01cc	cnt_min The minimum counter value

PWM_PWRMATCH_HD_ONE

Address: Operational Base + offset (0x0094)

Bit	Attr	Reset Value	Description
31:16	RW	0x06fe	cnt_max The maximum counter value
15:0	RW	0x0636	cnt_min The minimum counter value

PWM_PWRMATCH_VALUE0

Address: Operational Base + offset (0x0098)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pwrkey_match_value Power key match value

PWM_PWRMATCH_VALUE1

Address: Operational Base + offset (0x009c)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pwrkey_match_value Power key match value

PWM_PWRMATCH_VALUE2

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Address: Operational Base + offset (0x00a0)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pwrkey_match_value Power key match value

PWM_PWRMATCH_VALUE3

Address: Operational Base + offset (0x00a4)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pwrkey_match_value Power key match value

PWM_PWRMATCH_VALUE4

Address: Operational Base + offset (0x00a8)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pwrkey_match_value Power key match value

PWM_PWRMATCH_VALUE5

Address: Operational Base + offset (0x00ac)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pwrkey_match_value Power key match value

PWM_PWRMATCH_VALUE6

Address: Operational Base + offset (0x00b0)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pwrkey_match_value Power key match value

PWM_PWRMATCH_VALUE7

Address: Operational Base + offset (0x00b4)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pwrkey_match_value Power key match value

PWM_PWRMATCH_VALUE8

Address: Operational Base + offset (0x00b8)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pwrkey_match_value Power key match value

PWM_PWRMATCH_VALUE9

Address: Operational Base + offset (0x00bc)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pwrkey_match_value Power key match value

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

PWM_PWM3_PWRCAPTURE_VALUE

Address: Operational Base + offset (0x00cc)

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	pwrkey_capture_value Power key capture value

PWM_FILTER_CTRL

Address: Operational Base + offset (0x00d0)

Bit	Attr	Reset Value	Description
31:13	RO	0x0	Reserved
12:4	RW	0x000	filter_number Filter window number
3	RW	0x0	CH3_input_filter_enable 1'b0: Disabled 1'b1: Enabled
2	RW	0x0	CH2_input_filter_enable 1'b0: Disabled 1'b1: Enabled
1	RW	0x0	CH1_input_filter_enable 1'b0: Disabled 1'b1: Enabled
0	RW	0x0	CH0_input_filter_enable 1'b0: Disabled 1'b1: Enabled

27.5 Interface Description

Table 27-1 PWM Interface Description

Module Pin	Direction	Pad Name	IOMUX Setting
PWM0	I/O	GPIO0_B7/PWM0/OTG_DRV	GRF_GPIO0B_IOMUX[15:14]=2'b1
PWM1	I/O	GPIO0_C3/PWM1/UART3_TX	GRF_GPIO0C_IOMUX[7:6]=2'b1
PWM2	I/O	GPIO0_C5/PCIE_WAKE_M1/PWM2	GRF_GPIO0C_IOMUX[2:1]=2'b1
PWM3	I/O	GPIO0_C4/PWM3/UART3_RX	GRF_GPIO0C_IOMUX[9:8]=2'b1
PWM5	I/O	GPIO1_B7/SPI0_CLK/PWM5	GRF_GPIO1BH_IOMUX[14:12]=3'b010
PWM6	I/O	GPIO3_A1/I2S1_SCLK/PWM6	GRF_GPIO3AL_IOMUX[6:4]=3'b010
PWM7	I/O	GPIO3_A2/I2S1_MCLK/PWM7	GRF_GPIO3AL_IOMUX[10:8]=3'b010
PWM8	I/O	GPIO3_D0/LCDC_D14/PWM8/SPI1_MOSI_M1	GRF_GPIO3DL_IOMUX[2:0]=3'b010
PWM9	I/O	GPIO3_D1/LCDC_D15/PWM9/SPI1_CSN0_M1	GRF_GPIO3DL_IOMUX[6:4]=3'b010

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Module Pin	Direction	Pad Name	IOMUX Setting
PWM10	I/O	GPIO3_D2/LCDC_D16/PWM10/SPI1_MISO_M1	GRF_GPIO3DL_IOMUX[10:8]=3'b010
PWM11	I/O	GPIO3_D3/LCDC_D17/PWM11/SPI1_CSN1_M1	GRF_GPIO3DL_IOMUX[14:12]=3'b010

Notes: I=input, O=output, I/O=input/output.

27.6 Application Notes

27.6.1 PWM Capture Mode Standard Usage Flow

1. Set PWM_PWMx_CTRL.pwm_en to '0' to disable the PWM channel.
2. Choose the prescale factor and the scale factor for clk_pwm by programming PWM_PWMx_CTRL.prescale and PWM_PWMx_CTRL.scale, and select the clock needed by setting PWM_PWMx_CTRL.clk_sel.
3. Configure the channel to work in the capture mode.
4. Enable the PWM_INT_EN.chx_int_en to enable the interrupt generation.
5. Set PWM_FILTER_CTRL. filter_number, then Enable the PWM_FILTER_CTRL. CHx_input_filter_enable(Optional).
6. Enable the channel by writing '1' to PWM_PWMx_CTRL.pwm_en bit to start the channel.
7. When an interrupt is asserted, refer to INTSTS register to know the raw interrupt status. If the corresponding polarity flag is set, turn to PWM_PWMx_PERIOD_HPC register to know the effective high cycles of input waveforms, otherwise turn to PWM_PWMx_DUTY_LPC register to know the effective low cycles.
8. Write '0' to PWM_PWMx_CTRL.pwm_en to disable the channel.

27.6.2 PWM Capture DMA Mode Standard Usage Flow

1. Set PWM_PWMx_CTRL.pwm_en to '0' to disable the PWM channel.
2. Choose the prescale factor and the scale factor for clk_pwm by programming PWM_PWMx_CTRL.prescale and PWM_PWMx_CTRL.scale, and select the clock needed by setting PWM_PWMx_CTRL.clk_sel.
3. Configure the channel 3 to work in the capture mode.
4. Configure the PWM_FIFO_CTRL.dma_mode_en and PWM_FIFO_CTRL.fifo_mode_sel to enable the DMA mode. Configure PWM_FIFO_CTRL.almost_full_watermark at appropriate value.
5. Configure DMAC_BUS to transfer data from PWM to DDR.
6. Set PWM_FILTER_CTRL.filter_number, then Enable the PWM_FILTER_CTRL. CHx_input_filter_enable(Optional).
7. Enable the channel by writing '1' to PWM_PWMx_CTRL.pwm_en bit to start the channel.
8. When a dma_req is asserted, DMAC_BUS transfer the data of effective high cycles and low cycles of input waveforms to DDR.
9. Write '0' to PWM_PWMx_CTRL.pwm_en to disable the channel.

27.6.3 PWM Power key Capture Mode Standard Usage Flow

1. Set PWM_PWM3_CTRL.pwm_en to '0' to disable the PWM channel.
2. Choose the prescale factor and the scale factor for clk_pwm by programming PWM_PWM3_CTRL.prescale and PWM_PWM3_CTRL.scale, and select the clock needed by setting PWM_PWM3_CTRL.clk_sel. The clock should be 1 Mhz after division.
3. Configure the channel to work in the capture mode.
4. Enable the PWM_INT_EN.CH3_int_pwr to enable the interrupt generation.
5. Set the PWM_PWRMATCH_VALUE0~9 registers for the 10 power key match value.
6.Set max_cnt and min_cnt of follow register: PWM_PWRMATCH_LPRE, PWM_PWRMATCH_HPRE, PWM_PWRMATCH_LD, PWM_PWRMATCH_HD_ZERO, PWM_PWRMATCH_HD_ONE. It doesn't need to set these registers when the default value can meet the requirement.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

7.Set PWM_PWRMATCH_CTRL.CH3_pwrkey_polarity for the polarity of power key signal, the default value is 0. Enable the PWM_PWRMATCH_CTRL.CH3_pwrkey_enable.
8. Set PWM_FILTER_CTRL. filter_number, then Enable the PWM_FILTER_CTRL. CH3_input_filter_enable(Optional).
9. Enable the channel by writing '1' to PWM_PWM3_CTRL.pwm_en bit to start the channel.
10. When an interrupt is asserted, refer to INTSTS register to know the raw interrupt status, and refer to PWM_PWM3_PWRCAPTURE_VALUE to know the power key capture value.
11. Write '0' to PWM_PWM3_CTRL.pwm_en to disable the channel.

27.6.4 PWM One-shot Mode/Continuous Standard Usage Flow

1. Set PWM_PWMx_CTRL.pwm_en to '0' to disable the PWM channel.
2. Choose the prescale factor and the scale factor for pclk by programming PWM_PWMx_CTRL.prescale and PWM_PWMx_CTRL.scale, and select the clock needed by setting PWM_PWMx_CTRL.clk_sel.
3. Choose the output mode by setting PWM_PWMx_CTRL.output_mode, and set the duty polarity and inactive polarity by programming PWM_PWMx_CTRL.duty_pol and PWM_PWMx_CTRL.inactive_pol.
4. Set the PWM_PWMx_CTRL.rpt if the channel is desired to work in the one-shot mode.
5. Configure the channel to work in the one-shot mode or the continuous mode.
6. Enable the PWM_INT_EN.chx_int_en to enable the interrupt generation if if the channel is desired to work in the one-shot mode.
7. If the channel is working in the one-shot mode, an interrupt is asserted after the end of operation, and the PWM_PWMx_CTRL.pwm_en is automatically cleared. Whatever mode the channel is working in, write '0' to PWM_PWMx_CTRL.pwm_en bit to disable the PWM channel.

27.6.5 Low-power Usage Flow

The default value of PWM_PWMx_CTRL.force_clk_en is '0' which make the channel enter the low-power mode. In low-power mode, When the PWM channel is inactive, the clk_pwm to the clock prescale module is gated in order to reduce the power consumption. User can set PWM_PWMx_CTRL.force_clk_en to '1' which will make the channel quit the low-power mode. After the setting, the clk_pwm to the clock prescale module is always enable.

27.6.6 Other notes

When the channel is active to produce waveforms, it is free to program the PWM_PWMx_PERIOD_HPC and PWM_PWMx_DUTY_LPC register. User can use PWM_PWMx_CTRL.conlock to take period and duty effect at the same time. The usage flow is as follow:
1. Set PWM_PWMx_CTRL.conlock to '1'.
2. Set PWM_PWMx_PERIOD_HPC and PWM_PWMx_DUTY_LPC.
3. Set PWM_PWMx_CTRL.conlock to '0', others bits in PWM_PWMx_CTRL should be appropriate.
After above configuration, the change will not take effect immediately until the current period ends.
An active channel can be changed to another operation mode without disable the PWM channel. However, during the transition of the operation mode there may be some irregular output waveforms. So does changing the clock division factor when the channel is active.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 28 Watchdog Timer(WDT)

28.1 Overview

Watchdog Timer (WDT) is an APB slave peripheral that can be used to prevent system lockup that may becaused by conflicting parts or programs in a SoC.The WDT would generate interrupt or reset signal when it's counter reaches zero, then a reset controller would reset the system. In *RK1808* there is a Non-secure WDT(WDT_NS) and a Secure WDT(WDT_S);

WDT supports the following features:

- 32 bits APB bus width
- WDT counter's clock is pclk
- 32 bits WDT counter width
- Counter counts down from a preset value to 0 to indicate the occurrence of a timeout
- WDT can perform two types of operations when timeout occurs:
  - Generate a system reset
  - First generate an interrupt and if this is not cleared by the service routine by the time a second timeout occurs then generate a system reset
- Programmable reset pulse length
- Total 16 defined-ranges of main timeout period

28.2 Block Diagram

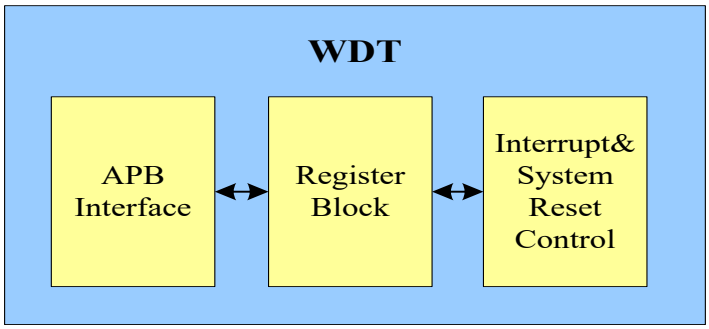


Fig. 28-1 WDT block diagram

Block Descriptions:

- APB Interface

The APB Interface implements the APB slave operation. Its data bus width is 32 bits.

- Register Block

A register block that read coherence for the current count register.

- Interrupt & system reset control

An interrupt/system reset generation block is comprised of a decrementing counter and control logic.

28.3 Function Description

28.3.1 Operation

Counter

The WDT counts from a preset (timeout) value in descending order to zero. When the counter reaches zero, depending on the output response mode selected, either a system reset or an interrupt occurs. When the counter reaches zero, it wraps to the selected timeout value and continues decrementing. The user can restart the counter to its initial value. This is programmed by writing to the restart register at any time. The process of restarting the watchdog counter is sometimes referred as kicking the dog. As a safety feature to prevent accidental restarts, the value 0x76 must be written to the Current Counter Value Register (WDT_CRR). (After the counter counts down to 0, it will reload back to the initial value and restart counting.)

Interrupts

The WDT can be programmed to generate an interrupt (and then a system reset) when a

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

timeout occurs. When a 1 is written to the response mode field (RMOD, bit 1) of the Watchdog Timer Control Register (WDT_CR), the WDT generates an interrupt. If it is not cleared by the time a second timeout occurs, then it generates a system reset. If a restart occurs at the same time the watchdog counter reaches zero, an interrupt is not generated.

System Resets

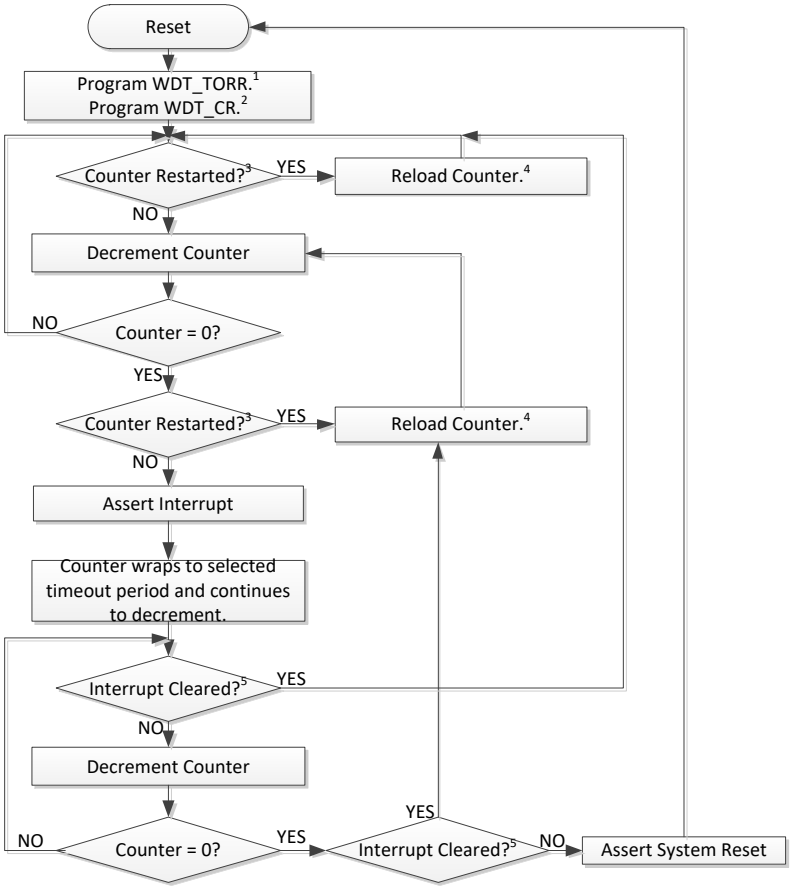
When a 0 is written to the output response mode field (RMOD, bit 1) of the Watchdog Timer Control Register (WDT_CR), the WDT generates a system reset when a timeout occurs.

Reset Pulse Length

The reset pulse length is the number of pclk cycles for which a system reset is asserted. When a system reset is generated, it remains asserted for the number of cycles specified by the reset pulse length or until the system is reset. A counter restart has no effect on the system reset once it has been asserted.

28.3.2 Programming sequence

Operation Flow Chart (Response mode=1)



1. Select required timeout period.
2. Set reset pulse length, response mode, and enable WDT.
3. Write 0x76 to WDT_CRR.
4. Starts back to selected timeout period.
5. Can clear by reading WDT_EOI or restarting (kicking) the counter by writing 0x76 to WDT_CRR.

Fig. 28-2 WDT Operation Flow

28.4 Register Description

This section describes the control/status registers of the design.

28.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
WDT_CR	0x0000	W	0x0000000a	Control Register

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Name	Offset	Size	Reset Value	Description
WDT_TORR	0x0004	W	0x00000000	Timeout range Register
WDT_CCVR	0x0008	W	0x0000ffff	Current counter value Register
WDT_CRR	0x000c	W	0x00000000	Counter restart Register
WDT_STAT	0x0010	W	0x00000000	Interrupt status Register
WDT_EOI	0x0014	W	0x00000000	Interrupt clear Register

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

28.4.2 Detail Register Description

WDT_CR

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4:2	RW	0x2	rst_pluse_lenth Reset pulse length. This is used to select the number of pclk cycles for which the system reset stays asserted. 3'b000: 2 pclk cycles 3'b001: 4 pclk cycles 3'b010: 8 pclk cycles 3'b011: 16 pclk cycles 3'b100: 32 pclk cycles 3'b101: 64 pclk cycles 3'b110: 128 pclk cycles 3'b111: 256 pclk cycles
1	RW	0x1	resp_mode Response mode. Selects the output response generated to a timeout. 1'b0: Generate a system reset. 1'b1: First generate an interrupt and if it is not cleared by the time a second timeout occurs then generate a system reset
0	RW	0x0	wdt_en WDT enable: 1'b0: WDT disabled; 1'b1: WDT enabled. Note: After starting the counter count, write wdt_en to 0, it is impossible to stop counter counting. When wdt_en is zero and the counter count is zero, the counter reloads the count value and counts down again.

WDT_TORR

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
3:0	RW	0x0	timeout_period Timeout period. This field is used to select the timeout period from which the watchdog counter restarts. A change of the timeout period takes effect only after the next counter restart (kick). The range of values available for a 32-bit watchdog counter are: 4'b0000: 0x0000ffff 4'b0001: 0x0001ffff 4'b0010: 0x0003ffff 4'b0011: 0x0007ffff 4'b0100: 0x000fffff 4'b0101: 0x001fffff 4'b0110: 0x003fffff 4'b0111: 0x007fffff 4'b1000: 0x00ffffff 4'b1001: 0x01ffffff 4'b1010: 0x03ffffff 4'b1011: 0x07ffffff 4'b1100: 0x0fffffff 4'b1101: 0x1fffffff 4'b1110: 0x3fffffff 4'b1111: 0x7fffffff

WDT_CCVR

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:0	RO	0x0000ffff	cur_cnt Current counter value. This register, when read, is the current value of the internal counter. This value is read coherently when ever it is read

WDT_CRR

Address: Operational Base + offset (0x000c)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	W1C	0x00	cnt_restart Counter restart. This register is used to restart the WDT counter. As a safety feature to prevent accidental restarts, the value 0x76 must be written. A restart also clears the WDT interrupt. Reading this register returns zero

WDT_STAT

Address: Operational Base + offset (0x0010)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RO	0x0	wdt_status This register shows the interrupt status of the WDT. 1'b1: Interrupt is active regardless of polarity; 1'b0: Interrupt is inactive

WDT_EOI

Address: Operational Base + offset (0x0014)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RO	0x0	wdt_int_clr Clears the watchdog interrupt. This can be used to clear the interrupt without restarting the watchdog counter

28.5 Application Notes

Please refer to the function description section

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 29 Successive Approximation Register ADC(SARADC)

29.1 Overview

The SARADC is a 4-channel signal-ended 10-bit Successive Approximation Register (SAR) A/D Converter. It uses the supply and ground as it reference which avoid use of any external reference. It converts the analog input signal into 10-bit binary digital codes at maximum conversion rate of 1MSPS with 13MHz A/D converter clock.

29.2 Block Diagram

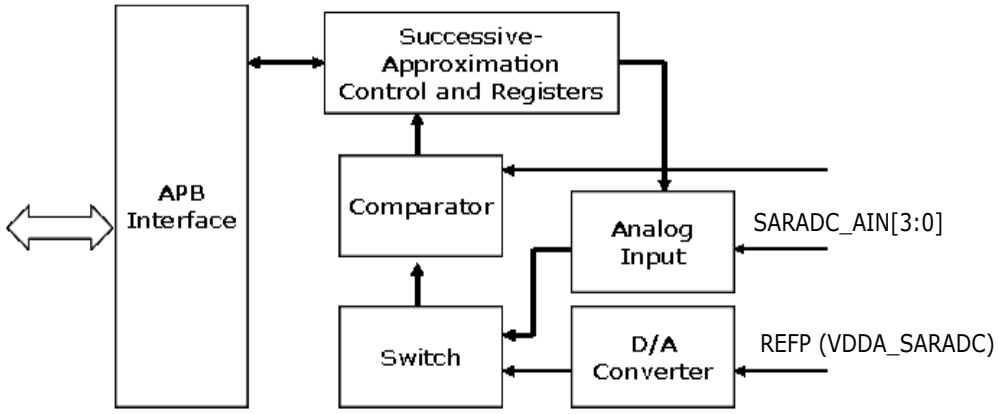


Fig. 29-1 SARADC block diagram

Successive-Approximate Register and Control Logic Block

This block is exploited to realize binary search algorithm, storing the intermediate result and generate control signal for analog block.

Comparator Block

This block compares the analog input SARADC_AIN[3:0] with the voltage generated from D/A Converter, and output the comparison result to SAR and Control Logic Block for binary search. Three level amplifiers are employed in this comparator to provide enough gain.

29.3 Function Description

29.3.1 APB Interface

SARADC works at single-sample operation mode. This mode is useful to sample an analog input when there is a gap between two samples to be converted. In this mode START is asserted only on the rising edge of CLKIN where conversion is needed. At the end of every conversion EOC signal is made high and valid output data is available at the rising edge of EOC. The detailed timing diagram will be shown in the following.

29.4 Register description

29.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
SARADC_DATA	0x0000	W	0x00000000	This register contains the data after A/D Conversion
SARADC_STAS	0x0004	W	0x00000000	The status register of A/D Converter
SARADC_CTRL	0x0008	W	0x00000000	The control register of A/D Converter
SARADC_DLY_PU_SOC	0x000c	W	0x00000000	Delay between power up and start command

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

29.4.2 Detail Register Description

SARADC_DATA

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:10	RO	0x0	Reserved
9:0	RO	0x000	adc_data

SARADC_STAS

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	Reserved
0	RO	0x0	adc_status 1'b0: ADC stop 1'b1: Conversion in progress

SARADC_CTRL

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:7	RO	0x0	Reserved
6	RW	0x0	int_status This bit will be set to 1 when end-of-conversion. Set 0 to clear the interrupt
5	RW	0x0	int_en 1'b0: Disable 1'b1: Enable
4	RO	0x0	Reserved
3	RW	0x0	adc_power_ctrl 1'b0: ADC power down; 1'b1: ADC power up and reset. Start signal will be asserted (DLY_PU_SOC + 2) sclk clock period later after power up
2:0	RW	0x0	adc_input_src_sel 3'b000 : Input source 0 (SARADC_AIN[0]) 3'b001 : Input source 1 (SARADC_AIN[1]) 3'b010 : Input source 2 (SARADC_AIN[2]) 3'b011 : Input source 3 (SARADC_AIN[3]) Others : Reserved

SARADC_DLY_PU_SOC

Address: Operational Base + offset (0x000c)

Bit	Attr	Reset Value	Description
31:6	RO	0x0	Reserved
5:0	RW	0x00	DLY_PU_SOC The start signal will be asserted (DLY_PU_SOC + 2) sclk clock period later after power up

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

29.5 Timing Diagram

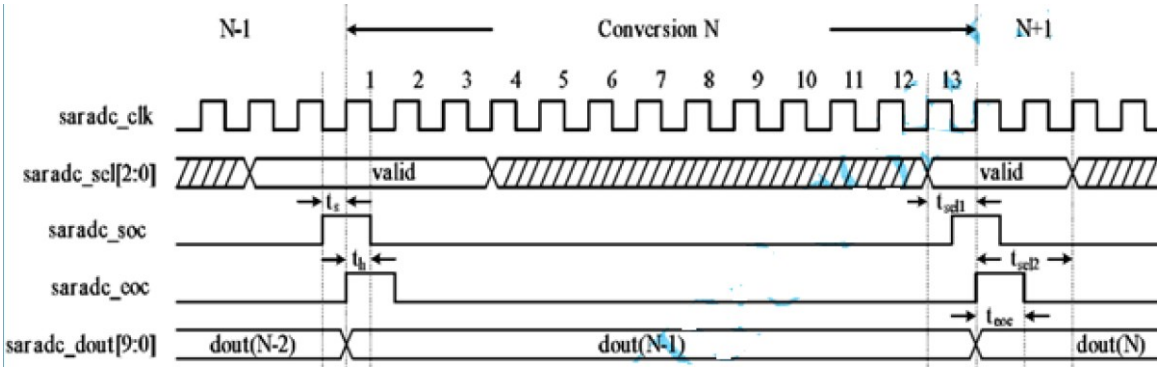


Fig. 29-2 SAR-ADC timing diagram in single-sample conversion mode

29.6 Application Notes

Steps of adc conversion:

- Write SARADC_CTRL[3] as 0 to power down adc converter.
- Write SARADC_CTRL[2:0] as n to select adc channel(n).
- Write SARADC_CTRL[5] as 1 to enable adc interrupt.
- Write SARADC_CTRL[3] as 1 to power up adc converter.
- Wait for adc interrupt or poll SARADC_STAS register to assert whether the conversion is completed
- Read the conversion result from SARADC_DATA[9:0]

Note: The A/D converter was designed to operate at maximum 1MHZ.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Chapter 30 Temperature Sensor ADC(TSADC)

30.1 Overview

TSADC controller module supports user-defined mode and automatic mode. All the control signals of TSADC are entirely by software writing to register for direct control when user-defined mode is enable. All the control signals are driven by hardware circuit and the results are checked automatically.

TS-ADC Controller supports the following features:

- Support User-Defined Mode and Automatic Mode
- In User-Defined Mode, start_of_conversion can be controlled completely by software, and also can be generated by hardware
- In Automatic Mode, the temperature of alarm(high/low temperature) interrupt can be configurable
- In Automatic Mode, the temperature of system reset can be configurable
- Support 1 channel TS-ADC, the temperature criteria can be configurable
- In Automatic Mode, the time interval of temperature detection can be configurable
- In Automatic Mode, when detecting a high temperature, the time interval of temperature detection can be configurable
- High temperature denounce can be configurable
- 10-bit SARADC up to 50KS/s sampling rate

30.2 Block Diagram

TSADC controller comprises with:

- APB Interface
- TS-ADC control logic

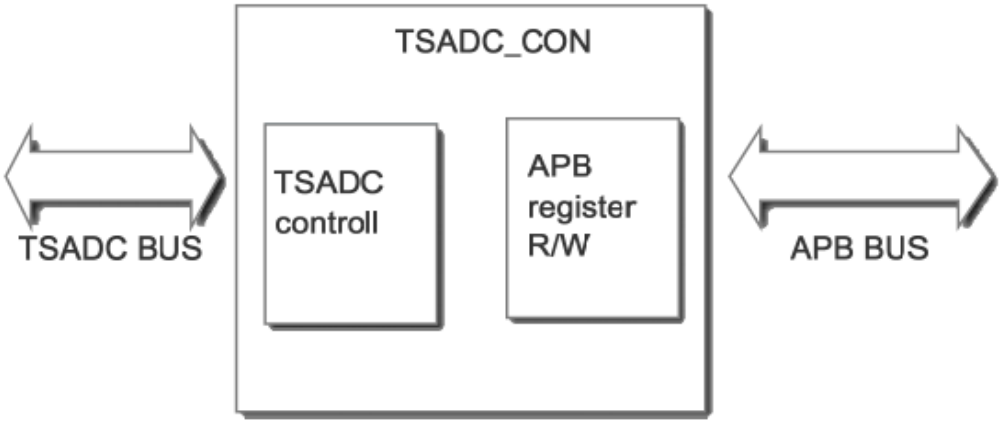


Fig. 30-1 TSADC Controller Block Diagram

30.3 Function Description

30.3.1 APB Interface

There is an APB Slave interface in TS-ADC Controller, which is used to configure the TS-ADC controller registers and look up the temperature from the temperature sensor.

30.3.2 TSADC Controller

This block is exploited to realize binary search algorithm, storing the intermediate result and generate control signal for analog block. This block compares the analog input with the voltage generated from D/A Converter, and output the comparison result to SAR and Control Logic Block for binary search. Three level amplifiers are employed in this comparator to provide enough gain.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

30.4 Register description

30.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
TSADC_USER_CON	0x0000	W	0x00000208	User mode control register
TSADC_AUTO_CON	0x0004	W	0x00000000	Auto mode control register
TSADC_INT_EN	0x0008	W	0x00000000	Interrupt enable register
TSADC_INT_PD	0x000c	W	0x00000000	Interrupt status register
TSADC_DATA0	0x0020	W	0x00000000	Channel0 data register
TSADC_COMP0_INT	0x0030	W	0x00000000	High temperature threshold
TSADC_COMP0_SHUT	0x0040	W	0x00000000	Shut high temperature threshold
TSADC_HIGHT_INT_DEBOUNCE	0x0060	W	0x00000003	High temperature debounce
TSADC_HIGHT_TSHUT_DEBOUNCE	0x0064	W	0x00000003	Shut high temperature debounce
TSADC_AUTO_PERIOD	0x0068	W	0x00010000	ADC auto access period
TSADC_AUTO_PERIOD_HT	0x006c	W	0x00010000	ADC auto access period when temperature is high
TSADC_COMP0_LOW_INT	0x0080	W	0x00000000	Low temperature threshold

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

30.4.2 Detail Register Description

TSADC_USER_CON

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:13	RO	0x0	Reserved
12	RO	0x0	adc_status 1'b0: ADC stop 1'b1: Conversion in progress
11:6	RW	0x08	inter_pd_soc interleave between power down and start of conversion
5	RW	0x0	start When software writes 1 to this bit, start_of_conversion will be assert. This bit will be cleared after ADC access finishing. When ADC_USER_CON[4] = 1'b1 take effect
4	RW	0x0	start_mode 1'b0: ADC controller will assert start_of_conversion after "inter_pd_soc" cycles. 1'b1: The start_of_conversion will be controlled by ADC_USER_CON[5]
3	RW	0x1	adc_power_ctrl 1'b0: ADC power down 1'b1: ADC power up and reset

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
2:0	RW	0x0	adc_input_src_sel 3'b000 : Input source 0 (ADC_AIN[0]) Others : Reserved

TSADC_AUTO_CON

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:26	RO	0x0	Reserved
25	RW	0x0	last_tshut_2cru This bit will set to 1 when tshut is valid, and only be cleared when application write 1 to it. This bit will not be cleared by system reset
24	RW	0x0	last_tshut_2gpio This bit will set to 1 when tshut is valid, and only be cleared when application write 1 to it. This bit will not be cleared by system reset
23:18	RO	0x0	Reserved
17	RO	0x0	sample_dly_sel 1'b0: AUTO_PERIOD is used 1'b1: AUTO_PERIOD_HT is used
16	RO	0x0	auto_status 1'b0: Auto mode stop 1'b1: Auto mode in progress
15:13	RO	0x0	Reserved
12	RW	0x0	src0_lt_en 1'b0: Do not care low level of source 0 1'b1: Enable the low level monitor of source 0
11:9	RO	0x0	Reserved
8	RW	0x0	tshut_prolarity 1'b0: Low active 1'b1: High active
7:5	RO	0x0	Reserved
4	RW	0x0	src0_en 1'b0: Do not monitor channel 0 result 1'b1: Monitor channel 0 in turn
3:2	RO	0x0	Reserved
1	RW	0x0	adc_q_sel 1'b0: Select adc_q, for the case that if the value of adc_q increases, temperature increases too. 1'b1: Select 4096 - adc_q, for the case that if the value of adc_q increases but temperature decreases.
0	RW	0x0	auto_en 1'b0: ADC controller works at user-define mode 1'b1: ADC controller works at auto mode

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

TSADC_INT_EN

Address: Operational Base + offset (0x0008)

Bit	Attr	Reset Value	Description
31:17	RO	0x0	Reserved
16	RW	0x0	eoc_int_en 1'b0: Disable eoc interrupt 1'b1: Enable eoc interrupt
15:13	RO	0x0	reserved
12	RW	0x0	lt_inten_src0 1'b0: Disable low temperature interrupt 1'b1: Enable low temperature interrupt
11:9	RO	0x0	reserved
8	RW	0x0	tshut_2cru_en_src0 1'b0: TSHUT output to CRU disabled. TSHUT output will always keep low. 1'b1: TSHUT output works
7:5	RO	0x0	Reserved
4	RW	0x0	tshut_2gpio_en_src0 1'b0: TSHUT output to GPIO disabled. TSHUT output will always keep low. 1'b1: TSHUT output works
3:1	RO	0x0	reserved
0	RW	0x0	ht_inten_src0 1'b0: Disable high temperature interrupt 1'b1: Enable high temperature interrupt

TSADC_INT_PD

Address: Operational Base + offset (0x000c)

Bit	Attr	Reset Value	Description
31:17	RO	0x0	Reserved
16	RW	0x0	eoc_int_pd This bit will be set to 1 when end-of-conversion. Set 0 to clear the interrupt
15:13	RO	0x0	Reserved
12	RW	0x0	lt_irq_src0 When ADC output is lower than COMP_INT_LOW, this bit will be valid, which means temperature is low, and the application should in charge of this. write 1 to it , this bit will be cleared
11:5	RO	0x0	Reserved
4	RW	0x0	tshut_o_src0 TSHUT output status When ADC output is bigger than COMP_SHUT, this bit will be set, which means temperature is VERY high, and the application should in charge of this. write 1 to it , this bit will be cleared

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
3:1	RO	0x0	Reserved
0	RW	0x0	ht_irq_src0 When ADC output is bigger than COMP_INT, this bit will be set, which means temperature is high, and the application should in charge of this. write 1 to it , this bit will be cleared

TSADC_DATA0

Address: Operational Base + offset (0x0020)

Bit	Attr	Reset Value	Description
31:12	RO	0x0	Reserved
11:0	RO	0x000	adc_data Output adc_q or 4096 - adc_q

TSADC_COMP0_INT

Address: Operational Base + offset (0x0030)

Bit	Attr	Reset Value	Description
31:12	RO	0x0	Reserved
11:0	RW	0x000	adc_comp_src0 High temperature threshold register. When ADC output is bigger than this filed, means the temperature is high, and ADC_HT_INT is set if interrupt is enabled

TSADC_COMP0_SHUT

Address: Operational Base + offset (0x0040)

Bit	Attr	Reset Value	Description
31:12	RO	0x0	Reserved
11:0	RW	0x000	adc_comp_shut_src0 High temperature shut threshold register. When ADC output is bigger than this filed, it will generate TSHUT signal to CRU or GPIO.

TSADC_HIGHT_INT_DEBOUNCE

Address: Operational Base + offset (0x0060)

Bit	Attr	Reset Value	Description
31:8	RO	0x0	Reserved
7:0	RW	0x03	debounce ADC controller will only generate interrupt when temperature / voltage is higher than COMP_INT for "debounce" times

TSADC_HIGHT_TSHUT_DEBOUNCE

Address: Operational Base + offset (0x0064)

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Bit	Attr	Reset Value	Description
31:8	RO	0x0	Reserved
7:0	RW	0x03	debounce ADC controller will only generate TSHUT when temperature / voltage is higher than COMP_SHUT for "debounce" times

TSADC_AUTO_PERIOD

Address: Operational Base + offset (0x0068)

Bit	Attr	Reset Value	Description
31:0	RW	0x00010000	auto_period When auto mode is enabled, this register controls the interleave between every two accessing of ADC

TSADC_AUTO_PERIOD_HT

Address: Operational Base + offset (0x006c)

Bit	Attr	Reset Value	Description
31:0	RW	0x00010000	auto_period This register controls the interleave between every two accessing of ADC after the temperature is higher than COMP_SHUT or COMP_INT

TSADC_COMP0_LOW_INT

Address: Operational Base + offset (0x0080)

Bit	Attr	Reset Value	Description
31:12	RO	0x0	Reserved
11:0	RW	0x000	adc_comp_src0 Low temperature shut threshold register. When ADC output is lower than this filed, ADC_LOW_INT is set if interrupt is enabled.

30.5 Application Notes

30.5.1 Single-sample conversion

- The start timing
  Before starting temperature sampling by TSADC, tsadc_tsen_pd need to be asserted by setting BUS_GRF_SOC_CON1[2] to low, and tsadc_ana_reg_2 need to be asserted by setting BUS_GRF_SOC_CON0[2] to high.

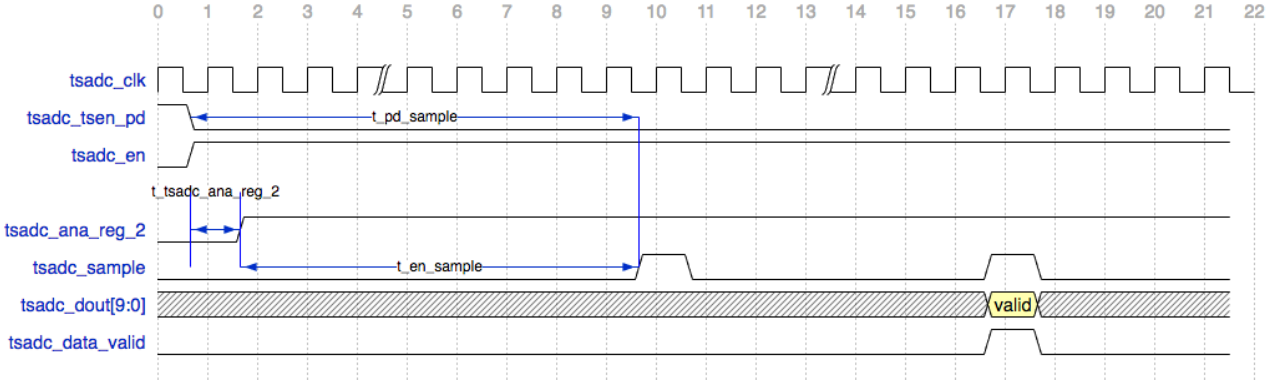


Fig. 30-2 The start flow to enable the Sensor and ADC

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

- Bypass mode(grf_sco_con1[3] = 1'b1)
  When the ADC bypass mode is enable(tsadc_dig_bypass = 1'b1), the ADC will cost 14 clock cycles to complete the conversion. The tsadc_dout will keep the valid data output unchanged untile the next clock cycles when the tsadc_sample is valid.

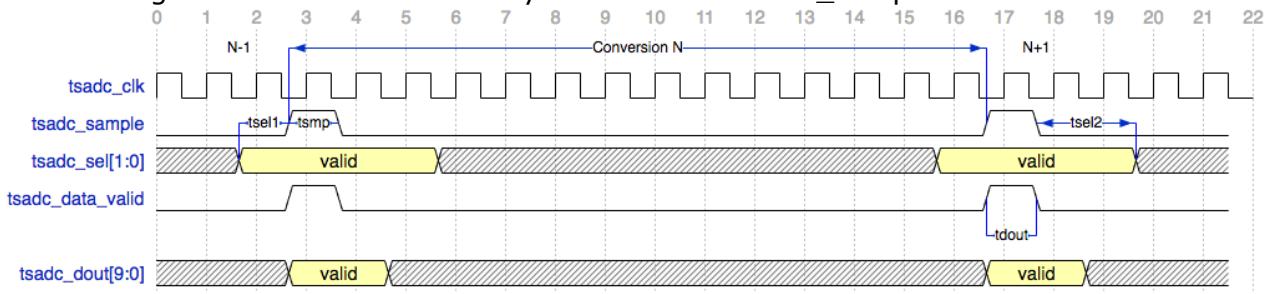


Fig. 30-3 TSADC timing diagram in bypass mode

- The Normal mode
  - tsadc_clk_sel = 1'b0(grf_sco_con1[4] = 1'b0)
    The ADC will cost 15 clock cycles to complete the conversion.The tsadc_dout will keep the valid data output unchanged untile the next two clock cycles when the tsadc_sample is valid.

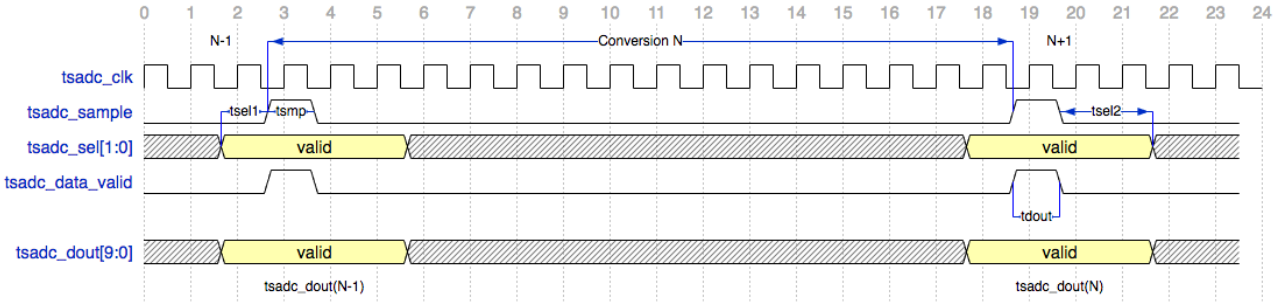


Fig. 30-4 TSADC timing diagram in normal mode with tsadc_clk_sel = 1'b0

  - tsadc_clk_sel = 1'b1(grf_sco_con1[4] = 1'b1)
    The ADC will cost 16 clock cycles to complete the conversion.The tsadc_dout will keep the valid data output unchanged untile the next three clock cycles when the tsadc_sample is valid.

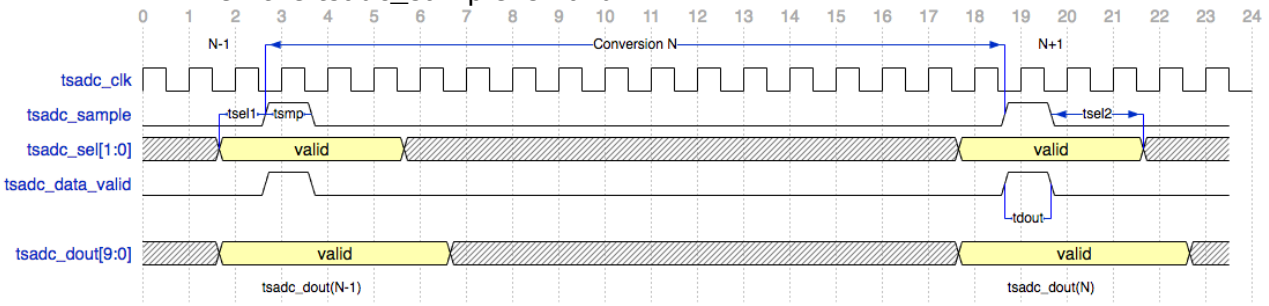


Fig. 30-5 TSADC timing diagram in normal mode with tsadc_clk_sel = 1'b1

30.5.2 Temperature-to-code mapping

Table 30-1 Temperature Code Mapping

Temperature/C	ADC Output Data		
	min	Typ	Max
-40		641	
-35		633	
-30		625	
-25		617	
-20		609	
-15		601	
-10		593	
-5		585	

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

Temperature/C	ADC Output Data		
	min	Typ	Max
0		577	
5		569	
10		561	
15		553	
20		545	
25		537	
30		529	
35		520	
40		512	
45		504	
50		496	
55		487	
60		479	
65		471	
70		463	
75		454	
80		446	
85		437	
90		429	
95		421	
100		412	
105		404	
110		395	
115		387	
120		378	
125		370	

Note: Table may be changed according to actually tested results.

30.5.3 User-Define Mode

- In user-define mode,tsadc_ana_reg_2 need to be asserted by setting BUS_GRF_SOC_CON0[2] to high,the tsadc_en and tsadc_sel[1:0] are generate by setting register TSADC_USER_CON, bit[3] and bit[2:0]. In order to ensure timing between tsadc_en and tsadc_sel[1:0], the tsadc_sel[1:0] must be set before the tsadc_en.
- In user-define mode, you can choose the method to control the tsadc_sample by setting bit[4] of TSADC_USER_CON. If set to 0, the tsadc_sample will be assert after "inter_pd_soc" cycles, which could be set by bit[11:6] of TSADC_USER_CON. And if start_mode was set 1, the tsadc_sample will be controlled by bit[5] of TSADC_USER_CON.
- Software can get the two channel temperature from TSADC_DATAn (n=0,1).

30.5.4 Automatic Mode

You can use the automatic mode with the following step:

- Set BUS_GRF_SOC_CON0[2] to high to assert tsadc_ana_reg_2.
- Set TSADC_AUTO_PERIOD，configure the interleave between every two accessing of TSADC in normal operation.
- Set TSADC_AUTO_PERIOD_HT. configure the interleave between every two accessing of TSADC after the temperature is higher than COMP_SHUT or COMP_INT.
- Set TSADC_COMPn_INT(n=0,1), configure the high temperature level, if tsadc output is smaller than the value, means the temperature is high, tsadc_int will be asserted.
- Set TSADC_COMPn_SHUT(n=0,1), configure the super high temperature level, if tsadc output is smaller than the value, means the temperature is too high, TSHUT will be asserted.
- Set TSADC_INT_EN, you can enable the high temperature interrupt for all channel; and

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.

you can also set TSHUT output to gpio to reset the whole chip; and you can set TSHUT output to cru to reset the whole chip.

- Set TSADC_HIGHT_INT_DEBOUNCE and TSADC_HIGHT_TSHUT_DEBOUNCE, if the temperature is higher than COMP_INT or COMP_SHUT for "debounce" times, TSADC controller will generate interrupt or TSHUT.
- Set TSADC_AUTO_CON, enable the TSADC controller.

Copyright 2019 © Fuzhou Rockchip Electronics Co., Ltd.