R329

User Manual

Revision 1.0

December 30, 2020

Revision History

Revision	Date	Author	Description
1.0	December 30, 2020	AWA0330	Initial release version

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Contents

Revision History i
Contents ii
1 About This Document 1
1.1 Purpose and Scope 1
1.2 Intended Audience 1
1.3 Symbol Conventions 1
1.3.1 Symbol Conventions 1
1.3.2 Table Content Conventions 1
1.3.3 Reset Value Conventions 1
1.3.4 Register Attributes 2
1.3.5 Numerical System 2
Contents 3
2 Product Description 4
2.1 Overview 4
2.1.1 Device Difference 4
2.2 Features 4
2.2.1 CPU Architecture 4
2.2.2 DSP Architecture 5
2.2.3 AIPU 5
2.2.4 Memory SubSystem 5
2.2.5 System Peripherals 7
2.2.6 Audio Interfaces 10
2.2.7 Security Engine 12
2.2.8 External Peripherals 13
2.2.9 Package 19
2.3 Block Diagram 19
Contents 22
Figures 25
Tables 27
3 System 28

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.1 Memory Mapping 28
3.2 CPUX Configuration 30
3.2.1 Overview 33
3.2.2 Functional Descriptions 33
3.2.3 Programming Guidelines 35
3.2.4 Cluster 0 Configuration Register List 35
3.2.5 Cluster 0 Configuration Register Description 36
3.2.6 CPU Subsystem Control Register List 44
3.2.7 CPU Subsystem Control Register Description 45
3.3 Clock Controller Unit (CCU) 50
3.3.1 Overview 50
3.3.2 Block Diagram 50
3.3.3 Functional Descriptions 51
3.3.4 Programming Guidelines 56
3.3.5 CPUX Domain Register List 62
3.3.6 CPUX Domain Register Description 63
3.3.7 CPUS Domain Register List 98
3.3.8 CPUS Domain Register Description 99
3.4 BROM System 124
3.4.1 Overview 124
3.4.2 Functional Descriptions 124
3.5 System Configuration 134
3.5.1 Overview 134
3.5.2 Register List 134
3.5.3 Register Description 134
3.6 Timer 137
3.6.1 Overview 137
3.6.2 Block Diagram 137
3.6.3 Functional Descriptions 138
3.6.4 Programming Guidelines 142
3.6.5 Register List 143
3.6.6 Register Description 144
3.7 High Speed Timer 156

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.7.1 Overview 156
3.7.2 Block Diagram 156
3.7.3 Functional Description 156
3.7.4 Programming Guidelines 158
3.7.5 Register List 160
3.7.6 Register Description 161
3.8 Generic Interrupt Controller (GIC) 168
3.9 Direct Memory Access Controller (DMAC) 174
3.9.1 Overview 174
3.9.2 Block Diagram 174
3.9.3 Functional Description 176
3.9.4 Programming Guidelines 187
3.9.5 Register List 188
3.9.6 Register Description 189
3.10 Thermal Sensor Controller (THS) 202
3.10.1 Overview 202
3.10.2 Block Diagram 202
3.10.3 Functional Descriptions 203
3.10.4 Programming Guidelines 204
3.10.5 Register List 206
3.10.6 Register Description 207
3.11 Peripheral System Interconnect (PSI) 212
3.11.1 Overview 212
3.11.2 Block Diagram 212
3.12 Message Box 213
3.12.1 Overview 213
3.12.2 Block Diagram 213
3.12.3 Functional Descriptions 214
3.12.4 Programming Guidelines 215
3.12.5 Register List 218
3.12.6 Register Description 219
3.13 Spinlock 228
3.13.1 Overview 228

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.13.2 Block Diagram 228
3.13.3 Functional Descriptions 228
3.13.4 Programming Guidelines 230
3.13.5 Register List 232
3.13.6 Register Description 233
3.14 RTC 237
3.14.1 Overview 237
3.14.2 RTC Clock Tree 238
3.14.3 Functional Descriptions 238
3.14.4 Programming Guidelines 241
3.14.5 Register List 243
3.14.6 Register Description 244
Contents 255
Figures 256
Tables 257
4 Memory 258
4.1 SDRAM Controller (DRAMC) 258
4.2 Nand Flash Controller (NDFC) 259
4.2.1 Overview 259
4.2.2 Block Diagram 260
4.2.3 Functional Descriptions 261
4.2.4 Programming Guidelines 274
4.2.5 Register List 276
4.2.6 Register Description 278
4.3 SD/MMC Host Controller (SMHC) 320
4.3.1 Overview 320
4.3.2 Block Diagram 320
4.3.3 Functional Descriptions 321
4.3.4 Programming Guidelines 335
4.3.5 Register List 340
4.3.6 Register Description 341
Contents 372
Figures 373

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Tables 374
5 Audio 375
5.1 I2S/PCM 375
5.1.1 Overview 375
5.1.2 Block Diagram 376
5.1.3 Functional Descriptions 377
5.1.4 Programming Guidelines 384
5.1.5 Register List 386
5.1.6 Register Description 387
5.2 DMIC 436
5.2.1 Overview 436
5.2.2 Block Diagram 436
5.2.3 Functional Descriptions 436
5.2.4 Register List 439
5.2.5 Register Description 439
5.3 OWA 454
5.3.1 Overview 454
5.3.2 Block Diagram 455
5.3.3 Functional Descriptions 455
5.3.4 Register List 460
5.3.5 Register Description 461
5.4 Audio Codec 477
5.4.1 Overview 477
5.4.2 Block Diagram 477
5.4.3 Functional Descriptions 478
5.4.4 Programming Guidelines 488
5.4.5 Register List 488
5.4.6 Register Description 492
Contents 554
6 AIPU 555
6.1 Overview 555
6.2 Block Diagram 555
6.3 Programming Guidelines 556

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

6.4 Register List 556
6.5 Register Description 557
6.5.1 0x0000 AIPU Control Register (Default Value: 0x0000_0000) 557
6.5.2 0x0004 AIPU Status Register (Default Value: 0x0007_0000) 558
6.5.3 0x0008 AIPU Start PC Register (Default Value: 0x0000_0000) 559
6.5.4 0x000C AIPU Interrupt PC Register (Default Value: 0x0000_0010) 560
6.5.5 0x0010 AIPU IPI Control Register (Default Value: 0x0000_0000) 560
6.5.6 0x0014 AIPU Data Address 0 Register (Default Value: 0x0000_0000) 560
6.5.7 0x0018 AIPU Data Address 1 Register (Default Value: 0x0000_0000) 560
6.5.8 0x003C AIPU Clock Control Register (Default Value: 0x0000_0000) 561
6.5.9 0x0400 AIPU AXI PMU Enable Register (Default Value: 0x0000_0000) 561
6.5.10 0x0404 AIPU AXI PMU Clear Register (Default Value: 0x0000_0000) 561
6.5.11 0x0408 AIPU AXI PMU Period Register (Default Value: 0x0000_0000) 562
6.5.12 0x040C AIPU AXI PMU Read Latency Register (Default Value: 0x0000_0000) 562
6.5.13 0x0410 AIPU AXI PMU Write Latency Register (Default Value: 0x0000_0000) 562
6.5.14 0x0414 AIPU AXI PMU Read Request Register (Default Value: 0x0000_0000) 562
6.5.15 0x0418 AIPU AXI PMU Write Request Register (Default Value: 0x0000_0000) 562
6.5.16 0x041C AIPU AXI PMU Read Bandwidth Register (Default Value: 0x0000_0000) 563
6.5.17 0x0420 AIPU AXI PMU Write Request Register (Default Value: 0x0000_0000) 563
Contents 564
Figures 568
Tables 572
7 Interfaces 574
7.1 TWI 574
7.1.1 Overview 574
7.1.2 Block Diagram 574
7.1.3 Functional Descriptions 576
7.1.4 Programming Guidelines 582
7.1.5 Register List 585
7.1.6 Register Description 586
7.2 UART 601
7.2.1 Overview 601
7.2.2 Block Diagram 601

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.2.3 Functional Descriptions 602
7.2.4 Programming Guidelines 608
7.2.5 Register List 611
7.2.6 Register Description 613
7.3 SPI 642
7.3.1 Overview 642
7.3.2 Block Diagram 642
7.3.3 Functional Descriptions 644
7.3.4 Programming Guidelines 652
7.3.5 Register List 655
7.3.6 Register Description 656
7.4 SPI_DBI 677
7.4.1 Overview 677
7.4.2 Block Diagram 677
7.4.3 Functional Descriptions 679
7.4.4 Programming Guidelines 699
7.4.5 Register List 707
7.4.6 Register Description 708
7.5 USB2.0 DRD 738
7.5.1 Overview 738
7.5.2 Block Diagram 739
7.5.3 Functional Descriptions 739
7.6 USB1.1 HOST 741
7.6.1 Overview 741
7.6.2 Block Diagram 741
7.6.3 Functional Descriptions 742
7.6.4 Register List 742
7.6.5 Register Description 744
7.7 GPIO 766
7.7.1 Overview 766
7.7.2 Block Diagram 766
7.7.3 Functional Descriptions 767
7.7.4 PB/PC/PF/PG/PH Register List 774

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5 PB/PC/PF/PG/PH Register Description 775
7.7.6 PL/PM/PN Register List 834
7.7.7 PL/PM/PN Register Description 835
7.8 GPADC 886
7.8.1 Overview 886
7.8.2 Block Diagram 886
7.8.3 Functional Descriptions 887
7.8.4 Programming Guidelines 889
7.8.5 Register List 890
7.8.6 Register Description 891
7.9 LRADC 902
7.9.1 Overview 902
7.9.2 Block Diagram 902
7.9.3 Functional Descriptions 903
7.9.4 Programming Guidelines 904
7.9.5 Register List 906
7.9.6 Register Description 906
7.10 PWM 911
7.10.1 Overview 911
7.10.2 Block Diagram 912
7.10.3 Functional Descriptions 912
7.10.4 Programming Guidelines 922
7.10.5 Register List 923
7.10.6 Register Description 925
7.11 LEDC 954
7.11.1 Overview 954
7.11.2 Block Diagram 954
7.11.3 Functional Descriptions 955
7.11.4 Programming Guidelines 959
7.11.5 Register List 964
7.11.6 Register Description 964
7.12 EMAC 975
7.12.1 Overview 975

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.12.2 Block Diagram 975
7.12.3 Functional Descriptions 976
7.12.4 Programming Guidelines 983
7.12.5 Register List 984
7.12.6 Register Description 985
7.13 CIR Receiver 1003
7.13.1 Overview 1003
7.13.2 Block Diagram 1003
7.13.3 Functional Descriptions 1004
7.13.4 Programming Guidelines 1008
7.13.5 Register List 1009
7.13.6 Register Description 1009
7.14 CIR Transmitter 1015
7.14.1 Overview 1015
7.14.2 Block Diagram 1015
7.14.3 Functional Descriptions 1016
7.14.4 Programming Guidelines 1019
7.14.5 Register List 1020
7.14.6 Register Description 1020
7.15 SCR 1028
7.15.1 Overview 1028
7.15.2 Block Diagram 1028
7.15.3 Functional Descriptions 1029
7.15.4 Register List 1031
7.15.5 Register Description 1032
Contents 1045
Figures 1046
8 Security System 1047
8.1 Crypto Engine 1047
8.1.1 Overview 1047
8.1.2 Block Diagram 1048
8.1.3 Functional Descriptions 1049
8.1.4 Programming Guidelines 1068

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

8.1.5 Register List 1071
8.1.6 Register Description 1072
8.2 Security ID 1076
Appendix: Glossary 1077

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

1 About This Document

1.1 Purpose and Scope

This document describes the features, logical structures, functions, operating modes, and related registers of each module about R329. For details about the interface timings and related parameters, the pins, pin usages, performance parameters, and package dimension, please refer to the *R329 Datasheet*.

1.2 Intended Audience

The document is intended for:

- Design and maintenance personnel for electronics
- Programmers in writing code or modifying the Allwinner provided code

1.3 Symbol Conventions

1.3.1 Symbol Conventions

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

Symbol	Description
WARNING	Indicates potential risk of injury or death exists if the instructions are not obeyed.
CAUTION	Indicates potential risk of equipment damage, data loss, performance degradation, or unexpected results exists if the instructions are not obeyed.
NOTE	Provides additional information to emphasize or supplement important points of the main text.

1.3.2 Table Content Conventions

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

Symbol	Description
/	The cell is blank.

1.3.3 Reset Value Conventions

In the register definition tables:

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

If other column value in a bit or multiple bits row is "/", that this bit or these multiple bits are unused.

If the default value of a bit or multiple bits is "UDF", that the default value is undefined.

1.3.4 Register Attributes

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

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

1.3.5 Numerical System

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

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

The expressions of addresses and data are described as follows.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Contents

2 Product Description 4
2.1 Overview 4
2.2 Features 4
2.2.1 CPU Architecture 4
2.2.2 DSP 5
2.2.3 Architecture 5
2.2.4 NPU 5
2.2.5 Memory SubSystem 5
2.2.6 System Peripherals 7
2.2.7 Audio Interfaces 10
2.2.8 Security Engine 12
2.2.9 External Peripherals 13
2.2.10 Package 19
2.3 Block Diagram 19

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

2 Product Description

2.1 Overview

Allwinner's R329 is a highly integrated dual-core SoC targeted for audio application markets. The R329 integrates a dual-core ARM Cortex™-A53 operating up to 1512 MHz, a dual-core HiFi4 DSP operating up to 400 MHz, and ARM Zhouyi Z1 AIPU 0.25T, which provides powerful and energy-efficient computing power for audio applications. An extensive set of audio interfaces such as audio codec (5 ADCs, 2 DACs), I2S/PCM, DMIC, one wire audio (OWA) are included for voice wake-up/recognition/record/playback applications on connected audio products. The outstanding audio subsystem supports all major high-resolution audio formats and works seamlessly with the CPU to accelerate multimedia algorithms and improve the user experience.

To reduce the BOM cost, a 256 MB or 128 MB DDR3 die is embedded for the R329. The R329 comes with extensive connectivity and interfaces, such as USB, SDIO, SPI, UART, TWI, EMAC, CIR, etc. And the R329 can connect with other different peripherals like WIFI and BT via SDIO and UART. Security functions are enabled and accelerated by hardware crypto engine, secure boot, and secure eFuse, etc.

2.1.1 Device Difference

The R329 is configured with different sets of features in different devices. Table 2-1 shows the feature differences across different devices.

Table 2-1 Device Feature Differences

Contents	R329-N4	R329-N3	R329-S3
SDRAM	SIP 256 MB DDR3	SIP 128 MB DDR3	SIP 128 MB DDR3
AIPU	Support	Support	No support
Package	LFBGA228 balls (G9, J7, K13: No ball), 12 mm x 14.5 mm body	LFBGA231 balls (G9, J7, K13: GND ball), 12 mm x 12 mm body	LFBGA231 balls (G9, J7, K13: GND ball), 12 mm x 12 mm body
PCB Footprint	LFBGA231 12 mm x 14.5 mm for PCB pin-to-pin design		

2.2 Features

2.2.1 CPU Architecture

- Dual-core ARM Cortex™-A53 Processor, frequency up to 1512 MHz
- Power-efficient ARM v8 architecture

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

- 64 and 32bit execution states for scalable high performance
- Trustzone technology supported
- Supports NEON Advanced Single Instruction Multiple Data (SIMD) instruction for acceleration of media and signal processing functions
- VFPv4 Floating Point Unit
- 32 KB L1 I-cache and 32 KB L1 D-cache for per core
- 256 KB L2 cache shared

2.2.2 DSP Architecture

- Dual-core HiFi4
- Up to 400 MHz
- 32 KB L1 I-cache and 32 KB L1 D-cache for per core
- 2 MB SRAM

2.2.3 AIPU

- ARM China Zhouyi Z1 AIPU, 1TEC+128mac FF
- Up to 0.25TOPS@600MHz

NOTE

R329-S3 does not support AIPU.

2.2.4 Memory SubSystem

2.2.4.1 Boot ROM (BROM)

- On-chip memory
- Supports system boot from the following devices:
  - SD card
  - eMMC

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

- NAND Flash
- SPI NOR Flash
- SPI NAND Flash

- Supports mandatory upgrade process through USB
- Supports mandatory upgrade process through SD card (This requires that the flash on the device has no valid firmware)
- Supports the GPIO pin to select the boot media type
- Supports eFuse to select the boot media type
- Supports normal booting and secure booting
- Secure BROM loads only certified firmware
- Secure BROM ensures that the secure boot is a trusted environment

2.2.4.2 SDRAM

- Embedded with 256 MB DDR3 (only for R329-N4)
- Embedded with 128 MB DDR3 (only for R329-N3 and R329-S3)
- Supports clock frequency up to 774 MHz for DDR3

2.2.4.3 NAND Flash

- Compliant with ONFI 2.0 and Toggle 2.0
- Up to 80-bit ECC per 1024 bytes
- Supports 1K/2K/4K/8K/16K/32K bytes page size
- Up to 8-bit data bus width
- Supports SLC/MLC/TLC flash and EF-NAND
- Supports SDR/Toggle DDR/ONFI DDR NAND interface

2.2.4.4 SMHC

- Two SD/MMC host controller (SMHC) interfaces
- The SMHC0 controls the devices that comply with the following protocols:

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

- Secure Digital Memory (SD mem-version 3.0)
- Multimedia Card (eMMC-version 5.1)

- The SMHC1 controls the device that complies with the protocol Secure Digital I/O (SDIO-version 3.0)
- Maximum performance:
  - SDR mode: 150 MHz@1.8 V IO pad
  - DDR mode: 50 MHz@1.8 V IO pad
  - DDR mode: 50 MHz@3.3 V IO pad
- Supports 1-bit or 4-bit data width
- Supports hardware CRC generation and error detection
- Supports block size of 1 to 65535 bytes

2.2.5 System Peripherals

2.2.5.1 Timer

- The timer module implements the timing and counting functions, which includes timer0, timer1, watchdog, and audio video synchronization (AVS)
- The timer0/timer1 is a 32-bit down counter. The timer0 and timer1 are completely consistent
- The watchdog is used to transmit a reset signal to reset the entire system when an exception occurs in the system
- The AVS is used to synchronize the audio and video. The AVS sub-block includes AVS0 and AVS1, which are completely consistent

2.2.5.2 High Speed Timer (HSTimer)

- The HSTimer module consists of HSTimer0 and HSTimer1. HSTimer0 and HSTimer1 are down counters that implement timing and counting functions. They are completely consistent.
- Configurable 56-bit down timer
- Supports 5 prescale factors
- The clock source is synchronized with AHB1 clock, much more accurate than other timers
- Supports 2 working modes: periodic mode and single counting mode
- Generates an interrupt when the count is decreased to 0

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

2.2.5.3 GIC

- Supports 16 Software Generated Interrupts (SGIs), 16 Private Peripheral Interrupts (PPIs), and 160 Shared Peripheral Interrupts (SPIs)
- Enabling, disabling, and generating processor interrupts from hardware interrupt
- Interrupt masking and prioritization
- Uniprocessor and multiprocessor environments
- ARM architecture security extensions
- ARM architecture virtualization extensions
- Wakeup events in power-management environments

2.2.5.4 DMAC

- Up to 8-ch DMA in CPUX domain and 8-ch DMA in CPUS domain
- Interrupt generated for each DMA channel
- Flexible data width of 8/16/32/64-bit
- Supports linear and IO address modes
- Supports data transfer types with memory-to-memory, memory-to-peripheral, peripheral-to-memory, peripheral-to-peripheral
- Supports transfer with linked list
- DRQ response includes waiting mode and handshake mode
- DMA channel supports pause function

2.2.5.5 Clock Controller Unit (CCU)

- 4 PLLs
- One on-chip RC oscillator
- Supports one external 24 MHz DCXO and one external 32.768 kHz oscillator
- Supports clock configuration and clock generation for corresponding modules
- Supports software-controlled clock gating and software-controlled reset for corresponding modules

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

2.2.5.6 Thermal Sensor Controller (THS)

- One thermal sensor located in CPU
- Temperature accuracy: ±3°C from 0°C to +100°C, ±5°C from -25°C to +125°C
- Supports over-temperature protection interrupt and over-temperature alarm interrupt
- Averaging filter for thermal sensor reading
- Supports over-temperature protection interrupt and over-temperature alarm interrupt

2.2.5.7 CPU Configuration

- CPU reset system: core reset, debug circuit reset, and other functional modules reset
- CPU related control: interface control, CP15 control, and power-on/off control
- CPU status check: idle status, SMP status, interrupt status, and so on
- Including CPU debug control and status register

2.2.5.8 LDO Power

- Integrated 2 LDOs (LDOA, LDOB)
- LDOA: 1.8 V power output, LDOB: 1.35 V/1.5 V/1.8 V power output
- LDOA for IO and analog module, LDOB for SDRAM
- Input voltage is 2.4 V to 3.6 V

2.2.5.9 RTC

- Implements time counter and timing wakeup
- Provides a 16-bit counter for counting day, 5-bit counter for counting hour, 6-bit counter for counting minute, 6-bit counter for counting second
- External connect a 32.768 kHz low-frequency oscillator for count clock
- Timer frequency is 1 kHz
- Configurable initial value by software anytime

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

- Supports timing alarm, and generates interrupt and wakeup the external devices
- Supports fanout function of internal 32K clock
- 8 general purpose registers for storing power-off information

2.2.5.10 Reset

- Integrated internal reset
- Reset R329 or other IC

2.2.6 Audio Interfaces

2.2.6.1 Audio Codec

- Two audio digital-to-analog (DAC) channels
  - Supports 16-bit and 20-bit sample resolution
  - 8 kHz to 192 kHz DAC sample rate
  - 100 ± 2dB SNR@A-weight, -85 ± 3dB THD+N
- Two audio outputs:
  - Two differential speaker outputs: SPKLP/N, SPKRP/N
- Five audio analog-to-digital (ADC) channels
  - Supports 16-bit and 20-bit sample resolution
  - 8 kHz to 48 kHz ADC sample rate
  - 95 ± 3dB SNR@A-weight, -80 ± 3dB THD+N
- Five audio inputs:
  - Five differential microphone inputs: MICIN1P/1N, MICIN2P/2N, MICIN3P/3N, MICIN4P/4N, MICIN5P/5N
- One low-noise analog microphone bias output
- Supports Dynamic Range Controller adjusting the DAC playback and ADC recording
- One 128x20-bits FIFO for DAC data transmit, one 256x20-bits FIFO for ADC data receive
- Programmable FIFO thresholds
- Supports interrupts and DMA

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

2.2.6.2 I2S/PCM

- Three I2S/PCM external interfaces (I2S0, I2S1, S-I2S0) for connecting external power amplifier and MIC ADC
- I2S0 supports 4 data channels; I2S1 and S-I2S0 support 2 data channels
- S-I2S0 supports asynchronous sample rate conversion (ASRC)
- Compliant with standard Philips Inter-IC sound (I2S) bus specification
- Compliant with Left-justified, Right-justified, PCM mode, and Time Division Multiplexing (TDM) format
- TDM mode supports maximum 16 channels output and 16 channels input
- Supports full-duplex synchronous work mode
- Supports Master/Slave mode
- Clock up to 49.152 MHz data output of I2S/PCM in Master mode when the IO PAD and peripheral I2S/PCM satisfy timing parameters
- Clock up to 12.288 MHz data input of I2S/PCM in Master mode when the IO PAD and peripheral I2S/PCM satisfy timing parameters
- Supports adjustable audio sample resolution from 8-bit to 32-bit
- Sample rate from 8 kHz to 384 kHz (channels =2)
- Supports 8-bit u-law and 8-bit A-law companded sample
- One 128 depth x 32-bit width TXFIFO for data transmit, one 64 depth x 32-bit width RXFIFO for data receive
- Supports programmable PCM frame width: 1 BCLK width (short frame) and 2 BCLKs width (long frame)
- Programmable FIFO thresholds

2.2.6.3 DMIC

- Supports maximum 8 digital PDM microphones
- Supports sample rate from 8 kHz to 48 kHz

2.2.6.4 One Wire Audio (OWA)

- One OWA TX and one OWA RX

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

- Compliance with S/PDIF Interface
- IEC-60958 transmitter and receiver functionality
- Supports 16-bit, 20-bit and 24-bit data formats
- Supports channel status insertion for the transmitter
- Supports channel status capture for the receiver
- Supports hardware parity on TX/RX
  - Hardware parity generation on the transmitter
  - Hardware parity checking on the receiver
- Transmit and receive data FIFOs
  - One 128×24 bits TXFIFO and 64x24 bits RXFIFO for audio data transfer
  - Programmable FIFO thresholds
- Supports interrupts and DMA

2.2.7 Security Engine

2.2.7.1 Crypto Engine (CE)

- Supports Symmetrical algorithm for encryption and decryption: AES, DES, TDES
  - Supports ECB, CBC, CTS, CTR, CFB, OFB mode for AES
  - Supports 128/192/256-bit key for AES
  - Supports ECB, CBC, CTR mode for DES/TDES
- Supports Hash algorithm for tamper proofing: MD5, SHA, HMAC
  - Supports SHA1, SHA224, SHA256, SHA384, SHA512 for SHA
  - Supports HMAC-SHA1, HMAC-SHA256 for HMAC
  - Supports multi-package mode for MD5/SHA1/SHA224/SHA256/SHA384/SHA512
- Supports Asymmetrical algorithm for signature verification: RSA
  - RSA supports 512/1024/2048-bit width
- Supports 160-bit hardware PRNG with 175-bit seed
- Supports 256-bit hardware TRNG
- Internal DMA controller for data transfer with memory

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

- Supports secure and non-secure interfaces respectively

2.2.7.2 Security ID (SID)

- Supports 1 Kbit eFuse for chip ID and security application
- The eFuse has secure zone and non-secure zone
- Backup eFuse information by using SID_SRAM
- Burning the key to the SID
- Reading the key use status in the SID
- Loading the key to the CE

2.2.7.3 Secure Memory Control (SMC)

- The SMC is always secure, only secure CPU can access the SMC
- Sets secure area of DRAM
- Sets secure property that Master accesses to DRAM

2.2.7.4 Secure Peripherals Control (SPC)

- The SPC is always secure, only secure CPU can access the SPC
- Sets secure property of peripherals

2.2.8 External Peripherals

2.2.8.1 USB

- One USB 2.0 DRD (USB0), with integrated USB 2.0 analog PHY
  - Compatible with USB 2.0 Specification
  - Supports High-Speed (HS, 480 Mbit/s), Full-Speed (FS, 12 Mbit/s), and Low-Speed (LS, 1.5 Mbit/s) in host mode
  - Supports High-Speed (HS, 480 Mbit/s), Full-Speed (FS, 12 Mbit/s) in device mode
  - Compatible with Enhanced Host Controller Interface (EHCI) Specification, Version 1.0, and the Open

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Host Controller Interface (OHCI) Specification, Version 1.0a for host mode

- Supports bi-directional endpoint0 (EP0) for Control transfer
- Up to 8 user-configurable endpoints (EP1 +, EP1 -, EP2 +, EP2 -, EP3 +, EP3 -, EP4 +, EP4 -) for Bulk, Isochronous and Interrupt bi-directional transfers
- Supports (4 KB + 64 Bytes) FIFO for EPs (including EP0)
- Supports point-to-point and point-to-multipoint transfer in both host and peripheral modes

- One USB 1.1 HOST (USB1), with integrated USB 1.1 analog PHY
  - Compatible with USB 1.1 Specification and Open Host Controller Interface (OHCI) Specification, Version 1.0a.
  - Supports Full-Speed (FS, 12 Mbit/s) and Low-Speed (LS, 1.5 Mbit/s) device

2.2.8.2 EMAC

- One EMAC interface for connecting external Ethernet PHY
- 10/100/1000 Mbps Ethernet port with RGMII and RMII interfaces
- Compliant with IEEE 802.3-2002 standard
- Supports both full-duplex and half-duplex operation
- Provides the management data input/output (MDIO) interface for PHY device configuration and management with configurable clock frequencies
- Programmable frame length to support Standard or Jumbo Ethernet frames with sizes up to 16 KB
- Supports a variety of flexible address filtering modes
- Separate 32-bit status returned for transmission and reception packets
- Optimization for packet-oriented DMA transfers with frame delimiters
- Supports linked-list descriptor list structure
- Descriptor architecture, allowing large blocks of data transfer with minimum CPU intervention; each descriptor can transfer up to 4 KB of data
- Comprehensive status reporting for normal operation and transfers with errors
- 4 KB TXFIFO for transmission packets and 16 KB RXFIFO for reception packets
- Programmable interrupt options for different operational conditions

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

2.2.8.3 UART

- Up to 5 UART controllers (UART0, UART1, UART2, UART3, S-UART0)
- UART0, S-UART0: 2-wire; UART1, UART2, UART3: 4-wire
- Compatible with industry-standard 16450/16550 UARTs
- Supports IrDA-compatible slow infrared (SIR) format
- Supports RS-485/9-bit mode
- Capable of speed up to 4 Mbit/s
- Supports 5 to 8 data bits and 1/1.5/2 stop bits
- Programmable parity (even, odd, or no parity)
- Supports DMA controller interface
- Supports software/hardware flow control
- Two separate FIFOs: one is RX FIFO, and the other is TX FIFO
  - Each of them is 64 bytes (For UART0 and S-UART0)
  - Each of them is 256 bytes (For UART1, UART2 and UART3)

2.2.8.4 Two Wire Interface (TWI)

- Up to 3 TWI controllers (TWI0, TWI1, S-TWI0)
- Software-programmable for slave or master
- Supports Standard mode (up to 100 kbit/s) and Fast mode (up to 400 kbit/s)
- Supports 7-bit and 10-bit device addressing modes
- Supports repeated START signal
- Performs arbitration and clock synchronization
- Own address and General Call address detection
- Interrupt on address detection
- Byte-by-byte data transfer
- Allows operation from a wide range of input clock frequency
- The TWI controller includes one TWI engine and one TWI driver. And the TWI driver supports packet transmission and DMA mode when TWI works in master mode

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

2.2.8.5 SPI and SPI_DBI

- Up to 2 SPI controllers (SPI0, SPI1)
- The SPI0 only supports SPI mode; The SPI1 supports SPI mode and display bus interface (DBI) mode
- SPI mode:
  - Full-duplex synchronous serial interface
  - Master/slave configurable
  - Mode0 to Mode3 are supported for both transmit and receive operations
  - 8-bit wide by 64-entry FIFO for both transmit and receive data
  - Polarity and phase of the Chip Select (SPI-CS) and SPI Clock (SPI-CLK) are configurable
  - Supports 3-wire/4-wire SPI
  - Supports programmable serial data frame length: 1-bit to 32-bit
  - Supports Standard SPI, Dual-Output/Dual-Input SPI, Dual IO SPI, Quad-Output/Quad-Input SPI
- DBI mode:
  - Supports DBI Type C 3 Line/4 Line Interface Mode
  - Supports 2 Data Lane Interface Mode
  - Supports RGB111/444/565/666/888 video format
  - Maximum resolution of RGB666 240 x 320@30Hz with single data lane
  - Maximum resolution of RGB888 240 x 320@60Hz or 320 x 480@30Hz with dual data lane
  - Supports Tearing effect
  - Supports software flexible control video frame rate

2.2.8.6 CIR Receiver (CIR_RX)

- Up to 2 CIR_RX interfaces (IR-RX, S-IR-RX)
- Full physical layer implementation
- Supports NEC format infra data
- Supports CIR for remote control or wireless keyboard
- 64x8 bits FIFO for data buffer
- Sample clock up to 1 MHz

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

2.2.8.7 CIR Transmitter (CIR_TX)

- 1 CIR_TX interface (IR-TX)
- Supports arbitrary wave generator
- Configurable carrier frequency
- Supports handshake mode and waiting mode of DMA
- 128 bytes FIFO for data buffer

2.2.8.8 PWM

- 15 PWM channels (PWM[8:0], S-PWM[5:0])
- Supports pulse, cycle and complementary pair output
- Supports capture input
- Programming deadzone output
- Build-in the programmable dead-time generatorand the controllable dead-time
- Three kinds of output waveform: continuous waveform, pulse waveform and complementary pair
- Output frequency range: 0 to 24 MHz or 100 MHz
- Various duty-cycle: 0% to 100%
- Minimum resolution: 1/65536
- Interrupt generation of PWM output and capture input

2.2.8.9 Low Rate ADC (LRADC)

- One LRADC input channel
- 6-bit sampling resolution and 5-bit precision
- Sample rate up to 2 kHz
- Supports hold Key and general Key
- Supports normal, continuous and single working mode
- Power supply voltage: AVCC, power reference voltage: 0.75*AVCC, analog input and detected voltage range: 0 to LEVELB

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

2.2.8.10 General Purpose ADC (GPADC)

- 4-ch successive approximation register (SAR) analog-to-digital converter (ADC)
- 12-bit sampling resolution and 8-bit precision
- 64 FIFO depth of data register
- Power reference voltage: AVCC, analog input voltage range: 0 to AVCC
- Maximum sampling frequency up to 1 MHz
- Supports three operation modes: single conversion mode, continuous conversion mode, burst conversion mode

2.2.8.11 Smart Card Reader (SCR)

- Supports the ISO/IEC 7816-3:1997(E) and EMV2000 (4.0) Specifications
- Performs functions needed for complete smart card sessions, including:
  - Card activation and deactivation
  - Cold/warm reset
  - Answer to Reset (ATR) response reception
  - Data transfers to and from the card
- Supports adjustable clock rate and bit rate
- Configurable automatic byte repetition
- Supports commonly used communication protocols:
  - T=0 for asynchronous half-duplex character transmission
  - T=1 for asynchronous half-duplex block transmission
- Supports FIFOs for receive and transmit buffers (up to 128 characters) with threshold
- Supports configurable timing functions:
  - Smart card activation time
  - Smart card reset time
  - Guard time
  - Timeout timers

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

- Supports synchronous and any other non-ISO 7816 and non-EMV cards

2.2.8.12 LEDC

- LEDC is used to control the external intelligent control LED lamp
- Configurable LED output high/low level width
- Configurable LED reset time
- LEDC data supports DMA configuration mode and CPU configuration mode
- Maximum 1024 LEDs serial connect
- LED data transfer rate up to 800 kbit/s

2.2.9 Package

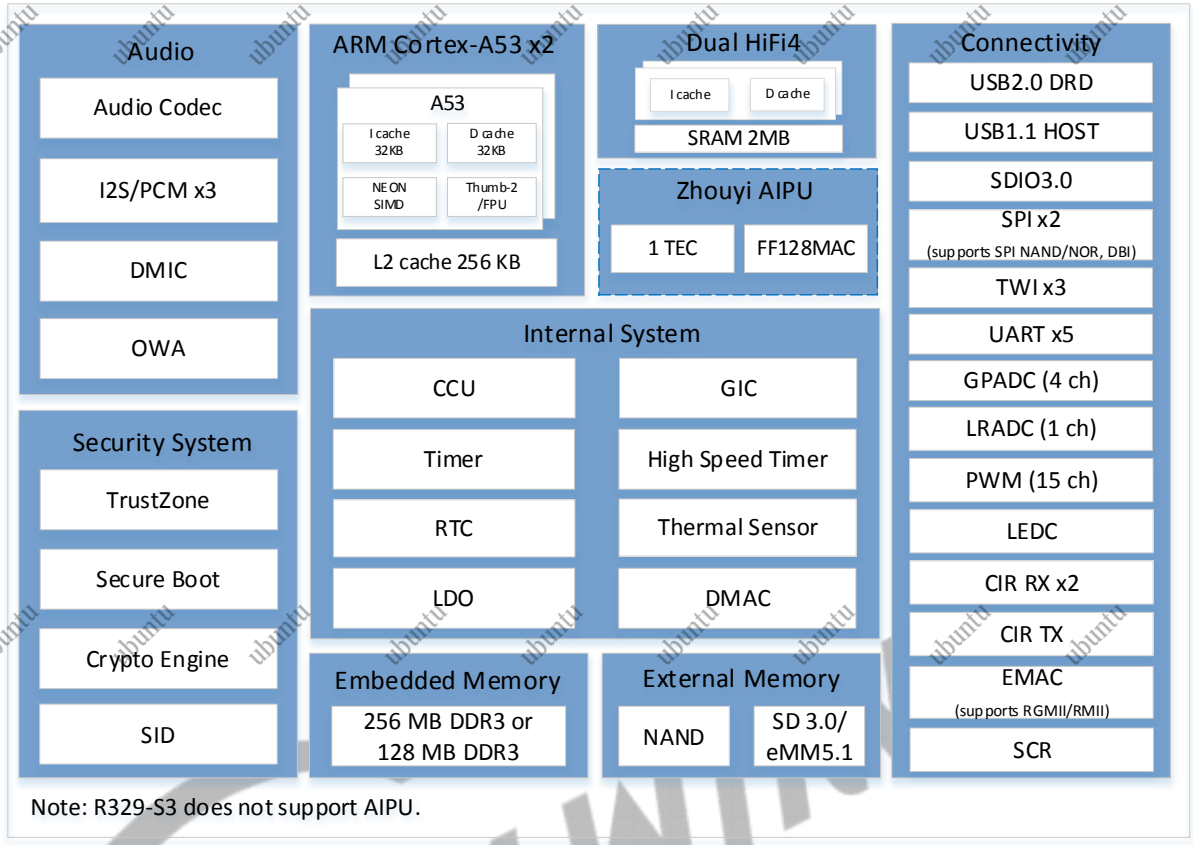
- LFBGA231 balls, 0.65 mm ball pitch, 0.35 mm ball size, 12 mm x 12 mm body (only for R329-N3 and R329-S3)
- LFBGA228 balls, 0.65 mm ball pitch, 0.35 mm ball size, 12 mm x 14.5 mm body (only for R329-N4)

2.3 Block Diagram

Figure 2-1 shows the system block diagram of the R329.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

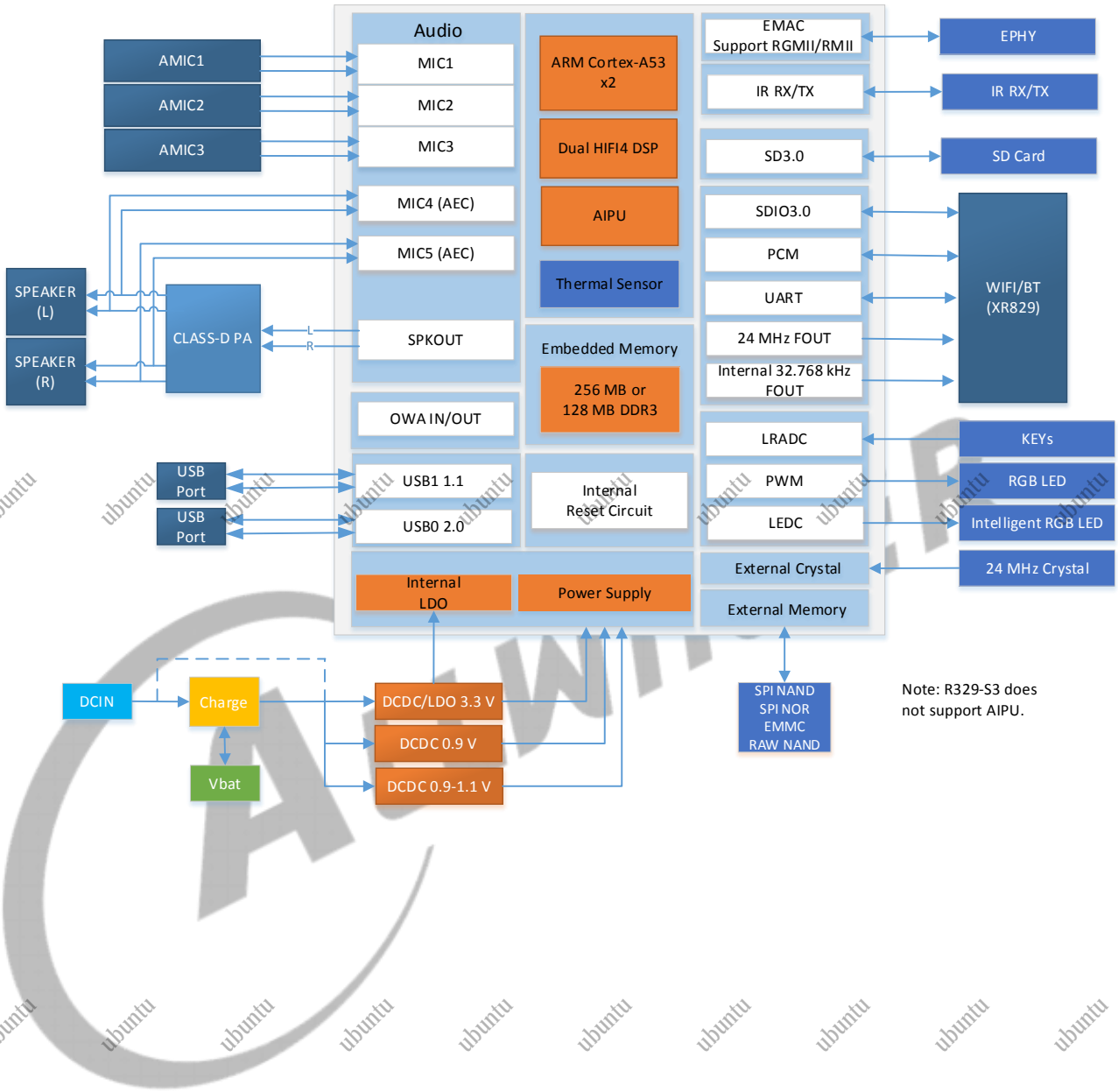
Figure 2-1 R329 System Block Diagram



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 2-2 shows the intelligent speaker solution of the R329.

Figure 2-2 R329 Intelligent Speaker Solution



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Contents

3 System 28

3.1 Memory Mapping 28

3.2 CPUX Configuration 30

3.2.1 Overview 33

3.2.2 Functional Descriptions 33

3.2.3 Programming Guidelines 35

3.2.4 Cluster 0 Configuration Register List 35

3.2.5 Cluster 0 Configuration Register Description 36

3.2.6 CPU Subsystem Control Register List 44

3.2.7 CPU Subsystem Control Register Description 45

3.3 Clock Controller Unit (CCU) 50

3.3.1 Overview 50

3.3.2 Block Diagram 50

3.3.3 Functional Descriptions 51

3.3.4 Programming Guidelines 56

3.3.5 CPUX Domain Register List 62

3.3.6 CPUX Domain Register Description 63

3.3.7 CPUS Domain Register List 98

3.3.8 CPUS Domain Register Description 99

3.4 BROM System 124

3.4.1 Overview 124

3.4.2 Functional Descriptions 124

3.5 System Configuration 134

3.5.1 Overview 134

3.5.2 Register List 134

3.5.3 Register Description 134

3.6 Timer 137

3.6.1 Overview 137

3.6.2 Block Diagram 137

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.7 High Speed Timer 156
3.7.1 Overview 156
3.7.2 Block Diagram 156
3.7.3 Functional Description 156
3.7.4 Programming Guidelines 158
3.7.5 Register List 160
3.7.6 Register Description 161
3.8 Generic Interrupt Controller (GIC) 168
3.9 Direct Memory Access Controller (DMAC) 174
3.9.1 Overview 174
3.9.2 Block Diagram 174
3.9.3 Functional Description 176
3.9.4 Programming Guidelines 187
3.9.5 Register List 188
3.9.6 Register Description 189
3.10 Thermal Sensor Controller (THS) 202
3.10.1 Overview 202
3.10.2 Block Diagram 202
3.10.3 Functional Descriptions 203
3.10.4 Programming Guidelines 204
3.10.5 Register List 206
3.10.6 Register Description 207
3.11 Peripheral System Interconnect (PSI) 212
3.11.1 Overview 212
3.11.2 Block Diagram 212
3.12 Message Box 213
3.12.1 Overview 213
3.12.2 Block Diagram 213
3.12.3 Functional Descriptions 214
3.12.4 Programming Guidelines 215
3.12.5 Register List 218
3.12.6 Register Description 219
3.13 Spinlock 228

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.13.1 Overview 228
3.13.2 Block Diagram 228
3.13.3 Functional Descriptions 228
3.13.4 Programming Guidelines 230
3.13.5 Register List 232
3.13.6 Register Description 233
3.14 RTC 237
3.14.1 Overview 237
3.14.2 RTC Clock Tree 238
3.14.3 Functional Descriptions 238
3.14.4 Programming Guidelines 241
3.14.5 Register List 243
3.14.6 Register Description 244

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figures

Figure 3-1 Default Power Domain of Cluster0 35
Figure 3-2 Functional Block Diagram of the CCU 50
Figure 3-3 System Bus Tree 52
Figure 3-4 Module Clock Generation - Part 1 53
Figure 3-5 Module Clock Generation - Part 2 54
Figure 3-6 PLL Distribution 55
Figure 3-7 Boot Process in Normal BROM Mode 128
Figure 3-8 Authentication Process in Secure BROM Mode 129
Figure 3-9 Mandatory Upgrade Process 130
Figure 3-10 USB FEL Process 132
Figure 3-11 Timer Block Diagram 138
Figure 3-12 Block Diagram for the Timer 138
Figure 3-13 Block Diagram for the Watchdog 140
Figure 3-14 Block Diagram for the AVS 141
Figure 3-15 HSTimer Block Diagram 156
Figure 3-16 Typical Application for HSTimer 157
Figure 3-17 HSTimer Initialization Process 159
Figure 3-18 DMAC Block Diagram 175
Figure 3-19 DMAC Typical Application Diagram 176
Figure 3-20 DMA Descriptor 179
Figure 3-21 DMA Chain Transfer 181
Figure 3-22 Workflow of the DMAC Handshake Mode 184
Figure 3-23 DMAC Transfer Process 186
Figure 3-24 THS Block Diagram 202
Figure 3-25 Thermal Sensor Timing Requirement 203
Figure 3-26 Thermal Sensor Controller Interrupt Source 203
Figure 3-27 THS Initial Process 204
Figure 3-28 PSI Block Diagram 212
Figure 3-29 Message Box Block Diagram 213
Figure 3-30 Message Box Typical Application Chart 215

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 3-31 User1 and User0 Working Process....217
Figure 3-32 Spinlock Block Diagram....228
Figure 3-33 Spinlock Typical Application Diagram....229
Figure 3-34 Spinlock State Machine....229
Figure 3-35 CPU0 and CPUS Taking/Freeing Spinlock0 Process....231
Figure 3-36 RTC Clock Tree....238
Figure 3-37 RTC Application Diagram....239
Figure 3-38 RTC Counter....240
Figure 3-39 RTC 1KHz Counter Step Structure....240

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Tables

Table 3-1 Reset Signal Description 34
Table 3-2 Power Domain of Modules in the CPU Subsystem 34
Table 3-3 PLL Features 55
Table 3-4 Recommended Values for Configuration Factors of PLL_AUDIO0 58
Table 3-5 BOOT_MODE Setting 125
Table 3-6 GPIO Boot Select Configuration 125
Table 3-7 Groups of eFuse_Boot_Select 126
Table 3-8 eFuse Boot Select Setting 126
Table 3-9 Fast Boot Select Setting 133
Table 3-10 Interrupt Sources 168
Table 3-11 DMAC Sub-blocks 175
Table 3-12 DMA DRQ Type 177
Table 3-13 Source/Destination Address Distribution 180
Table 3-14 THS Information in the SID 205
Table 3-15 RTC External Signals 238
Table 3-16 RTC Counter Changing Range 240

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3 System

3.1 Memory Mapping

Module	Address (It is for Cluster CPU)	Size (Bytes)
N-BROM	0x0000 0000—0x0000 DFFF	56K
S-BROM	0x0000 0000—0x0000 BFFF	48K
SRAM A2	0x0010 0000---0x002C FFFF	1856K (29*64K) 0x0010 0000---0x0011 FFFF (2*64K) 0x0012 0000---0x002C FFFF (27*64K)
DSP0 IRAM	0x002D 0000---0x002E FFFF	128K
DSP0 DRAM0	0x002F 0000---0x002F FFFF	64K
DSP0 DRAM1	0x0030 0000---0x0030 FFFF	64K
CPUX Resources		
TIMER	0x0200 0000---0x0200 03FF	1K
GPIO	0x0200 0400---0x0200 07FF	1K
SPC	0x0200 0800---0x0200 0BFF	1K
PWM	0x0200 0C00---0x0200 0FFF	1K
CCU	0x0200 1000---0x0200 1FFF	4K
LEDC	0x0200 2800---0x0200 2BFF	1K
OWA	0x0200 2C00---0x0200 2FFF	1K
IR_TX	0x0200 3000 – 0x0200 33FF	1K
IR_RX	0x0200 3400---0x0200 37FF	1K
I2S0	0x0200 4000---0x0200 4FFF	4K
I2S1	0x0200 5000---0x0200 5FFF	4K
/	/	/
UART0	0x0250 0000---0x0250 03FF	1K
UART1	0x0250 0400---0x0250 07FF	1K
UART2	0x0250 0800---0x0250 0BFF	1K
UART3	0x0250 0C00---0x0250 0FFF	1K
TWI0	0x0250 2000---0x0250 23FF	1K
TWI1	0x0250 2400---0x0250 27FF	1K
SCR	0x0250 5000---0x0250 53FF	1K
/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Module	Address (It is for Cluster CPU)	Size (Bytes)
SYS_CFG	0x0300 0000---0x0300 0FFF	4K
DMAC	0x0300 2000---0x0300 2FFF	4K
MSGBOX0	0x0300 3000---0x0300 3FFF	4K
SPINLOCK	0x0300 4000---0x0300 4FFF	4K
HSTIMER	0x0300 5000---0x0300 5FFF	4K
SID	0x0300 6000---0x0300 6FFF	4K
SMC	0x0300 7000---0x0300 7FFF	4K
MSGBOX1	0x0300 8000---0x0300 8FFF	4K
DCU	0x0301 0000---0x0301 FFFF	64K
GIC	0x0302 0000---0x0302 FFFF	64K
CE_NS	0x0304 0000---0x0304 07FF	2K
CE_S	0x0304 0800---0x0304 0FFF	2K
AIPU	0x0305 0000---0x0305 07FF	2K
MSI_CTRL	0x0310 2000---0x0310 2FFF	4K
DRAM_CTRL & PHY	0x0310 3000---0x0310 5FFF	12K
/	/	/
NAND	0x0401 1000---0x0401 1FFF	4K
SMHC0	0x0402 0000---0x0402 0FFF	4K
SMHC1	0x0402 1000---0x0402 1FFF	4K
SPI0	0x0402 5000---0x0402 5FFF	4K
SPI1 (SPI_DBI)	0x0402 6000---0x0402 6FFF	4K
USB0 (USB2.0_DRD)	0x0410 0000---0x041F FFFF	1M
USB1 (USB1.1_HOST)	0x0420 1000---0x042F FFFF	1M
EMAC	0x0450 0000---0x0450 FFFF	64K
/	/	/
CPU_SUBSYS_CFG	0x0810 0000---0x0810 03FF	1K
TIMESTAMP_STU	0x0811 0000---0x0811 0FFF	4K
TIMESTAMP_CTRL	0x0812 0000---0x0812 0FFF	4K
IDC	0x0813 0000---0x0813 0FFF	4K
C0_CPUX_CFG	0x0901 0000---0x0901 03FF	1K
C0_CPUX_MBIST	0x0902 0000---0x0902 0FFF	4K
CPUS Resources		
R_CPUS_CFG	0x0700 0400---0x0700 0BFF	2K
R_PPU	0x0700 1000---0x0700 13FF	1K

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Module	Address (It is for Cluster CPU)	Size (Bytes)
R_PRCM	0x0701 0000---0x0701 FFFF	64K
R_TIMER	0x0702 0000 – 0x0702 03FF	1K
R_WDG	0x0702 0400 – 0x0702 07FF	1K
R_TWD	0x0702 0800 – 0x0702 0BFF	1K
R_PWM	0x0702 0C00 – 0x0702 0FFF	1K
R_INTC	0x0702 1000 – 0x0702 13FF	1K
R_GPIO	0x0702 2000 – 0x0702 23FF	1K
GPADC	0x0703 0000 – 0x0703 03FF	1K
THS	0x0703 0400 – 0x0703 07FF	1K
LRADC	0x0703 0800 – 0x0703 0BFF	1K
DMIC	0x0703 1000 – 0x0703 13FF	1K
AUDIO_CODEC	0x0703 2000 – 0x0703 2FFF	4K
R_I2S0	0x0703 3000 – 0x0703 3FFF	4K
R_IR_RX	0x0704 0000---0x0704 03FF	1K
R_DSP0	0x0705 0000---0x0705 07FF	2K
R_DSP1	0x0705 1000---0x0705 17FF	2K
/	/	/
R_UART0	0x0708 0000---0x0708 03FF	1K
R_TWI0	0x0708 1400---0x0708 17FF	1K
/	/	/
RTC	0x0709 0000---0x0709 03FF	1K
R_DMAC	0x0709 1000---0x0709 1FFF	4K
R_MSGBOX	0x0709 2000---0x0709 2FFF	4K
R_SPINLOCK	0x0709 3000---0x0709 3FFF	4K
DRAM		
DRAM SPACE	0x4000 0000---0xFFFF FFFF	3G

The following is the system memory mapping seen by the DSP0/1 host.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Module	Cacheable Property	Address	Size (Bytes)	Address Mapping Description
SRAM A2	Non-cacheable (Total 512 MB)	0x0010 0000---0x002C FFFF	1856 K	1856 KB (29*64 KB) 0x0010 0000---0x0011 FFFF (2*64 KB) 0x0012 0000---0x002C FFFF (27*64 KB)
DSP0 IRAM		0x002D 0000---0x002E FFFF	128 K	The host (except DSP0) accesses the address through the external bus of DSP0 to operate the internal local SRAM.
DSP0 DRAM0		0x002F 0000---0x002F FFFF	64 K	
DSP0 DRAM1		0x0030 0000---0x0030 FFFF	64 K	
DSP0 IRAM		0x0040 0000---0x0041 FFFF	128 K	The DSP accesses the address through the internal bus to directly access the local SRAM, which is more efficient than the external bus.
DSP0 DRAM0		0x0042 0000---0x0042 FFFF	64 K	
DSP0 DRAM1		0x0044 0000---0x0044 FFFF	64 K	
SYS_PERI & DSP_PERI		0x0200 0000---0x09FF FFFF	128 M	The peripheral address space in SYS area and CPUS area. The address is fixed as non-cacheable.
DRAM SPACE		0x1800 0000---0x1FFF FFFF	128 M	The front 128 M space of DDR (0x4000 0000--0x47FF FFFF)
Cacheable Mapping Area				
SRAM A2	Cacheable (Total 512 MB)	0x2010 0000---0x202C FFFF	1856 K	1856 KB (29*64 KB) 0x2010 0000---0x2011 FFFF (2*64 KB) 0x2012 0000---0x202C FFFF (27*64 KB)
DSP0 IRAM		0x202D 0000---0x202E FFFF	128 K	The host (except DSP0) accesses these address through the external bus of DSP0 to operate the internal local SRAM.
DSP0 DRAM0		0x202F 0000---0x202F FFFF	64 K	
DSP0 DRAM1		0x2030 0000---0x2030 FFFF	64 K	
DRAM SPACE		0x3800 0000---0x3FFF FFFF	128 M	The front 128 M space of DDR (0x4000 0000--0x47FF FFFF)
DRAM Special Mapping Area				
DRAM SPACE	Non-Cacheable	0x4000 0000---0x7FFF FFFF	1 G	

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Module	Cacheable Property	Address	Size (Bytes)	Address Mapping Description
DRAM SPACE	Cacheable/ Non-Cacheable	0x8000 0000---0xBFFF FFFF	1 G	The reserved space
DRAM SPACE	Cacheable	0xC000 0000---0xFFFF FFFF	1 G	The front 1 GB space of DDR (0x4000 0000--0x7FFF FFFF)

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.2 CPUX Configuration

3.2.1 Overview

CPUX indicates the ARM CPU subsystem. You can configure the CPUX via the C0_CPUX_CFG and CPU_SUBSYS_CTRL modules.

The C0_CPUX_CFG module is used for configuring Cluster0. Cluster0 is the only cluster of R329 which consists of dual-core Cortex™-A53, 32 KB I-cache and 32 KB D-cache per core, and the shared 256 KB L2 cache. The C0_CPUX_CFG module includes the following features:

- CPU reset system: core reset, debug circuit reset, and other functional modules reset
- CPU related control: interface control, CP15 control, and power-on/off control
- CPU status check: idle status, SMP status, interrupt status, and so on
- CPU debug-related register for control and status

The CPU_SUBSYS_CTRL module is used for controlling CPU subsystem resources, such as GIC-400 and JTAG.

3.2.2 Functional Descriptions

3.2.2.1 Signal Description

For details of CPUX signals, refer to the technical reference manual of *ARM Cortex-A53 DDI0464F_cortex_a53_mpcore_r0p3_trm.pdf*.

3.2.2.2 L2 Idle Mode

Enter the L2 idle mode by configuring the L2 cache of Cluster to enter the wait for interrupt (WFI) mode.

Follow the steps below:

1. Make sure CPU[1:0] of Cluster has entered the WFI mode. You can check it through bit[17:16] of Cluster0 CPU Status Register.
2. Pull high the ACINACTM of Cluster by writing 1 to the bit0 of Cluster 0 Control Register1.
3. Check whether L2 enters idle status by checking whether the STANDBYWFIL2 is high.

To exit the L2 idle mode, set the ACINACTM to low.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.2.2.3 CPUX Reset System

There are three levels of CPUX reset, with the scope size from small to large: core reset, power-on Reset, and H_Reset.

The following table describes all the reset signals in the CPUX reset system.

Table 3-1 Reset Signal Description

Reset Signal	Description
CORE_RST	This is the primary reset signal which resets the corresponding core logic, including NEON, VFP, Debug, ETM, breakpoint, and watchpoint logic. It maps to a warm reset that covers the reset of the processor logic.
PWRON_RST	This power-on reset signal resets all the processor logic, including the Debug, ETM trace unit, breakpoint, watchpoint logic, and performance monitors logic. It maps to a cold reset that covers the reset of the processor logic and the integrated debug functionality. It does not reset debug logic in the debug power domain. PWRON_RST includes CORE_RST, ETM_RST, and DBG_RST.
H_RST	H_RST includes PWRON_RST, L2_RST, MBIST_RST, SOC_DBG_RST, and C0_CPUX_CFG.
AXI2MBUS_RST	This reset signal resets the AXI2MBUS interface logic circuit.
L2_RST	This single, cluster-wide signal resets the L2 memory system and the logic in the SCU.
ETM_RST	This reset signal resets ETM debug logic circuit.
DBG_RST	In the processor power domain, this reset signal resets only the debug, breakpoint, and watchpoint logic. In the debug power domain, it resets the debug logic for each processor.
SOC_DBG_RST	This reset signal resets all the debug logic. It includes DBG_RST.
MBIST_RST	This reset signal resets all resettable registers in the cluster, for entry into, and exit from, MBIST mode.
CPU_SUBSYS_RST	CPU_SUBSYS_RST includes C0_H_RST, GIC-400, and CPU_SUBSYS_CTRL.

3.2.2.4 CPUX Power Block Diagram

The following table shows the power domain of the modules in the CPU subsystem.

Table 3-2 Power Domain of Modules in the CPU Subsystem

Power Domain	Modules	Description
Cluster0	Cluster0, C0_CPUX_CFG, and C0_MBIST	Cluster0 circuit, C0_CPUX_CFG module, and CPU reset/power/mbist
System	Timestamp, GIC, CPU_SUBSYS_CTRL, and Clock	Provides system source of CPU sub-system

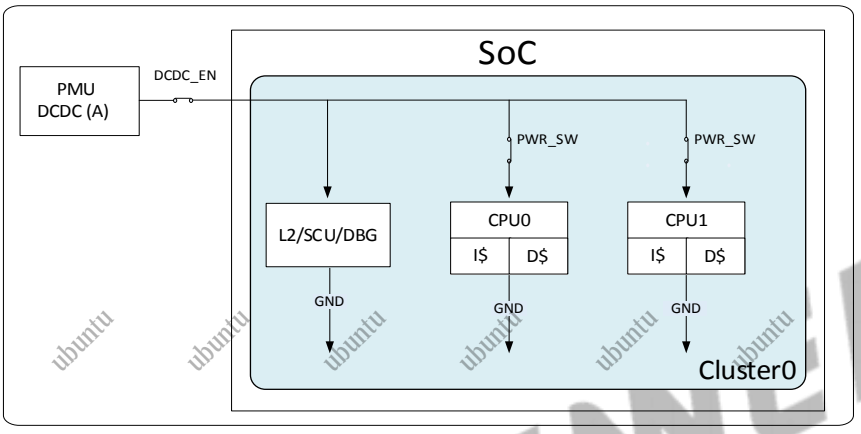
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

As shown in the table, C0_CPUX_CFG belongs to Cluster0 power domain and CPU_SUBSYS_CTRL belongs to the system power domain.

Because C0_CPUX_CFG and Cluster0 belong to the same power domain, C0_CPUX_CFG holds in the default state when Cluster0 restarts. You need to initial C0_CPUX_CFG via software after C0_H_RST is de-asserted.

The following figure shows the default power domain of Cluster0.

Figure 3-1 Default Power Domain of Cluster0



For Cluster0, the power switch of all CPU cores are on, the PWRON_RST of all CPU cores are de-asserted, the CORE_RST of CPU0 is de-asserted, and the CORE_RST of CPU1 is asserted.

3.2.2.5 Operation Principle

The CPU-related operations, such as opening/closing cores, clustering switches, and querying status, require proper configurations of C0_CPUX_CFG module and the related system control resources like the bus and clock.

3.2.3 Programming Guidelines

For CPU core and cluster operation, refer to the *R329_CPU_AP_Note*.

3.2.4 Cluster 0 Configuration Register List

Module Name	Base Address
C0_CPUX_CFG	0x09010000

Register Name	Offset	Description
C0_RST_CTRL	0x0000	Cluster 0 Reset Control Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
C0_CTRL_REG0	0x0010	Cluster 0 Control Register0
C0_CTRL_REG1	0x0014	Cluster 0 Control Register1
C0_CTRL_REG2	0x0018	Cluster 0 Control Register2
CACHE_CFG_REG	0x0024	Cache Configuration Register
C0_CPU_STATUS	0x0080	Cluster 0 CPU Status Register
L2_STATUS_REG	0x0084	Cluster 0 L2 Status Register
DBG_REG0	0x00C0	Cluster 0 Debug Control Register0
DBG_REG1	0x00C4	Cluster 0 Debug Control Register1
AXI_MNT_CTRL_REG	0x00D0	AXI Monitor Control Register
AXI_MNT_PRD_REG	0x00D4	AXI Monitor Period Register
AXI_MNT_RLTCY_REG	0x00D8	AXI Monitor Read Total Latency Register
AXI_MNT_WLTCY_REG	0x00DC	AXI Monitor Write Total Latency Register
AXI_MNT_RREQ_REG	0x00E0	AXI Monitor Read Request Times Register
AXI_MNT_WREQ_REG	0x00E4	AXI Monitor Write Request Times Register
AXI_MNT_RBD_REG	0x00E8	AXI Monitor Read Bandwidth Register
AXI_MNT_WBD_REG	0x00EC	AXI Monitor Write Bandwidth Register

3.2.5 Cluster 0 Configuration Register Description

3.2.5.1 0x0000 Cluster 0 Reset Control Register (Default Value: 0x13FF_0101)

Offset: 0x0000			Register Name: C0_RST_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	/	/	/
28	R/W	0x1	DDR_RST AXI2MBUS Logic Circuit Reset 0: Assert 1: De-assert
27:26	/	/	/
25	R/W	0x1	MBIST_RST CPUBIST Reset The reset signal is for test. 0: Assert 1: De-assert

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: C0_RST_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
24	R/W	0x1	SOC_DBG_RST Cluster SoC Debug Reset 0: Assert 1: De-assert
23:18	/	/	/
17:16	R/W	0x3	DBG_RST Cluster Debug Reset Assert bit for CPU[1:0] The meaning of each bit: 0: Assert 1: De-assert
15:9	/	/	/
8	R/W	0x1	L2_RST Cluster L2 Cache Reset 0: Assert 1: De-assert
7:2	/	/	/
1:0	R/W	0x1	CORE_RESET Cluster CPU[1:0] Reset Assert 0: Assert 1: De-assert

3.2.5.2 0x0010 Cluster 0 Control Register0 (Default Value: 0x8000_0000)

Offset: 0x0010			Register Name: C0_CTRL_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x1	SYSBAR_DISABLE Disable broadcasting barriers to the system bus. 0: Barriers are broadcasted to the system bus. This requires an AMBA4 interconnection. 1: Barriers are broadcasted to the system bus. This is compatible with an AXI3 interconnection.
30	R/W	0x0	BROADCAST_INNER Enable broadcasting inner shareable transactions. 0: Inner shareable transactions are not broadcasted externally. 1: Inner shareable transactions are broadcasted externally.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0010			Register Name: C0_CTRL_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
29	R/W	0x0	BROADCAST_OUTER Enable broadcasting outer shareable transactions. 0: Outer Shareable transactions are not broadcasted externally. 0: Outer Shareable transactions are broadcasted externally.
28	R/W	0x0	BROADCAST_CACHE_MAINT Enable broadcasting cache maintenance operations to downstream caches. 0: Cache maintenance operations are not broadcasted to downstream caches. 1: Cache maintenance operations are broadcasted to downstream caches.
27:10	/	/	/
9:8	R/W	0x0	CP15S_DISABLE Disable write access to some secure CP15 registers of CPU[1:0]. The meaning of each bit: 0: Enable write access to CP15 register of the corresponding CPU. 1: Disable write access to CP15 register of the corresponding CPU.
7:5	/	/	/
4	R/W	0x0	L2_RST_DISABLE Disable automatic L2 cache invalidation at reset. 0: L2 cache is reset by hardware. 1: L2 cache is not reset by hardware.
3:2	R	0x0	/
1:0	R/W	0x0	L1_RST_DISABLE Disable automatic Cluster CPU[1:0] L1 cache invalidation at reset. 0: L1 cache is reset by hardware. 1: L1 cache is not reset by hardware.

3.2.5.3 0x0014 Cluster 0 Control Register1 (Default Value: 0x0000_0000)

Offset: 0x0014			Register Name: C0_CTRL_REG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0014			Register Name: C0_CTRL_REG1
Bit	**Read/Write**	**Default/Hex**	**Description**
1	R/W	0x0	CRM_LowFreq_AutoSelect Enable CRM to automatically select low frequency. 0: Disabled 1: Enabled
0	R/W	0x0	ACINACTM Deactivate snoop interface. 0: Snoop interface is active. 1: Snoop interface is inactive and no longer accepts requests.

3.2.5.4 0x0018 Cluster 0 Control Register2 (Default Value: 0x0000_0010)

Offset: 0x0018			Register Name: C0_CTRL_REG2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:25	/	/	/
24	R/W	0x0	EVENTI Event input for processor to wake up from WFE state. This bit must remain high for at least one clock cycle to be visible by the cores.
23:22	/	/	/
21:20	R/W	0x0	EXM_CLR[1:0] Clear the status of the corresponding interface. The meaning of each bit: 0: Not to clear 1: Clear
19:0	/	/	/

3.2.5.5 0x0024 Cache Configuration Register (Default Value: 0x0018_001A)

Offset: 0x0024			Register Name: CACHE_CFG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:22	/	/	/
21:19	R/W	0x3	EMA_L2D Indicates the port number of L2 Cache SRAM EMA control port.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0024			Register Name: CACHE_CFG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
18:17	R/W	0x0	EMAW_L2D Indicates the port number of L2 Cache SRAM EMAW control port.
16	R/W	0x0	EMAS_L2D Indicates the port number of L2 Cache SRAM EMAS control port.
15:6	/	/	/
5:3	R/W	0x3	EMA Indicates the port number of Cache SRAM EMA control port.
2:1	R/W	0x1	EMAW Indicates the port number of Cache SRAM EMAW control port.
0	R/W	0x0	EMAS Indicates the port number of Cache SRAM EMAS control port.

3.2.5.6 0x0080 Cluster0 CPU Status Register (Default Value: 0x000E_0000)

Offset: 0x0080			Register Name: C0_CPU_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	/	/	/
25:24	R	0x0	SMP_AMP Configure the operating mode for CPU[1:0] as Symmetric Multiprocessing mode or Asymmetric Multiprocessing mode. 0: AMP mode 1: SMP mode
23:18	/	/	/
17:16	R	0xE	STANDBYWFI Indicates if CPU[1:0] of Cluster0 is in WFI standby mode. 0: The processor is not in WFI standby mode. 1: The processor is in WFI standby mode.
15:10	/	/	/
9:8	R	0x0	STANDBYWFE Indicates if CPU[1:0] of Cluster0 is in the WFE standby mode. 0: The processor is not in WFE standby mode. 1: The processor is in WFE standby mode.
7:1	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0080			Register Name: C0_CPU_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
0	R	0x0	STANDBYWFIL2 Indicates if the L2 memory system of Cluster0 is in WFI standby mode. 0: Cluster0 L2 is not in WFI standby mode. 1: Cluster0 L2 is in WFI standby mode.

3.2.5.7 0x0084 L2 Status Register (Default Value: 0x0000_0000)

Offset: 0x0084			Register Name: L2_STATUS_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:10	/	/	/
9	R	0x0	EVENTO Event output. This bit is asserted high for 3 clock cycles when any core in the cluster executes an SEV instruction.
8:0	/	/	/

3.2.5.8 0x00C0 Cluster 0 Debug Control Register0 (Default Value: 0x0000_000F)

Offset: 0x00C0			Register Name: DBG_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:10	/	/	/
9:8	R/W	0x0	DBGRESTART[1:0] Indicates if there are external restart requests for CPU[1:0]. The meaning of each bit: 0: There are no such requests. 1: There are such requests.
7:2	/	/	/
1:0	R/W	0x3	C_DBGPWRDUP Indicates if CPU[1:0] is powered-up. The meaning of each bit: 0: The core is powered down. 1: The core is powered up.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.2.5.9 0x00C4 Cluster 0 Debug Control Register1 (Default Value: 0x0000_0000)

Offset: 0x00C4			Register Name: DBG_REG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:14	/	/	/
13:12	R/W	0x0	DBGRESTARTED[1:0] Handshake for DBGRESTART.
11:6	/	/	/
5:4	R	0x0	C_DBGNOPWRDWN Do Not Power Down Request. Indicates if the debugger requests CPU[1:0] not to power down. The meaning of each bit: 0: There are no such requests. 1: There are such requests.
3:2	/	/	/
1:0	R	0x0	C_DBGPWRUPREQ Power Up Request. Indicates if the debugger requests CPU[1:0] to power up. The meaning of each bit: 0: There are no such requests. 1: There are such requests.

3.2.5.10 0x00D0 CPU AXI Monitor Control Register (Default Value: 0x0000_0000)

Offset: 0x00D0			Register Name: AXI_MNT_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1	R/W	0x0	AXI_MONITOR function clear 0: Not to clear. 1: Clear the AXI_MONITOR related registers with the offset address from 0x00D8 to 0x00EC.
0	R/W	0x0	Enable AXI_MONITOR function 0: Disabled 1: Enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.2.5.11 0x00D4 CPU AXI Monitor Period Register (Default Value: 0x0000_0000)

Offset: 0x00D4			Register Name: AXI_MNT_PRD_REG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	The monitoring period of AXI_MONITOR function (unit: cycles)

3.2.5.12 0x00D8 CPU AXI Monitor Read Latency Register (Default Value: 0x0000_0000)

Offset: 0x00D8			Register Name: AXI_MNT_RLTCY_REG
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	The total latency of AXI_MONITOR function read channel (unit: cycles)

3.2.5.13 0x00DC CPU AXI Monitor Write Latency Register (Default Value: 0x0000_0000)

Offset: 0x00DC			Register Name: AXI_MNT_WLTCY_REG
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	The total latency of AXI_MONITOR function write channel (unit: cycles)

3.2.5.14 0x00E0 CPU AXI Monitor Read Request Register (Default Value: 0x0000_0000)

Offset: 0x00E0			Register Name: AXI_MNT_RREQ_REG
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	The total times of AXI_MONITOR function read channel requests.

3.2.5.15 0x00E4 CPU AXI Monitor Write Request Register (Default Value: 0x0000_0000)

Offset: 0x00E4			Register Name: AXI_MNT_WREQ_REG
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	The total times of AXI_MONITOR function write channel requests.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.2.5.16 0x00E8 CPU AXI Monitor Read Bandwidth Register (Default Value: 0x0000_0000)

Offset: 0x00E8			Register Name: AXI_MNT_RBD_REG
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	The bandwidth of AXI_MONITOR function read channel (Unit: KB)

3.2.5.17 0x00EC CPU AXI Monitor Write Bandwidth Register (Default Value: 0x0000_0000)

Offset: 0x00EC			Register Name: AXI_MNT_WBD_REG
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	The bandwidth of AXI_MONITOR function write channel (Unit: KB)

3.2.6 CPU Subsystem Control Register List

Module Name	Base Address
CPU_SUBSYS_CTRL	0x08100000

Register Name	Offset	Description
GENER_CTRL_REG0	0x0000	General Control Register0
GIC_JTAG_RST_CTRL	0x000C	GIC and JTAG Reset Control Register
C0_INT_EN	0x0010	Cluster0 Interrupt Enable Control Register
IRQ_FIQ_STATUS	0x0014	IRQ/FIQ Status Register
GENER_CTRL_REG2	0x0018	General Control Register2
DBG_STATE	0x001C	Debug State Register
RVBARADDR0_L	0x0040	Reset Vector Base Address Register0_L
RVBARADDR0_H	0x0044	Reset Vector Base Address Register0_H
RVBARADDR1_L	0x0048	Reset Vector Base Address Register1_L
RVBARADDR1_H	0x004C	Reset Vector Base Address Register1_H
RVBARADDR2_L	0x0050	Reset Vector Base Address Register2_L
RVBARADDR2_H	0x0054	Reset Vector Base Address Register2_H
RVBARADDR3_L	0x0058	Reset Vector Base Address Register3_L
RVBARADDR3_H	0x005C	Reset Vector Base Address Register3_H

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.2.7 CPU Subsystem Control Register Description

3.2.7.1 0x0000 General Control Register0 (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: GENER_CTRL_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:6	/	/	/
5:4	R/W	0x0	Cluster 0 AA64NAA32 Register width state of CPU[1:0]. The meaning of each bit: 0: AArch32 1: AArch64 This pin is sampled only during the reset of the processor.
3	/	/	/
2	R/W	0x0	Cluster 0 Corepll Select Register width state 0: CCU clock 1: Disp PLL clock
1	R/W	0x0	IDC Clock Enable 0: IDC clock is disabled 1: IDC clock is enabled
0	R/W	0x0	GIC_CFGSDISABLE Disable write access to some secure GIC registers. 0: Write access to some secure GIC registers is enabled. 1: Write access to some secure GIC registers is disabled.

3.2.7.2 0x000C GIC and JTAG Reset Control Register (Default Value: 0x0000_0F07)

Offset: 0x000C			Register Name: GIC_JTAG_RST_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17:16	R/W	0x0	EXM_CLR[1:0] Clear the status of interface for debug.
15:12	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x000C			Register Name: GIC_JTAG_RST_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
11	R/W	0x1	CS_RST CoreSight reset 0: Assert 1: De-assert
10	R/W	0x1	DAP_RST DAP reset 0: Assert 1: De-assert
9	R/W	0x1	PORTRST DAP reset signal control 0: Assert 1: De-assert
8	R/W	0x1	TRST JTAG RTST signal 0: Assert 1: De-assert
7:2	/	/	/
1	R/W	0x1	IDC_RST Interrupt delay controller reset 0: Assert 1: De-assert
0	R/W	0x1	GIC_RST GIC_reset_cpu_reg 0: Assert 1: De-assert

3.2.7.3 0x0010 Cluster 0 Interrupt Enable Register (Default Value: 0x0000_FFFF)

Offset: 0x0010			Register Name: C0_INT_EN
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0010			Register Name: C0_INT_EN
Bit	Read/Write	Default/Hex	Description
15:0	R/W	0xFFFF	C0_GIC_EN Mask the FIQ_OUT[15:0] and IRQ_OUT[15:0] signals to system domain. The meaning of each bit: 0: Mask the FIQ_OUT and IRQ_OUT signals to system domain. 1: Not to mask the FIQ_OUT and IRQ_OUT signals to system domain.

3.2.7.4 0x0014 GIC IRQ/FIQ Status Register (Default Value: 0x0000_0000)

Offset: 0x0014			Register Name: IRQ_FIQ_STATUS
Bit	Read/Write	Default/Hex	Description
31:16	R/W	0x0000	FIQ_OUT[15:0]
15:0	R/W	0x0000	IRQ_OUT[15:0]

3.2.7.5 0x0018 General Control Register2 (Default Value: 0x0000_0000)

Offset: 0x0018			Register Name: GENER_CTRL_REG2
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	CDBGRSTACK Debug Reset ACK 0: There are no debug reset ACKs. 1: There are debug reset ACKs.
15:1	/	/	/
0	R/W	0x0	C0_TSCLKCHANGE Cluster 0 time stamp change. 0: There are no time stamp changes. 1: There are time stamp changes.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.2.7.6 0x001C Debug State Register (Default Value: 0x0000_0000)

Offset: 0x001C			Register Name: DBG_STATE
Bit	Read/Write	Default/Hex	Description
31:24	R	0x0	CLU_PWRSW_STA
23:1	/	/	/
0	R	0x0	C0_DBG_STATE Indicates if Cluster0 is in debug mode or normal mode. 0: Normal mode 1: Debug mode

3.2.7.7 0x0040 Reset Vector Base Address Register0_L (Default Value: 0x0000_0000)

Offset: 0x0040			Register Name: RVBARADDR0_L
Bit	Read/Write	Default/Hex	Description
31:2	R/W	0x0	RVBARADDR[31:2] Reset Vector Base Address[31:2] for executing in 64-bit state (AArch64) of CPU0.
1:0	/	/	/

3.2.7.8 0x0044 Reset Vector Base Address Register0_H (Default Value: 0x0000_0000)

Offset: 0x0044			Register Name: RVBARADDR0_H
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R/W	0x0	RVBARADDR[39:32] Reset Vector Base Address[39:32] for executing in 64-bit state (AArch64) of CPU0.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.2.7.9 0x0048 Reset Vector Base Address Register1_L (Default Value: 0x0000_0000)

Offset: 0x0048			Register Name: RVBARADDR1_L
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	R/W	0x0	RVBARADDR[31:2] Reset Vector Base Address[31:2] for executing in 64-bit state (AArch64) of CPU1.
1:0	/	/	/

3.2.7.10 0x004C Reset Vector Base Address Register1_H (Default Value: 0x0000_0000)

Offset: 0x004C			Register Name: RVBARADDR1_H
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R/W	0x0	RVBARADDR[39:32] Reset Vector Base Address[39:32] for executing in 64-bit state (AArch64) of CPU1.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3 Clock Controller Unit (CCU)

3.3.1 Overview

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

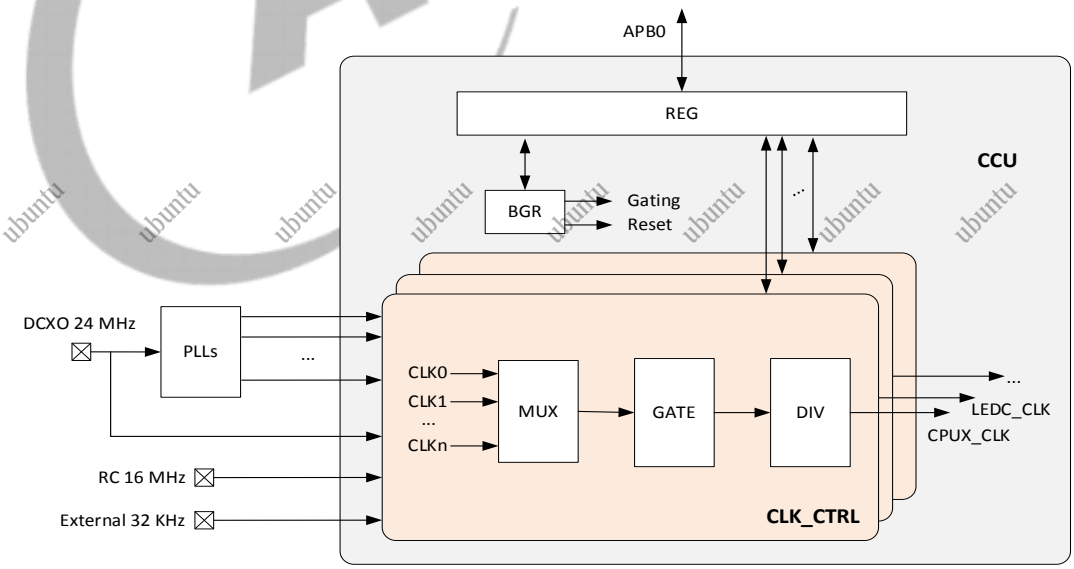
The CCU includes the following features:

- 4 PLLs
- Bus source and divisions
- Clock output control
- Configuring modules clock
- Bus clock gating
- Bus software reset

3.3.2 Block Diagram

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

Figure 3-2 Functional Block Diagram of the CCU



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.3 Functional Descriptions

3.3.3.1 System Bus Tree

The system buses consist of advanced high-performance buses (AHBs) and advanced peripheral buses (APBs), including:

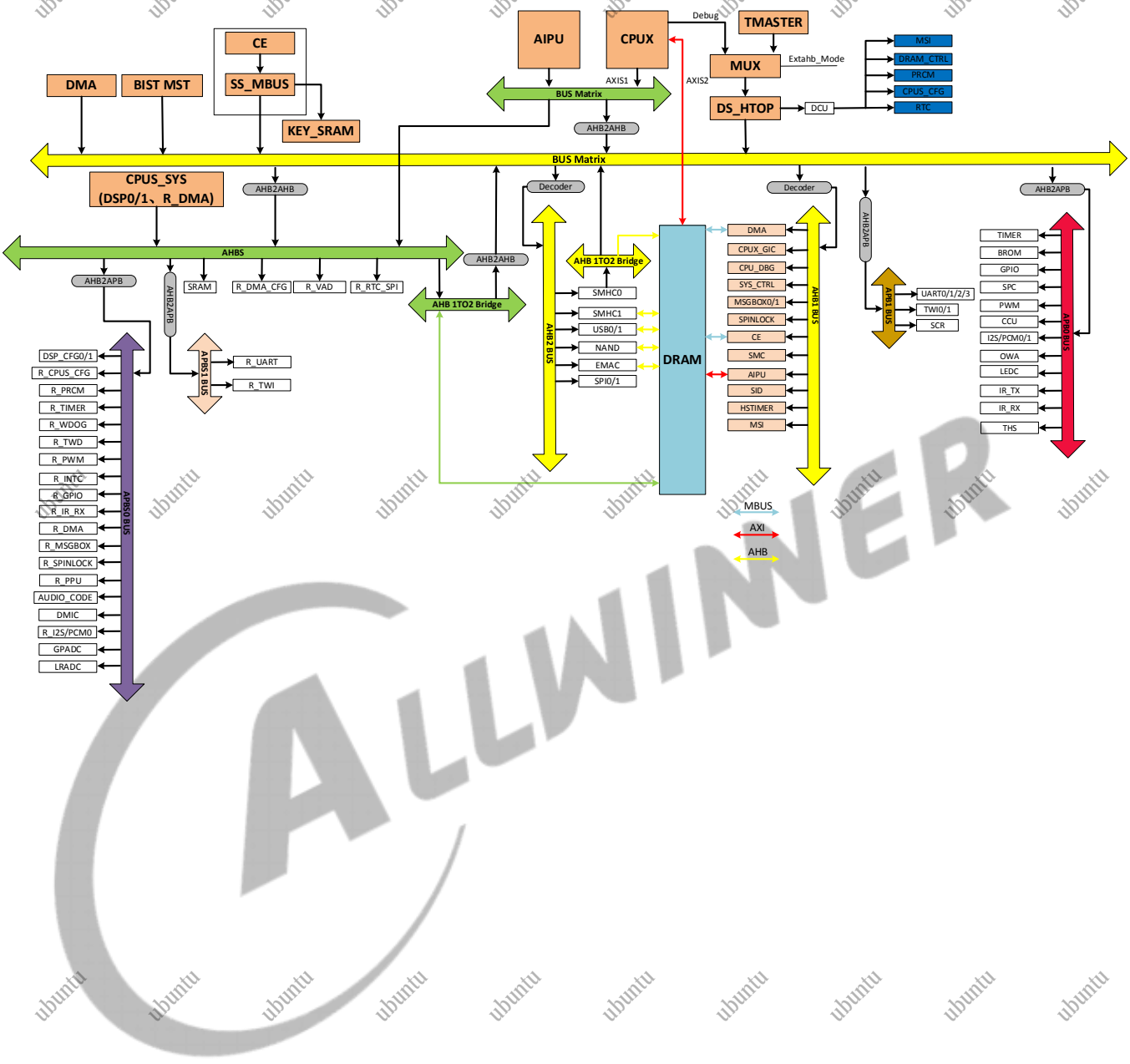
- AHB1, AHB2, APB0, and APB1 in CPUX domain
  - AHB1 is used for mounting the AHB devices related to the system resources.
  - AHB2 is used for mounting storage AHB devices like SD cards and NAND.
  - APB0 is used for mounting APB devices related to the system configurations.
  - APB1 is used for mounting the low-speed APB devices like TWI and UART devices.
- AHBS, APBS0, and APBS1 in CPUS domain
  - AHBS is used for mounting AHB devices that are in the CPUS_SYS domain.
  - APBS0 is used for mounting APB devices that are in the CPUS_SYS domain.
  - APBS1 is used for mounting the low-speed APB devices, like TWI and UART devices, in the CPUS_SYS domain.

The AHB1 and AHB2 are synchronous, and the others are asynchronous. All devices mounted at the bus should use the related bus clocks, and the gating signals for the bus are from the CCU module.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

The following figure shows the diagram of the System Bus Tree.

Figure 3-3 System Bus Tree

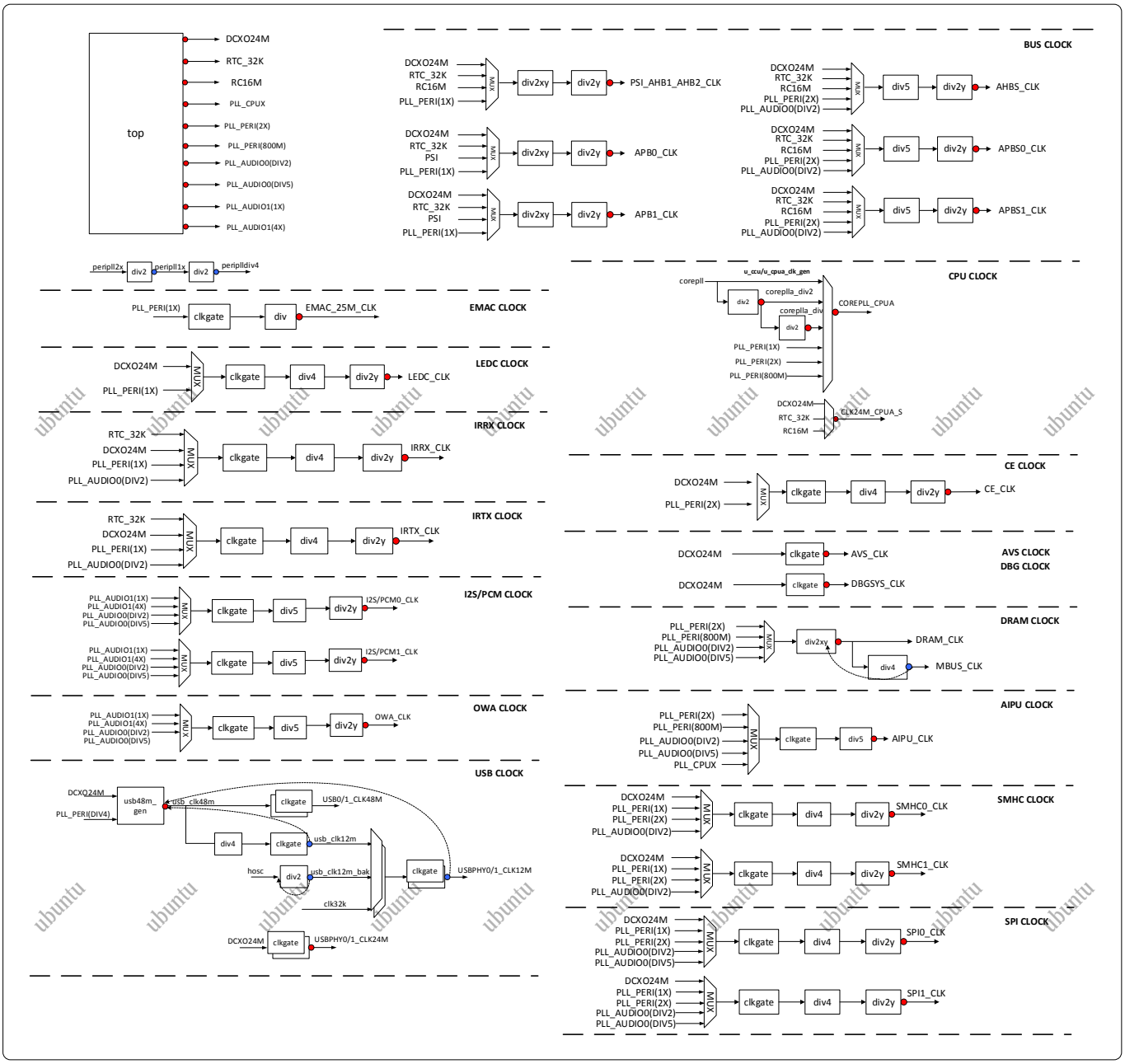


Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.3.2 Module Clock Generation

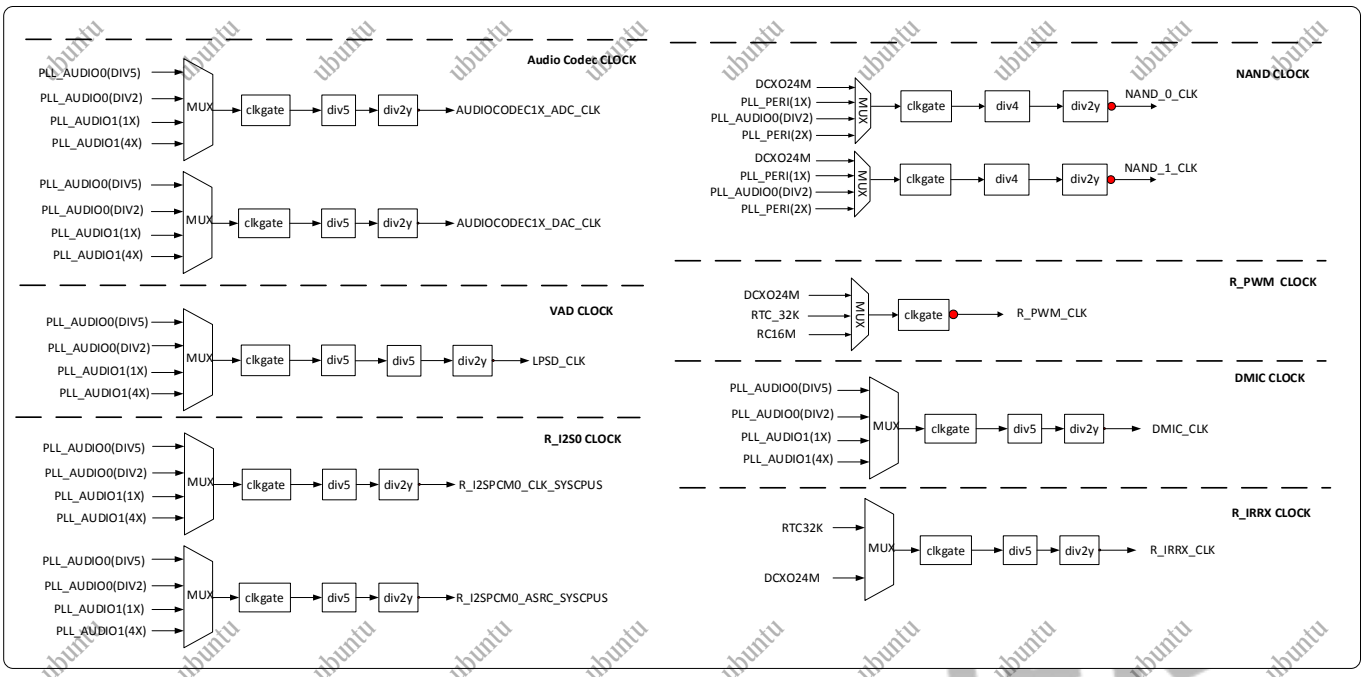
The following figure describes module clock generation.

Figure 3-4 Module Clock Generation - Part 1



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 3-5 Module Clock Generation - Part 2

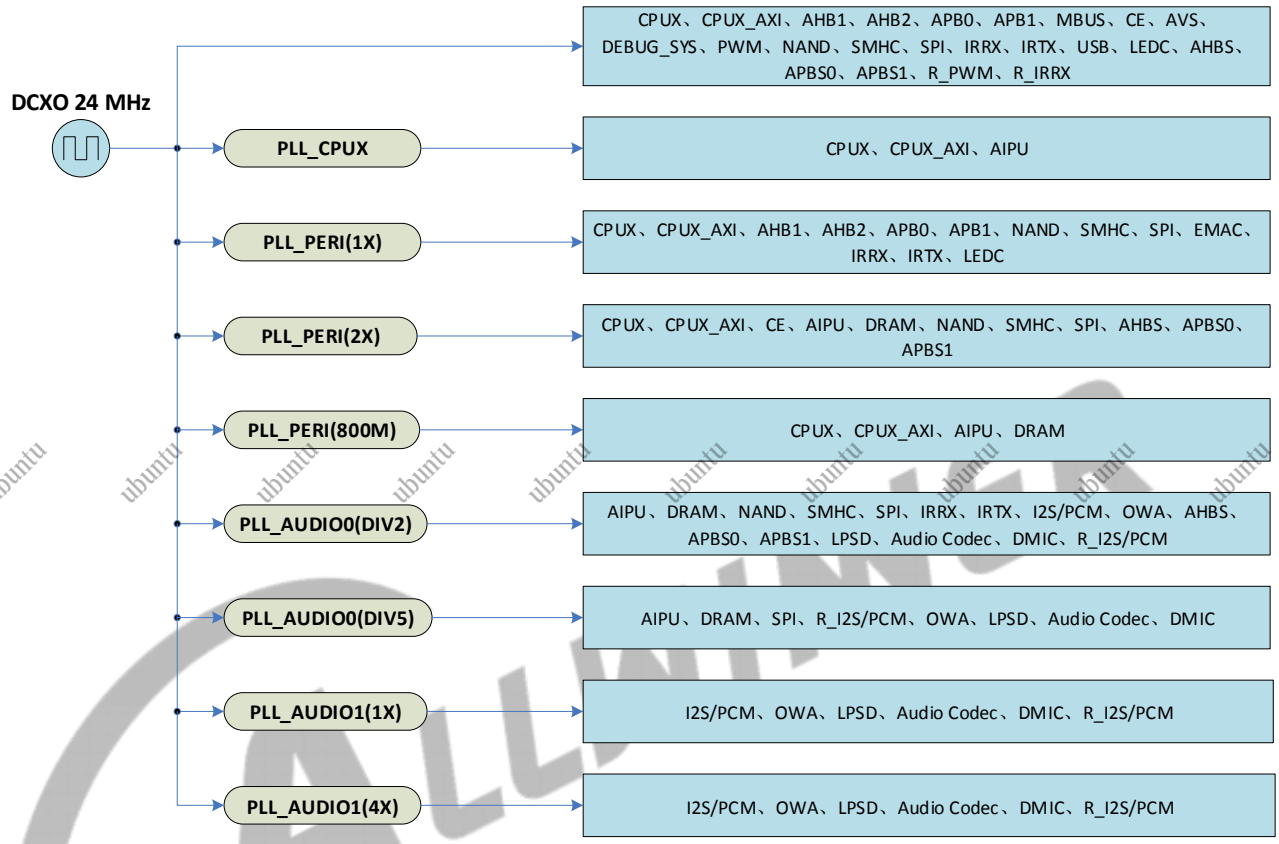


Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.3.3 PLL Distribution

The following figure shows the block diagram of the PLL distribution.

Figure 3-6 PLL Distribution



3.3.3.4 PLL Features

The following table shows the PLL features.

Table 3-3 PLL Features

PLL	Stable Operating Frequency	Actual Operating Frequency	Default Frequency	Spread Spectrum	Linear FM	Rate Control	Pk-Pk	Lock Time
PLL_CPU	288 MHz to 5.0 GHz	288 MHz to 1.8 GHz	480 MHz	No	No	No	< 200 ps	1.5 ms

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

PLL	Stable Operating Frequency	Actual Operating Frequency	Default Frequency	Spread Spectrum	Linear FM	Rate Control	Pk-Pk	Lock Time
PLL_AUDIO1	180 MHz to 3.2 GHz	PLL_AUDIO1(1X): 24.576 MHz 22.5792 MHz PLL_AUDIO1(4X): 98.304 MHz 90.3168 MHz	24.576 MHz	Yes	No	No	< 200 ps	500 us
PLL_AUDIO0	180 MHz to 3.072 GHz	Fvco/Div (Div: 1 to 8)	DIV2: 1.179648 GHz DIV5: 471.8592 MHz	Yes	No	No	< 200 ps	500 us
PLL_PERI	180 MHz to 3.2 GHz	Fvco/Div (Div: 1 to 8)	Fvco: 2.4 GHz DIV2: PLL_PERI(2X), 1.2 GHz DIV3: PLL_PERI(800M), 0.8 GHz	Yes	No	No	< 200 ps	500 us

3.3.4 Programming Guidelines

3.3.4.1 Configuring the Frequency of PLL_CPUX

The frequency configuration formula of PLL_CPUX:

PLL_CPUX = 24 MHz*N/P

The parameter N is the frequency-doubling factor of PLL. The next parameter configuration can proceed after the PLL relock. The parameter P is a digital post-frequency-division factor, which can be dynamically switched in real-time, without affecting the normal work of PLL.

NOTE

PLL_CPUX supports dynamic adjustment (modifying the value of N). However, for the system stability, to configure the frequency of the PLL_CPUX from a higher value to a lower one, switch the clock source of the CPU to another clock whose frequency is not higher than the current one first, and configure PLL_CPUX to the target low frequency, and then switch the clock source of the CPU back to PLL_CPUX.

Follow the steps below to adjust the frequency of PLL_CPUX:

Step 1 Before you configure PLL_CPUX, switch the clock source of CPU to PLL_PERI0(1X).

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 2 Modify the parameters N and P of PLL_CPUX.

Step 3 Write the PLL Lock Enable bit (bit[29]) of PLL_CPUX_CTRL register to 0 and then to 1.

Step 4 Wait for the Lock bit (bit[28]) of PLL_CPUX_CTRL register to change to 1.

Step 5 Switch the clock source of the CPU to PLL_CPUX.

3.3.4.2 Configuring the Frequency of PLL_AUDIO

NOTE

PLL_AUDIO includes PLL_AUDIO0 and PLL_AUDIO1.

The frequency configuration formula of PLL_AUDIO:

PLL_AUDIO0 = 24 MHz*N/M/P0(P1)

PLL_AUDIO1 = 24 MHz*N/M0/M1/P

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

Generally, the frequency of PLL_AUDIO is either 24.576 MHz or 22.5792 MHz, and there are recommended values of the configuration factors for the two frequency points. To implement the desired frequency point of PLL_AUDIO, follow the steps below:

Step 1 Configure the N, M, and P factors.

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

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

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

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

NOTE

When the P factor of PLL_AUDIO1 is an odd number, the clock output is an unequal-duty-cycle signal.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

The recommended values for configuration factors of PLL_AUDIO0 are as follows.

Table 3-4 Recommended Values for Configuration Factors of PLL_AUDIO0

Clock Source (MHz)	Frequency-doubling N	VCO (MHz)	Post frequency-division	PLL output (MHz)	Divisor	Actual Operating Frequency (MHz)	Description
24	98.304	2359.296	2	1179.648	2	589.824	Provides clock source for peripheral devices
					3	393.216	
					6	196.608	
					12	98.304	
					48	24.576	For audio-related modules
			5	471.8592			

3.3.4.3 Configuring the Frequency of General PLLs

Step 1 Make sure the PLL is enabled. If not, refer to section 3.3.4.4 Enabling the PLL to enable the PLL.

Step 2 Configure the PLL_OUTPUT_ENABLE bit (bit[27]) of the PLL control register as 0 to disable the output gate of the PLL because general PLLs are unavailable in the process of frequency modulation.

Step 3 Configure the N and M factors. (It is not suggested to configure M1 factor)

Step 4 Write the LOCK_ENABLE bit (bit[29]) of the PLL control register to 0 and then to 1.

Step 5 Wait for the LOCK bit (bit[28]) of the PLL control register to 1.

Step 6 Configure PLL_OUTPUT_ENABLE bit (bit[27]) of the PLL control register to 1.

3.3.4.4 Enabling the PLL

Follow the steps below to enable the PLL:

Step 1 Configure the N, M1, and M0 factors of the PLL control register.

Step 2 Write the PLL_ENABLE bit of the PLL control register to 1.

Step 3 Write the LOCK_ENABLE bit of the PLL control register to 1.

Step 4 Wait for the status of the Lock to change to 1.

Step 5 Delay 20 us, and then the PLL will be available.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.4.5 Disabling the PLL

Follow the steps below to disable the PLL:

Step 1 Write the PLL_ENABLE bit (bit[31]) of the PLL control register to 0.

Step 2 Write the LOCK_ENABLE bit (bit[29]) of the PLL control register to 0.

CAUTION

In the normal use of PLLs, it is unsuggested to enable and disable the PLLs frequently. Turning on and off the PLLs will cause mutual interference between PLLs, which will affect the stability of the system. When the clock is unnecessary, you can write 0 to the PLL_OUTPUT_EN bit of the PLL control register to disable the output gate of the PLL, instead of writing 0 to the Enable bit to disable the PLL.

3.3.4.6 Configuring Bus Clock

The bus clock supports dynamic switching, but the process of switching needs to follow the following two rules.

- From a higher frequency to a lower frequency: switch the clock source first, and then set the frequency division factor;
- From a lower frequency to a higher frequency: configure the frequency division factor first, and then switch the clock source.

3.3.4.7 Configuring Module Clocks

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

For all module clocks except the DDR clock, configure the clock source and frequency division factor first, and then release the clock gating (that is, set to 1). For the configuration order of the clock source and frequency division factor, follow the rules below:

- With the increasing of the clock source frequency, configure the frequency division factor before the clock source;
- With the decreasing of the clock source frequency, configure the clock source before the frequency division factor.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.4.8 Implementing Spread Spectrum

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

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

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

Where,

P is the frequency division factor of module or PLL;

M0 is the post-frequency division factor of PLL;

M1 is the pre-frequency division factor of PLL;

N is the frequency doubling factor of PLL;

X is the amplitude coefficient of the spread spectrum.

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

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

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

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

$$SDM_BOT = 2^{17}\cdot X_1$$

$$\text{WAVE_STEP} = 2^{17}\cdot(X_2 - X_1)/(24\text{MHz}/PREQ)\cdot 2$$

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

NOTE

Different PLLs have different calculate formulas, refer to the CTRL register of the corresponding PLL in section 3.3.6 CPUX Domain Register Description.

Configuration Procedure

Follow the steps below to implement the spread spectrum:

Step 1 Configure the control register of the corresponding PLL

a) Calculate the factor N and decimal value X according to the PLL frequency and PLL frequency formula. Refer to the control register of the corresponding PLL (named PLL_xxx_CTRL_REG,

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

where xxx is the module name) in 3.3.6 CPUX Domain Register Description for the corresponding PLL frequency formula.

b) Write M0, M1, N, and PLL frequency to the PLL control register.

c) Configure the SDM_Enable bit (bit[24]) of the PLL control register to 1 to enable the spread spectrum function.

Step 2 Configure the pattern control register of the corresponding PLL

a) Calculate the SDM_BOT and WAVE_STEP of the pattern control register according to decimal value X and spread spectrum frequency (the bit[18:17] of the PLL_PAT register)

b) Configure the spread spectrum mode (SPR_FREQ_MODE) to 2 or 3.

c) If the bit PLL_INPUT_DIV_M of the PLL control register is 1, configure the spread spectrum clock source select bit (SDM_CLK_SEL) of the PLL pattern control register to 1. Otherwise, configure SDM_CLK_SEL to the default value 0.

d) Write SDM_BOT, WAVE_STEP, PREQ, SPR_FREQ_MODE, and SDM_CLK_SEL to the PLL pattern control register, and configure the SIG_DELT_PAT_EN bit (bit[31]) of this register to 1.

Step 3 Delay 20 us.

Configuration Example

The following example shows how to configure the spread spectrum frequency as 605.3 MHz to 609.7 MHz.

If M1 = 0, M0 = 0, P = 1, according to the formula $[f_1, f_2] = (N+1+[X_1, X_2]) \cdot 24MHz$, you can get:

$$\begin{aligned} N+1+[X_1, X_2] &= \frac{[605.3, 609.7]}{24} \\ &= \frac{600+[5.3, 9.7]}{24} \\ &= 24+1+[5.3/24, 9.7/24] \end{aligned}$$

Obviously,

N = 24, X_1 = 5.3/24, X_2 = 9.7/24

SDM_BOT = 2^{17} * X_1 = 0x7111

WAVE_STEP = 2^{17} * ($X_2 - X_1$) /(24M/PREQ)*2 = 0x3f; PREQ = 31.5 kHz

If M0 = 1, M1=0, P = 1, then total frequency division factor is (M0 + 1) *1 = 2, so the actual output frequency of PLL is 1212.1 MHz to 1219.4 MHz.

Similarly, you can get:

N = 49, X1 = 12.1/24, X2 = 19.4/24

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Then calculate the values of SDM_BOT and WAVE_STEP according to the formulas, and follow the steps described in Configuration Procedure

3.3.5 CPUX Domain Register List

Module Name	Base Address
CCU	0x02001000

Register Name	Offset	Description
CPUX_AXI_CFG_REG	0x0500	CPUX_AXI Configuration Register
AHB_CFG_REG	0x0510	AHB Configuration Register
APB0_CFG_REG	0x0520	APB0 Configuration Register
APB1_CFG_REG	0x0524	APB1 Configuration Register
MBUS_CFG_REG	0x0540	MBUS Configuration Register
CE_CLK_REG	0x0680	CE Clock Register
CE_BGR_REG	0x068C	CE Bus Gating Reset Register
AIPU_CLK_REG	0x06F0	AIPU Clock Register
AIPU_BGR_REG	0x06FC	AIPU Bus Gating Reset Register
DMAC_BGR_REG	0x070C	DMAC Bus Gating Reset Register
MSGBOX_BGR_REG	0x071C	MSGBOX Bus Gating Reset Register
SPINLOCK_BGR_REG	0x072C	SPINLOCK Bus Gating Reset Register
HSTIMER_BGR_REG	0x073C	HSTIMER Bus Gating Reset Register
AVS_CLK_REG	0x0740	AVS Clock Register
DBGSYS_BGR_REG	0x078C	DBGSYS Bus Gating Reset Register
PWM_BGR_REG	0x07AC	PWM Bus Gating Reset Register
DRAM_CLK_REG	0x0800	DRAM Clock Register
MBUS_MAT_CLK_GATING_REG	0x0804	MBUS Master Clock Gating Register
DRAM_BGR_REG	0x080C	DRAM Bus Gating Reset Register
NAND_0_CLK_REG	0x0810	NAND_0 Clock Register
NAND_1_CLK_REG	0x0814	NAND_1 Clock Register
NAND_BGR_REG	0x082C	NAND Bus Gating Reset Register
SMHC0_CLK_REG	0x0830	SMHC0 Clock Register
SMHC1_CLK_REG	0x0834	SMHC1 Clock Register
SMHC_BGR_REG	0x084C	SMHC Bus Gating Reset Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
UART_BGR_REG	0x090C	UART Bus Gating Reset Register
TWI_BGR_REG	0x091C	TWI Bus Gating Reset Register
SRC_BGR_REG	0x093C	SRC Bus Gating Reset Register
SPI0_CLK_REG	0x0940	SPI0 Clock Register
SPI1_CLK_REG	0x0944	SPI1 Clock Register
SPI_BGR_REG	0x096C	SPI Bus Gating Reset Register
EMAC_25M_CLK_REG	0x0970	EMAC_25M Clock Register
EMAC_BGR_REG	0x097C	EMAC Bus Gating Reset Register
IRRX_CLK_REG	0x0990	IRRX Clock Register
IRRX_BGR_REG	0x099C	IRRX Bus Gating Reset Register
IRTX_CLK_REG	0x09C0	IRTX Clock Register
IRTX_BGR_REG	0x09CC	IRTX Bus Gating Reset Register
I2S/PCM0_CLK_REG	0x0A10	I2S/PCM0 Clock Register
I2S/PCM1_CLK_REG	0x0A14	I2S/PCM1 Clock Register
I2S/PCM_BGR_REG	0x0A1C	I2S/PCM Bus Gating Reset Register
OWA_CLK_REG	0x0A20	OWA Clock Register
OWA_BGR_REG	0x0A2C	OWA Bus Gating Reset Register
USB0_CLK_REG	0x0A70	USB0 Clock Register
USB1_CLK_REG	0x0A74	USB1 Clock Register
USB_BGR_REG	0x0A8C	USB Bus Gating Reset Register
LEDC_CLK_REG	0x0BF0	LEDC Clock Register
LEDC_BGR_REG	0x0BFC	LEDC BUS GATING RESET Register
CCU_SEC_SWITCH_REG	0x0F00	CCU Security Switch Register

3.3.6 CPUX Domain Register Description

3.3.6.1 0x0500 CPUX_AXI Configuration Register (Default Value: 0x0000_0301)

Offset: 0x0500			Register Name: CPUX_AXI_CFG_REG
Bit	Read/Write	Default/Hex	Description
31:27	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0500			Register Name: CPUX_AXI_CFG_REG
Bit	Read/Write	Default/Hex	Description
26:24	R/W	0x0	CLK_SRC_SEL Clock source select The clock source supports dynamic configurations because the MUX for the clock supports glitch-free clock switching. 000: DCXO24M 001: RTC_32K 010: RC16M 011: PLL_CPUX/P 100: PLL_PERI(1X) 101: PLL_PERI(2X) 110: PLL_PERI(800M) CPUX Clock = Clock Source CPUX_AXI Clock = Clock Source/M CPUX_APB Clock = Clock Source/N
23:18	/	/	/
17:16	R/W	0x0	PLL_OUT_EXT_DIVP PLL output external divider P When the output clock is less than 288 MHz, it can divide P to get the required clock frequency. The parameter P supports dynamic configuration. 00: 1 01: 2 10: 4 11: Reserved
15:10	/	/	/
9:8	R/W	0x3	CPUX_APB_FACTOR_N Factor N (N = FACTOR_N +1) FACTOR_N is from 0 to 3. The clock division is no-burr switch, and supports dynamic configuration.
7:2	/	/	/
1:0	R/W	0x1	FACTOR_ M Factor M (M = FACTOR_M +1) FACTOR_M is from 0 to 3. The clock division is no-burr switch, and supports dynamic configuration.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.6.2 0x0510 AHB Configuration Register (Default Value: 0x0000_0000)

Offset: 0x0510			Register Name: AHB_CFG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select PSI_AHB1_AHB2 CLK = Clock Source/M/N. The clock source is no-burr switch, and supports dynamic configuration. 00: DCXO24M 01: RTC_32K 10: RC16M 11: PLL_PERI(1X)
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N. Factor N The clock division is no-burr switch, and supports dynamic configuration. 00: 1 01: 2 10: 4 11: 8
7:2	/	/	/
1:0	R/W	0x0	FACTOR_M Factor M (M= FACTOR_M +1) FACTOR_M is from 0 to 3. The clock division is no-burr switch, and supports dynamic configuration.

3.3.6.3 0x0520 APB0 Configuration Register (Default Value: 0x0000_0000)

Offset: 0x0520			Register Name: APB0_CFG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:26	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0520			Register Name: APB0_CFG_REG
Bit	Read/Write	Default/Hex	Description
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select APB0 CLK = Clock Source/M/N. The clock source is no-burr switch, and supports dynamic configuration. 00: DCXO24M 01: RTC_32K 10: PSI 11: PLL_PERI(1X)
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N. Factor N The clock division is no-burr switch, and supports dynamic configuration. 00: 1 01: 2 10: 4 11: 8
7:2	/	/	/
1:0	R/W	0x0	FACTOR_M Factor M (M= FACTOR_M +1) FACTOR_M is from 0 to 3. The clock division is no-burr switch, and supports dynamic configuration.

3.3.6.4 0x0524 APB1 Configuration Register (Default Value: 0x0000_0000)

Offset: 0x0524			Register Name: APB1_CFG_REG
Bit	Read/Write	Default/Hex	Description
31:26	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0524			Register Name: APB1_CFG_REG
Bit	Read/Write	Default/Hex	Description
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select APB1 CLK = Clock Source/M/N. The clock source is no-burr switch, and supports dynamic configuration. 00: DCXO24M 01: RTC_32K 10: PSI 11: PLL_PERI(1X)
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N The clock division is no-burr switch, and supports dynamic configuration. 00: 1 01: 2 10: 4 11: 8
7:2	/	/	/
1:0	R/W	0x0	FACTOR_M Factor M (M= FACTOR_M +1) FACTOR_M is from 0 to 3. The clock division is no-burr switch, and supports dynamic configuration.

3.3.6.5 0x0540 MBUS Configuration Register (Default Value: 0x4000_0000)

Offset: 0x0540			Register Name: MBUS_CFG_REG
Bit	Read/Write	Default/Hex	Description
31	/	/	/
30	R/W	0x1	MBUS_RST MBUS Reset 0: Assert 1: De-assert Note: The clock of MBUS is from the 4-bit divider of DDR_CLK.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0540			Register Name: MBUS_CFG_REG
Bit	Read/Write	Default/Hex	Description
29:0	/	/	/

3.3.6.6 0x0680 CE Clock Register (Default Value: 0x0000_0000)

Offset: 0x0680			Register Name: CE_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SCLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON SCLK = Clock Source/M/N.
30:25	/	/	/
24	R/W	0x0	CLK_SRC_SEL Clock Source Select 0: DCXO24M 1: PLL_PERI(2X)
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M (M= FACTOR_M +1) FACTOR_M is from 0 to 15.

3.3.6.7 0x068C CE Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x068C			Register Name: CE_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

3.3.6.8 0x06F0 AIPU Clock Register (Default Value: 0x0000_0000)

Offset: 0x06F0			Register Name: AIPU_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	SCLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON SCLK = Clock Source/M/N.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: PLL_PERI(2X) 001: PLL_PERI(800M) 010: PLL_AUDIO0(DIV2) 011: PLL_AUDIO0(DIV5) 100: PLL_CPUX Others: Reserved
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x06F0			Register Name: AIPU_CLK_REG
Bit	Read/Write	Default/Hex	Description
7:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M (M = FACTOR_M +1) The value of FACTOR_M is from 0 to 15.

3.3.6.9 0x06FC AIPU Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x06FC			Register Name: AIPU_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	AIPU_RST AIPU AHB Reset 0: Assert 1: De-assert
15:2	/	/	/
1	R/W	0x0	AIPU_MST_GATING Gating Clock For AIPU AHB Master 0: Mask 1: Pass
0	R/W	0x0	AIPU_SLV_GATING Gating Clock For AIPU AHB Slave 0: Mask 1: Pass

3.3.6.10 0x070C DMAC Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x070C			Register Name: DMAC_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	DMAC_RST DMAC Reset 0: Assert 1: De-assert

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x070C			Register Name: DMAC_BGR_REG
Bit	Read/Write	Default/Hex	Description
15:1	/	/	/
0	R/W	0x0	DMAC_GATING Gating Clock for DMAC The clock of DMAC is from AHB1. 0: Mask 1: Pass

3.3.6.11 0x071C MSGBOX Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x071C			Register Name: MSGBOX_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17	R/W	0x0	MSGBOX1_RST MSGBOX1 Reset 0: Assert 1: De-assert
16	R/W	0x0	MSGBOX0_RST MSGBOX0 Reset 0: Assert 1: De-assert
15:2	/	/	/
1	R/W	0x0	MSGBOX1_GATING Gating Clock for MSGBOX1 0: Mask 1: Pass
0	R/W	0x0	MSGBOX0_GATING Gating Clock for MSGBOX0 0: Mask 1: Pass

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.6.12 0x072C SPINLOCK Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x072C			Register Name: SPINLOCK_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	SPINLOCK_RST SPINLOCK Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	SPINLOCK_GATING Gating Clock for SPINLOCK 0: Mask 1: Pass

3.3.6.13 0x073C HSTIMER Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x073C			Register Name: HSTIMER_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	HSTIMER_RST HSTIMER Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	HSTIMER_GATING Gating Clock for HSTIMER The clock of HSTIMER is from AHB1. 0: Mask 1: Pass

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.6.14 0x0740 AVS Clock Register (Default Value: 0x0000_0000)

Offset: 0x0740			Register Name: AVS_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	SCLK_GATING Gating Special Clock SCLK = DCXO24M. 0: Clock is OFF 1: Clock is ON
30:0	/	/	/

3.3.6.15 0x078C DBGSYS Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x078C			Register Name: DBGSYS_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	DBGSYS_RST DBGSYS Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	DBGSYS_GATING Gating Clock for DBGSYS The clock of DBGSYS is from DCXO24M. 0: Mask 1: Pass

3.3.6.16 0x07AC PWM Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x07AC			Register Name: PWM_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	PWM_RST PWM Reset 0: Assert 1: De-assert

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x07AC			Register Name: PWM_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
15:1	/	/	/
0	R/W	0x0	PWM_GATING Gating Clock for PWM The clock of PWM is from APB0 or DCXO24M. 0: Mask 1: Pass

3.3.6.17 0x0800 DRAM Clock Register (Default Value: 0x8000_0000)

Offset: 0x0800			Register Name: DRAM_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x1	SCLK_GATING Gating Special Clock SCLK = Clock Source/M/N. 0: Clock is OFF 1: Clock is ON
30	R/W	0x0	MODULE_RST Module Reset 0: Assert 1: De-assert
29:28	/	/	/
27	R/WAC	0x0	SDRCLK_UPD SDRCLK Configuration 0 Update Setting 1 will validate the bit. It will be auto cleared after the bit is valid. 0: Invalid 1: Valid
26	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0800			Register Name: DRAM_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select The clock source is no-burr switch, and supports dynamic configuration. 00: PLL_PERI(2X) 01: PLL_PERI(800M) 10: PLL_AUDIO0(DIV2) 11: PLL_AUDIO0(DIV5)
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N The clock division is no-burr switch, and supports dynamic configuration. 00: 1 01: 2 10: 4 11: 8
7:2	/	/	/
1:0	R/W	0x0	FACTOR_M Factor M (M= FACTOR_M +1) FACTOR_M is from 0 to 3. The clock division is no-burr switch, and supports dynamic configuration.

3.3.6.18 0x0804 MBUS Master Clock Gating Register (Default Value: 0x0000_0000)

Offset: 0x0804			Register Name: MBUS_MAT_CLK_GATING_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	AIPU_MCLK_GATING Gating MBUS Clock for AIPU 0: Mask 1: Pass
15:6	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0804			Register Name: MBUS_MAT_CLK_GATING_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
5	R/W	0x0	NAND_MCLK_GATING Gating MBUS Clock for NAND 0: Mask 1: Pass
4	/	/	/
3	R/W	0x0	R_DMAC_MCLK_GATING Gating MBUS Clock for R_DMAC 0: Mask 1: Pass
2	R/W	0x0	CE_MCLK_GATING Gating MBUS Clock for CE 0: Mask 1: Pass
1	/	/	/
0	R/W	0x0	DMAC_MCLK_GATING Gating MBUS Clock for DMAC 0: Mask 1: Pass

3.3.6.19 0x080C DRAM Bus Gating Reset Register (Default Value: 0x0000_0001)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.6.20 0x0810 NAND_0 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0810			Register Name: NAND_0_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	SCLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON SCLK = Clock Source/Divider N/Divider M.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: DCXO24M 001: PLL_PERI(1X) 010: PLL_AUDIO0(DIV2) 011: PLL_PERI(2X) 100: / 1XX:/
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M (M= FACTOR_M +1) FACTOR_M is from 0 to 15.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.6.21 0x0814 NAND_1 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0814			Register Name: NAND_1_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	SCLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON SCLK = Clock Source/Divider N/Divider M.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: DCXO24M 001: PLL_PERI(1X) 010: PLL_AUDIO0(DIV2) 011: PLL_PERI(2X) The others: Reserved
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M (M= FACTOR_M +1) FACTOR_M is from 0 to 15.

3.3.6.22 0x082C NAND Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x082C			Register Name: NAND_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x082C			Register Name: NAND_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
16	R/W	0x0	NAND_RST NAND Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	NAND_GATING Gating Clock for NAND 0: Mask 1: Pass

3.3.6.23 0x0830 SMHC0 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0830			Register Name: SMHC0_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	SCLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON SCLK = Clock Source/M/N.
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: DCXO24M 01: PLL_PERI(1X) 10: PLL_PERI(2X) 11: PLL_AUDIO0(DIV2)
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:4	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0830			Register Name: SMHC0_CLK_REG
Bit	Read/Write	Default/Hex	Description
3:0	R/W	0x0	FACTOR_M Factor M (M= FACTOR_M +1) FACTOR_M is from 0 to 15.

3.3.6.24 0x0834 SMHC1 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0834			Register Name: SMHC1_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SCLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON SCLK = Clock Source/M/N.
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: DCXO24M 01: PLL_PERI(1X) 10: PLL_PERI(2X) 11: PLL_AUDIO0(DIV2)
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M (M= FACTOR_M +1) FACTOR_M is from 0 to 15.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.6.25 0x084C SMHC Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x084C			Register Name: SMHC_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17	R/W	0x0	SMHC1_RST SMHC1 Reset 0: Assert 1: De-assert
16	R/W	0x0	SMHC0_RST SMHC0 Reset 0: Assert 1: De-assert
15:2	/	/	/
1	R/W	0x0	SMHC1_GATING Gating Clock for SMHC1 0: Mask 1: Pass
0	R/W	0x0	SMHC0_GATING Gating Clock for SMHC0 0: Mask 1: Pass

3.3.6.26 0x090C UART Bus Gating Reset Register (Default Value: 0x0000_0000)

NOTE

The clock of the UART is from APB1.

Offset: 0x090C			Register Name: UART_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:20	/	/	/
19	R/W	0x0	UART3_RST UART3 Reset 0: Assert 1: De-assert

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x090C			Register Name: UART_BGR_REG
Bit	Read/Write	Default/Hex	Description
18	R/W	0x0	UART2_RST UART2 Reset 0: Assert 1: De-assert
17	R/W	0x0	UART1_RST UART1 Reset 0: Assert 1: De-assert
16	R/W	0x0	UART0_RST UART0 Reset 0: Assert 1: De-assert
15:4	/	/	/
3	R/W	0x0	UART3_GATING Gating Clock for UART3 0: Mask 1: Pass
2	R/W	0x0	UART2_GATING Gating Clock for UART2 0: Mask 1: Pass
1	R/W	0x0	UART1_GATING Gating Clock for UART1 0: Mask 1: Pass
0	R/W	0x0	UART0_GATING Gating Clock for UART0 0: Mask 1: Pass

3.3.6.27 0x091C TWI Bus Gating Reset Register (Default Value: 0x0000_0000)

NOTE

The clock of the TWI is from APB1.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

3.3.6.28 0x093C SCR Bus Gating Reset Register (Default Value: 0x0000_0000)

NOTE

The clock of the SCR is from APB1.

Offset: 0x093C			Register Name: SCR_CLK_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	SCR_RST SCR Reset 0: Assert 1: De-assert
15:1	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x093C			Register Name: SCR_CLK_REG
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	SCR_GATING Gating Clock For SCR 0: Mask 1: Pass

3.3.6.29 0x0940 SPI0 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0940			Register Name: SPI0_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SCLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON SCLK = Clock Source/M/N.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: DCXO24M 001: PLL_PERI(1X) 010: PLL_PERI(2X) 011: PLL_AUDIO0(DIV2) 100: PLL_AUDIO0(DIV5) The others: Reserved
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M (M= FACTOR_M +1) FACTOR_M is from 0 to 15.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.6.30 0x0944 SPI1 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0944			Register Name: SPI1_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	SCLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON SCLK = Clock Source/M/N.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: DCXO24M 001: PLL_PERI(1X) 010: PLL_PERI(2X) 011: PLL_AUDIO0(DIV2) 100: PLL_AUDIO0(DIV5) 1XX:/
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M (M= FACTOR_M +1) FACTOR_M is from 0 to 15.

3.3.6.31 0x096C SPI Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x096C			Register Name: SPI_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x096C			Register Name: SPI_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
17	R/W	0x0	SPI1_RST SPI1 Reset 0: Assert 1: De-assert
16	R/W	0x0	SPI0_RST SPI0 Reset 0: Assert 1: De-assert
15:2	/	/	/
1	R/W	0x0	SPI1_GATING Gating Clock for SPI1 0: Mask 1: Pass
0	R/W	0x0	SPI0_GATING Gating Clock for SPI0 0: Mask 1: Pass

3.3.6.32 0x0970 EMAC_25M Clock Register (Default: 0x0000_0000)

Offset: 0x0970			Register Name: EMAC_25M_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	EMAC_25M_CLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON CLK = PLL_PERI(1X)/24 = 25 MHz
30	R/W	0x0	EMAC_25M_CLK_SRC_GATING Gating the Source Clock of Special Clock It is for low power design. 0: Clock is OFF 1: Clock is ON
29:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.6.33 0x097C EMAC Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x097C			Register Name: EMAC_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	EMAC_RST EMAC Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	EMAC_GATING Gating Clock for EMAC 0: Mask 1: Pass

3.3.6.34 0x0990 IRRX Clock Register (Default Value: 0x0000_0000)

Offset: 0x0990			Register Name: IRRX_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	IRRT CLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON CLK = Clock Source/M/N.
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: RTC_32K 01: DCXO24M 10: PLL_PERI(1X) 11: PLL_AUDIO0(DIV2)
23:10	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0990			Register Name: IRRX_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M (M= FACTOR_M +1) FACTOR_M is from 0 to 15.

3.3.6.35 0x099C IRRX Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x099C			Register Name: IRRX_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	IRRX_RST IRRX Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	IRRX_GATING Gating Clock for IRRX 0: Mask 1: Pass

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.6.36 0x09C0 IRTX Clock Register (Default: 0x0000_0000)

Offset: 0x09C0			Register Name: IRTX_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	IRTX CLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON CLK = Clock Source/M/N.
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: RTC_32K 01: DCXO24M 10: PLL_PERI(1X) 11: PLL_AUDIO0(DIV2)
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M (M= FACTOR_M +1) FACTOR_M is from 0 to 15.

3.3.6.37 0x09CC IRTX Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x09CC			Register Name: IRTX_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	IRTX_RST IRTX Reset 0: Assert 1: De-assert

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x09CC			Register Name: IRTX_BGR_REG
Bit	Read/Write	Default/Hex	Description
15:1	/	/	/
0	R/W	0x0	IRTX_GATING Gating Clock for IRTX 0: Mask 1: Pass

3.3.6.38 0x0A10 I2S/PCM0 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0A10			Register Name: I2S/PCM0_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SCLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON SCLK = Clock Source/N/M.
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: PLL_AUDIO1(1X) 01: PLL_AUDIO1(4X) 10: PLL_AUDIO0(DIV2) 11: PLL_AUDIO0(DIV5)
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M (M= FACTOR_M +1) The value of FACTOR_M is from 0 to 31.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.6.39 0x0A14 I2S/PCM1 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0A14			Register Name: I2S/PCM1_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	SCLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON SCLK = Clock Source/N/M.
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: PLL_AUDIO1(1X) 01: PLL_AUDIO1(4X) 10: PLL_AUDIO0(DIV2) 11: PLL_AUDIO0(DIV5)
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M (M= FACTOR_M +1) FACTOR_M is from 0 to 31.

3.3.6.40 0x0A1C I2S/PCM Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x0A1C			Register Name: I2S/PCM_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17	R/W	0x0	I2S/PCM1_RST I2S/PCM1 Reset 0: Assert 1: De-assert

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0A1C			Register Name: I2S/PCM_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
16	R/W	0x0	I2S/PCM0_RST I2S/PCM0 Reset 0: Assert 1: De-assert
15:2	/	/	/
1	R/W	0x0	I2S/PCM1_GATING Gating Clock for I2S/PCM1 0: Mask 1: Pass
0	R/W	0x0	I2S/PCM0_GATING Gating Clock for I2S/PCM0 0: Mask 1: Pass

3.3.6.41 0x0A20 OWA Clock Register (Default Value: 0x0000_0000)

Offset: 0x0A20			Register Name: OWA_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	SCLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON SCLK = Clock Source/N/M.
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: PLL_AUDIO1(1X) 01: PLL_AUDIO1(4X) 10: PLL_AUDIO0(DIV2) 11: PLL_AUDIO0(DIV5)
23:10	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0A20			Register Name: OWA_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M (M= FACTOR_M +1) FACTOR_M is from 0 to 31.

3.3.6.42 0x0A2C OWA Bus Gating Reset Register (Default Value: 0x0000_0000)

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

3.3.6.43 0x0A70 USB0 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0A70			Register Name: USB0_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	SCLK_GATING_OHCI0 Gating Special Clock for OHCI0 0: Clock is OFF 1: Clock is ON

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0A70			Register Name: USB0_CLK_REG
Bit	Read/Write	Default/Hex	Description
30	R/W	0x0	USBPHY0_RST USB PHY0 Reset 0: Assert 1: De-assert
29	R/W	0x0	SCLK_GATING_USBPHY0 Gating Special Clock for USB PHY0 0: Clock is OFF 1: Clock is ON The SCLK is from DCXO24M.
28:26	/	/	/
25:24	R/W	0x0	OHCI0_12M_SRC_SEL OHCI0 12M Source Select 00: 12M divided from 48 MHz 01: 12M divided from 24 MHz 10: RTC_32K 11: /
23:0	/	/	/

3.3.6.44 0x0A74 USB1 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0A74			Register Name: USB1_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SCLK_GATING_OHCI1 Gating Special Clock for OHCI1 0: Clock is OFF 1: Clock is ON
30	R/W	0x0	USBPHY1_RST USB PHY1 Reset 0: Assert 1: De-assert
29	R/W	0x0	SCLK_GATING_USBPHY1 Gating Special Clock for USB PHY1 0: Clock is OFF 1: Clock is ON The SCLK is from DCXO24M.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0A74			Register Name: USB1_CLK_REG
Bit	Read/Write	Default/Hex	Description
28:26	/	/	/
25:24	R/W	0x0	OHCI0_12M_SRC_SEL OHCI0 12M Source Select 00: 12M divided from 48 MHz 01: 12M divided from 24 MHz 10: RTC_32K 11: /
23:0	/	/	/

3.3.6.45 0x0A8C USB Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x0A8C			Register Name: USB_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:25	/	/	/
24	R/W	0x0	USBDRD0_RST USBDRD0 Reset 0: Assert 1: De-assert
23:21	/	/	/
20	R/W	0x0	USBEHCI0_RST USBEHCI0 Reset 0: Assert 1: De-assert
19:18	/	/	/
17	R/W	0x0	USBOHCI1_RST USBOHCI1 Reset 0: Assert 1: De-assert
16	R/W	0x0	USBOHCI0_RST USBOHCI0 Reset 0: Assert 1: De-assert
15:9	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0A8C			Register Name: USB_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
8	R/W	0x0	USBDRD0_GATING Gating Clock for USBDRD0 0: Mask 1: Pass
7:5	/	/	/
4	R/W	0x0	USBEHCI0_GATING Gating Clock for USBEHCI0 0: Mask 1: Pass
3:2	/	/	/
1	R/W	0x0	USBOHCI1_GATING Gating Clock for USBOHCI1 0: Mask 1: Pass
0	R/W	0x0	USBOHCI0_GATING Gating Clock for USBOHCI0 0: Mask 1: Pass

3.3.6.46 0x0BF0 LEDC Clock Register (Default Value: 0x0000_0000)

Offset: 0x0BF0			Register Name: LEDC_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	SCLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON SCLK = Clock Source/M/N.
30:25	/	/	/
24	R/W	0x0	CLK_SRC_SEL Clock Source Select 0: DCXO24M 1: PLL_PERI(1X)
23:10	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0BF0			Register Name: LEDC_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
9:8	R/W	0x0	FACTOR_N. Factor N 00: 1 01: 2 10: 4 11: 8
7:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M (M= FACTOR_M +1) FACTOR_M is from 0 to 15.

3.3.6.47 0x0BFC LEDC Bus Gating Reset Register (Default: 0x0000_0000)

Offset: 0x0BFC			Register Name: LEDC_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	LEDC_RST LEDC Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	LEDC_GATING Gating Clock for LEDC 0: Mask 1: Pass

3.3.6.48 0x0F00 CCU Security Switch Register (Default Value: 0x0000_0000)

NOTE

In non-secure mode, the register is invalid.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0F00			Register Name: CCU_SEC_SWITCH_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2	R/W	0x0	MBUS_SEC MBUS Clock Register Security 0: Secure 1: Non-secure
1	R/W	0x0	BUS_SEC Bus Relevant Registers Security 0: Secure 1: Non-secure
0	R/W	0x0	PLL_SEC PLL Relevant Registers Security 0: Secure 1: Non-secure

3.3.7 CPUS Domain Register List

Module Name	Base Address
PRCM	0x07010000

Register Name	Offset	Description
AHBS_CFG_REG	0x0000	AHBS Configuration Register
APBS0_CFG_REG	0x000C	APBS0 Configuration Register
APBS1_CFG_REG	0x0010	APBS1 Configuration Register
GPADC_BGR_REG	0x00EC	GPADC Bus Gating Reset Register
THS_BGR_REG	0x00FC	THS Bus Gating Reset Register
R_PWM_CLK_REG	0x0130	R_PWM Clock Register
R_PWM_BGR_REG	0x013C	R_PWM Bus Gating Reset Register
AUDIO_CODEC_ADC_CLK_REG	0x0140	AUDIO_CODEC_ADC Clock Register
AUDIO_CODEC_DAC_CLK_REG	0x0144	AUDIO_CODEC_DAC Clock Register
AUDIO_CODEC_BGR_REG	0x014C	AUDIO_CODEC Bus Gating Reset Register
DMIC_CLK_REG	0x0150	DMIC Clock Register
DMIC_BGR_REG	0x015C	DMIC Bus Gating Reset Register
LRADC_BGR_REG	0x016C	LRADC Bus Gating Reset Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
R_I2S/PCM0_CLK_REG	0x0170	R_I2S/PCM0 Clock Register
R_I2S/PCM0_ASCR_CLK_REG	0x0174	R_I2S/PCM0_ASRC Clock Register
R_I2S/PCM_BGR_REG	0x017C	R_I2S/PCM Bus Gating Reset Register
R_UART_BGR_REG	0x018C	R_UART Bus Gating Reset Register
R_TWI_BGR_REG	0x019C	R_TWI Bus Gating Reset Register
R_IRRX_CLK_REG	0x01C0	R_IRRX Clock Register
R_IRRX_BGR_REG	0x01CC	R_IRRX Bus Gating Reset Register
R_MSGBOX_BGR_REG	0x01DC	R_MSGBOX Bus Gating Reset Register
R_SPINLOCK_BGR_REG	0x01EC	R_SPINLOCK Bus Gating Reset Register
RTC_BGR_REG	0x020C	RTC Bus Gating Reset Register
PLL_CPUX_CTRL_REG	0x1000	PLL_CPUX Control Register
PLL_PERI_CTRL_REG	0x1010	PLL_PERI Control Register
PLL_AUDIO0_CTRL_REG	0x1020	PLL_AUDIO0 Control Register
PLL_AUDIO1_CTRL_REG	0x1030	PLL_AUDIO1 Control Register
PLL_PERI_PAT0_CTRL_REG	0x1110	PLL_PERI Pattern0 Control Register
PLL_PERI_PAT1_CTRL_REG	0x1114	PLL_PERI Pattern1 Control Register
PLL_AUDIO0_PAT0_CTRL_REG	0x1120	PLL_AUDIO0 Pattern0 Control Register
PLL_AUDIO0_PAT1_CTRL_REG	0x1124	PLL_AUDIO0 Pattern1 Control Register
PLL_AUDIO1_PAT0_CTRL_REG	0x1130	PLL_AUDIO1 Pattern0 Control Register
PLL_AUDIO1_PAT1_CTRL_REG	0x1134	PLL_AUDIO1 Pattern1 Control Register
PLL_CPUX_BIAS_REG	0x1300	PLL_CPUX Bias Register
PLL_PERI0_BIAS_REG	0x1310	PLL_PERI0 Bias Register
PLL_AUDIO0_BIAS_REG	0x1320	PLL_AUDIO0 Bias Register
PLL_AUDIO1_BIAS_REG	0x1330	PLL_AUDIO1 Bias Register
PLL_CPUX_TUN_REG	0x1400	PLL_CPUX Tuning Register

3.3.8 CPUS Domain Register Description

3.3.8.1 0x0000 AHBS Configuration Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: AHBS_CFG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	/	/	/
26:24	R/W	0x0	AHBS_CLK_SRC_SEL AHBS Clock Source Select

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: AHBS_CFG_REG
Bit	Read/Write	Default/Hex	Description
			000: DCXO24M 001: RTC_32K 010: RC16M 011: PLL_PERI(2X) 100: PLL_AUDIO0(DIV2) Others: / CLK = Clock Source/CLK_DIV_RATIO_N/FACTOR_M.
23:10	/	/	/
9:8	R/W	0x0	CLK_DIV_RATIO_N Clock Post Divide Ratio (n) 00: 1 01: 2 10: 4 11: 8
7:2	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M.(M= FACTOR_M +1) FACTOR_M is from 0 to 31.

3.3.8.2 0x000C APBS0 Configuration Register (Default Value: 0x0000_0000)

Offset: 0x000C			Register Name: APBS0_CFG_REG
Bit	Read/Write	Default/Hex	Description
31:27	/	/	/
26:24	R/W	0x0	APBS0_CLK_SRC_SEL APBS0 Clock Source Select 000: DCXO24M 001: RTC_32K 010: RC16M 011: PLL_PERI(2X) 100: PLL_AUDIO0(DIV2) Others: / CLK = Clock Source/CLK_DIV_RATIO_N/FACTOR_M.
23:10	/	/	/
9:8	R/W	0x0	CLK_DIV_RATIO_N Clock Post Divide Ratio (n) 00: 1

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x000C			Register Name: APBS0_CFG_REG
Bit	Read/Write	Default/Hex	Description
			01: 2 10: 4 11: 8
7:2	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M.(M= FACTOR_M +1) FACTOR_M is from 0 to 31.

3.3.8.3 0x0010 APBS1 Configuration Register (Default Value: 0x0000_0000)

Offset: 0x0010			Register Name: APBS1_CFG_REG
Bit	Read/Write	Default/Hex	Description
31:27	/	/	/
26:24	R/W	0x0	APBS1_CLK_SRC_SEL APBS1 Clock Source Select 000: DCXO24M 001: RTC_32K 010: RC16M 011: PLL_PERI(2X) 100: PLL_AUDIO0(DIV2) Others: / CLK = Clock Source/CLK_DIV_RATIO_N/ FACTOR_M.
23:10	/	/	/
9:8	R/W	0x0	CLK_DIV_RATIO_N Clock Post Divide Ratio (n) 00: 1 01: 2 10: 4 11: 8
7:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M.(M= FACTOR_M +1) FACTOR_M is from 0 to 31.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.8.4 0x00EC GPADC Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x00EC			Register Name: GPADC_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	GPADC_RST GPADC Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	GPADC_GATING Gating Clock For GPADC 0: Mask 1: Pass

3.3.8.5 0x00FC THS Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x00FC			Register Name: THS_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	THS_RST THS Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	THS_GATING Gating Clock For THS 0: Mask 1: Pass

3.3.8.6 0x0130 R_PWM Clock Register (Default Value: 0x0000_0000)

Offset: 0x0130			Register Name: R_PWM_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SCLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON SCLK = Clock Source.
30:26	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0130			Register Name: R_PWM_CLK_REG
Bit	Read/Write	Default/Hex	Description
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: DCXO24M 01: RTC_32K 10: RC16M 11:/
23:0	/	/	/

3.3.8.7 0x013C R_PWM Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x013C			Register Name: R_PWM_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	R_PWM_RST R_PWM Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	R_PWM_GATING Gating Clock For R_PWM 0: Mask 1: Pass

3.3.8.8 0x0140 AUDIO_CODEC_ADC Clock Register (Default Value: 0x0000_0000)

Offset: 0x0140			Register Name: AUDIO_CODEC_ADC_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SCLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON SCLK = Clock Source/M/N.
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: PLL_AUDIO0(DIV5)

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0140			Register Name: AUDIO_CODEC_ADC_CLK_REG
Bit	Read/Write	Default/Hex	Description
			01: PLL_AUDIO0(DIV2) 10: PLL_AUDIO1(1X) 11: PLL_AUDIO1(4X)
23:10	/	/	/
9:8	R/W	0x0	CLK_DIV_RATIO_N Clock Pre Divide Ratio (n) 00: /1 01: /2 10: /4 11: /8
7:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M (M = FACTOR_M +1) FACTOR_M is from 0 to 31.

3.3.8.9 0x0144 AUDIO_CODEC_DAC Clock Register (Default: 0x0000_0000)

Offset: 0x0144			Register Name: AUDIO_CODEC_DAC_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SCLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON SCLK = Clock Source/M/N
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: PLL_AUDIO0(DIV5) 01: PLL_AUDIO0(DIV2) 10: PLL_AUDIO1(1X) 11: PLL_AUDIO1(4X)
23:10	/	/	/
9:8	R/W	0x0	CLK_DIV_RATIO_N Clock Pre Divide Ratio (n) 00: /1 01: /2

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0144			Register Name: AUDIO_CODEC_DAC_CLK_REG
Bit	Read/Write	Default/Hex	Description
			10: /4 11: /8
7:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M (M = FACTOR_M +1) FACTOR_M is from 0 to 31.

3.3.8.10 0x014C AUDIO_CODEC Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x014C			Register Name: AUDIO_CODEC_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	AUDIO_CODEC_RST AUDIO_CODEC Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	AUDIO_CODEC_GATING Gating Clock For AUDIO_CODEC 0: Mask 1: Pass

3.3.8.11 0x0150 DMIC Clock Register (Default: 0x0000_0000)

Offset: 0x0150			Register Name: DMIC_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SCLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON SCLK = Clock Source/M/N
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: PLL_AUDIO0(DIV5)

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0150			Register Name: DMIC_CLK_REG
Bit	Read/Write	Default/Hex	Description
			01: PLL_AUDIO0(DIV2) 10: PLL_AUDIO1(1X) 11: PLL_AUDIO1(4X)
23:10	/	/	/
9:8	R/W	0x0	CLK_DIV_RATIO_N Clock Pre Divide Ratio (n) 00: /1 01: /2 10: /4 11: /8
7:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M (M = FACTOR_M +1) FACTOR_M is from 0 to 31.

3.3.8.12 0x015C DMIC Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x015C			Register Name: DMIC_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	DMIC_RST DMIC Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	DMIC_GATING Gating Clock For DMIC 0: Mask 1: Pass

3.3.8.13 0x016C LRADC Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x016C			Register Name: LRADC_BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x016C			Register Name: LRADC_BGR_REG
Bit	Read/Write	Default/Hex	Description
16	R/W	0x0	LRADC_RST. LRADC Reset. 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	LRADC_GATING. Gating Clock For LRADC 0: Mask 1: Pass

3.3.8.14 0x0170 R_I2S/PCM0 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0170			Register Name: R_I2S/PCM0_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SCLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON SCLK = Clock Source/M/N.
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: PLL_AUDIO0(DIV5) 01: PLL_AUDIO0(DIV2) 10: PLL_AUDIO1(1X) 11: PLL_AUDIO1(4X)
23:10	/	/	/
9:8	R/W	0x0	CLK_DIV_RATIO_N Clock Pre Divide Ratio (n) 00: /1 01: /2 10: /4 11: /8
7:5	/	/	/
4:0	R/W	0x0	FACTOR_M

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0170			Register Name: R_I2S/PCM0_CLK_REG
Bit	Read/Write	Default/Hex	Description
			Factor M.(M= FACTOR_M +1) FACTOR_M is from 0 to 31.

3.3.8.15 0x0174 R_I2S/PCM0_ASRC Clock Register (Default Value: 0x0000_0000)

Offset: 0x0174			Register Name: R_I2S/PCM0_ASRC_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SCLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON SCLK = Clock Source/M/N.
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: PLL_AUDIO0(DIV5) 01: PLL_AUDIO0(DIV2) 10: PLL_AUDIO1(1X) 11: PLL_AUDIO1(4X)
23:10	/	/	/
9:8	R/W	0x0	CLK_DIV_RATIO_N Clock Pre Divide Ratio (n) 00: /1 01: /2 10: /4 11: /8
7:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M.(M= FACTOR_M +1) FACTOR_M is from 0 to 31.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.8.16 0x017C R_I2S/PCM Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x017C			Register Name: R_I2S/PCM_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	R_I2S/PCM0_RST R_I2S/PCM0 Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	R_I2S/PCM0_GATING Gating Clock For R_I2S/PCM0 0: Mask 1: Pass

3.3.8.17 0x018C R_UART Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x018C			Register Name: R_UART_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	R_UART0_RST R_UART0 Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	R_UART0_GATING Gating Clock For R_UART0 0: Mask 1: Pass

3.3.8.18 0x019C R_TWI Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x019C			Register Name: R_TWI_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	R_TWI0_RST R_TWI0 Reset

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x019C			Register Name: R_TWI_BGR_REG
Bit	Read/Write	Default/Hex	Description
			0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	R_TWI0_GATING Gating Clock For R_TWI0 0: Mask 1: Pass

3.3.8.19 0x01C0 R_IRRX Clock Register (Default Value: 0x0000_0000)

Offset: 0x01C0			Register Name: R_IRRX_CLK_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SCLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON SCLK = Clock Source/M/N.
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: RTC_32K 01: DCXO24M 10:/ 11:/
23:10	/	/	/
9:8	R/W	0x0	CLK_DIV_RATIO_N Clock Pre Divide Ratio (n) 00: /1 01: /2 10: /4 11: /8
7:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M.(M= FACTOR_M +1) FACTOR_M is from 0 to 31.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.8.20 0x01CC R_IRRX Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x01CC			Register Name: R_IRRX_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	R_IRRX_RST R_IRRX Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	R_IRRX_GATING Gating Clock For R_IRRX 0: Mask 1: Pass

3.3.8.21 0x01DC R_MSGBOX Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x01DC			Register Name: R_MSGBOX_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	R_MSGBOX_RST R_MSGBOX Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	R_MSGBOX_GATING Gating Clock For R_MSGBOX 0: Mask 1: Pass

3.3.8.22 0x01EC R_SPINLOCK Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x01EC			Register Name: R_SPINLOCK_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x01EC			Register Name: R_SPINLOCK_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
16	R/W	0x0	R_SPINLOCK_RST R_SPINLOCK Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	R_SPINLOCK_GATING Gating Clock For R_SPINLOCK 0: Mask 1: Pass

3.3.8.23 0x020C RTC Bus Gating Reset Register (Default Value: 0x0001_0001)

Offset: 0x020C			Register Name: RTC_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x1	RTC_RST RTC Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x1	RTC_GATING Gating Clock For RTC 0: Mask 1: Pass

3.3.8.24 0x1000 PLL_CPUX Control Register (Default Value: 0x0A00_1000)

Offset: 1000			Register Name: PLL_CPUX_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	PLL_ENABLE 0: Disabled 1: Enabled PLL_CPUX = 24 MHz*N **Note: The range of PLL_CPUX output frequency is from 200 MHz to 3 GHz, and the default value is 408 MHz.**

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 1000			Register Name: PLL_CPUX_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
30	/	/	/
29	R/W	0x0	LOCK_ENABLE Lock Enable 0: Disabled 1: Enabled
28	R	0x0	LOCK 0: Unlocked 1: Locked (It indicates that the PLL has been stable.)
27	R/W	0x1	PLL_OUTPUT_ENABLE Enable or disable the output of the PLL. 0: Disabled 1: Enabled
26:24	R/W	0x2	PLL_LOCK_TIME Indicates the step amplitude that the clock switch from one frequency to another.
23:16	/	/	/
15:8	R/W	0x10	PLL_FACTOR_N PLL Factor N N = PLL_FACTOR_N + 1 PLL_FACTOR_N is from 0 to 254. In application, the value of PLL_FACTOR_N should be no less than 11.
7:2	/	/	/
1:0	R/W	0x0	PLL_FACTOR_M. PLL Factor M M = PLL_FACTOR_M + 1 The value of PLL_FACTOR_M is from 0 to 3. **Note: The bit is only for testing.**

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.8.25 0x1010 PLL_PERI Control Register (Default Value: 0x0821_6300)

Offset: 0x1010			Register Name: PLL_PERI_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	PLL_ENABLE 0: Disabled 1: Enabled The PLL_PERI(2X) = 24MHz*N /M /P0 PLL_PERI(1X) = 24MHz*N/M/P0/2 PLL_PERI(800M) = 24MHz*N/M/P1 By default, PLL_PERI(2X) = 1.2 GHz, PLL_PERI(1X) = 600 MHz, and PLL_PERI(800M) = 80 MHz. **Note: Do not modify the output value of PLL_PERI(2X), it should be locked as 1.2 GHz.**
30	/	/	/
29	R/W	0x0	LOCK_ENABLE Lock Enable 0: Disabled 1: Enabled
28	R	0x0	LOCK 0:Unlocked 1: Locked (It indicates that the PLL has been stable.)
27	R/W	0x1	PLL_OUTPUT_ENABLE Enable or disable the output of the PLL. 0: Disabled 1: Enabled
26:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE 0: Disabled 1: Enabled
23	/	/	/
22:20	R/W	0x2	PLL_OUTPUT_DIV_P1 PLL Output Div P1 P1 = PLL_OUTPUT_DIV_P1 + 1 The value of PLL_OUTPUT_DIV_P1 is from 0 to 7.
19	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x1010			Register Name: PLL_PERI_CTRL_REG
Bit	Read/Write	Default/Hex	Description
18:16	R/W	0x1	PLL_OUTPUT_DIV_P0 PLL Output Div P0 P0 = PLL_OUTPUT_DIV_P0 + 1 The value of PLL_OUTPUT_DIV_P0 is from 0 to 7.
15:8	R/W	0x63	PLL_FACTOR_N PLL Factor N N = PLL_FACTOR_N +1 The value of PLL_FACTOR_N is from 0 to 254. In application, the value of PLL_FACTOR_N should be no less than 11.
7:2	/	/	/
1	R/W	0x0	PLL_INPUT_DIV_M PLL Input Div M M = PLL_INPUT_DIV_M + 1 The value of PLL_INPUT_DIV_M is from 0 to 1.
0	/	/	/

3.3.8.26 0x1020 PLL_AUDIO0 Control Register (Default Value: 0x0841_7F00)

Offset: 0x1020			Register Name: PLL_AUDIO0_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	PLL_ENABLE 0: Disabled 1: Enabled The PLL_AUDIO0 = 24MHz*N/M The PLL_AUDIO0(DIV2) = 24MHz*N/M/P0 The PLL_AUDIO0(DIV5) = 24MHz*N/M/P1 The range of 24MHz*N/M should be from 180 MHz to 3.5 GHz. By default, PLL_AUDIO0 is 3072 MHz, PLL_AUDIO0(DIV2) is 1536 MHz, and PLL_AUDIO0(DIV5) is 614.4 MHz (24.576 MHz*25).
30	/	/	/
29	R/W	0x0	LOCK_ENABLE Lock Enable 0: Disabled 1: Enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x1020			Register Name: PLL_AUDIO0_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
28	R	0x0	LOCK 0:Unlocked 1: Locked (It indicates that the PLL has been stable.) Only effective when bit[29] is 1.
27	R/W	0x1	PLL_OUTPUT_ENABLE Enable or disable the output of the PLL. 0: Disabled 1: Enabled
26:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE. Enable or disable Spread Spectrum and Decimal Frequency Division. 0: Disabled. 1: Enabled.
23	/	/	/
22:20	R/W	0x4	PLL_OUTPUT_DIV_P1 PLL Output Div P1 P1 = PLL_OUTPUT_DIV_P1 + 1 The value of PLL_OUTPUT_DIV_P1 is from 0 to 7.
19	/	/	/
18:16	R/W	0x1	PLL_OUTPUT_DIV_P0 PLL Output Div P0 P0 = PLL_OUTPUT_DIV_P0 + 1 The value of PLL_OUTPUT_DIV_P0 is from 0 to 7.
15:8	R/W	0x7F	PLL_FACTOR_N PLL Factor N N = PLL_FACTOR_N +1 PLL_FACTOR_N is from 0 to 254. In application, the value of PLL_FACTOR_N should be no less than 11.
7:2	/	/	/
1	R/W	0x0	PLL_INPUT_DIV_M PLL Input Div M M = PLL_INPUT_DIV_M + 1 PLL_INPUT_DIV_M is from 0 to 1.
0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.8.27 0x1030 PLL_AUDIO1 Control Register (Default Value: 0x0814_2A01)

Offset: 0x1030			Register Name: PLL_AUDIO1_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	PLL_ENABLE 0: Disabled 1: Enabled The PLL_AUDIO1(4X) = 24MHz*N/M0/M1/P The PLL_AUDIO1(2X) = (24MHz*N/M0/M1) /P/2 The PLL_AUDIO1(1X) = (24MHz*N/M0/M1) /P/4 **Note: 7.5 ≤ N/M0/M1 ≤ 125, and 12 ≤ N.** The range of 24MHz*N/M0/M1 should be from 180 MHz to 3.0 GHz. By default, PLL_AUDIO1(1X) is 24.5714 MHz.
30	/	/	/
29	R/W	0x0	LOCK_ENABLE Lock Enable 0: Disabled 1: Enabled
28	R	0x0	LOCK 0: Unlocked 1: Locked (It indicates that the PLL has been stable.) Only effective when bit[29] is 1.
27	R/W	0x1	PLL_OUTPUT_ENABLE Enable or disable the output of the PLL. 0: Disabled 1: Enabled
26:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE Enable or disable Spread Spectrum and Decimal Frequency Division. 0: Disabled. 1: Enabled.
23:22	/	/	/
21:16	R/W	0x14	PLL_POST_DIV_P PLL Post-div P P = PLL_POST_DIV_P +1 The value of PLL_POST_DIV_P is from 0 to 63.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x1030			Register Name: PLL_AUDIO1_CTRL_REG
Bit	Read/Write	Default/Hex	Description
15:8	R/W	0x2A	PLL_FACTOR_N PLL Factor N N = PLL_FACTOR_N +1 The value of PLL_FACTOR_N is from 0 to 254. In application, the value of PLL_FACTOR_N should be no less than 11.
7:2	/	/	/
1	R/W	0x0	PLL_INPUT_DIV_M1 PLL Input Div M1 M1 = PLL_INPUT_DIV_M1 + 1 PLL_INPUT_DIV_M1 is from 0 to 1.
0	R/W	0x1	PLL_OUTPUT_DIV_M0 PLL Output Div M0 M0 = PLL_OUTPUT_DIV_M0 + 1 PLL_OUTPUT_DIV_M0 is from 0 to 1.

3.3.8.28 0x1110 PLL_PERI Pattern0 Control Register (Default Value: 0x0000_0000)

Offset: 0x1110			Register Name: PLL_PERI_PAT0_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SIG_DELT_PAT_EN Sigma-Delta Pattern Enable
30:29	R/W	0x0	SPR_FREQ_MODE Spread Frequency Mode 00: DC = 0 01: DC = 1 10: Triangular (1bit) 11: Triangular (nbit)
28:20	R/W	0x0	WAVE_STEP Wave Step
19	R/W	0x0	SDM_CLK_SEL SDM Clock Select 0: 24 MHz 1: 12 MHz **Note: Set this bit to 1 when PLL_INPUT_DIV_M1 is 1.**

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x1110			Register Name: PLL_PERI_PAT0_CTRL_REG
Bit	Read/Write	Default/Hex	Description
18:17	R/W	0x0	FREQ Frequency 00: 31.5 KHz 01: 32 KHz 10: 32.5 KHz 11: 33 KHz
16:0	R/W	0x0	WAVE_BOT Wave Bottom

3.3.8.29 0x1114 PLL_PERI Pattern1 Control Register (Default Value: 0x0000_0000)

Offset: 0x1114			Register Name: PLL_PERI_PAT1_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:25	/	/	/
24	R/W	0x0	DITHER_EN
23:21	/	/	/
20	R/W	0x0	FRAC_EN
19:17	/	/	/
16:0	R/W	0x0	FRAC_IN

3.3.8.30 0x1120 PLL_AUDIO0 Pattern0 Control Register (Default Value: 0x0000_0000)

Offset: 0x1120			Register Name: PLL_AUDIO0_PAT0_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	SIG_DELT_PAT_EN Sigma-Delta Pattern Enable
30:29	R/W	0x0	SPR_FREQ_MODE Spread Frequency Mode 00: DC = 0 01: DC = 1 10: Triangular (1bit) 11: Triangular (nbit)
28:20	R/W	0x0	WAVE_STEP Wave Step

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x1120			Register Name: PLL_AUDIO0_PAT0_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
19	R/W	0x0	SDM_CLK_SEL SDM Clock Select 0: 24 MHz 1: 12 MHz **Note: Set this bit to 1 when PLL_INPUT_DIV_M1 is 1.**
18:17	R/W	0x0	FREQ Frequency 00: 31.5 kHz 01: 32 kHz 10: 32.5 kHz 11: 33 kHz
16:0	R/W	0x0	WAVE_BOT Wave Bottom

3.3.8.31 0x1124 PLL_AUDIO0 Pattern1 Control Register (Default Value: 0x0000_0000)

Offset: 0x1124			Register Name: PLL_AUDIO0_PAT1_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:25	/	/	/
24	R/W	0x0	DITHER_EN
23:21	/	/	/
20	R/W	0x0	FRAC_EN
19:17	/	/	/
16:0	R/W	0x0	FRAC_IN

3.3.8.32 0x1130 PLL_AUDIO1 Pattern0 Control Register (Default Value: 0x0000_0000)

Offset: 0x1130			Register Name: PLL_AUDIO1_PAT0_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	SIG_DELT_PAT_EN Sigma-Delta Pattern Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x1130			Register Name: PLL_AUDIO1_PAT0_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
30:29	R/W	0x0	SPR_FREQ_MODE Spread Frequency Mode 00: DC = 0 01: DC = 1 10: Triangular (1bit) 11: Triangular (nbit)
28:20	R/W	0x0	WAVE_STEP Wave Step
19	R/W	0x0	SDM_CLK_SEL SDM Clock Select 0: 24 MHz 1: 12 MHz **Note: Set this bit to 1 when PLL_INPUT_DIV_M1 is 1.**
18:17	R/W	0x0	FREQ Frequency 00: 31.5 kHz 01: 32 kHz 10: 32.5 kHz 11: 33 kHz
16:0	R/W	0x0	WAVE_BOT Wave Bottom

3.3.8.33 0x1134 PLL_AUDIO1 Pattern1 Control Register (Default Value: 0x0000_0000)

Offset: 0x1134			Register Name: PLL_AUDIO1_PAT1_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:25	/	/	/
24	R/W	0x0	DITHER_EN
23:21	/	/	/
20	R/W	0x0	FRAC_EN
19:17	/	/	/
16:0	R/W	0x0	FRAC_IN

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.3.8.34 0x1300 PLL_CPUX Bias Register (Default Value: 0x8010_0000)

Offset: 0x1300			Register Name: PLL_CPUX_BIAS_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x1	VCO_RST VCO reset in
30:21	/	/	/
20:16	R/W	0x10	PLL_BIAS_CURRENT PLL current bias control [4:0], CPU_CP
15:0	/	/	/

3.3.8.35 0x1310 PLL_PERI0 Bias Register (Default Value: 0x0003_0000)

Offset: 0x1310			Register Name: PLL_PERI0_BIAS_REG
Bit	Read/Write	Default/Hex	Description
31:21	/	/	/
20:16	R/W	0x3	PLL_BIAS_CTRL PLL bias control [4:0]
15:0	/	/	/

3.3.8.36 0x1320 PLL_AUDIO0 Bias Register (Default Value: 0x0003_0000)

Offset: 0x1320			Register Name: PLL_AUDIO0_BIAS_REG
Bit	Read/Write	Default/Hex	Description
31:21	/	/	/
20:16	R/W	0x3	PLL_BIAS_CTRL PLL bias control [4:0]
15:0	/	/	/

3.3.8.37 0x1330 PLL_AUDIO1 Bias Register (Default Value: 0x0003_0000)

Offset: 0x1330			Register Name: PLL_AUDIO1_BIAS_REG
Bit	Read/Write	Default/Hex	Description
31:21	/	/	/
20:16	R/W	0x3	PLL_BIAS_CTRL PLL bias control [4:0]

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x1330			Register Name: PLL_AUDIO1_BIAS_REG
Bit	Read/Write	Default/Hex	Description
15:0	/	/	/

3.3.8.38 0x1400 PLL_CPUX Tuning Register (Default Value: 0x4440_4000)

Offset: 0x1400			Register Name: PLL_CPUX_TUN_REG
Bit	Read/Write	Default/Hex	Description
31	/	/	/
30:28	R/W	0x4	VCO_RNG_CTRL VCO range control [2:0]
27	/	/	/
26:24	R/W	0x4	KVCO_GAIN_CTRL KVCO gain control [2:0]
23	/	/	/
22:16	R/W	0x40	CNT_INIT_CTRL Counter initial control [6:0]
15	R/W	0x0	C_OD0 C-REG-OD0 for verification
14:8	R/W	0x40	C_B_IN C-B-IN [6:0] for verification
7	R/W	0x0	C_OD1 C-REG-OD1 for verification
6:0	RO	0x0	C_B_OUT C-B-OUT [6:0] for verification

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.4 BROM System

3.4.1 Overview

The system has several ways to boot. It has an integrated on-chip BROM that is considered the primary program-loader. On the startup process, the R329 starts to fetch the first instruction from address 0x0, where is the BROM located at.

The BROM system is divided into two parts: the firmware exchange launch (FEL) module and the Media Boot module. FEL is responsible for writing the external data to the local NVM, and Media Boot is responsible for loading an effective and legitimate BOOT0 from NVM and running.

The BROM system includes the following features:

- Supports CPU0 boot process
- Supports mandatory upgrade process through USB
- Supports mandatory upgrade process through SD card (This requires that the flash on the device has no valid firmware)
- Supports the GPIO pin to select the boot media type
- Supports eFuse to select the boot media type
- Supports normal booting and secure booting
- Secure BROM loads only certified firmware
- Secure BROM ensures that the Secure Boot is in a trusted environment

3.4.2 Functional Descriptions

BROM configurations mainly involve selecting the boot medium and boot modes.

3.4.2.1 Selecting the Boot Medium

The BROM system supports the following boot media:

- SD card
- eMMC
- NAND FLASH
- SPI NOR FLASH

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

- SPI NAND FLASH

There are two ways to select the boot medium: GPIO Pin Select and eFuse Select. On startup, the BROM will read the state of BOOT_MODE, and decide whether the GPIO or eFuse to select the type of boot medium based on the state of BOOT_MODE. The BOOT_MODE is a bit in the SID module.

The following table shows BOOT_MODE Setting.

Table 3-5 BOOT_MODE Setting

BOOT_MODE	Boot_Select Type
0	GPIO boot select, indicates that the boot medium is decided by the value of the GPIO pin.
1	eFuse boot select, indicates that the boot medium is decided by the value of the eFuse type.

GPIO Boot Select

If the state of the BOOT_MODE is 0, the boot medium is decided by the value of the GPIO pin. The following table shows the boot medium priority when the value of the GPIO pin is 0 and 1. The boot medium priority describes the possibility that each medium to be selected as the boot medium. The BROM reads the boot0 of the medium with the highest priority first. If the medium does not exist or has any problems, the BROM will try the next medium. Otherwise, the medium will be selected as the boot medium.

Table 3-6 GPIO Boot Select Configuration

Pin_Boot_Select	Boot Medium Priority
0	SPI NOR -> SPI NAND -> eMMC -> RAW NAND -> SD card
1	RAW NAND -> eMMC -> SPI NOR -> SPI NAND -> SD card

NOTE

The status of the GPIO boot select pin is the bit[11] of the SYSCTRL_REG (0x03000024).

eFuse Boot Select

If the state of the BOOT_MODE is 1, the boot medium is decided by the value of eFuse_Boot_Select_Cfg. The eFuse_Boot_Select_Cfg is divided into 4 groups and each group is 3-bit. The following table shows the groups of eFUSE_Boot_Select.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Table 3-7 Groups of eFuse_Boot_Select

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

The four groups take effect with the following priority:

eFuse_Boot_Select_1 -> eFuse_Boot_Select_2 -> eFuse_Boot_Select_3 -> eFuse_Boot_Select_4

For example, eFuse_Boot_Select_2 will not take effect unless eFuse_Boot_Select_1 is set as 0x111, eFuse_Boot_Select_3 will not take effect unless eFuse_Boot_Select_2 is set as 0x111, etc.

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

SPI NOR -> SPI NAND -> eMMC -> NAND FLASH -> SD card

The value 0x111 of eFuse_Boot_Select_n (n = 1 to 3) indicates the boot medium is decided by the value of eFuse_Boot_Select_(n + 1). For eFuse_Boot_Select_4, if the value is 0x111, the BROM will select the boot medium in the Try mode.

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

Table 3-8 eFuse Boot Select Setting

eFuse_Boot_Select_n	Boot Medium Priority
000	Select the boot medium in Try mode.
001	SLC NAND -> MLC NAND
010	EMMC_USER -> EMMC_BOOT
011	SPI NOR
100	SPI NAND
101	MLC NAND -> SLC NAND
110	MMC_BOOT -> EMMC_USER
111	When n is 1 to 3: The boot medium is decided by the value of **eFuse_Boot_Select_(n + 1)**. When n is 4: Select the boot medium in Try mode.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.4.2.2 Selecting the Boot Modes

For SoCs that have implemented and enabled the ARM TrustZone technology, there are two boot modes: Normal BROM Mode and Secure BROM Mode. Secure BROM Mode is designed to protect against the potential threat that attackers modify the code or data areas in programmable memory.

On startup, the BROM will select the boot mode according to the value of the Secure Enable bit. If the value of Secure Enable bit is 0, the system will boot in Normal BROM Mode. Otherwise, it will boot in Secure BROM Mode.

NOTE

The ARM TrustZone feature is the minimal security functionality an ARM-TrustZone-technology-based system must implement for its Trusted Boot. You can enable the ARM TrustZone feature by configuring the Secure Enable Bit as enabled.

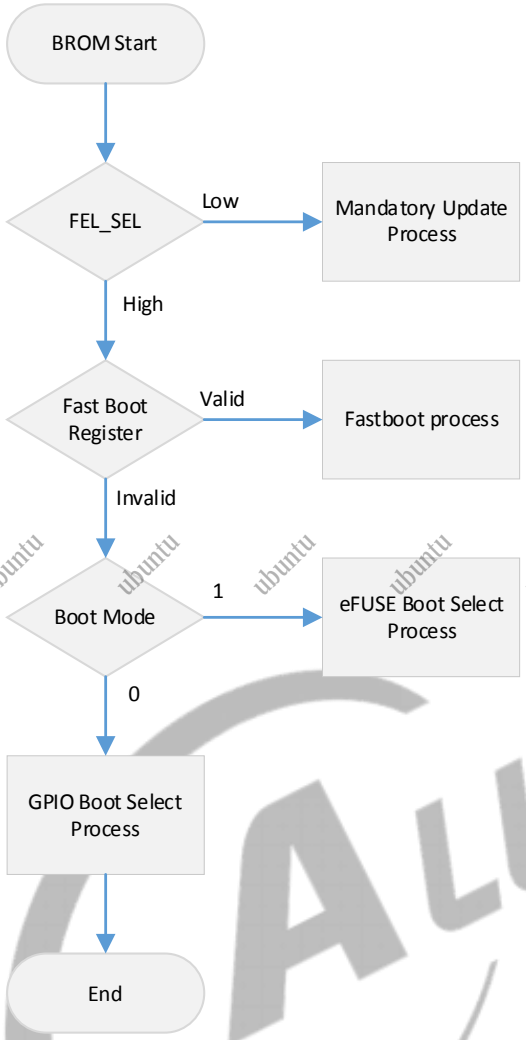
Normal BROM Mode

In Normal BROM Mode, the system boot starts from CPU0, and then the BROM will read the state of the FEL pin. If the FEL pin is high, the system will jump to the fast boot process. If it is low, the system will jump to the mandatory upgrade process.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

The following figure shows the boot process in Normal BROM Mode.

Figure 3-7 Boot Process in Normal BROM Mode



Secure BROM Mode

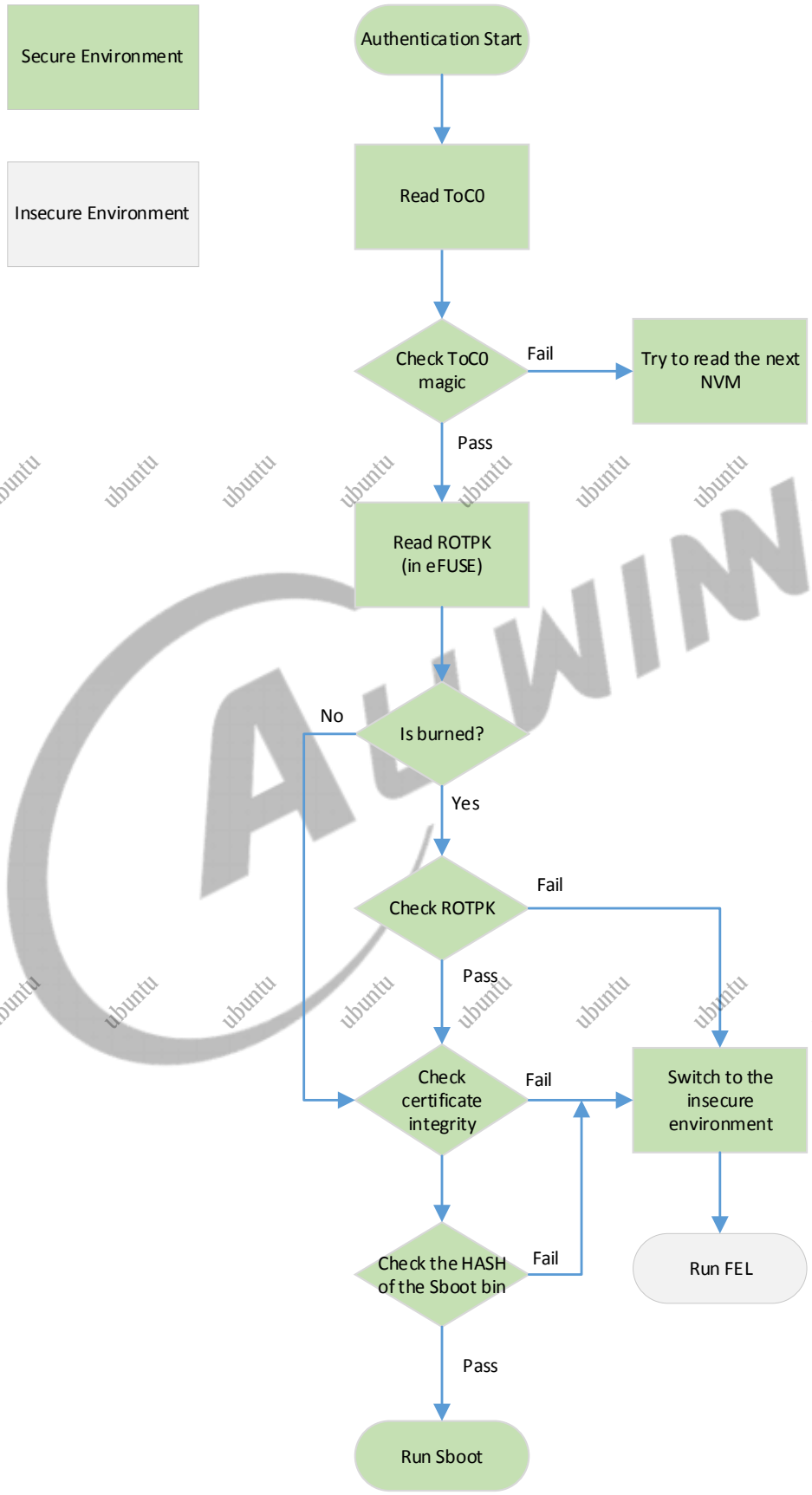
The process of selecting the boot medium in Secure Boot Mode is the same as that in Normal Boot Mode.

In Secure Boot Mode, after the boot medium is selected, the system additionally runs the Security Boot software to authenticate the Sboot bin file.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

The following figure shows the authentication process.

Figure 3-8 Authentication Process in Secure BROM Mode



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Secure BROM Requirements

To boot in Secure BROM Mode, the system needs to meet the following requirements:

- Supports X509 certificate

  The certificate is used to check whether the Security Boot software is modified or replaced. Before running the Security Boot software, the system checks the integrity of the certificate make sure the software has not been modified or replaced.

- Supports cryptographic algorithms

  - SHA-256

  - RSA-2048

  The system uses the Crypto Engine (CE) hardware module to accelerate the speed of encryption and decryption. The use of standard cryptography ensures the reliability of the firmware images. The reliable firmware image ensures that the system security state can be as expected.

- Supports OTP/eFuse

3.4.2.3 Mandatory Upgrade Process

If the FEL pin is detected to pull low, the system will jump to the mandatory upgrade process. The following figure shows the mandatory upgrade process.

Figure 3-9 Mandatory Upgrade Process

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

NOTE

The FEL address of the Normal BROM is 0x20.

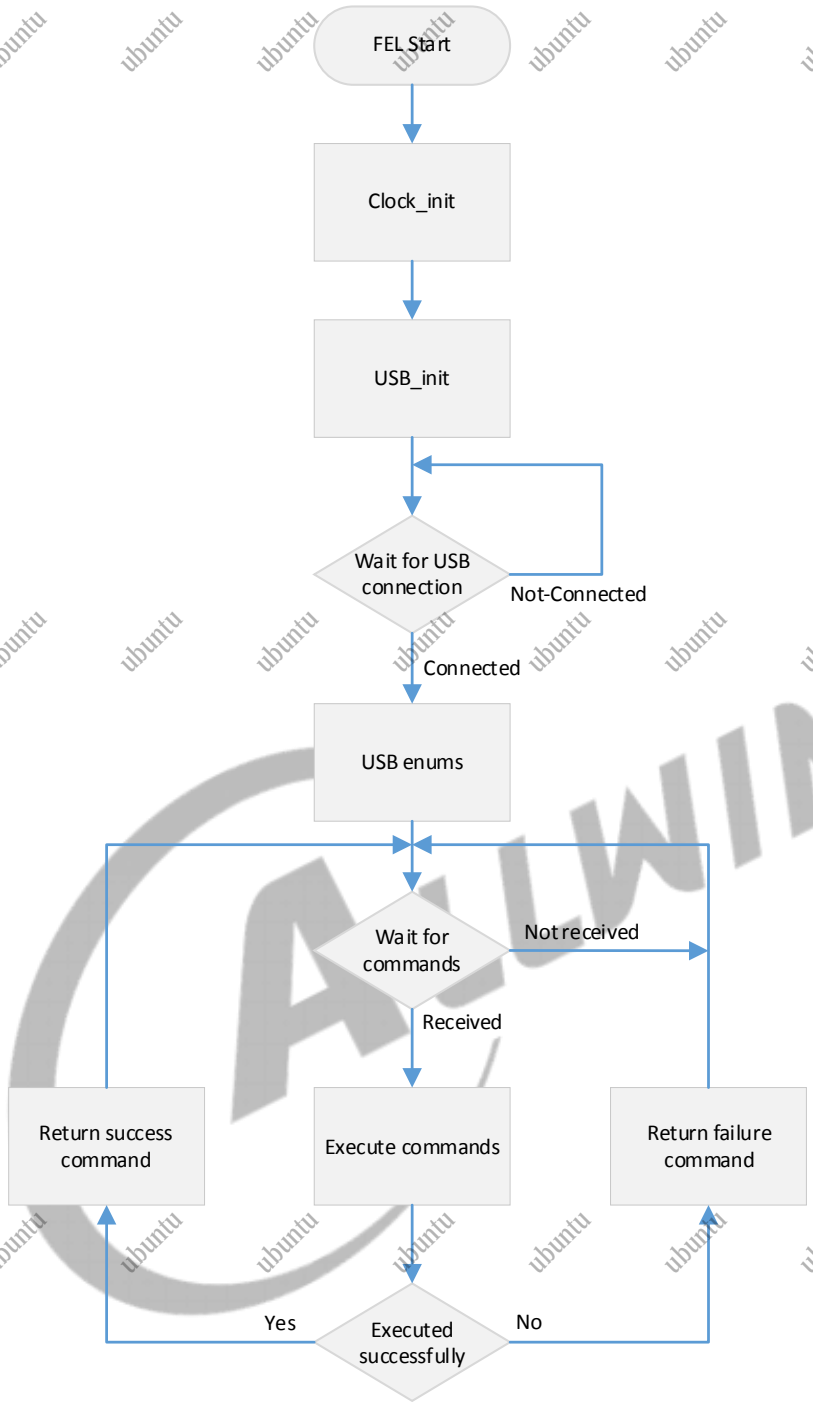
The FEL address of the Secure BROM is 0x64.

3.4.2.4 FEL Process

When the system chooses to enter the Mandatory Upgrade Process, the system will jump to the FEL process. The following figure shows the FEL upgrade process.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 3-10 USB FEL Process



3.4.2.5 Fast Boot Process

If the value of the Fast Boot register (0x07090120) in RTC module is not zero, the system will enter the Fast Boot Process. The following table shows the boot medium priority for different values of the Fast Boot register.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Table 3-9 Fast Boot Select Setting

Reg_bit[31:28]	Boot Medium Priority
1	MLC NAND -> SLC NAND -> TRY
2	EMMC_USER -> EMMC_BOOT -> TRY
3	SPI NOR -> TRY
4	SPI NAND -> TRY
5	EMMC_BOOT -> EMMC_USER -> TRY
6	SLC NAND -> MLC NAND -> TRY
7	TRY

NOTE

The Fast Boot register bit[27:0] is used to record the media information.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.5 System Configuration

3.5.1 Overview

The system configuration module is used to configure parameters for system domain, such as SRAM, CPU, PLL, BROM, and so on.

3.5.2 Register List

Module Name	Base Address
SYS_CFG	0x03000000

Register Name	Offset	Description
VER_REG	0x0024	Version Register
EMAC_EPHY_CLK_REG0	0x0030	EMAC-EPHY Clock Register 0
BROM_OUTPUT_REG	0x00A4	BROM Output Register

3.5.3 Register Description

3.5.3.1 0x0024 Version Register (Default Value: UDF)

Offset: 0x0024			Register Name: VER_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:12	/	/	/
11	R	UDF	BOOT_SEL_PAD_STA The value of this bit decides the priority order for each media type to be selected as the boot media. 0: SPI_NOR -> SPI_NAND -> eMMC -> RAW_NAND -> SD card 1: RAW_NAND -> eMMC -> SPI_NOR -> SPI_NAND -> SD card
10:9	/	/	/
8	R	UDF	FEL_SEL_PAD_STA Fel_Select_Pin_Status 0: Run_FEL 1: Try Media Boot
7:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.5.3.2 0x0030 EMAC-EPHY Clock Register 0 (Default Value: 0x0005_8000)

Offset: 0x0030			Register Name: EMAC_EPHY_CLK_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x0	BPS_EFUSE
27	R/W	0x0	XMII_SEL 0: Internal SMI and MII 1: External SMI and MII
26:25	R/W	0x0	EPHY_MODE Operation Mode Selection 00: Normal Mode 01: Sim Mode 10: AFE Test Mode 11: Reserved
24:20	R/W	0x0	PHY_ADDR PHY Address
19	R/W	0x0	Reserved
18	R/W	0x1	CLK_SEL 0: 25 MHz 1: 24 MHz
17	R/W	0x0	LED_POL 0: High active 1: Low active
16	R/W	0x1	SHUTDOWN 0: Power up 1: Shutdown
15	R/W	0x1	PHY_SELECT 0: External PHY 1: Internal PHY
14	/	/	/
13	R/W	0x0	RMII_EN 0: Disable RMII Module 1: Enable RMII Module This bit is prior to bit[2]. When this bit is asserted, the MII and RGMII interfaces will be both disabled.
12:10	R/W	0x0	ETXDC Configure EMAC Transmit Clock Delay Chain

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0030			Register Name: EMAC_EPHY_CLK_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
9:5	R/W	0x0	ERXDC Configure EMAC Receive Clock Delay Chain
4	R/W	0x0	ERXIE Enable EMAC Receive Clock Invertor 0: Disabled 1: Enabled
3	R/W	0x0	ETXIE Enable EMAC Transmit Clock Invertor 0: Disabled 1: Enabled
2	R/W	0x0	EPIT EMAC PHY Interface Type 0: MII 1: RGMII Only effective when bit[13] is de-asserted.
1:0	R/W	0x0	ETCS EMAC Transmit Clock Source 00: Transmit clock source for MII 01: External transmit clock source for GMII and RGMII 10: Internal transmit clock source for GMII and RGMII 11: Reserved

3.5.3.3 0x00A4 BROM Output Register (Default Value: 0x0000_0000)

Offset: 0x00A4			Register Name: BROM_OUTPUT_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1	R/W	0x0	BROM_OUTPUT_VALUE 0: U-Boot pin output 0 1: U-Boot pin output 1
0	R/W	0x0	BROM_OUTPUT_ENABLE 0: Disabled U-Boot pin output 1: Enabled U-Boot pin output

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.6 Timer

3.6.1 Overview

The timer module implements the timing and counting functions. The timer module includes timer0, timer1, watchdog and audio video synchronization (AVS).

The timer0 and timer1 are completely consistent. The main features for timer0 and timer1 are as follows:

- Alternative count clock: LOSC or OSC24M. The LOSC can be either the internal or external low-frequency clock, and the external one has more accuracy.
- Supports 8 prescale factors
- Programmable 32-bit down timer
- Supports two timing modes: periodic mode and single counting mode
- Generates an interrupt when the count is decreased to 0

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

- Single clock source: 32.768 kHz. It can be either the LOSC_32K or HOSC_32K (OSC24M/750).
- Supports 12 initial values
- Supports the generation of timeout interrupts
- Supports the generation of reset signals
- Supports Watchdog Restart

The AVS is used to synchronize the audio and video. The AVS module includes AVS0 and AVS1, which are completely consistent. The main features for the AVS are as follows:

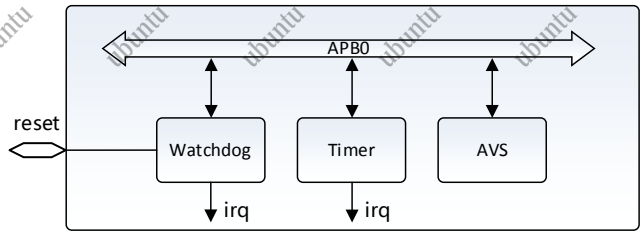
- Single clock source: OSC24M
- Programmable 33-bit up timer
- Supports updating the initial value anytime
- 12-bit frequency divider factor
- Supports Pause/Start function

3.6.2 Block Diagram

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 3-11 Timer Block Diagram



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

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

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

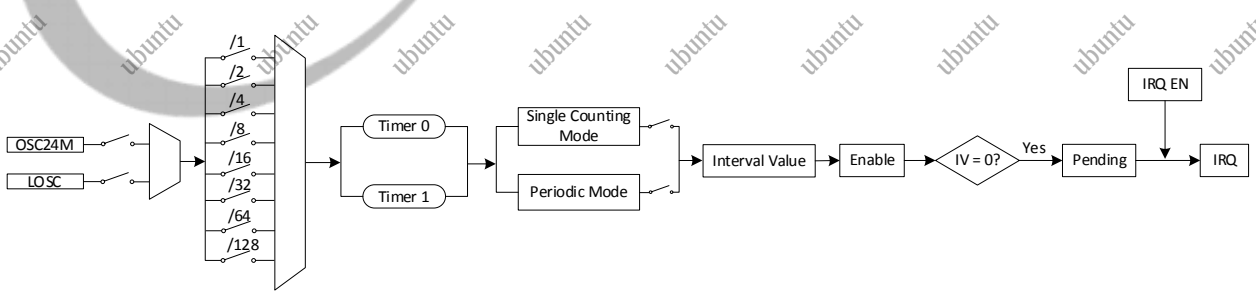
3.6.3 Functional Descriptions

3.6.3.1 Timer

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

The following figure shows the block diagram for the timer.

Figure 3-12 Block Diagram for the Timer



The clock source for the timer can be either OSC24M or LOSC. For LOSC, it can be either the internal or external low-frequency clock. The external one has more accuracy.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Timing Modes

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

- Single Counting Mode

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

- Periodic Mode

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

Formula for Calculating the Timer Time

The following formula decribes the relationship among timer parameters.

$$T_{timer} = \frac{TMRn_INTV_VALUE_REG - TMRn_CUR_VALUE_REG}{TMRn_CLK_SRC} \times TMRn_CLK_PRES$$

Where,

The parameter n is either 0 or 1;

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

TMRn_INTV_VALUE_REG is the interval value of the timer;

TMRn_CUR_VALUE_REG is the current value of the timer;

TMRn_CLK_SRC is the frequency of the timer clock source;

TMRn_CLK_PRES is the prescale ratio of the timer clock.

Initializing the Timer

Follow the steps below to initialize the timer:

1. Configure the timer parameters clock source, prescale factor, and timing mode by writing **TMRn_CTRL_REG.** There is no sequence requirement of configuring the parameters.
2. Write the interval value.
   a) Write **TMRn_INTV_VALUE_REG** to configure the interval value for the timer.
   b) Write bit[1] of **TMRn_CTRL_REG** to load the interval value to the timer. The value of the bit will be cleared automatically after loading the interval value.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Processing the Interrupt

Follow the steps below to process the interrupt:

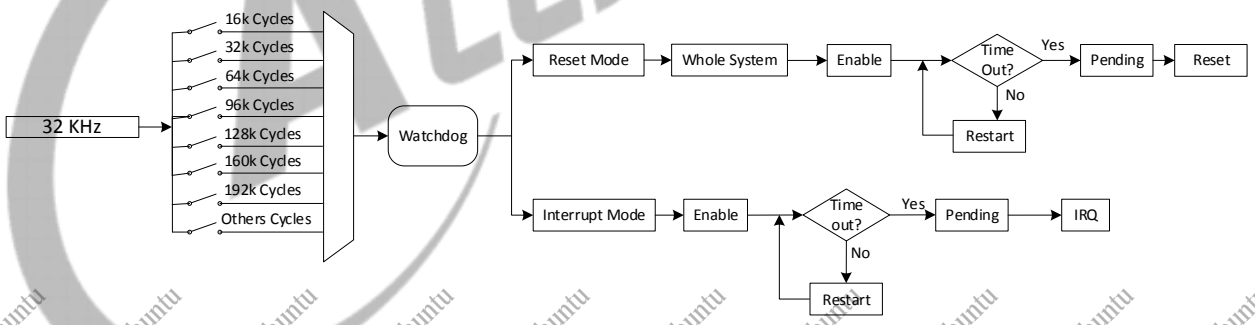
1. Enable interrupts for the timer: write the enable bit of the corresponding interrupt in TMR_IRQ_EN_REG for the timer. The timer will generate an interrupt everytime the count value reaches 0.
2. After entering the interrupt process, write the pending bit of the corresponding interrupt in TMR_IRQ_STA_REG to clear the interrupt pending, and execute the process of waiting for the interrupt.
3. Resume the interrupt and continue to execute the interrupted process.

3.6.3.2 Watchdog

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

The following figure shows the block diagram for the watchdog.

Figure 3-13 Block Diagram for the Watchdog



The clock source of the watchdog is 32 KHz. It can be either the LOSC_32K or HOSC_32K (OSC24M/750). There are 12 configurable initial count values.

Operating Modes

The watchdog has two operating modes: the interrupt mode and reset mode. In the interrupt mode, when the counter value reaches 0 and WDOG_IRQ_EN_REG is enabled, the watchdog generates an interrupt. In the reset mode, when the counter value reaches 0, the watchdog generates a reset signal to reset the entire system.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Initializing the Watchdog

Follow the steps below to initialize the watchdog:

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

Processing the Interrupt

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

Follow the steps below to process the interrupt:

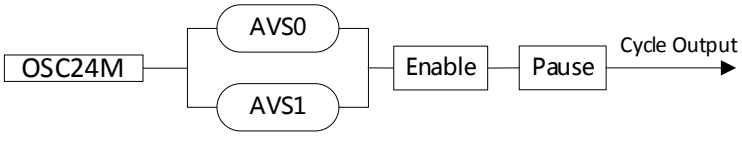
1. Write the enable bit of **WDOG_IRQ_EN_REG** to enable the interrupt.
2. After entering the interrupt process, write the pending bit of **WDOG_IRQ_STA_REG** to clear the interrupt pending and execute the process of waiting for the interrupt.
3. Resume the interrupt and continue to execute the interrupted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

3.6.3.3 AVS

The AVS is a 33-bit up counter. The counter value is increased by 1 on each rising edge of the count clock. There is a clock gate in CCU module to control the output of the AVS counter. To operate the AVS, open the clock gate first.

The following figure shows the block diagram for the AVS.

Figure 3-14 Block Diagram for the AVS



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

The clock source of the AVS is OSC24M. There is a 12-bit division factor (N0 or N1) for each AVS, N0 for AVS0 and N1 for AVS1. When the timer increases from 0 to N1 or N2, the AVS counter adds 1. When the counter reaches 33-bit upper limit, the AVS will start to count from the initial value again.

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

Starting or Pausing the AVS

Follow the steps below to

1. Write **AVS_CNT_DIV_REG** to configure the division factor.
2. Write **AVS_CNTn_REG** (n = 0 or 1) to configure the initial value.
3. Write **AVS_CNT_CTL_REG** to enable the AVS. You can pause the AVS at any time.

3.6.4 Programming Guidelines

3.6.4.1 Configuring the Timer

The following example shows how to make a one-millisecond delay with the clock source selected as OSC24M, the operating mode sets as single counting mode, and the pre-scale sets as 2.

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

3.6.4.2 Resetting the Watchdog

The following example shows how to make the watchdog to generate a reset signal to the whole system after 1 second. The clock source for the watchdog is OSC24M/750.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.6.4.3 Restarting the Watchdog

The following example shows how to restart the watchdog. In this example, the clock source is OSC24M/750, the interval value is 1 second, and the watchdog operating mode is the reset mode.

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

```
writel(0x1, WDOG_CONFIG);                          //To whole system
writel(0x10, WDOG_MODE);                           //Interval Value set 1s
writel(readl(WDOG_MODE)|(1<<0), WDOG_MODE);        //Enable Watchdog
----other codes---
writel(readl(WDOG_CTRL)|(0xA57<<1)|(1<<0), WDOG_CTRL); //Write 0xA57 at Key Field and Restart Watchdog
```

3.6.5 Register List

Module Name	Base Address
Timer	0x02000000

Register Name	Offset	Description
TMR_IRQ_EN_REG	0x0000	Timer IRQ Enable Register
TMR_IRQ_STA_REG	0x0004	Timer Status Register
TMR0_CTRL_REG	0x0010	Timer0 Control Register
TMR0_INTV_VALUE_REG	0x0014	Timer0 Interval Value Register
TMR0_CUR_VALUE_REG	0x0018	Timer0 Current Value Register
TMR1_CTRL_REG	0x0020	Timer1 Control Register
TMR1_INTV_VALUE_REG	0x0024	Timer1 Interval Value Register
TMR1_CUR_VALUE_REG	0x0028	Timer1 Current Value Register
WDOG_IRQ_EN_REG	0x00A0	Watchdog IRQ Enable Register
WDOG_IRQ_STA_REG	0x00A4	Watchdog Status Register
WDOG_SOFT_RST_REG	0x00A8	Watchdog Software Reset Register
WDOG_CTRL_REG	0x00B0	Watchdog Control Register
WDOG_CFG_REG	0x00B4	Watchdog Configuration Register
WDOG_MODE_REG	0x00B8	Watchdog Mode Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
WDOG_OUTPUT_CFG_REG	0x00BC	Watchdog Output Configuration Register
AVS_CNT_CTL_REG	0x00C0	AVS Control Register
AVS_CNT0_REG	0x00C4	AVS Counter 0 Register
AVS_CNT1_REG	0x00C8	AVS Counter 1 Register
AVS_CNT_DIV_REG	0x00CC	AVS Divisor Register

3.6.6 Register Description

3.6.6.1 0x0000 Timer IRQ Enable Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: TMR_IRQ_EN_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1	R/W	0x0	TMR1_IRQ_EN Timer1 Interrupt Enable 0: Disabled 1: Enabled
0	R/W	0x0	TMR0_IRQ_EN Timer0 Interrupt Enable 0: Disabled 1: Enabled

3.6.6.2 0x0004 Timer IRQ Status Register (Default Value: 0x0000_0000)

Offset: 0x0004			Register Name: TMR_IRQ_STA_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1	R/W1C	0x0	TMR1_IRQ_PEND The IRQ pending bit for Timer1. 0: No effect 1: Pending, indicates that the interval value of the timer1 is reached. Write 1 to clear the pending status.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0004			Register Name: TMR_IRQ_STA_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
0	R/W1C	0x0	TMR0_IRQ_PEND The IRQ pending bit for Timer0. 0: No effect 1: Pending, indicates that the interval value of the timer0 is reached. Write 1 to clear the pending status.

3.6.6.3 0x0010 Timer0 Control Register (Default Value: 0x0000_0004)

Offset: 0x0010			Register Name: TMR0_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R/W	0x0	TMR0_MODE Select the timing mode for timer0. 0: Periodic mode. When the interval value of the timer0 is reached, the timer will restart another round of counting automatically. 1: Single counting mode. When the interval value of the timer0 is reached, the timer will stop counting.
6:4	R/W	0x0	TMR0_CLK_PRES Select the pre-scale of Timer0 clock source. 000: /1 001: /2 010: /4 011: /8 100: /16 101: /32 110: /64 111: /128
3:2	R/W	0x1	TMR0_CLK_SRC Select the clock source for Timer0. 00: LOSC 01: OSC24M 10: / 11: /

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0010			Register Name: TMR0_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
1	R/W	0x0	TMR0_RELOAD Timer0 Reload 0: No effect 1: Reload the Interval value for Timer0 After the bit is set, it can not be written again before it is cleared automatically.
0	R/W	0x0	TMR0_EN Timer0 Enable 0: Stop/Pause 1: Start By setting the bit to 1, the timer will be started. It reloads the interval value register and then counts from the interval value to 0. By setting the bit to 0 before the timer counts to 0, the timer will be paused. The bit will be locked to 0 for at least 2 cycles. Within the 2 cycles, you cannot set the bit to 1 to restart the timer. The timer supports updating the interval value in the pause state. To start to down-count from the updated interval value, set both the reload bit and enable bit to 1. Additionally, in the single counting mode, after the count value reaches 0, the system will automatically change the bit to 0 to stop the timer.

3.6.6.4 0x0014 Timer0 Interval Value Register (Default Value: 0x0000_0000)

Offset: 0x0014			Register Name: TMR0_INTV_VALUE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	TMR0_INTV_VALUE Timer0 Interval Value

NOTE

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.6.6.5 0x0018 Timer0 Current Value Register (Default Value: 0x0000_0000)

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

3.6.6.6 0x0020 Timer1 Control Register (Default Value: 0x0000_0004)

Offset: 0x0020			Register Name: TMR1_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R/W	0x0	TMR1_MODE Select the timing mode for timer1. 0: Periodic mode. When the interval value of the timer1 is reached, the timer will restart another round of counting automatically. 1: Single counting mode. When the interval value of the timer1 is reached, the timer will stop counting.
6:4	R/W	0x0	TMR1_CLK_PRES Select the pre-scale of timer1 clock source 000: /1 001: /2 010: /4 011: /8 100: /16 101: /32 110: /64 111: /128
3:2	R/W	0x1	TMR1_CLK_SRC Select the pre-scale of timer1 clock source. 00: LOSC 01: OSC24M 10: / 11: /

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0020			Register Name: TMR1_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
1	R/W	0x0	TMR1_RELOAD Timer1 Reload 0: No effect 1: Reload the Interval value for Timer1. After the bit is set, it can not be written again before it is cleared automatically.
0	R/W	0x0	TMR1_EN Timer1 Enable 0: Stop/Pause 1: Start By setting the bit to 1, the timer will be started. It reloads the interval value register and then counts from the interval value to 0. By setting the bit to 0 before the timer counts to 0, the timer will be paused. The bit will be locked to 0 for at least 2 cycles. Within the 2 cycles, you cannot set the bit to 1 to restart the timer. The timer supports updating the interval value in the pause state. To start from the updated interval value, set both the reload bit and enable bit to 1. Additionally, in the single counting mode, after the count value reaches 0, the system will automatically change the bit to 0 to stop the timer.

3.6.6.7 0x0024 Timer1 Interval Value Register (Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: TMR1_INTV_VALUE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	TMR1_INTV_VALUE Timer1 Interval Value

NOTE

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.6.6.8 0x0028 Timer1 Current Value Register (Default Value: 0x0000_0000)

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

3.6.6.9 0x00A0 Watchdog IRQ Enable Register (Default Value: 0x0000_0000)

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

3.6.6.10 0x00A4 Watchdog Status Register (Default Value: 0x0000_0000)

Offset: 0x00A4			Register Name: WDOG_IRQ_STA_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W1C	0x0	WDOG_IRQ_PEND The IRQ pending bit for the watchdog. Write 1 to clear the pending status. 0: No effect 1: Pending, indicates that the interval value of the watchdog is reached.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.6.6.11 0x00A8 Watchdog Software Reset Register (Default Value: 0x0000_0000)

Offset: 0x00A8			Register Name: WDOG_SOFT_RST_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	W	0x0	KEY_FIELD Key Field. To change the value of bit[0], this field should be filled with 0x16AA.
15:1	/	/	/
0	R/W1C	0x0	Soft Reset Enable 0: De-assert 1: Reset the system **Note: To use the bit to reset the system, disable the watchdog first.**

3.6.6.12 0x00B0 Watchdog Control Register (Default Value: 0x0000_0000)

Offset: 0x00B0			Register Name: WDOG_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:13	/	/	/
12:1	W	0x0	WDOG_KEY_FIELD Watchdog Key Field It should be written to 0xA57. Writing any other values in this field aborts the write operation.
0	R/W1S	0x0	WDOG_RESTART Watchdog Restart 0: No effect 1: Restart the watchdog

3.6.6.13 0x00B4 Watchdog Configuration Register (Default Value: 0x0000_0001)

Offset: 0x00B4			Register Name: WDOG_CFG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:9	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x00B4			Register Name: WDOG_CFG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
8	R/W	0x0	WDOG_CLK_SRC Select the clock source for the watchdog. 0: HOSC_32K, that is, OSC24M/750. It is a 32 KHz clock divided from the OSC24M. 1: LOSC_32K. A clock provided by the LOSC. The exact frequency is 32.768 kHz.
7:2	/	/	/
1:0	R/W	0x1	WDOG_MODE Configure the operating mode for the watchdog 00: / 01: Reset mode 10: Interrupt mode 11: /

3.6.6.14 0x00B8 Watchdog Mode Register (Default Value: 0x0000_0000)

Offset: 0x00B8			Register Name: WDOG_MODE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x00B8			Register Name: WDOG_MODE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
7:4	R/W	0x0	WDOG_INTV_VALUE Watchdog Interval Value 0000: 0.5 second 0001: 1 second 0010: 2 seconds 0011: 3 seconds 0100: 4 seconds 0101: 5 seconds 0110: 6 seconds 0111: 8 seconds 1000: 10 seconds 1001: 12 seconds 1010: 14 seconds 1011: 16 seconds Others: Reserved **Note: The corresponding clock cycles for the interval value (IV) depends on the frequency of the clock: Cycles = F_{CLK} * IV.** For example, to get a interval value of 0.5 second, if the clock source is HOSC_32K (whose frequency is 32 KHz), the cycle number is 16,000; if the clock source is LOSC_32K (whose frequency is 32.768 kHz), the cycle number is 16,384.
3:1	/	/	/
0	R/W	0x0	WDOG_EN Watchdog Enable 0: No effect 1: Enable the Watchdog

3.6.6.15 0x00BC Watchdog Output Config Register (Default Value: 0x0000_001F)

Offset: 0x00BC			Register Name: WDOG_OUTPUT_CFG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:12	/	/	/
11:0	R/W	0x1F	WDOG OUTPUT CONFIG Configure the valid time for the watchdog reset signal. T = 1/32ms*(N + 1) The default value is 1 ms.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.6.6.16 0x00C0 AVS Counter Control Register (Default Value: 0x0000_0000)

Offset: 0x00C0			Register Name: AVS_CNT_CTL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:10	/	/	/
9	R/W	0x0	AVS_CNT1_PS Audio/Video Sync Counter 1 Pause Control 0: Do not pause. 1: Pause the AVS counter1.
8	R/W	0x0	AVS_CNT0_PS Audio/Video Sync Counter 0 Pause Control 0: Do not pause. 1: Pause the AVS counter0.
7:2	/	/	/
1	R/W	0x0	AVS_CNT1_EN Audio/Video Sync Counter 1 Enable/Disable The clock source is OSC24M. 0: Disabled 1: Enabled
0	R/W	0x0	AVS_CNT0_EN Audio/Video Sync Counter 0 Enable/Disable The clock source is OSC24M. 0: Disabled 1: Enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.6.6.17 0x00C4 AVS Counter 0 Register (Default Value: 0x0000_0000)

Offset: 0x00C4			Register Name: AVS_CNT0_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	AVS_CNT0 The higher 32 bits of AVS counter0. AVS counter0 is a 33-bit up counter. The initial value consists of two parts: this register forms the bit[32:1] of AVS counter0 and the bit[0] is zero. You can set the initial value of the AVS counter0 by software. The initial value can be updated at anytime. You can also pasuse the counter by setting AVS_CNT0_PS to ”1”. The counter value will not increase when it is paused.

3.6.6.18 0x00C8 AVS Counter 1 Register (Default Value: 0x0000_0000)

Offset: 0x00C8			Register Name: AVS_CNT1_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	AVS_CNT1 The higher 32 bits of AVS counter1. AVS counter1 is a 33-bit up counter. The initial value consists of two parts: this register forms the bit[32:1] of AVS counter0 and the bit[0] is zero. You can set the initial value of the AVS counter1 by software. The initial value can be updated at anytime. You can also pasuse the counter by setting AVS_CNT1_PS to ”1”. The counter value will not increase when it is paused.

3.6.6.19 0x00CC AVS Counter Divisor Register (Default Value: 0x05DB_05DB)

Offset: 0x00CC			Register Name: AVS_CNT_DIV_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x00CC			Register Name: AVS_CNT_DIV_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
27:16	R/W	0x5DB	AVS_CNT1_D N1, the divisor factor for AVS1. The clock for AVS1 is 24 MHz/N1. N1 = Bit[27:16] + 1. The valid value for N1 is from 1 to 0x7ff. There is an internal 12-bit counter maintained by the engine of the 33-bit AVS1. The 12-bit counter is used for counting the cycle number of the clock OSC24M. When the value of the 12-bit counter reaches N1, the internal 33-bit counter register will increase 1 and the 12-bit counter will reset to zero and restart again. You can change the value of N1 via the software at any time.
15:12	/	/	/
11:0	R/W	0x5DB	AVS_CNT0_D N0, the divisor factor for AVS0. The clock for AVS0 is 24MHz/N0. N0 = Bit[27:16] + 1. The valid value for N0 is from 1 to 0x7ff. There is an internal 12-bit counter maintained by the engine of the 33-bit AVS0. The 12-bit counter is used for counting the cycle number of the clock OSC24M. When the value of the 12-bit counter reaches N0, the internal 33-bit counter register will increase 1 and the 12-bit counter will reset to zero and restart again. You can change the value of N0 via the software at any time.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.7 High Speed Timer

3.7.1 Overview

The high speed timer (HSTimer) module consists of HSTimer0 and HSTimer1. HSTimer0 and HSTimer1 are down counters that implement timing and counting functions. They are completely consistent. Compared with the timer module, the HSTimer module provides more precise timing and counting.

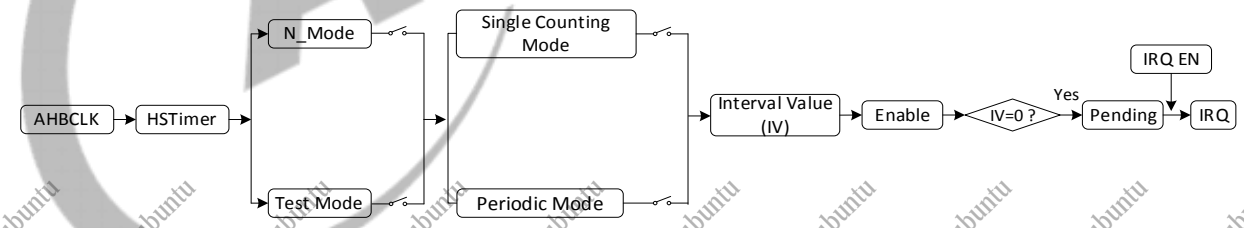
The HSTimer has the following features:

- Single clock source: AHB1
- Supports 5 prescale factors
- Configurable 56-bit down timer
- Supports 2 timing modes: periodic mode and single counting mode
- Supports the test mode
- Generates an interrupt when the count is decreased to 0

3.7.2 Block Diagram

The following figure shows the block diagram of the HSTimer.

Figure 3-15 HSTimer Block Diagram



3.7.3 Functional Description

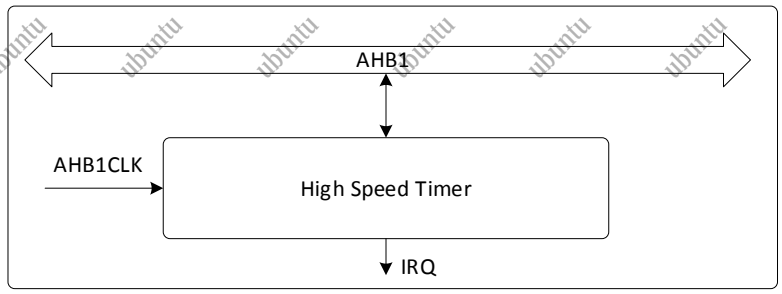
The HSTimers are 56-bit down counters. The counter value is decremented by 1 on each rising edge of the count clock. Each HSTimer has a prescaler that divides the working clock frequency of each working timer by 1, 2, 4, 8, or 16.

3.7.3.1 Typical Application

The following figure shows a typical application of HSTimer module.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 3-16 Typical Application for HSTimer



The HSTimer module is mounted at AHB1, and can control the registers via AHB1. AHB1 is the single clock source of the HSTimer. When the count value reaches zero, the HSTimer generates an interrupt.

3.7.3.2 Timing Modes

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

- Single Counting Mode

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

- Periodic Mode

  In the periodic mode, the HSTimer restarts another round of counting after generating the interrupt. It reloads data from **HS_TMRn_INTV_LO_REG** and **HS_TMRn_INTV_HI_REG** and then continues to count.

3.7.3.3 Operating Modes

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

- Normal Mode

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

- Test Mode

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

3.7.3.4 HSTimer Formula

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

$$T_{HSTimer} = \frac{(HS_TMRn_INTV_HI_REG << 32 + HS_TMRn_INTV_LO_REG) - (HS_TMRn_CURNT_HI_REG << 32 + HS_TMRn_CURNT_LO_REG)}{AHB1CLK} \times HS_TMRn_CLK$$

Where,

The parameter n is either 0 or 1;

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

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

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

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

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

AHB1CLK is the frequency of AHB1 Clock (the HSTimer clock source);

HS_TMRn_CLK is the prescale ratio of the HSTimer clock.

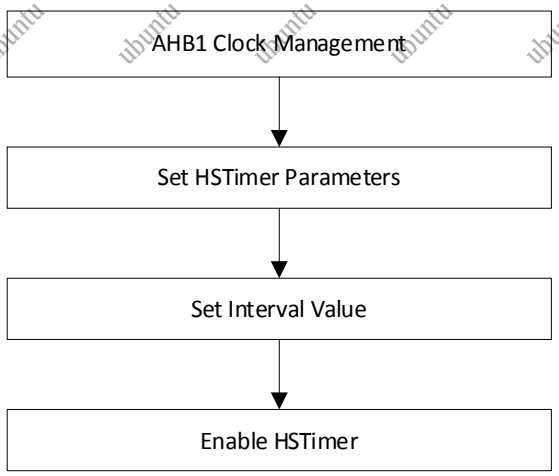
3.7.4 Programming Guidelines

3.7.4.1 Initializing the HSTimer

The following figure shows the process of HSTimer initialization.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 3-17 HSTimer Initialization Process



1. AHB1 clock management: Open the clock gate of AHB1 and de-assert the soft reset of AHB1 in CCU.
2. Configure the corresponding parameters of the HSTimer: clock source, prescaler factor, operating mode, and timing mode. There is no sequence requirement when writing the above parameters to **HS_TMRn_CTRL_REG**.
3. Write the interval value: Write the lower 32 bits of the interval value to **HS_TMRn_INTV_LO_REG** first, and then the higher 24 bits to **HS_TMRn_INTV_HI_REG**. Normally, write the bit[1] of **HS_TMRn_CTRL_REG** to load the interval value. If the HSTimer is in the timing stop stage, write the bit[1:0] of **HS_TMRn_CTRL_REG** to reload the interval value.
4. Enable HSTimer: Write the bit[0] of **HS_TMRn_CTRL_REG** to enable HSTimer to count.
5. Reading **HS_TMRn_CURNT_LO_REG** and **HS_TMRn_CURNT_HI_REG** can get current counting value.

3.7.4.2 Processing the HSTimer Interrupt

Follow the steps below to process the HSTimer interrupt:

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.7.4.3 Making 1 us Delay

The following example shows how to make a 1 us delay with HSTimer0. The frequency of the clock AHB1CLK is 100 MHz, the operating mode is the normal mode, the timing mode is single counting mode, and the pre-scale is 2.

```
writel(0x32, HS_TMR0_INTV_LO);                    //Set bit[31:0] of the interval value as 0x32.
writel(0x0, HS_TMR0_INTV_HI);                     //Set bit[55:32] of the interval value as 0x0.
writel(0x90, HS_TMR0_CTRL);
//Set the operating mode as Normal Mode, the pre-scale as 2, and the timing mode as Single Counting Mode.
writel(readl(HS_TMR0_CTRL)|(1<<1), HS_TMR0_CTRL);  //Set the reload bit.
writel(readl(HS_TMR0_CTRL)|(1<<0), HS_TMR0_CTRL);  //Enable HSTimer0.
while(!(readl(HS_TMR_IRQ_STAS)&1));               //Wait for HSTimer0 to generate pending.
writel(1,HS_TMR_IRQ_STAS);                        //Clear HSTimer0 pending.
```

3.7.5 Register List

Module Name	Base Address
HSTimer	0x03005000

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.7.6 Register Description

3.7.6.1 0x0000 HS Timer IRQ Enable Register (Default Value: 0x0000_0000)

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

3.7.6.2 0x0004 HS Timer IRQ Status Register (Default Value: 0x0000_0000)

Offset: 0x0004			Register Name: HS_TMR_IRQ_STAS_REG
Bit	Read/Write	Default/Hex	Description
31:2	/	/	/
1	R/W1C	0x0	HS_TMR1_IRQ_PEND HSTimer1 IRQ Pending The IRQ pending bit for HSTimer1. Write 1 to clear the pending status. 0: No effect 1: Pending, indicates that the count value of the timer has been decreased to 0.
0	R/W1C	0x0	HS_TMR0_IRQ_PEND HSTimer0 IRQ Pending The IRQ pending bit for HSTimer0. Write 1 to clear the pending status. 0: No effect 1: Pending, indicates that the count value of the timer has been decreased to 0.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.7.6.3 0x0020 HS Timer0 Control Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: HS_TMR0_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	HS_TMR0_TEST Select the operating mode for HSTimer0. 0: Normal mode 1: Test mode In the test mode, the HS_TMR0_INTV_LO_REG must be set to 0x1, and HS_TMR0_INTV_HI_REG acts as the interval value for HSTimer0.
30:8	/	/	/
7	R/W	0x0	HS_TMR0_MODE Select the timing mode for HSTimer0. 0: Periodic mode. When the count value is decreased to 0, the timer will restart another round of counting automatically. 1: Single counting mode. When the count value is decreased to 0, the timer will stop counting.
6:4	R/W	0x0	HS_TMR0_CLK Select the pre-scale for the HSTimer0 clock sources. 000: /1 001: /2 010: /4 011: /8 100: /16 101: / 110: / 111: /
3:2	/	/	/
1	R/W1S	0x0	HS_TMR0_RELOAD HSTimer0 Reload 0: No effect 1: Reload the HSTimer0 interval value.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0020			Register Name: HS_TMR0_CTRL_REG
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	HS_TMR0_EN HSTimer0 Enable 0: Stop/Pause 1: Start By setting the bit to 1, the timer will be started. It reloads the interval value register and then counts from the interval value to 0. By setting the bit to 0 before the timer counts to 0, the timer will be paused. The bit will be locked to 0 for at least 2 cycles. Within the 2 cycles, you cannot set the bit to 1 to restart the timer. The timer supports updating the interval value in the pause state. To start from the updated interval value, set both the reload bit and enable bit to 1. Additionally, in the single counting mode, after the count value reaches 0, the system will automatically change the bit to 0 to stop the timer.

3.7.6.4 0x0024 HS Timer0 Interval Value Lo Register (Default Value: 0x0000_0000)

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

3.7.6.5 0x0028 HS Timer0 Interval Value Hi Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

NOTE

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

3.7.6.6 0x002C HS Timer0 Current Value Lo Register (Default Value: 0x0000_0000)

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

3.7.6.7 0x0030 HS Timer0 Current Value Hi Register (Default Value: 0x0000_0000)

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

NOTE

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.7.6.8 0x0040 HS Timer1 Control Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

3.7.6.9 0x0044 HS Timer1 Interval Value Lo Register (Default Value: 0x0000_0000)

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

3.7.6.10 0x0048 HS Timer1 Interval Value Hi Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

NOTE

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

3.7.6.11 0x004C HS Timer1 Current Value Lo Register (Default Value: 0x0000_0000)

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

3.7.6.12 0x0050 HS Timer1 Current Value Hi Register (Default Value: 0x0000_0000)

Offset: 0x0050			Register Name: HS_TMR1_CURNT_HI_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	/	/	/
23:0	R/W	0x0	HS_TMR1_CUR_VALUE_HI Bit[55:32] of the HSTimer0 current value.

NOTE

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.8 Generic Interrupt Controller (GIC)

The following table describes the details of interrupt sources.

Table 3-10 Interrupt Sources

Interrupt Number	Interrupt Source	Interrupt Vector	Description
0	SGI 0	0x0000	SGI 0 interrupt
1	SGI 1	0x0004	SGI 1 interrupt
2	SGI 2	0x0008	SGI 2 interrupt
3	SGI 3	0x000C	SGI 3 interrupt
4	SGI 4	0x0010	SGI 4 interrupt
5	SGI 5	0x0014	SGI 5 interrupt
6	SGI 6	0x0018	SGI 6 interrupt
7	SGI 7	0x001C	SGI 7 interrupt
8	SGI 8	0x0020	SGI 8 interrupt
9	SGI 9	0x0024	SGI 9 interrupt
10	SGI 10	0x0028	SGI 10 interrupt
11	SGI 11	0x002C	SGI 11 interrupt
12	SGI 12	0x0030	SGI 12 interrupt
13	SGI 13	0x0034	SGI 13 interrupt
14	SGI 14	0x0038	SGI 14 interrupt
15	SGI 15	0x003C	SGI 15 interrupt
16	PPI 0	0x0040	PPI 0 interrupt
17	PPI 1	0x0044	PPI 1 interrupt
18	PPI 2	0x0048	PPI 2 interrupt
19	PPI 3	0x004C	PPI 3 interrupt
20	PPI 4	0x0050	PPI 4 interrupt
21	PPI 5	0x0054	PPI 5 interrupt
22	PPI 6	0x0058	PPI 6 interrupt
23	PPI 7	0x005C	PPI 7 interrupt
24	PPI 8	0x0060	PPI 8 interrupt
25	PPI 9	0x0064	PPI 9 interrupt
26	PPI 10	0x0068	PPI 10 interrupt
27	PPI 11	0x006C	PPI 11 interrupt
28	PPI 12	0x0070	PPI 12 interrupt

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Interrupt Number	Interrupt Source	Interrupt Vector	Description
29	PPI 13	0x0074	PPI 13 interrupt
30	PPI 14	0x0078	PPI 14 interrupt
31	PPI 15	0x007C	PPI 15 interrupt
32	MSGBOX0	0x0080	
33	MSGBOX1	0x0084	
34	UART0	0x0088	
35	UART1	0x008C	
36	UART2	0x0090	
37	UART3	0x0094	
38		0x0098	
39		0x009C	
40		0x00A0	
41	TWI0	0x00A4	
42	TWI1	0x00A8	
43		0x00AC	
44		0x00B0	
45		0x00B4	
46	SCR	0x00B8	
47	SPI0	0x00BC	
48	SPI1	0x00C0	
49	LEDC	0x00C4	
50	PWM	0x00C8	
51	IR_TX	0x00CC	
52	IR_RX	0x00D0	
53		0x00D4	
54		0x00D8	
55	OWA	0x00DC	
56		0x00E0	
57		0x00E4	
58	I2S/PCM0	0x00E8	
59	I2S/PCM1	0x00EC	
60		0x00F0	
61	USB0_DEVICE	0x00F4	
62	USB0_EHCI	0x00F8	

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Interrupt Number	Interrupt Source	Interrupt Vector	Description
63	USB0_OHCI	0x00FC	
64		0x0100	
65		0x0104	
66	USB1_OHCI	0x0108	
67		0x010C	
68		0x0110	
69		0x0114	
70	NAND	0x0118	
71		0x011C	
72	SMHC0	0x0120	
73	SMHC1	0x0124	
74		0x0128	
75	MSI	0x012C	
76	SMC	0x0130	
77		0x0134	
78	EMAC	0x0138	
79		0x013C	
80	CCU_FERR	0x0140	
81	HRDY_TOUT	0x0144	
82	DMAC_NS	0x0148	
83	DMAC_S	0x014C	
84	CE_NS	0x0150	
85	CE_S	0x0154	
86	SPINLOCK	0x0158	
87	HSTIMER0	0x015C	
88	HSTIMER1	0x0160	
89		0x0164	
90		0x0168	
91	TIMER0	0x016C	
92	TIMER1	0x0170	
93		0x0174	
94		0x0178	
95	WATCHDOG	0x017C	
96		0x0180	

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Interrupt Number	Interrupt Source	Interrupt Vector	Description
97	AIPU	0x0184	
98		0x0188	
99		0x018C	
100		0x0190	
101	GPIOB	0x0194	
102	GPIOC	0x0198	
103	GPIOF	0x019C	
104	GPIOG	0x01A0	
105	GPIOH	0x01A4	
106		0x01A8	
107		0x01AC	
108		0x01B0	
109		0x01B4	
110		0x01B8	
111		0x01BC	
112		0x01C0	
113		0x01C4	
114		0x01C8	
115		0x01CC	
116		0x01D0	
117		0x01D4	
118		0x01D8	
119		0x01DC	
CPUS Domain Related			
120	NMI	0x01E0	
121	R_MSGBOX	0x01E4	
122	R_DMAC_S	0x01E8	
123	R_DMAC_NS	0x01EC	
124		0x01F0	
125	R_I2S/PCM0	0x01F4	
126		0x01F8	
127	DMIC	0x01FC	
128	AUDIO_CODEC	0x0200	
129		0x0204	

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Interrupt Number	Interrupt Source	Interrupt Vector	Description
130	R_PPU	0x0208	
131	R_SPINLOCK	0x020C	
132	R_TWD	0x0210	
133	R_WATCHDOG	0x0214	
134	R_TIMER0	0x0218	
135	R_TIMER1	0x021C	
136	R_TIMER2	0x0220	
137	R_TIMER3	0x0224	
138	R_ALARM	0x0228	
139	R_GPIOL	0x022C	
140	R_GPIOM	0x0230	
141	R_GPION	0x0234	
142		0x0238	
143	R_UART0	0x023C	
144	R_TWI0	0x0240	
145	R_IRRX	0x0244	
146	R_PWM	0x0248	
147	LRADC	0x024C	
148	GPADC	0x0250	
149	THS	0x0254	
150	AHBS_HREADY_TOUT	0x0258	
151	R_DSP0_DE_ERR	0x025C	
152	R_DSP0_PF_ERR	0x0260	
153	R_DSP1_DE_ERR	0x0264	
154	R_DSP1_PF_ERR	0x0268	
CPUX Related			
160	C0_CTI0	0x0280	
161	C0_CTI1	0x0284	
162		0x0288	
163		0x028C	
164	C0_COMMTX0	0x0290	
165	C0_COMMTX1	0x0294	
166		0x0298	
167		0x029C	

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Interrupt Number	Interrupt Source	Interrupt Vector	Description
168	C0_COMMRX0	0x02A0	
169	C0_COMMRX1	0x02A4	
170		0x02A8	
171		0x02AC	
172	C0_PMU0	0x02B0	
173	C0_PMU1	0x02B4	
174		0x02B8	
175		0x02BC	
176	C0_AXI_ERROR	0x02C0	
177	AXI_WR_IRQ	0x02C4	
178	AXI_RD_IRQ	0x02C8	
179	DBGRSTREQ0	0x02CC	
180	DBGRSTREQ1	0x02D0	
181		0x02D4	
182		0x02D8	
183	nVCPUMNTIRQ0	0x02DC	
184	nVCPUMNTIRQ1	0x02E0	
185		0x02E4	
186		0x02E8	
187	nCOMMIRQ0	0x02EC	
188	nCOMMIRQ1	0x02F0	
189		0x02F4	
190		0x02F8	
191	DBGPWRUPREQ_out	0x02FC	

For complete GIC information, refer to the ***GIC PL400 technical reference manual*** and ***ARM GIC Architecture Specification V2.0***.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.9 Direct Memory Access Controller (DMAC)

3.9.1 Overview

The direct memory access (DMA) is a method of transferring data between peripherals and memories (including the SRAM and DRAM) without using the CPU. It is an efficient way to offload data transfer duties from the CPU. Without DMA, the CPU has to control all the data transfers. While with DMA, the DMAC directly transfers data between a peripheral and a memory, between peripherals, or between memories.

The DMAC has the following features:

- Provides 8 DMA channels in CPUX domain and 8 DMA channels in CPUS domain
- Provides 32 peripheral DMA requests for data reading and 32 peripheral DMA requests for data writing
- Supports transferring data with a linked list
- Supports programmable 8-bit, 16-bit, 32-bit, and 64-bit data width
- Supports programmable DMA burst length
- DRQ response includes the waiting mode and handshake mode
- Memory devices support non-aligned transform
- DMA channel supports pause function

NOTE

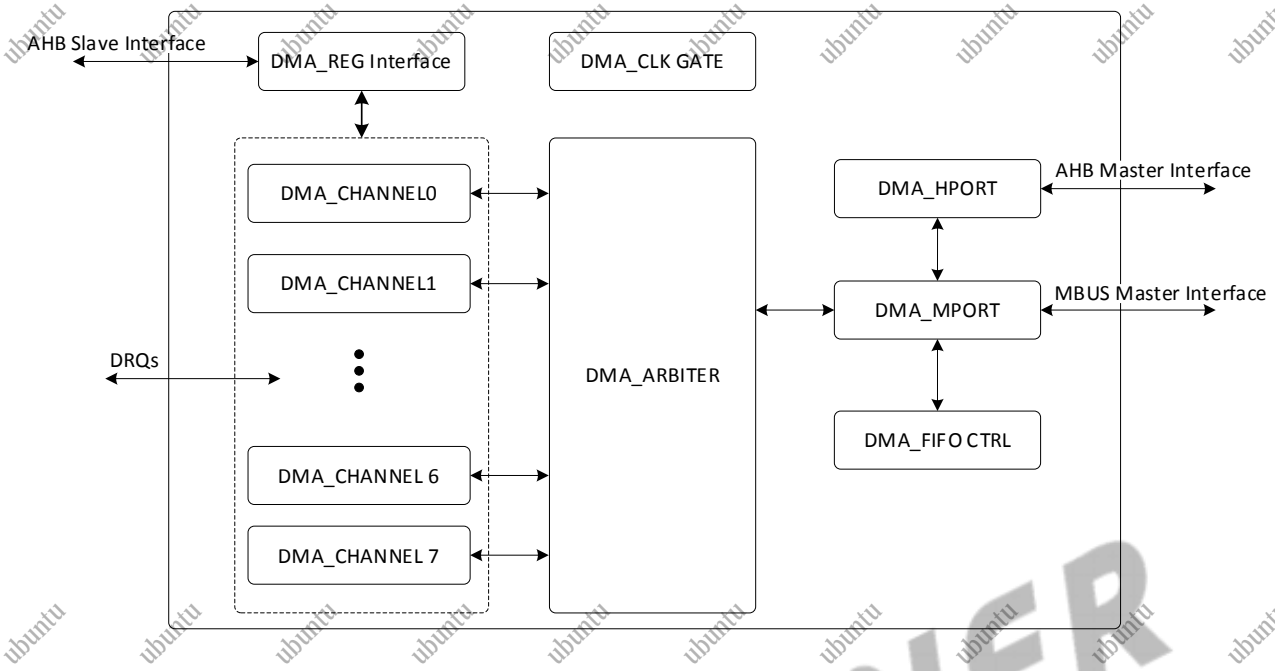
The following description focuses on the DMA of the CPUX domain.

3.9.2 Block Diagram

The following figure shows a block diagram of DMAC.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 3-18 DMAC Block Diagram



DMAC contains the following sub-blocks:

Table 3-11 DMAC Sub-blocks

Sub-block	Description
DMA_ARBITER	Arbitrates the DMA read/write requests from all channels, and converts the requests to the read/write requests of ports.
DMA_CHANNELs	DMA transfer engine. Each channel is independent. When the DMA requests from multiple peripherals are valid simultaneously, the channel with the highest priority starts data transfer first. The system uses the polling mechanism to decide the priorities of DMA channels. When DMA_ARBITER is idle, channel 0 has the highest priority, whereas channel 7 has the lowest priority. When DMA_ARBITER is busy processing the request from channel n, channel (n+1) has the highest priority. For n = 7, the channel (n + 1) should be channel 0.
DRQs	DMA requests. Peripherals use the DMA request signals to request a data transfer.
DMA_MPORT	Receives the read/write requests from DMA_ARBITER, and converts the requests to the corresponding MBUS access requests. It is mainly used for accessing the DRAM.
DMA_HPORT	The port for accessing the AHB Master. It is mainly used for accessing the SRAM and IO devices.
DMA_FIFO CTRL	Internal FIFO cell control module.
DMA_REG Interface	DMA_REG is the common register module that is mainly used to resolve AHB1 demands.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Sub-block	Description
DMA_CLKGATE	The control module for hardware auto clock gating.

The DMAC integrates 8 independent DMA channels and each channel has an independent FIFO controller. When the DMA channel starts, the DMAC gets a DMA descriptor from the DMA_DESC_ADDR_REG and uses it as the configuration information for the data transfer of the current DMA package. Then the DMAC can transfer data between the specified devices. After transferring a DMA package, the DMAC judges if the current channel transfer is finished via the linked address in the descriptor. If the linked address shows all the packages are transferred, the DMAC will end the chain transmission and close the channel.

3.9.3 Functional Description

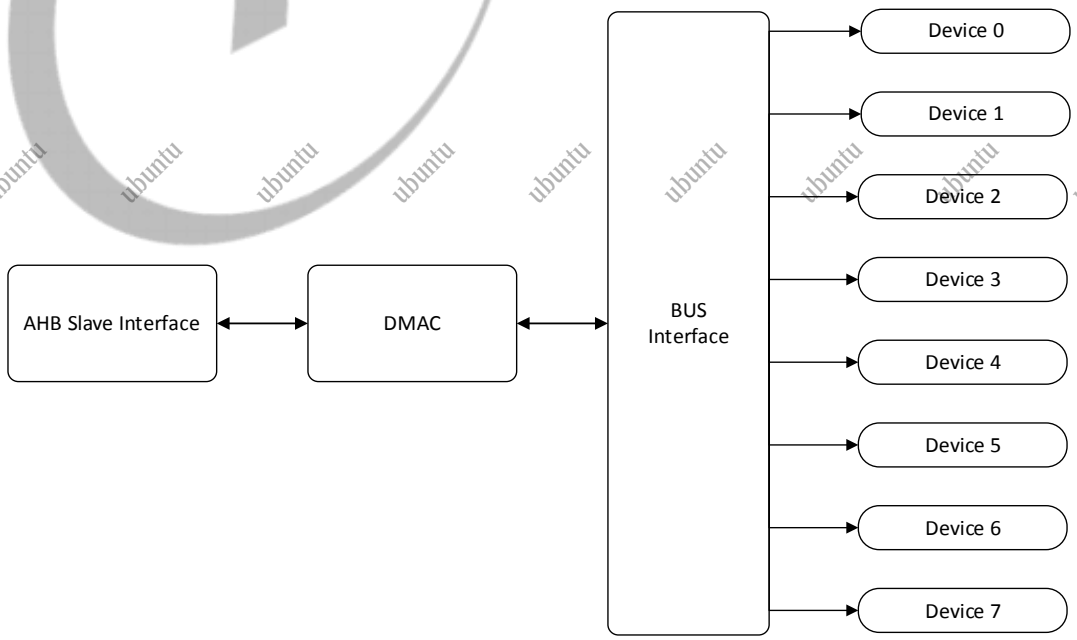
3.9.3.1 Clock

The DMAC is on AHB1. The clock of AHB1 influences the transfer efficiency of the DMAC.

3.9.3.2 Typical Application

The following figure shows a typical application of the DMAC.

Figure 3-19 DMAC Typical Application Diagram



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.9.3.3 DRQ Port of Peripherals

The following table shows the source DRQ types and destination DRQ types of different ports.

Table 3-12 DMA DRQ Type

Source DRQ Type		Destination DRQ Type	
port0	SRAM	port0	SRAM
port1	DRAM	port1	DRAM
port2	OWA-RX	port2	OWA-TX
port3	I2S/PCM0-RX	port3	I2S/PCM0-TX
port4	I2S/PCM1-RX	port4	I2S/PCM1-TX
port5		port5	
port6	Audio Codec	port6	Audio Codec
port7	DMIC	port7	
port8		port8	
port9	NAND	port9	NAND
port10		port10	
port11		port11	
port12	GPADC	port12	
port13		port13	IR-TX
port14	UART0-RX	port14	UART0-TX
port15	UART1-RX	port15	UART1-TX
port16	UART2-RX	port16	UART2-TX
port17	UART3-RX	port17	UART3-TX
port18		port18	
port19		port19	
port20		port20	
port21		port21	
port22	SPI0-RX	port22	SPI0-TX
port23	SPI1-RX	port23	SPI1-TX
port24		port24	
port25		port25	
port26		port26	
port27		port27	
port28		port28	

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Source DRQ Type		Destination DRQ Type	
port29		port29	
Port30	USB0_EP1	Port30	USB0_EP1
Port31	USB0_EP2	Port31	USB0_EP2
Port32	USB0_EP3	Port32	USB0_EP3
Port33	USB0_EP4	Port33	USB0_EP4
Port34	USB0_EP5	Port34	USB0_EP5
Port35		Port35	
Port36		Port36	
Port37		Port37	
Port38		Port38	
Port39		Port39	
Port40		Port40	
Port41		Port41	
Port42		Port42	LEDC
Port43	TWI0	Port43	TWI0
Port44	TWI1	Port44	TWI1
Port45		Port45	
Port46		Port46	
Port47		Port47	
Port48		Port48	
Port49	R_TWI0	Port49	R_TWI0
Port50	R_UART0	Port50	R_UART0
Port51	R_I2S/PCM0	Port51	R_I2S/PCM0
Port52		Port52	
Port53		Port53	

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.9.3.4 DMA Descriptor

The DMAC descriptor is the configuration information of DMA transfer that decides the DMA working mode. Each descriptor includes 6 words: Configuration, Source Address, Destination Address, Byte Counter, Parameter, and Link. The following figure shows the structure of the DMA descriptor.

Figure 3-20 DMA Descriptor

1. **Configuration:** Configure the following information by DMA_CFG_REG.
   - DRQ type: DRQ type of the source and destination devices.
   - Address counting mode: For both the source and destination devices, there are two address counting modes: the IO mode and linear mode.

     The IO mode is for IO devices whose address is fixed during the data transfer and the linear mode is for the memory whose address is increasing during the data transfer.
   - Transferred block length: The amount of data that non-memory peripherals can transfer in a valid DRQ. The block length supports 1 bit, 4 bits, 8 bits, and 16 bits.
   - Transferred data width: The data width of operating the non-memory peripherals. The data width supports 8 bits, 16 bits, 32 bits, and 64 bits.

   NOTE

   The configuration supports BMODE mode. The BMODE is used in the following scenario: the source is an IO device, and the destination is a memory device. Setting the BMODE mode can limit the amount of block data transferred in DMA block transmission to the amount of data transferred when the DRQ threshold of the source IO device is 1.

2. **Source Address:** Configure the address of the source device.
3. **Destination Address:** Configure the address of the destination device.

DMA reads data from the source address and then writes data to the destination address.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Both the DMA source and destination addresses have 34 bits. In the descriptor, because there are only 32 bits in the **Source/Destination Address** field, another 2 bits are stored in the **Parameter** field.

The following table shows the details of the related fields in the descriptor.

Table 3-13 Source/Destination Address Distribution

Descriptor Group	Bit	Description
Source Address	31:0	DMA transfers the lower 32 bits of the 34-bit source address
Destination Address	31:0	DMA transfers the lower 32 bits of the 34-bit destination address
Parameter	31:20	Reserved
	19:18	DMA transfers the higher 2 bits of the 34-bit destination address
	17:16	DMA transfers the high 2 bits of the 34-bit source address
	15:8	Reserved
	7:0	Wait Clock Cycles Set the waiting time in DRQ mode
Link	31:2	The address of the next group descriptor, the lower 30 bits of the word address
	1:0	The address of the next group descriptor, the higher 2 bits of the word address

From the above table, you can get:

Real DMA source address (in byte mode) = {Parameter [17:16], Source Address [31:0]};

Real DMA destination address (in byte mode) = {Parameter [19:18], Destination Address [31:0]};

Real link address (in byte mode) = {Link[1:0], Link[31:2], 2'b00}.

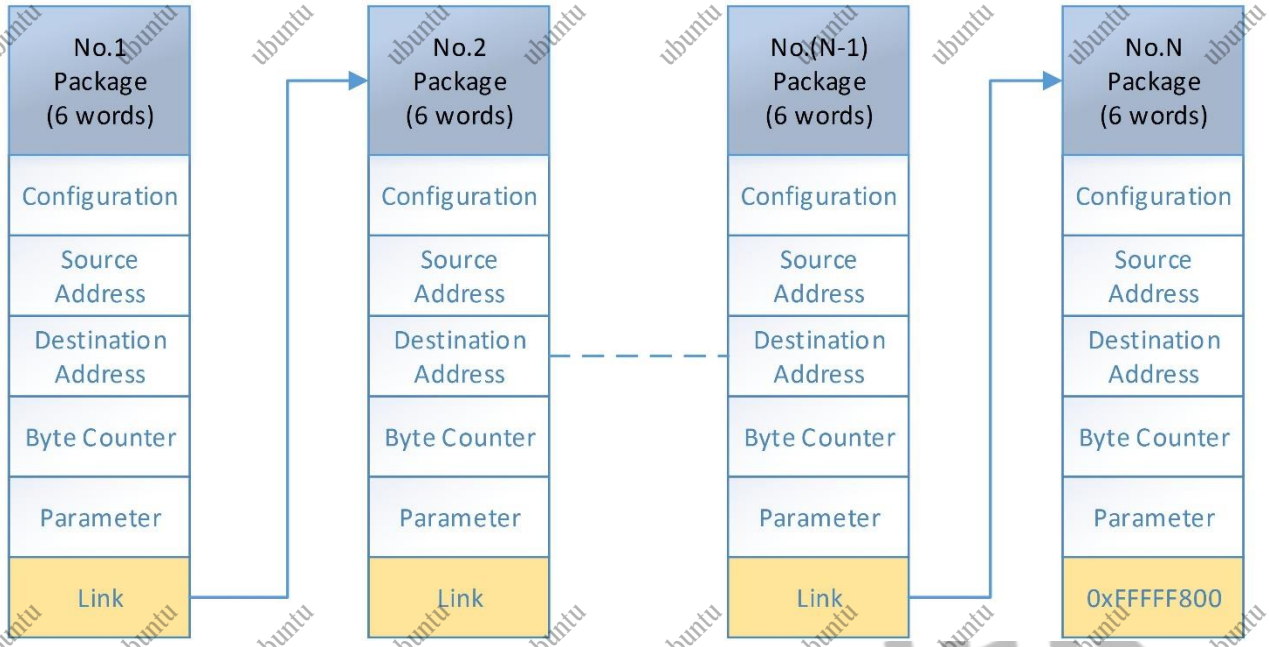
4. **Byte counter:** Configure the data amount of a package. The maximum value is (2^25-1) bytes. If the data amount of the package reaches the maximum value, even if DRQ is valid, the DMA will stop the current transfer.

5. **Parameter:** Configure the interval between the data block. The parameter is valid for non-memory peripherals. When DMA detects that the DRQ is high, the DMA transfers the data block and ignores the status changes of the DRQ until the data transfer finishes. After that, the DMA waits for certain clock cycles (WAIT_CYC) and executes the next DRQ detection.

6. **Link:** If the value of the link is 0xFFFFF800, the current package is at the end of the linked list. The DMAC will stop the data transfer after transferring the package; otherwise, the value of the link is considered as the descriptor address of the next package.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 3-21 DMA Chain Transfer



3.9.3.5 Interrupts

There are three kinds of DMA interrupts: the half package interrupt, package end interrupt, and queue end interrupt.

Half package interrupt: When enabled, the DMAC sends out a half package interrupt after transferring half of a package.

Package end interrupt: When enabled, the DMAC sends out a package end interrupt after transferring a complete package.

Queue end interrupt: When enabled, the DMAC sends out a queue end interrupt after transferring a complete queue.

Notice that when CPU does not respond to the interrupts timely, or two DMA interrupts are generated very closely, the later interrupt may override the former one. That is, from the perspective of the CPU, the DMAC has only a system interrupt source.

3.9.3.6 Security

The DMAC supports the system Trustzone and the security mode configuration of the DMA channel.

Each DMA channel is secure by default. When the system Trustzone is enabled, the DMAC is secure, only the secure devices can access the DMAC.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

When a DMA channel is configured to the non-secure mode, this channel can only access the non-secure area of the memory, that is, the DMAC cannot write data to the secure area of the memory, and the read-back data from the secure area of the memory is 0.

3.9.3.7 Clock Gating

The DMA_CLK_GATE module is a hardware module for controlling the clock gating automatically. It provides clock sources for sub-modules in DMAC and the module local circuits.

The DMA_CLK_GATE module consists of two parts: the channel clock gate and the common clock gate.

Channel clock gate: Controls the DMA clock of the DMA channels. When the system accesses the register of the current DMA channel and the DMA channel is enabled, the channel clock gate automatically opens the DMA clock. With a 16-HCLK-cycle delay after the system finishes accessing the register or the DMA data transfer is completed, the channel clock gate automatically closes the DMA clock. Also, the clock for the related circuits, such as for the channel control and FIFO control modules, will be closed.

Common clock gate: Controls the clocks of the DMA common circuits. The common circuits include the common circuit of the FIFO control module, MPORT module, and MBUS. When all the DMA channels are enabled, the common clock gate automatically closes the clocks for the above circuits.

The DMA clock gating can support all the functions stated above or not by software.

3.9.3.8 Transfer Mode

The peripherals initiate data transfer by transmitting DMA request signals to the DMAC. After receiving the request signal, the DMAC converts it to the internal DRQ signal and controls the DMA data transfer.

The DMAC supports two data transfer modes: the waiting mode and handshake mode.

Waiting mode

- When the DMAC detects a valid external request signal, the DMAC starts to operate the peripheral device. The internal DRQ always holds high before the transferred data amount reaches the transferred block length.
- When the transferred data amount reaches the transferred block length, the internal DRQ pulls low automatically.
- The internal DRQ holds low for certain clock cycles (WAIT_CYC), and then the DMAC restarts to detect the external requests. If the external request signal is valid, then the next transfer starts.

Handshake Mode

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

- When the DMAC detects a valid external request signal, the DMAC starts to operate the peripheral device. The internal DRQ always holds high before the transferred data amount reaches the transferred block length.
- When the transferred data amount reaches the transferred block length, the internal DRQ will be pulled down automatically. For the last data transfer of the block, the DMAC sends a DMA Last signal with the DMA commands to the peripheral device. The DMA Last signal will be packed as part of the DMA commands and transmitted on the bus. It is used to inform the peripheral device that it is the end of the data transfer for the current DRQ.
- When the peripheral device receives the DMA Last signal, it can judge that the data transfer for the current DRQ is finished. To continue the data transfer, it sends a DMA Active signal to the DMAC.

NOTE

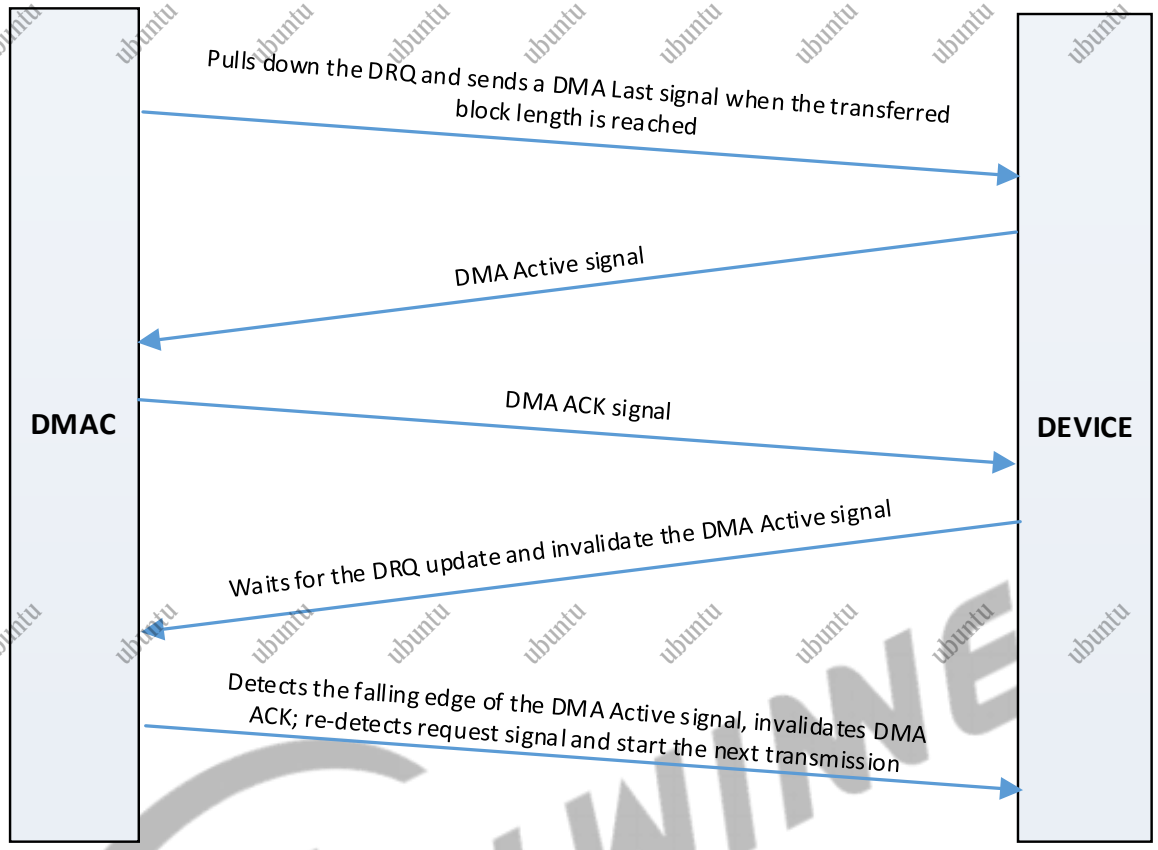
One DMA Active signal will be converted to one DRQ signal in the DMA module. To generate multiple DRQs, the peripheral device needs to send out multiple DMA Active signals via the bus protocol.

- When the DMAC received the DMA Active signal, it sends back a DMA ACK signal to the peripheral device.
- When the peripheral device receives the DMA ACK signal, it waits for all the operations on the local device completed, and both the FIFO and DRQ status refreshed. Then it invalidates the DMA Active signal.
- When the DMAC detects the falling edge of the DMA Active signal, it invalidates the corresponding DMA ACK signal, and restarts to detect the external request signals. If a valid request signal is detected, the next data transfer starts.

The following figure shows the workflow of the handshake mode.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 3-22 Workflow of the DMAC Handshake Mode



3.9.3.9 Address Auto-Alignment

For the non-IO devices whose start address is not 32-byte-aligned, the DMAC will adjust the address to 32-byte-aligned through the burst transfer within 32 bytes. Adjusting address to 32-byte-aligned improves the DRAM access efficiency.

The following example shows how the DMAC adjusts the address: when the peripheral device of a DMA channel is a non-IO device whose start address is 0x86 (not 32-byte-aligned), the DMAC firstly uses a 26-byte burst transfer to align the address to 0xA0 (32-byte-aligned), and then transfers data by 64-byte burst (the maximum transfer amount that MBUS allows).

The IO devices do not support address alignment, so the bit width of IO devices must match the address offset; otherwise, the DMAC will ignore the inconsistency and directly transmit data of the corresponding bit width to the address.

The address of the DMA descriptor does not support the address auto-alignment. Make sure the address is word-aligned; otherwise the DMAC cannot identify the descriptor.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.9.3.10 DMAC Clock Control

- The DMAC clock is synchronous with the AHB1 clock. Make sure that the DMAC gating bit of AHB1 clock is enabled before accessing the DMAC register.
- The reset input signal of the DMAC is asynchronous with AHB1 and is low valid by default. Make sure that the reset signal of the DMAC is de-asserted before accessing the DMA register.
- To avoid the indefinite state within registers, de-assert the reset signal first, and then open the gating bit of AHB1.
- The DMAC supports Clock Auto Gating, that is, to reduce power consumption, the system will automatically disable the DMAC clock in the DMAC idle state. Clock Auto Gating is enabled by default.

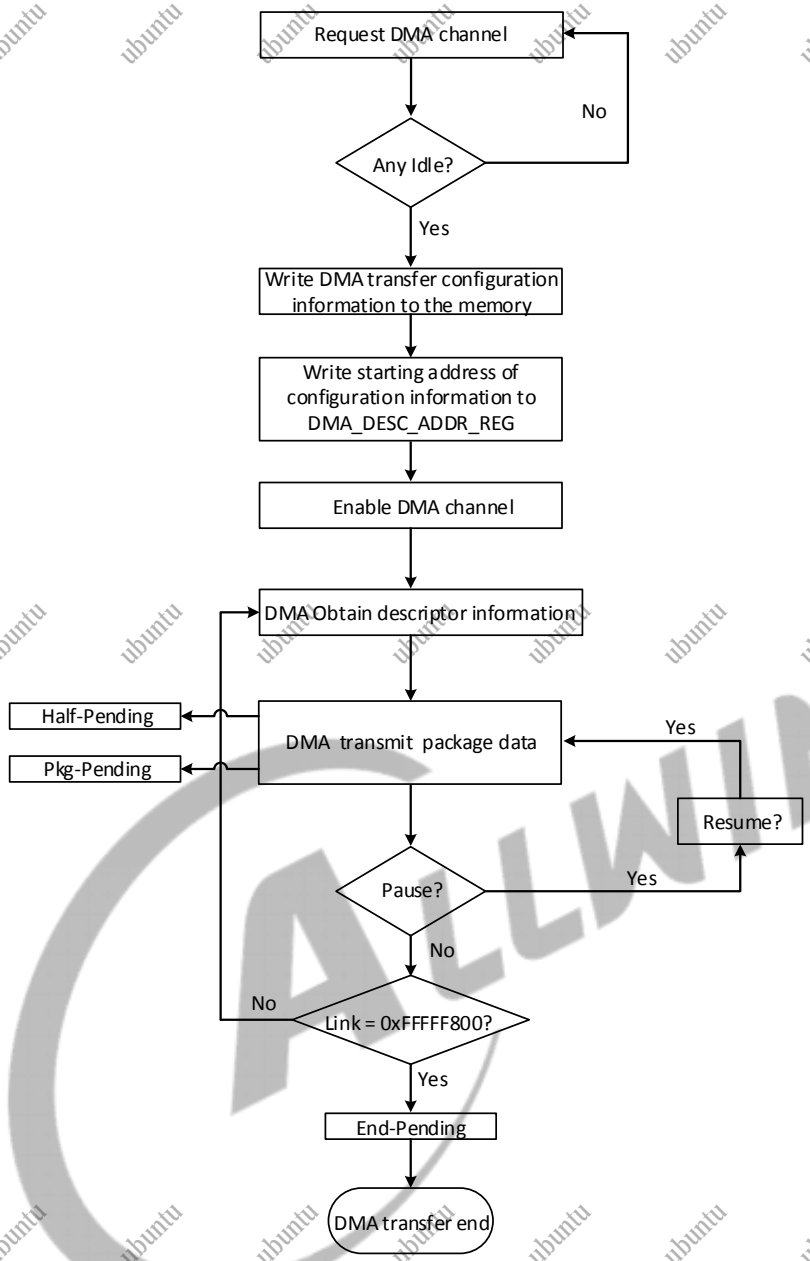
3.9.3.11 DMAC Transfer Process

The DMAC transfer process is as follows.

1. Request DMA channel, and check if the DMA channel is idle by checking if it is enabled. A disabled channel indicates it is idle, while an enabled channel indicates it is busy.
2. Write the descriptor with 6 words into the memory. The descriptor must be word-aligned. For more details, refer to section 3.9.3.4 "DMA Descriptor".
3. Write the start address of the descriptor to **DMA_DESC_ADDR_REGN**.
4. Enable the DMA channel, and write the corresponding channel to **DMAC_EN_REGN**.
5. The DMA obtains the descriptor information.
6. Start to transmit a package. When half of the package is completed, the DMA sends a *Half Package Transfer Interrupt*; when a total package is completed, the DMA sends a *Package End Transfer Interrupt*. These interrupt status can be read by **DMAC_IRQ_PEND_REG0**.
7. Set **DMAC_PAU_REGN** to pause or resume the data transfer.
8. After completing a total package transfer, the DMA decides to start the next package transfer or end the transfer by the link of the descriptor. If the link is 0xFFFFF800, the transfer ends; otherwise, the next package starts to transmit. When the transfer ends, the DMA sends a *Queue End Transfer Interrupt*.
9. Disable the DMA channel.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 3-23 DMAC Transfer Process



3.9.3.12 Processing the DMAC Interrupt

Follow the steps below to process the DMAC interrupt:

1. Enable interrupt: write the corresponding interrupt enable bit of **DMAC_IRQ_EN_REG0**. The system generates an interrupt when the corresponding condition is satisfied.
2. After entering the interrupt process, write to clear the interrupt pending and execute the process of waiting for the interrupt.
3. Resume the interrupt and continue to execute the interrupted process.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.9.4 Programming Guidelines

To configure the DMAC, follow the guidelines below:

- Make sure the transfer bit width of IO devices is consistent with the offset of the start address.
- The MBUS protocol does not support the read operation of non-integer words. For the devices whose bit width is not word-aligned, after receiving the read command, they should resolve the read command according to their FIFO bit width instead of the command bit width, and ignore the redundant data caused by the inconsistency of the bit width.
- When the DMA transfer is paused, this is equivalent to invalid DRQ. Because there is a certain time delay between DMA transfer commands, the DMAC will not stop data transfer until the DMAC finishes processing the current command and the commands in Arbiter (at most 32 bytes data).

DMAC application example:

```
writel(0x00000000, mem_address + 0x00);   //Set configurations. The mem_address must be word-aligned.
writel(0x00001000, mem_address + 0x04);    // Set the start address for the source device.
writel(0x20000000, mem_address + 0x08);   //Set the start address for the destination device.
writel(0x00000020, mem_address + 0x0C);   // Set the data package size.
writel(0x00000000, mem_address + 0x10);   //Set the parameters.
writel(0xFFFFF800, mem_address + 0x14);   //Set the start address for the next descriptor.
writel(mem_address, 0x01C02000+ 0x100 + 0x08); //Set the start address for the DMA channel0 descriptor.
do{
If(mem_address == readl(0x01C02000 + 0x100 + 0x08));
break;
}while(1);              //Make sure that the writing operation is valid.
writel(0x000000001, 0x01C02000 + 0x100 + 0x00);      // Enable DMA channel0 transfer.
```

The DMAC supports increasing data package in transfer, pay attention to the following points:

- The 0xFFFFF800 value of **DMAC_FDESC_ADDR_REGN** indicates that the DMA channel has got back the descriptor of the last package. The DMA channel will automatically stop the data transfer after transferring the current package.
- To add a package during the data transfer, check if the DMA channel has got back the descriptor of the last package. If yes, you cannot add any package in the current queue. Request another DMA channel with a new DRQ to transfer the package. Otherwise, you can add the package by modifying the

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

DMAC_FDESC_ADDR_REGN of the last package from 0xFFFFF800 to the start address of the to-be-added package.

To ensure that the modification is valid, read the value of DMAC_FDESC_ADDR_REGN after the modification. The value 0xFFFFF800 indicates the modification fails and the other values indicate you have successfully added packages to the queue.

Another problem is, the system needs some time to process the modification, during which the DMA channel may get back the descriptor of the last package. You can read DMAC_CUR_SRC_REGN and DMAC_CUR_DEST_REGN and check if the increasing memory address accords with the information of the added package. If yes, the package is added successfully; otherwise, the modification failed.

To ensure a higher rate of success, it is suggested that you add the package before the half package interrupt of the penultimate package.

3.9.5 Register List

Module Name	Base Address
DMAC	0x03002000
R_DMAC	0x07091000

Register Name	Offset	Description
DMAC_IRQ_EN_REG0	0x0000	DMAC IRQ Enable Register 0
DMAC_IRQ_PEND_REG0	0x0010	DMAC IRQ Pending Register 0
DMAC_SEC_REG	0x0020	DMAC Security Register
DMAC_AUTO_GATE_REG	0x0028	DMAC Auto Gating Register
DMAC_STA_REG	0x0030	DMAC Status Register
DMAC_EN_REGN	0x0100 + N*0x0040	DMAC Channel Enable Register N (N = 0 to 7)
DMAC_PAU_REGN	0x0104 + N*0x0040	DMAC Channel Pause Register N (N = 0 to 7)
DMAC_DESC_ADDR_REGN	0x0108 + N*0x0040	DMAC Channel Start Address Register N (N = 0 to 7)
DMAC_CFG_REGN	0x010C + N*0x0040	DMAC Channel Configuration Register N (N = 0 to 7)
DMAC_CUR_SRC_REGN	0x0110 + N*0x0040	DMAC Channel Current Source Register N (N = 0 to 7)
DMAC_CUR_DEST_REGN	0x0114 + N*0x0040	DMAC Channel Current Destination Register N (N = 0 to 7)
DMAC_BCNT_LEFT_REGN	0x0118 + N*0x0040	DMAC Channel Byte Counter Left Register N (N = 0 to 7)
DMAC_PARA_REGN	0x011C + N*0x0040	DMAC Channel Parameter Register N (N = 0 to 7)
DMAC_MODE_REGN	0x0128 + N*0x0040	DMAC Mode Register N (N = 0 to 7)

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
DMAC_FDESC_ADDR_REGN	0x012C + N*0x0040	DMAC Former Descriptor Address Register N (N = 0 to 7)
DMAC_PKG_NUM_REGN	0x0130 + N*0x0040	DMAC Package Number Register N (N = 0 to 7)

3.9.6 Register Description

3.9.6.1 0x0000 DMAC IRQ Enable Register0 (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: DMAC_IRQ_EN_REG0
Bit	Read/Write	Default/Hex	Description
31	/	/	/
30	R/W	0x0	DMA7_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 7. 0: Disabled 1: Enabled
29	R/W	0x0	DMA7_PKG_IRQ_EN Enable the package end interrupt of DMA channel 7. 0: Disabled 1: Enabled
28	R/W	0x0	DMA7_HLAF_IRQ_EN Enable the half package interrupt of DMA channel 7. 0: Disabled 1: Enabled
27	/	/	/
26	R/W	0x0	DMA6_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 6. 0: Disabled 1: Enabled
25	R/W	0x0	DMA6_PKG_IRQ_EN Enable the package end interrupt of DMA channel 6. 0: Disabled 1: Enabled
24	R/W	0x0	DMA6_HLAF_IRQ_EN Enable the half package interrupt of DMA channel 6. 0: Disabled 1: Enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: DMAC_IRQ_EN_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
23	/	/	/
22	R/W	0x0	DMA5_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 5. 0: Disabled 1: Enabled
21	R/W	0x0	DMA5_PKG_IRQ_EN Enable the package end interrupt of DMA channel 5. 0: Disabled 1: Enabled
20	R/W	0x0	DMA5_HLAF_IRQ_EN Enable the half package interrupt of DMA channel 5. 0: Disabled 1: Enabled
19	/	/	/
18	R/W	0x0	DMA4_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 4. 0: Disabled 1: Enabled
17	R/W	0x0	DMA4_PKG_IRQ_EN Enable the package end interrupt of DMA channel 4. 0: Disabled 1: Enabled
16	R/W	0x0	DMA4_HLAF_IRQ_EN Enable the half package interrupt of DMA channel 4. 0: Disabled 1: Enabled
15	/	/	/
14	R/W	0x0	DMA3_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 3. 0: Disabled 1: Enabled
13	R/W	0x0	DMA3_PKG_IRQ_EN Enable the package end interrupt of DMA channel 3. 0: Disabled 1: Enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: DMAC_IRQ_EN_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
12	R/W	0x0	DMA3_HLAF_IRQ_EN Enable the half package interrupt of DMA channel 3. 0: Disabled 1: Enabled
11	/	/	/
10	R/W	0x0	DMA2_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 2. 0: Disabled 1: Enabled
9	R/W	0x0	DMA2_PKG_IRQ_EN Enable the package end interrupt of DMA channel 2. 0: Disabled 1: Enabled
8	R/W	0x0	DMA2_HLAF_IRQ_EN Enable the half package interrupt of DMA channel 2. 0: Disabled 1: Enabled
7	/	/	/
6	R/W	0x0	DMA1_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 1. 0: Disabled 1: Enabled
5	R/W	0x0	DMA1_PKG_IRQ_EN Enable the package end interrupt of DMA channel 1. 0: Disabled 1: Enabled
4	R/W	0x0	DMA1_HLAF_IRQ_EN Enable the half package interrupt of DMA channel 1. 0: Disabled 1: Enabled
3	/	/	/
2	R/W	0x0	DMA0_QUEUE_IRQ_EN Enable the queue end interrupt of DMA channel 0. 0: Disabled 1: Enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: DMAC_IRQ_EN_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
1	R/W	0x0	DMA0_PKG_IRQ_EN Enable the package end interrupt of DMA channel 0. 0: Disabled 1: Enabled
0	R/W	0x0	DMA0_HLAF_IRQ_EN Enable the half package interrupt of DMA channel 0. 0: Disabled 1: Enabled

3.9.6.2 0x0010 DMAC IRQ Pending Status Register 0 (Default Value: 0x0000_0000)

Offset: 0x0010			Register Name: DMAC_IRQ_PEND_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31	/	/	/
30	R/W1C	0x0	DMA7_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 7. Write 1 to clear the pending status. 0: No effect 1: Pending
29	R/W1C	0x0	DMA7_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 7. Write 1 to clear the pending status. 0: No effect 1: Pending
28	R/W1C	0x0	DMA7_HLAF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 7. Write 1 to clear the pending status. 0: No effect 1: Pending
27	/	/	/
26	R/W1C	0x0	DMA6_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 6. Write 1 to clear the pending status. 0: No effect 1: Pending

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0010			Register Name: DMAC_IRQ_PEND_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
25	R/W1C	0x0	DMA6_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 6. Write 1 to clear the pending status. 0: No effect 1: Pending
24	R/W1C	0x0	DMA6_HLAF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 6. Write 1 to clear the pending status. 0: No effect 1: Pending
23	/	/	/
22	R/W1C	0x0	DMA5_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 5. Write 1 to clear the pending status. 0: No effect 1: Pending
21	R/W1C	0x0	DMA5_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 5. Write 1 to clear the pending status. 0: No effect 1: Pending
20	R/W1C	0x0	DMA5_HLAF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 5. Write 1 to clear the pending status. 0: No effect 1: Pending
19	/	/	/
18	R/W1C	0x0	DMA4_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 4. Write 1 to clear the pending status. 0: No effect 1: Pending
17	R/W1C	0x0	DMA4_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 4. Write 1 to clear the pending status. 0: No effect 1: Pending

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0010			Register Name: DMAC_IRQ_PEND_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
16	R/W1C	0x0	DMA4_HLAF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 4. Write 1 to clear the pending status. 0: No effect 1: Pending
15	/	/	/
14	R/W1C	0x0	DMA3_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 3. Write 1 to clear the pending status. 0: No effect 1: Pending
13	R/W1C	0x0	DMA3_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 3. Write 1 to clear the pending status. 0: No effect 1: Pending
12	R/W1C	0x0	DMA3_HLAF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 3. Write 1 to clear the pending status. 0: No effect 1: Pending
11	/	/	/
10	R/W1C	0x0	DMA2_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 2. Write 1 to clear the pending status. 0: No effect 1: Pending
9	R/W1C	0x0	DMA2_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 2. Write 1 to clear the pending status. 0: No effect 1: Pending
8	R/W1C	0x0	DMA2_HLAF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 2. Write 1 to clear the pending status. 0: No effect 1: Pending

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0010			Register Name: DMAC_IRQ_PEND_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
7	/	/	/
6	R/W1C	0x0	DMA1_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 1. Write 1 to clear the pending status. 0: No effect 1: Pending
5	R/W1C	0x0	DMA1_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 1. Write 1 to clear the pending status. 0: No effect 1: Pending
4	R/W1C	0x0	DMA1_HLAF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 1. Write 1 to clear the pending status. 0: No effect 1: Pending
3	/	/	/
2	R/W1C	0x0	DMA0_QUEUE_IRQ_PEND The IRQ pending bit for the queue end interrupt of the DMA channel 0. Write 1 to clear the pending status. 0: No effect 1: Pending
1	R/W1C	0x0	DMA0_PKG_IRQ_PEND The IRQ pending bit for the package end interrupt of the DMA channel 0. Write 1 to clear the pending status. 0: No effect 1: Pending
0	R/W1C	0x0	DMA0_HLAF_IRQ_PEND The IRQ pending bit for the half package interrupt of the DMA channel 0. Write 1 to clear the pending status. 0: No effect 1: Pending

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.9.6.3 0x0020 DMAC Security Register (Default Value: 0x0000_00FF)

Offset: 0x0020			Register Name: DMAC_SEC_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R/W	0xFF	DMA_SEC Configure the security mode for DMA channel[7:0]. The meaning of each bit: 0: Secure mode 1: Non-secure mode

3.9.6.4 0x0028 DMAC Auto Gating Register (Default Value: 0x0000_0000)

Offset: 0x0028			Register Name: DMAC_AUTO_GATE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2	R/W	0x0	DMA_MCLK_CIRCUIT Auto gating bit of DMA MCLK interface circuit 0: Auto gating enabled 1: Auto gating disabled
1	R/W	0x0	DMA_COMMON_CIRCUIT Auto gating bit of DMA common circuit 0: Auto gating enabled 1: Auto gating disabled
0	R/W	0x0	DMA_CHAN_CIRCUIT Auto gating bit of DMA channel circuit 0: Auto gating enabled 1: Auto gating disabled

NOTE

When initializing the DMA Controller, the bit[2] should be set up.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.9.6.5 0x0030 DMAC Status Register (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: DMAC_STA_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R	0x0	MBUS FIFO Status 0: Empty 1: Not Empty
30:8	/	/	/
7:0	R	0x0	DMA_STATUS DMA Channel[7:0] Status The meaning of each bit: 0: Idle 1: Busy

3.9.6.6 0x0100 + N*0x0040 DMAC Channel Enable Register N (Default Value: 0x0000_0000)

Offset: 0x0100 + N*0x0040 (N = 0 to 7)			Register Name: DMAC_EN_REGN
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	DMA_EN DMA Channel Enable 0: Disabled 1: Enabled

3.9.6.7 0x0104 + N*0x0040 DMAC Channel Pause Register N (Default Value: 0x0000_0000)

Offset: 0x0104 + N*0x0040 (N = 0 to 7)			Register Name: DMAC_PAU_REGN
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	DMA_PAUSE Pause the DMA Channel Transfer Data 0: Resume Transferring 1: Pause Transferring

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.9.6.8 0x0108 + N*0x0040 DMAC Channel Descriptor Address Register N (Default Value: 0x0000_0000)

Offset: 0x0108 + N*0x0040 (N = 0 to 7)			Register Name: DMAC_DESC_ADDR_REGN
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	R/W	0x0	DMA_DESC_ADDR Lower 30 bits of DMA channel descriptor address The descriptor address must be word-aligned.
1:0	R/W	0x0	DMA_DESC_HIGH_ADDR Higher 2 bits of DMA channel descriptor high address The real address is as follows. DMA Channel Descriptor Address = {bit[1:0], bit[31:2], 2'b00}

3.9.6.9 0x010C + N*0x0040 DMAC Channel Configuration Register N (Default Value: 0x0000_0000)

Offset: 0x010C + N*0x0040 (N = 0 to 7)			Register Name: DMAC_CFG_REGN
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R	0x0	BMODE_SEL 0: Normal Mode 1: BMODE
30:27	/	/	/
26:25	R	0x0	DMA_DEST_DATA_WIDTH DMA Destination Data Width 00: 8-bit 01: 16-bit 10: 32-bit 11: 64-bit
24	R	0x0	DMA_ADDR_MODE DMA Destination Address Mode 0: Linear Mode 1: IO Mode
23:22	R	0x0	DMA_DEST_BLOCK_SIZE DMA Destination Block Size 00: 1 01: 4 10: 8 11: 16

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x010C + N*0x0040 (N = 0 to 7)			Register Name: DMAC_CFG_REGN
Bit	**Read/Write**	**Default/Hex**	**Description**
21:16	R	0x0	DMA_DEST_DRQ_TYPE DMA Destination DRQ Type The details in DRQ Type and Port Corresponding Relation.
15:11	/	/	/
10:9	R	0x0	DMA_SRC_DATA_WIDTH DMA Source Data Width 00: 8-bit 01: 16-bit 10: 32-bit 11: 64-bit
8	R	0x0	DMA_SRC_ADDR_MODE DMA Source Address Mode 0: Linear Mode 1: IO Mode
7:6	R	0x0	DMA_SRC_BLOCK_SIZE DMA Source Block Size 00: 1 01: 4 10: 8 11: 16
5:0	R	0x0	DMA_SRC_DRQ_TYPE DMA Source DRQ Type The details in DRQ Type and Port Corresponding Relation.

3.9.6.10 0x0110 + N*0x0040 DMAC Channel Current Source Address Register N (Default Value: 0x0000_0000)

Offset: 0x0110 + N*0x0040 (N = 0 to 7)			Register Name: DMAC_CUR_SRC_REGN
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	DMA_CUR_SRC DMA Channel Current Source Address.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.9.6.11 0x0114 + N*0x0040 DMAC Channel Current Destination Address Register N (Default Value: 0x0000_0000)

Offset: 0x0114 + N*0x0040 (N = 0 to 7)			Register Name: DMAC_CUR_DEST_REGN
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	DMA_CUR_DEST DMA Channel Current Destination Address.

3.9.6.12 0x0118 + N*0x0040 DMAC Channel Byte Counter Left Register N (Default Value: 0x0000_0000)

Offset: 0x0118 + N*0x0040 (N = 0 to 7)			Register Name: DMAC_BCNT_LEFT_REGN
Bit	**Read/Write**	**Default/Hex**	**Description**
31:25	/	/	/
24:0	R	0x0	DMA_BCNT_LEFT DMA Channel Byte Counter Left.

3.9.6.13 0x011C + N*0x0040 DMAC Channel Parameter Register N (Default Value: 0x0000_0000)

Offset: 0x011C + N*0x0040 (N = 0 to 7)			Register Name: DMAC_PARA_REGN
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R	0x0	WAIT_CYC Wait Clock Cycles

3.9.6.14 0x0128 + N*0x0040 DMAC Mode Register N (Default Value: 0x0000_0000)

Offset: 0x0128 + N*0x0040 (N = 0 to 7)			Register Name: DMAC_MODE_REGN
Bit	**Read/Write**	**Default/Hex**	**Description**
31:4	/	/	/
3	R/W	0x0	DMA_DST_MODE 0: Waiting mode 1: Handshake mode
2	R/W	0x0	DMA_SRC_MODE 0: Waiting mode 1: Handshake mode

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0128 + N*0x0040 (N = 0 to 7)			Register Name: DMAC_MODE_REGN
Bit	Read/Write	Default/Hex	Description
1:0	/	/	/

3.9.6.15 0x012C + N*0x0040 DMAC Former Descriptor Address Register N (Default Value: 0x0000_0000)

Offset: 0x012C + N*0x0040 (N = 0 to 7)			Register Name: DMAC_FDESC_ADDR_REGN
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	DMA_FDESC_ADDR This register is used to store the former value of DMA Channel Descriptor Address Register.

3.9.6.16 0x0130 + N*0x0040 DMAC Package Number Register N (Default Value: 0x0000_0000)

Offset: 0x0130 + N*0x0040 (N = 0 to 7)			Register Name: DMAC_PKG_NUM_REGN
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	DMA_PKG_NUM This register records the number of packages which has been completed in one transmission.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.10 Thermal Sensor Controller (THS)

3.10.1 Overview

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

The thermal sensor controller (THS) embeds one thermal sensor located in the CPU. When the temperature reaches a certain thermal threshold, thermal sensor can generate interrupts to the software to lower the temperature via the dynamic voltage and frequency scaling (DVFS) technology.

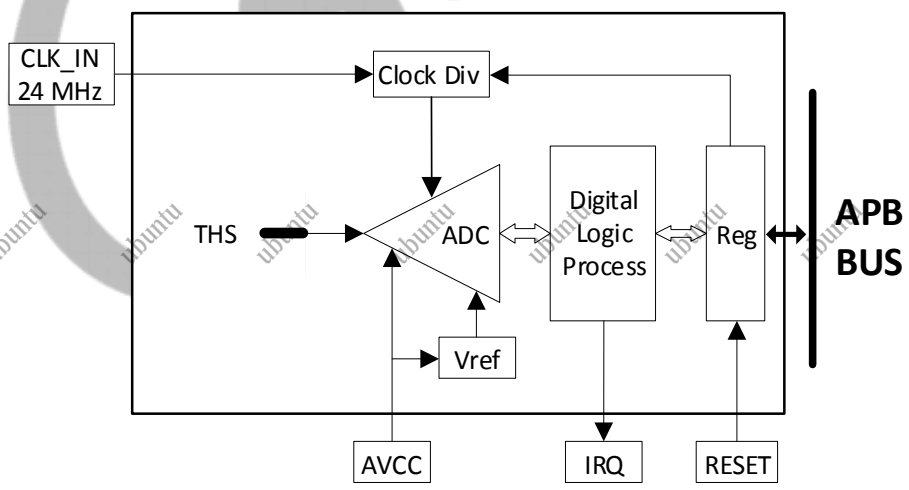
The THS has the following features:

- Temperature accuracy: ±3°C from 0°C to +100°C, ±5°C from -25°C to +125°C
- Power supply voltage: 1.8 V
- Averaging filter for thermal sensor reading
- Supports over-temperature protection interrupt and over-temperature alarm interrupt

3.10.2 Block Diagram

The following figure shows a block diagram of the THS.

Figure 3-24 THS Block Diagram



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.10.3 Functional Descriptions

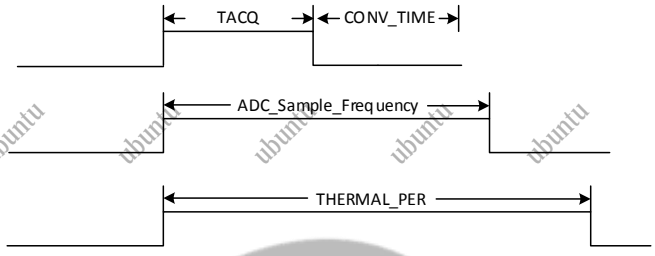
3.10.3.1 Clock Source

The THS gets one clock source: DCXO24M. For details about clock configurations, refer to section 3.3 “CCU”.

3.10.3.2 Timing Requirements

The following figure shows the timing requirements for the THS.

Figure 3-25 Thermal Sensor Timing Requirement



CLK_IN = 24 MHz

CONV_TIME (Conversion Time) = 1/24 MHz x 14 Cycles = 0.583 us

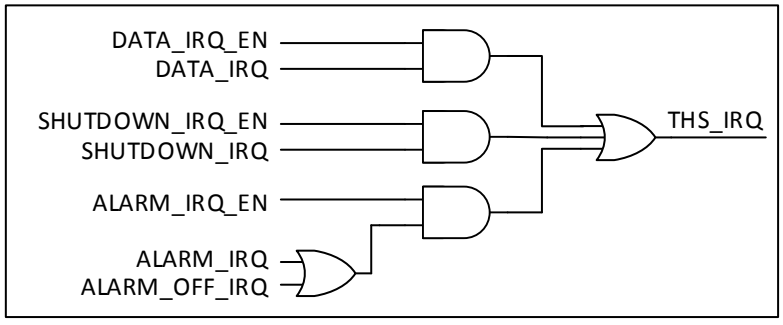
TACQ > 1/24 MHz x 24 Cycles

THERMAL_PER > ADC_Sample_Frequency > TACQ + CONV_TIME

3.10.3.3 Interrupts

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

Figure 3-26 Thermal Sensor Controller Interrupt Source



DATA_IRQ: The interrupt is generated when the measured sensor_data is updated.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

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

ALARM_OFF_IRQ: The interrupt is generated when the temperature drops to lower than the Alarm_Off_Thershold. It is triggered at the fall edge.

3.10.3.4 THS Temperature Conversion Formula

T = (sensor_data - 2794)/(-14.882)

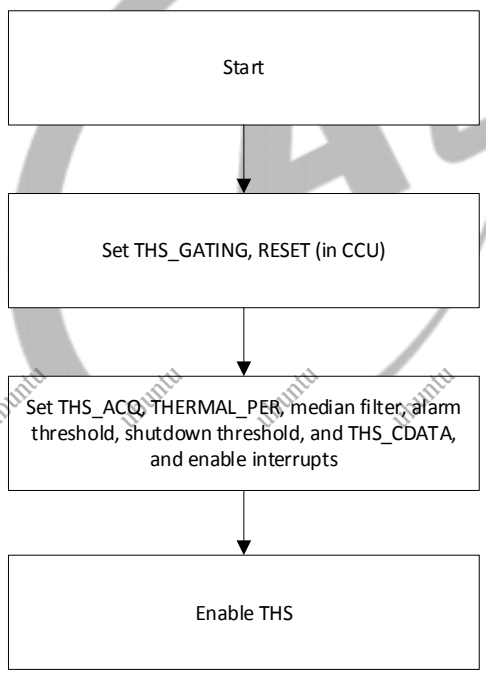
Unit of T: Celsius degree (°C).

The sensor_data is read from the sensor data register.

3.10.4 Programming Guidelines

The initial process of the THS is as follows.

Figure 3-27 THS Initial Process



In the final test (FT) stage, THS is calibrated through the ambient temperature, and the calibration value is written in the SID module. The following table shows the THS information in the SID.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Table 3-14 THS Information in the SID

Base Address: 0x14	Register Name: THS
Bit	**Description**
31:28	The highest 4 bits of the Audio_bias calibration value.
27:16	The calibration value of the T-sensor.
15:12	The lowest 4 bits of the Audio_bias calibration value.
11:0	The value of the ambient temperature.

Before enabling THS, read eFuse value and write the value to **THS_CDATA**.

Query Mode

Step 1 Write 0x1 to the bit[16] of **THS_BGR_REG** to dessert the reset.

Step 2 Write 0x1 to the bit[0] of **THS_BGR_REG** to open the THS clock.

Step 3 Write 0x2F to the bit[15:0] of **THS_CTRL** to set the ADC acquire time.

Step 4 Write 0x1DF to the bit[31:16] of **THS_CTRL** to set the ADC sample frequency divider.

Step 5 Write 0x3A to the bit[31:12] of **THS_PER** to set the THS work period.

Step 6 Write 0x1 to the bit[2] of **THS_FILTER** to enable the temperature convert filter.

Step 7 Write 0x1 to the bit[1:0] of **THS_FILTER** to select the filter type.

Step 8 Read THS eFuse value from SID, then write the eFuse value to **THS_CDATA** to calibrate THS.

Step 9 Write 0x1 to the bit[0] 0f THS0_EN to enable THS.

Step 10 Read the bit[0] of **THS_DATA_INTS**. If it is 1, the temperature conversion is complete.

Step 11 Read the bit[11:0] of **THS_DATA**, and calculate the THS temperature based on section 3.10.3.4 "THS Temperature Conversion Formula".

Interrupt Mode

Step 1 Write 0x1 to the bit16 of **THS_BGR_REG** to dessert the reset.

Step 2 Write 0x1 to the bit0 of **THS_BGR_REG** to open the THS clock.

Step 3 Write 0x2F to the bit[15:0] of **THS_CTRL** to set the ADC acquire time.

Step 4 Write 0x1DF to the bit[31:16] of **THS_CTRL** to set the ADC sample frequency divider.

Step 5 Write 0x3A to the bit[31:12] of **THS_PER** to set the THS work period.

Step 6 Write 0x1 to the bit2 of **THS_FILTER** to enable the temperature convert filter.

Step 7 Write 0x1 to the bit[1:0] of **THS_FILTER** to select the filter type.

Step 8 Read THS eFuse value from SID, and then write the eFuse value to **THS_CDATA** to calibrate THS.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 9 Write 0x1 to the bit[0] of **THS_DATA_INTC** to enable the interrupt of THS.

Step 10 Set GIC interface based on IRQ 149, write the bit[21] of the **0x03021110** register to 0x1.

Step 11 Put the interrupt handler address into the interrupt vector table.

Step 12 Write 0x1 to the bit[0] 0f THS0_EN to enable THS.

Step 13 Read the bit[0] of **THS_DATA_INTS**. If it is 1, the temperature conversion is complete.

Step 14 Read the bit[11:0] of **THS_DATA**, and calculate the THS temperature based on section 3.10.3.4 "THS Temperature Conversion Formula".

3.10.5 Register List

Module Name	Base Address
THS	0x07030400

Register Name	Offset	Description
THS_CTRL	0x0000	THS Control Register
THS_EN	0x0004	THS Enable Register
THS_PER	0x0008	THS Period Control Register
THS_DATA_INTC	0x0010	THS Data Interrupt Control Register
THS_SHUT_INTC	0x0014	THS Shut Interrupt Control Register
THS_ALARM_INTC	0x0018	THS Alarm Interrupt Control Register
THS_DATA_INTS	0x0020	THS Data Interrupt Status Register
THS_SHUT_INTS	0x0024	THS Shut Interrupt Status Register
THS_ALARM0_INTS	0x0028	THS Alarm off Interrupt Status Register
THS_ALARM_INTS	0x002C	THS Alarm Interrupt Status Register
THS_FILTER	0x0030	THS Median Filter Control Register
THS_ALARM_CTRL	0x0040	THS Alarm Threshold Control Register
THS_SHUTDOWN_CTRL	0x0080	THS Alarm Threshold Control Register
THS_CDATA	0x00A0	THS Calibration Data
THS_DATA	0x00C0	THS Data Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.10.6 Register Description

3.10.6.1 0x0000 THS Control Register (Default Value: 0x01DF_002F)

Offset: 0x0000			Register Name: THS_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	R/W	0x1DF	FS_DIV ADC Sample Frequency Divider CLK_IN/(N + 1), N > 0x17 The default value is 50 kHz.
15:0	R/W	0x2F	TACQ ADC Acquire Time CLK_IN/(n + 1) The default value is 2 us.

3.10.6.2 0x0004 THS Enable Register (Default Value: 0x0000_0000)

Offset: 0x0004			Register Name: THS_EN
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	THS0_EN Enable the temperature measurement sensor 0: Disabled 1: Enabled

3.10.6.3 0x0008 THS Period Control Register (Default Value: 0x0003_A000)

Offset: 0x0008			Register Name: THS_PER
Bit	**Read/Write**	**Default/Hex**	**Description**
31:12	R/W	0x3A	THERMAL_PER 4096*(n + 1)/CLK_IN The default value is 10 ms.
11:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.10.6.4 0x0010 THS Data Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0010			Register Name: THS_DATA_INTC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	THS_DATA_IRQ_EN Enable the interrupt of sensor_data update. If enabled, when the measured sensor_data is updated, it will generate an interrupt. 0: Disabled 1: Enabled

3.10.6.5 0x0014 THS Shut Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0014			Register Name: THS_SHUT_INTC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	SHUT_INT_EN Enable the shutdown interrupt for the sensor. 0: Disabled 1: Enabled

3.10.6.6 0x0018 THS Alarm Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0018			Register Name: THS_ALARM_INTC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	ALARM_INT_EN Enable the alarm interrupt for the sensor. 0: Disabled 1: Enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.10.6.7 0x0020 THS Data Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: THS_DATA_INTS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W1C	0x0	THS_DATA_IRQ_STS Indicates the pending status of the sensor's data interrupt. Write 1 to clear the pending status. 0: No effect 1: Pending

3.10.6.8 0x0024 THS Shut Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: THS_SHUT_INTS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W1C	0x0	SHUT_INT_STS Indicates the pending status of the sensor's shutdown interrupt. Write 1 to clear the pending status. 0: No effect 1: Pending

3.10.6.9 0x0028 THS Alarm Off Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0028			Register Name: THS_ALARM_INTS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W1C	0x0	ALARM_OFF_STS Alarm interrupt off pending for sensor Write 1 to clear the pending status. 0: No effect 1: Pending

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.10.6.10 0x002C THS Alarm Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x002C			Register Name: THS_ALARM_INTS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W1C	0x0	ALARM_INT_STS Alarm interrupt pending for sensor Write 1 to clear the pending status. 0: No effect 1: Pending

3.10.6.11 0x0030 Median Filter Control Register (Default Value: 0x0000_0001)

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

3.10.6.12 0x0040 THS Alarm Threshold Control Register (Default Value: 0x05A0_0684)

Offset: 0x0040			Register Name: THS_ALARM_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	/	/	/
27:16	R/W	0x5A0	ALARM_T_HOT Thermal Sensor alarm threshold for hot temperature
15:12	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0040			Register Name: THS_ALARM_CTRL
Bit	Read/Write	Default/Hex	Description
11:0	R/W	0x684	ALARM_T_HYST Thermal Sensor alarm threshold for hysteresis temperature

3.10.6.13 0x0080 THS Shutdown Threshold Control Register (Default Value: 0x04E9_04E9)

Offset: 0x0080			Register Name: THS_SHUTDOWN_CTRL
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
11:0	R/W	0x4E9	SHUT_T_HOT Thermal Sensor shutdown threshold for hot temperature

3.10.6.14 0x00A0 THS Calibration Data Register (Default Value: 0x0000_0800)

Offset: 0x00A0			Register Name: THS_CDATA
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:0	R/W	0x800	THS_CDATA Thermal Sensor calibration data

3.10.6.15 0x00C0 THS Data Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.11 Peripheral System Interconnect (PSI)

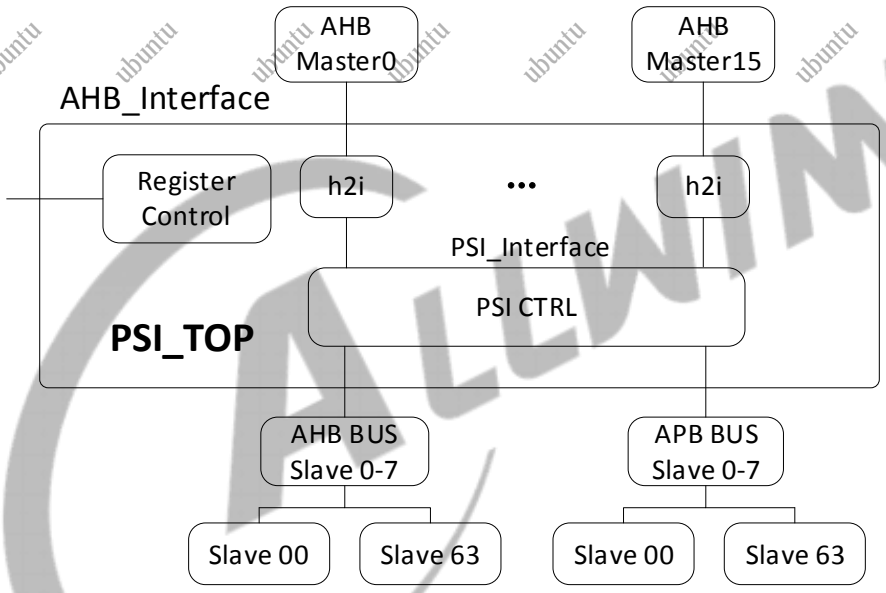
3.11.1 Overview

The Peripheral System Interconnect (PSI) is a peripheral bus interconnect device based on AHB and APB protocols. The PSI supports 16 AHB master and 16 slave bus. The type of the slave bus can be the AHB bus or APB bus. Each bus supports 64 slave devices.

3.11.2 Block Diagram

The following figure shows the block diagram of PSI.

Figure 3-28 PSI Block Diagram



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.12 Message Box

3.12.1 Overview

The Message Box (MSGBOX) provides interrupt communication mechanism for on-chip processor.

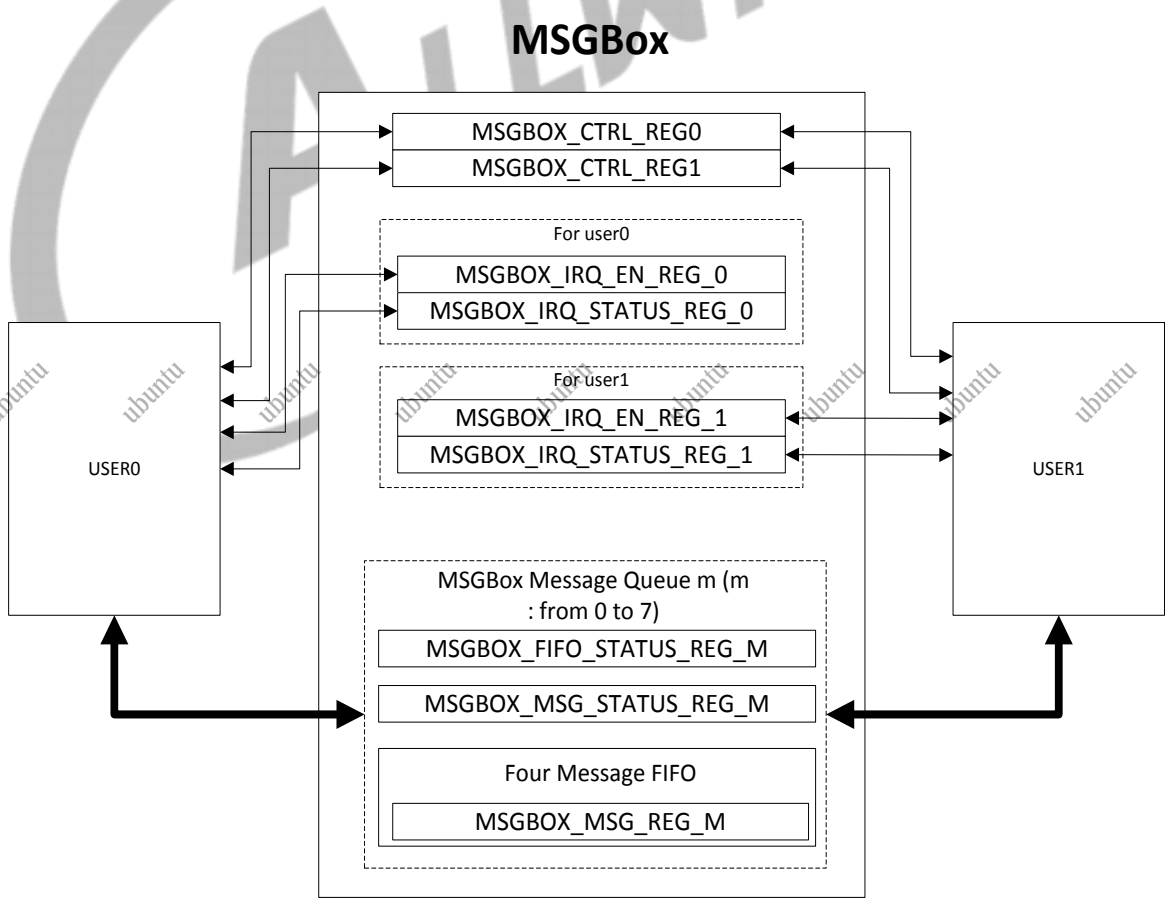
The MSGBOX has the following features:

- The communication parties transmit information through channels
- The FIFO depth of the channels is 8 x 32 bits
- The communication parties of MSGBOX0 are CPU and DSP0, the communication parties of MSGBOX1 are CPU and DSP1, and the communication parties of R_MSGBOX are DSP0 and DSP1
- Supports interrupts

3.12.2 Block Diagram

The following figure shows the block diagram of the message box.

Figure 3-29 Message Box Block Diagram



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Message Box supports a set of registers for a processor to establish an interconnection channel with the others. The processor determines message queue numbers for interconnection and the used queues to be transmitter or receiver for itself and the interconnected one. Every queue has one MSGBox FIFO Status Register for processor to check out queue FIFO full status and one MSGBox Message Status Register for processor to check out message numbers in queue FIFO. Otherwise, every queue has a corresponding IRQ status bit and a corresponding IRQ enable bit, which are used for requesting an interrupt.

3.12.3 Functional Descriptions

3.12.3.1 Clock and Reset

The MSGBOX is on AHB1. Before accessing the MSGBOX registers, you need to de-assert the MSGBOX reset signal and then open the MSGBOX gating signal.

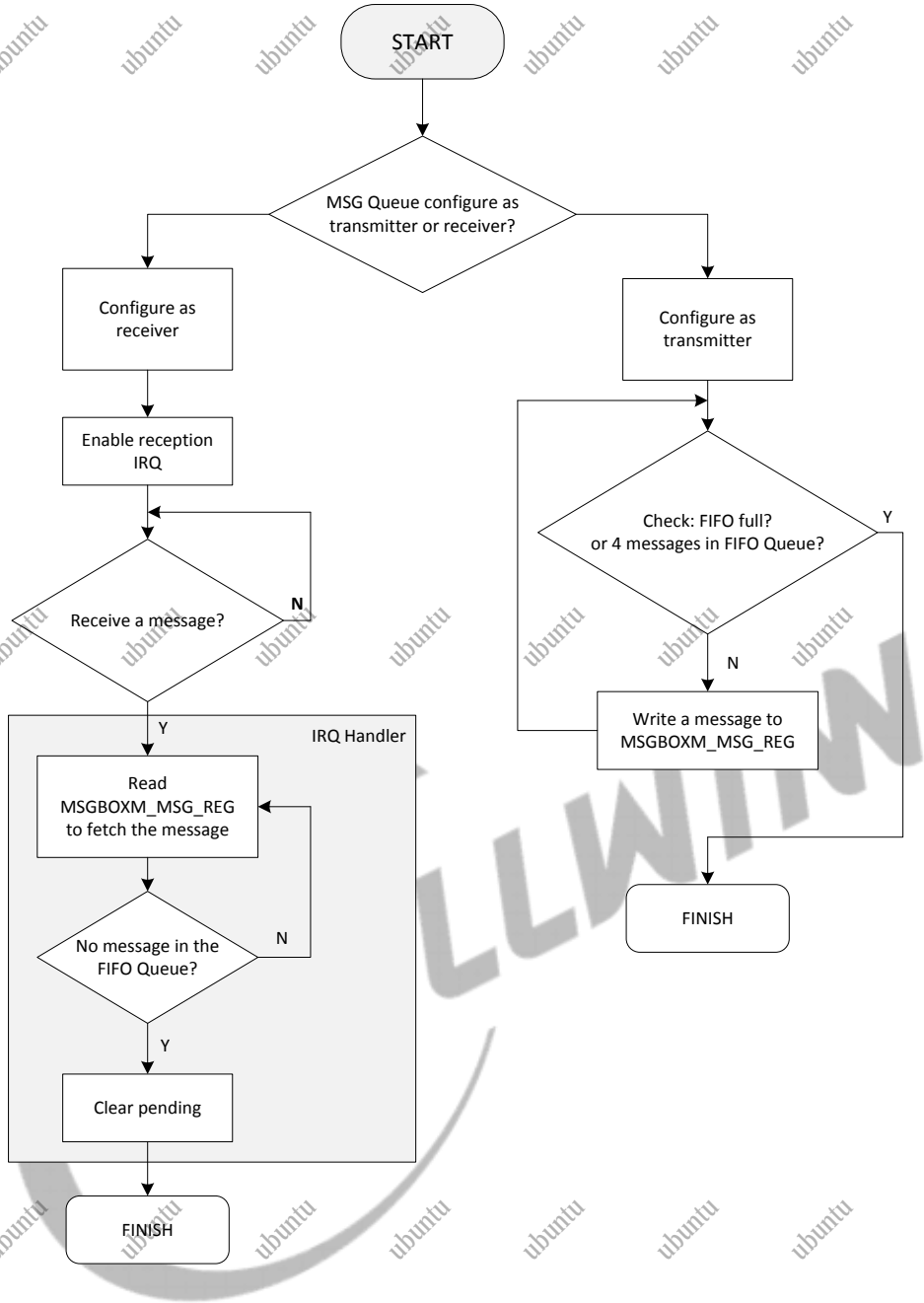
3.12.3.2 Typical Application

Two different CPUs can build communication by configuring the MSGBOX. The communication parties have 8 bidirectional channels. One CPU acts as the transmitter and the other acts as the receiver. During the communication process, the current status can be judged through the interrupt or FIFO status.

The following figure shows the chart flow of a typical application of the message box.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 3-30 Message Box Typical Application Chart



3.12.4 Programming Guidelines

3.12.4.1 Configuring the Transfer Mode

Either user of the channel can acts as the transmitter or receiver. The two users of the channels should be in different transfer modes: one in the transmitter mode and the other in receiver mode.

Follow the steps below to configure the transfer mode of Queue n:

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

For n = 0 to 3:

Step 1 Transmitter mode: Write 1 to the bit[8*n + 4] of **MSGBOX_CTRL_REG0**.

Step 2 Receiver mode: Write 1 to the bit[8*n] of **MSGBOX_CTRL_REG0**.

For n = 4 to 7:

Step 1 Transmitter mode: Write 1 to the bit[8*(m - 4) + 4] of **MSGBOX_CTRL_REG1**.

Step 2 Receiver mode: Write 1 to the bit[8*(m-4)] of **MSGBOX_CTRL_REG1**.

3.12.4.2 Checking the Transfer Status via the Interrupt Status

You can check the transfer status via the interrupt status. Follow the steps below:

Step 1 Configure the transfer mode for the two ends of the channel. For details, refer to section 3.12.4.1 "Configuring the Transfer Mode".

Step 2 Enable the interrupt for the channel: Configure the interrupt enable bits of transmitter/receiver through **MSGBOXU_IRQ_EN_REG**.

Step 3 Check the IRQ status of the corresponding queue through **MSGBOXU_IRQ_STATUS_REG**.

- If the FIFO is not full, the channel generates a transmission interrupt to remind the transmitter to transmit data. Write data to the FIFO in the interrupt handler, then clear the pending bit of the transmitter in **MSGBOXU_IRQ_STATUS_REG** and enable bit of the transmitter in **MSGBOXU_IRQ_EN_REG**.
- If there are new data added to the FIFO, the channel generates a reception interrupt to remind the receiver to receive data. Read data from the FIFO in interrupt handler, then clear the pending bit of the receiver in **MSGBOXU_IRQ_STATUS_REG** and enable bit of the receiver in **MSGBOXU_IRQ_EN_REG**.

3.12.4.3 Checking the Transfer Status via the FIFO Status

You can check whether the FIFO is full via **MSGBOXM_MSG_STATUS_REG**. Follow the steps below:

Step 1 Configure the transfer mode for the two ends of the channel. For details, refer to section 3.12.4.1 "Configuring the Transfer Mode".

Step 2 Check the FIFO status of the corresponding queue through **MSGBOXM_FIFO_STATUS_REG**.

- If the FIFO is not full, the transmitter fills the FIFO to 8*32 bits.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

- If the FIFO is full, the receiver reads the FIFO data. To ensure data integrity, it also reads MSGBOXM_MSG_STATUS_REG to acquire the FIFO data amount before and after it reads the FIFO data. If the difference between the two values is equal to the amount of the data that reads from the FIFO, it means no data is dropped.

3.12.4.4 Debugging

To use MSGBOX in debug mode, follow the steps below:

Step 1 Write 1 to the bit[0] of MSGBOX_DEBUG_REG.

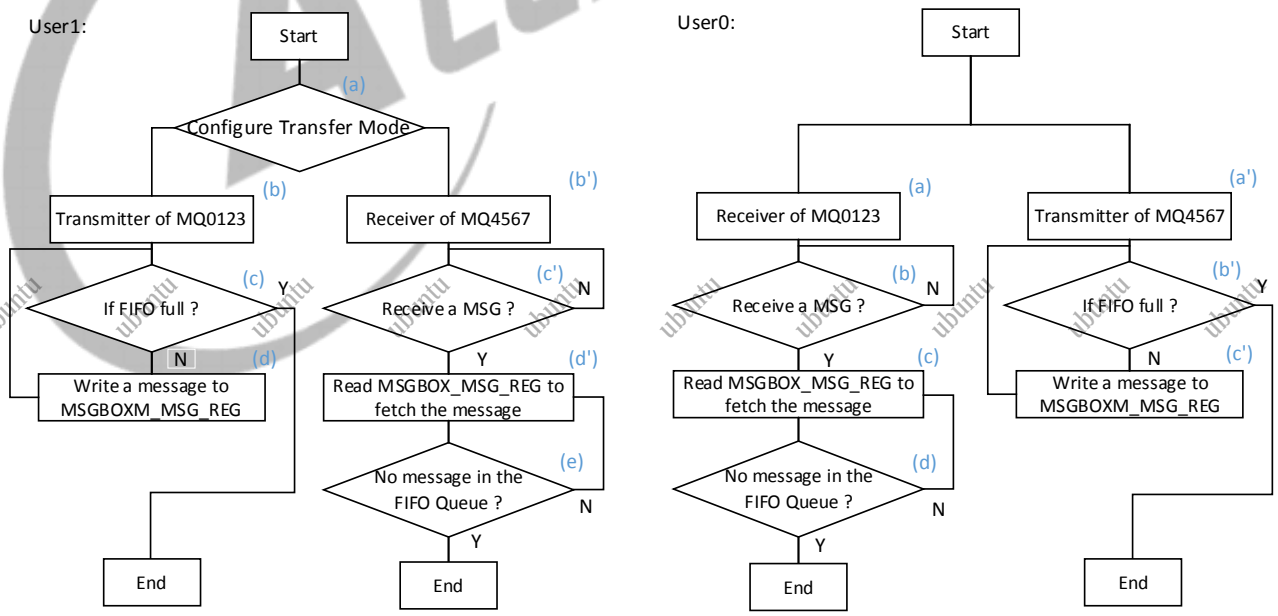
Step 2 Write 1 to the control bit of the corresponding channel.

3.12.4.5 Transmitting and Receiving Messages

Example: User1 as transmitter of MQ0123 (MQ: Message Queue) and as receiver of MQ4567, User0 as receiver of MQ0123 and as transmitter of MQ4567.

The working process of user1 and user0 is as follows.

Figure 3-31 User1 and User0 Working Process



User1:

(a) User1 acts as the transmitter of MQ0123 and receiver of MQ4567.

(b) Queue n (n = 0 to 3) transmitter mode: write 1 to the bit[8*n + 4] of MSGBOX_CTRL_REG0.

(b') Queue m (m = 4 to 7) receiver mode: write 1 to the bit[8*(m-4)] of MSGBOX_CTRL_REG1.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

(c) Check whether the FIFO is full by the status of MSGBOXM_FIFO_STATUS_REG. If the FIFO is full, the program ends; otherwise, go to step (d).

(c') Check whether the receiver receives a message by the status of MSGBOXM_MSG_STATUS_REG. If yes, go to step (d'); otherwise, continue to wait.

(d) Write a message to MSGBOXM_MSG_REG.

(d') Read MSGBOXM_MSG_REG to fetch the message.

(e) If there is no message in FIFO Queue, step up to (d'); otherwise, the program ends.

User0:

(a) User0 acts as the receiver of MQ0123.

(a') User0 acts as the transmitter of MQ4567.

(b) Check the receiver receives a message by the status of MSGBOXM_MSG_STATUS_REG. If yes, go to step (c); otherwise, continue to wait.

(b') Check whether the FIFO is full by the status of MSGBOXM_FIFO_STATUS_REG. If the FIFO is full, the program ends; otherwise, step up to (c').

(c) Read MSGBOXM_MSG_REG to fetch the message.

(c') Write a message to MSGBOXM_MSG_REG.

(d) If there is no message in FIFO Queue, go to step (c); otherwise, the program ends.

3.12.5 Register List

Module Name	Base Address
MSGBOX0	0x03003000
MSGBOX1	0x03008000
R_MSGBOX	0x07092000

Register Name	Offset	Description
MSGBOX_CTRL_REG0	0x0000	Message Queue Attribute Control Register 0
MSGBOX_CTRL_REG1	0x0004	Message Queue Attribute Control Register 1
MSGBOXU_IRQ_EN_REG	0x0040 + n*0x0020	IRQ Enable for User n (n = 0, 1)
MSGBOXU_IRQ_STATUS_REG	0x0050 + n*0x0020	IRQ Status for User n (n = 0, 1)
MSGBOXM_FIFO_STATUS_REG	0x0100 + n*0x0004	FIFO Status for Message Queue n (n = 0 to 7)

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
MSGBOXM_MSG_STATUS_REG	0x0140 + n*0x0004	Message Status for Message Queue n (n = 0 to 7)
MSGBOXM_MSG_REG	0x0180 + n*0x0004	Message Register for Message Queue n (n = 0 to 7)
MSGBOX_DEBUG_REG	0x01C0	MSGBOX Debug Register

3.12.6 Register Description

3.12.6.1 0x0000 MSGBox Control Register 0 (Default Value: 0x1010_1010)

Offset: 0x0000			Register Name: MSGBOX_CTRL_REG0
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28	R/W	0x1	TRANSMIT_MQ3 Indicates which user is the transmitter of Message Queue 3. 0: user0 1: user1
27:25	/	/	/
24	R/W	0x0	RECEPTION_MQ3 Indicates whether user0 or user1 is the receiver of Message Queue 3. 0: user0 1: user1
23:21	/	/	/
20	R/W	0x1	TRANSMIT_MQ2 Indicates which user is the transmitter of Message Queue 2. 0: user0 1: user1
19:17	/	/	/
16	R/W	0x0	RECEPTION_MQ2 Indicates which user is the receiver of Message Queue 2. 0: user0 1: user1
15:13	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: MSGBOX_CTRL_REG0
Bit	Read/Write	Default/Hex	Description
12	R/W	0x1	TRANSMIT_MQ1 Indicates which user is the transmitter of Message Queue 1. 0: user0 1: user1
11:9	/	/	/
8	R/W	0x0	RECEPTION_MQ1 Indicates which user is the receiver of Message Queue 1. 0: user0 1: user1
7:5	/	/	/
4	R/W	0x1	TRANSMIT_MQ0 Indicates which user is the transmitter of Message Queue 0. 0: user0 1: user1
3:1	/	/	/
0	R/W	0x0	RECEPTION_MQ0 Indicates which user is the receiver of Message Queue 0. 0: user0 1: user1

3.12.6.2 0x0004 MSGBox Control Register 1 (Default Value: 0x1010_1010)

Offset: 0x0004			Register Name: MSGBOX_CTRL_REG1
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28	R/W	0x1	TRANSMIT_MQ7 Indicates which user is the transmitter of Message Queue 7. 0: user0 1: user1
27:25	/	/	/
24	R/W	0x0	RECEPTION_MQ7 Indicates which user is the receiver of Message Queue 7. 0: user0 1: user1

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0004			Register Name: MSGBOX_CTRL_REG1
Bit	Read/Write	Default/Hex	Description
23:21	/	/	/
20	R/W	0x1	TRANSMIT_MQ6 Indicates which user is the transmitter of Message Queue 6. 0: user0 1: user1
19:17	/	/	/
16	R/W	0x0	RECEPTION_MQ6 Indicates which user is the receiver of Message Queue 6. 0: user0 1: user1
15:13	/	/	/
12	R/W	0x1	TRANSMIT_MQ5 Indicates which user is the transmitter of Message Queue 5. 0: user0 1: user1
11:9	/	/	/
8	R/W	0x0	RECEPTION_MQ5 Indicates which user is the receiver of Message Queue 5. 0: user0 1: user1
7:5	/	/	/
4	R/W	0x1	TRANSMIT_MQ4 Indicates which user is the transmitter of Message Queue 4. 0: user0 1: user1
3:1	/	/	/
0	R/W	0x0	RECEPTION_MQ4 Indicates which user is the receiver of Message Queue 4. 0: user0 1: user1

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.12.6.3 0x0040 + N*0x20 MSGBox IRQ Enable Register u (Default Value: 0x0000_0000)

Offset: 0x0040 + N*0x0020 (N = 0, 1)			Register Name: MSGBOXU_IRQ_EN_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R/W	0x0	TRANSMIT_MQ7_IRQ_EN 0: Disabled 1: Enabled (It will notify the user by interrupt when Message Queue 7 is not full.)
14	R/W	0x0	RECEPTION_MQ7_IRQ_EN 0: Disabled 1: Enabled (It will notify the user by interrupt when Message Queue 7 has received a new message.)
13	R/W	0x0	TRANSMIT_MQ6_IRQ_EN 0: Disabled 1: Enabled (It will notify the user by interrupt when Message Queue 6 is not full.)
12	R/W	0x0	RECEPTION_MQ6_IRQ_EN 0: Disabled 1: Enabled (It will notify the user by interrupt when Message Queue 6 has received a new message.)
11	R/W	0x0	TRANSMIT_MQ5_IRQ_EN 0: Disabled 1: Enabled (It will notify the user by interrupt when Message Queue 5 is not full.)
10	R/W	0x0	RECEPTION_MQ5_IRQ_EN 0: Disabled 1: Enabled (It will notify the user by interrupt when Message Queue 5 has received a new message.)
9	R/W	0x0	TRANSMIT_MQ4_IRQ_EN 0: Disabled 1: Enabled (It will notify the user by interrupt when Message Queue 4 is not full.)
8	R/W	0x0	RECEPTION_MQ4_IRQ_EN 0: Disabled 1: Enabled (It will notify the user by interrupt when Message Queue 4 has received a new message.)

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0040 + N*0x0020 (N = 0, 1)			Register Name: MSGBOXU_IRQ_EN_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
7	R/W	0x0	TRANSMIT_MQ3_IRQ_EN 0: Disabled 1: Enabled (It will notify the user by interrupt when Message Queue 3 is not full.)
6	R/W	0x0	RECEPTION_MQ3_IRQ_EN 0: Disabled 1: Enabled (It will notify the user by interrupt when Message Queue 3 has received a new message.)
5	R/W	0x0	TRANSMIT_MQ2_IRQ_EN 0: Disabled 1: Enabled (It will notify the user by interrupt when Message Queue 2 is not full.)
4	R/W	0x0	RECEPTION_MQ2_IRQ_EN 0: Disabled 1: Enabled (It will notify the user by interrupt when Message Queue 2 has received a new message.)
3	R/W	0x0	TRANSMIT_MQ1_IRQ_EN 0: Disabled 1: Enabled (It will notify the user by interrupt when Message Queue 1 is not full.)
2	R/W	0x0	RECEPTION_MQ1_IRQ_EN 0: Disabled 1: Enabled (It will notify the user by interrupt when Message Queue 1 has received a new message.)
1	R/W	0x0	TRANSMIT_MQ0_IRQ_EN 0: Disabled 1: Enabled (It will notify the user by interrupt when Message Queue 0 is not full.)
0	R/W	0x0	RECEPTION_MQ0_IRQ_EN 0: Disabled 1: Enabled (It will notify the user by interrupt when Message Queue 0 has received a new message.)

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.12.6.4 0x0050 + N*0x0020 MSGBox IRQ Status Register u (Default Value: 0x0000_AAAA)

Offset: 0x0050 + N*0x0020 (N = 0, 1)			Register Name: MSGBOXU_IRQ_STATUS_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R/W1C	0x1	TRANSMIT_MQ7_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for the transmitter when Message Queue 7 is not full. Setting 1 to this bit will clear the pending status.
14	R/W1C	0x0	RECEPTION_MQ7_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for the receiver when Message Queue 7 has received a new message. Setting 1 to this bit will clear the pending status.
13	R/W1C	0x1	TRANSMIT_MQ6_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for the transmitter when Message Queue 6 is not full. Setting 1 to this bit will clear the pending status.
12	R/W1C	0x0	RECEPTION_MQ6_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for the receiver when Message Queue 6 has received a new message. Setting 1 to this bit will clear the pending status.
11	R/W1C	0x1	TRANSMIT_MQ5_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for the transmitter when Message Queue 5 is not full. Setting 1 to this bit will clear the pending status.
10	R/W1C	0x0	RECEPTION_MQ5_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for the receiver when Message Queue 5 has received a new message. Setting 1 to this bit will clear the pending status.
9	R/W1C	0x1	TRANSMIT_MQ4_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for the transmitter when Message Queue 4 is not full. Setting 1 to this bit will clear the pending status.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0050 + N*0x0020 (N = 0, 1)			Register Name: MSGBOXU_IRQ_STATUS_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
8	R/W1C	0x0	RECEPTION_MQ4_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for the receiver when Message Queue 4 has received a new message. Setting 1 to this bit will clear the pending status.
7	R/W1C	0x1	TRANSMIT_MQ3_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for the transmitter when Message Queue 3 is not full. Setting 1 to this bit will clear the pending status.
6	R/W1C	0x0	RECEPTION_MQ3_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for the receiver when Message Queue 3 has received a new message. Setting 1 to this bit will clear the pending status.
5	R/W1C	0x1	TRANSMIT_MQ2_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for the transmitter when Message Queue 2 is not full. Setting 1 to this bit will clear the pending status.
4	R/W1C	0x0	RECEPTION_MQ2_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for the receiver when Message Queue 2 has received a new message. Setting 1 to this bit will clear the pending status.
3	R/W1C	0x1	TRANSMIT_MQ1_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for the transmitter when Message Queue 1 is not full. Setting 1 to this bit will clear the pending status.
2	R/W1C	0x0	RECEPTION_MQ1_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for the receiver when Message Queue 1 has received a new message. Setting 1 to this bit will clear the pending status.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0050 + N*0x0020 (N = 0, 1)			Register Name: MSGBOXU_IRQ_STATUS_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
1	R/W1C	0x1	TRANSMIT_MQ0_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for the transmitter when Message Queue 0 is not full. Setting 1 to this bit will clear the pending status.
0	R/W1C	0x0	RECEPTION_MQ0_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for the receiver when Message Queue 0 has received a new message. Setting 1 to this bit will clear the pending status.

3.12.6.5 0x0100 + N*0x0004 MSGBox FIFO Status Register m (Default Value: 0x0000_0000)

Offset: 0x0100 + N*0x0004 (N = 0 to 7)			Register Name: MSGBOXM_FIFO_STATUS_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R	0x0	FIFO_FULL_FLAG 0: The Message FIFO queue is not full (still has space available) 1: The Message FIFO queue is full. This FIFO status register has the status related to the message queue.

3.12.6.6 0x0140 + N*0x0004 MSGBox Message Status Register m (Default Value: 0x0000_0000)

Offset: 0x0140 + N*0x0004 (N = 0 to 7)			Register Name: MSGBOXM_MSG_STATUS_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2:0	R	0x0	MSG_NUM Number of unread messages in the message queue. Every message queue can store up to eight messages. 000: There is no message in the message FIFO queue. 001: There is 1 message in the message FIFO queue. 010: There are 2 messages in the message FIFO queue. 011: There are 3 messages in the message FIFO queue. 100: There are 4 messages in the message FIFO queue. 101 to 111: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.12.6.7 0x0180 + N*0x0004 MSGBox Message Queue Register m (Default Value: 0x0000_0000)

Offset: 0x0180 + N*0x0004 (N = 0 to 7)			Register Name: MSGBOXM_MSG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	The message register stores the next to be read message in the FIFO queue. The message will be removed from the FIFO queue after being read.

3.12.6.8 0x01C0 MSGBox Debug Register (Default Value: 0x0000_0000)

Offset: 0x01C0			Register Name: MSGBOX_DEBUG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:8	R/W	0x0	FIFO_CTRL MQ[7:0] Control. In the debug mode, the corresponding FIFO channel will be disabled and only one register space is valid for a message exchange. 0: Normal Mode 1: Disable the corresponding FIFO (Clear FIFO)
7:1	/	/	/
0	R/W	0x0	DEBUG_MODE In the debug mode, each user can transmit messages to itself through each message queue. 0: Normal Mode 1: Debug Mode

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.13 Spinlock

3.13.1 Overview

The spinlock provides hardware synchronization mechanism in multi-core systems. With the lock operation, the spinlock prevents multiple processors from handling the sharing data simultaneously and thus ensure the coherence of data.

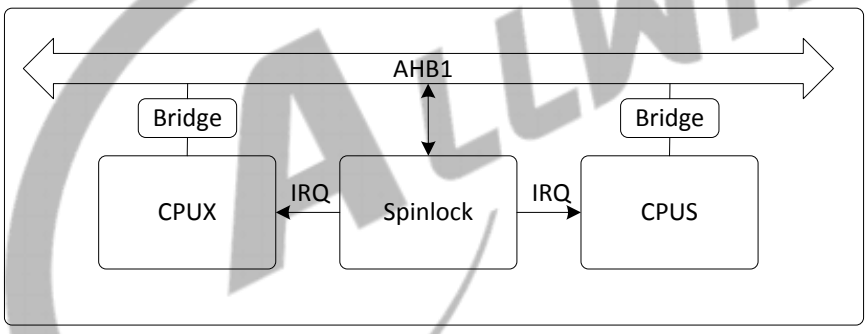
The spinlock has the following features:

- Supports 32 lock units
- Two kinds of lock status: locked and unlocked
- Lock time of the processor is predictable (less than 200 cycles)

3.13.2 Block Diagram

The following figure shows the block diagram of the spinlock.

Figure 3-32 Spinlock Block Diagram



3.13.3 Functional Descriptions

3.13.3.1 Clock and Reset

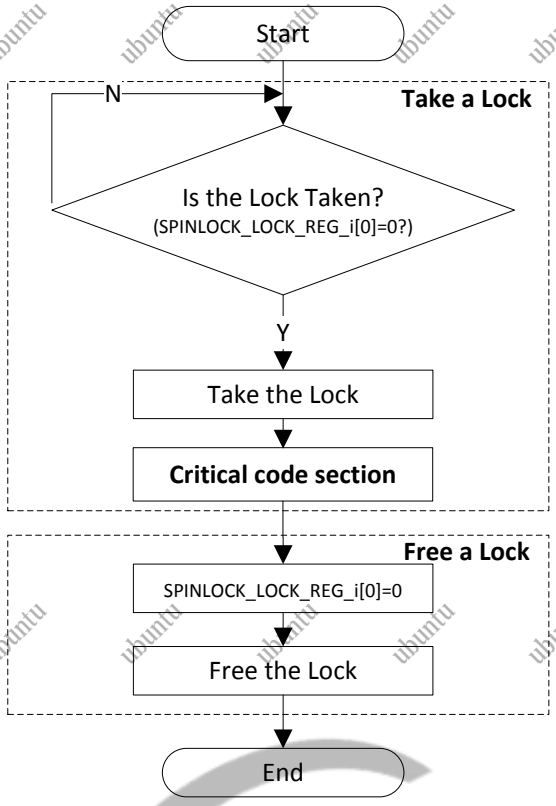
The spinlock is on AHB1. Before accessing the spinlock registers, you need to de-assert the reset signal and then open the corresponding gating signal on AHB1.

3.13.3.2 Typical Application

The following figure shows a typical application of the spinlock. A processor locks spinlock0 before executing specific codes, and then unlocks the codes. After the lock is freed, other processors can read or write the data.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

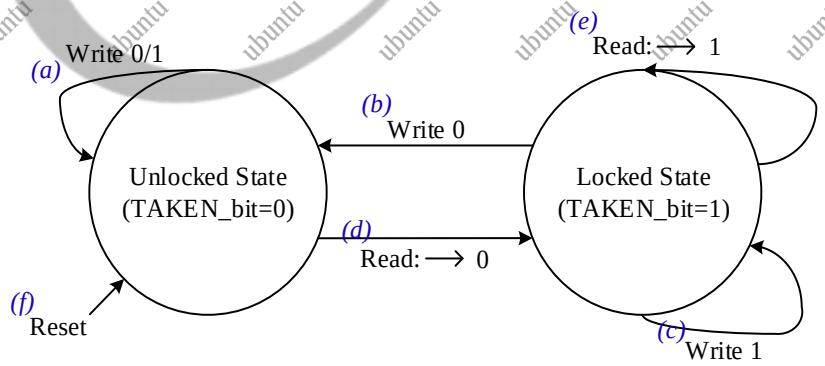
Figure 3-33 Spinlock Typical Application Diagram



3.13.3.3 Spinlock State Machine

When a processor uses spinlock, it needs to acquire the spinlock status through **SPINLOCK_STATUS_REG**. The following figure shows the spinlock state machine.

Figure 3-34 Spinlock State Machine



a) When spinlock is in Unlocked state, writing 0/1 has no effect;
b) When spinlock is in Locked state, writing 0 can convert the corresponding spinlock to the Unlocked state;
c) When spinlock is in Locked state, writing 1 has no effect;
d) When spinlock is in Unlocked state, reading the bit can return 0 (it indicates spinlock enters into the Locked state);

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

e) When spinlock is in Locked state, reading the bit can return 1 (it indicates spinlock is in the Locked state);
f) After reset, spinlock is in Unlock state by default.

3.13.3.4 Lock State of Spinlock

Checking out SPINLOCK_STATUS_REG is necessary when a processor would like to take a spinlock. This register stores the status of all 32 lock registers: TAKEN or NOT TAKEN (free).

To request to take a spinlock, a processor needs to do a read-access to the corresponding lock register. If the lock register returns 0, the processor takes this spinlock. If the lock register returns 1, the processor needs to retry.

Writing 0 to a lock register frees the corresponding spinlock. If the lock register is not taken, the write-access has no effect. For a taken spinlock, every processor has the privilege to free this spinlock. But it is suggested that the processor which has taken the spinlock free it for strictness.

3.13.4 Programming Guidelines

3.13.4.1 Switching the Status

Follow the steps below to switch the lock status of a spinlock.

Step 1 When the read value from SPINLOCKN_LOCK_REG is 0, the spinlock comes into the locked status.

Step 2 Execute the application codes, and the status of SPINLOCK_STATUS_REG is 1.

Step 3 Write 0 to SPINLOCKN_LOCK_REG, the spinlock converts into the unlocked status, and the corresponding spinlock is released.

3.13.4.2 Processing the Interrupt

The spinlock generates an interrupt when a lock is freed (the lock status converts from locked to unlocked).

Follow the steps below to process the interrupt:

Step 1 Configure the interrupt enable bit of the corresponding spinlock in SPINLOCK_IRQ_EN_REG to enable the interrupt.

Step 2 The spinlock generates an interrupt when its status converts from locked to unlocked, and the corresponding bit of the SPINLOCK_IRQ_STA_REG turns to 1.

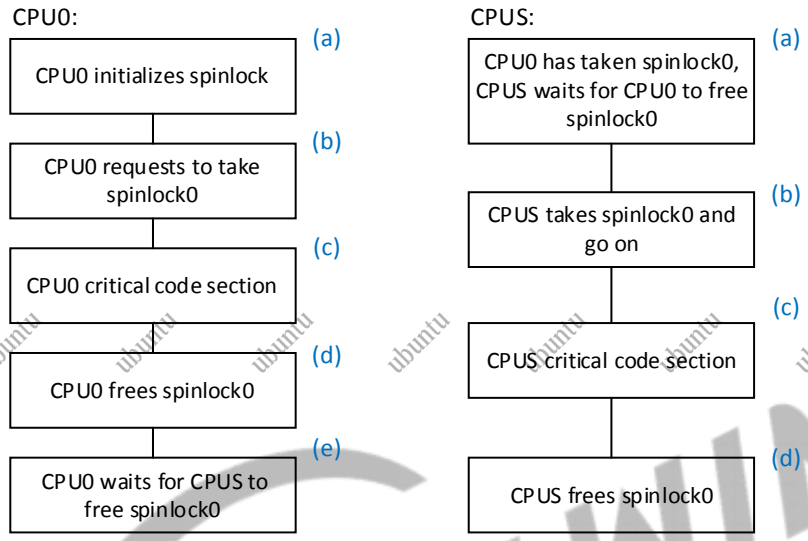
Step 3 Execute the interrupt handle function and clear the pending bit.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.13.4.3 Taking/Freeing Spinlock

Take Spinlock0 as an example, the CPU0 takes the spinlock0 firstly in the instance. To take/free spinlock0, the CPU0 and CPUS perform the following steps:

Figure 3-35 CPU0 and CPUS Taking/Freeing Spinlock0 Process



CPU0:

(a) The CPU0 initializes Spinlock.

(b) Check lock register0 (SPINLOCK_STATUS_REG0) status. If it is taken, check until CPU0 frees spinlock0 and then request to take spinlock0. Otherwise, retry until the lock register0 is taken.

(c) Execute CPU0 critical code.

(d) After executing CPU0 critical code, the CPU0 frees spinlock0.

(e) The CPU0 waits for CPUS to free spinlock0.

CPUS:

(a) If the CPU0 has taken spinlock0, the CPUS waits for CPU0 to free spinlock0.

(b) The CPUS requests to take spinlock0. If it fails, retry until the lock register0 is taken.

(c) Execute CPUS critical code.

(d) After executing CPUS critical code, the CPUS frees spinlock0.

The following codes are for reference.

----------**CPU0 of Cluster0**----------

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 1 CPU0 initializes Spinlock

```
put_wvalue(SPINLOCK_BGR_REG,0x00010000);
put_wvalue(SPINLOCK_BGR_REG,0x00010001);
```

Step 2 CPU0 requests to take spinlock0

```
rdata=readl(SPINLOCK_STATUS_REG0);                  //Check lock register0 status
if(rdata != 0)    writel(0, SPINLOCK_LOCK_REG0);    //If it is taken, check till CPU0 frees spinlock0
rdata=readl(SPINLOCK_LOCK_REG0);                    //Request to take spinlock0
if(rdata != 0)    rdata=readl(SPINLOCK_LOCK_REG0);  //If it fails, retry till lock register0 is taken
```

---------- CPU0 critical code section ----------

Step 3 CPU0 free spinlock0

```
writel(0, SPINLOCK_LOCK_REG0);                      //CPU0 frees spinlock0
```

Step 4 CPU0 waits for CPUS' freeing spinlock0

```
writel(readl(SPINLOCK_STATUS_REG0) == 1);           //CPU0 waits for CPUS' freeing spinlock0
```

----------**CPUS**----------

Step 1 CPU0 has taken spinlock0, CPUS waits for CPU0' freeing spinlock0

```
while(readl(SPINLOCK_STATUS_REG0) == 1);            //CPUS waits for CPU0' freeing spinlock0
```

Step 2 CPUS takes spinlock0 and go on

```
rdata=readl(SPINLOCK_LOCK_REG0);                    //Request to take spinlock0
if(rdata != 0)    rdata=readl(SPINLOCK_LOCK_REG0);  //If it fails, retry till lock register0 is taken
```

---------- CPUS critical code section ----------

Step 3 CPUS frees spinlock0

```
writel(0, SPINLOCK_LOCK_REG0);                      //CPUS frees spinlock0
```

3.13.5 Register List

Module Name	Base Address
Spinlock	0x03004000
R_Spinlock	0x07093000

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
SPINLOCK_SYSTATUS_REG	0x0000	Spinlock System Status Register
SPINLOCK_STATUS_REG	0x0010	Spinlock Status Register
SPINLOCK_IRQ_EN_REG	0x0020	Spinlock Interrupt Enable Register
SPINLOCK_IRQ_STA_REG	0x0040	Spinlock Interrupt Status Register
SPINLOCK_LOCKID0_REG	0x0080	Spinlock Lockid0 Register
SPINLOCK_LOCKID1_REG	0x0084	Spinlock Lockid1 Register
SPINLOCK_LOCKID2_REG	0x0088	Spinlock Lockid2 Register
SPINLOCK_LOCKID3_REG	0x008C	Spinlock Lockid3 Register
SPINLOCK_LOCKID4_REG	0x0090	Spinlock Lockid4 Register
SPINLOCK_LOCK_REGN	0x0100 + N*0x0004	Spinlock Register N (N = 0 to 31)

3.13.6 Register Description

3.13.6.1 0x0000 Spinlock System Status Register (Default Value: 0x1000_0000)

Offset: 0x0000			Register Name: SPINLOCK_SYSTATUS_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/
29:28	R	0x1	LOCKS_NUM Number of lock registers implemented. 00: This instance has 32 lock registers. 01: This instance has 64 lock registers. 10: This instance has 128 lock registers. 11: This instance has 256 lock registers.
27:9	/	/	/
8	R	0x0	IU0 In-Use flag0, covering lock register0-31 0: All lock registers 0-31 are in the NotTaken state. 1: At least one of the lock register 0-31 is in the Taken state.
7:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.13.6.2 0x0010 Spinlock Status Register (Default Value: 0x0000_0000)

Offset: 0x0010			Register Name: SPINLOCK_STATUS_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
[i] (i = 0 to 31)	R	0x0	LOCK_REG_STATUS SpinLock[i] status (i = 0 to 31) 0: The Spinlock is free. 1: The Spinlock is taken.

3.13.6.3 0x0020 Spinlock Interrupt Enable Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: SPINLOCK_IRQ_EN_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	LOCK_IRQ_EN SpinLock[i] interrupt enable 0: Disable 1: Enable

3.13.6.4 0x0040 Spinlock Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0040			Register Name: SPINLOCK_IRQ_STA_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	LOCK_IRQ_STATUS SpinLock[i] interrupt status 0: No effect 1: Pending Writing 1 clears this bit.

3.13.6.5 0x0080 Spinlock Lockid0 Register (Default Value: 0x7777_7777)

Offset: 0x0080			Register Name: SPINLOCK_LOCKIN0_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x77777777	LOCKID0

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.13.6.6 0x0084 Spinlock Lockid1 Register (Default Value: 0x7777_7777)

Offset: 0x0084			Register Name: SPINLOCK_LOCKIN1_REG
Bit	Read/Write	Default/Hex	Description
31:0	R	0x77777777	LOCKID1

3.13.6.7 0x0088 Spinlock Lockid2 Register (Default Value: 0x7777_7777)

Offset: 0x0088			Register Name: SPINLOCK_LOCKIN2_REG
Bit	Read/Write	Default/Hex	Description
31:0	R	0x77777777	LOCKID2

3.13.6.8 0x008C Spinlock Lockid3 Register (Default Value: 0x7777_7777)

Offset: 0x008C			Register Name: SPINLOCK_LOCKIN3_REG
Bit	Read/Write	Default/Hex	Description
31:0	R	0x77777777	LOCKID3

3.13.6.9 0x0090 Spinlock Lockid4 Register (Default Value: 0x7777_7777)

Offset: 0x0090			Register Name: SPINLOCK_LOCKIN4_REG
Bit	Read/Write	Default/Hex	Description
31:0	R	0x77777777	LOCKID4

3.13.6.10 0x0100 + N*0x04 Spinlock Register N (N = 0 to 31) (Default Value: 0x0000_0000)

Offset: 0x0100 + N*0x0004 (N = 0 to 31)			Register Name: SPINLOCKN_LOCK_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0100 + N*0x0004 (N = 0 to 31)			Register Name: SPINLOCKN_LOCK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
0	R/W	0x0	TAKEN Lock State Read 0x0: The lock was previously Not Taken (free). The requester is granted the lock. Write 0x0: Set the lock to Not Taken (free). Read 0x1: The lock was previously Taken. The requester is not granted the lock and must retry. Write 0x1: No update to the lock value.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.14 RTC

3.14.1 Overview

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

The RTC has the following features:

- Provides a 16-bit counter for counting day, 5-bit counter for counting hour, 6-bit counter for counting minute, 6-bit counter for counting second
- External connect a 32.768 kHz low-frequency oscillator for count clock
- Timer frequency is 1 kHz
- Configurable initial value by software anytime
- Supports timing alarm, and generates interrupt and wakeup the external devices
- Supports fanout function of internal 32K clock
- 8 general purpose registers for storing power-off information

CAUTION

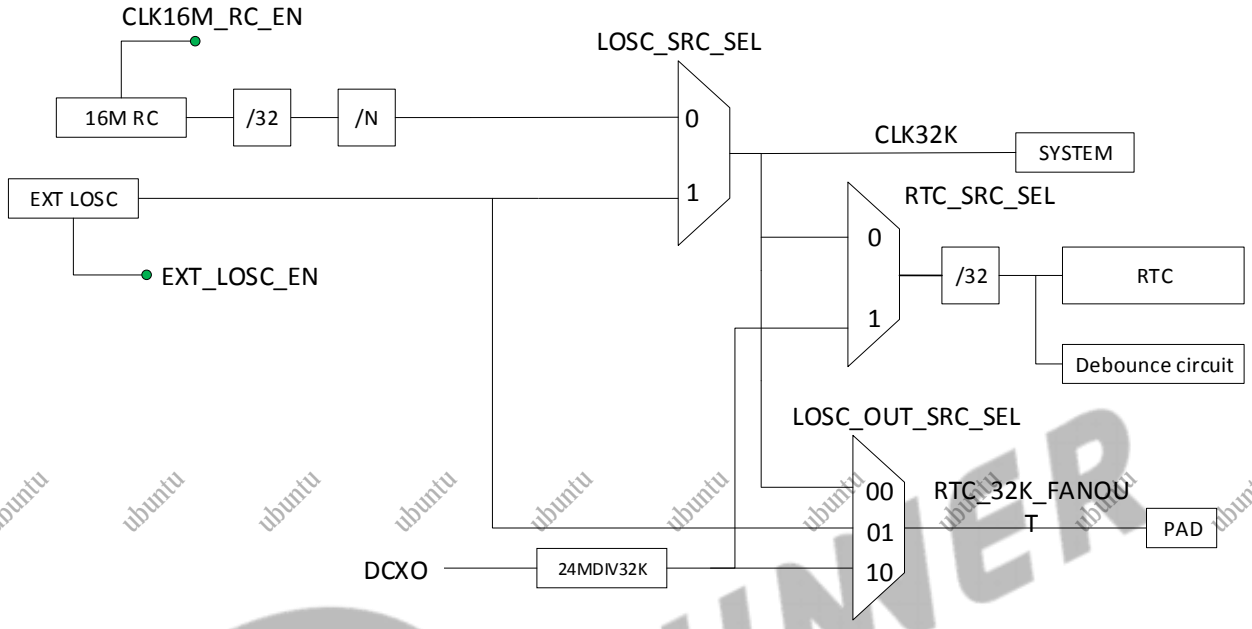
The register configuration of RTC is AHB bus, it only can support word operation, not byte operation and half-word operation.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.14.2 RTC Clock Tree

The following figure shows the clock tree of the RTC.

Figure 3-36 RTC Clock Tree



3.14.3 Functional Descriptions

3.14.3.1 External Signals

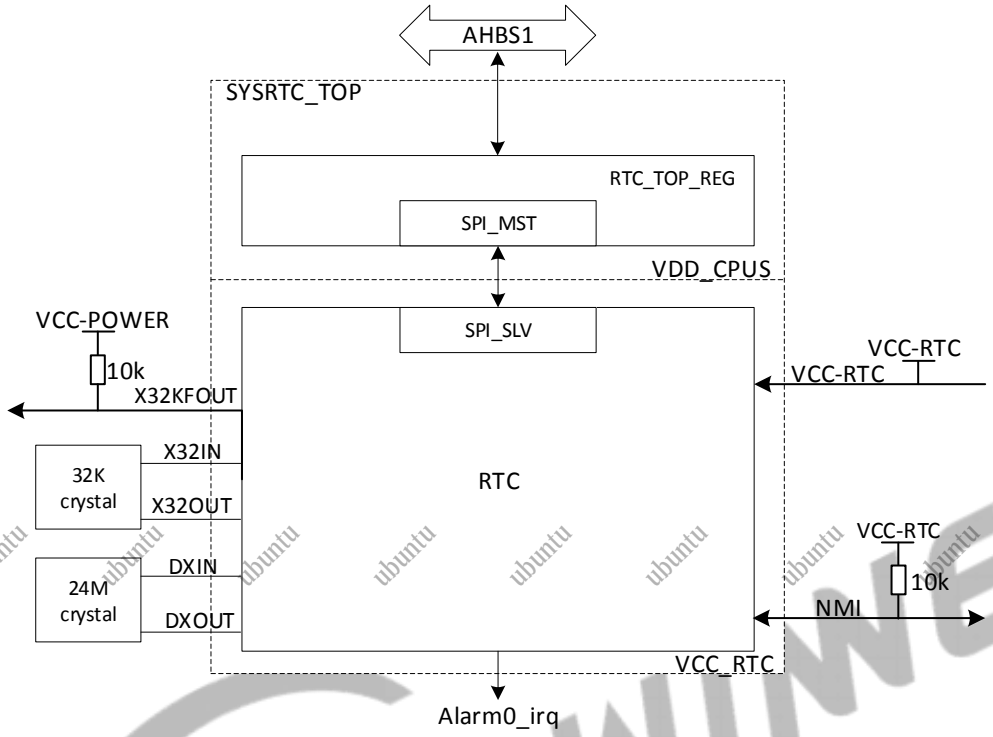
Table 3-15 RTC External Signals

Signal	Description
X32KIN	32.768 kHz oscillator input
X32KOUT	32.768 kHz oscillator output
X32KFOUT	32.768 kHz clock fanout Provides low frequency clock for external devices
NMI	Non-maskable Interrupt Alarm wakeup generates low level into NMI
RTC_VIO	RTC low voltage, generated via internal LDO
VCC_RTC	RTC high voltage, generated via external power

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.14.3.2 Typical Application

Figure 3-37 RTC Application Diagram



The external low-frequency oscillator must be 32.768 kHz. If the external peripheral needs low frequency crystal, which can be provided by X32KFOUT.

The NMI and alarm0 in common generate low level signal.

3.14.3.3 Clock Sources

The RTC has 2 clock sources: internal RC, external low frequency crystal.

The RTC selects the internal RC by default, when the system starts, the RTC can select by software the external low frequency crystal to provide much accuracy clock. The clock accurate of the RTC is related to the accurate of the external low frequency crystal. Usually select 32.768 kHz crystal with ±20 ppm frequency tolerance.

When using internal RC, the clock of RTC can be changed by changing division ratio. When using external clock, the clock cannot be changed.

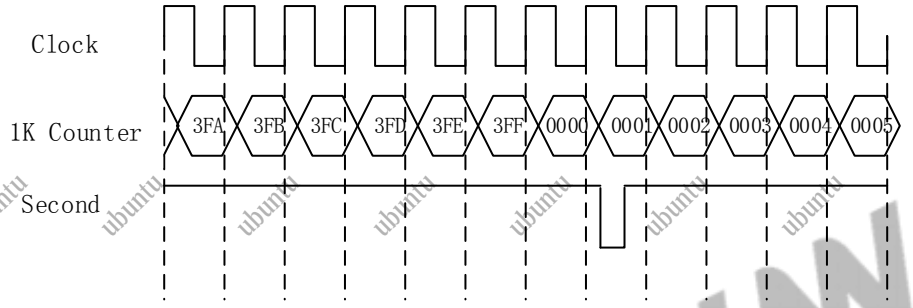
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.14.3.4 Clock and Reset

The RTC module has the independent reset signal, the signal follows VCC_RTC. When VCC_RTC powers on, the reset signal resets the RTC module; after VCC_RTC reaches stable, the reset signal always holds high level. Watchdog Reset cannot reset RTC.

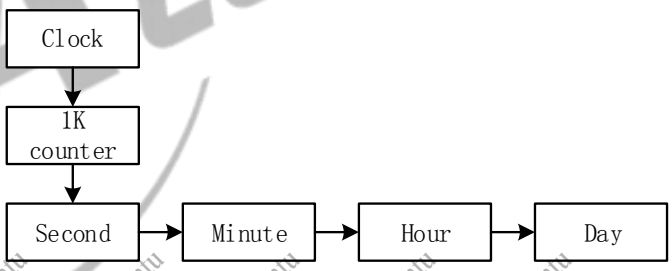
3.14.3.5 Real Time Clock

Figure 3-38 RTC Counter



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

Figure 3-39 RTC 1KHz Counter Step Structure



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

Table 3-16 RTC Counter Changing Range

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

CAUTION

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

3.14.3.6 Alarm 0

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

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

3.14.3.7 Power-off Storage

The RTC provides eight 32-bit general purpose register to store power-off information.

Because VCC-RTC always holds non-power-off state after VCC-RTC cold starts, when the system is in shutdown or standby scene, CPU can judge software process by the storing information.

3.14.3.8 RTC_VIO

The RTC module has a LDO, the input source of the LDO is VCC_RTC, the output of the LDO is RTC_VIO, the value of RTC_VIO is adjustable, the RTC_VIO is mainly used for internal digital logic.

3.14.3.9 RC Calibration Usage Scenario

- Power-on: Select non-accurate 32K divided by internal RC.
- Normal scenario: Select external accurate 32K, or external calibrated 32K.
- Standby or power-off scenario: Select external accurate 32K, or external calibrated 32K.

3.14.4 Programming Guidelines

3.14.4.1 RTC Clock Control

Step 1 Select clock source: Select clock source by the bit0 of **LOSC_CTRL_REG**, the clock source is the internal RC oscillator by default. When the system starts, the clock source can be switched to the external 32K oscillator by software.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 2 Auto switch: After enabled the bit[14] of **LOSC_CTRL_REG**, the RTC automatically switches clock source to the internal oscillator when the external crystal could not output waveform, the switch status can query by the bit[1] of **LOSC_AUTO_SWT_STA_REG**.

Step 3 After auto switch is valid, the clock source status bit cannot be changed, because the two functions are independent.

3.14.4.2 RTC Calendar

Step 1 Write time initial value: Write the current time to **RTC_DAY_REG** and **RTC_HH_MM_SS_REG**.

Step 2 After configured time, read the bit[8:7] of **LOSC_CTRL_REG** to ensure that configuration is completed.

Step 3 After update time, the RTC restarts to count again. The software can read the current time anytime.

NOTE

The RTC can only provide day counter, so the current day counter need be converted to year, month, day and week by software.

After configured time at each time, you need ensure the bit[8:7] of LOSC_CTRL_REG is 0 before the next setting is performed.

3.14.4.3 Alarm0

Step 1 Enable alram0 interrupt by writing **ALARM0_IRQ_EN**.

Step 2 Set the counter comparator, write the count-down day, hour, minute, second number to **ALARM0_DAY_SET_REG** and **ALARM0_HH-MM-SS_SET_REG**.

Step 3 Enable alarm0 function by writing **ALARM0_ENABLE_REG**, then the software can query alarm count value in real time by **ALARM0_DAY_SET_REG** and **ALARM0_HH-MM-SS_SET_REG.** When the setting time reaches, **ALARM0_IRQ_STA_REG** is set to 1 to generate interrupt.

Step 4 After enter the interrupt process, write **ALARM0_IRQ_STA_REG** to clear the interrupt pending, and execute the interrupt process.

Step 5 Resume the interrupt and continue to execute the interrupted process.

Step 6 Power-off wakeup is generated via SoC hardware and PMIC, the software only need set pending condition of alarm0, and set 1 to **ALARM_CONFIG_REG**.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.14.4.4 Fanout

Set the bit0 of 32K_FOUT_CTRL_GATING_REG to 1, and ensure external pull-up resistor, voltage, clock source are normal, then 32.768 kHz square wave can be output.

3.14.5 Register List

Module Name	Base Address
RTC	0x07090000

Register Name	Offset	Description
LOSC_CTRL_REG	0x0000	Low Oscillator Control Register
LOSC_AUTO_SWT_STA_REG	0x0004	LOSC Auto Switch Status Register
INTOSC_CLK_PRESCAL_REG	0x0008	Internal OSC Clock Pre-scalar Register
RTC_DAY_REG	0x0010	RTC Year-Month-Day Register
RTC_HH_MM_SS_REG	0x0014	RTC Hour-Minute-Second Register
ALARM0_COUNTER_REG	0x0020	Alarm 0 Counter Register
ALARM0_CUR_VLU_REG	0x0024	Alarm 0 Counter Current Value Register
ALARM0_ENABLE_REG	0x0028	Alarm 0 Enable Register
ALARM0_IRQ_EN	0x002C	Alarm 0 IRQ Enable Register
ALARM0_IRQ_STA_REG	0x0030	Alarm 0 IRQ Status Register
ALARM_CONFIG_REG	0x0050	Alarm Configuration Register
32K_FOUT_CTRL_GATING_REG	0x0060	32K Fanout Control Gating Register
GP_DATA_REG	0x0100 + N*0x04	General Purpose Register (N=0 to 7)
FBOOT_INFO_REG0	0x0120	Fast Boot Information Register0
FBOOT_INFO_REG1	0x0124	Fast Boot Information Register1
DCXO_CTRL_REG	0x0160	DCXO Control Register
RTC_VIO_REG	0x0190	RTC_VIO Regulate Register
IC_CHARA_REG	0x01F0	IC Characteristic Register
VDD_OFF_GATING_CTRL_REG	0x01F4	VDD Off Gating Control Register
EFUSE_HV_PWRSWT_CTRL_REG	0x0204	Efuse High Voltage Power Switch Control Register
RTC_SPI_CLK_CTRL_REG	0x0310	RTC SPI Clock Control Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

3.14.6 Register Description

3.14.6.1 0x0000 LOSC Control Register (Default Value: 0x0000_4010)

Offset:0x0000			Register Name: LOSC_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	W	0x0	KEY_FIELD Key Field This field should be filled with 0x16AA, and then the bit0 and bit1 can be written with the new value.
15	R/W	0x0	LOSC_AUTO_SWT_FUNCTION LOSC auto switch function disable 0: Enable 1: Disable
14	R/W	0x1	LOSC_AUTO_SWT_32K_SEL_EN LOSC auto switch 32K clk source sel enable 0: Disable. When the losc losts, the 32K clk source will not change to RC 1: Enable. When the losc losts, the 32K clk source will change to RC (LOSC_SRC_SEL will be changed from 1 to 0)
13:9	/	/	/
8	R/W	0x0	RTC_HHMMSS_ACCE RTC HH-MM-SS access After writing the RTC HH-MM-SS Register, this bit is set and it will be cleared until the real writing operation is finished. After writing the RTC HH-MM-SS Register, the RTC HH-MM-SS Register will be refreshed for at most one second. **Note: Make sure that the bit is 0 for time configuration.**
7	R/W	0x0	RTC_DAY_ACCE RTC DAY access After writing the RTC DAY register, this bit is set and it will be cleared until the real writing operation is finished. After writing the RTC DAY register, the DAY register will be refreshed for at most one second. **Note: Make sure that the bit is 0 for time configuration.**
6:5	/	/	/
4	R/W	0x1	EXT_LOSC_EN External 32.768 kHz Crystal Enable 0: Disable 1: Enable
3:2	R/W	0x0	EXT_LOSC_GSM

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0000			Register Name: LOSC_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
			External 32.768 kHz Crystal GSM 00: Low 01: / 10: / 11: High When GSM is changed, 32K oscillation circuit will arise transient instability. If the autoswitch function (bit 15) is enabled, 32K changes to RC16M with certain probability. The GSM can influence the time of 32K starting oscillation, the more the GSM, the shorter the time of starting oscillation. So modifying GSM is not recommended. If you need to modify the GSM, firstly disable the auto switch function (bit 15), with a delay of 50 us, then change the GSM, the 32K clock source is changed to external clock.
1	R/W	0x0	RTC_SRC_SEL RTC_TIMER Clock Source Select 0: LOSC_SRC 1: 24MDIV32K Before switching the bit, make sure that the 24MDIV32K function is enabled, that is, the bit16 of the 32K Fanout Control Register is 1.
0	R/W	0x0	LOSC_SRC_SEL LOSC Clock source Select. 0: Low Frequency Clock from 16M RC oscillation 1: External 32.768 kHz OSC

NOTE

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

3.14.6.2 0x0004 LOSC Auto Switch Status Register (Default Value: 0x0000_0000)

Offset:0x0004			Register Name: LOSC_AUTO_SWT_STA_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2	R	0x0	EXT_LOSC_STA

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0004			Register Name: LOSC_AUTO_SWT_STA_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
			Work only when AUTO SWITCH function is enabled. 0: External 32.768 kHz OSC work normally 1: External 32.768 kHz OSC work abnormally
1	R/W1C	0x0	LOSC_AUTO_SWT_PEND LOSC auto switch pending 0: No effect 1: Auto switch pending, it means LOSC_SRC_SEL is changed from 1 to 0. Setting 1 to this bit will clear it.
0	R	0x0	LOSC_SRC_SEL_STA Checking LOSC Clock Source Status 0: Low Frequency Clock from 16M RC 1: External 32.768 kHz OSC

3.14.6.3 0x0008 Internal OSC Clock Prescalar Register (Default Value: 0x0000_000F)

Offset:0x0008			Register Name: INTOSC_CLK_PRESCAL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:5	/	/	/
4:0	R/W	0xF	INTOSC_32K_CLK_PRESCAL Internal OSC 32K Clock Prescalar value N. The clock output = Internal RC/32/N. 00000: 1 00001: 2 00002: 3 ..... 11111: 32

3.14.6.4 0x0010 RTC DAY Register

Offset:0x0010			Register Name: RTC_DAY_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	UDF	DAY Day Range from 1 to 65535.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

NOTE

Ensure that the bit[7] of LOSC_CTRL_REG is 0 before updating RTC_DAY_REG.

3.14.6.5 0x0014 RTC HH-MM-SS Register

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

NOTE

Ensure that the bit[8] of LOSC_CTRL_REG is 0 before updating RTC_HH_MM_SS_REG.

3.14.6.6 0x0020 Alarm 0 Day Setting Register (Default Value: 0x0000_0000)

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

3.14.6.7 0x0024 Alarm 0 HH-MM-SS Setting Register

Offset:0x0024			Register Name: ALARM0_HH-MM-SS_SET_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:21	/	/	/
20:16	R/W	UDF	HOUR Range from 0 to 23

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0024			Register Name: ALARM0_HH-MM-SS_SET_REG
Bit	Read/Write	Default/Hex	Description
15:14	/	/	/
13:8	R/W	UDF	MINUTE Range from 0 to 59
7:6	/	/	/
5:0	R/W	UDF	SECOND Range from 0 to 59

3.14.6.8 0x0028 Alarm 0 Enable Register (Default Value: 0x0000_0000)

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

3.14.6.9 0x002C Alarm 0 IRQ Enable Register (Default Value: 0x0000_0000)

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

3.14.6.10 0x0030 Alarm 0 IRQ Status Register (Default Value: 0x0000_0000)

Offset:0x0030			Register Name: ALARM0_IRQ_STA_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W1C	0x0	ALARM0_IRQ_PEND Alarm 0 IRQ Pending bit 0: No effect 1: Pending, alarm 0 counter value is reached

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0030			Register Name: ALARM0_IRQ_STA_REG
Bit	Read/Write	Default/Hex	Description
			If alarm 0 irq enable is set to 1, the pending bit will be sent to the interrupt controller.

3.14.6.11 0x0050 Alarm Configuration Register (Default Value: 0x0000_0000)

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

3.14.6.12 0x0060 32K Fanout Control Gating Register (Default Value: 0x0000_0000)

Offset:0x0060			Register Name: 32K_FOUT_CTRL_GATING_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	HOSC_TO_32K_DIVIDER_ENABLE 0: Disable the hosc 24M to 32K divider circuit 1: Enable the hosc 24M to 32K divider circuit
15:3	/	/	/
2:1	R/W	0x0	LOSC_OUT_SRC_SEL 00: RTC_32K (select by RC_CLK_SRC_SEL & LOSC_SRC_SEL) 01: LOSC 10: HOSC divided 32K
0	R/W	0x0	32K_FANOUT_GATING Configuration of LOSC output, and no LOSC output by default. 0: Mask LOSC output gating 1: Enable LOSC output gating

3.14.6.13 0x0100+N*0x0004 General Purpose Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

NOTE

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

3.14.6.14 0x0120 Fast Boot Information Register0 (Default Value: 0x0000_0000)

Offset:0x0120			Register Name: FBOOT_INFO_REG0
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	Fast Boot Information, refer to BROM spec.

3.14.6.15 0x0124 Fast Boot Information Register1 (Default Value: 0x0000_0000)

Offset:0x0124			Register Name: FBOOT_INFO_REG1
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	Fast Boot Information, refer to BROM spec.

3.14.6.16 0x0160 DCXO Control Register (Default Value: 0x883F_10F7)

Offset:0x0160			Register Name: DCXO_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x1	CLK_REQ_ENB 0: Enable DCXO wake up function 1: Disable DCXO wake up function
30:28	/	/	/
27:24	R/W	0x8	DCXO_ICTRL DCXO current control value
23	/	/	/
22:16	R/W	0x3F	DCXO_TRIM DCXO cap array value The capacity cell is 55 fF.
15:13	/	/	/
12:8	R/W	0x10	DCXO_BG DCXO bandgap output voltage
7	R/W	0x1	DCXO_LDO_INRUSHB

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0160			Register Name: DCXO_CTRL_REG
Bit	Read/Write	Default/Hex	Description
			DCXO LDO driving capacity signal, active high
6	R/W	0x1	XTAL_MODE Xtal mode enable signal, active high 0: For external clk input mode 1: For normal mode
5:4	R/W	0x3	DCXO_RFCLK_ENHANCE DCXO rfclk enhance Enhance driving capacity of output OUT_RF_REFCLK, 0x0 for 5 pF, 0x1 for 10 pF, 0x2 for 15 pF, 0x3 for 20 pF.
3	/	/	/
2	R/W	0x1	RSTO_DLY_SEL For Debug Use Only. It cannot configure to 0 in normal state.
1	R/W	0x1	DCXO_EN DCXO enable 1: Enabled 0: Disabled
0	R/W	0x1	CLK16M_RC_EN 1: Enable 0: Disable The related register configuration is necessary to ensure the reset debounce circuit has a stable clock source. The first time SoC starts up, by default, the reset debounce circuit of SoC uses 32K divided by RC16M. In power-off, software reads the related bit to ensure whether EXT32K is working normally, if it is normal, first switch the clock source of debounce circuit to EXT32K, then close RC16M. Without EXT32K scenario or external RTC scenario, software confirms firstly whether EXT32K is working normally before switching, or software does not close RC16M.

3.14.6.17 0x0190 RTC_VIO Regulation Register (Default Value: 0x0000_0004)

Offset:0x0190			Register Name:RTC_VIO_REG
Bit	Read/Write	Default/Hex	Description
31:5	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0190			Register Name:RTC_VIO_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
4	R/W	0x0	V_SEL 0: resistance divider 1: band gap
3	/	/	/
2:0	R/W	0x4	RTC_VIO_REGU These bits are useful for regulating the RTC_VIO from 0.6 V to 1.3 V, and the regulation step is 0.1 V. 000: 1.0 V 001: 0.6 V (the configuration can cause RTC reset) 010: 0.7 V 011: 0.8 V 100: 0.9 V 101: 1.1 V 110: 1.2 V 111: 1.3 V RTC-VIO is provided power for RTC digital part, the default value is 0.9 V. After power-on, the software sets the field to 0.8 V to save power-consumption.

3.14.6.18 0x01F0 IC Characteristic Register (Default Value: 0x0000_0000)

Offset:0x01F0			Register Name: IC_CHARA_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	R/W	0x0	KEY_FIELD Key Field This field should be written as 0x16AA. Writing any other value in this field aborts the write-operation.
15:0	R/W	0x0	ID_DATA Return 0x16AA only if the KEY_FIELD is set as 0x16AA when read those bits, otherwise return 0x0.

3.14.6.19 0x01F4 VDD Off Gating Control Register (Default Value: 0x0000_0021)

Offset:0x01F4			Register Name: VDD_OFF_GATING_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	W	0x0	KEY_FIELD Key Field

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x01F4			Register Name: VDD_OFF_GATING_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
			This field should be written as 0x16AA, and then the bit 15 can be configured.
15	WAC	0x0	RTC_ISO_SW_CTRL (For Debug Use Only) 0: Hardware sets the isolation function 1: Software sets the isolation function To prevent the hardware detection from failing when VDD-SYS is powered off, the bit implements the function of setting isolation by software backup method. The bit can be only cleared to 0 after releasing the reset_b (reset_b indicates the reset signal is active low).
14:12	/	/	/
11:4	R/W	0x2	VCCIO_DET_SPARE Bit[7:5]: Reserved, default=0 Bit[4]: Bypass debounce circuit, defaule=0 (bypass debounce circuit) Bit[3]: Enable control, defaule=0 (disable VCC-IO detection, force the detection output to "1") Bit[2:0]: Gear adjustment 000: Detection threshold is 2.5 V 001: Detection threshold is 2.6 V 010: Detection threshold is 2.7 V (default) 011: Detection threshold is 2.8 V 100: Detection threshold is 2.9 V 101: Detection threshold is 3 V 110: N/A 111: N/A
3:1	/	/	/
0	R/W	0x1	VCCIO_DET_BYPASS_EN 0: not bypass 1: bypass

3.14.6.20 0x0204 Efuse High Voltage Power Switch Control Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0204			Register Name: EFUSE_HV_PWRSWT_CTRL_REG
Bit	Read/Write	Default/Hex	Description
			0: close power switch

3.14.6.21 0x0310 RTC SPI Clock Control Register (Default Value: 0x0000_0000)

Offset:0x0310			Register Name: RTC_SPI_CLK_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	RTC Reg CFG SPI Clock Gating 0: Gating 1: Not Gating Before configurating RTC register, the clock divider of SPI needs be configured firstly, then clock gating needs be enabled. **Note: Frequency division and clock gating can not be set at the same time.**
30:5	/	/	/
4:0	R/W	0x9	RTC Reg CFG SPI Clock Divider: M Actual SPI Clock = AHBS1/(M+1), (0 to 15) The default frequency of AHBS1 is 200 MHz, and the default frequency of SPI Clock is 20 MHz. **Note: The SPI clock can not exceed 50 MHz, or else RTC register may be abnormal.**

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Contents

4 Memory 258
4.1 SDRAM Controller (DRAMC) 258
4.2 Nand Flash Controller (NDFC) 259
4.2.1 Overview 259
4.2.2 Block Diagram 260
4.2.3 Operations and Functional Descriptions 261
4.2.4 Programming Guidelines 274
4.2.5 Register List 276
4.2.6 Register Description 278
4.3 SD/MMC Host Controller (SMHC) 320
4.3.1 Overview 320
4.3.2 Block Diagram 320
4.3.3 Operations and Functional Descriptions 321
4.3.4 Programming Guidelines 335
4.3.5 Register List 340
4.3.6 Register Description 341

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figures

Figure 4-1 NDFC Block Diagram 260
Figure 4-2 Conventional Serial Access Cycle Diagram (SAM0) 263
Figure 4-3 EDO Type Serial Access after Read Cycle (SAM1) 263
Figure 4-4 Extending EDO Type Serial Access Mode (SAM2) 263
Figure 4-5 Command Latch Cycle 264
Figure 4-6 Address Latch Cycle 264
Figure 4-7 Write Data to Flash Cycle 264
Figure 4-8 Waiting R/B# Ready Diagram 265
Figure 4-9 WE# High to RE# Low Timing Diagram 265
Figure 4-10 RE# High to WE# Low Timing Diagram 265
Figure 4-11 Address to Data Loading Timing Diagram 266
Figure 4-12 Page Read Command Diagram 267
Figure 4-13 Page Program Diagram 268
Figure 4-14 EF-NAND Page Read Diagram 268
Figure 4-15 Interleave Page Read Diagram 269
Figure 4-16 Internal DMA Descriptor Chain Structure 270
Figure 4-17 SMHC Block Diagram 321
Figure 4-18 Command Token Format 323
Figure 4-19 Response Token Format 324
Figure 4-20 Data Packet Format for SDR 325
Figure 4-21 Data Packet Format for DDR 326
Figure 4-22 Data Packet Format for DDR in HS400 Mode 327
Figure 4-23 Single-Block and Multi-Block Read Operation 328
Figure 4-24 Single-Block and Multi-Block Read Operation 328
Figure 4-25 Phase Offset Diagram of Commands and Data in SDR Mode 330
Figure 4-26 Phase Offset Diagram of Commands and Data in DDR4 Mode (SMHC_NTSR[31] = 0) 330
Figure 4-27 Phase Offset Diagram of Commands and Data in DDR4 (2x) Mode (SMHC_NTSR[31] = 1) 331
Figure 4-28 IDMAC Descriptor Structure Diagram 332

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Tables

Table 4-1 NDFC Sub-blocks 260
Table 4-2 NDFC External Signals 261
Table 4-3 NDFC Clock Sources 262
Table 4-4 Definition of the Config Field in DMA Descriptor 270
Table 4-5 Definition of the Size Field in DMA Descriptor 271
Table 4-6 Definition of the Buff_Addr Field in DMA Descriptor 271
Table 4-7 Definition of the Next_Desc Field in DMA Descriptor 271
Table 4-8 Data Structures of the Sequential Type and Interleaved Type 272
Table 4-9 SMHC Sub-blocks 321
Table 4-10 SMHC External Signals 322
Table 4-11 SMHC Clock Sources 322
Table 4-12 Speed Modes Supported by SD 3.0 328
Table 4-13 Speed Modes Supported by eMMC 5.1 329
Table 4-14 Phase Offset Value of Commands and Data in SDR Mode 330
Table 4-15 Phase Offset Value of Commands and Data in DDR4 Mode 331
Table 4-16 Phase Offset Value of Commands and Data in DDR4 (2x) Mode 331
Table 4-17 DES0 Definition 332
Table 4-18 DES1 Definition 333
Table 4-19 DES2 Definition 333
Table 4-20 DES3 Definition 334

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4 Memory

4.1 SDRAM Controller (DRAMC)

The DRAMC is embedded with 256 MB or 128 MB DDR3.

The DRAMC has the following features:

- Embedded with 256 MB DDR3 (only for R329-N4)
- Embedded with 128 MB DDR3 (only for R329-N3 and R329-S3)
- Supports clock frequency up to 774 MHz for DDR3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.2 Nand Flash Controller (NDFC)

4.2.1 Overview

The NDFC is the NAND flash controller which supports all NAND flash memory available in the market. New types of flash can be supported by software re-configuration.

The NDFC has a built-in on-the-fly error correction code (ECC) feature with BCH algorithm to enhance the reliability. It can detect and correct up to 80 bits error per 1024 bytes data. With the on-chip BCH code ECC circuit, the CPU is freed for other tasks. You can disable the ECC feature by software.

The NDFC supports transferring data via the DMA or CPU memory-mapped IO. It provides automatic timing control for reading or writing external Flash and maintains the proper relativity for CLE, CE#, and ALE control signal lines. There are three modes for serial read access: the mode0 is for conventional serial access, the mode1 is for EDO type, and the mode2 is for extension EDO type. The NDFC can monitor the status of the R/B# signal line.

Block management and wear leveling management are implemented in software.

The NDFC has the following features:

- Supports all SLC/MLC/TLC flash and EF-NAND memory available in the market
- Supports configuring randomize seed by software
- Software configuration method for various systems and memory types
- Up to 8-bit data bus width
- Supports 1024, 2048, 4096, 8192, 16384, and 32768 bytes size per page
- Conventional and EDO serial access method for serial reading Flash
- 80 bits/1 KB on-the-fly BCH code ECC check and error correction
- Output bits number information about the corrected errors
- ECC automatic disable function for all 0xff data
- NDFC status information is reported by its registers, and interrupt is supported
- One command FIFO
- Internal DMA controller based on the chain-structured descriptor list
- Two 256x32-bit RAM for Pipeline Procession
- Supports SDR, ONFI DDR1.0, Toggle DDR1.0, ONFI DDR2.0, and Toggle DDR2.0 RAW NAND FLASH
- Maximum IO rate of 50 MHz in SDR mode, and 100 MHz in both DDR1.0 and DDR2.0 modes

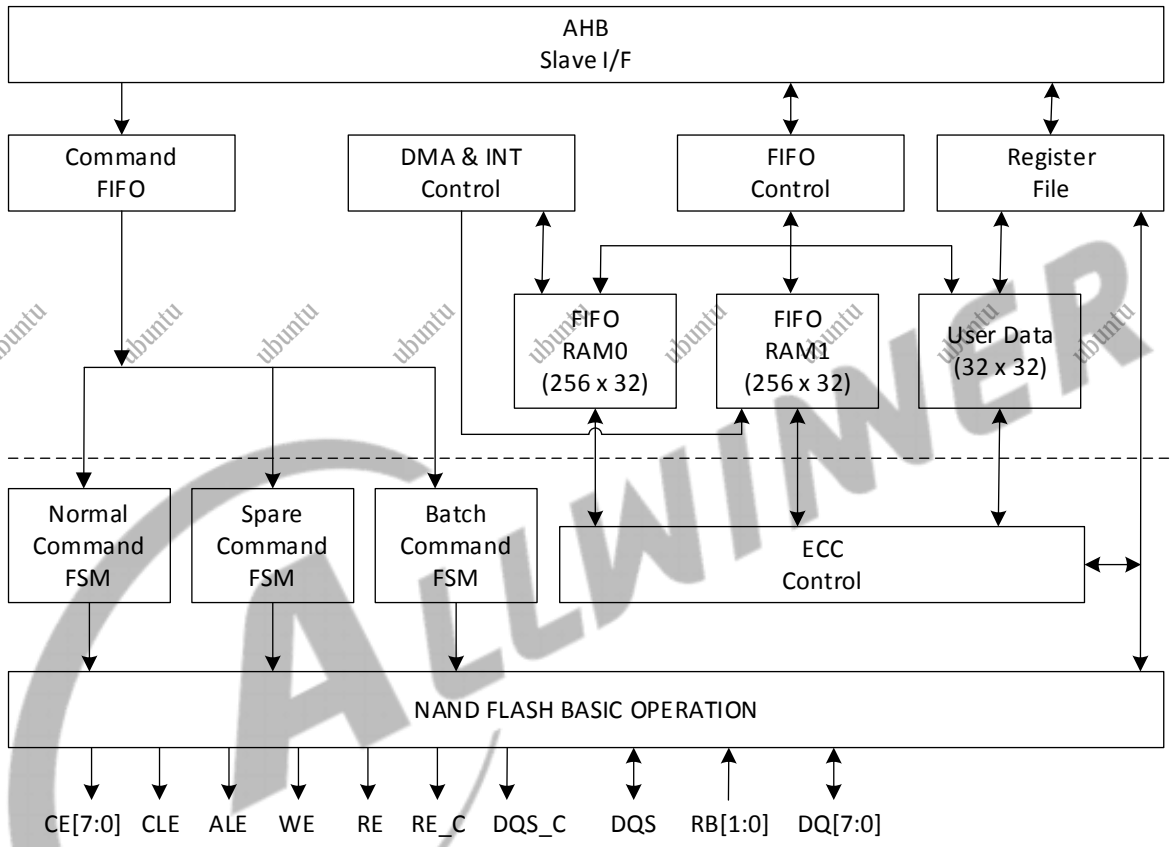
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

- Self-debug for NDFC debug

4.2.2 Block Diagram

The following figure shows the block diagram of the NDFC.

Figure 4-1 NDFC Block Diagram



NDFC contains the following sub-blocks:

Table 4-1 NDFC Sub-blocks

Sub-block	Description
AHB Slave I/F	The AHB protocol resolution module, acts as the AHB slave.
Command FIFO	Used for pipelining commands.
DMA & INT Control	The internal DMA module that is used to read and write data between the NAND flash and memory.
FIFO Control	A buffer for the data stream between the bus and the working clock domain.
Register File	The register module, including the module control registers, DMA control registers, and so on.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Sub-block	Description
FIFO RAM0/1	The ping-pong RAMs that are used to implement the data parallel processing among the DMA, CPU, and NDFC. The size of each RAM is 1024 bytes.
User Data	Used for storing the user data. It can be read or written via the bus.
Normal Command FSM	The finite state machine for implementing the general commands. It is used to transmit the main area data and commands.
Spare Command FSM	The finite state machine for implementing the redundant commands. It is used to transmit the redundant data and the related commands.
Batch Command FSM	The finite state machine for implementing the batch commands. It is used to transmit multi-block data in a batch.
ECC Control	The ECC module, used for generating the ECC codes and error correction.
NAND Flash Basic Operation	The interface timing generating module. It is used to implement the protocol of the NAND flash.

4.2.3 Functional Descriptions

4.2.3.1 External Signals

The following table describes the external signals of NDFC. DQ0 to DQ7 and DQS are bidirectional I/O. WE, ALE, CLE, CE, and RE are output pins, and RB0 and RB1 are input pins. The RB pin in the NAND device is an open-drain driver, which needs a pull-up resistor.

Table 4-2 NDFC External Signals

Signal	Description	Type
NAND-WE	Write enable	O
NAND-RE	Read enable	O
NAND-ALE	Address latch enable, high is active	O
NAND-CLE	Command latch enable, high is active	O
NAND-CE0	Chip enable, low is active	O
NAND-RB0	Ready/busy, low is active	I
NAND-DQ0	Data input/output	I/O
NAND-DQ1	Data input/output	I/O
NAND-DQ2	Data input/output	I/O
NAND-DQ3	Data input/output	I/O
NAND-DQ4	Data input/output	I/O

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Signal	Description	Type
NAND-DQ5	Data input/output	I/O
NAND-DQ6	Data input/output	I/O
NAND-DQ7	Data input/output	I/O
NAND-DQS	Data strobe	I/O

4.2.3.2 Clock Sources

To ensure the ECC efficiency, the ECC engine and NDFC internal logic use different clocks. The clock of NDFC internal logic is set by **NAND_0_CLK_REG**, and the clock of the ECC engine is set by **NAND_1_CLK_REG**. Note that **NAND_0_CLK_REG** sets the internal logic clock of NDFC, but the frequency for the external NAND Flash device is half of the frequency for the NDFC internal logic clock. That is, if external Nand Flash runs at 40 MHz, then NDFC needs to set to 80 MHz.

Both the ECC engine and NDFC internal logic have 4 different clock sources. You can select one of them as the ECC engine or internal logic clock source. The following table describes the clock sources of NDFC.

For the clock setting, configuration, and gating information, refer to section 3.3 "CCU".

Table 4-3 NDFC Clock Sources

Clock Sources	Description
DCXO24M	24 MHz Crystal
PLL_PERI(1X)	Peripheral Clock, the default value is 600 MHz
PLL_PERI(2X)	Peripheral Clock, the default value is 1.2 GHz
PLL_AUDIO0(DIV2)	Audio clock, the default value is 1536 MHz

4.2.3.3 Timing Diagram

Typically, there are two kinds of serial access methods. One is the conventional method which fetches data at the rising edge of the NDFC_RE# signal line. The other is the EDO type which fetches data at the next falling edge of the NDFC_RE# signal line. The following figures show the timing diagrams.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 4-2 Conventional Serial Access Cycle Diagram (SAM0)

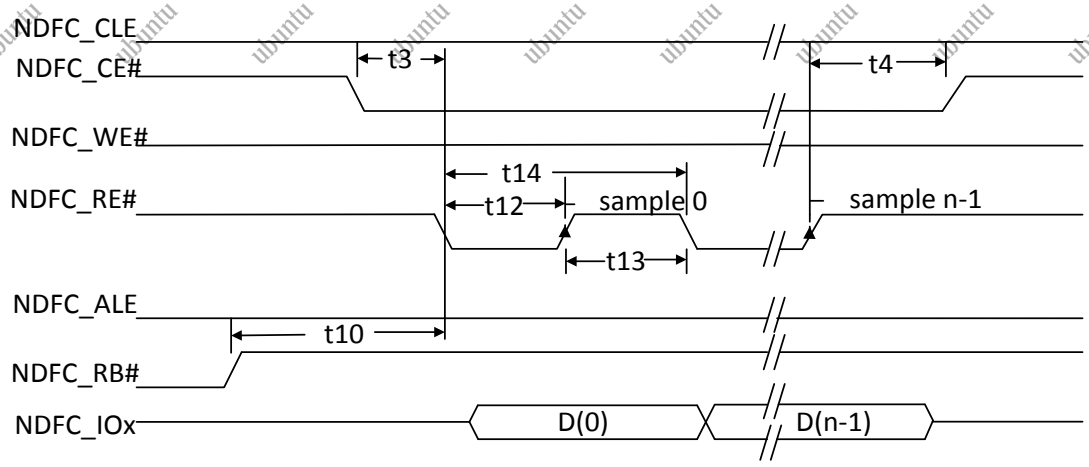


Figure 4-3 EDO Type Serial Access after Read Cycle (SAM1)

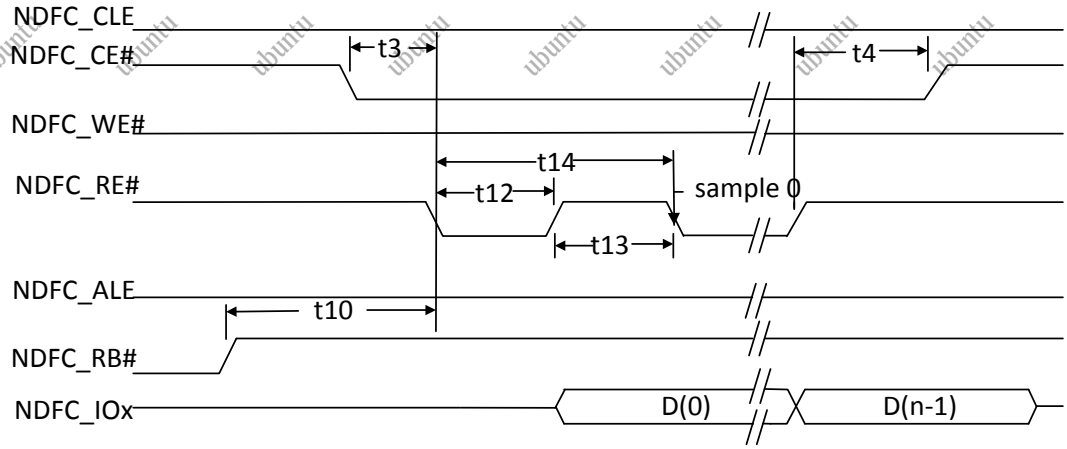
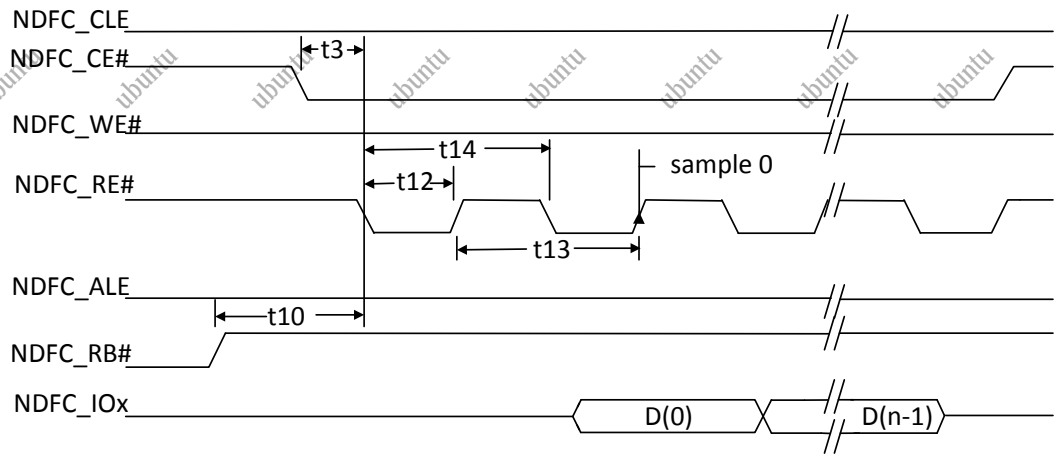


Figure 4-4 Extending EDO Type Serial Access Mode (SAM2)



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 4-5 Command Latch Cycle

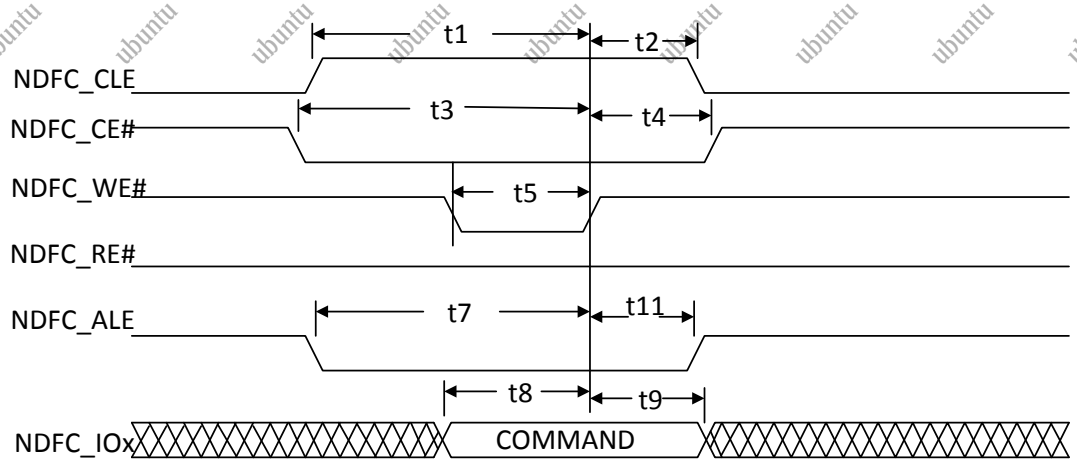


Figure 4-6 Address Latch Cycle

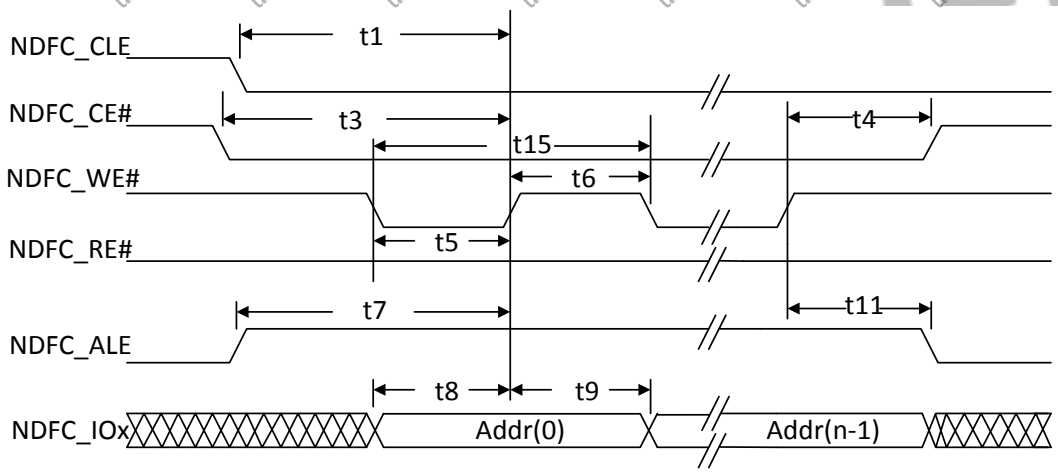
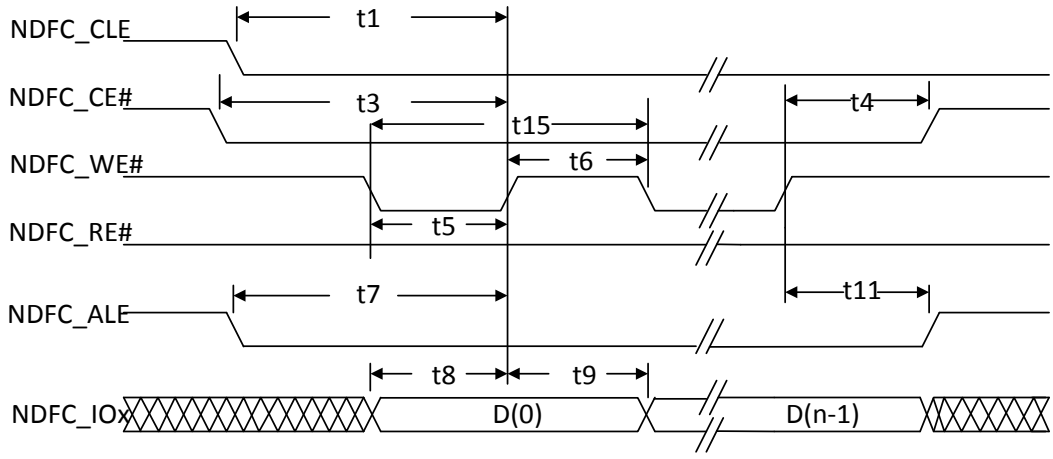


Figure 4-7 Write Data to Flash Cycle



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 4-8 Waiting R/B# Ready Diagram

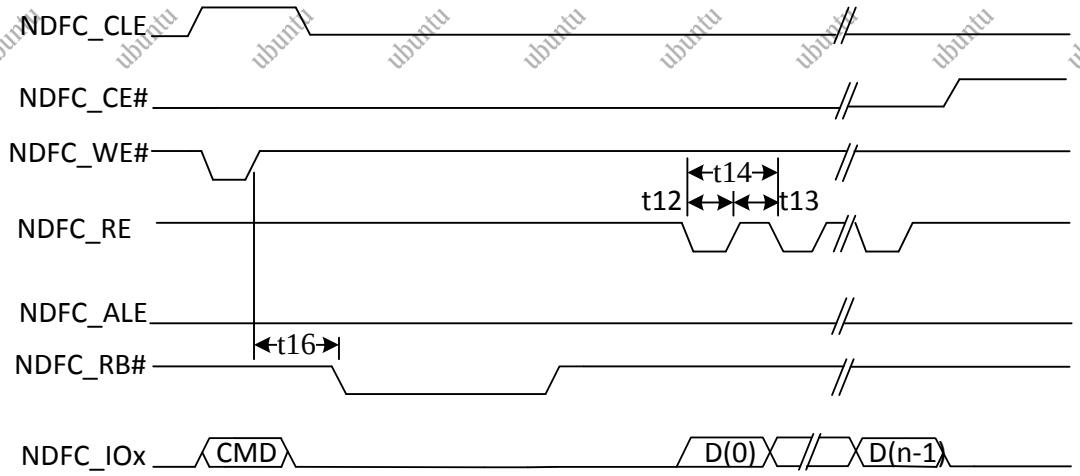


Figure 4-9 WE# High to RE# Low Timing Diagram

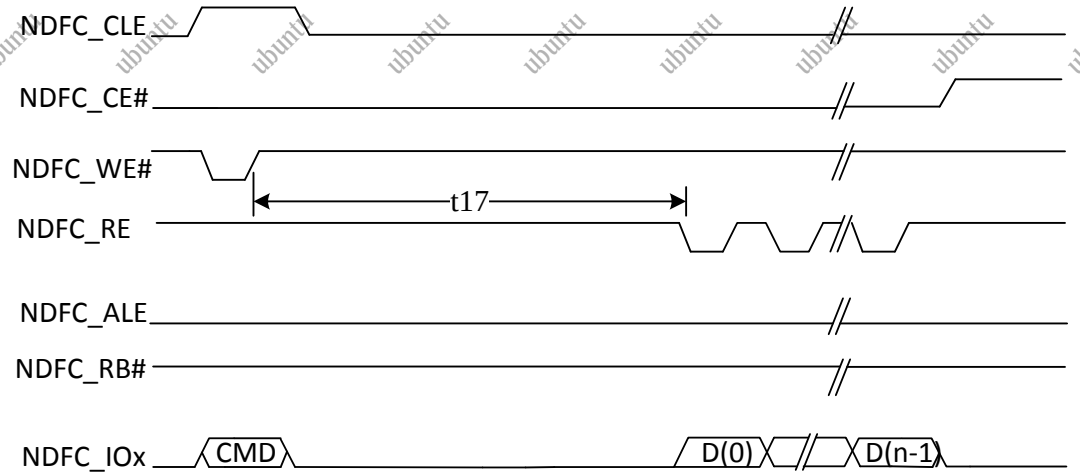
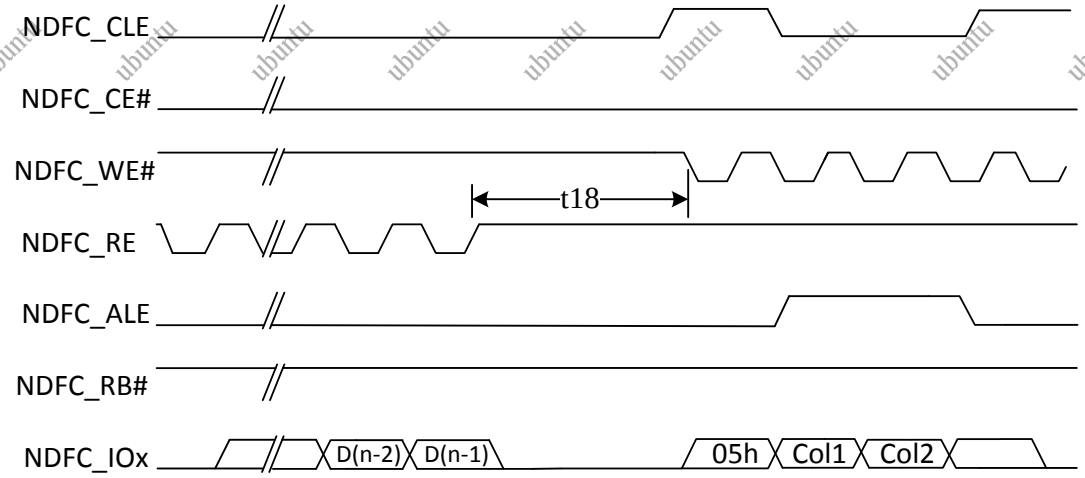
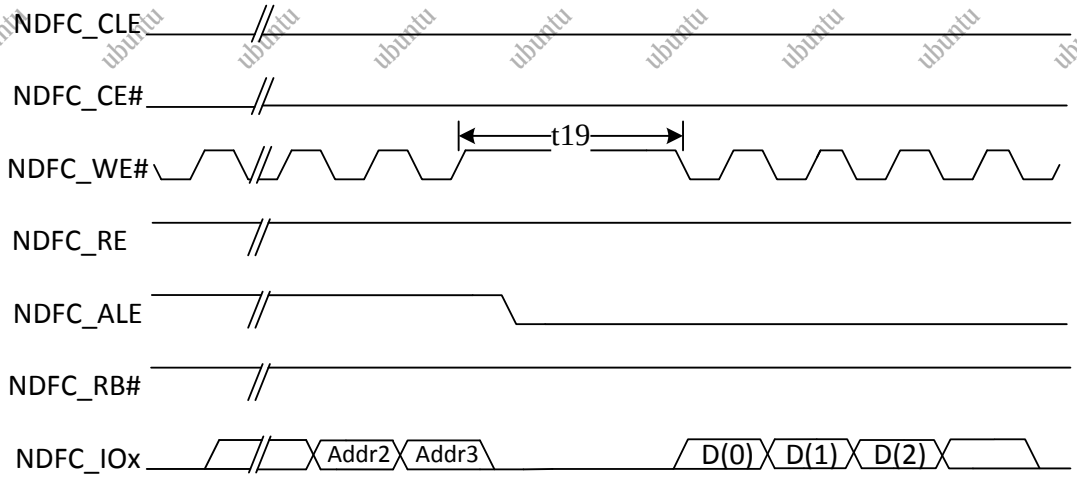


Figure 4-10 RE# High to WE# Low Timing Diagram



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 4-11 Address to Data Loading Timing Diagram



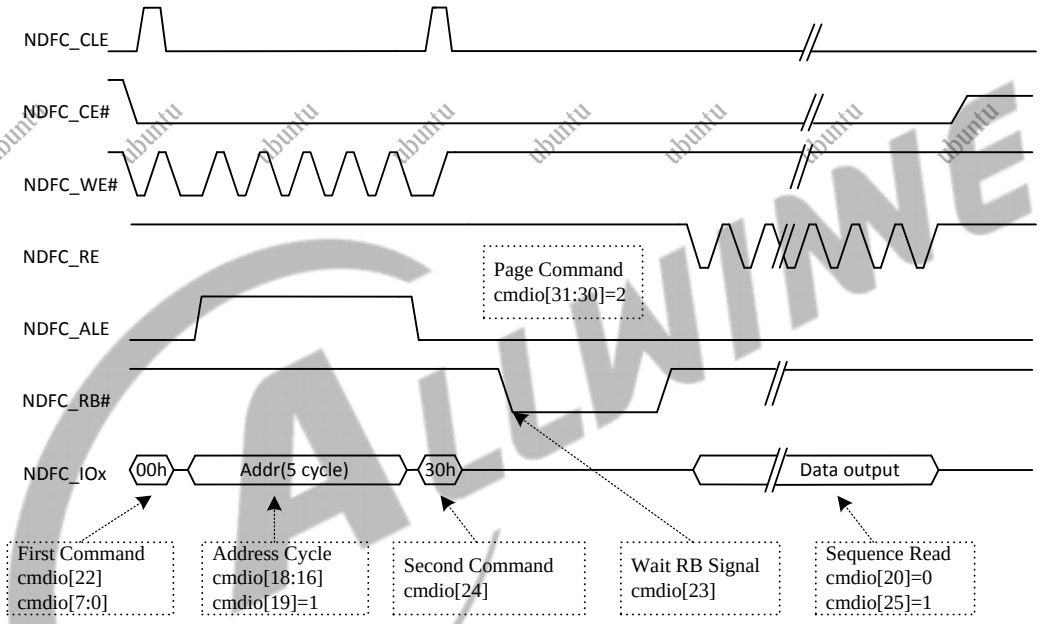
Timing cycle list:

ID	Parameter	Timing (ns)	Notes
t1	NDFC_CLE setup time	2T	
t2	NDFC_CLE hold time	2T	
t3	NDFC_CE setup time	2T	
t4	NDFC_CE hold time	2T	
t5	NDFC_WE# pulse width	T(1)	
t6	NDFC_WE# hold time	T	
t7	NDFC_ALE setup time	2T	
t8	Data setup time	T	
t9	Data hold time	T	
t10	Ready to NDFC_RE# low	3T	
t11	NDFC_ALE hold time	2T	
t12	NDFC_RE# pulse width	T	
t13	NDFC_RE# hold time	T	
t14	Read cycle time	2T	
t15	Write cycle time	2T	
t16	NDFC_WE# high to R/B# busy	T_WB(2)	Specified by timing configure register (NDFC_TIMING_CFG)
t17	NDFC_WE# high to NDFC_RE# low	T_WHR(3)	Specified by timing configure register (NDFC_TIMING_CFG)
t18	NDFC_RE# high to NDFC_WE# low	T_RHW(4)	Specified by timing configure register (NDFC_TIMING_CFG)

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

ID	Parameter	Timing (ns)	Notes
t19	Address to Data Loading time	T_ADL(5)	Specified by timing configure register (NDFC_TIMING_CFG)
Note(1): T is the cycle of the internal clock. Note(2), (3), (4), and (5): These values are configurable in the NAND flash controller. The value of T_WB could be 14*2T/22*2T/30*2T/38*2T, the value of T_WHR could be 0*2T/6*2T/14*2T/22*2T, the value of T_RHW could be 4*2T/12*2T/20*2T/28*2T, and the value of T_ADL could be 0*2T/6*2T/14*2T/22*2T.			

Figure 4-12 Page Read Command Diagram



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 4-13 Page Program Diagram

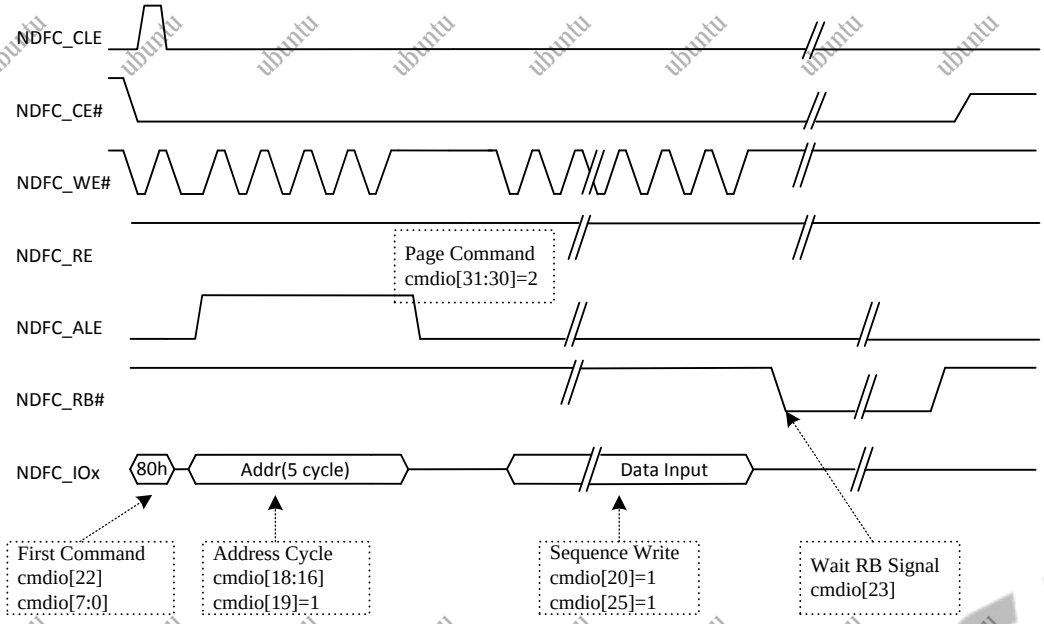
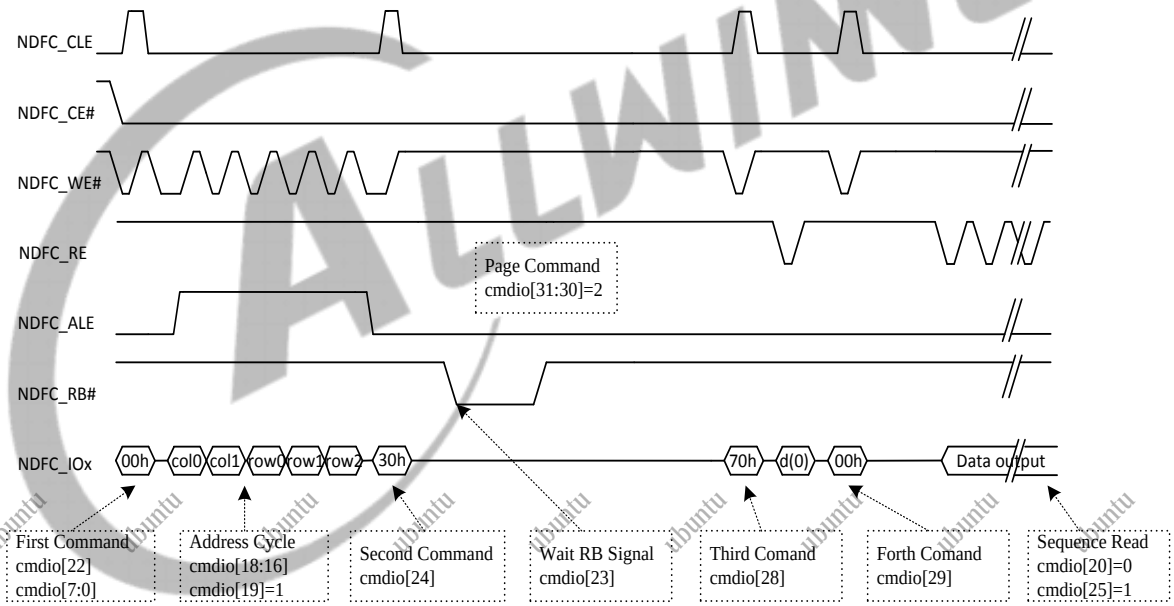
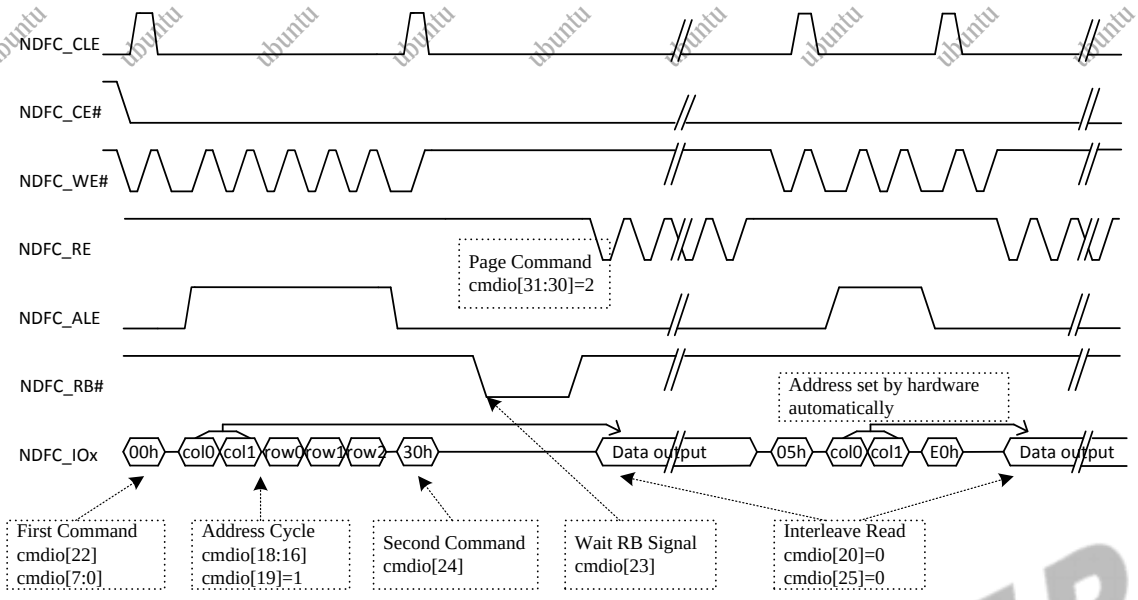


Figure 4-14 EF-NAND Page Read Diagram



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 4-15 Interleave Page Read Diagram



4.2.3.4 Data Path

Writing Data from the External Master to the NAND Flash

The external master writes data to the FIFO via the AHB slave interface, and then the data are stored in the FIFO RAM0/1. The internal FSM reads data from the FIFO RAM0/1 and writes the data to the ECC control module for generating the ECC code. After the ECC coding, the main-area data will be transferred to the **NAND Flash Basic Operation** via the **Normal Command FSM** or **Batch Command FSM** to implement the protocol, and then written into the NAND flash; while the spare-area data will be transferred to the **NAND Flash Basic Operation** via the **Spare Command FSM** or **Batch Command FSM** to implement the protocol, and then written into the NAND flash.

Reading Data from the External Master to the NAND Flash

The internal FSM reads the main-area data from the NAND flash via a read NAND flash command, and transmit the data to the FIFO RAM0/1 and ECC control module. Then the internal FSM reads the spare-area data, and initiate the ECC control module to implement ECC decoding and error correction. After that, the internal FSM will inform the external master to read data via the AHB slave interface.

Transferring Data from the Memory to the NAND Flash via the Internal DMA

The internal DMA transfer data from the specified address of the memory to the FIFO RAM0/1, and then the internal FSM reads the data from the FIFO RAM0/1 and writes the data to the ECC control module for ECC coding. After ECC coding, the main-area data will be transferred to the **NAND Flash Basic Operation** via the **Normal Command FSM** or **Batch Command FSM** to implement the protocol, and then written into the NAND

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

flash; while the spare-area data will be transferred to the NAND Flash Basic Operation via the Spare Command FSM or Batch Command FSM to implement the protocol, and then written into the NAND flash.

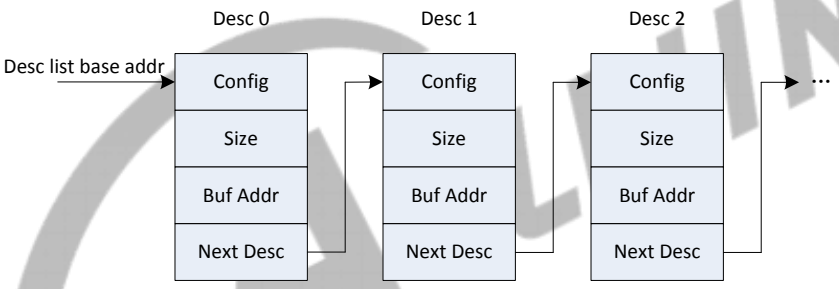
Transferring Data from the NAND Flash to the Memory via the Internal DMA

The internal DMA reads the main-area data from the NAND flash via a read NAND flash command, and transmit the data to the FIFO RAM0/1 and ECC control module. Then the internal DMA reads the spare-area data, and initiate the ECC control module to implement ECC decoding and error correction. After that, the internal DMA transfer the data to the specified address of the memory.

4.2.3.5 Internal DMA Controller Descriptors

The internal DMA controller of the NDFC can transfer data between DMA FIFO in NDFC and DMA buffer in host memory using DMA descriptors. The following figure shows the DMA descriptors in the host memory with the chain structure.

Figure 4-16 Internal DMA Descriptor Chain Structure



The start address of the DMA descriptor list must be word (32-bit) aligned and will be configured to NDFC_MDMA_DLBA_REG. Each DMA descriptor consists of four words (32 bits).

The following tables show the field definition of the DMA descriptor.

Table 4-4 Definition of the Config Field in DMA Descriptor

Config	
Bit	**Description**
31:4	/
3	FIRST_FLAG When set, this bit indicates that this descriptor contains the first buffer of data. Must be set to 1 in the first descriptor.
2	LAST_FLAG When set, this bit indicates that the buffers pointed by this descriptor are the last data buffer.
1:0	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Table 4-5 Definition of the Size Field in DMA Descriptor

Size	
Bit	**Description**
31:16	/
15:0	BUFF_SIZE These bits indicate the data buffer byte size, which must be a multiple of 8 bytes. If this field is 0, the DMA ignores this buffer and proceeds to the next descriptor.

Table 4-6 Definition of the Buff_Addr Field in DMA Descriptor

Buff Addr	
Bit	**Description**
31:0	BUFF_ADDR These bits indicate the physical address of the DMA data buffer in host memory. The buffer address must be 4 bytes aligned.

Table 4-7 Definition of the Next_Desc Field in DMA Descriptor

Next Description	
Bit	**Description**
31:0	NEXT_DESC_ADDR These bits indicate the pointer to the physical host memory of the next descriptor is present.

4.2.3.6 Randomizer

Data randomization is a solution for suppressing errors in data storage systems. The NDFC providers the randomizer for the large-capacity NAND flash to randomize data.

When enabled, during the write operation, the NDFC randomizes the data before writing the data to the flash, and during the read operation, the NDFC de-randomizes the data that it reads from the flash before writing the data to the buffer.

4.2.3.7 Error Correction Code (ECC)

Error correction code (ECC) is another solution for suppressing errors. It is used to correct the errors present in the received data bit-stream so that we can get the original data. The ECC processes data based on 1 KB data block. Therefore, the ECC performance of the NDFC is described in the format of n bits/ 1 KB, which means at most n bits of errors in the checked 1 KB data can be corrected.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

The NDFC supports 15 ECC modes, ranging from 16 bits/1 KB to 80 bits/1 KB. The ECC mode is decided by the device spare-area size by default and can be configured by NDFC_ECC_CTL. If the corresponding ECC data block exceeds the error-correcting capability, the controller will report an ECC error interrupt (NDFC_ECC_ST).

When the error correction executed successfully, you can check the number of the corrected bits by NDFC_ERR_CNT_N.

4.2.3.8 Data Structures

Based on the user_data location, the NDFC has two types of data structure: the interleaved type and sequential type. The following table shows the data structures of the two types.

Table 4-8 Data Structures of the Sequential Type and Interleaved Type

Type	Main Area	Spare Area
Interleaved	Data (Page Size)	User Data + ECC (14*t/8*Page Size)
Sequential	Data (ECC Data Block) + User Data + ECC (14*t/8*ECC Block Size) + Data (ECC Data Block) + User Data + ECC (14*t/8*ECC Block Size) + Data (ECC Data Block) + User Data + ECC (14*t/8*ECC Block Size) + ...	

NOTE

- The length of ECC code is decided by the error correction capability and the error correction unit. For the error correction unit 1 KB, the length of the correction data is 14*t bits.
- The length of the user data can be configured by NDFC_USER_DATA_LEN_N, and the value of the corresponding user data can be configured by NDFC_USER_DATA_LEN_N.
- When the ECC is disabled, the ECC part should be omitted. That is, the data structure consists of only the Data and User Data parts.

For the interleaved type, the main area stores the valid written data with the length of the page size, and the spare area stores the user data and the ECC code. That is, the user data and the ECC code are always stored from the page size position. The ECC code is generated by the BCH calculation of the main-area data and the user data.

For the sequential type, every 1 KB main-area data is grouped with its user data and ECC code. The data structure of the sequential type consists of n groups of “Data (ECC Data Block) + User Data + ECC (14*t/8*ECC Block Size)”, and the group number n is decided by the page size.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.2.3.9 NDFC Data Block Mask Register

ECC_DATA_BLOCK is written or read by the value of **NDFC_DATA_BLOCK_MASK.** In the practical application scenarios, to save the capacity, only the reading operation uses the function and the writing operation does not.

Based on the location of the user_data, there are two modes of reading operation: the sequential mode and interleaved mode.

- **Sequential mode**: For every 1 KB main-area data, the user_data, and ECC value are stored next to the main-area data.
- **Interleaved mode**: The starting position for storing the user_data and ECC values are always from the page_size position.

The usage of the register is different for the NDFC data of interleaved type and sequential type when any ECC_DATA_BLOCK within a page is read through batch command (**NDFC_CMD**[NDFC_CMD_TYPE] is 0x10).

- The sequential type can only support successive ECC_DATA_BLOCK, and the register value can only be 0x1, 0x3, 0x7, and so on. While the interleaved type has no limit.
- For the both types, the first reading ECC_DATA_BLOCK is used to calculate the corresponding column address, and the column address is written to 0x14 and 0x18 register.

4.2.3.10 NDFC Enhanced Feature Register

The bit[24] and bit[23:16] of the register are used to judge whether free space needs to be padded random data except valid data when the batch command function is used.

Take a SanDisk chip (SDTNQGAMA-008G) as an example:

Refer to the specification of the SanDisk chip, the page_size of the SanDisk chip is (16384 + 1280) bytes, but the BCH level uses 40bit/1K, if user_data is 32 bytes, then the used space is 1152 bytes(14*40/8*16 + 32), the 128 bytes (1280-1152) space is not written. If there need be filled with 1 page, then the bit[24] of the register can be set to 1, and the bit[23:16] is written to 0x80, that the controller can automatically pad 128 bytes random data.

NOTE

Make sure that random function is enabled if there need be sent random data, that is, the **NDFC_ECC_CTL**[NDFC_RANDOM_EN] is 0x1, or else the padding data is non-random, is all-0.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.2.4 Programming Guidelines

4.2.4.1 Initializing NAND Flash

Follow the steps below to initialize the NAND flash.

Step 1 Read NDFC_ST[NDFC_RB_STATE0] to wait for the flash to enter the idle status.

Step 2 Configure NDFC_CMD[NDFC_SEND_FIRST_CMD] to 1 to send the first command, configure NDFC_CMD[NDFC_WAIT_FLAG] to 1 to set wait RB; write 0xFF to NDFC_CMD[NDFC_CMD_LOW_BYTE] to send reset command.

Step 3 Read NDFC_ST[NDFC_CMD_INT_FLAG] to wait for the transfer command end interrupt flag pending, after pending, write 1 to clear the flag.

4.2.4.2 Reading NAND Flash ID

To read the NAND flash ID, follow the steps below:

Step 1 Write 1 to NDFC_CMD[NDFC_SEND_ADR] to allow the NDFC to transfer addresses and configure NDFC_CMD[NDFC_ADR_NUM] to specify the number to be transferred addresses. Then calculate the address of the block to be operated and write the address in NDFC_ADDR_LOW and NDFC_ADDR_HIGH.

Step 2 Write 1 to NDFC_CMD[NDFC_SEND_FIRST_CMD] to send the first command; write 1 to the NDFC_CMD[NDFC_DATA_TRANS] to enable the data transfer; write 0 to NDFC_CMD[NDFC_ACCESS_DIR] to configure the direction as read NAND flash; write 0x90 to the NDFC_CMD[NDFC_CMD_LOW_BYTE] to send the read ID command.

Step 3 Read NDFC_ST[NDFC_CMD_INT_FLAG] to wait for the transfer command end interrupt flag pending. After pending, write 1 to clear the flag.

Step 4 Read the value of the ID from the RAM0_BASE.

4.2.4.3 Erasing NAND Flash

To erase the NAND flash, follow the steps below:

Step 1 Read NDFC_ST[NDFC_RB_STATE0] to wait for the flash to enter the idle status.

Step 2 Configure NDFC_CMD[NDFC_SEND_FIRST_CMD] to 1 to send the first command, configure NDFC_CMD[NDFC_WAIT_FLAG] to 1 to set wait RB;

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Configure NDFC_CMD[NDFC_SEND_ADR] to 1 to enable transfer address, configure NDFC_CMD[NDFC_ADR_NUM] to set the number of address to be transferred;

Write the address of the block to be erased in NDFC_ADDR_LOW and NDFC_ADDR_HIGH;

Set NDFC_CMD[NDFC_CMD_LOW_BYTE] to 0x60 to send block erase command.

Step 3 Read NDFC_ST[NDFC_CMD_INT_FLAG] to wait for the transfer command end interrupt flag pending, after pending, write 1 to clear the flag.

Step 4 Read NDFC_ST[NDFC_RB_STATE0] to wait for the flash to enter the idle status.

Step 5 Set NDFC_CMD[NDFC_WAIT_FLAG] to 1 to ensure wait RB, set NDFC_CMD[NDFC_SEND_FIRST_CMD] to 1 to send the first command; set NDFC_CMD[NDFC_CMD_LOW_BYTE] to 0xD0 to send erasing command.

Step 6 Read NDFC_ST[NDFC_CMD_INT_FLAG] to wait for the transfer command end interrupt flag pending, after pending, write 1 to clear the flag.

Step 7 Read flash state until flash is ready, configure NDFC_CNT[NDFC_DATA_CNT] to set 1byte transfer data, set NDFC_CMD[NDFC_SEND_FIRST_CMD, NDFC_DATA_TRANS] to 0x3 to send the first command and transfer data. Set NDFC_CMD[NDFC_CMD_LOW_BYTE] to 0x70 to send read status command, read RAM0_BASE to wait ready status.

4.2.4.4 Writing Nand Flash

To write the NAND flash, follow the steps below:

Step 1 Erase the address of the block to be operated.

Step 2 Read NDFC_ST[NDFC_RB_STATE0] to wait for the flash to enter the idle status.

Step 3 Configure RAM0_BASE to write data to RAM0.

Step 4 Configure NDFC_CNT[NDFC_DATA_CNT] to set transferred data;

Set NDFC_CMD[NDFC_SEND_FIRST_CMD] to 1 to send the first command, configure NDFC_CMD[NDFC_DATA_TRANS, NDFC_ACCESS_DIR] to 0x3 to set access direction as writing;

Set NDFC_CMD[NDFC_SEND_ADR] to 1 to enable transfer address, configure NDFC_CMD[NDFC_ADR_NUM] to set the number of the address to be transferred, write the address of the block to be operated in NDFC_ADDR_LOW and NDFC_ADDR_HIGH;

Set NDFC_CMD[NDFC_CMD_LOW_BYTE] to 0x80 to send page program command.

Step 5 Read NDFC_ST[NDFC_RB_STATE0] to wait for the flash to enter the idle status.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 6 Configure NDFC_CMD[NDFC_SEND_FIRST_CMD] to 1 to send the first command, configure NDFC_CMD[NDFC_WAIT_FLAG] to 1 to set wait RB; configure NDFC_CMD[NDFC_CMD_LOW_BYTE] to 0x10 to send end command.

Step 7 Read NDFC_ST[NDFC_CMD_INT_FLAG] to wait for the transfer command end interrupt flag pending, after pending, write 1 to clear the flag.

4.2.4.5 Reading Nand Flash

To read the NAND flash, follow the steps below:

Step 1 Read NDFC_ST[NDFC_RB_STATE0] to wait for the flash to enter the idle status.

Step 2 Configure NDFC_CNT[NDFC_DATA_CNT] to set transferred data;

Step 3 Configure NDFC_CMD[NDFC_SEND_FIRST_CMD] to 1 to send the first command;

Step 4 Configure NDFC_CMD[NDFC_ACCESS_DIR] to 0 to set access direction as reading;

Step 5 Set NDFC_CMD[NDFC_SEND_ADR] to 1 to enable transfer address, configure NDFC_CMD[NDFC_ADR_NUM] to set the number of the address to be transferred, write the address of the block to be operated in NDFC_ADDR_LOW and NDFC_ADDR_HIGH;

Step 6 Set NDFC_CMD[NDFC_CMD_LOW_BYTE] to 0x00 to send page read command.

Step 7 Read NDFC_ST[NDFC_RB_STATE0] to wait for the flash to enter the idle status.

Step 8 Configure NDFC_CMD[NDFC_SEND_FIRST_CMD] to 1 to send the first command, configure NDFC_CMD[NDFC_WAIT_FLAG] to 1 to set wait RB; configure NDFC_CMD[NDFC_CMD_LOW_BYTE] to 0x30 to send end command.

Step 9 Read RAM0_BASE to get data from flash.

Step 10 Read NDFC_ST[NDFC_CMD_INT_FLAG] to wait for the transfer command end interrupt flag pending, after pending, write 1 to clear the flag.

4.2.5 Register List

Module Name	Base Address
NDFC	0x04011000

Register Name	Offset	Description
NDFC_CTL	0x0000	NDFC Configure and Control Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
NDFC_ST	0x0004	NDFC Status Information Register
NDFC_INT	0x0008	NDFC Interrupt Control Register
NDFC_TIMING_CTL	0x000C	NDFC Timing Control Register
NDFC_TIMING_CFG	0x0010	NDFC Timing Configure Register
NDFC_ADDR_LOW	0x0014	NDFC Low Word Address Register
NDFC_ADDR_HIGH	0x0018	NDFC High Word Address Register
NDFC_DATA_BLOCK_MASK	0x001C	NDFC Data Block Mask Register
NDFC_CNT	0x0020	NDFC Data Counter Register
NDFC_CMD	0x0024	NDFC Commands IO Register
NDFC_RCMD_SET	0x0028	Read Command Set Register for Vendor's NAND Memory
NDFC_WCMD_SET	0x002C	Write Command Set Register for Vendor's NAND Memory
NDFC_ECC_CTL	0x0034	NDFC ECC Control Register
NDFC_ECC_ST	0x0038	NDFC ECC Status Register
NDFC_DATA_PAT_STA	0x003C	NDFC Data Pattern Status Register
NDFC_EFR	0x0040	NDFC Enhanced Feature Register
NDFC_RDATA_STA_CTL	0x0044	NDFC Read Data Status Control Register
NDFC_RDATA_STA_0	0x0048	NDFC Read Data Status Register 0
NDFC_RDATA_STA_1	0x004C	NDFC Read Data Status Register 1
NDFC_ERR_CNT_N	0x0050 + 0x0004*N	NDFC Error Counter Register (N from 0 to 7)
NDFC_USER_DATA_LEN_N	0x0070 + 0x0004*N	NDFC User Data Length Register (N from 0 to 3)
NDFC_USER_DATA_N	0x0080 + 0x0004*N	NDFC User Data Field Register N (N from 0 to 31)
NDFC_EFNAND_STA	0x0110	NDFC EFNAND Status Register
NDFC_SPARE_AREA	0x0114	NDFC Spare Area Register
NDFC_PAT_ID	0x0118	NDFC Pattern ID Register
NDFC_DDR2_SPEC_CTL	0x011C	NDFC DDR2 Specific Control Register
NDFC_NDMA_MODE_CTL	0x0120	NDFC Normal DMA Mode Control Register
NDFC_MDMA_DLBA_REG	0x0200	NDFC MBUS DMA Descriptor List Base Address Register
NDFC_MDMA_STA	0x0204	NDFC MBUS DMA Interrupt Status Register
NDFC_DMA_INT_MASK	0x0208	NDFC MBUS DMA Interrupt Enable Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
NDFC_MDMA_CUR_DESC_ADDR	0x020C	NDFC MBUS DMA Current Descriptor Address Register
NDFC_MDMA_CUR_BUF_ADDR	0x0210	NDFC MBUS DMA Current Buffer Address Register
NDFC_DMA_CNT	0x0214	NDFC DMA Byte Counter Register
NDFC_IO_DATA	0x0300	NDFC Input/Output Data Register
RAM0_BASE	0x0400	1024 Bytes RAM0 Base Register
RAM1_BASE	0x0800	1024 Bytes RAM1 Base Register

4.2.6 Register Description

4.2.6.1 0x0000 NDFC Control Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: NDFC_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	/	/	/
28	R/W	0x0	NDFC_DDR_TYPE Type of DDR data interface This bit is valid when NF_TYPE is 0x2 or 0x3. 0: DDR 1: DDR2

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: NDFC_CTL
Bit	Read/Write	Default/Hex	Description
27:24	R/W	0x0	NDFC_CE_SEL Chip Select for NAND Flash Chips 0000: NDFC Select Chip 0 0001: NDFC Select Chip 1 0010: NDFC Select Chip 2 0011: NDFC Select Chip 3 0100: NDFC Select Chip 4 0101: NDFC Select Chip 5 0110: NDFC Select Chip 6 0111: NDFC Select Chip 7 1000: NDFC Select Chip 8 1001: NDFC Select Chip 9 1010: NDFC Select Chip 10 1011: NDFC Select Chip 11 1100: NDFC Select Chip 12 1101: NDFC Select Chip 13 1110: NDFC Select Chip 14 1111: NDFC Select Chip 15
23:22	/	/	/
21	R/W	0x0	NDFC_DDR_RM DDR Repeat Data Mode 0: Lower byte 1: Higher byte
20	R/W	0x0	NDFC_DDR_REN DDR Repeat Enable 0: Disabled 1: Enabled
19:18	R/W	0x0	NF_TYPE NAND Flash Type 00: Normal SDR NAND 01: Reserved 10: ONFI DDR NAND 11: Toggle DDR NAND

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: NDFC_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
17	R/W	0x0	NDFC_CLE_POL NDFC Command Latch Enable (CLE) Signal Polarity Select 0: High active 1: Low active
16	R/W	0x0	NDFC_ALE_POL NDFC Address Latch Enable (ALE) Signal Polarity Select 0: High active 1: Low active
15	R/W	0x0	NDFC_DMA_TYPE 0: Dedicated DMA 1: Normal DMA
14	R/W	0x0	NDFC_RAM_METHOD Access internal RAM method 0: Access internal RAM by AHB method 1: Access internal RAM by DMA method
13:12	/	/	/
11:8	R/W	0x0	NDFC_PAGE_SIZE 000: 1 KB 001: 2 KB 010: 4 KB 011: 8 KB 100: 16 KB 101: 32 KB The page size is for main field data.
7	/	/	/
6	R/W	0x0	NDFC_CE_ACT Chip Select Signal CE# Control during NAND Operation 0: De-active Chip Select Signal NDFC_CE# during data loading, serial access and other no operation stage for power consumption. NDFC automatic controls Chip Select Signals. 1: Chip select signal NDFC_CE# is always active after NDFC is enabled.
5	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: NDFC_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
4:3	R/W	0x0	NDFC_RB_SEL NDFC External R/B Signal Select The value 0-3 selects the external R/B signal. The same R/B signal can be used for multiple chip select flash.
2	R/W	0x0	NDFC_BUS_WIDTH 0: 8-bit bus 1: 16-bit bus
1	R/W1C	0x0	NDFC_RESET NDFC Reset Write 1 to reset NDFC and clear to 0 after reset
0	R/W	0x0	NDFC_EN NDFC Enable Control 0: Disable NDFC 1: Enable NDFC

4.2.6.2 0x0004 NDFC Status Register (Default Value: 0x0000_0F00)

Offset: 0x0004			Register Name: NDFC_ST
Bit	**Read/Write**	**Default/Hex**	**Description**
31:14	/	/	/
13	R	0x0	NDFC_RDATA_STA_0 0: The number of bit 1 during the current read operation is more than the threshold value. 1: The number of bit 1 during the current read operation is less than or equal to the threshold value. This field only is valid when NDFC_RDATA_STA_EN is 1. The threshold value is configured in NDFC_RDATA_STA_TH.
12	R	0x0	NDFC_RDATA_STA_1 0: The number of bit 0 during the current read operation is more than the threshold value. 1: The number of bit 0 during the current read operation is less than or equal to the threshold value. This field only is valid when NDFC_RDATA_STA_EN is 1. The threshold value is configured in NDFC_RDATA_STA_TH.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0004			Register Name: NDFC_ST
Bit	**Read/Write**	**Default/Hex**	**Description**
11	R	0x1	NDFC_RB_STATE3 NAND Flash R/B 3 Line State 0: NAND Flash in BUSY State 1: NAND Flash in READY State
10	R	0x1	NDFC_RB_STATE2 NAND Flash R/B 2 Line State 0: NAND Flash in BUSY State 1: NAND Flash in READY State
9	R	0x1	NDFC_RB_STATE1 NAND Flash R/B 1 Line State 0: NAND Flash in BUSY State 1: NAND Flash in READY State
8	R	0x1	NDFC_RB_STATE0 NAND Flash R/B 0 Line State 0: NAND Flash in BUSY State 1: NAND Flash in READY State
7:5	/	/	/
4	R	0x0	NDFC_STA 0: NDFC FSM in IDLE state 1: NDFC FSM in the BUSY state When NDFC_STA is 0, the NDFC can accept new command and process command.
3	R	0x0	NDFC_CMD_FIFO_STATUS 0: Command FIFO not full and can receive new commands. 1: Full and waiting NDFC to process commands in FIFO. Since there is only one 32-bit FIFO for command. When NDFC latches one command, command FIFO is free and can accept another new command.
2	R/W1C	0x0	NDFC_DMA_INT_FLAG When it is 1, it means that a pending DMA is completed. It will be cleared after writing 1 to this bit or it will be automatically cleared before FSM processing a new command.
1	R/W1C	0x0	NDFC_CMD_INT_FLAG When it is 1, it means that NDFC has finished one Normal Command Mode or one Batch Command Work Mode. It will be cleared after writing 1 to this bit or it will be automatically cleared before FSM processing a new command.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0004			Register Name: NDFC_ST
Bit	Read/Write	Default/Hex	Description
0	R/W1C	0x0	NDFC_RB_B2R When it is 1, it means that NDFC_R/B# signal is transferred from the BUSY state to READY state. It will be cleared after writing 1 to this bit.

4.2.6.3 0x0008 NDFC Interrupt and DMA Enable Register (Default Value: 0x0000_0000)

Offset: 0x0008			Register Name: NDFC_INT
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/
2	R/W	0x0	NDFC_DMA_INT_ENABLE Enable or disable interrupt when a pending DMA is completed.
1	R/W	0x0	NDFC_CMD_INT_ENABLE Enable or disable interrupt when NDFC has finished the procession of a single command in normal command work mode or one batch command work mode. 0: Disabled 1: Enabled
0	R/W	0x0	NDFC_B2R_INT_ENABLE Enable or disable interrupt when NDFC_RB# signal is transferring from the BUSY state to READY state. 0: Disabled 1: Enabled

4.2.6.4 0x000C NDFC Timing Control Register (Default Value: 0x0000_0000)

Offset: 0x000C			Register Name: NDFC_TIMING_CTL
Bit	Read/Write	Default/Hex	Description
31:12	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x000C			Register Name: NDFC_TIMING_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
11:8	R/W	0x0	NDFC_READ_PIPE **In SDR mode:** 0000: Normal 0001: EDO 0010: E-EDO Others: Reserved **In DDR mode:** The bits are used to configure the number of the clock when the data is valid after RE#'s falling edge. Valid values are from 1 to 15.
7:6	/	/	/
5:0	R/W	0x0	NDFC_DC_CTL NDFC Delay Chain Control The bits are only valid in DDR data interface, and configure the relative phase between DQS and DQ[0:7] .

4.2.6.5 0x0010 NDFC Timing Configure Register (Default Value: 0x0000_0095)

Offset: 0x0010			Register Name: NDFC_TIMING_CFG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:20	/	/	/
19:18	R/W	0x0	T_WC Write Cycle Time 00: 1*2T 01: 2*2T 10: 3*2T 11: 4*2T
17:16	R/W	0x0	T_CCS Change Column Setup Time 00: 12*2T 01: 20*2T 10: 28*2T 11: 60*2T

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0010			Register Name: NDFC_TIMING_CFG
Bit	**Read/Write**	**Default/Hex**	**Description**
15:14	R/W	0x0	T_CLHZ CLE High to Output Hi-z 00: 2*2T 01: 8*2T 10: 16*2T 11: 31*2T
13:12	R/W	0x0	T_CS CE Setup Time 00: 2*2T 01: 8*2T 10: 16*2T 11: 31*2T
11	R/W	0x0	T_CDQSS DQS Setup Time for Data Input Start 0: 4*2T 1: 20*2T
10:8	R/W	0x0	T_CAD Command, Address, Data Delay 000: 2*2T 001: 6*2T 010: 10*2T 011: 14*2T 100: 22*2T 101: 30*2T 110/111: 62*2T
7:6	R/W	0x2	T_RHW Cycle Number from RE# High to WE# Low 00: 4*2T 01: 12*2T 10: 20*2T 11: 28*2T

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0010			Register Name: NDFC_TIMING_CFG
Bit	**Read/Write**	**Default/Hex**	**Description**
5:4	R/W	0x1	T_WHR Cycle Number from WE# High to RE# Low 00: 0*2T 01: 6*2T 10: 14*2T 11: 22*2T
3:2	R/W	0x1	T_ADL Cycle Number from Address to Data Loading 00: 0*2T 01: 6*2T 10: 14*2T 11: 22*2T
1:0	R/W	0x1	T_WB Cycle Number from WE# High to Busy 00: 14*2T 01: 22*2T 10: 30*2T 11: 38*2T

4.2.6.6 0x0014 NDFC Address Low Word Register (Default Value: 0x0000_0000)

Offset: 0x0014			Register Name: NDFC_ADDR_LOW
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	R/W	0x0	ADDR_DATA4 NAND Flash 4th Cycle Address Data
23:16	R/W	0x0	ADDR_DATA3 NAND Flash 3rd Cycle Address Data
15:8	R/W	0x0	ADDR_DATA2 NAND Flash 2nd Cycle Address Data
7:0	R/W	0x0	ADDR_DATA1 NAND Flash 1st Cycle Address Data

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.2.6.7 0x0018 NDFC Address High Word Register (Default Value: 0x0000_0000)

Offset: 0x0018			Register Name: NDFC_ADDR_HIGH
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	R/W	0x0	ADDR_DATA8 NAND Flash 8th Cycle Address Data
23:16	R/W	0x0	ADDR_DATA7 NAND Flash 7th Cycle Address Data
15:8	R/W	0x0	ADDR_DATA6 NAND Flash 6th Cycle Address Data
7:0	R/W	0x0	ADDR_DATA5 NAND Flash 5th Cycle Address Data

4.2.6.8 0x001C NDFC Data Block Mask Register (Default Value: 0x0000_0000)

Offset: 0x001C			Register Name: NDFC_DATA_BLOCK_MASK
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 31 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
30	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 30 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
29	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 29 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x001C			Register Name: NDFC_DATA_BLOCK_MASK
Bit	**Read/Write**	**Default/Hex**	**Description**
28	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 28 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
27	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 27 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
26	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 26 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
25	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 25 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
24	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 24 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x001C			Register Name: NDFC_DATA_BLOCK_MASK
Bit	**Read/Write**	**Default/Hex**	**Description**
23	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 23 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
22	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 22 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
21	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 21 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
20	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 20 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
19	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 19 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x001C			Register Name: NDFC_DATA_BLOCK_MASK
Bit	**Read/Write**	**Default/Hex**	**Description**
18	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 18 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
17	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 17 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
16	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 16 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
15	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 15 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
14	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 14 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x001C			Register Name: NDFC_DATA_BLOCK_MASK
Bit	**Read/Write**	**Default/Hex**	**Description**
13	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 13 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
12	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 12 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
11	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 11 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
10	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 10 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
9	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 9 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x001C			Register Name: NDFC_DATA_BLOCK_MASK
Bit	**Read/Write**	**Default/Hex**	**Description**
8	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 8 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
7	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 7 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
6	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 6 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
5	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 5 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
4	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 4 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x001C			Register Name: NDFC_DATA_BLOCK_MASK
Bit	**Read/Write**	**Default/Hex**	**Description**
3	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 3 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
2	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 2 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
1	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 1 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.
0	R/W	0x0	NDFC_DATA_BLOCK_MASK It is used to indicate the data block 0 should be written or read during batch command procession (NDFC_CMD_TYPE = 0x10 in NDFC_CMD). 0: Disabled 1: Enabled 1 data block = 1024 bytes main field data.

4.2.6.9 0x0020 NDFC Data Counter Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: NDFC_CNT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:10	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0020			Register Name: NDFC_CNT
Bit	**Read/Write**	**Default/Hex**	**Description**
9:0	R/W	0x0	NDFC_DATA_CNT Transfer Data Byte Counter The length can be set from 1 byte to 1024 bytes. However, 1024 bytes is set when it is zero.

4.2.6.10 0x0024 NDFC Command IO Register (Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: NDFC_CMD
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	R/W	0x0	NDFC_CMD_TYPE 00: Common command for normal operation 01: Special command for Flash spare field operation 10: Page command for batch process operation 11: Reserved
29	R/W	0x0	NDFC_SEND_FOURTH_CMD 0: Do not send fourth set command 1: Send it on the external memory's bus It is used for the EF-NAND page read.
28	R/W	0x0	NDFC_SEND_THIRD_CMD 0: Do not send third set command 1: Send it on the external memory's bus It is used for the EF-NAND page read.
27	R/W	0x0	NDFC_SEND_RANDOM_CMD2_CTL 0: Do not send random cmd2 (NDFC_RANDOM_CMD2) 1: Send random cmd2 Note: It is only valid in batch cmd operation and writing operation.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0024			Register Name: NDFC_CMD
Bit	**Read/Write**	**Default/Hex**	**Description**
26	R/W	0x0	NDFC_DATA_METHOD Data swap method when the internal RAM and system memory It is only active for common command and special command. 0: No action 1: DMA transfer automatically It only is active when NDFC_RAM_METHOD is 1. If this bit is set to 1, NDFC should set up DRQ to fetch data before output to Flash, or NDFC should set up DRQ to send to system memory after fetching data from Flash. If this bit is set to 0, NDFC output the data in internal RAM or do nothing after fetching data from Flash.
25	R/W	0x0	NDFC_SEQ User data & BCH check word position. It only is active for Page Command. Do not care about this bit for the other two commands. 0: Interleave Method (on page spare area) 1: Sequence Method (following data block)
24	R/W	0x0	NDFC_SEND_SECOND_CMD 0: Do not send second set command 1: Send it on the external memory's bus
23	R/W	0x0	NDFC_WAIT_FLAG 0: NDFC can transfer data regardless of the internal NDFC_RB wire 1: NDFC can transfer data when the internal NDFC_RB wire is READY; otherwise it cannot when the internal NDFC_RB wire is BUSY.
22	R/W	0x0	NDFC_SEND_FIRST_CMD 0: Do not send first set command 1: Send it on the external memory's bus
21	R/W	0x0	NDFC_DATA_TRANS 0: No data transfer on external memory bus. 1: Data transfer and direction is decided by the field. NDFC_ACCESS_DIR
20	R/W	0x0	NDFC_ACCESS_DIR 0: Read NAND Flash 1: Write NAND Flash

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0024			Register Name: NDFC_CMD
Bit	**Read/Write**	**Default/Hex**	**Description**
19	R/W	0x0	NDFC_SEND_ADR 0: Do not send ADDRESS. 1: Send N cycles ADDRESS, the number N is specified by NDFC_ADR_NUM field.
18:16	R/W	0x0	NDFC_ADR_NUM Address Cycles' Number 000: 1 cycle address field 001: 2 cycles address field 010: 3 cycles address field 011: 4 cycles address field 100: 5 cycles address field 101: 6 cycles address field 110: 7 cycles address field 111: 8 cycles address field
15:10	/	/	/
9:8	R/W	0x0	NDFC_ADR_NUM_IN_PAGE_CMD The number of address cycles during page command. 00: 2 address cycles 11: 5 address cycles Others: reserved
7:0	R/W	0x0	NDFC_CMD_LOW_BYTE NDFC command low byte data This command will be sent to external Flash by NDFC.

4.2.6.11 0x0028 NDFC Command Set Register 0 (Default Value: 0x00E0_0530)

Offset: 0x0028			Register Name: NDFC_CMD_SET0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	R/W	0x00	NDFC_RANDOM_CMD2 Used for Batch Operation
23:16	R/W	0xE0	NDFC_RANDOM_READ_CMD1 Used for Batch Read Operation
15:8	R/W	0x05	NDFC_RANDOM_READ_CMD0 Used for Batch Read Operation

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0028			Register Name: NDFC_CMD_SET0
Bit	Read/Write	Default/Hex	Description
7:0	R/W	0x30	NDFC_READ_CMD Used for Batch Read Operation

4.2.6.12 0x002C NDFC Command Set Register 1 (Default Value: 0x7000_8510)

Offset: 0x002C			Register Name: NDFC_CMD_SET1
Bit	Read/Write	Default/Hex	Description
31:24	R/W	0x70	NDFC_READ_CMD0 Used for EF-NAND Page Read Operation
23:16	R/W	0x00	NDFC_READ_CMD1 Used for EF-NAND Page Read Operation
15:8	R/W	0x85	NDFC_RANDOM_WRITE_CMD Used for Batch Write Operation
7:0	R/W	0x10	NDFC_PROGRAM_CMD Used for Batch Write Operation

4.2.6.13 0x0034 NDFC ECC Control Register (Default Value: 0x4A80_0008)

Offset: 0x0034			Register Name: NDFC_ECC_CTL
Bit	Read/Write	Default/Hex	Read/Write
31	/	/	/
30:16	R/W	0x4a80	NDFC_RANDOM_SEED The seed value for randomize engine. It is only active when NDFC_RANDOM_EN is set to '1'.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0034			Register Name: NDFC_ECC_CTL
Bit	Read/Write	Default/Hex	Read/Write
15:8	R/W	0x0	NDFC_ECC_MODE 00000000: BCH-16 00000001: BCH-24 00000010: BCH-28 00000011: BCH-32 00000100: BCH-40 00000101: BCH-44 00000110: BCH-48 00000111: BCH-52 00001000: BCH-56 00001001: BCH-60 00001010: BCH-64 00001011: BCH-68 00001100: BCH-72 00001101: BCH-76 00001110: BCH-80 Others: Reserved
7	R/W	0x0	NDFC_RANDOM_SIZE 0: ECC block size 1: Page size
6	R/W	0x0	NDFC_RANDOM_DIRECTION 0: LSB first 1: MSB first
5	R/W	0x0	NDFC_RANDOM_EN 0: Disable Data Randomize 1: Enable Data Randomize
4	R/W	0x0	NDFC_ECC_EXCEPTION 0: Normal ECC 1: For ECC, there is an exception. If all data is 0xff or 0x00 for the block. When reading this page, ECC assumes that it is right. In this case, no error information is reported. Note: It is only active when ECC is ON.
3	R/W	0x1	NDFC_ECC_PIPELINE Pipeline function enable or disable for batch command 0: Error Correction function no pipeline with next block operation 1: Error Correction pipeline
2:1	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0034			Register Name: NDFC_ECC_CTL
Bit	Read/Write	Default/Hex	Read/Write
0	R/W	0x0	NDFC_ECC_EN 0: ECC is OFF 1: ECC is ON

4.2.6.14 0x0038 NDFC ECC Status Register (Default Value: 0x0000_0000)

Offset: 0x0038			Register Name: NDFC_ECC_ST
Bit	Read/Write	Default/Hex	Description
31	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 31 0: ECC can correct these error bits or there is no error bit. 1: Error bits number beyond of ECC correction capability and cannot correct them. The bit[31] of this register is corresponding to the 31st ECC data block. 1 ECC Data Block = 1024 bytes.
30	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 30 0: ECC can correct these error bits or there is no error bit. 1: Error bits number beyond of ECC correction capability and cannot correct them. The bit[30] of this register is corresponding to the 30th ECC data block. 1 ECC Data Block = 1024 bytes.
29	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 29 0: ECC can correct these error bits or there is no error bit. 1: Error bits number beyond of ECC correction capability and cannot correct them. The bit[29] of this register is corresponding to the 29th ECC data block. 1 ECC Data Block = 1024 bytes.
28	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 28 0: ECC can correct these error bits or there is no error bit. 1: Error bits number beyond of ECC correction capability and cannot correct them. The bit[28] of this register is corresponding to the 28th ECC data block. 1 ECC Data Block = 1024 bytes.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0038			Register Name: NDFC_ECC_ST
Bit	**Read/Write**	**Default/Hex**	**Description**
27	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 27 0: ECC can correct these error bits or there is no error bit. 1: Error bits number beyond of ECC correction capability and cannot correct them. The bit[27] of this register is corresponding to the 27th ECC data block. 1 ECC Data Block = 1024 bytes.
26	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 26 0: ECC can correct these error bits or there is no error bit. 1: Error bits number beyond of ECC correction capability and cannot correct them. The bit[26] of this register is corresponding to the 26th ECC data block. 1 ECC Data Block = 1024 bytes.
25	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 25 0: ECC can correct these error bits or there is no error bit 1: Error bits number beyond of ECC correction capability and cannot correct them The bit[25] of this register is corresponding to the 25th ECC data block. 1 ECC Data Block = 1024 bytes.
24	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 24 0: ECC can correct these error bits or there is no error bit. 1: Error bits number beyond of ECC correction capability and cannot correct them. The bit[24] of this register is corresponding to the 24th ECC data block. 1 ECC Data Block = 1024 bytes.
23	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 23 0: ECC can correct these error bits or there is no error bit. 1: Error bits number beyond of ECC correction capability and cannot correct them. The bit[23] of this register is corresponding to the 23rd ECC data block. 1 ECC Data Block = 1024 bytes.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0038			Register Name: NDFC_ECC_ST
Bit	**Read/Write**	**Default/Hex**	**Description**
22	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 22 0: ECC can correct these error bits or there is no error bit. 1: Error bits number beyond of ECC correction capability and cannot correct them. The bit[22] of this register is corresponding to the 22nd ECC data block. 1 ECC Data Block = 1024 bytes.
21	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 21 0: ECC can correct these error bits or there is no error bit. 1: Error bits number beyond of ECC correction capability and cannot correct them. The bit[21] of this register is corresponding to the 21st ECC data block. 1 ECC Data Block = 1024 bytes.
20	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 20 0: ECC can correct these error bits or there is no error bit. 1: Error bits number beyond of ECC correction capability and cannot correct them. The bit[20] of this register is corresponding to the 20th ECC data block. 1 ECC Data Block = 1024 bytes.
19	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 19 0: ECC can correct these error bits or there is no error bit. 1: Error bits number beyond of ECC correction capability and cannot correct them. The bit[19] of this register is corresponding to the 19th ECC data block. 1 ECC Data Block = 1024 bytes.
18	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 18 0: ECC can correct these error bits or there is no error bit. 1: Error bits number beyond of ECC correction capability and cannot correct them. The bit[18] of this register is corresponding to the 18th ECC data block. 1 ECC Data Block = 1024 bytes.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0038			Register Name: NDFC_ECC_ST
Bit	**Read/Write**	**Default/Hex**	**Description**
17	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 17 0: ECC can correct these error bits or there is no error bit. 1: Error bits number beyond of ECC correction capability and cannot correct them. The bit[17] of this register is corresponding to the 17th ECC data block. 1 ECC Data Block = 1024 bytes.
16	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 16 0: ECC can correct these error bits or there is no error bit. 1: Error bits number beyond of ECC correction capability and cannot correct them. The bit[16] of this register is corresponding to the 16th ECC data block. 1 ECC Data Block = 1024 bytes.
15	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 15 0: ECC can correct these error bits or there is no error bit. 1: Error bits number beyond of ECC correction capability and cannot correct them. The bit[15] of this register is corresponding to the 15th ECC data block. 1 ECC Data Block = 1024 bytes.
14	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 14 0: ECC can correct these error bits or there is no error bit. 1: Error bits number beyond of ECC correction capability and cannot correct them. The bit[14] of this register is corresponding to the 14th ECC data block. 1 ECC Data Block = 1024 bytes.
13	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 13 0: ECC can correct these error bits or there is no error bit. 1: Error bits number beyond of ECC correction capability and cannot correct them. The bit[13] of this register is corresponding to the 13th ECC data block. 1 ECC Data Block = 1024 bytes.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0038			Register Name: NDFC_ECC_ST
Bit	**Read/Write**	**Default/Hex**	**Description**
12	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 12 0: ECC can correct these error bits or there is no error bit 1: Error bits number beyond of ECC correction capability and cannot correct them The bit[12] of this register is corresponding to the 12th ECC data block. 1 ECC Data Block = 1024 bytes.
11	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 11 0: ECC can correct these error bits or there is no error bit 1: Error bits number beyond of ECC correction capability and cannot correct them The bit[11] of this register is corresponding to the 11th ECC data block. 1 ECC Data Block = 1024 bytes.
10	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 10 0: ECC can correct these error bits or there is no error bit 1: Error bits number beyond of ECC correction capability and cannot correct them The bit[10] of this register is corresponding to the 10th ECC data block. 1 ECC Data Block = 1024 bytes.
9	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 9 0: ECC can correct these error bits or there is no error bit 1: Error bits number beyond of ECC correction capability and cannot correct them The bit[9] of this register is corresponding to the 9th ECC data block. 1 ECC Data Block = 1024 bytes.
8	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 8 0: ECC can correct these error bits or there is no error bit 1: Error bits number beyond of ECC correction capability and cannot correct them The bit[8] of this register is corresponding to the 8th ECC data block. 1 ECC Data Block = 1024 bytes.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0038			Register Name: NDFC_ECC_ST
Bit	**Read/Write**	**Default/Hex**	**Description**
7	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 7 0: ECC can correct these error bits or there is no error bit 1: Error bits number beyond of ECC correction capability and cannot correct them The bit[7] of this register is corresponding to the 7th ECC data block. 1 ECC Data Block = 1024 bytes.
6	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 6 0: ECC can correct these error bits or there is no error bit 1: Error bits number beyond of ECC correction capability and cannot correct them The bit[6] of this register is corresponding to the 6th ECC data block. 1 ECC Data Block = 1024 bytes.
5	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 5 0: ECC can correct these error bits or there is no error bit 1: Error bits number beyond of ECC correction capability and cannot correct them The bit[5] of this register is corresponding to the 5th ECC data block. 1 ECC Data Block = 1024 bytes.
4	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 4 0: ECC can correct these error bits or there is no error bit 1: Error bits number beyond of ECC correction capability and cannot correct them The bit[4] of this register is corresponding to the 4th ECC data block. 1 ECC Data Block = 1024 bytes.
3	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 3 0: ECC can correct these error bits or there is no error bit 1: Error bits number beyond of ECC correction capability and cannot correct them The bit[3] of this register is corresponding to the 3rd ECC data block. 1 ECC Data Block = 1024 bytes.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0038			Register Name: NDFC_ECC_ST
Bit	**Read/Write**	**Default/Hex**	**Description**
2	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 2 0: ECC can correct these error bits or there is no error bit 1: Error bits number beyond of ECC correction capability and cannot correct them The bit[2] of this register is corresponding to the 2nd ECC data block. 1 ECC Data Block = 1024 bytes.
1	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 1 0: ECC can correct these error bits or there is no error bit 1: Error bits number beyond of ECC correction capability and cannot correct them The bit[1] of this register is corresponding to the 1st ECC data block. 1 ECC Data Block = 1024 bytes.
0	R	0x0	NDFC_ECC_ERR Error information bit of Data Block 0 0: ECC can correct these error bits or there is no error bit 1: Error bits number beyond of ECC correction capability and cannot correct them The bit[0] of this register is corresponding to the 0 ECC data block. 1 ECC Data Block = 1024 bytes.

4.2.6.15 0x003C NDFC Data Pattern Status Register (Default Value: 0x0000_0000)

Offset: 0x003C			Register Name: NDFC_DATA_PAT_STA
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 31 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x003C			Register Name: NDFC_DATA_PAT_STA
Bit	**Read/Write**	**Default/Hex**	**Description**
30	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 30 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
29	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 29 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
28	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 28 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
27	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 27 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
26	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 26 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
25	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 25 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x003C			Register Name: NDFC_DATA_PAT_STA
Bit	**Read/Write**	**Default/Hex**	**Description**
24	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 24 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
23	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 23 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
22	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 22 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
21	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 21 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
20	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 20 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
19	R	0x0	Special pattern (all 0x00 or all 0xff) found Flag for Data Block 19 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x003C			Register Name: NDFC_DATA_PAT_STA
Bit	**Read/Write**	**Default/Hex**	**Description**
18	R	0x0	Special pattern (all 0x00 or all 0xff) Found flag for Data Block 18 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
17	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 17 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
16	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 16 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
15	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 15 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
14	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 14 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
13	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 13 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x003C			Register Name: NDFC_DATA_PAT_STA
Bit	**Read/Write**	**Default/Hex**	**Description**
12	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 12 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
11	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 11 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
10	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 10 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
9	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 9 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
8	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 8 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
7	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 7 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x003C			Register Name: NDFC_DATA_PAT_STA
Bit	**Read/Write**	**Default/Hex**	**Description**
6	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 6 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
5	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 5 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
4	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 4 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
3	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 3 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
2	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 2 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.
1	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 1 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x003C			Register Name: NDFC_DATA_PAT_STA
Bit	Read/Write	Default/Hex	Description
0	R	0x0	Special pattern (all 0x00 or all 0xff) found flag for Data Block 0 when read from external NAND flash. 0: No found 1: Special pattern is found The register of NDFC_PAT_ID would indicate which kind of pattern is found.

4.2.6.16 0x0040 NDFC Enhanced Feature Register (Default Value: 0x0000_0000)

Offset: 0x0040			Register Name: NDFC_EFR
Bit	Read/Write	Default/Hex	Description
31:25	/	/	/
24	R/W	0x0	DB_CNT_EN Dummy_Byte_Count_EN 0: Disable fill Dummy Byte 1: Enable fill Dummy Byte
23:16	R/W	0x0	DB_CNT Dummy_Byte_Count After PAGE CMD operation finishing sending out the main data, user data, and ECC code, the controller would send a dummy byte to fill the unused space in one page. Note: It is only valid in PAGE CMD operation(NDFC_CMD_TYPE = 0x3), and this function is disabled when Dummy_Byte_Count_EN is 0. If the NDFC_RANDOM_EN = 0x0, the value of the dummy byte is 0, so in order to improve the stability, when using this function, it is better to set the NDFC_RANDOM_EN to 0x1.
15:9	/	/	/
8	R/W	0x0	NDFC_WP_CTRL NAND Flash Write Protect Control Bit 0: Write Protect is active 1: Write Protect is not active When this bit is '0', the WP signal line is low level and the external NAND flash is in the protected state.
7	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0040			Register Name: NDFC_EFR
Bit	Read/Write	Default/Hex	Description
6:0	R/W	0x0	NDFC_ECC_DEBUG For the purpose of debugging the ECC engine, the special error bits are inserted before writing external Flash Memory. 0: No error is inserted (ECC Normal Operation). n: N bits of error is inserted.

4.2.6.17 0x0044 NDFC Read Data Status Control Register (Default Value: 0x0100_0000)

Offset: 0x0044			Register Name: NDFC_RDATA_STA_CTL
Bit	Read/Write	Default/Hex	Description
31:25	/	/	/
24	R/W	0x1	NDFC_RDATA_STA_EN 0: Disable to count the number of bit 1 and bit 0 during the current read operation. 1: Enable to count the number of bit 1 and bit 0 during the current read operation. The number of bit 1 and bit 0 during the current read operation can be used to check whether a page is blank or bad.
23:19	/	/	/
18:0	R/W	0x0	NDFC_RDATA_STA_TH The threshold value to generate data status If the number of bit 1 during the current read operation is less than or equal to the threshold value, the bit[13] of NDFC_ST register will be set. If the number of bit 0 during the current read operation is less than or equal to the threshold value, the bit[12] of NDFC_ST register will be set.

4.2.6.18 0x0048 NDFC Read Data Status Register 0 (Default Value: 0x0000_0000)

Offset: 0x0048			Register Name: NDFC_RDATA_STA_0
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	BIT_CNT_1 The number of input bit 1 during the current command. It will be cleared automatically when the next command is executed.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.2.6.19 0x004C NDFC Read Data Status Register 1 (Default Value: 0x0000_0000)

Offset: 0x004C			Register Name: NDFC_RDATA_STA_1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	BIT_CNT_0 The number of inputs bit 0 during the current command. It will be cleared automatically when the next command is executed.

4.2.6.20 0x0050 + N*0x0004 NDFC Error Counter Register N (Default Value: 0x0000_0000)

Offset: 0x0050 + N*0x04 (N = 0 to 7)			Register Name: NDFC_ERR_CNT_N
Bit	**Read/Write**	**Default/Hex**	**Description**
[8M + 7: 8M] (M = 0 to 3)	R	0x0	ECC_COR_NUM ECC Corrected Bits Number for ECC Data Block[N*0x04+M] 00000000: No corrected bits 00000001: 1 corrected bit 00000010: 2 corrected bits ... 01010000: 80 corrected bits Others: Reserved Note: 1 ECC Data Block =1024 bytes

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.2.6.21 0x0070 + N*0x0004 NDFC User Data Length Register N (Default Value: 0x0000_0000)

Offset: 0x0070 + N*0x0004 (N = 0 to 3)			Register Name: NDFC_USER_DATA_LEN_N
Bit	**Read/Write**	**Default/Hex**	**Description**
[4M + 3: 4M] (M = 0 to 7)	R/W	0x0	It is used to indicate the length of the user data for ECC DATA BLOCK[0x08*N + M]. 0000: no user data 0001: 4 bytes of user data 0010: 8 bytes of user data 0011: 12 bytes of user data 0100: 16 bytes of user data 0101: 20 bytes of user data 0110: 24 bytes of user data 0111: 28 bytes of user data 1000: 32 bytes of user data Others: reserved

4.2.6.22 0x0080 + N*0x0004 NDFC User Data Register N (Default Value: 0xFFFF_FFFF)

Offset: 0x0080 + N*0x0004 (N = 0 to 31)			Register Name: NDFC_USER_DATA_N
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0xffffffff	USER_DATA All of the user data in one page is stored in NDFC_USER_DATA_N. The start register address of each ECC DATA BLOCK's user data is determined by its length configured in NDFC_USER_DATA_LEN_N. For example: ECC DATA BLOCK[0] user data len = 8 Bytes, address = 0x80 ECC DATA BLOCK[1] user data len = 0 Bytes, ECC DATA BLOCK[2] user data len = 4 Bytes, address = 0x80 + 8 ECC DATA BLOCK[3] user data len = 4 Bytes, address = 0x80 + 8 + 4 ECC DATA BLOCK[4] user data len = 0 Bytes ECC DATA BLOCK[5] user data len = 16 Bytes, address = 0x80 + 8 + 4 + 4 ECC DATA BLOCK[6] user data len = 0 Bytes ECC DATA BLOCK[7] user data len = 0 Bytes

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.2.6.23 0x110 NDFC EFNAND Status Register (Default Value: 0x0000_0000)

Offset: 0x0110			Register Name: NDFC_EFNAND_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R	0x0	EF_NAND_STATUS The status value for EF-NAND page read operation

4.2.6.24 0x0114 NDFC Spare Area Register (Default Value: 0x0000_0400)

Offset: 0x0114			Register Name: NDFC_SPARE_AREA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x400	NDFC_SPARE_ADR This value indicates the spare area first byte address for NDFC interleave page operation.

4.2.6.25 0x0118 NDFC Pattern ID Register (Default Value: 0x0000_0000)

Offset: 0x0118			Register Name: NDFC_PAT_ID
Bit	**Read/Write**	**Default/Hex**	**Description**
n (N = 0 to 31)	R	0x0	PAT_ID Special Pattern ID for ECC data block[n] 0: All 0x00 is found 1: All 0xFF is found

4.2.6.26 0x011C NDFC DDR2 Specific Control Register (Default Value: 0x0000_0000)

Offset: 0x011C			Register Name: NDFC_DDR2_SPEC_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x011C			Register Name: NDFC_DDR2_SPEC_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
15:12	R/W	0x0	DLEN_WR The number of latency DQS cycle for a write 0000: No latency 0001: One latency DQS cycle 0010: Two latency DQS cycle 0011: Four latency DQS cycle
11:8	R/W	0x0	DLEN_RD The number of latency DQS cycle for a read 0000: No latency 0001: One latency DQS cycle 0010: Two latency DQS cycle 0011: Four latency DQS cycle
7:3	/	/	/
2	R/W	0x0	EN_RE_C Enable the complementary RE# signal 0: Disabled 1: Enabled
1	R/W	0x0	EN_DQS_C Enable the complementary DQS signal 0: Disabled 1: Enabled
0	/	/	/

4.2.6.27 0x0120 NDFC Normal DMA Mode Control Register (Default Value: 0x0000_00E5)

Offset: 0x0120			Register Name: NDFC_NDMA_MODE_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:6	R/W	0x11	DMA_ACT_STA 00: dma_active is low 01: dma_active is high 10: dma_active is controlled by dma_request (DRQ) 11: dma_active is controlled by controller

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0120			Register Name: NDFC_NDMA_MODE_CTL
Bit	Read/Write	Default/Hex	Description
5	R/W	0x1	DMA_ACK_EN 0: active fall do not care ack 1: active fall must after detect ack is high
4:0	R/W	0x05	DELAY_CYCLE The counts of hold cycles from DMA last signal high to dma_active high

4.2.6.28 0x0200 NDFC MBUS DMA Descriptor List Base Address Register (Default Value: 0x0000_0000)

Offset: 0x0200			Register Name: NDFC_MDMA_DLBA_REG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	NFC_MDMA_DESC_BASE_ADDR Start Address 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 DMA internally. Hence these LSB bits are read-only.

4.2.6.29 0x0204 NDFC MBUS DMA Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0204			Register Name: NDFC_MDMA_STA
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	NFC_MDMA_TRANS_FINISH_INT Transfer Finish Interrupt Indicates that data transmission is finished for a descriptor. Writing a '1' clears this bit. Bit 0: Corresponding DMA descriptor 0 Bit 1: Corresponding DMA descriptor 1 … Bit 31: Corresponding DMA descriptor 31

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.2.6.30 0x0208 NDFC MBUS DMA Interrupt Enable Register (Default Value: 0x0000_0000)

Offset: 0x0208			Register Name: NDFC_DMA_INT_MASK
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	NFC_MDMA_TRANS_INT_ENB Transfer Interrupt Enable When set, the Transmit Interrupt is enabled. When reset, the Transmit Interrupt is disabled. Bit 0: Corresponding DMA descriptor 0 Bit 1: Corresponding DMA descriptor 1 … Bit 31: Corresponding DMA descriptor 31

4.2.6.31 0x020C NDFC MBUS DMA Current Descriptor Address Register (Default Value: 0x0000_0000)

Offset: 0x020C			Register Name: NDFC_MDMA_CUR_DESC_ADDR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	CUR_DESC_ADDR Current Descriptor Address Pointer Cleared on reset. The pointer is updated by DMA during operation. This register points to the start address of the current descriptor read by the DMA.

4.2.6.32 0x0210 NDFC MBUS DMA Current Buffer Address Register (Default Value: 0x0000_0000)

Offset: 0x0210			Register Name: NDFC_MDMA_CUR_BUF_ADDR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	CUR_BUFF_ADDR Current Buffer Address Pointer Cleared on Reset. The pointer is updated by DMA during operation. This register points to the current Data Buffer Address accessed by the DMA.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.2.6.33 0x0214 NDFC DMA Byte Counter Register (Default Value: 0x0000_0000)

Offset: 0x0214			Register Name: NDFC_DMA_CNT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x0	DMA_CNT DMA data counter, including MBUS DMA and Normal DMA

4.2.6.34 0x0300 NDFC IO Data Register (Default Value: 0x0000_0000)

Offset: 0x0300			Register Name: NDFC_IO_DATA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	NDFC_IO_DATA Read/Write data into internal RAM Access unit is 32-bit.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.3 SD/MMC Host Controller (SMHC)

4.3.1 Overview

The SMHC controls the read/write operations on the secure digital (SD) cards, multimedia cards (MMC), and various extended devices that is based on the secure digital input/output (SDIO) protocol. The processor provides three SMHC interfaces for controlling the SD cards, MMCs, and SDIO devices.

The SMHC has the following features:

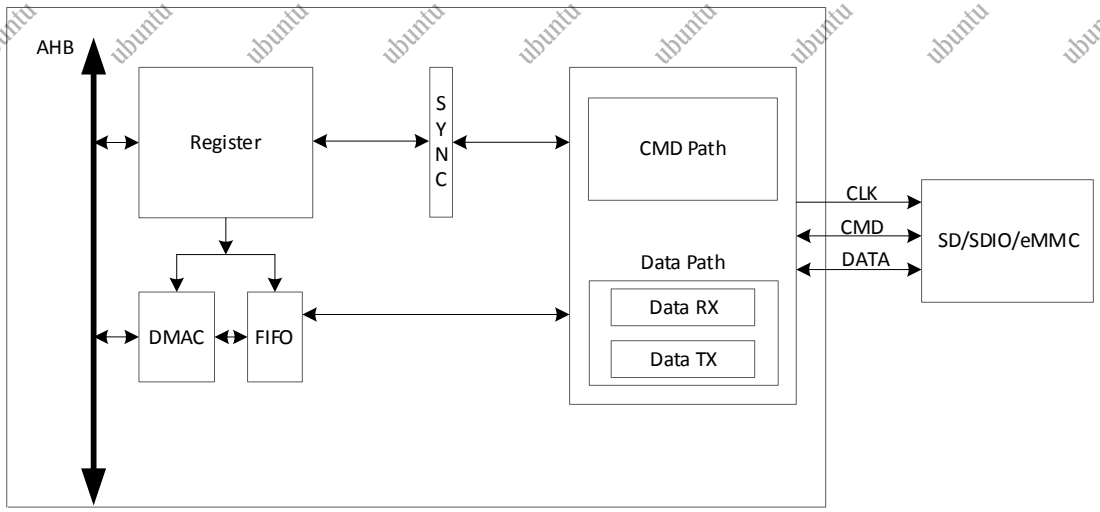
- Supports Command Completion signals and interrupts to host processor, and Command Completion signal disable feature
- The SMHC0 controls the devices that comply with the following protocols:
  - Secure Digital Memory (SD mem-version 3.0)
  - Multimedia Card (eMMC-version 5.1)
- The SMHC1 controls the device that complies with the protocol Secure Digital I/O (SDIO-version 3.0)
- Maximum performance:
  - SDR mode: 150 MHz@1.8 V IO pad
  - DDR mode: 50 MHz@1.8 V IO pad
  - DDR mode: 50 MHz@3.3 V IO pad
- Supports 1-bit or 4-bit data width
- Hardware CRC generation and error detection
- Programmable baud rate
- Host pull-up control
- Supports SDIO interrupt in 1-bit and 4-bit modes
- Block size of 1 to 65535 bytes
- Descriptor-based internal DMA controller
- Internal 1 KB FIFO for data transfer

4.3.2 Block Diagram

The following figure shows a block diagram of the SMHC.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 4-17 SMHC Block Diagram



SMHC contains the following sub-blocks:

Table 4-9 SMHC Sub-blocks

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

4.3.3 Functional Descriptions

4.3.3.1 External Signals

The following table describes the external signals of SMHC.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Table 4-10 SMHC External Signals

Port Name	Type	Description
SDC0-CMD	I/O, OD	Command Signal for SD Card or eMMC
SDC0-CLK	O	Clock for SD Card or eMMC
SDC0-D[3:0]	I/O	Data Input and Output for SD Card or eMMC
SDC0-RST	O	Reset for SD Card or eMMC
SDC1-CMD	I/O, OD	Command Signal for SDIO Wi-Fi
SDC1-CLK	O	Clock for SDIO Wi-Fi
SDC1-D[3:0]	I/O	Data Input and Output for SDIO Wi-Fi

4.3.3.2 Clock Sources

Each SMHC gets 4 different clocks. You can select one of them as the SMHC clock source. The following table describes the clock sources of the SMHC.

For clock setting, configurations, and gating information, refer to section 3.3 "CCU".

Table 4-11 SMHC Clock Sources

Clock Sources	Description
DCXO24M	24 MHz Crystal
PLL_PERI(1X)	Peripheral Clock, the default value is 600 MHz
PLL_PERI(2X)	Peripheral Clock, the default value is 1.2 GHz
PLL_AUDIO0(DIV2)	Audio clock, the default value is 1536 MHz

4.3.3.3 Timing Diagram

Refer to the following relative specifications:

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.3.3.4 Data Path

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

Reading Data from the SD/SDIO/eMMC

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

Writing Data to the SD/SDIO/eMMC

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

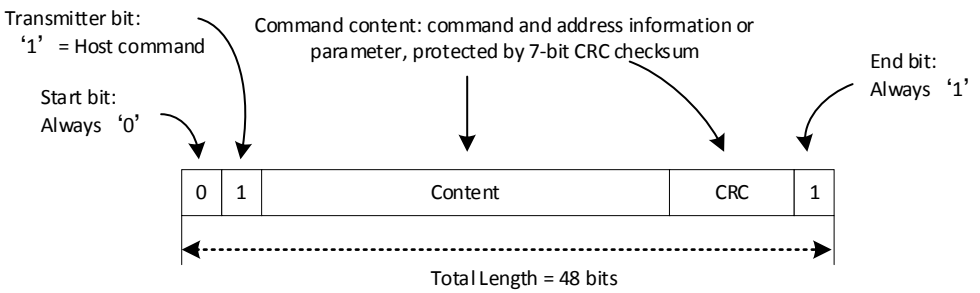
4.3.3.5 Package Format

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

Command Tokens

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

Figure 4-18 Command Token Format



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

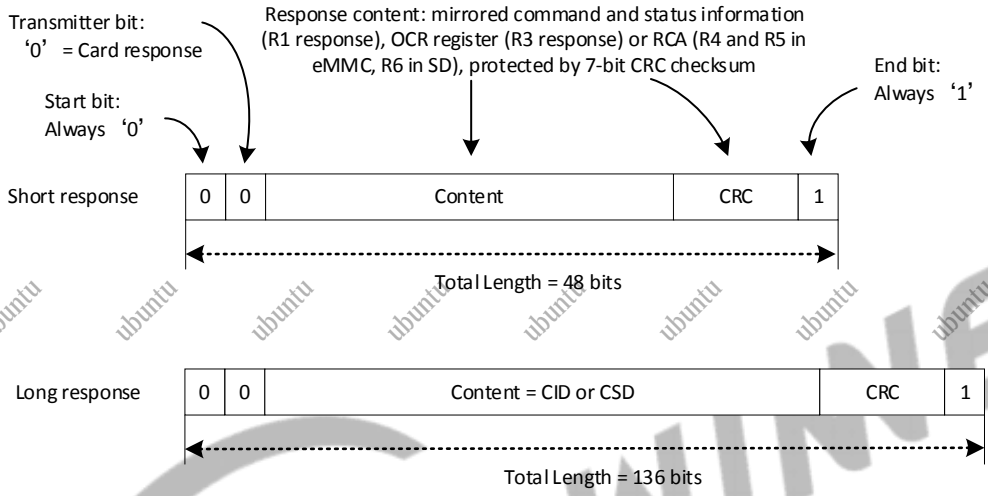
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Response Tokens

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

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

Figure 4-19 Response Token Format



Data Packets

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

Data packets have the following coding scheme:

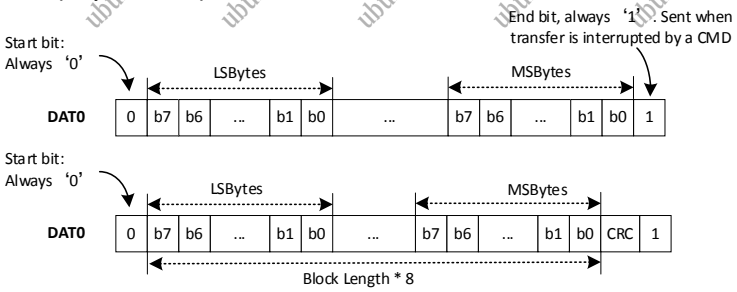
NOTE

R329 does not support the bus width of 8 bits.

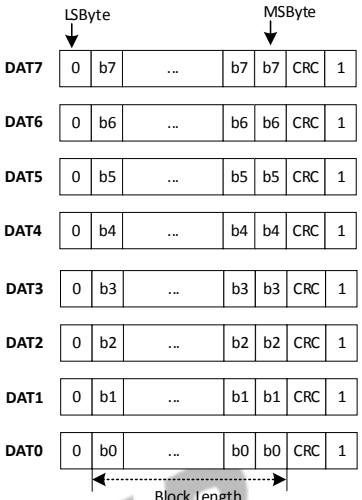
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 4-20 Data Packet Format for SDR

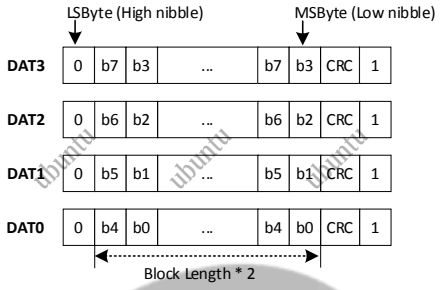
1 Bit Bus (only DAT0 used)



8 Bits Bus (DAT7 – DAT0 used)



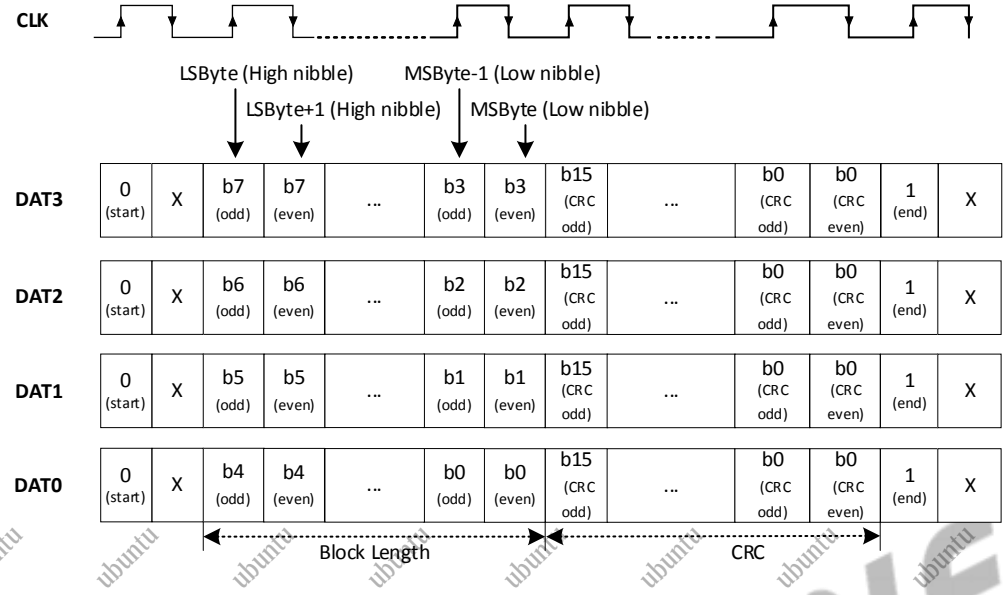
4 Bits Bus (DAT3 – DAT0 used)



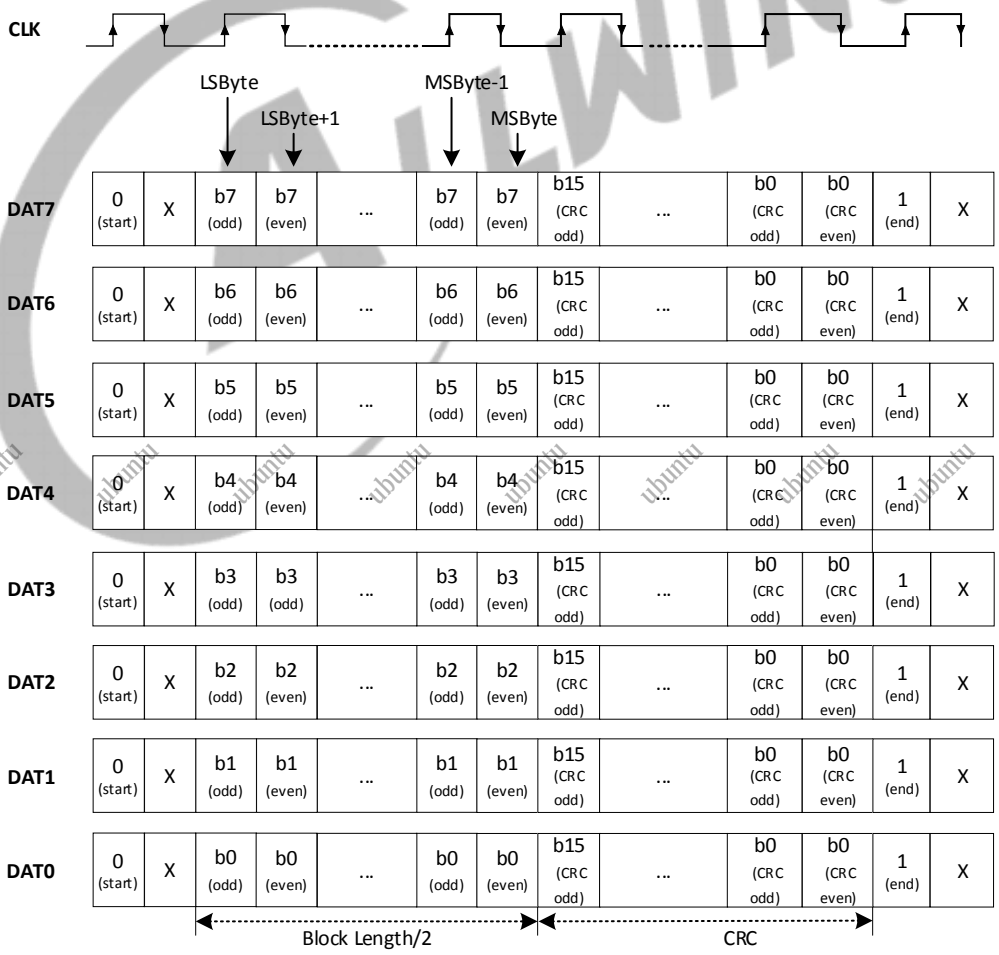
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 4-21 Data Packet Format for DDR

4 Bits Bus DDR (DAT3 – DAT0 used)



8 Bits Bus DDR (DAT7 – DAT0 used)



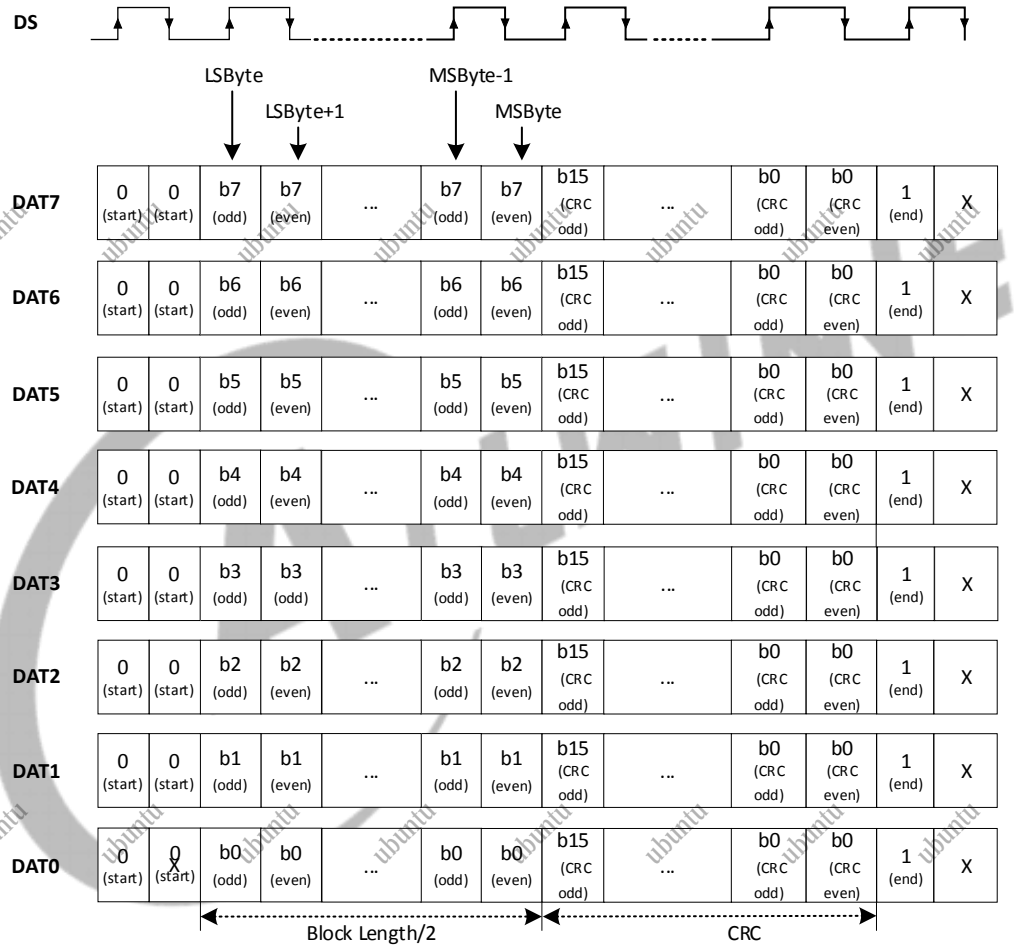
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

NOTE

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

Figure 4-22 Data Packet Format for DDR in HS400 Mode

8 Bits Bus DDR for HS400 Output (DAT7 – DAT0 used)



NOTE

- Bytes data are not interleaved but CRCs are interleaved.
- Start bits are valid when Data Strobe is High and Low.
- End bits are only valid when Data Strobe is High (“X” indicates “undefined”).

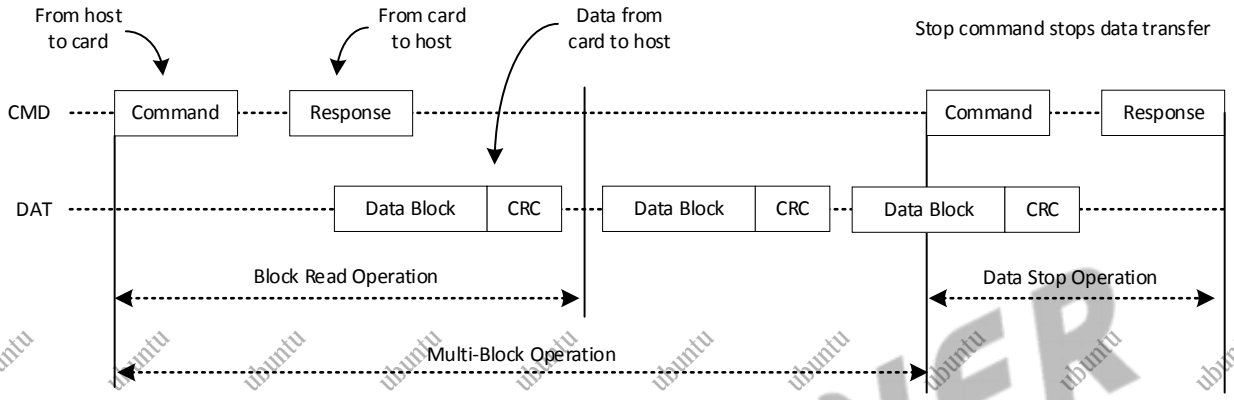
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.3.3.6 Data Transfer

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

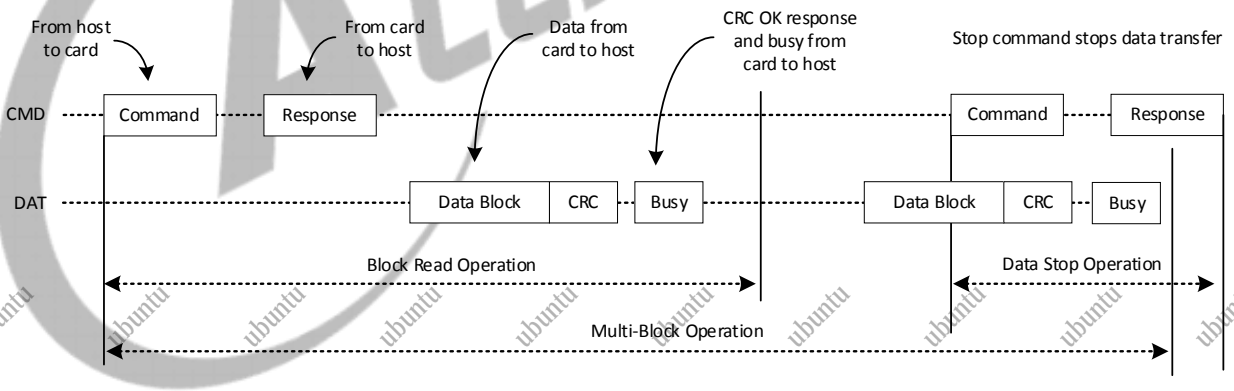
The following figure shows the single-block and multi-block read operation.

Figure 4-23 Single-Block and Multi-Block Read Operation



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

Figure 4-24 Single-Block and Multi-Block Read Operation



4.3.3.7 Bus Speed Modes

The following table shows the bus speed modes supported by SD 3.0.

Table 4-12 Speed Modes Supported by SD 3.0

Mode	I/O Voltage	Bus Width	Frequency	Throughput
SDR104	1.8 V	1, 4 bits	0 to 208 MHz	0 to 104 MB/s
SDR50	1.8 V	1, 4 bits	0 to 100MHz	0 to 50 MB/s

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Mode	I/O Voltage	Bus Width	Frequency	Throughput
DDR50	1.8 V	1, 4 bits	0 to 50 MHz	0 to 50 MB/s
SDR25	1.8 V	1, 4 bits	0 to 50 MHz	0 to 25 MB/s
SDR12	1.8 V	1, 4 bits	0 to 25 MHz	0 to 12 MB/s
High Speed Mode (HS)	3.3 V	1, 4 bits	0 to 50 MHz	0 to 25 MB/s
Default Speed Mode (DS)	3.3 V	1, 4 bits	0 to 25 MHz	0 to 12 MB/s

The following table shows the bus speed modes supported by eMMC 5.1.

Table 4-13 Speed Modes Supported by eMMC 5.1

Mode	Data Rate	I/O Voltage	Bus Width	Frequency	Throughput
Backwards Compatibility with legacy MMC card	Single	3 V/1.8 V	1, 4, 8 bits	0 to 26 MHz	0 to 26 MB/s
High Speed SDR	Single	3 V/1.8 V	1, 4, 8 bits	0 to 52 MHz	0 to 52 MB/s
High Speed DDR	Dual	3 V/1.8 V	4, 8 bits	0 to 52 MHz	0 to 104 MB/s
HS200	Single	1.8 V	4, 8 bits	0 to 200 MHz	0 to 200 MB/s
HS400	Dual	1.8 V	8 bits	0 to 200 MHz	0 to 400 MB/s

NOTE

R329 does not support the bus width of 8 bits.

4.3.3.8 Phase Offset of the Commands and Data

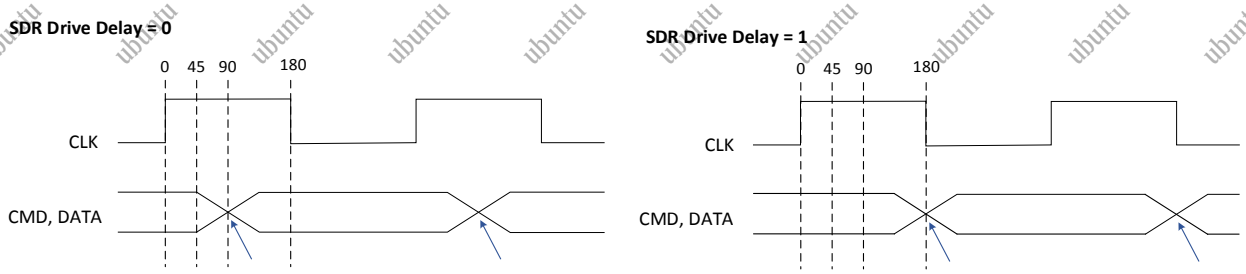
Configure the phase offset of the commands and data by the SMHC_DRV_DL register.

SDR Mode

The following figure shows the phase offset of commands and data in SDR mode.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 4-25 Phase Offset Diagram of Commands and Data in SDR Mode



For SDR mode, the command and data have the same phase offset when the drive delay is the same.

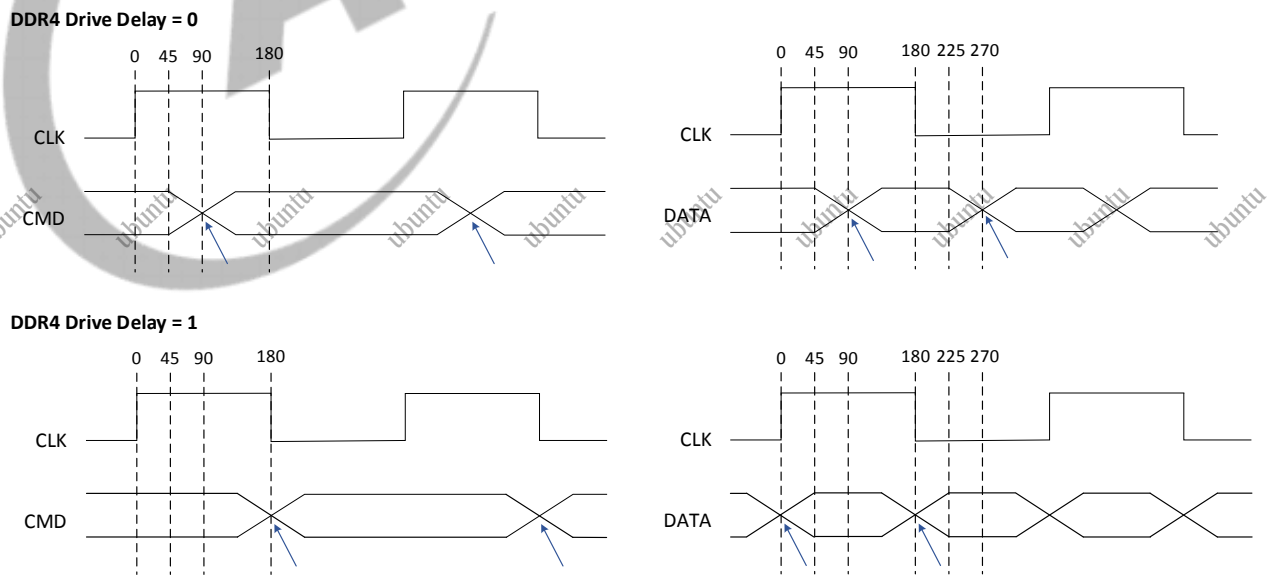
Table 4-14 Phase Offset Value of Commands and Data in SDR Mode

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

DDR4 Mode

The following figure shows the phase offset of commands and data in DDR4 mode.

Figure 4-26 Phase Offset Diagram of Commands and Data in DDR4 Mode (SMHC_NTSR[31] = 0)



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Table 4-15 Phase Offset Value of Commands and Data in DDR4 Mode

Drive Delay	Command	Data
0	The command is updated in 90° clock position	The data is updated in 90° or 270° clock position
1	The command is updated in 180° clock position	The command is updated in 0° or 180° clock position

DDR4 (2x) Mode

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

Figure 4-27 Phase Offset Diagram of Commands and Data in DDR4 (2x) Mode (SMHC_NTSR[31] = 1)

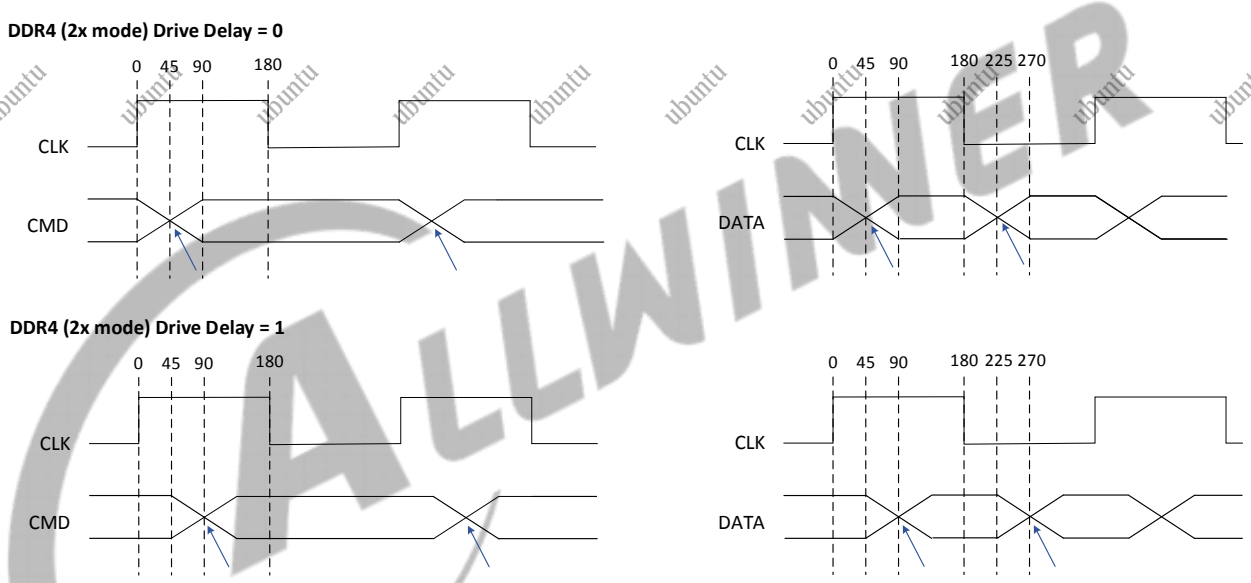


Table 4-16 Phase Offset Value of Commands and Data in DDR4 (2x) Mode

Drive Delay	Command	Data
0	The command is updated in 45° clock position	The data is updated in 45° or 225° clock position
1	The command is updated in 90° clock position	The command is updated in 90° or 270° clock position

4.3.3.9 Internal DMA Controller Description

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

the host driver should construct a descriptor list, configure arguments of every DMA transfer, and then launch the descriptor and start the DMA.

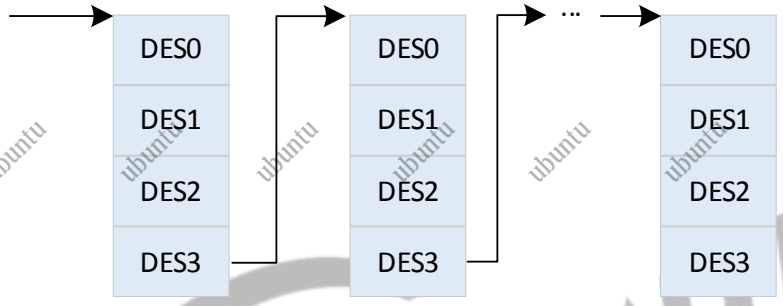
The IDMAC has an interrupt controller. When enabled, it generates an interrupt to the HOST CPU in situations such as data transmission is completed or some errors happen.

IDMAC Descriptor Structure

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

The following figure shows the internal formats of a descriptor.

Figure 4-28 IDMAC Descriptor Structure Diagram



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

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

The following table shows the bit definition of DES0.

Table 4-17 DES0 Definition

Bits	Name	Description
31	HOLD	DES_OWN_FLAG When set, this bit indicates that the descriptor is owned by the IDMAC. When this bit is reset, it indicates that the descriptor is owned by the host. This bit is cleared when the transfer is over.
30	ERROR	ERR_FLAG When some errors happen in transfer, this bit will be set to 1.
29:5	/	/
4	Chain Flag	CHAIN_MOD When set to 1, this bit indicates that the second address in the descriptor is the next descriptor address. It must be set to 1.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Bits	Name	Description
3	First DES Flag	FIRST_FLAG When set to 1, this bit indicates that this descriptor contains the first buffer of data. It must be set to 1 in the first DES.
2	Last DES Flag	LAST_FLAG When set to 1, this bit indicates that the buffers this descriptor points to are the last data buffer.
1	Disable Interrupt on completion	CUR_TXRX_OVER_INT_DIS When set to 1, this bit will prevent the setting of the TX/RX interrupt bit of the IDMAC status register for data that ends in the buffer the descriptor points to.
0	/	/

The following table shows the bit definition of DES1.

Table 4-18 DES1 Definition

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

The following table shows the bit definition of DES2.

Table 4-19 DES2 Definition

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

The following table shows the bit definition of DES3.

Table 4-20 DES3 Definition

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

4.3.3.10 Calibrating the Delay Chain

There are two delay chains in SMHC: data strobe delay chain and sample delay chain.

Data strobe delay chain: used to generate delay to make proper timing between Data Strobe and data signals.

Sample delay chain: used to generate delay to make proper timing between the internal card clock signal and data signals.

Each delay chain is made up with 64 delay cells. The delay time of one delay cell can be estimated through delay chain calibration.

Follow the steps below to calibrate the delay chain:

Step 1 Enable SMHC. In order to calibrate the delay chain by operation registers in SMHC, SMHC must be enabled through SMHC Bus Gating Reset Register and SMHC0 Clock Register/SMHC1 Clock Register.

Step 2 Configure a proper clock for SMHC. The delay chain calibration is based on the clock for SMHC from Clock Control Unit (CCU). The delay chain calibration is an internal function in SMHC and needs no devices. So it is unnecessary to open the clock signal for devices. The recommended clock frequency is 200 MHz.

Step 3 Set proper initial delay value. Writing 0xA0 to **delay control register** enables **Delay Software Enable** (bit[7]) and sets initial delay value 0x20 to **Delay chain** (bit[5:0]). Then write 0x0 to **delay control register** to clear the value.

Step 4 Write 0x8000 to **delay control register** to start calibrating the delay chain.

Step 5 Wait until the flag (bit14 in **delay control register**) of calibration done is set. The number of delay cells is shown at bit[13:8] in **delay control register**. The delay time generated by these delay cells is equal to the cycle of the SMHC clock nearly. This value is the result of calibration.

Step 6 Calculate the delay time of one delay cell according to the cycle of the SMHC clock and the result of calibration.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.3.4 Programming Guidelines

4.3.4.1 Initializing SMHC

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

Step 1 Configure the corresponding GPIO register as an SMHC by Port Controller module; reset clock by writing 1 to SMHC_BGR_REG[SMHCx_RST], and open clock gating by writing 1 to SMHC_BGR_REG[SMHCx_GATING]; select clock sources and set the division factor by configuring the SMHCx_CLK_REG (x = 0, 1) register.

Step 2 Configure SMHC_CTRL to reset the FIFO and controller, and enable the global interrupt; configure SMHC_INTMASK to 0xFFCE to enable normal interrupts and error abnormal interrupts, and then register the interrupt function.

Step 3 Configure SMHC_CLKDIV to open clock for devices; configure SMHC_CMD as the change clock command (for example 0x80202000); send the update clock command to deliver clocks to devices.

Step 4 Configure SMHC_CMD as a normal command. Configure SMHC_CMDARG to set command parameters. Configure SMHC_CMD to set parameters like whether to send the response, the response type, and the response length and then send the commands. According to the initialization process in the protocol, you can finish SMHC initialization by sending the corresponding command one by one.

4.3.4.2 Writing a Single Data Block

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

Step 1 Write 0x1 to SMHC_CTRL[DMA_RST] to reset the internal DMA controller; write 0x82 to SMHC_IDMAC to enable the IDMAC interrupt and configure AHB master burst transfers; configure SMHC_IDIE to enable the transfer interrupt, receive interrupt, and abnormal interrupt.

Step 2 Configure SMHC_FIFOTH to determine the burst size and TX/RX trigger level. For example, if SMHC_FIFOTH is configured as 0x300F00F0, it indicates the burst size is 16, TX_TL is 15, and RX_TL is 240. Configure SMHC_DLBA to determine the start address of the DMA descriptor.

Step 3 To write one block data to sector1, configure SMHC_BYTCNT[BYTE_CNT] to 0x200 and configure the descriptor according to the data size; set the data sector address of CMD24 (Single Data Block Write) to 0x1, write 0x80002758 to SMHC_CMD, and send CMD24 command to write data to the device.

Step 4 Check whether SMHC_RINTSTS[CC] is 1. If yes, the command is sent successfully; otherwise, continue to wait until timeout, and then exit the process.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 5 Check whether SMHC_IDST[TX_INT] is 1. If yes, the data transfer for writing DMA is completed. Write 0x337 to SMHC_IDST to clear the interrupt flag. Otherwise, continue to wait until timeout, and then exit the process.

Step 6 Check whether SMHC_RINTSTS[DTC] is 1. If yes, the data transfer and CMD24 writing operations are completed. Otherwise, abnormity exists. Read SMHC_RINTSTS and SMHC_STATUS to query the existing abnormity.

Step 7 Send CMD13 command to query whether the device writing operation is completed and returns to the idle status. For example, device RCA is 0x1234, first set SMHC_CMDARG to 0x12340000, write 0x8000014D to SMHC_CMD, go to step4 to ensure command transfer completed, and then check whether the highest bit of SMHC_RESP0 (CMD13 response) is 1. If yes, the device is in the idle status, and the next command can be sent. Otherwise, the device is in the busy status. Continue to send CMD13 to wait for the device to enter the idle status until timeout.

4.3.4.3 Reading a Single Data Block

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

Step 1 Write 0x1 to SMHC_CTRL[DMA_RST] to reset the internal DMA controller; write 0x82 to SMHC_IDMAC to enable the IDMAC interrupt and configure AHB master burst transfers; configure SMHC_IDIE to enable the transfer interrupt, receive interrupt, and abnormal interrupt.

Step 2 Configure SMHC_FIFOTH to determine the burst size and TX/RX trigger level. For example, if SMHC_FIFOTH is configured as 0x300F00F0, it indicates the burst size is 16, TX_TL is 15, and RX_TL is 240. Configure SMHC_DLBA to determine the start address of the DMA descriptor.

Step 3 To read one block data from sector1, configure SMHC_BYTCNT[BYTE_CNT] to 0x200 and configure the descriptor according to the data size; set the data sector address of CMD17 command (Single Data Block Read) to 0x1, write 0x80002351 to SMHC_CMD, and send CMD17 command to read data from the device to DRAM/SRAM.

Step 4 Check whether SMHC_RINTSTS[CC] is 1. If yes, the command is sent successfully; otherwise, continue to wait until timeout, and then exit the process.

Step 5 Check whether SMHC_IDST[RX_INT] is 1. If yes, the data transfer for writing DMA is completed. Write 0x337 to SMHC_IDST to clear the interrupt flag. Otherwise, continue to wait until timeout, and then exit the process.

Step 6 Check whether SMHC_RINTSTS[DTC] is 1. If yes, data transfer and CMD17 reading operation are completed. Otherwise, abnormity exists. Read SMHC_RINTSTS and SMHC_STATUS to query the existing abnormity.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.3.4.4 Writing Open-Ended Multiple Data Blocks (CMD25 + Auto CMD12)

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

Step 1 Write 0x1 to SMHC_CTRL[DMA_RST] to reset the internal DMA controller; write 0x82 to SMHC_IDMAC to enable the IDMAC interrupt and configure AHB master burst transfers; configure SMHC_IDIE to enable the transfer interrupt, receive interrupt, and abnormal interrupt.

Step 2 Configure SMHC_FIFOTH to determine the burst size and TX/RX trigger level. For example, if SMHC_FIFOTH is configured as 0x300F00F0, it indicates the burst size is 16, TX_TL is 15, and RX_TL is 240. Configure SMHC_DLBA to determine the start address of the DMA descriptor.

Step 3 To write three blocks of data to sectors begin with sector0, configure SMHC_BYTCNT[BYTE_CNT] to 0x600 and configure the descriptor according to the data size; set the data sector address of CMD25 command (Multiple Data Blocks Write) to 0x0, write 0x80003759 to SMHC_CMD, and send CMD25 command to read data from the device to DRAM/SRAM.

Step 4 Check whether SMHC_RINTSTS[CC] is 1. If yes, the command is sent successfully; otherwise, continue to wait until timeout, and then exit the process.

Step 5 Check whether SMHC_IDST[RX_INT] is 1. If yes, the data transfer for writing DMA is completed. Write 0x337 to SMHC_IDST to clear the interrupt flag. Otherwise, continue to wait until timeout, and then exit the process.

Step 6 Check whether SMHC_RINTSTS[ACD] and SMHC_RINTSTS[DTC] are both 1. If yes, the data transfer, CMD12 transfer, and CMD25 writing operations are completed. Otherwise, abnormity exists. Read SMHC_RINTSTS and SMHC_STATUS to query the existing abnormity.

Step 7 Send CMD13 command to query whether the device writing operation is completed and returns to the idle status. For example, device RCA is 0x1234, first set SMHC_CMDARG to 0x12340000, write 0x8000014D to SMHC_CMD, go to step4 to ensure command transfer completed, and then check whether the highest bit of SMHC_RESP0 (CMD13 response) is 1. If yes, the device is in the idle status, and the next command can be sent. Otherwise, the device is in the busy status. Continue to send CMD13 to wait for the device to enter the idle status until timeout.

4.3.4.5 Reading Open-Ended Multiple Data Blocks (CMD18 + Auto CMD12)

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

Step 1 Write 0x1 to SMHC_CTRL[DMA_RST] to reset the internal DMA controller; write 0x82 to SMHC_IDMAC to enable the IDMAC interrupt and configure AHB master burst transfers; configure SMHC_IDIE to enable the transfer interrupt, receive interrupt, and abnormal interrupt.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 2 Configure SMHC_FIFOTH to determine the burst size and TX/RX trigger level. For example, if SMHC_FIFOTH is configured as 0x300F00F0, it indicates the burst size is 16, TX_TL is 15, and RX_TL is 240. Configure SMHC_DLBA to determine the start address of the DMA descriptor.

Step 3 To read three blocks of data from sectors begin with sector0, configure SMHC_BYTCNT[BYTE_CNT] to 0x600 and configure the descriptor according to the data size; set the data sector address of CMD18 command (Multiple Data Blocks Read) to 0x0, write 0x80003352 to SMHC_CMD, and send CMD18 command to read data to the device. When the data transfer is completed, CMD12 will be sent automatically.

Step 4 Check whether SMHC_RINTSTS[CC] is 1. If yes, the command is sent successfully; otherwise, continue to wait until timeout, and then exit the process.

Step 5 Check whether SMHC_IDST[RX_INT] is 1. If yes, the data transfer for writing DMA is completed. Write 0x337 to SMHC_IDST to clear the interrupt flag. Otherwise, continue to wait until timeout, and then exit the process.

Step 6 Check whether SMHC_RINTSTS[ACD] and SMHC_RINTSTS[DTC] are both 1. If yes, data transfer, CMD12 transfer, and CMD18 reading operation are completed. Otherwise, abnormity exists. Read SMHC_RINTSTS and SMHC_STATUS to query the existing abnormity.

4.3.4.6 Writing Pre-Defined Multiple Data Blocks (CMD23 + CMD25)

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

Step 1 Write 0x1 to SMHC_CTRL[DMA_RST] to reset the internal DMA controller; write 0x82 to SMHC_IDMAC to enable the IDMAC interrupt and configure AHB master burst transfers; configure SMHC_IDIE to enable the transfer interrupt, receive interrupt, and abnormal interrupt.

Step 2 Configure SMHC_FIFOTH to determine the burst size and TX/RX trigger level. For example, if SMHC_FIFOTH is configured as 0x300F00F0, it indicates the burst size is 16, TX_TL is 15, and RX_TL is 240. Configure SMHC_DLBA to determine the start address of the DMA descriptor.

Step 3 To write three blocks of data, configure SMHC_CMDARG to 0x3 to specify the number of data blocks as three. Then write 0x80000157 to SMHC_CMD to send the CMD23 command. Check whether SMHC_RINTSTS[CC] is 1. If yes, the command is sent successful; otherwise, continue to wait until timeout, and then exit the process.

Step 4 Configure SMHC_BYTCNT[BYTE_CNT] to 0x600 and configure the descriptor according to the data size; set the data sector address of CMD25 command (Multiple Data Blocks Write) to 0x0, write 0x80002759 to SMHC_CMD, and send CMD25 command to write data to the device.

Step 5 Check whether SMHC_RINTSTS[CC] is 1. If yes, the command is sent successful; otherwise, continue to wait until timeout, and then exit the process.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 6 Check whether SMHC_IDST[TX_INT] is 1. If yes, the data transfer for writing DMA is completed. Write 0x337 to SMHC_IDST to clear the interrupt flag. Otherwise, continue to wait until timeout, and then exit the process.

Step 7 Check whether SMHC_RINTSTS[DTC] is 1. If yes, the data transfer and CMD25 writing operations are completed. Otherwise, abnormity exists. Read SMHC_RINTSTS and SMHC_STATUS to query the existing abnormity.

Step 8 Send CMD13 command to query whether the device writing operation is completed and returns to the idle status. For example, device RCA is 0x1234, first set SMHC_CMDARG to 0x12340000, write 0x8000014D to SMHC_CMD, go to step4 to ensure command transfer completed, and then check whether the highest bit of SMHC_RESP0 (CMD13 response) is 1. If yes, the device is in the idle status, and the next command can be sent. Otherwise, the device is in the busy status. Continue to send CMD13 to wait for the device to enter the idle status until timeout.

4.3.4.7 Reading Pre-Defined Multiple Data Blocks (CMD23 + CMD18)

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

Step 1 Write 0x1 to SMHC_CTRL[DMA_RST] to reset the internal DMA controller; write 0x82 to SMHC_IDMAC to enable the IDMAC interrupt and configure AHB master burst transfers; configure SMHC_IDIE to enable the transfer interrupt, receive interrupt, and abnormal interrupt.

Step 2 Configure SMHC_FIFOTH to determine the burst size and TX/RX trigger level. For example, if SMHC_FIFOTH is configured as 0x300F00F0, it indicates the burst size is 16, TX_TL is 15, and RX_TL is 240. Configure SMHC_DLBA to determine the start address of the DMA descriptor.

Step 3 To read three blocks of data, configure SMHC_CMDARG to 0x3 to specify the number of data blocks as three. Then write 0x80000157 to SMHC_CMD to send the CMD23 command. Check whether SMHC_RINTSTS[CC] is 1. If yes, the command is sent successful; otherwise, continue to wait until timeout, and then exit the process.

Step 4 Configure SMHC_BYTCNT[BYTE_CNT] to 0x600 and configure the descriptor according to the data size; set the data sector address of CMD18 (Multiple Data Blocks Read) to 0x0, write 0x80002352 to SMHC_CMD, and send CMD18 command to read data from device to DRAM/SRAM.

Step 5 Check whether SMHC_RINTSTS[CC] is 1. If yes, the command is sent successful; otherwise, continue to wait until timeout, and then exit the process.

Step 6 Check whether SMHC_IDST[TX_INT] is 1. If yes, the data transfer for writing DMA is completed. Write 0x337 to SMHC_IDST to clear the interrupt flag. Otherwise, continue to wait until timeout, and then exit the process.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 7 Check whether SMHC_RINTSTS[DTC] is 1. If yes, the data transfer and CMD18 writing operations are completed. Otherwise, abnormity exists. Read SMHC_RINTSTS and SMHC_STATUS to query the existing abnormity.

4.3.5 Register List

Module Name	Base Address
SMHC0	0x04020000
SMHC1	0x04021000

Register Name	Offset	Description
SMHC_CTRL	0x0000	Control Register
SMHC_CLKDIV	0x0004	Clock Control Register
SMHC_TMOUT	0x0008	Time Out Register
SMHC_CTYPE	0x000C	Bus Width Register
SMHC_BLKSIZ	0x0010	Block Size Register
SMHC_BYTCNT	0x0014	Byte Count Register
SMHC_CMD	0x0018	Command Register
SMHC_CMDARG	0x001C	Command Argument Register
SMHC_RESP0	0x0020	Response 0 Register
SMHC_RESP1	0x0024	Response 1 Register
SMHC_RESP2	0x0028	Response 2 Register
SMHC_RESP3	0x002C	Response 3 Register
SMHC_INTMASK	0x0030	Interrupt Mask Register
SMHC_MINTSTS	0x0034	Masked Interrupt Status Register
SMHC_RINTSTS	0x0038	Raw Interrupt Status Register
SMHC_STATUS	0x003C	Status Register
SMHC_FIFOTH	0x0040	FIFO Water Level Register
SMHC_FUNS	0x0044	FIFO Function Select Register
SMHC_TCBCNT	0x0048	Transferred Byte Count between Controller and Card
SMHC_TBBCNT	0x004C	Transferred Byte Count between Host Memory and Internal FIFO
SMHC_DBGC	0x0050	Current Debug Control Register
SMHC_CSDC	0x0054	CRC Status Detect Control Registers
SMHC_A12A	0x0058	Auto Command 12 Argument Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
SMHC_NTSR	0x005C	SD New Timing Set Register
SMHC_HWRST	0x0078	Hardware Reset Register
SMHC_IDMAC	0x0080	IDMAC Control Register
SMHC_DLBA	0x0084	Descriptor List Base Address Register
SMHC_IDST	0x0088	IDMAC Status Register
SMHC_IDIE	0x008C	IDMAC Interrupt Enable Register
SMHC_THLD	0x0100	Card Threshold Control Register
SMHC_SFC	0x0104	Sample FIFO Control Register
SMHC_A23A	0x0108	Auto Command 23 Argument Register
EMMC_DDR_SBIT_DET	0x010C	eMMC4.5 DDR Start Bit Detection Control Register
SMHC_EXT_CMD	0x0138	Extended Command Register
SMHC_EXT_RESP	0x013C	Extended Response Register
SMHC_DRV_DL	0x0140	Drive Delay Control Register
SMHC_SMAP_DL	0x0144	Sample Delay Control Register
SMHC_DS_DL	0x0148	Data Strobe Delay Control Register
SMHC_HS400_DL	0x014C	HS400 Delay Control Register
SMHC_FIFO	0x0200	Read/Write FIFO

4.3.6 Register Description

4.3.6.1 0x0000 SMHC Global Control Register (Default Value: 0x0000_0100)

Offset: 0x0000			Register Name: SMHC_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	FIFO_AC_MOD FIFO Access Mode 0: DMA bus 1: AHB bus
30:13	/	/	/
12	R/W	0x0	TIME_UNIT_CMD Time unit for command line The time unit is used to calculate the command line time out value defined in RTO_LMT. 0: 1 card clock period 1: 256 card clock period

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: SMHC_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
11	R/W	0x0	TIME_UNIT_DAT Time unit for data line Time unit is used to calculate the data line time out value defined in DTO_LMT. 0: 1 card clock period 1: 256 card clock period
10	R/W	0x0	DDR_MOD_SEL DDR Mode Select Although the HS400 speed mode of eMMC is 8-bit DDR, this field should be cleared when HS400_MD_EN is set. 0: SDR mode 1: DDR mode
9	/	/	/
8	R/W	0x1	CD_DBC_ENB Card Detect (Data[3] status) De-bounce Enable 0: Disable de-bounce 1: Enable de-bounce
7:6	/	/	/
5	R/W	0x0	DMA_ENB DMA Global Enable 0: Disable DMA to transfer data via AHB bus 1: Enable DMA to transfer data
4	R/W	0x0	INT_ENB Global Interrupt Enable 0: Disable interrupts 1: Enable interrupts
3	/	/	/
2	R/W	0x0	DMA_RST DMA Reset
1	R/W	0x0	FIFO_RST FIFO Reset 0: No effect 1: Reset the FIFO This bit is auto-cleared after the completion of the reset operation.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: SMHC_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
0	R/W	0x0	SOFT_RST Software Reset 0: No effect 1: Reset SD/MMC controller This bit is auto-cleared after the completion of reset operation.

4.3.6.2 0x0004 SMHC Clock Control Register (Default Value: 0x0000_0000)

Offset: 0x0004			Register Name: SMHC_CLKDIV
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	MASK_DATA0 0: Do not mask data0 when update clock 1: Mask data0 when update clock
30:18	/	/	/
17	R/W	0x0	CCLK_CTRL Card Clock Output Control 0: Card clock is always on. 1: Turn off card clock when FSM is in IDLE state.
16	R/W	0x0	CCLK_ENB Card Clock Enable 0: Card Clock is off. 1: Card Clock is on.
15:8	/	/	/
7:0	R/W	0x0	CCLK_DIV Card Clock Divider n: Source clock is divided by 2*n. (n = 0 to 255) When HS400_MD_EN is set, this field must be cleared.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.3.6.3 0x0008 SMHC Timeout Register (Default Value: 0xFFFF_FF40)

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

4.3.6.4 0x000C SMHC Bus Width Register (Default Value: 0x0000_0000)

Offset: 0x000C			Register Name: SMHC_CTYPE
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1:0	R/W	0x0	CARD_WID Card Width 00: 1-bit width 01: 4-bit width 1x: 8-bit width

4.3.6.5 0x0010 SMHC Block Size Register (Default Value: 0x0000_0200)

Offset: 0x0010			Register Name: SMHC_BLKSIZ
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

4.3.6.6 0x0014 SMHC Byte Count Register (Default Value: 0x0000_0200)

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

4.3.6.7 0x0018 SMHC Command Register (Default Value: 0x0000_0000)

Offset: 0x0018			Register Name: SMHC_CMD
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	CMD_LOAD Start Command This bit is automatically cleared when the current command is sent. If there is no response error happens, a command complete interrupt bit (CMD_OVER) will be set in the interrupt register. Do not write any other commands until this bit is cleared.
30:29	/	/	/
28	R/W	0x0	VOL_SW Voltage Switch 0: Normal command 1: Voltage switch command, set for CMD11 only.
27	R/W	0x0	BOOT_ABT Boot Abort Setting this bit will terminate the boot operation.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0018			Register Name: SMHC_CMD
Bit	**Read/Write**	**Default/Hex**	**Description**
26	R/W	0x0	EXP_BOOT_ACK Expect Boot Acknowledge When the software sets this bit along in mandatory boot operation, the controller expects a boot acknowledge start pattern of 0-1-0 from the selected card.
25:24	R/W	0x0	BOOT_MOD Boot Mode 00: Normal command 01: Mandatory Boot operation 10: Alternate Boot operation 11: Reserved
23:22	/	/	/
21	R/W	0x0	PRG_CLK Change Clock 0: Normal command 1: Change Card Clock When this bit is set, the controller will change the clock domain and clock output. No commands will be sent.
20:16	/	/	/
15	R/W	0x0	SEND_INIT_SEQ Send Initialization 0: Normal command sending 1: Send initialization sequence before sending this command.
14	R/W	0x0	STOP_ABT_CMD Stop Abort Command 0: Normal command sending 1: Send *Stop* or *Abort* command to stop the current data transfer in progress. (CMD12, CMD52 for writing “I/O Abort” in SDIO CCCR)
13	R/W	0x0	WAIT_PRE_OVER Wait for Data Transfer Over 0: Send command at once, does not care about data transferring. 1: Wait for data transfer completion before sending the current command.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0018			Register Name: SMHC_CMD
Bit	**Read/Write**	**Default/Hex**	**Description**
12	R/W	0x0	STOP_CMD_FLAG Send Stop CMD Automatically (CMD12) 0: Do not send stop command at the end of the data transfer. 1: Send stop command automatically at the end of the data transfer. If set, the SMHC_RESP1 will record the response of auto CMD12.
11	R/W	0x0	TRANS_MODE Transfer Mode 0: Block data transfer command 1: Stream data transfer command
10	R/W	0x0	TRANS_DIR Transfer Direction 0: Read operation 1: Write operation
9	R/W	0x0	DATA_TRANS Data Transfer 0: Without data transfer 1: With data transfer
8	R/W	0x0	CHK_RESP_CRC Check Response CRC 0: Do not check response CRC 1: Check response CRC
7	R/W	0x0	LONG_RESP Response Type 0: Short Response (48 bits) 1: Long Response (136 bits)
6	R/W	0x0	RESP_RCV Response Receive 0: Command without response 1: Command with response
5:0	R/W	0x0	CMD_IDX CMD Index Command index value

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.3.6.8 0x001C SMHC Command Argument Register (Default Value: 0x0000_0000)

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

4.3.6.9 0x0020 SMHC Response 0 Register (Default Value: 0x0000_0000)

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

4.3.6.10 0x0024 SMHC Response 1 Register (Default Value: 0x0000_0000)

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

4.3.6.11 0x0028 SMHC Response 2 Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.3.6.12 0x002C SMHC Response 3 Register (Default Value: 0x0000_0000)

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

4.3.6.13 0x0030 SMHC Interrupt Mask Register (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: SMHC_INTMASK
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	CARD_REMOVAL_INT_EN Card Removed Interrupt Enable
30	R/W	0x0	CARD_INSERT_INT_EN Card Inserted Interrupt Enable
29:17	/	/	/
16	R/W	0x0	SDIO_INT_EN SDIO Interrupt Enable
15	R/W	0x0	DEE_INT_EN Data End-bit Error Interrupt Enable
14	R/W	0x0	ACD_INT_EN Auto Command Done Interrupt Enable
13	R/W	0x0	DSE_BC_INT_EN Data Start Error Interrupt Enable
12	R/W	0x0	CB_IW_INT_EN Command Busy and Illegal Write Interrupt Enable
11	R/W	0x0	FU_FO_INT_EN FIFO Underrun/Overflow Interrupt Enable
10	R/W	0x0	DSTO_VSD_INT_EN Data Starvation Timeout/V1.8 Switch Done Interrupt Enable
9	R/W	0x0	DTO_BDS_INT_EN Data Timeout/Boot Data Start Interrupt Enable
8	R/W	0x0	RTO_BACK_INT_EN Response Timeout/Boot ACK Received Interrupt Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0030			Register Name: SMHC_INTMASK
Bit	**Read/Write**	**Default/Hex**	**Description**
7	R/W	0x0	DCE_INT_EN Data CRC Error Interrupt Enable
6	R/W	0x0	RCE_INT_EN Response CRC Error Interrupt Enable
5	R/W	0x0	DRR_INT_EN Data Receive Request Interrupt Enable
4	R/W	0x0	DTR_INT_EN Data Transmit Request Interrupt Enable
3	R/W	0x0	DTC_INT_EN Data Transfer Complete Interrupt Enable
2	R/W	0x0	CC_INT_EN Command Complete Interrupt Enable
1	R/W	0x0	RE_INT_EN Response Error Interrupt Enable
0	/	/	/

4.3.6.14 0x0034 SMHC Masked Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0034			Register Name: SMHC_MINTSTS
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R	0x0	M_CARD_REMOVAL_INT Card Removed
30	R	0x0	M_CARD_INSERT Card Inserted
29:17	/	/	/
16	R	0x0	M_SDIO_INT SDIO Interrupt
15	R	0x0	M_DEE_INT Data End-bit Error When the bit is set during receiving data, it means that the host controller does not receive the valid data end bit. When the bit is set during transmitting data, it means that the host controller does not receive the CRC status token. Write 1 to clear this bit.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0034			Register Name: SMHC_MINTSTS
Bit	**Read/Write**	**Default/Hex**	**Description**
14	R	0x0	M_ACD_INT Auto Command Done When set, it means auto-stop command (CMD12) completed.
13	R	0x0	M_DSE_BC_INT Data Start Error/Busy Clear When set during receiving data, it means that the host controller found an error start bit. When the bit is set during transmitting data, it means that the busy signal is cleared after the last block.
12	R	0x0	M_CB_IW_INT Command Busy and Illegal Write
11	R	0x0	M_FU_FO_INT FIFO Underrun/Overflow
10	R	0x0	M_DSTO_VSD_INT Data Starvation Timeout/V1.8 Switch Done
9	R	0x0	M_DTO_BDS_INT Data Timeout/Boot Data Start
8	R	0x0	M_RTO_BACK_INT Response Timeout/Boot ACK Received
7	R	0x0	M_DCE_INT Data CRC Error When the bit is set during receiving data, it means that the received data have data CRC error. When the bit is set during transmitting data, it means that the received CRC status taken is negative.
6	R	0x0	M_RCE_INT Response CRC Error
5	R	0x0	M_DRR_INT Data Receive Request When set, it means that there are enough data in FIFO during receiving data.
4	R	0x0	M_DTR_INT Data Transmit Request When set, it means that there is enough space in FIFO during transmitting data.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0034			Register Name: SMHC_MINTSTS
Bit	**Read/Write**	**Default/Hex**	**Description**
3	R	0x0	M_DTC_INT Data Transfer Complete
2	R	0x0	M_CC_INT Command Complete
1	R	0x0	M_RE_INT Response Error When set, Transmit Bit error or End Bit error or CMD Index error may occurs.
0	/	/	/

4.3.6.15 0x0038 SMHC Raw Interrupt Status Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.3.6.16 0x003C SMHC Status Register (Default Value: 0x0000_0006)

Offset: 0x003C			Register Name: SMHC_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R	0x0	DMA_REQ DMA Request DMA request signal state
30:26	/	/	/
25:17	R	0x0	FIFO_LEVEL FIFO Level Number of filled locations in FIFO
16:11	R	0x0	RESP_IDX Response Index Index of previous response, including any auto-stop sent by the controller.
10	R	0x0	FSM_BUSY Data FSM Busy Data transmit or receive state-machine is busy.
9	R	0x0	CARD_BUSY Card Data Busy Inverted version of DATA[0] 0: Card data is not busy. 1: Card data is busy.
8	R	0x0	CARD_PRESENT Data[3] Status The level of DATA[3], checks whether the card is present. 0: The card is not present. 1: The card is present.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x003C			Register Name: SMHC_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
7:4	R	0x0	FSM_STA Command FSM States 0000: Idle 0001: Send init sequence 0010: TX CMD start bit 0011: TX CMD TX bit 0100: TX CMD index + argument 0101: TX CMD CRC7 0110: TX CMD end bit 0111: RX response start bit 1000: RX response IRQ response 1001: RX response TX bit 1010: RX response CMD index 1011: RX response data 1100: RX response CRC7 1101: RX response end bit 1110: CMD path wait NCC 1111: Wait; CMD-to-response turn around
3	R	0x0	FIFO_FULL FIFO Full 1: FIFO is full. 0: FIFO is not full.
2	R	0x1	FIFO_EMPTY FIFO Empty 1: FIFO is empty 0: FIFO is not empty
1	R	0x1	FIFO_TX_LEVEL FIFO TX Water Level Flag 0: FIFO does not reach the transmit trigger level 1: FIFO reaches the transmit trigger level
0	R	0x0	FIFO_RX_LEVEL FIFO RX Water Level Flag 0: FIFO does not reach the receive trigger level. 1: FIFO reaches the receive trigger level.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.3.6.17 0x0040 SMHC FIFO Water Level Register (Default Value: 0x000F_0000)

Offset: 0x0040			Register Name: SMHC_FIFOTH
Bit	**Read/Write**	**Default/Hex**	**Description**
31	/	/	/
30:28	R/W	0x0	BSIZE_OF_TRANS Burst Size of Multiple Transaction 000: 1 transfers 001: 4 010: 8 011: 16 Others: Reserved It should be programmed the same as the DMA controller multiple transaction size. The units for the transfer are the DWORD. A single transfer would be signaled based on this value. Value should be sub-multiple of (RX_TL + 1) and (FIFO_DEPTH - TX_TL) Recommended: FIFO_DEPTH = 256, FIFO_SIZE = 256 * 32 = 1K MSize = 16, TX_TL = 240, RX_TL = 15
27:24	/	/	/
23:16	R/W	0xF	RX_TL RX Trigger Level 0x0 to 0xFE: RX Trigger Level is 0 to 254. 0xFF: Reserved Indicates the FIFO threshold for the FIFO request host to receive data from the FIFO. When the FIFO data level is greater than this value, the DMA request is raised if DMA enabled, or the RX interrupt bit is set if interrupt enabled. At the end of the packet, if the last transfer is less than this level, the value is ignored and the relative request will be raised as usual. Recommended: 15 (means greater than 15)
15:8	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

4.3.6.18 0x0044 SMHC Function Select Register (Default Value: 0x0000_0000)

Offset: 0x0044			Register Name: SMHC_FUNS
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/
2	R/W	0x0	ABT_RDATA Abort Read Data 0: Ignored 1: After the suspend command is issued during the read-transfer, the software polls card to find when the suspend happens. Once the suspend occurs, the software sets the bit to reset the data state-machine, which is waiting for the next block of data. This bit is used in the SDIO card suspends sequence and is auto-cleared once the controller resets to the idle state.
1	R/W	0x0	READ_WAIT Read Wait 0: Clear SDIO read wait 1: Assert SDIO read wait

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0044			Register Name: SMHC_FUNS
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	HOST_SEND_MMC_IRQRESQ Host Send MMC IRQ Response 0: Ignored 1: Send auto IRQ response When the host is waiting for the MMC card interrupt response, setting this bit will make the controller cancel the waiting state and return to the idle state, at which time, the controller will receive the IRQ response sent by itself. This bit is auto-cleared after the response is sent.

4.3.6.19 0x0048 SMHC Transferred Byte Count Register 0 (Default Value: 0x0000_0000)

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

4.3.6.20 0x004C SMHC Transferred Byte Count Register 1 (Default Value: 0x0000_0000)

Offset: 0x004C			Register Name: SMHC_TBC1
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	TBC1 Transferred Count 1 The number of bytes transferred between the Host/DMA memory and internal FIFO. The register should be accessed in full to avoid read-coherency problems and read only after the data transfer completes.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.3.6.21 0x0054 SMHC CRC Status Detect Control Register (Default Value: 0x0000_0003)

Offset: 0x0054			Register Name: SMHC_CSDC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:4	/	/	/
3:0	R/W	0x3	CRC_DET_PARA 110: HS400 speed mode 011: Other speed mode Others: Reserved

4.3.6.22 0x0058 SMHC Auto Command 12 Argument Register (Default Value: 0x0000_FFFF)

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

4.3.6.23 0x005C SMHC New Timing Set Register (Default Value: 0x8171_0000)

Offset: 0x005C			Register Name: SMHC_NTSR
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x1	MODE_SELECT 0: Old mode of Sample/Output Timing 1: New mode of Sample/Output Timing
30:25	/	/	/
24	R/W	0x1	CMD_DAT_RX_PHASE_CLR Clear the input phase of command lines and data lines during the update clock operation. 0: Disabled 1: Enabled
23	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x005C			Register Name: SMHC_NTSR
Bit	**Read/Write**	**Default/Hex**	**Description**
22	R/W	0x1	DAT_CRC_STATUS_RX_PHASE_CLR Clear the input phase of data lines before receiving the CRC status. 0: Disabled 1: Enabled
21	R/W	0x1	DAT_TRANS_RX_PHASE_CLR Clear the input phase of data lines before transferring the data. 0: Disabled 1: Enabled
20	R/W	0x1	DAT_RECV_RX_PHASE_CLR Clear the input phase of data lines before receiving the data. 0: Disabled 1: Enabled
19:17	/	/	/
16	R/W	0x1	CMD_SEND_RX_PHASE_CLR Clear command rx phase before sending the command. 0: Disabled 1: Enabled
15:10	/	/	/
9:8	R/W	0x0	DAT_SAMPLE_TIMING_PHASE 00: Sample timing phase offset 90^0 01: Sample timing phase offset 180^0 10: Sample timing phase offset 270^0 11: Sample timing phase offset 0^0 (only for SD2 hs400 mode)
7:6	/	/	/
5:4	R/W	0x0	CMD_SAMPLE_TIMING_PHASE 00: Sample timing phase offset 90^0 01: Sample timing phase offset 180^0 10: Sample timing phase offset 270^0 11: Ignore
3:1	/	/	/
0	R/W	0x0	HS400_NEW_SAMPLE_EN 0: Disable hs400 new sample method 1: Enable hs400 new sample method

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.3.6.24 0x0078 SMHC Hardware Reset Register (Default Value: 0x0000_0001)

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

4.3.6.25 0x0080 SMHC IDMAC Control Register (Default Value: 0x0000_0000)

Offset: 0x0080			Register Name: SMHC_IDMAC
Bit	**Read/Write**	**Default/Hex**	**Description**
31	W	0x0	DES_LOAD_CTRL When IDMAC fetches a descriptor, if the valid bit of a descriptor is not set, IDMAC FSM will go to the suspend state. Setting this bit will make the IDMAC refetch descriptor again and do the transfer normally.
30:11	/	/	/
10:8	R	0x0	Reserved
7	R/W	0x0	IDMAC_ENB IDMAC Enable When set, the IDMAC is enabled.
6:2	R/W	0x0	Reserved
1	R/W	0x0	FIX_BUST_CTRL Fixed Burst Controls whether the AHB Master interface performs fixed burst transfers or not. When set, the AHB will use only SINGLE, INCR4, and INCR8 during the start of normal burst transfers. When reset, the AHB will use SINGLE and INCR burst transfer operations.
0	R/W	0x0	IDMAC_RST DMA Reset When set, the DMA Controller resets all its internal registers. SWR is read/write. It is automatically cleared after 1 clock cycle.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.3.6.26 0x0084 SMHC Descriptor List Base Address Register (Default Value: 0x0000_0000)

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

4.3.6.27 0x0088 SMHC IDMAC Status Register (Default Value: 0x0000_0000)

Offset: 0x0088			Register Name: SMHC_IDST_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16:13	R	0x0	Reserved
12:10	R	0x0	IDMAC_ERR_STA Error Bits Indicates the type of error that caused a Bus Error. Valid only with Fatal Bus Error bit (IDSTS[2]) set. This field does not generate an interrupt. 001: Host Abort received during the transmission. 010: Host Abort received during the reception. Others: Reserved
9	R/W1C	0x0	ABN_INT_SUM(AIS) Abnormal Interrupt Summary Logical OR of the following: IDSTS[2]: Fatal Bus Interrupt IDSTS[4]: Descriptor Unavailable Bit Interrupt IDSTS[5]: Card Error Summary Interrupt Only unmasked bits affect this bit. This is a sticky bit and must be cleared each time a corresponding bit that causes AIS to be set is cleared.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0088			Register Name: SMHC_IDST_REG
Bit	Read/Write	Default/Hex	Description
8	R/W1C	0x0	NOR_INT_SUM(NIS) Normal Interrupt Summary Logical OR of the following: IDSTS[0]: Transmit Interrupt 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.
7:6	/	/	/
5	R/W1C	0x0	ERR_FLAG_SUM Card Error Summary Indicates the status of the transaction to/from the card; also present in RINTSTS. Indicates the logical OR of the following bits: EBE: End Bit Error RTO: Response Timeout RCRC: Response CRC SBE: Start Bit Error DRTO: Data Read Timeout DCRC: Data CRC for Receive RE: Response Error
4	R/W1C	0x0	DES_UNAVL_INT Descriptor Unavailable Interrupt This bit is set when the descriptor is unavailable due to OWN bit = 0 (DES0[31] =0). Writing a 1 clears this bit.
3	/	/	/
2	R/W1C	0x0	FATAL_BERR_INT Fatal Bus Error Interrupt Indicates that a Bus Error occurred (IDSTS[12:10]). When this bit is set, the DMA disables all its bus accesses. Writing a 1 clears this bit.
1	R/W1C	0x0	RX_INT Receive Interrupt Indicates the completion of data reception for a descriptor.
0	R/W1C	0x0	TX_INT Transmit Interrupt Indicates that data transmission is finished for a descriptor.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.3.6.28 0x008C SMHC IDMAC Interrupt Enable Register (Default Value: 0x0000_0000)

Offset: 0x008C			Register Name: SMHC_IDIE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:6	/	/	/
5	R/W	0x0	ERR_SUM_INT_ENB Card Error Summary Interrupt Enable. When set, it enables the Card Interrupt Summary.
4	R/W	0x0	DES_UNAVL_INT_ENB Descriptor Unavailable Interrupt. When set along with Abnormal Interrupt Summary Enable, the Descriptor Unavailable Interrupt is enabled.
3	/	/	/
2	R/W	0x0	FERR_INT_ENB Fatal Bus Error Enable When set with Abnormal Interrupt Summary Enable, the Fatal Bus Error Interrupt is enabled. When reset, Fatal Bus Error Enable Interrupt is disabled.
1	R/W	0x0	RX_INT_ENB Receive Interrupt Enable. When set with Normal Interrupt Summary Enable, Receive Interrupt is enabled. When reset, the Receive Interrupt is disabled.
0	R/W	0x0	TX_INT_ENB Transmit Interrupt Enable. When set with Normal Interrupt Summary Enable, Transmit Interrupt is enabled. When reset, the Transmit Interrupt is disabled.

4.3.6.29 0x0100 SMHC Card Threshold Control Register (Default Value: 0x0000_0000)

Offset: 0x0100			Register Name: SMHC_THLD
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	/	/	/
27:16	R/W	0x0	CARD_WR_THLD Card Read/Write Threshold Size
15:3	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

4.3.6.30 0x0104 SMHC Sample FIFO Control Register (Default Value: 0x0000_0006)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0104			Register Name: SMHC_SFC
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	BYPASS_EN Bypass enable When set, sample FIFO will be bypassed.

4.3.6.31 0x0108 SMHC Auto Command 23 Argument Register (Default Value: 0x0000_0000)

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

4.3.6.32 0x010C SMHC eMMC4.5 DDR Start Bit Detection Control Register (Default Value: 0x0000_0000)

Offset: 0x010C			Register Name: EMMC_DDR_SBIT_DET
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	HS400_MD_EN HS400 Mode Enable 0: Disabled 1: Enabled It is required to set this bit to '1' before initiating any data transfer CMD in HS400 mode.
30:1	/	/	/
0	R/W	0x0	HALF_START_BIT Control for start bit detection mechanism inside mstorage based on duration of start bit. For eMMC 4.5, start bit can be: 0: Full cycle 1: Less than one full cycle Set HALF_START_BIT = 1 for eMMC 4.5 and above; set to 0 for SD applications.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.3.6.33 0x0138 SMHC Extended Command Register (Default Value: 0x0000_0000)

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

4.3.6.34 0x013C SMHC Extended Response Register (Default Value: 0x0000_0000)

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

4.3.6.35 0x0140 SMHC Drive Delay Control Register (Default Value: 0x0001_0000)

Offset: 0x0140			Register Name: SMHC_DRV_DL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17	R/W	0x0	DAT_DRV_PH_SEL Data Drive Phase Select **When 0x005C[31]=0:** 0: Data drive phase offset is 90° at SDR mode, 45° at DDR8 mode, and 90° at DDR4 mode. 1: Data drive phase offset is 180° at SDR mode, 90° at DDR8 mode, and 0° at DDR4 mode. **When 0x005C[31]=1:** 0: Data drive phase offset is 90° at SDR mode, and 45° at DDR mode. 1: Data drive phase offset is 180° at SDR mode, and 90° at DDR mode.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0140			Register Name: SMHC_DRV_DL
Bit	Read/Write	Default/Hex	Description
16	R/W	0x1	CMD_DRV_PH_SEL Command Drive Phase Select **When 0x005C[31]=0:** 0: Command drive phase offset is 90° at SDR mode, 45° at DDR8 mode, and 90° at DDR4 mode. 1: Command drive phase offset is 180° at SDR mode, 90° at DDR8 mode, and 180° at DDR4 mode. **When 0x005C[31]=1:** 0: Command drive phase offset is 90° at SDR mode and 45° at DDR mode. 1: Command drive phase offset is 180° at SDR mode and 90° at DDR mode.
15:0	/	/	/

4.3.6.36 0x0144 SMHC Sample Delay Control Register (Default Value: 0x0000_2000)

Offset: 0x0144			Register Name: SMHC_SAMP_DL
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15	R/W	0x0	SAMP_DL_CAL_START Sample Delay Calibration Start When set, it means that start sample delay chain calibration.
14	R	0x0	SAMP_DL_CAL_DONE Sample Delay Calibration Done When set, it means that sample delay chain calibration is done and the result of calibration is shown in SAMP_DL.
13:8	R	0x20	SAMP_DL Sample Delay It indicates the number of delay cells corresponding to the current card clock. The delay time generated by these delay cells is equal to the cycle of the card clock nearly. Generally, it is necessary to do drive delay calibration when the card clock is changed. This bit is valid only when SAMP_DL_CAL_DONE is set.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0144			Register Name: SMHC_SAMP_DL
Bit	**Read/Write**	**Default/Hex**	**Description**
7	R/W	0x0	SAMP_DL_SW_EN Sample Delay Software Enable When set, it means that enable sample delay specified at SAMP_DL_SW.
6	/	/	/
5:0	R/W	0x0	SAMP_DL_SW Sample Delay Software The relative delay between the clock line and command line, data line. It can be determined according to the value of SAMP_DL, the cycle of the card clock and the input timing requirement of the device.

4.3.6.37 0x0148 SMHC Data Strobe Delay Control Register (Default Value: 0x0000_2000)

Offset: 0x0148			Register Name: SMHC_DS_DL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R/W	0x0	DS_DL_CAL_START Data Strobe Delay Calibration Start When set, it means that start sample delay chain calibration.
14	R	0x0	DS_DL_CAL_DONE Data Strobe Delay Calibration Done When set, it means that sample delay chain calibration is done and the result of calibration is shown in DS_DL.
13:8	R	0x20	DS_DL Data Strobe Delay It indicates the number of delay cells corresponding to the current card clock. The delay time generated by these delay cells is equal to the cycle of SMHC's clock nearly. This bit is valid only when SAMP_DL_CAL_DONE is set.
7	R/W	0x0	DS_DL_SW_EN Sample Delay Software Enable
6	/	/	/
5:0	R/W	0x0	DS_DL_SW Data Strobe Delay Software

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4.3.6.38 0x014C SMHC HS400 New Timing Delay Control Register (Default Value: 0x0000_8000)

Offset: 0x014C			Register Name: SMHC_HS400_DL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R/W	0x0	HS400_DL_CAL_START HS400 Delay Calibration Start When set, start sample delay chain calibration.
14	R	0x0	HS400_DL_CAL_DONE HS400 Delay Calibration Done When set, it means that sample delay chain calibration is done and the result of calibration is shown in HS400_DL.
13:12	/	/	/
11:8	R	0x8	HS400_DL HS400 Delay It indicates the number of delay cells corresponding to the current card clock. The delay time generated by these delay cells is equal to the cycle of the SMHC clock nearly. This bit is valid only when HS400_DL_CAL_DONE is set.
7	R/W	0x0	HS400_DL_SW_EN Sample Delay Software Enable
6	/	/	/
3:0	R/W	0x0	HS400_DL_SW HS400 Delay Software

4.3.6.39 0x0200 SMHC FIFO Register (Default Value: 0x0000_0000)

Offset: 0x0200			Register Name: SMHC_FIFO
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	TX/RX_FIFO Data FIFO

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Contents

5 Audio 375
5.1 I2S/PCM 375
5.1.1 Overview 375
5.1.2 Block Diagram 376
5.1.3 Functional Descriptions 377
5.1.4 Programming Guidelines 384
5.1.5 Register List 386
5.1.6 Register Description 387
5.2 DMIC 436
5.2.1 Overview 436
5.2.2 Block Diagram 436
5.2.3 Functional Descriptions 436
5.2.4 Register List 439
5.2.5 Register Description 439
5.3 OWA 454
5.3.1 Overview 454
5.3.2 Block Diagram 455
5.3.3 Functional Descriptions 455
5.3.4 Register List 460
5.3.5 Register Description 461
5.4 Audio Codec 477
5.4.1 Overview 477
5.4.2 Block Diagram 477
5.4.3 Functional Descriptions 478
5.4.4 Programming Guidelines 488
5.4.5 Register List 488
5.4.6 Register Description 492

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figures

Figure 5-1 I2S/PCM Interface System Block Diagram 376
Figure 5-2 Typical Application of I2S/PCM Interface 376
Figure 5-3 I2S Standard Mode Timing 378
Figure 5-4 Left-Justified Mode Timing 379
Figure 5-5 Right-Justified Mode Timing 379
Figure 5-6 PCM Long Frame Mode Timing 379
Figure 5-7 PCM Short Frame Mode Timing 380
Figure 5-8 16-Channel Mapping Configuration 381
Figure 5-9 Timing Requirements for Inputs 382
Figure 5-10 Timing Requirements for Outputs 382
Figure 5-11 I2S/PCM Operation Flow 383
Figure 5-12 DMIC Block Diagram 436
Figure 5-13 DMIC Operation Mode 438
Figure 5-14 OWA Block Diagram 455
Figure 5-15 OWA Biphase-Mark Code 457
Figure 5-16 OWA Sub-Frame Format 458
Figure 5-17 OWA Frame/Block Format 459
Figure 5-18 OWA Operation Flow 459
Figure 5-19 Audio Codec Block Diagram 478
Figure 5-20 Audio Codec Clock Diagram 480
Figure 5-21 Audio Codec Digital Part Reset System 480
Figure 5-22 Audio Codec Analog Part Reset System 481
Figure 5-23 Audio Codec Data Path Diagram 482
Figure 5-24 Audio Codec Interrupt System 484
Figure 5-25 DAP Data Flow 484
Figure 5-26 HPF Logic Structure 485
Figure 5-27 DRC Block Diagram 485
Figure 5-28 DRC Static Curve Parameters 485
Figure 5-29 Energy Filter Structure 486
Figure 5-30 Gain Smooth Filter 487

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Tables

Table 5-1 I2S/PCM External Signals 377
Table 5-2 I2S/PCM Clock Sources 378
Table 5-3 DIN Slot ID and Encoder 380
Table 5-4 Proper MCLK Values with Different Fsin and Fsout 382
Table 5-5 DMIC External Signals 437
Table 5-6 DMIC Clock Sources 437
Table 5-7 OWA Sub-blocks 455
Table 5-8 OWA External Signals 456
Table 5-9 OWA Clock Sources 456
Table 5-10 Biphase-Mark Encoder 457
Table 5-11 Preamble Codes 458

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5 Audio

5.1 I2S/PCM

5.1.1 Overview

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

The I2S/PCM controller includes the following features:

- Three I2S/PCM external interfaces (I2S0, I2S1, S-I2S0) for connecting external power amplifier and MIC ADC
- I2S0 supports 4 data channels; I2S1 and S-I2S0 support 2 data channels
- S-I2S0 supports asynchronous sample rate conversion (ASRC)
- Compliant with standard Philips Inter-IC sound (I2S) bus specification
- Compliant with Left-justified, Right-justified, PCM mode, and TDM (Time Division Multiplexing) format
- TDM mode supports maximum 16 channels output and 16 channels input
- Supports full-duplex synchronous work mode
- Supports Master/Slave mode
- Adjustable interface voltage
- Supports clock up to 49.152 MHz data output of I2S/PCM in Master mode (Only if the IO PAD and peripheral I2S/PCM satisfy timing parameters)
- Supports clock up to 12.288 MHz data input of I2S/PCM in Master mode (Only if the IO PAD and peripheral I2S/PCM satisfy timing parameters)
- Adjustable audio sample resolution from 8-bit to 32-bit
- Sample rate from 8 kHz to 384 kHz (channels=2)
- Supports 8-bit u-law and 8-bit A-law companded sample
- One 128 depth x 32-bit width TXFIFO for data transmit, one 64 depth x 32-bit width RXFIFO for data receive
- Programmable PCM frame width: 1 BCLK width (short frame) and 2 BCLKs width (long frame)
- Programmable FIFO thresholds

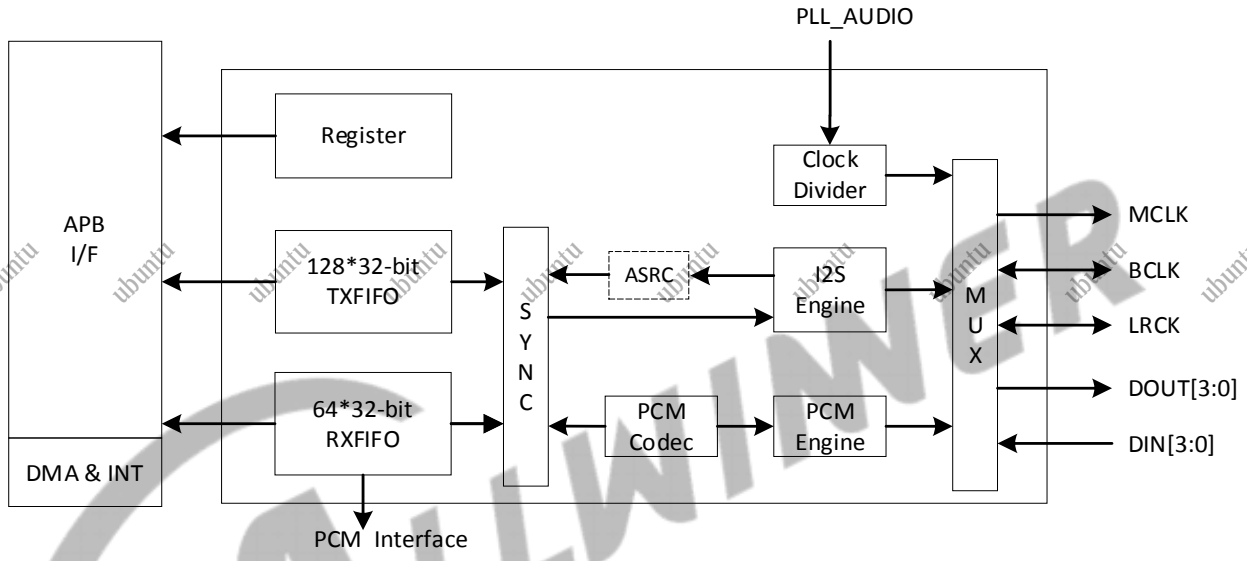
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

- Supports interrupts and DMA
- Supports loopback mode for test

5.1.2 Block Diagram

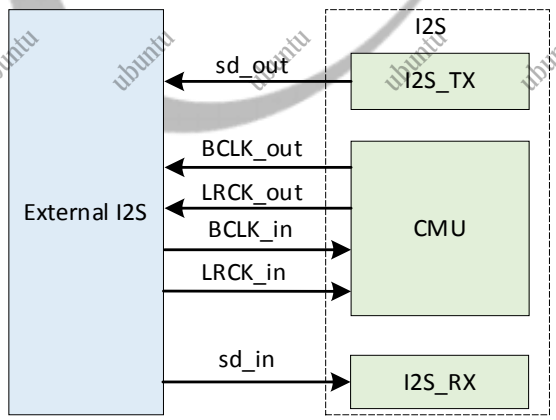
The following figure shows the functional block diagram of the I2S/PCM interface.

Figure 5-1 I2S/PCM Interface System Block Diagram



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

Figure 5-2 Typical Application of I2S/PCM Interface



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

- The I2S_TX is for playing music in I2S or PCM format.
- The I2S_RX is for receiving data in I2S or PCM format.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

- When the I2S works in the master mode, the external I2S module provides BCLK_in and LRCK_in for the clock management unit (CMU), and the I2S_TX and I2S_RX work with the two external clocks.
- When the I2S works in the slave mode, the CMU provides clocks BCLK_out and LRCK_out for the external I2S module, and the I2S_TX and I2S_RX work with the internal clocks.

5.1.3 Functional Descriptions

5.1.3.1 External Signals

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

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

MCLK is an output pin for external devices. DOUT are the serial data output pins and DIN are the serial data input pins. For details about General Purpose I/O port, refer to section 7.7 "GPIO".

Table 5-1 I2S/PCM External Signals

Signal Name	Description	Type
I2S0-MCLK	I2S0 Master Clock	O
I2S0-LRCK	I2S0/PCM0 Sample Rate Clock/Sync	I/O
I2S0-BCLK	I2S0/PCM0 Sample Rate Clock	I/O
I2S0-DOUT[3:0]	I2S0/PCM0 Serial Data Output Channel [3:0]	O
I2S0-DIN[1:0]	I2S0/PCM0 Serial Data Input Channel [1:0]	I
I2S1-MCLK	I2S1 Master Clock	O
I2S1-LRCK	I2S1/PCM1 Sample Rate Clock/Sync	I/O
I2S1-BCLK	I2S1/PCM1 Sample Rate Clock	I/O
I2S1-DOUT[1:0]	I2S1/PCM1 Serial Data Output Channel [1:0]	O
I2S1-DIN[1:0]	I2S1/PCM1 Serial Data Input Channel [1:0]	I
S-I2S0-MCLK	S-I2S0 Master Clock	O
S-I2S0-LRCK	S-I2S0/PCM0 Sample Rate Clock/Sync	I/O
S-I2S0-BCLK	S-I2S0/PCM0 Sample Rate Clock	I/O
S-I2S0-DOUT[1:0]	S-I2S0/PCM0 Serial Data Output Channel [1:0]	O
S-I2S0-DIN[1:0]	S-I2S0/PCM0 Serial Data Input Channel [1:0]	I

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.1.3.2 Clock Sources

The following table describes the clock sources for I2S/PCM. For clock setting, configurations, and gating information, refer to section 3.3 "CCU".

Table 5-2 I2S/PCM Clock Sources

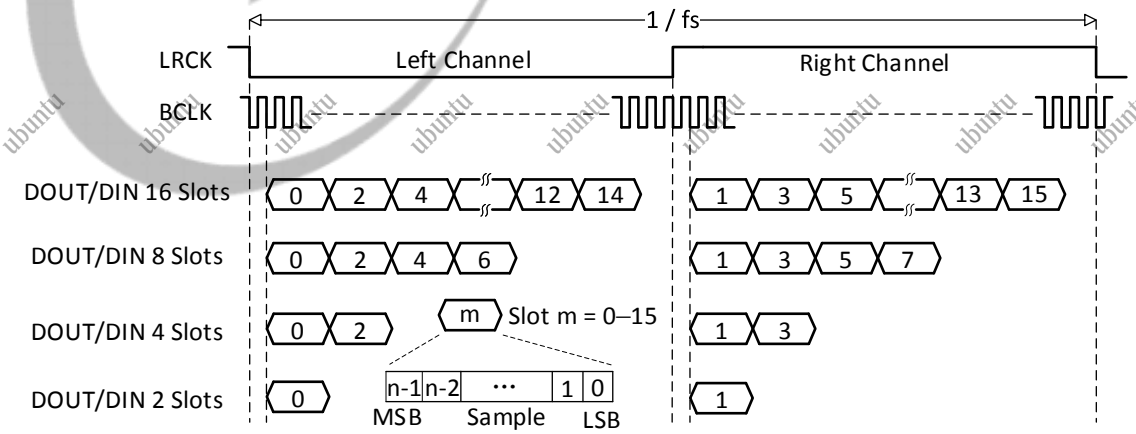
Clock Name	Description
PLL_AUDIO1(1X)	By default, PLL_AUDIO1(1X) is 24.5714 MHz, and PLL_AUDIO1(4X) is 98.2856 MHz.
PLL_AUDIO1(4X)	
PLL_AUDIO0(DIV2)	By default, PLL_AUDIO0 is 3072 MHz, PLL_AUDIO0(DIV2) is 1536 MHz, and PLL_AUDIO0(DIV5) is 614.4 MHz (24.576 MHz*25).
PLL_AUDIO0(DIV5)	

5.1.3.3 Timing Diagram

The I2S/PCM supports standard I2S mode, Left-justified I2S mode, Right-justified I2S mode, PCM mode, and TDM mode. The software can select the modes by setting I2S/PCM_CTL. The following figures describe the waveforms for SYNC, BCLK, DOUT, and DIN in different modes.

Each sampling period contains an LRCK. The low level of LRCK is the left channel corresponding to the even slots, and the high level is the right channel corresponding to the odd slots. Each slot is the sampling point of a mono channel. The sampling period can support the transmission of 2/4/8/16 slots. The BCLK corresponds to the serial data bit.

Figure 5-3 I2S Standard Mode Timing



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 5-4 Left-Justified Mode Timing

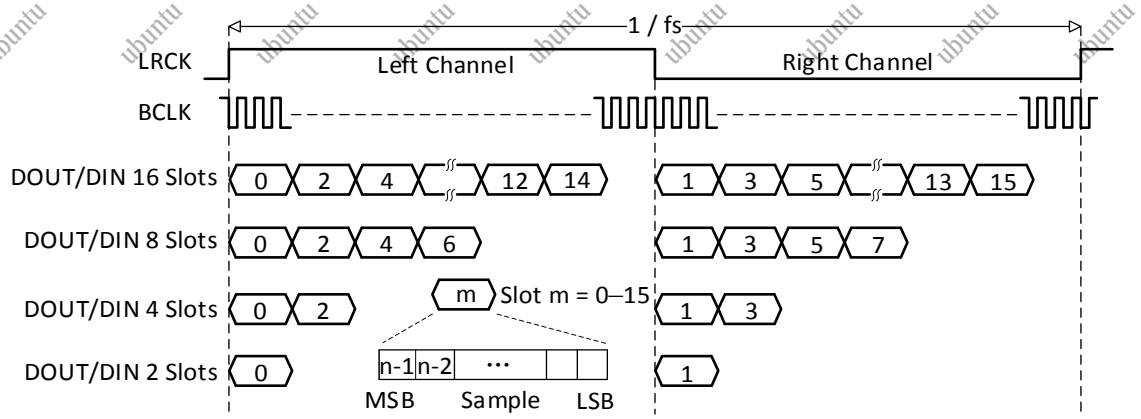


Figure 5-5 Right-Justified Mode Timing

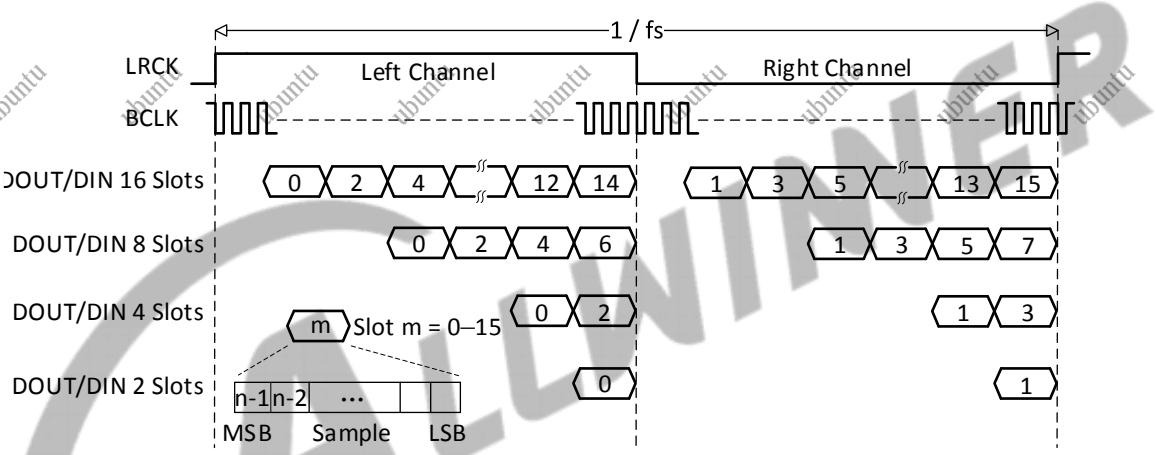
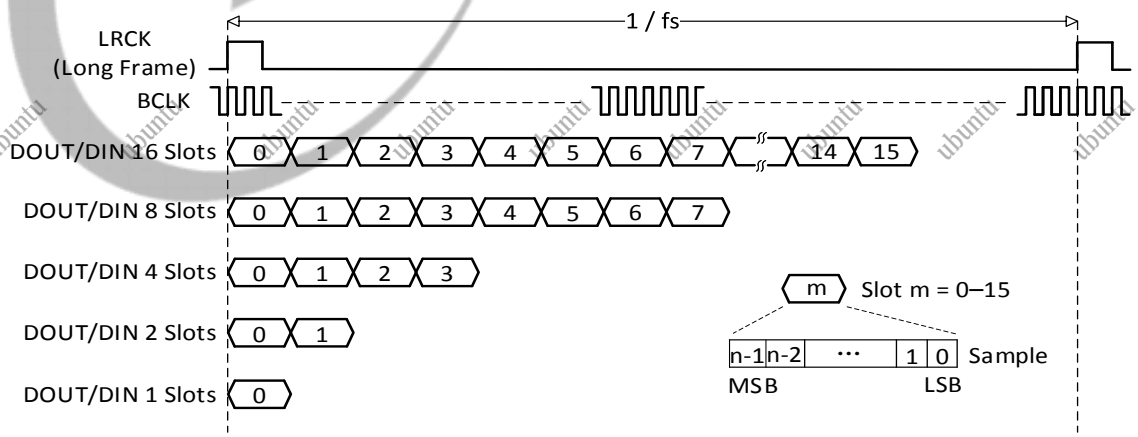
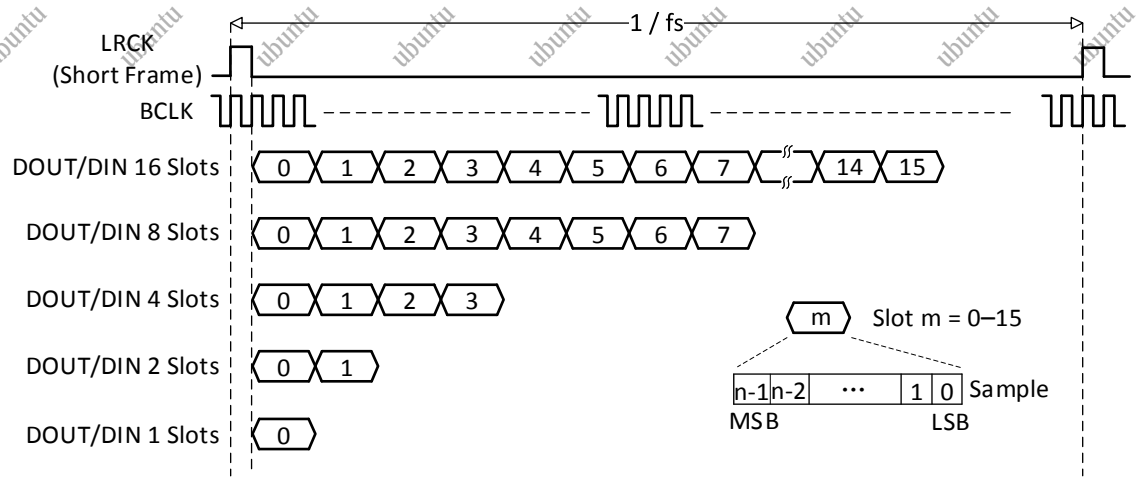


Figure 5-6 PCM Long Frame Mode Timing



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 5-7 PCM Short Frame Mode Timing



5.1.3.4 DIN Slot Mapping

The 4-wire DIN has 64 slots, each wire DIN has 16 slots. However, only 16 slots are valid and act as the RX channels.

The following table shows the relationship between the slot id and encoder.

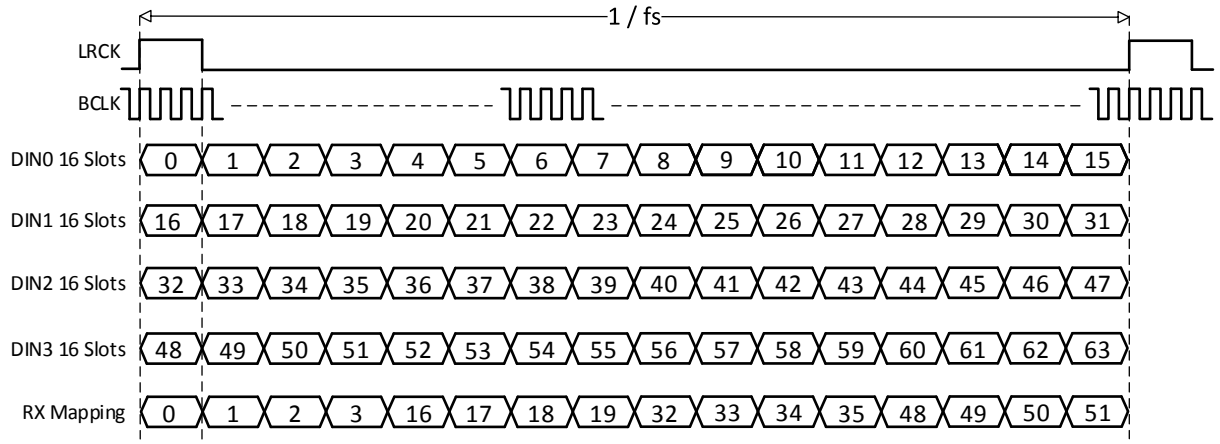
Table 5-3 DIN Slot ID and Encoder

DIN0 Slot ID	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
DIN1 Slot ID	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
DIN2 Slot ID	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
DIN3 Slot ID	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

There are 16 channels mapping configuration, each wire selects four slots for RX. The following figure shows the 16-channel mapping configuration.

Figure 5-8 16-Channel Mapping Configuration



5.1.3.5 ASRC

The ASRC module supports sampling rate conversion between the up-sampling and down-sampling. The ASRC also supports sampling rate conversion between the dual-channel audio data, and the size of the sampling data is up to 24 bits.

The ASRC module has the following features:

- Typical THD + N: -130 dB (Range: -125 dB to -139 dB)
- Supports sampling rate conversion between the up-sampling and down-sampling to implement the sampling rate conversion for stereo data
  - The up-sampling ratio ranges from 1 to 7.5x
  - The down-sampling ratio ranges from 8 to 1x
- Supports sampling rate conversion between two identical frequencies
- Sampling rate for both the input and output range is from 8 kHz to 192 kHz and can be decimal
- Sampling rate can be configured manually or via adaptive generation
- The ASRC input is connected to I2S RX_FIFO_WDATA [31:8], and the input data is 24-bit MSB big-endian. For the input data that is less than 24 bits, use zeros to pad out the values at the low bits instead of high bits
- The ASRC needs some time to calculate the result. The output outsamplea/b will keep 0 during the calculation, and then change to the valid value when the result comes out

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Calculating the ASRC Latency

Calculate the ASRC up-sampling and down-sampling latency according to the following formulas.

UpsamplingLatency=PhaseDelay+FIFODelay=32+16=48InputSamplePeriods

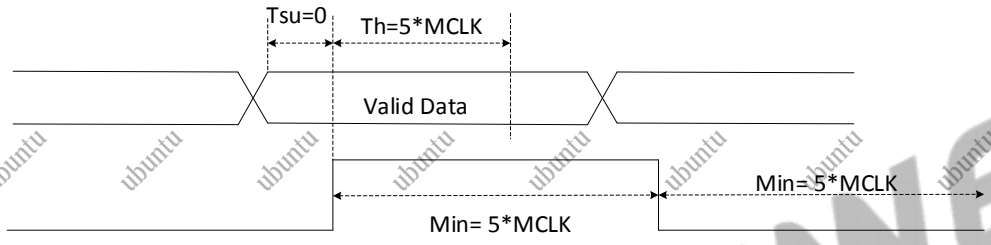
DownsamplingLatency=PhaseDelay+FIFODelay=(32*f_{sout}/f_{sin})+16InputSamplePeriods

ASRC Timing

The MCLK samples the input clock CLKIN to generate pulse signals.

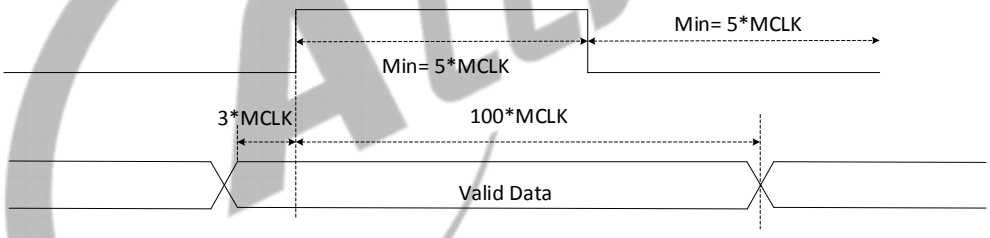
The following figure shows the timing requirements for the inputs.

Figure 5-9 Timing Requirements for Inputs



The following figure shows the timing requirements for the outputs.

Figure 5-10 Timing Requirements for Outputs



For the up-sampling, $F_{MCLK}=F_{sout}*1350$.

For the down-sampling, $F_{MCLK}=F_{sin}*0.30+F_{sout}*295$.

The following table provides the proper values of MCLK in MHz with different Fsin and Fsout in kHz.

Table 5-4 Proper MCLK Values with Different Fsin and Fsout

Fsin \ Fsout	32	44.1	48	88.2	96	144	192
32	45	60	65	120	130	195	260
44.1	55	60	65	120	130	195	260
48	60	65	65	120	130	195	260
88.2	105	105	110	120	130	195	260
96	110	115	115	125	130	195	260

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Fsout / Fsin	32	44.1	48	88.2	96	144	192
144	160	165	165	175	180	195	260
192	210	215	215	225	230	245	260

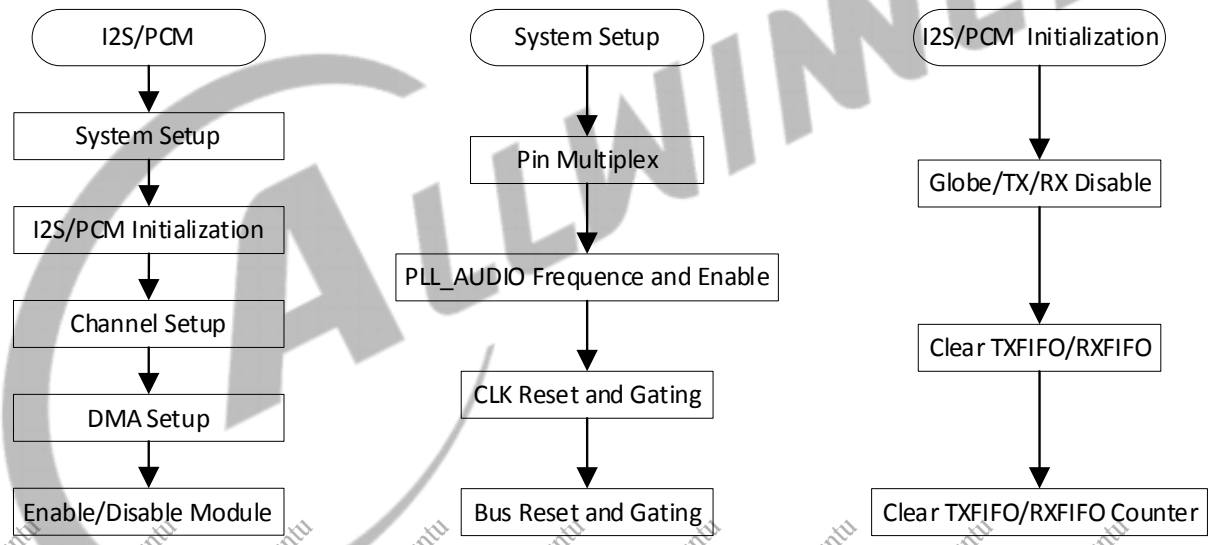
Note: The units for Fsin and Fsout are kHz and MCLK is MHz.

5.1.3.6 Operation Modes

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

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

Figure 5-11 I2S/PCM Operation Flow



1. System Setup and I2S/PCM Initialization

The clock source for the I2S/PCM should be followed. Firstly, disable the PLL_AUDIO through PLL_AUDIOx Control Register[PLL_ENABLE] in the CCU. Secondly, set up the frequency of the PLL_AUDIO in the PLL_AUDIOx Control Register. After that, enable the I2S/PCM gating through the **I2S/PCMx_CLK_REG** when you checkout that the PLL_AUDIOx Control Register[LOCK] becomes to 1. At last, reset and enable the I2S/PCM bus gating by setting **I2S/PCM_BGR_REG**.

After the system setup, the register of I2S/PCM can be setup. Firstly, initialize the I2S/PCM. You should close the Globe Enable bit (**I2S/PCM_CTL**[0]), Transmitter Block Enable bit (**I2S/PCM_CTL**[2]), and Receiver Block Enable bit (**I2S/PCM_CTL**[1]) by writing 0 to it. After that, clear the TX/RX FIFO by writing 0 to the

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

bit[25:24] of I2S/PCM_FCTL. At last, you can clear the TX FIFO and RX FIFO counter by writing 0 to I2S/PCM_TXCNT and I2S/PCM_RXCNT.

2. **Channel Setup and DMA Setup**

First, you can set up the I2S/PCM of master and slave. The configuration can be referred to the protocol of I2S/PCM. Then, you can set up the translation mode, the sample resolution, the wide of the slot, the channel slot number, and the trigger level, and so on. The setup of the register can be found in the specification.

The I2S/PCM supports two methods to transfer the data. The most common way is DMA, the setup of DMA can be found in the **"**DMA**"**. In this module, you just enable the DRQ.

3. **Enable and Disable I2S/PCM**

To enable the function, you can enable TX/RX by writing I2S/PCM_CTL[TXEN]/I2S/PCM_CTL[RXEN]. After that, enable I2S/PCM by writing 1 to I2S/PCM_CTL[Globe Enable]. Write 0 to the Globe Enable bit to disable I2S/PCM.

5.1.4 Programming Guidelines

5.1.4.1 Application Example of Processing ASRC Input and Output Data

The following example shows a typical application of ASRC: the input data is 24-bit valid, and the output data is a 32-bit data whose highest 24 bits are valid output and the lowest eight bits are padded out with zeros.

To implement the application, configure the sample resolution and slot width as 32 bits. Follow the steps below:

Step 1 For the input register: 0x04 [6:4] sample_res = 3`h7, 0x04 [2:0] slot_width = 3`h7.

The format of the input data: 32'hXXXXXXXX, where, bit[31] is the MSB and X is the valid data bit.

Step 2 For the output register: 0x04 [6:4] sample_res = 3`h7, 0x04 [2:0] slot_width = 3`h7

The format of the output data: 32'hXXXXXX00, where, bit[31] is the MSB, X is the valid data bit, and bit[7:0] are the padded zeros.

5.1.4.2 Converting the Sampling Rate with ASRC

Converting a 48 kHz sampling rate to 16 kHz is the most common scenario in actual applications. Follow the steps below to convert the sampling rate from 48 kHz to 16 kHz for the 32-bit data.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

1. Configure the PLL_AUDIO Register

a) Configure the PLL_AUDIO1_CTRL_REG[31:0] as 0x8814AB01. That is, PLL_AUDIO1 = 24*(171+1)/(1+1)/(1+0)/(1+20) = 98.286 MHz. According to the relationship among the Fsin, Fsout, and MCLK, the MCLK should be greater than 60 MHz. In the simulation phase, the HOSC frequency is 25 MHz, so the output frequency of PLL_AUDIO1 should be 25*(171+1)/(1+1)/(1+0)/(1+20) = 102.381 MHz. In the IC test phase, configure the frequency of PLL_AUDIO1 according to its actual output frequency.

b) It is suggested that you configure the ASRC MCLK as an equal-duty-cycle signal. You can specify an odd number for bit[21:16] (PLL_POST_DIV_P) of PLL_AUDIO1_CTRL_REG to get an equal-duty-cycle output clock of PLL_AUDIO1.

c) Configure bit[25:24] of R_I2S/PCM0_ASRC_CLK_REG as 0x11 to select the PLL_AUDIO(4X).

2. Configure the I2S Registers

a) Configure bit[7:4] (BCLKDIV) of I2S/PCM_CLKD as 4`h9 so that the BCLK will be PLL_AUDIO divided by 32, that is, the frequency of BCLK will be 98.286 MHz/32 = 3.072 MHz.

b) Configure bit[17:8] (LRCK_PERIOD) of I2S/PCM_FMT0 as 10`h1F. That is, the LRCK_PERIOD width is configured as 32 BLCKs and can generate the ASRC CLKIN with a 48 kHz sampling rate. $(\frac{3.072\text{MHz}}{32*2}=48\text{kHz})$

c) Configure bit[6:4] (Sample Resolution bits) of I2S/PCM_FMT0 as 3`h7 to specify the sample resolution as 32-bit.

d) Configure bit[2:0] (Slot Width bits) of I2S/PCM_FMT0 as 3`h7 to specify the slot width as 32-bit.

3. Configure the ASRC Registers

a) Configure bit[16] (clock gate) of MCLKCFG as 1`h1 to open the clock gating.

b) Configure bit[3:0] (division factor) of MCLKCFG as 1`h1 to specify the division factor as 1。

c) Configure bit[20] (clock gate) of F_{SOUT}CFG as 1`h1 to open the clock gating.

d) Configure bit[19:16] (clock select) of F_{SOUT}CFG as 4`h0 to select I2S0_ASRC_CLK as the clock source.

e) Configure bit[7:4] (the first division factor) of F_{SOUT}CFG as 16`h13 to configure the first division factor as 128.

f) Configure bit[3:0] (the second division factor) of F_{SOUT}CFG as 16`h10 to configure the second division factor as 48.

g) Configure the ASRC ratio.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

To configure the ASRC ratio manually, configure bit[31] (Manual Configuration of ASRC Ratio Enable) of ASRCMANCFG as 1`h1 to enable the manual configuration of ASRC ratio. Configure bit[25:0] of ASRCMANCFG as 26`h155555 to specify the ratio value as 0x155555. The calculation formula for the ratio value: Dec2Hex (Fsout/Fsin)*2^{22}). In this example, Fsout/Fsin = 16 kHz/48 kHz =1/3, then the ratio is 0x155555.

To configure the ASRC ratio automatically, configure bit[31] (Manual Configuration of ASRC Ratio Enable) of ASRCMANCFG as 1`h0 to enable the automatic configuration of ASRC ratio. Then the system will automatically calculate the ratio value based on the MCLK, Fsout, and Fsin.

5.1.5 Register List

Module Name	Base Address
I2S/PCM0	0x02004000
I2S/PCM1	0x02005000
R_I2S/PCM0	0x07033000

Register Name	Offset	Description
I2S/PCM_CTL	0x0000	I2S/PCM Control Register
I2S/PCM_FMT0	0x0004	I2S/PCM Format Register 0
I2S/PCM_FMT1	0x0008	I2S/PCM Format Register 1
I2S/PCM_ISTA	0x000C	I2S/PCM Interrupt Status Register
I2S/PCM_RXFIFO	0x0010	I2S/PCM RXFIFO Register
I2S/PCM_FCTL	0x0014	I2S/PCM FIFO Control Register
I2S/PCM_FSTA	0x0018	I2S/PCM FIFO Status Register
I2S/PCM_INT	0x001C	I2S/PCM DMA & Interrupt Control Register
I2S/PCM_TXFIFO	0x0020	I2S/PCM TXFIFO Register
I2S/PCM_CLKD	0x0024	I2S/PCM Clock Divide Register
I2S/PCM_TXCNT	0x0028	I2S/PCM TX Sample Counter Register
I2S/PCM_RXCNT	0x002C	I2S/PCM RX Sample Counter Register
I2S/PCM_CHCFG	0x0030	I2S/PCM Channel Configuration Register
I2S/PCM_TX0CHSEL	0x0034	I2S/PCM TX0 Channel Select Register
I2S/PCM_TX1CHSEL	0x0038	I2S/PCM TX1 Channel Select Register
I2S/PCM_TX2CHSEL	0x003C	I2S/PCM TX2 Channel Select Register
I2S/PCM_TX3CHSEL	0x0040	I2S/PCM TX3 Channel Select Register
I2S/PCM_TX0CHMAP0	0x0044	I2S/PCM TX0 Channel Mapping Register0

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
I2S/PCM_TX0CHMAP1	0x0048	I2S/PCM TX0 Channel Mapping Register1
I2S/PCM_TX1CHMAP0	0x004C	I2S/PCM TX1 Channel Mapping Register0
I2S/PCM_TX1CHMAP1	0x0050	I2S/PCM TX1 Channel Mapping Register1
I2S/PCM_TX2CHMAP0	0x0054	I2S/PCM TX2 Channel Mapping Register0
I2S/PCM_TX2CHMAP1	0x0058	I2S/PCM TX2 Channel Mapping Register1
I2S/PCM_TX3CHMAP0	0x005C	I2S/PCM TX3 Channel Mapping Register0
I2S/PCM_TX3CHMAP1	0x0060	I2S/PCM TX3 Channel Mapping Register1
I2S/PCM_RXCHSEL	0x0064	I2S/PCM RX Channel Select Register
I2S/PCM_RXCHMAP0	0x0068	I2S/PCM RX Channel Mapping Register0
I2S/PCM_RXCHMAP1	0x006C	I2S/PCM RX Channel Mapping Register1
I2S/PCM_RXCHMAP2	0x0070	I2S/PCM RX Channel Mapping Register2
I2S/PCM_RXCHMAP3	0x0074	I2S/PCM RX Channel Mapping Register3
MCLKCFG	0x0080	ASRC MCLK Configuration Register
FsoutCFG	0x0084	ASRC Out Sample Rate Configuration Register
FsinEXTCFG	0x0088	ASRC Input Sample Pulse Extend Configuration Register
ASRCCFG	0x008C	ASRC Enable Register
ASRCMANCFG	0x0090	ASRC Manual Ratio Configuration Register
ASRCRATIOSTAT	0x0094	ASRC Status Register
ASRCFIFOSTAT	0x0098	ASRC FIFO Level Status Register
ASRCMBISTCFG	0x009C	ASRC MBIST Test Configuration Register
ASRCMBISTSTAT	0x00A0	ASRC MBIST Test Status Register

5.1.6 Register Description

5.1.6.1 0x0000 I2S/PCM Control Register (Default Value: 0x0006_0000)

Offset: 0x0000			Register Name: I2S/PCM_CTL
Bit	Read/Write	Default/Hex	Description
31:22	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: I2S/PCM_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
21	R/W	0x0	RX_SYNC_EN_START The bit takes effect only when RX_SYNC_EN is set to 1. CPUX Domain: I2S0/I2S1/OWA RX Synchronize Enable Start. CPUS Domain: R_I2S0/Audiocodec/DMIC RX Synchronize Enable Start. 0: Disabled 1: Enabled
20	R/W	0x0	RX_SYNC_EN I2S RX Synchronize Enable 0: Disabled 1: Enabled
19	/	/	/
18	R/W	0x1	BCLK_OUT Bit Clock Direction Select 0: Input 1: Output
17	R/W	0x1	LRCK_OUT LRCK Direction Select 0: Input 1: Output
16:12	/	/	/
11	R/W	0x0	DOUT3_EN Data Output3 Enable 0: Disabled, Hi-Z State 1: Enabled
10	R/W	0x0	DOUT2_EN Data Output2 Enable 0: Disabled, Hi-Z State 1: Enabled
9	R/W	0x0	DOUT1_EN Data Output1 Enable 0: Disabled, Hi-Z State 1: Enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: I2S/PCM_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
8	R/W	0x0	DOUT0_EN Data Output0 Enable 0: Disabled, Hi-Z State 1: Enabled
7	R/W	0x0	Reserved
6	R/W	0x0	OUT_MUTE Data Output Mute Enable 0: Normal Transfer 1: Force DOUT to output 0
5:4	R/W	0x0	MODE_SEL Mode Selection 00: PCM Mode (offset 0: Long Frame; offset 1: Short Frame) 01: Left Mode (offset 0: LJ Mode; offset 1: I2S Mode) 10: Right-Justified Mode 11: Reserved
3	R/W	0x0	LOOP Loopback Test 0: Normal Mode 1: Loopback Test When set to '1', the bit indicates that the DOUT is connected to the DIN.
2	R/W	0x0	TXEN Transmitter Block Enable 0: Disabled 1: Enabled
1	R/W	0x0	RXEN Receiver Block Enable 0: Disabled 1: Enabled
0	R/W	0x0	GEN Globe Enable 0: Disabled 1: Enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.1.6.2 0x0004 I2S/PCM Format Register 0 (Default Value: 0x0000_0033)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0004			Register Name: I2S/PCM_FMT0
Bit	**Read/Write**	**Default/Hex**	**Description**
6:4	R/W	0x3	SR Sample Resolution 000: Reserved 001: 8-bit 010: 12-bit 011: 16-bit 100: 20-bit 101: 24-bit 110: 28-bit 111: 32-bit
3	R/W	0x0	EDGE_TRANSFER Edge Transfer 0: DOUT drives data and DIN sample data at the different BCLK edge 1: DOUT drives data and DIN sample data at the same BCLK edge BCLK_POLARITY = 0, EDGE_TRANSFER = 0, DIN sample data at positive edge; BCLK_POLARITY = 0, EDGE_TRANSFER = 1, DIN sample data at negative edge; BCLK_POLARITY = 1, EDGE_TRANSFER = 0, DIN sample data at negative edge; BCLK_POLARITY = 1, EDGE_TRANSFER = 1, DIN sample data at positive edge.
2:0	R/W	0x3	SW Slot Width Select 000: Reserved 001: 8-bit 010: 12-bit 011: 16-bit 100: 20-bit 101: 24-bit 110: 28-bit 111: 32-bit

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.1.6.3 0x0008 I2S/PCM Format Register 1 (Default Value: 0x0000_0030)

Offset: 0x0008			Register Name: I2S/PCM_FMT1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R/W	0x0	RX MLS MSB/LSB First Select 0: MSB First 1: LSB First
6	R/W	0x0	TX MLS MSB/LSB First Select 0: MSB First 1: LSB First
5:4	R/W	0x3	SEXT Sign Extend in Slot [Sample Resolution < Slot Width] 00: Zeros or audio gain padding at LSB position 01: Sign extension at MSB position 10: Reserved 11: Transfer 0 after each sample in each Slot
3:2	R/W	0x0	RX_PDM PCM Data Mode 00: Linear PCM 01: Reserved 10: 8-bit u-law 11: 8-bit A-law
1:0	R/W	0x0	TX_PDM PCM Data Mode 00: Linear PCM 01: Reserved 10: 8-bit u-law 11: 8-bit A-law

5.1.6.4 0x000C I2S/PCM Interrupt Status Register (Default Value: 0x0000_0010)

Offset: 0x000C			Register Name: I2S/PCM_ISTA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:7	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x000C			Register Name: I2S/PCM_ISTA
Bit	**Read/Write**	**Default/Hex**	**Description**
6	R/W1C	0x0	TXU_INT TXFIFO Underrun Pending Interrupt 0: No pending interrupt 1: TXFIFO underrun pending interrupt Write '1' to clear this interrupt.
5	R/W1C	0x0	TXO_INT TXFIFO Overrun Pending Interrupt 0: No pending interrupt 1: TXFIFO overrun pending interrupt Write '1' to clear this interrupt.
4	R	0x1	TXE_INT TXFIFO Empty Pending Interrupt 0: No pending IRQ 1: TXFIFO empty pending interrupt when data in TXFIFO are less than TX trigger level
3	/	/	/
2	R/W1C	0x0	RXU_INT RXFIFO Underrun Pending Interrupt 0: No pending interrupt 1: RXFIFO underrun pending interrupt Write '1' to clear this interrupt.
1	R/W1C	0x0	RXO_INT RXFIFO Overrun Pending Interrupt 0: No pending IRQ 1: RXFIFO overrun pending IRQ Write '1' to clear this interrupt.
0	R/W	0x0	RXA_INT RXFIFO Data Available Pending Interrupt 0: No pending IRQ 1: Data available pending IRQ when data in RXFIFO are more than RX trigger level

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.1.6.5 0x0010 I2S/PCM RXFIFO Register (Default Value: 0x0000_0000)

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

5.1.6.6 0x0014 I2S/PCM FIFO Control Register (Default Value: 0x0004_00F0)

Offset: 0x0014			Register Name: I2S/PCM_FCTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	HUB_EN Audio Hub Enable The bit takes effect only when TXEN is set to 1. CPUX Domain: I2S0/I2S1/OWA TXFIFO Hub Enable. CPUS Domain: R_I2S0/Audiocodec TXFIFO Hub Enable. 0: Disabled 1: Enabled
30:26	/	/	/
25	R/WAC	0x0	FTX Write '1' to flush TXFIFO, self clear to '0'.
24	R/WAC	0x0	FRX Write '1' to flush RXFIFO, self clear to '0'.
23:19	/	/	/
18:12	R/W	0x40	TXTL TXFIFO Empty Trigger Level Interrupt and DMA request trigger level for TXFIFO normal condition. Trigger Level = TXTL
11:10	/	/	/
9:4	R/W	0xF	RXTL RXFIFO Trigger Level Interrupt and DMA request trigger level for RXFIFO normal condition. Trigger Level = RXTL + 1

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0014			Register Name: I2S/PCM_FCTL
Bit	Read/Write	Default/Hex	Description
3	/	/	/
2	R/W	0x0	TXIM TXFIFO Input Mode (Mode 0, 1) 0: Valid data at the MSB of TXFIFO register 1: Valid data at the LSB of TXFIFO register Example for 20-bit transmitted audio sample: Mode 0: TXFIFO[31:0] = {APB_WDATA[31:12], 12'h0} Mode 1: TXFIFO[31:0] = {APB_WDATA[19:0], 12'h0}
1:0	R/W	0x0	RXOM RXFIFO Output Mode (Mode 0, 1, 2, 3) 00: Expanding '0' at LSB of RXFIFO register 01: Expanding received sample sign bit at MSB of RXFIFO register 10: Truncating received samples at high half-word of RXFIFO register and low half-word of RXFIFO register is filled by '0' 11: Truncating received samples at low half-word of RXFIFO register and high half-word of RXFIFO register is expanded by its sign bit Example for 20-bit received audio sample: Mode 0: APB_RDATA[31:0] = {RXFIFO[31:12], 12'h0} Mode 1: APB_RDATA[31:0] = {12{RXFIFO[31]}, RXFIFO[31:12]} Mode 2: APB_RDATA [31:0] = {RXFIFO[31:16], 16'h0} Mode 3: APB_RDATA[31:0] = {16{RXFIFO[31], RXFIFO[31:16]}

5.1.6.7 0x0018 I2S/PCM FIFO Status Register (Default Value: 0x1080_0080)

Offset: 0x0018			Register Name: I2S/PCM_FSTA
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28	R	0x1	TXE TXFIFO Empty 0: No room for new sample in TXFIFO 1: More than one room for new sample in TXFIFO (>= 1 Word)
27:24	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0018			Register Name: I2S/PCM_FSTA
Bit	**Read/Write**	**Default/Hex**	**Description**
23:16	R	0x80	TXE_CNT TXFIFO Empty Space Word Counter
15:9	/	/	/
8	R	0x0	RXA RXFIFO Available 0: No available data in RXFIFO 1: More than one sample in RXFIFO (>= 1 Word)
7	R	0x1	Reserved
6:0	R	0x0	RXA_CNT RXFIFO available sample word counter

5.1.6.8 0x001C I2S/PCM DMA & Interrupt Control Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

5.1.6.9 0x0020 I2S/PCM TXFIFO Register (Default Value: 0x0000_0000)

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

5.1.6.10 0x0024 I2S/PCM Clock Divide Register (Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: I2S/PCM_CLKD
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0024			Register Name: I2S/PCM_CLKD
Bit	**Read/Write**	**Default/Hex**	**Description**
8	R/W	0x0	MCLKO_EN 0: Disable MCLK Output 1: Enable MCLK Output **Note: Whether in slave or master mode, when this bit is set to '1', MCLK should be output.**
7:4	R/W	0x0	BCLKDIV BCLK Divide Ratio from PLL_AUDIO 0000: Reserved 0001: Divide by 1 0010: Divide by 2 0011: Divide by 4 0100: Divide by 6 0101: Divide by 8 0110: Divide by 12 0111: Divide by 16 1000: Divide by 24 1001: Divide by 32 1010: Divide by 48 1011: Divide by 64 1100: Divide by 96 1101: Divide by 128 1110: Divide by 176 1111: Divide by 192

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

5.1.6.11 0x0028 I2S/PCM TX Counter Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.1.6.12 0x002C I2S/PCM RX Counter Register (Default Value: 0x0000_0000)

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

5.1.6.13 0x0030 I2S/PCM Channel Configuration Register (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: I2S/PCM_CHCFG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:10	/	/	/
9	R/W	0x0	TX_SLOT_HIZ 0: Normal mode for the last half-cycle of BCLK in the slot 1: Turn to Hi-Z state for the last half-cycle of BCLK in the slot
8	R/W	0x0	TX_STATE 0: Transfer level 0 in non-transferring slot 1: Turn to Hi-Z State (TDM) in non-transferring slot
7:4	R/W	0x0	RX_SLOT_NUM RX Channel/Slot number between CPU/DMA and RXFIFO 0000: 1 channel or slot ... 0111: 8 channels or slots 1000: 9 channels or slots ... 1111: 16 channels or slots

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

5.1.6.14 0x0034 I2S/PCM TX0 Channel Select Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.1.6.15 0x0038 I2S/PCM TX1 Channel Select Register (Default Value: 0x0000_0000)

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

5.1.6.16 0x003C I2S/PCM TX2 Channel Select Register (Default Value: 0x0000_0000)

Offset: 0x003C			Register Name: I2S/PCM_TX2CHSEL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:22	/	/	/
21:20	R/W	0x0	TX2_OFFSET TX2 Offset Tune (TX2 Data offset to LRCK) 0: No offset n: Data is offset by n BCLKs to LRCK

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x003C			Register Name: I2S/PCM_TX2CHSEL
Bit	**Read/Write**	**Default/Hex**	**Description**
19:16	R/W	0x0	TX2_CHSEL TX2 Channel (Slot) Number Select for Each Output 0000: 1 channel or slot ... 0111: 8 channels or slots 1000: 9 channels or slots ... 1111:16 channels or slots
15:0	R/W	0x0	TX2_CHEN TX2 Channel (Slot) Enable The bit[15:0] refer to Slot [15:0]. When one or more slots are disabled, the affected slots are set to the disable state. 0: Disabled 1: Enabled

5.1.6.17 0x0040 I2S/PCM TX3 Channel Select Register (Default Value: 0x0000_0000)

Offset: 0x0040			Register Name: I2S/PCM_TX3CHSEL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:22	/	/	/
21:20	R/W	0x0	TX3_OFFSET TX3 Offset Tune (TX3 Data offset to LRCK) 0: No offset n: Data is offset by n BCLKs to LRCK
19:16	R/W	0x0	TX3_CHSEL TX3 Channel (Slot) Number Select for Each Output 0000: 1 channel or slot ... 0111: 8 channels or slots 1000: 9 channels or slots ... 1111:16 channels or slots

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0040			Register Name: I2S/PCM_TX3CHSEL
Bit	Read/Write	Default/Hex	Description
15:0	R/W	0x0	TX3_CHEN TX3 Channel (Slot) Enable The bit[15:0] refers to Slot [15:0]. When one or more slots are disabled, the affected slots are set to the disable state. 0: Disabled 1: Enabled

5.1.6.18 0x0044 I2S/PCM TX0 Channel Mapping0 Register 0 (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0044			Register Name: I2S/PCM_TX0CHMAP0
Bit	**Read/Write**	**Default/Hex**	**Description**
19:16	R/W	0x0	TX0_CH12_MAP TX0 Channel 12 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
15:12	R/W	0x0	TX0_CH11_MAP TX0 Channel 11 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
11:8	R/W	0x0	TX0_CH10_MAP TX0 Channel 10 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
7:4	R/W	0x0	TX0_CH9_MAP TX0 Channel 9 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0044			Register Name: I2S/PCM_TX0CHMAP0
Bit	Read/Write	Default/Hex	Description
3:0	R/W	0x0	TX0_CH8_MAP TX0 Channel 8 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample

5.1.6.19 0x0048 I2S/PCM TX0 Channel Mapping1 Register 1 (Default Value: 0x0000_0000)

Offset: 0x0048			Register Name: I2S/PCM_TX0CHMAP1
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0x0	TX0_CH7_MAP TX0 Channel 7 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample
27:24	R/W	0x0	TX0_CH6_MAP TX0 Channel 6 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0048			Register Name: I2S/PCM_TX0CHMAP1
Bit	**Read/Write**	**Default/Hex**	**Description**
7:4	R/W	0x0	TX0_CH1_MAP TX0 Channel 1 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample
3:0	R/W	0x0	TX0_CH0_MAP TX0 Channel 0 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample

5.1.6.20 0x004C I2S/PCM TX1 Channel Mapping0 Register 0 (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x004C			Register Name: I2S/PCM_TX1CHMAP0
Bit	**Read/Write**	**Default/Hex**	**Description**
27:24	R/W	0x0	TX1_CH14_MAP TX1 Channel 14 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
23:20	R/W	0x0	TX1_CH13_MAP TX1 Channel 13 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
19:16	R/W	0x0	TX1_CH12_MAP TX1 Channel 12 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
15:12	R/W	0x0	TX1_CH11_MAP TX1 Channel 11 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x004C			Register Name: I2S/PCM_TX1CHMAP0
Bit	**Read/Write**	**Default/Hex**	**Description**
11:8	R/W	0x0	TX1_CH10_MAP TX1 Channel 10 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample
7:4	R/W	0x0	TX1_CH9_MAP TX1 Channel 9 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample
3:0	R/W	0x0	TX1_CH8_MAP TX1 Channel 8 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.1.6.21 0x0050 I2S/PCM TX1 Channel Mapping1 Register 1 (Default Value: 0x0000_0000)

Offset: 0x0050			Register Name: I2S/PCM_TX1CHMAP1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x0	TX1_CH7_MAP TX1 Channel 7 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
27:24	R/W	0x0	TX1_CH6_MAP TX1 Channel 6 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
23:20	R/W	0x0	TX1_CH5_MAP TX1 Channel 5 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
19:16	R/W	0x0	TX1_CH4_MAP TX1 Channel 4 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0050			Register Name: I2S/PCM_TX1CHMAP1
Bit	**Read/Write**	**Default/Hex**	**Description**
15:12	R/W	0x0	TX1_CH3_MAP TX1 Channel 3 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
11:8	R/W	0x0	TX1_CH2_MAP TX1 Channel 2 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
7:4	R/W	0x0	TX1_CH1_MAP TX1 Channel 1 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
3:0	R/W	0x0	TX1_CH0_MAP TX1 Channel 0 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.1.6.22 0x0054 I2S/PCM TX2 Channel Mapping0 Register 0 (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0054			Register Name: I2S/PCM_TX2CHMAP0
Bit	**Read/Write**	**Default/Hex**	**Description**
15:12	R/W	0x0	TX2_CH11_MAP TX2 Channel 11 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
11:8	R/W	0x0	TX2_CH10_MAP TX2 Channel 10 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
7:4	R/W	0x0	TX2_CH9_MAP TX2 Channel 9 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
3:0	R/W	0x0	TX2_CH8_MAP TX2 Channel 8 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.1.6.23 0x0058 I2S/PCM TX2 Channel Mapping1 Register 1 (Default Value: 0x0000_0000)

Offset: 0x0058			Register Name: I2S/PCM_TX2CHMAP1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x0	TX2_CH7_MAP TX2 Channel 7 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
27:24	R/W	0x0	TX2_CH6_MAP TX2 Channel 6 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
23:20	R/W	0x0	TX2_CH5_MAP TX2 Channel 5 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
19:16	R/W	0x0	TX2_CH4_MAP TX2 Channel 4 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0058			Register Name: I2S/PCM_TX2CHMAP1
Bit	**Read/Write**	**Default/Hex**	**Description**
15:12	R/W	0x0	TX2_CH3_MAP TX2 Channel 3 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample
11:8	R/W	0x0	TX2_CH2_MAP TX2 Channel 2 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample
7:4	R/W	0x0	TX2_CH1_MAP TX2 Channel 1 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample
3:0	R/W	0x0	TX2_CH0_MAP TX2 Channel 0 Mapping 0000: The first sample ... 0111: The eighth sample 1000: The ninth sample ... 1111: The sixteenth sample

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.1.6.24 0x005C I2S/PCM TX3 Channel Mapping0 Register 0 (Default Value: 0x0000_0000)

Offset: 0x005C			Register Name: I2S/PCM_TX3CHMAP0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x0	TX3_CH15_MAP TX3 Channel 15 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
27:24	R/W	0x0	TX3_CH14_MAP TX3 Channel 14 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
23:20	R/W	0x0	TX3_CH13_MAP TX3 Channel 13 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
19:16	R/W	0x0	TX3_CH12_MAP TX3 Channel 12 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x005C			Register Name: I2S/PCM_TX3CHMAP0
Bit	**Read/Write**	**Default/Hex**	**Description**
15:12	R/W	0x0	TX3_CH11_MAP TX3 Channel 11 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
11:8	R/W	0x0	TX3_CH10_MAP TX3 Channel 10 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
7:4	R/W	0x0	TX3_CH9_MAP TX3 Channel 9 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
3:0	R/W	0x0	TX3_CH8_MAP TX3 Channel 8 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.1.6.25 0x0060 I2S/PCM TX3 Channel Mapping1 Register 1 (Default Value: 0x0000_0000)

Offset: 0x0060			Register Name: I2S/PCM_TX3CHMAP1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x0	TX3_CH7_MAP TX3 Channel 7 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
27:24	R/W	0x0	TX3_CH6_MAP TX3 Channel 6 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
23:20	R/W	0x0	TX3_CH5_MAP TX3 Channel 5 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
19:16	R/W	0x0	TX3_CH4_MAP TX3 Channel 4 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0060			Register Name: I2S/PCM_TX3CHMAP1
Bit	**Read/Write**	**Default/Hex**	**Description**
15:12	R/W	0x0	TX3_CH3_MAP TX3 Channel 3 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
11:8	R/W	0x0	TX3_CH2_MAP TX3 Channel 2 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
7:4	R/W	0x0	TX3_CH1_MAP TX3 Channel 1 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
3:0	R/W	0x0	TX3_CH0_MAP TX3 Channel 0 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.1.6.26 0x0064 I2S/PCM RX Channel Select Register (Default Value: 0x0000_0000)

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

5.1.6.27 0x0068 I2S/PCM RX Channel Mapping Register0 (Default Value: 0x0000_0000)

Offset: 0x0068			Register Name: I2S/PCM_RXCHMAP0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/
29:28	R/W	0x0	RX_CH15_SELECT RX Channel 15 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0068			Register Name: I2S/PCM_RXCHMAP0
Bit	Read/Write	Default/Hex	Description
27:24	R/W	0x0	RX_CH15_MAP RX Channel 15 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
23:22	/	/	/
21:20	R/W	0x0	RX_CH14_SELECT RX Channel 14 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3
19:16	R/W	0x0	RX_CH14_MAP RX Channel 14 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
15:14	/	/	/
13:12	R/W	0x0	RX_CH13_SELECT RX Channel 13 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0068			Register Name: I2S/PCM_RXCHMAP0
Bit	Read/Write	Default/Hex	Description
11:8	R/W	0x0	RX_CH13_MAP RX Channel 13 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
7:6	/	/	/
5:4	R/W	0x0	RX_CH12_SELECT RX Channel 12 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3
3:0	R/W	0x0	RX_CH12_MAP RX Channel 12 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample

5.1.6.28 0x006C I2S/PCM RX Channel Mapping Register1 (Default Value: 0x0000_0000)

Offset: 0x006C			Register Name: I2S/PCM_RXCHMAP1
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:28	R/W	0x0	RX_CH11_SELECT RX Channel 11 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x006C			Register Name: I2S/PCM_RXCHMAP1
Bit	Read/Write	Default/Hex	Description
27:24	R/W	0x0	RX_CH11_MAP RX Channel 11 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
23:22	/	/	/
21:20	R/W	0x0	RX_CH10_SELECT RX Channel 10 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3
19:16	R/W	0x0	RX_CH10_MAP RX Channel 10 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
15:14	/	/	/
13:12	R/W	0x0	RX_CH9_SELECT RX Channel 9 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

5.1.6.29 0x0070 I2S/PCM RX Channel Mapping Register2 (Default Value: 0x0000_0000)

Offset: 0x0070			Register Name: I2S/PCM_RXCHMAP2
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:28	R/W	0x0	RX_CH7_SELECT RX Channel 7 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

5.1.6.30 0x0074 I2S/PCM RX Channel Mapping Register3 (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0074			Register Name: I2S/PCM_RXCHMAP3
Bit	**Read/Write**	**Default/Hex**	**Description**
27:24	R/W	0x0	RX_CH3_MAP RX Channel 3 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
23:22	/	/	/
21:20	R/W	0x0	RX_CH2_SELECT RX Channel 2 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3
19:16	R/W	0x0	RX_CH2_MAP RX Channel 2 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
15:14	/	/	/
13:12	R/W	0x0	RX_CH1_SELECT RX Channel 1 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0074			Register Name: I2S/PCM_RXCHMAP3
Bit	Read/Write	Default/Hex	Description
11:8	R/W	0x0	RX_CH1_MAP RX Channel 1 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample
7:6	/	/	/
5:4	R/W	0x0	RX_CH0_SELECT RX Channel 0 Select 00: SDI0 01: SDI1 10: SDI2 11: SDI3
3:0	R/W	0x0	RX_CH0_MAP RX Channel 0 Mapping 0000: The first sample … 0111: The eighth sample 1000: The ninth sample … 1111: The sixteenth sample

5.1.6.31 0x0080 I2S/PCM ASRC MCLK Configuration Register (Default Value: 0x0000_0000)

Offset: 0x0080			Register Name: MCLKCFG
Bit	Read/Write	Default/Hex	Description
31:17	R	0x0	/
16	R/W	0x0	ASRC_MCLK_GATE ASRC Clock Gate Enable Control 0: Gated 1: Not gated
15:4	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0080			Register Name: MCLKCFG
Bit	Read/Write	Default/Hex	Description
3:0	R/W	0x0	ASRC_MCLK_FREQ_DIV_COE Frequency Division Coefficient 4'd0 = Res (no output), 4'd1 = 1x, 4'd2 = 1/2x, 4'd3 = 1/4x, 4'd4 = 1/6x, 4'd5 = 1/8x, 4'd6 = 1/12x, 4'd7 = 1/16x, 4'd8 = 1/24x, 4'd9 = 1/32x, 4'd10 = 1/48, 4'd11 = 1/64x, 4'd12 = 1/96x, 4'd13 = 1/128x, 4'd14 = 1/176x, 4'd15 = 1/192x, others = Res

5.1.6.32 0x0084 I2S/PCM ASRC OUT Sample Configuration Register (Default Value: 0x0000_0000)

Offset: 0x0084			Register Name: F_{SOUT}CFG
Bit	Read/Write	Default/Hex	Description
31:21	R	0x0	
20	R/W	0x0	Clock Gate 0: Gated 1: Not gated
19:16	R/W	0x0	Clock Select 01: ACLK 10: ACLKM 11: BCLK 00: I2S0_ASRC_CLK Others: Reserved
15:8	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0084			Register Name: F_{SOUT}CFG
Bit	**Read/Write**	**Default/Hex**	**Description**
7:4	R/W	0x0	The First Division Factor It has two levels of frequency division, the first level is bit[7:4], the second level is bit[3:0], and the frequency division factors are multiplied by the two division factors, the division relationship of the two divisions are the same. 4'd0 = Res (no output), 4'd1 = 1x, 4'd2 = 1/2x, 4'd3 = 1/4x, 4'd4 = 1/6x, 4'd5 = 1/8x, 4'd6 = 1/12x, 4'd7 = 1/16x, 4'd8 = 1/24x, 4'd9 = 1/32x, 4'd10 = 1/48, 4'd11 = 1/64x, 4'd12 = 1/96x, 4'd13 = 1/128x, 4'd14 = 1/176x, 4'd15 = 1/192x

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0084			Register Name: F_{SOUT}CFG
Bit	Read/Write	Default/Hex	Description
3:0	R/W	0x0	The Second Division Factor 4'd0 = Res (no output), 4'd1 = 1x, 4'd2 = 1/2x, 4'd3 = 1/4x, 4'd4 = 1/6x, 4'd5 = 1/8x, 4'd6 = 1/12x, 4'd7 = 1/16x, 4'd8 = 1/24x, 4'd9 = 1/32x, 4'd10 = 1/48, 4'd11 = 1/64x, 4'd12 = 1/96x, 4'd13 = 1/128x, 4'd14 = 1/176x, 4'd15 = 1/192x

5.1.6.33 0x0088 I2S/PCM IN Sample Pulse Extend Configuration Register (Default Value: 0x0000_0000)

Offset: 0x0088			Register Name: FsinEXTCFG
Bit	Read/Write	Default/Hex	Description
31:17	R	0x0	/
16	R/W	0x0	Extend Enable 1: Enabled 0: Disabled Enable the bit when using ASRC.
15:0	R/W	0x0	The Cycle Number of Pulse Extend The cycle is BCLK clock and is 1 at least.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.1.6.34 0x008C I2S/PCM ASRC Enable Register (Default Value: 0x0000_0000)

Offset: 0x008C			Register Name: ASRCEN
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	R	0x0	/
0	R/W	0x0	ASRC Function Enable 1: Enabled 0: Disabled

5.1.6.35 0x0090 I2S/PCM ASRC Manual Configuration Register (Default Value: 0x0000_0000)

Offset: 0x0090			Register Name: ASRCMANCFG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	ASRC_RATIO_MANUAL_EN Manual Configuration of ASRC Ratio Enable 0: Disabled 1: Enabled
30:26	R	0x0	/
25:0	R/W	0x0	ASRC Ratio Value The ratio value is an unsigned 26-bit number and uses 4.22 data format, which means there are 4 bits to the left of the decimal point and 22 bits to the right of the decimal point.

5.1.6.36 0x0094 I2S/PCM ASRC Ratio State Register (Default Value: 0x0040_0000)

Offset: 0x0094			Register Name: ASRCRATIOSTAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	R	0x0	/
29	R	0x0	ASRC_BUF_OVERFLOW_STA ASRC Receive Data Buffer Overflow State It can control the mute with lock. 0: No overflow 1: Overflow

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0094			Register Name: ASRCRATIOSTAT
Bit	Read/Write	Default/Hex	Description
28	R	0x0	ADAPT_COMPUT_LOCK Adaptive Ratio Computational Lock 0: Unlocked 1: Locked
27:26	R	0x0	/
25:0	R	0x400000	ADAPT_COMPUT_VALUE Adaptive Ratio Computational Value

5.1.6.37 0x0098 I2S/PCM ASRC FIFO State Register (Default Value: 0x0000_0000)

Offset: 0x0098			Register Name: ASRCFIFOSTAT
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8:0	R	0x0	ASRC_RX_FIFO_FULL_LEVAL ASRC RXFIFO Full Level The manually-configured FIFO fill level for the ratio value of the received data.

5.1.6.38 0x009C I2S/PCM MBIST Test Configuration Register (Default Value: 0x0000_0000)

Offset: 0x009C			Register Name: ASRCMBISTCFG
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8	R/W	0x0	Enable the RAM BIST
7:1	/	/	/
0	R/W	0x0	Enable the ROM BIST

5.1.6.39 0x00A0 I2S/PCM ASRC MBIST Test State Register (Default Value: 0x0000_0002)

Offset: 0x00A0			Register Name: ASRCMBISTSTA
Bit	Read/Write	Default/Hex	Description
31:19	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x00A0			Register Name: ASRCMBISTSTA
Bit	**Read/Write**	**Default/Hex**	**Description**
18	R	0x0	ROM_BIST_ERROR_XOR ROM BIST errors xor
17	R	0x0	ROM_BIST_ERROR_SUM ROM BIST errors sum
16	R	0x0	ROM_BUSY_STATUS ROM busy status 1: ROM busy 0: ROM idle
15:8	/	/	/
7	R	0x0	RAM_BIST_ERR_STATUS RAM BIST errors status 1: Error 0: No effect
6:4	R	0x0	RAM_BIST_ERROR_PATTERN RAM BIST error pattern
3:2	R	0x0	RAM_BIST_ERROR_CYCLE RAM BIST error cycle
1	R	0x1	RAM_STOP_STATUS RAM stop status 1: Stopped 0: Running
0	R	0x0	RAM_BUSY_STATUS RAM busy status 1: RAM busy 0: RAM idle

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.2 DMIC

5.2.1 Overview

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

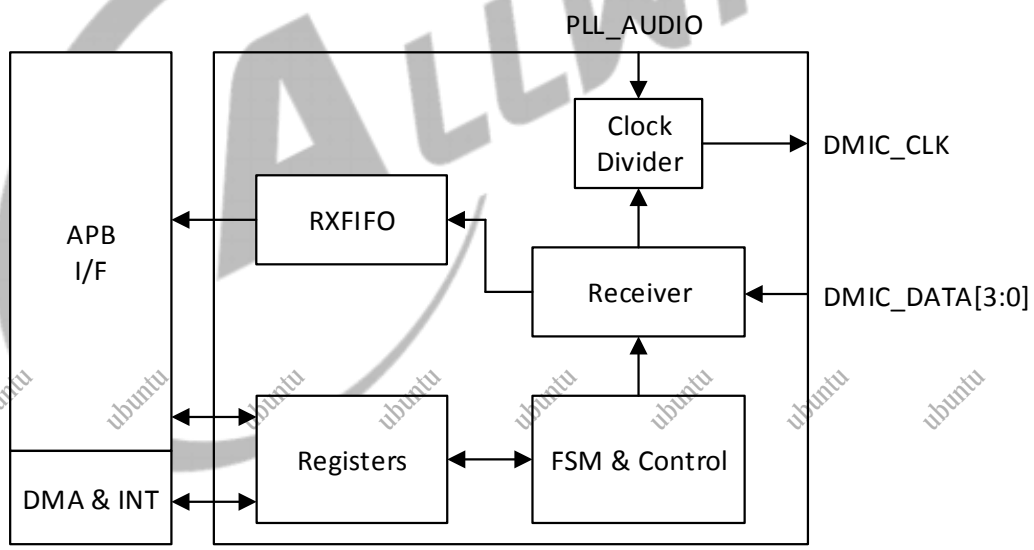
The DMIC controller includes the following features:

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

5.2.2 Block Diagram

The following figure shows a block diagram of the DMIC.

Figure 5-12 DMIC Block Diagram



5.2.3 Functional Descriptions

5.2.3.1 External Signals

The following table describes the external signals of DMIC.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Table 5-5 DMIC External Signals

Signal	Description	Type
S-DMIC-CLK	Digital Microphone Clock Output	O
S-DMIC-DATA0	Digital Microphone Data Input	I
S-DMIC-DATA1	Digital Microphone Data Input	I
S-DMIC-DATA2	Digital Microphone Data Input	I
S-DMIC-DATA3	Digital Microphone Data Input	I

5.2.3.2 Clock Sources

The following table describes the clock source for DMIC. For clock setting, configurations, and gating information, refer to section 3.3 "CCU".

Table 5-6 DMIC Clock Sources

Clock Sources	Description
PLL_AUDIO1(1X)	By default, PLL_AUDIO1(1X) is 24.5714 MHz, and PLL_AUDIO1(4X) is 98.2856 MHz.
PLL_AUDIO1(4X)	
PLL_AUDIO0(DIV2)	By default, PLL_AUDIO0 is 3072 MHz, PLL_AUDIO0(DIV2) is 1536 MHz, and PLL_AUDIO0(DIV5) is 614.4 MHz (24.576 MHz*25).
PLL_AUDIO0(DIV5)	

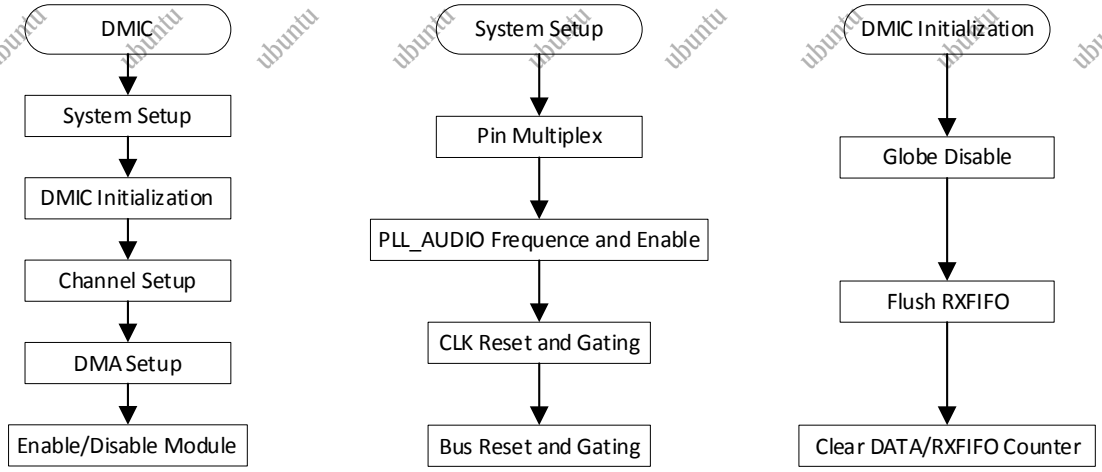
5.2.3.3 Operation Mode

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

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 5-13 DMIC Operation Mode



1. System Setup and DMIC Initialization

The first step in the system setup is properly programming the GPIO because the DMIC port is a multiplex pin. For functions of the multiplex pins, refer to the pin multiplex specification.

Perform the following steps for the clock source. Firstly, disable the PLL_AUDIO through PLL_AUDIOx Control Register[PLL_ENABLE]. Secondly, set up the frequency of the PLL_AUDIO in PLL_AUDIOx Control Register. Then enable PLL_AUDIO. After that, enable the DMIC gating through **DMIC_CLK_REG** when you checkout that the LOCK bit of PLL_AUDIOx Control Register becomes 1. At last, reset and enable the DMIC bus gating by **DMIC_BGR_REG**.

After the system setup, the register of DMIC can be setup. Firstly, initialize the DMIC. You should close the globe enable bit (**DMIC_EN**[8]), data channel enable bit (**DMIC_EN**[7:0]) by writing 0 to it. After that, flush the RXFIFO by writing 1 to **DMIC_RXFIFO_CTR**[31]. At last, you can clear the Data/RXFIFO counter by writing 1 to **DMIC_RXFIFO_STA**, **DMIC_CNT**.

2. Channel Setup and DMA Setup

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

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

3. Enable and Disable DMIC

To enable the function, you can enable the data channel enable bit (**DMIC_EN**[7:0]) by writing 1 to it. After that, enable DMIC by writing 1 to the Globe Enable bit (**DMIC_EN**[8]). Write 0 to **DMIC_EN**[8] to disable DMIC.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.2.4 Register List

Module Name	Base Address
DMIC	0x07031000

Register Name	Offset	Description
DMIC_EN	0x0000	DMIC Enable Control Register
DMIC_SR	0x0004	DMIC Sample Rate Register
DMIC_CTR	0x0008	DMIC Control Register
DMIC_DATA	0x0010	DMIC Data Register
DMIC_INTC	0x0014	MIC Interrupt Control Register
DMIC_INTS	0x0018	DMIC Interrupt Status Register
DMIC_RXFIFO_CTR	0x001C	DMIC RXFIFO Control Register
DMIC_RXFIFO_STA	0x0020	DMIC RXFIFO Status Register
DMIC_CH_NUM	0x0024	DMIC Channel Numbers Register
DMIC_CH_MAP	0x0028	DMIC Channel Mapping Register
DMIC_CNT	0x002C	DMIC Counter Register
DATA0_DATA1_VOL_CTR	0x0030	Data0 and Data1 Volume Control Register
DATA2_DATA3_VOL_CTR	0x0034	Data2 And Data3 Volume Control Register
HPF_EN_CTR	0x0038	High Pass Filter Enable Control Register
HPF_COEF_REG	0x003C	High Pass Filter Coefficient Register
HPF_GAIN_REG	0x0040	High Pass Filter Gain Register

5.2.5 Register Description

5.2.5.1 0x0000 DMIC Enable Control Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: DMIC_EN
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	Reserved
30	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: DMIC_EN
Bit	**Read/Write**	**Default/Hex**	**Description**
29	R/W	0x0	RX_SYNC_EN_START The bit takes effect only when RX_SYNC_EN is set to 1. CPUX Domain: I2S0/I2S1/OWA RX Synchronize Enable Start. CPUS Domain: R_I2S0/Audiocodec/DMIC RX Synchronize Enable Start. 0: Disabled 1: Enabled
28	R/W	0x0	RX_SYNC_EN DMIC RX Synchronize Enable 0: Disabled 1: Enabled
27:9	/	/	/
8	R/W	0x0	GLOBE_EN DMIC Globe Enable 0: Disabled 1: Enabled
7	R/W	0x0	DATA3_CHR_EN DATA3 Right Channel Enable 0: Disabled 1: Enabled
6	R/W	0x0	DATA3_CHL_EN DATA3 Left Channel Enable 0: Disabled 1: Enabled
5	R/W	0x0	DATA2_CHR_EN DATA2 Right Channel Enable 0: Disabled 1: Enabled
4	R/W	0x0	DATA2_CHL_EN DATA2 Left Channel Enable 0: Disabled 1: Enabled
3	R/W	0x0	DATA1_CHR_EN DATA1 Right Channel Enable 0: Disabled 1: Enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

5.2.5.2 0x0004 DMIC Sample Rate Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.2.5.3 0x0008 DMIC Control Register (Default Value: 0x0000_0000)

Offset: 0x0008			Register Name: DMIC_CTR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:9	R/W	0x0	DMICFDT DMIC RXFIFO Delay Time for Writing Data after GLOBE_EN 00: 5ms 01: 10ms 10: 20ms 11: 30ms
8	R/W	0x0	DMICDFEN DMIC RXFIFO Delay Function for Writing Data after GLOBE_EN 0: Disabled 1: Enabled
7	R/W	0x0	DATA3 Left Data and Right Data Swap Enable 0: Disabled 1: Enabled
6	R/W	0x0	DATA2 Left Data and Right Data Swap Enable 0: Disabled 1: Enabled
5	R/W	0x0	DATA1 Left Data and Right Data Swap Enable 0: Disabled 1: Enabled
4	R/W	0x0	DATA0 Left Data and Right Data Swap Enable 0: Disabled 1: Enabled
3:1	/	/	/
0	R/W	0x0	DMIC Oversample Rate 0: 128 (Supports 8 kHz to 24 kHz) 1: 64 (Supports 16 kHz to 48 kHz)

5.2.5.4 0x0010 DMIC DATA Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.2.5.5 0x0014 DMIC Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0014			Register Name: DMIC_INTC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2	R/W	0x0	RXFIFO_DRQ_EN DMIC RXFIFO Data Available DRQ Enable 0: Disabled 1: Enabled
1	R/W	0x0	RXFIFO_OVERRUN_IRQ_EN DMIC RXFIFO Overrun IRQ Enable 0: Disabled 1: Enabled
0	R/W	0x0	DATA_IRQ_EN DMIC RXFIFO Data Available IRQ Enable 0: Disabled 1: Enabled

5.2.5.6 0x0018 DMIC Interrupt Status Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.2.5.7 0x001C DMIC RXFIFO Control Register (Default Value: 0x0000_0040)

Offset: 0x001C			Register Name: DMIC_RXFIFO_CTR
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W1C	0x0	DMIC_RXFIFO_FLUSH DMIC RXFIFO Flush Writing '1' to flush RXFIFO, self clear to '0'
30:10	/	/	/
9	R/W	0x0	RXFIFO_MODE RXFIFO Output Mode (Mode 0, 1) 0: Expanding '0' at LSB of RXFIFO register 1: Expanding received sample sign bit at MSB of RXFIFO register For 24-bit received audio sample: Mode 0: RXDATA[31:0] = {RXFIFO_O[20:0], 11'h0} Mode 1: RXDATA[31:0] = {8{RXFIFO_O[20]}, RXFIFO_O[20:0], 3'h0} For 16-bit received audio sample: Mode 0: RXDATA[31:0] = {RXFIFO_O[20:5], 16'h0} Mode 1: RXDATA[31:0] = {16{RXFIFO_O[20]}, RXFIFO_O[20:5]}
8	R/W	0x0	Sample_Resolution 0: 16-bit 1: 24- bit
7:0	R/W	0x40	RXFIFO_TRG_LEVEL RXFIFO Trigger Level (TRLV[7:0]) Interrupt and DMA request trigger level for DMIC RXFIFO normal condition IRQ/DRQ Generated when WLEVEL > TRLV[7:0]) WLEVEL represents the number of valid samples in the DMIC RXFIFO

5.2.5.8 0x0020 DMIC RXFIFO Status Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: DMIC_RXFIFO_STA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:9	/	/	/
8	R	0x1	Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0020			Register Name: DMIC_RXFIFO_STA
Bit	Read/Write	Default/Hex	Description
7:0	R/W	0x0	DMIC_DATA_CNT DMIC RXFIFO Available Sample Word Counter

5.2.5.9 0x0024 DMIC Channel Numbers Register (Default Value: 0x0000_0001)

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

5.2.5.10 0x0028 DMIC Channel Mapping Register (Default Value: 0x7654_3210)

Offset: 0x0028			Register Name: DMIC_CH_MAP
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0x7	DMIC_CH7_MAP DMIC Channel 7 Mapping 0000: DATA0 Left Channel 0001: DATA0 Right Channel 0010: DATA1 Left Channel 0011: DATA1 Right Channel 0100: DATA2 Left Channel 0101: DATA2 Right Channel 0110: DATA3 Left Channel 0111: DATA3 Right Channel

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0028			Register Name: DMIC_CH_MAP
Bit	**Read/Write**	**Default/Hex**	**Description**
3:0	R/W	0x0	DMIC_CH0_MAP DMIC Channel0 Mapping 0000: DATA0 Left Channel 0001: DATA0 Right Channel 0010: DATA1 Left Channel 0011: DATA1 Right Channel 0100: DATA2 Left Channel 0101: DATA2 Right Channel 0110: DATA3 Left Channel 0111: DATA3 Right Channel

5.2.5.11 0x002C DMIC Counter Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.2.5.12 0x0030 DATA0 and DATA1 Volume Control Register (Default Value: 0xA0A0_A0A0)

Offset: 0x0030			Register Name: DATA0_DATA1_VOL_CTR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	R/W	0xA0	DATA1L_VOL Data1 Left Channel Volume Control (-119.25 dB to 71.25 dB, 0.75 dB/Step) 0x00: Mute 0x01: -119.25 dB ... 0x9F: -0.75 dB 0xA0: 0 dB 0xA1: 0.75 dB ... 0xFF: 71.25 dB
23:16	R/W	0xA0	DATA1R_VOL Data1 Right Channel Volume Control (-119.25 dB to 71.25 dB, 0.75 dB/Step) 0x00: Mute 0x01: -119.25 dB ... 0x9F: -0.75 dB 0xA0: 0 dB 0xA1: 0.75 dB ... 0xFF: 71.25 dB
15:8	R/W	0xA0	DATA0L_VOL Data0 Left Channel Volume Control (-119.25 dB to 71.25 dB, 0.75 dB/Step) 0x00: Mute 0x01: -119.25 dB ... 0x9F: -0.75 dB 0xA0: 0 dB 0xA1: 0.75 dB ... 0xFF: 71.25 dB

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

5.2.5.13 0x0034 DATA2 and DATA3 Volume Control Register (Default Value: 0xA0A0_A0A0)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.2.5.14 0x0038 High Pass Filter Enable Control Register (Default Value: 0x0000_0000)

Offset: 0x0038			Register Name: HPF_EN_CTR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R/W	0x0	HPF_DATA3_CHR_EN High Pass Filter DATA3 Right Channel Enable 0: Disabled 1: Enabled
6	R/W	0x0	HPF_DATA3_CHL_EN High Pass Filter DATA3 Left Channel Enable 0: Disabled 1: Enabled
5	R/W	0x0	HPF_DATA2_CHR_EN High Pass Filter DATA2 Right Channel Enable 0: Disabled 1: Enabled
4	R/W	0x0	HPF_DATA2_CHL_EN High Pass Filter DATA2 Left Channel Enable 0: Disabled 1: Enabled
3	R/W	0x0	HPF_DATA1_CHR_EN High Pass Filter DATA1 Right Channel Enable 0: Disabled 1: Enabled
2	R/W	0x0	HPF_DATA1_CHL_EN High Pass Filter DATA1 Left Channel Enable 0: Disabled 1: Enabled
1	R/W	0x0	HPF_DATA0_CHR_EN High Pass Filter DATA0 Right Channel Enable 0: Disabled 1: Enabled
0	R/W	0x0	HPF_DATA0_CHL_EN High Pass Filter DATA0 Left Channel Enable 0: Disabled 1: Enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.2.5.15 0x003C High Pass Filter Coefficient Register (Default Value: 0x00FF_AA45)

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

5.2.5.16 0x0040 High Pass Filter Gain Register (Default Value: 0x00FF_D522)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.3 OWA

5.3.1 Overview

The One Wire Audio (OWA) provides a serial bus interface for audio data. This interface is widely used for consumer audio.

The OWA includes the following features:

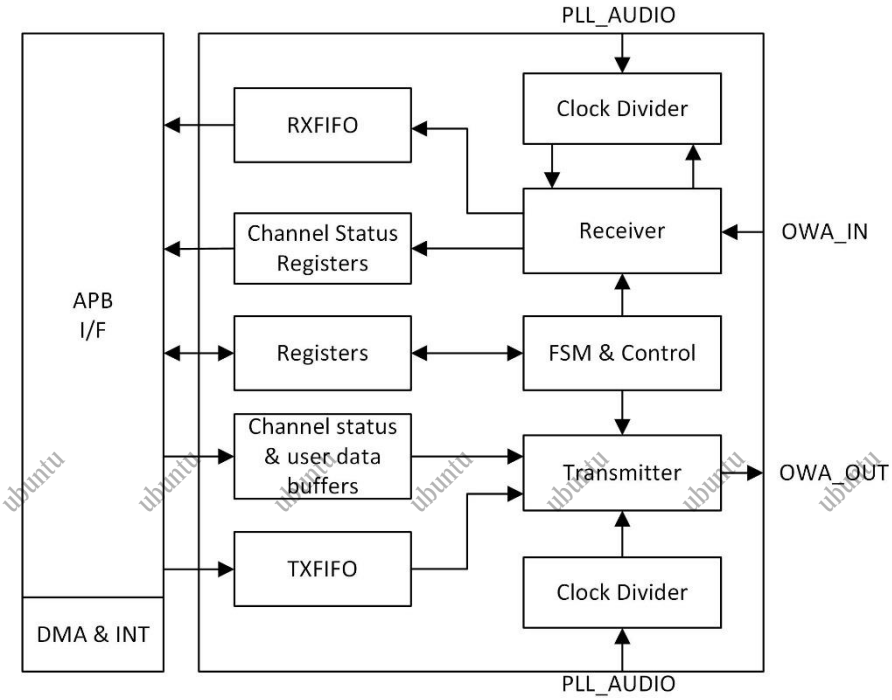
- Compliance with S/PDIF Interface
- IEC-60958 transmitter and receiver functionality
- Supports 16-bit, 20-bit and 24-bit data formats
- Supports channel status insertion for the transmitter
- Supports channel status capture for the receiver
- Supports hardware parity on TX/RX
  - Hardware parity generation on the transmitter
  - Hardware parity checking on the receiver
- Transmit and receive data FIFOs
  - One 128×24 bits TXFIFO and 64x24 bits RXFIFO for audio data transfer
  - Programmable FIFO thresholds
- Supports interrupts and DMA

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.3.2 Block Diagram

The following figure shows the OWA block diagram.

Figure 5-14 OWA Block Diagram



OWA contains the following sub-blocks:

Table 5-7 OWA Sub-blocks

Sub-block	Description
Registers	Analyzes the configuration parameter, DMA requests, and IRQ feedbacks
Receiver	Parses the frame header and receives the data
Transmitter	Sends the data
FSM	Finite state machine
Clock Divider	Clock divider circuit

5.3.3 Functional Descriptions

5.3.3.1 External Signals

OWA is a Biphase-Mark Encoding Digital Audio Transfer protocol. In this protocol, the CLK signal and data signal are transferred in the same line. The following figure describes the external signals of OWA. OWA-OUT is the output pin for output CLK and DATA, and OWA-IN is input pin for the input CLK and DATA.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Table 5-8 OWA External Signals

Signal Name	Description	Type
OWA-OUT	OWA output	O
OWA-IN	OWA input	I

5.3.3.2 Clock Sources

The following table describes the clock sources for OWA. For clock setting, configurations and gating information, refer to section 3.3 "CCU".

Table 5-9 OWA Clock Sources

Clock Name	Description
PLL_AUDIO1(1X)	By default, PLL_AUDIO1(1X) is 24.5714 MHz, and PLL_AUDIO1(4X) is 98.2856 MHz.
PLL_AUDIO1(4X)	
PLL_AUDIO0(DIV2)	By default, PLL_AUDIO0 is 3072 MHz, PLL_AUDIO0(DIV2) is 1536 MHz, and PLL_AUDIO0(DIV5) is 614.4 MHz (24.576 MHz*25).
PLL_AUDIO0(DIV5)	

5.3.3.3 Biphase-Mark Code (BMC)

In the OWA format, the digital signal is coded using the biphase-mark code (BMC). The clock, frame, and data are embedded in only one signal—the data pin. In the BMC system, each data bit is encoded into two logical states (00, 01, 10, or 11) at the pin. The following figure and table show how data is encoded to the BMC format.

The frequency of the clock is twice the data bit rate, as shown in the following figure. Also, the clock is always programmed to 128 x fs, where fs is the sample rate. The device receiving in the OWA format can recover the clock and frame information from the BMC signal.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 5-15 OWA Biphase-Mark Code

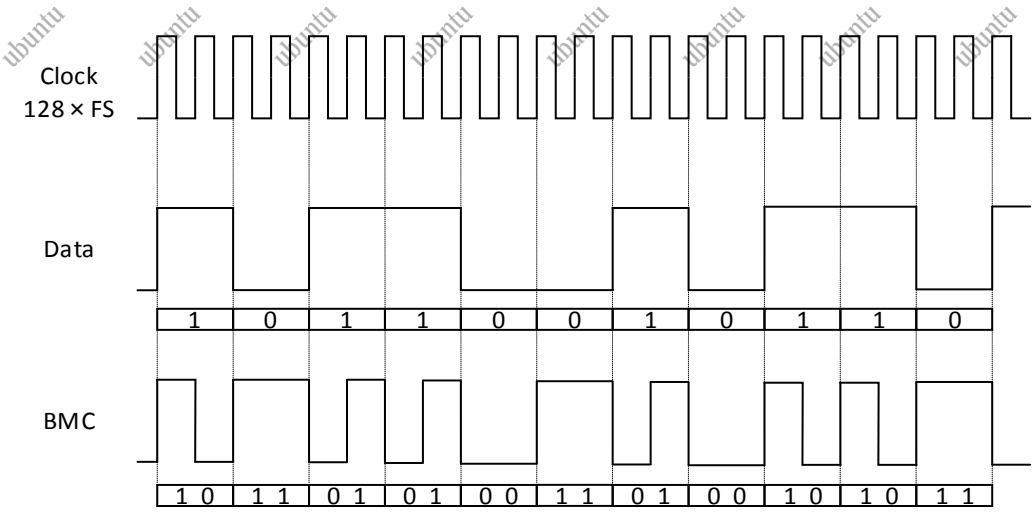


Table 5-10 Biphase-Mark Encoder

Data	Previous State	BMC
0	0	11
0	1	00
1	0	10
1	1	01

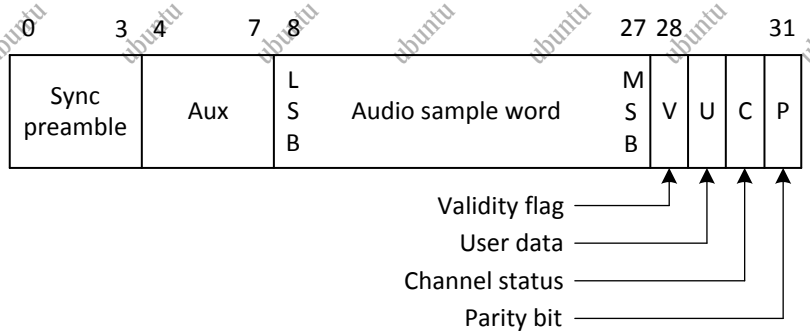
5.3.3.4 OWA Transmit Format

The OWA supports digital audio data transfer and receive. It also supports full-duplex synchronous work mode. The software can set the work mode by the OWA Control Register.

Every audio sample transmitted in a sub-frame consists of 32-bit, numbered from 0 to 31. The following figure shows a sub-frame.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 5-16 OWA Sub-Frame Format



Bits 0-3 carry one of the four permitted preambles to signify the type of audio sample in the current sub-frame. The preamble is not encoded in BMC format, and therefore the preamble code can contain more than two consecutive 0 or 1 logical states in a row.

Bits 4-27 carry the audio sample word in linear 2s-complement representation. The most-significant bit (MSB) is carried by bit 27. When a 24-bit coding range is used, the least-significant bit (LSB) is in bit 4. When a 20-bit coding range is used, bit[8:27] carry the audio sample word with the LSB in bit 8. Bit[4:7] may be used for other applications and are designated auxiliary sample bits.

If the source provides fewer bits than the interface allows (either 20 or 24), the unused LSBs are set to logical 0. For a nonlinear PCM audio application or a data application, the main data field may carry any other information.

Bit 28 carries the validity bit (V) associated with the main data field in the sub-frame.

Bit 29 carries the user data channel (U) associated with the main data field in the sub-frame.

Bit 30 carries the channel status information (C) associated with the main data field in the sub-frame. The channel status indicates if the data in the sub-frame is a digital audio or some other type of data.

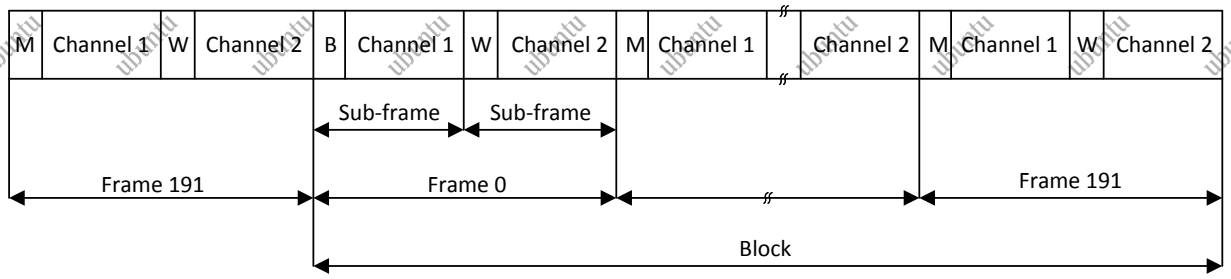
Bit 31 carries a parity bit (P) such that bit 4-31 carry an even number of 1s and an even number of 0s (even parity). As shown in the following table, the preambles (bit 0-3) are also defined with even parity.

Table 5-11 Preamble Codes

Preamble Code	Previous Logical State	Logical State	Description
B (or Z)	0	1110 1000	Start of a block and sub-frame 1
M (or X)	0	1110 0010	Sub-frame 1
W (or Y)	0	1110 0100	Sub-frame 2
B (or Z)	1	0001 0111	Start of a block and sub-frame 1
M (or X)	1	00011101	Sub-frame 1

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

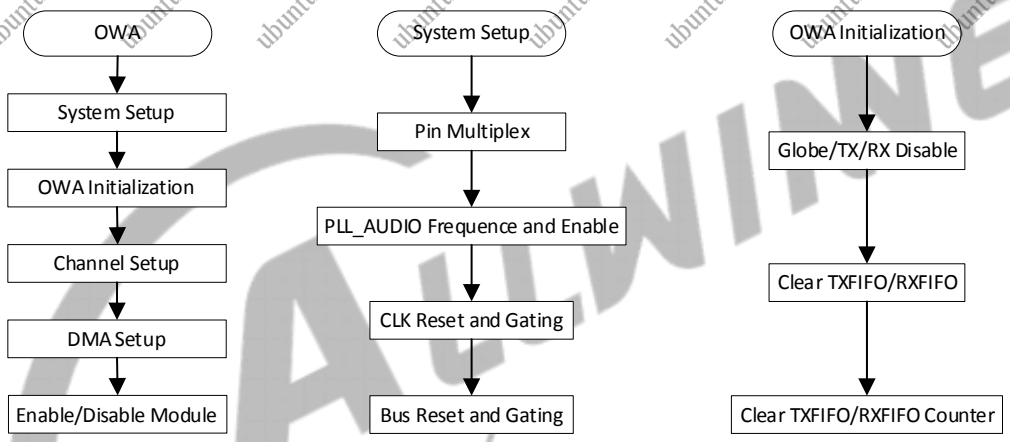
Figure 5-17 OWA Frame/Block Format



5.3.3.5 Operation Modes

The software operation of the OWA is divided into five steps: system setup, OWA initialization, channel setup, DMA setup and enable/disable module. The following sections describe these five steps.

Figure 5-18 OWA Operation Flow



1. System Setup and OWA Initialization

The first step in the OWA initialization is properly programming the GPIO because the OWA port is a multiplex pin. You can find the function in section 7.7 "GPIO".

The clock source for the OWA should be followed. Firstly, reset the audio PLL in PLL_AUDIOx Control Register. Secondly, set up the frequency of the Audio PLL in the PLL_AUDIOx Control Register. After that, enable the OWA gating. Lastly, enable the OWA bus gating.

After the system setup, the register of OWA can be set up. Firstly, reset the OWA by writing 1 to **OWA_CTL**[0] and clear the TX/RX FIFO by writing 1 to **OWA_FCTL**[17:16]. After that, enable the globe enable bit by writing 1 to **OWA_CTL**[1] and clear the interrupt and TX/RX counter by setting **OWA_ISTA** and **OWA_TX_CNT**/**OWA_RX_CNT**.

2. Channel Setup and DMA Setup

You can set up the audio type, clock divider ratio, the sample format, and the trigger level, and so on. The setup of the register can be found in the specification.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

The OWA supports two methods to transfer the data. The most common way is DMA, the configuration of DMA can be found in section 3.9 "DMA". In this module, you just enable the DRQ in **OWA_INT**[7].

3. Enable and Disable OWA

To enable the function, you can enable TX/RX by writing **OWA_TX_CFIG**[31]/**OWA_RX_CFIG**[0]. After that, enable OWA by writing 1 to **OWA_CTL**[1]. Writing 0 to **OWA_CTL**[1] to disable process.

5.3.4 Register List

Module Name	Base Address
OWA	0x02002C00

Register Name	Offset	Description
OWA_GEN_CTL	0x0000	OWA General Control Register
OWA_TX_CFIG	0x0004	OWA TX Configuration Register
OWA_RX_CFIG	0x0008	OWA RX Configuration Register
OWA_ISTA	0x000C	OWA Interrupt Status Register
OWA_RXFIFO	0x0010	OWA RXFIFO Register
OWA_FCTL	0x0014	OWA FIFO Control Register
OWA_FSTA	0x0018	OWA FIFO Status Register
OWA_INT	0x001C	OWA Interrupt Control Register
OWA_TX_FIFO	0x0020	OWA TX FIFO Register
OWA_TX_CNT	0x0024	OWA TX Counter Register
OWA_RX_CNT	0x0028	OWA RX Counter Register
OWA_TX_CHSTA0	0x002C	OWA TX Channel Status Register0
OWA_TX_CHSTA1	0x0030	OWA TX Channel Status Register1
OWA_RXCHSTA0	0x0034	OWA RX Channel Status Register0
OWA_RXCHSTA1	0x0038	OWA RX Channel Status Register1

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.3.5 Register Description

5.3.5.1 0x0000 OWA General Control Register (Default Value: 0x0000_0080)

Offset: 0x0000			Register Name: OWA_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R	0x1	Reserved
6:3	/	/	/
2	R/W	0x0	LOOP Loopback Test 0: Normal Mode 1: Loopback Test When the bit is set to '1', DOUT and DIN need be connected.
1	R/W	0x0	GEN Global Enable Disabling this bit overrides the operations of enabling and flushing all FIFOs by any other blocks or channels. 0: Disabled 1: Enabled
0	R/W	0x0	RST Reset 0: Normal 1: Reset Self clear to 0.

5.3.5.2 0x0004 OWA TX Configure Register (Default Value: 0x0000_00F0)

Offset: 0x0004			Register Name: OWA_TX_CFIG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	TX_SINGLE_MODE Tx Single Channel Mode 0: Disabled 1: Enabled
30:18	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0004			Register Name: OWA_TX_CFIG
Bit	Read/Write	Default/Hex	Description
17	R/W	0x0	ASS Audio Sample Select when TX FIFO Underrun 0: Sending 0 1: Sending the last audio **Note: This bit is only valid in PCM mode.**
16	R/W	0x0	TX_AUDIO TX Data Type 0: Linear PCM (Valid bit of both sub-frame set to 0) 1: Non-audio (Valid bit of both sub-frame set to 1)
15:9	/	/	/
8:4	R/W	0xF	TX_RATIO TX Clock Divide Ratio Clock divide ratio = TX_TATIO + 1 Fs = PLL_AUDIO/[(TX_TATIO + 1)*64*2]
3:2	R/W	0x0	TX_SF TX Sample Format 00: 16 bits 01: 20 bits 10: 24 bits 11: Reserved
1	R/W	0x0	TX_CHM CHSTMODE 0: Channel status A and B set to 0 1: Channel status A and B generated from TX_CHSTA
0	R/W	0x0	TXEN 0: Disabled 1: Enabled

5.3.5.3 0x0008 OWA RX Configure Register (Default Value: 0x0000_0000)

Offset: 0x0008			Register Name: OWA_RX_CFIG
Bit	Read/Write	Default/Hex	Description
31:5	/	/	/
4	R	0x0	RX_LOCK_FLAG 0: Unlocked 1: Locked

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0008			Register Name: OWA_RX_CFIG
Bit	Read/Write	Default/Hex	Description
3	R/W	0x0	RX_CHST_SRC 0: RX_CH_STA Register Holds Status from Channel A 1: RX_CH_STA Register Holds Status from Channel B
2	/	/	/
1	R/W	0x0	CHST_CP Channel Status Capture 0: Idle or Capture End 1: Capture Channel Status Start When set to '1', the system starts to capture the channel status. When finished, the bit will automatically turn to '0'.
0	R/W	0x0	RXEN 0: Disabled 1: Enabled

5.3.5.4 0x000C OWA Interrupt Status Register (Default Value: 0x0000_0010)

Offset: 0x000C			Register Name: OWA_ISTA
Bit	Read/Write	Default/Hex	Description
31:19	/	/	/
18	R/W1C	0x0	RX_LOCK_INT 0: No Pending IRQ 1: RX Lock Pending Interrupt (RX_LOCK_FLAG turns from 0 to 1) Write '1' to clear this interrupt.
17	R/W1C	0x0	RX_UNLOCK_INT RX Unlock Pending Interrupt 0: No Pending IRQ 1: RX Unlock Pending Interrupt (RX_LOCK_FLAG turns from 0 to 1) Write 1 to clear this interrupt.
16	R/W1C	0x0	RX_PARERRI_INT RX Parity Error Pending Interrupt 0: No Pending IRQ 1: RX Parity Error Pending Interrupt Write "1" to clear this interrupt.
15:7	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x000C			Register Name: OWA_ISTA
Bit	**Read/Write**	**Default/Hex**	**Description**
6	R/W1C	0x0	TXU_INT TX FIFO Underrun Pending Interrupt 0: No Pending IRQ 1: FIFO Underrun Pending Interrupt Writing “1” to clear this interrupt.
5	R/W1C	0x0	TXO_INT TX FIFO Overrun Pending Interrupt 0: No Pending IRQ 1: FIFO Overrun Pending Interrupt Writing “1” to clear this interrupt.
4	R/W1C	0x1	TXE_INT TX FIFO Empty Pending Interrupt 0: No Pending IRQ 1: FIFO Empty Pending Interrupt Writing “1” to clear this interrupt or automatically clear if the interrupt condition fails.
3:2	/	/	/
1	R/W1C	0x0	RXO_INT RXFIFO Overrun Pending Interrupt 0: RXFIFO Overrun Pending Write ‘1’ to clear this interrupt.
0	R/W1C	0x0	RXA_INT RXFIFO Available Pending Interrupt 0: No Pending IRQ 1: Data Available Pending IRQ Write ‘1’ to clear this interrupt or automatically clear if the interrupt condition fails.

5.3.5.5 0x0010 OWA RXFIFO Register (Default Value: 0x0000_0000)

Offset: 0x0010			Register Name: OWA_RXFIFO
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	RX_DATA The host can get one sample by reading this register, A channel data is first, and then the B channel data.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.3.5.6 0x0014 OWA FIFO Control Register (Default Value: 0x0004_0200)

Offset: 0x0014			Register Name: OWA_FCTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	HUB_EN Audio Hub Enable The bit takes effect only when the TXEN is set to 1. CPUX Domain: I2S0/I2S1/OWA TXFIFO Hub Enable. CPUS Domain: R_I2S0/Audiocodec TXFIFO Hub Enable. 0: Disabled 1: Enabled
30	R/W1C	0x0	FTX Write '1' to flush TXFIFO, self clear to '0'.
29	R/W1C	0x0	FRX Write '1' to flush RXFIFO, self clear to '0'.
28:22	/	/	/
21	R/W	0x0	RX_SYNC_EN_START The bit takes effect only when the RX_SYNC_EN is set to 1. CPUX Domain: I2S0/I2S1/OWA RX Synchronize Enable Start. CPUS Domain: R_I2S0/Audiocodec/DMIC RX Synchronize Enable Start. 0: Disabled 1: Enabled
20	R/W	0x0	RX_SYNC_EN OWA RX Synchronize Enable 0: Disabled 1: Enabled
19:12	R/W	0x40	TXTL TX FIFO Empty Trigger Level Interrupt and DMA request trigger level for TX FIFO normal condition. Trigger Level = TXTL
11	/	/	/
10:4	R/W	0x20	RXTL RX FIFO Trigger Level Interrupt and DMA request trigger level for RX FIFO normal condition. Trigger Level = RXTL + 1
3	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0014			Register Name: OWA_FCTL
Bit	Read/Write	Default/Hex	Description
2	R/W	0x0	TXIM TXFIFO Input Mode (Mode0, 1) 0: Valid data at the MSB of TXFIFO Register 1: Valid data at the LSB of TXFIFO Register Example for 20-bit transmitted audio sample: Mode 0: TXFIFO[23:0] = {APB_WDATA[31:12], 4'h0} Mode 1: TXFIFO[23:0] = {APB_WDATA[19:0], 4'h0}
1:0	R/W	0x0	RXOM RXFIFO Output Mode(Mode 0,1,2,3) 00: Expanding '0' at LSB of RXFIFO Register 01: Expanding received sample sign bit at MSB of RXFIFO Register 10: Truncating received samples at high half-word of RXFIFO Register and low half-word of RXFIFO Register is filled by '0' 11: Truncating received samples at low half-word of RXFIFO Register and high half-word of RXFIFO Register is expanded by its sigh bit Mode 0: APB_RDATA[31:0] = {RXFIFO[23:0], 8'h0} Mode 1: APB_RDATA[31:0] = {8'RXFIFO[23], RXFIFO[23:0]} Mode 2: APB_RDATA[31:0] = {RXFIFO[23:8], 16'h0} Mode 3: APB_RDATA[31:0] = {16'RXFIFO[23], RXFIFO[23:8]}

5.3.5.7 0x0018 OWA FIFO Status Register (Default Value: 0x8080_0000)

Offset: 0x0018			Register Name: OWA_FSTA
Bit	Read/Write	Default/Hex	Description
31	R	0x1	TXE TXFIFO Empty (indicate TXFIFO is not full) 0: No room for new sample in TXFIFO 1: More than one room for new sample in TXFIFO (>= 1 Word)
30:24	/	/	/
23:16	R	0x80	TXE_CNT TXFIFO Empty Space Word Counter
15	R	0x0	RXA RXFIFO Available 0: No available data in RXFIFO 1: More than one sample in RXFIFO (>= 1 Word)

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0018			Register Name: OWA_FSTA
Bit	Read/Write	Default/Hex	Description
14:7	/	/	/
6:0	R	0x0	RXA_CNT RXFIFO Available Sample Word Counter

5.3.5.8 0x001C OWA Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x001C			Register Name: OWA_INT
Bit	Read/Write	Default/Hex	Description
31:19	/	/	/
18	R/W	0x0	RX_LOCKI_EN RX LOCK Interrupt Enable 0: Disabled 1: Enabled
17	R/W	0x0	RX_UNLOCKI_EN RX UNLOCK Interrupt Enable 0: Disabled 1: Enabled
16	R/W	0x0	RX_PARERRI_EN RX PARITY ERORR Interrupt Enable 0: Disabled 1: Enabled
15:8	/	/	/
7	R/W	0x0	TX_DRQ TXFIFO Empty DRQ Enable 0: Disabled 1: Enabled
6	R/W	0x0	TXUI_EN TXFIFO Underrun Interrupt Enable 0: Disabled 1: Enabled
5	R/W	0x0	TXOI_EN TXFIFO Overrun Interrupt Enable 0: Disabled 1: Enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x001C			Register Name: OWA_INT
Bit	Read/Write	Default/Hex	Description
4	R/W	0x0	TXEI_EN TXFIFO Empty Interrupt Enable 0: Disabled 1: Enabled
3	/	/	/
2	R/W	0x0	RX_DRQ RXFIFO Data Available DRQ Enable When set to '1', RXFIFO DMA Request is asserted if data is available in RXFIFO. 0: Disabled 1: Enabled
1	R/W	0x0	RXOI_EN RXFIFO Overrun Interrupt Enable 0: Disabled 1: Enabled
0	R/W	0x0	RXAI_EN RXFIFO Data Available Interrupt Enable 0: Disabled 1: Enabled

5.3.5.9 0x0020 OWA TX FIFO Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: OWA_TXFIFO
Bit	Read/Write	Default/Hex	Description
31:0	W	0x0	TX_DATA Transmitting A, B channel data should be written this register one by one. A channel data is first, and then the B channel data.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.3.5.10 0x0024 OWA TX Counter Register (Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: OWA_TX_CNT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	TX_CNT TX Sample Counter The audio sample number of sending into TXFIFO. When one sample is put into TXFIFO by DMA or by host IO, the TX sample counter register increases by one. The TX sample counter register can be set to any initial value at any time. After updated by the initial value, the counter register should count on the base of this initial value.

5.3.5.11 0x0028 OWA RX Counter Register (Default Value: 0x0000_0000)

Offset: 0x0028			Register Name: OWA_RX_CNT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	RX_CNT RX Sample Counter The audio sample number of writing into RXFIFO. When one sample is written by Codec, the RX sample counter register increases by one. The RX Counter register can be set to any initial value at any time. After being updated by the initial value, the counter register should count on the base of this value.

5.3.5.12 0x002C OWA TX Channel Status Register0 (Default Value: 0x0000_0000)

Offset: 0x002C			Register Name: OWA_TX_CHSTA0
Bit	**Read/Write**	**Default/Hex**	**Description**
31: 30	/	/	/
29:28	R/W	0x0	CA Clock Accuracy 00: Level 2 01: Level 1 10: Level 3 11: Not matched

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x002C			Register Name: OWA_TX_CHSTA0
Bit	**Read/Write**	**Default/Hex**	**Description**
27:24	R/W	0x0	FREQ Sampling Frequency 0000: 44.1 kHz 0001: Not indicated 0010: 48 kHz 0011: 32 kHz 0100: 22.05 kHz 0101: Reserved 0110: 24 kHz 0111: Reserved 1000: Reserved 1001: 768 kHz 1010: 96 kHz 1011: Reserved 1100: 176.4 kHz 1101: Reserved 1110: 192 kHz 1111: Reserved
23:20	R/W	0x0	CN Channel Number
19:16	R/W	0x0	SN Source Number
15:8	R/W	0x0	CC Category Code Indicates the kind of equipment that generates the digital audio interface signal.
7:6	R/W	0x0	MODE Mode 00: Default Mode 01 to 11: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x002C			Register Name: OWA_TX_CHSTA0
Bit	**Read/Write**	**Default/Hex**	**Description**
5:3	R/W	0x0	EMP Emphasis Additional format information For bit 1 = “0”, Linear PCM audio mode: 000: 2 audio channels without pre-emphasis 001: 2 audio channels with 50 μs/15 μs pre-emphasis 010: Reserved (for 2 audio channels with pre-emphasis) 011: Reserved (for 2 audio channels with pre-emphasis) 100 to 111: Reserved For bit 1 = “1”, other than Linear PCM applications: 000: Default state 001 to 111: Reserved
2	R/W	0x0	CP Copyright 0: Copyright is asserted 1: No copyright is asserted
1	R/W	0x0	TYPE Audio Data Type 0: Linear PCM samples 1: Non-linear PCM audio
0	R/W	0x0	PRO Application Type 0: Consumer application 1: Professional application This bit must be fixed to “0”.

5.3.5.13 0x0030 OWA TX Channel Status Register1 (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: OWA_TX_CHSTA1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:10	/	/	/
9:8	R/W	0x0	CGMS_A 00: Copying is permitted without restriction 01: One generation of copies may be made 10: Condition not be used 11: No copying is permitted

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0030			Register Name: OWA_TX_CHSTA1
Bit	**Read/Write**	**Default/Hex**	**Description**
7:4	R/W	0x0	ORIG_FREQ Original Sampling Frequency 0000: Not indicated 0001: 192 kHz 0010: 12 kHz 0011: 176.4 kHz 0100: Reserved 0101: 96 kHz 0110: 8 kHz 0111: 88.2 kHz 1000: 16 kHz 1001: 24 kHz 1010: 11.025 kHz 1011: 22.05 kHz 1100: 32 kHz 1101: 48 kHz 1110: Reserved 1111: 44.1 kHz
3:1	R/W	0x0	WL Sample Word Length For bit 0 = “0”: 000: Not indicated 001: 16 bits 010: 18 bits 100: 19 bits 101: 20 bits 110: 17 bits 111: Reserved For bit 0 = “1”: 000: Not indicated 001: 20 bits 010: 22 bits 100: 23 bits 101: 24 bits 110: 21 bits 111: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0030			Register Name: OWA_TX_CHSTA1
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	MWL Max Word Length 0: Maximum audio sample word length is 20 bits 1: Maximum audio sample word length is 24 bits

5.3.5.14 0x0034 OWA RX Channel Status Register0 (Default Value: 0x0000_0000)

Offset: 0x0034			Register Name: OWA_RX_CHSTA0
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:28	R/W	0x0	CA Clock Accuracy 00: Level 2 01: Level 1 10: Level 3 11: Not Matched
27:24	R/W	0x0	FREQ Sampling Frequency 0000: 44.1 kHz 0001: Not Indicated 0010: 48 kHz 0011: 32 kHz 0100: 22.05 kHz 0101: Reserved 0110: 24 kHz 0111: Reserved 1000: Reserved 1001: 768 kHz 1010: 96 kHz 1011: Reserved 1100: 176.4 kHz 1101: Reserved 1110: 192 kHz 1111: Reserved
23:20	R/W	0x0	CN Channel Number

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0034			Register Name: OWA_RX_CHSTA0
Bit	**Read/Write**	**Default/Hex**	**Description**
19:16	R/W	0x0	SN Source Number
15:8	R/W	0x0	CC Category Code Indicates the Kind of Equipment that Generates the digital audio interface Signal.
7:6	R/W	0x0	MODE Mode 00: Default mode 01 to 11: Reserved
5:3	R/W	0x0	EMP Emphasis Additional Format Information For bit 1 = '0', Linear PCM Audio mode: 000: 2 Audio channels without pre-emphasis 001: 2 Audio channels with 50 μs/15 μs pre-emphasis 010: Reserved (For 2 Audio channels with pre-emphasis) 011: Reserved (For 2 Audio channels with pre-emphasis) 100 to 111: Reserved For bit 1 = '1', Other than Linear PCM applications: 000: Default state 001 to 111: Reserved
2	R/W	0x0	CP Copyright 0: Copyright is asserted 1: No Copyright is asserted
1	R/W	0x0	TYPE Audio Data Type 0: Linear PCM samples 1: Non-linear PCM audio
0	R/W	0x0	PRO Application Type 0: Consumer application 1: Professional application

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.3.5.15 0x0038 OWA RX Channel Status Register1 (Default Value: 0x0000_0000)

Offset: 0x0038			Register Name: OWA_RX_CHSTA1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:10	/	/	/
9:8	R/W	0x0	CGMS_A 00: Copying is permitted without restriction 01: One generation of copies may be made 10: Condition is not be used 11: No copying is permitted
7:4	R/W	0x0	ORIG_FREQ Original Sampling Frequency 0000: Not indicated 0001: 192 kHz 0010: 12 kHz 0011: 176.4 kHz 0100: Reserved 0101: 96 kHz 0110: 8 kHz 0111: 88.2 kHz 1000: 16 kHz 1001: 24 kHz 1010: 11.025 kHz 1011: 22.05 kHz 1100: 32 kHz 1101: 48 kHz 1110: Reserved 1111: 44.1 kHz

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0038			Register Name: OWA_RX_CHSTA1
Bit	**Read/Write**	**Default/Hex**	**Description**
3:1	R/W	0x0	WL Sample Word Length For bit 0 = '0': 000: Not indicated 001: 16 bits 010: 18 bits 100: 19 bits 101: 20 bits 110: 17 bits 111: Reserved For bit 0 = '1': 000: Not indicated 001: 20 bits 010: 22 bits 100: 23 bits 101: 24 bits 110: 21 bits 111: Reserved
0	R/W	0x0	MWL Max Word Length 0: Maximum Audio sample word length is 20 bits 1: Maximum Audio sample word length is 24 bits

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4 Audio Codec

5.4.1 Overview

The Audio Codec has 2 I2S/PCM interfaces, 2-ch DAC, and 5-ch ADC with a high level of mixed-signal integration which is ideal for smartphones and other portable devices. The DRC with an integrated hardware digital audio processor (DAP) engine can be used in record and playback paths.

The Audio Codec has the following features:

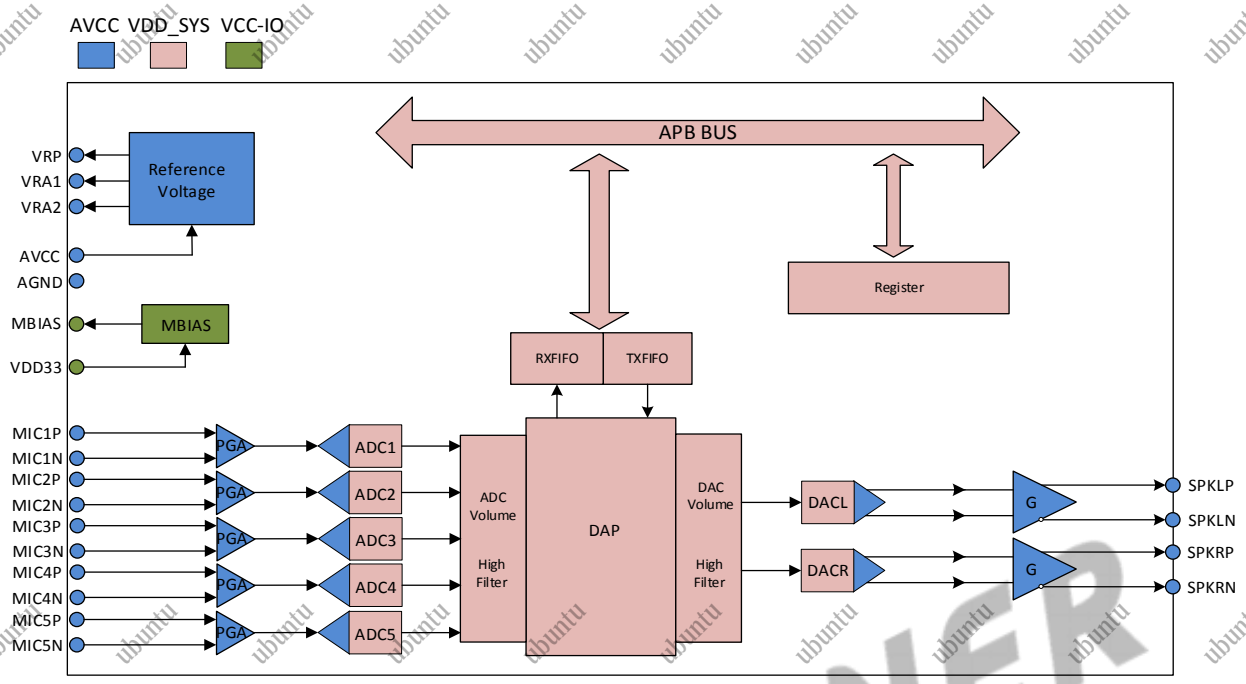
- Two audio digital-to-analog (DAC) channels
  - Supports the DAC sample rate from 8 kHz to 192 kHz
  - 100 ± 2 dB SNR@A-weight, -85 ± 3 dB THD + N
  - Supports 16-bit and 20-bit audio sample resolution
- Two audio outputs
  - Two differential speaker outputs (SPKLP/N and SPKRP/N)
- Five audio analog-to-digital (ADC) channels
  - Supports the ADC sample rate from 8 kHz to 48 kHz
  - 95 ± 3 dB SNR@A-weight, -80 ± 3 dB THD + N
  - Supports 16-bit and 20-bit audio sample resolution
- Five audio inputs
  - Five differential microphone inputs (MICIN1P/N, MICIN2P/N, MICIN3P/N, MICIN4P/N, and MICIN5P/N)
- Supports Dynamic Range Controller (DRC) adjusting the ADC recording and DAC playback
- One low-noise analog microphone bias output
- One 128x20-bit FIFO for DAC data transmit, one 256x20-bit FIFO for ADC data receive
- Programmable FIFO thresholds
- Supports interrupts and DMA

5.4.2 Block Diagram

The following figure shows the block diagram of Audio Codec.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 5-19 Audio Codec Block Diagram



5.4.3 Functional Descriptions

5.4.3.1 External Signals

Analog I/O Pins

Signal	Type	Description
MICIN1P	AI	Positive differential input for MIC1
MICIN1N	AI	Negative differential input for MIC1
MICIN2P	AI	Positive differential input for MIC2
MICIN2N	AI	Negative differential input for MIC2
MICIN3P	AI	Positive differential input for MIC3
MICIN3N	AI	Negative differential input for MIC3
MICIN4P	AI	Positive differential input for MIC4
MICIN4N	AI	Negative differential input for MIC4
MICIN5P	AI	Positive differential input for MIC5
MICIN5N	AI	Negative differential input for MIC5
SPKLP	AO	Left speaker positive output
SPKLN	AO	Left speaker negative output
SPKRP	AO	Right speaker positive output

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Signal	Type	Description
SPKRN	AO	Right speaker negative output

Reference

Signal	Type	Description
MBIAS	AO	First bias voltage output for main microphone
VRA1	AO	Internal reference voltage
VRA2	AO	Internal reference voltage
VRP	AO	Internal reference voltage

Power/Ground

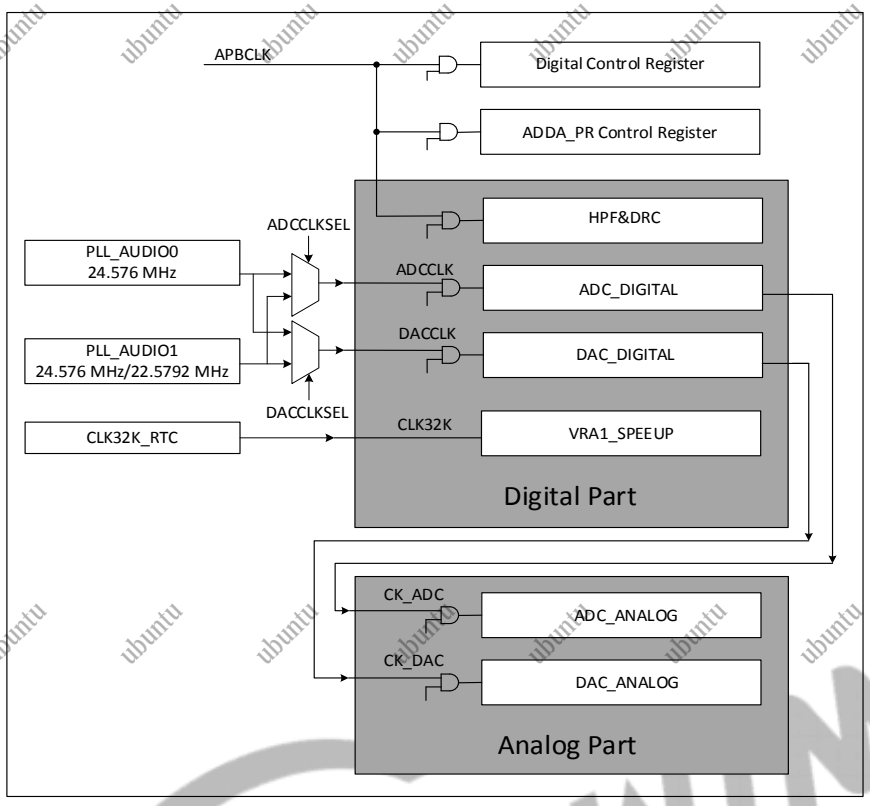
Signal	Type	Description
AVCC	P	Analog power
AGND	G	Analog ground

5.4.3.2 Clock Sources

The following figure describes the clock source of Audio Codec. For clock setting, configuration, and gating information, refer to section 3.3 "CCU".

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 5-20 Audio Codec Clock Diagram



The clock source for the digital part is the PLL_AUDIO0 and PLL_AUDIO1. For the ADC clock, configure **AUDIO_CODEC_ADC_CLK_REG**[25:24] to select the clock source. For the DAC clock, configure **AUDIO_CODEC_DAC_CLK_REG**[25:24] to select the clock source. The PK-PK jitter of PLL_AUDIO0 and PLL_AUDIO1 should be less than 200 ps.

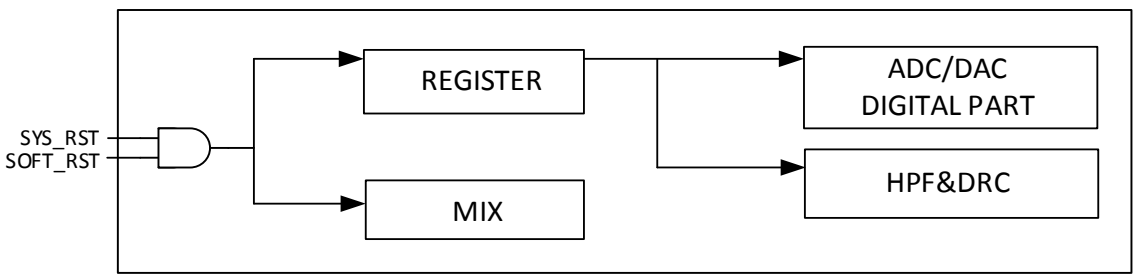
The clock source for the analog part is the CK_ADC and CK_DAC, both of which are divided from the digital part.

5.4.3.3 Reset System

Digital Part Reset System

The following figure shows the reset system of the audio codec digital part.

Figure 5-21 Audio Codec Digital Part Reset System



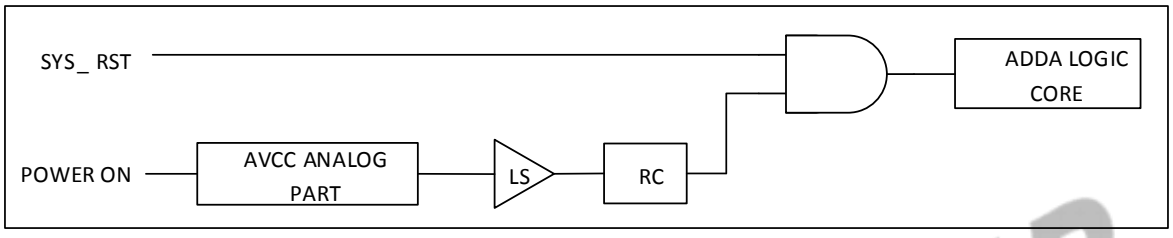
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

The SYS_RST comes from the VDD-SYS domain and is produced by the RTC domain. Each domain has the de-bounce to confirm the reset system is strong. For the codec register part, MIX can be reset by the SYS_RST when being powered on or the system soft is writing the reset control logic. The other parts can be reset by the soft configuration through writing the register.

Analog Part Reset System

The following figure shows the reset system of the audio codec analog part.

Figure 5-22 Audio Codec Analog Part Reset System



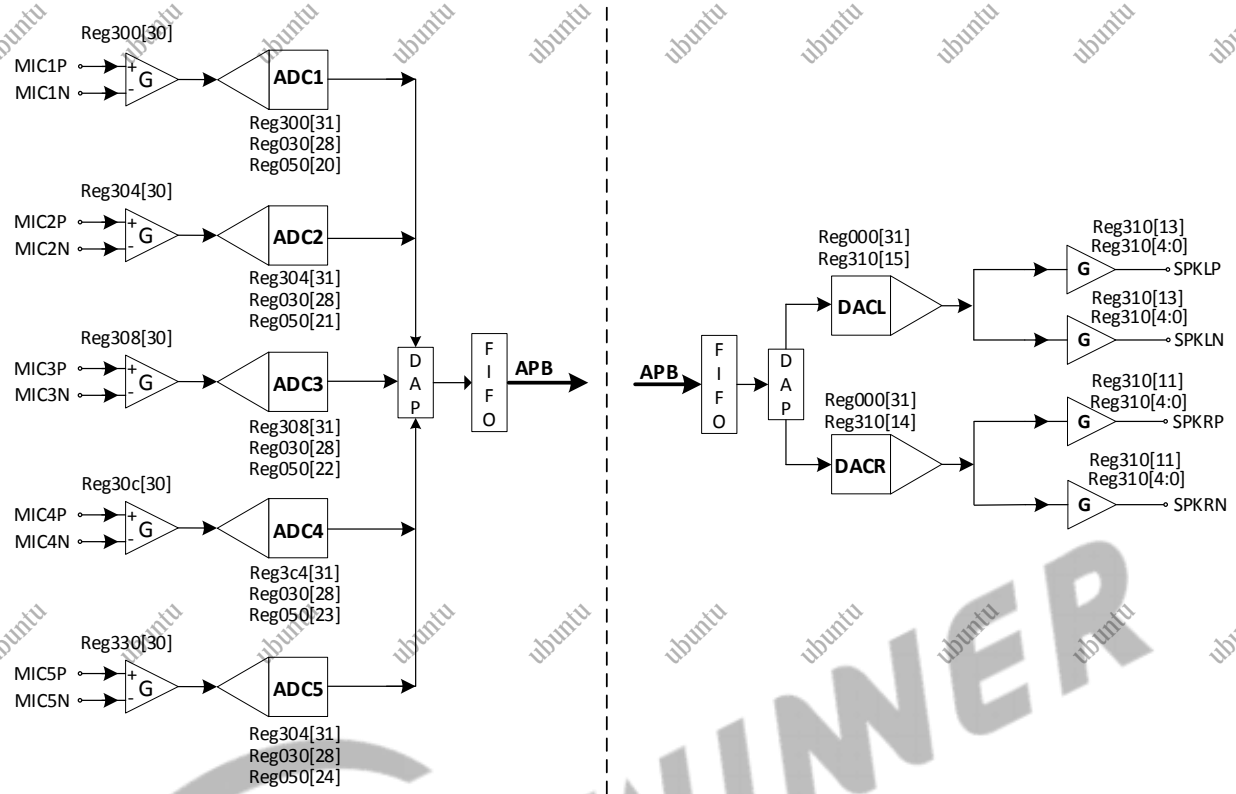
When AVCC is powered on, it sends the AVCC_POR signal. The AVCC_POR signal passes the level shift and RC filter part to the ADDA logic core.

5.4.3.4 Data Path Diagram

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 5-23 Audio Codec Data Path Diagram



5.4.3.5 Five ADCs

The five ADCs are used for recording stereo sound and a reference signal. The sample rates of the five ADCs are independent of the DAC sample rate. The digital ADC part can be enabled or disabled by the bit[28] of the **AC_ADC_FIFOC** register.

5.4.3.6 Stereo DAC

The stereo DAC sample rate can be configured by setting the register. To save power, the analog DACL can be enabled or disabled by setting the bit[15] of the **DAC_REG** register, and the analog DACR can be enabled or disabled by setting the bit[14] of the **DAC_REG** register. The digital DAC part can be enabled or disabled by the bit[31] of the **AC_DAC_DPC** register.

5.4.3.7 Analog Audio Input Path

The Audio Codec supports 5 analog audio input paths:

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

- MICIN1P/N
- MICIN2P/N
- MICIN3P/N
- MICIN4P/N
- MICIN5P/N

MICIN1P/N, MICIN2P/N, MICIN3P/N, MICIN4P/N, and MICIN5P/N provide differential input that can be mixed into the ADC record mixer. The MICIN is a high impedance, low capacitance input suitable for connecting to various differential microphones of different dynamics and sensitivity. The gain for each pre-amplifier can be set independently. MBIAS provides the reference voltage for electret condenser type (ECM) microphones.

5.4.3.8 Analog Audio Output Path

The Audio Codec has 2 analog output paths:

- SPKLP/N
- SPKRP/N

SPKLP/N and SPKRP/N provide two differential output to driveline signals to external audio equipment. The SPKLP/N outputs source are from DACL and the SPKRP/N outputs source are from DACR. The volume control is logarithmic with a 43.5 dB range in 1.5 dB step from -43.5 dB to 0 dB. The SPKLP/N output buffer power up or down by the bit[13] of the DAC_REG register and the SPKRP/N output buffer can power up or down by the bit[11] of DAC_REG.

5.4.3.9 Microphone BIAS (MBIAS)

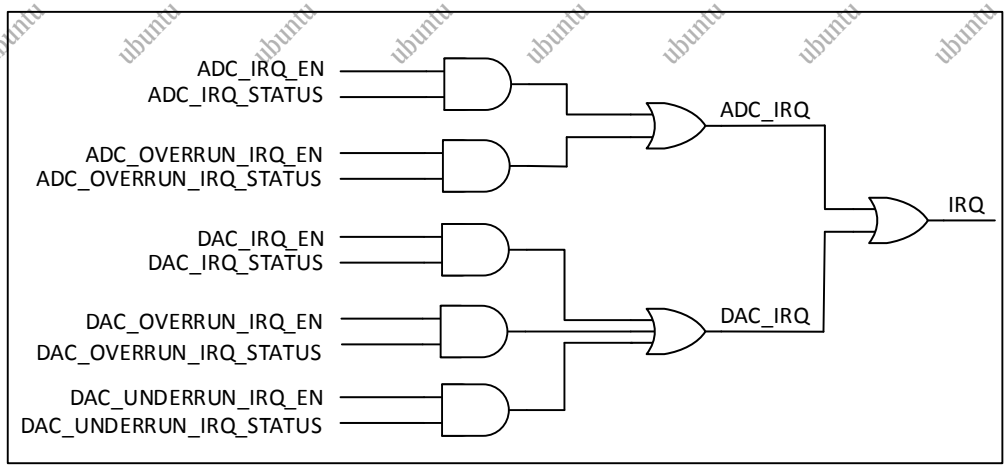
The MBIAS output provides a low noise reference voltage suitable for biasing electrets type microphones and the associated external resistor biasing network. Refer to the application information section for recommended external components.

5.4.3.10 Interrupts

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 5-24 Audio Codec Interrupt System

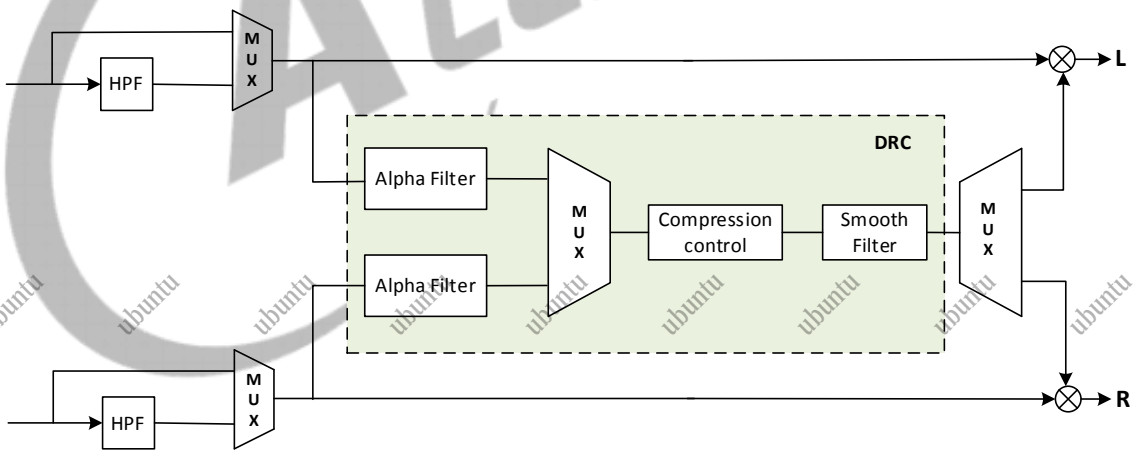


5.4.3.11 Digital Audio Processor (DAP)

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

The following figure shows the DAP data flow.

Figure 5-25 DAP Data Flow

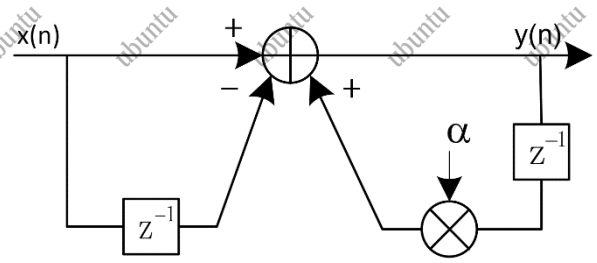


HPF

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 5-26 HPF Logic Structure



DRC

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

Figure 5-27 DRC Block Diagram

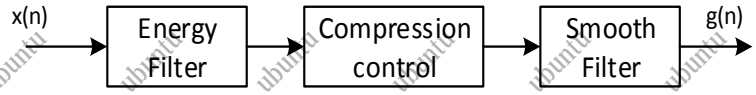
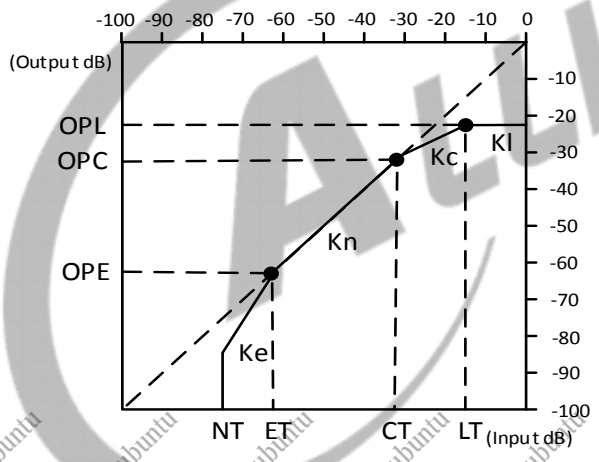


Figure 5-28 DRC Static Curve Parameters



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

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

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

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Configure the DRC parameters according to the following guidelines:

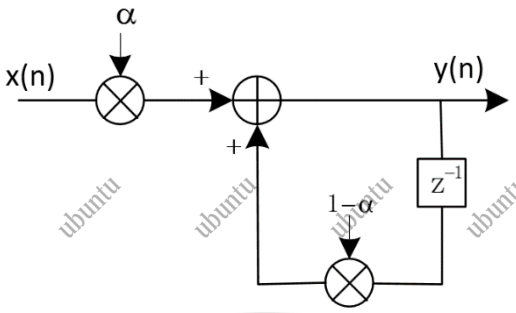
- Number format

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

- Energy Filter

  The following figure shows the structure of the energy filter.

  Figure 5-29 Energy Filter Structure



The Energy Filter is to estimate the RMS value of the audio data stream into DRC and has two parameters, which determine the time window over which RMS to be made. The parameter is computed by $\alpha = 1 - e^{-2.2Ts/ta}$.

For the Compression Control, there are ten parameters (ET, CT, LT, Ke, Kn, Kc, Kl, OPL, OPC, and OPE), which are all programmable, and the computation will be explained as follows.

- Threshold Parameter Computation (T parameter)

  The threshold is the value that determines the signal to be compressed or not. When the RMS of the signal is larger than the threshold, the signal will be compressed. The value of threshold input to the coefficient register is computed by $Tin = -\frac{T_{dB}}{6.0206}$.

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

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

  Therefore, CT_{in} = 6.644 = 0000 0110.1010 0100 1101 0011 1100 0000 = 0x06A4 D3C0 in 8.24 format.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

- Slope Parameter Computation (K parameter)

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

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

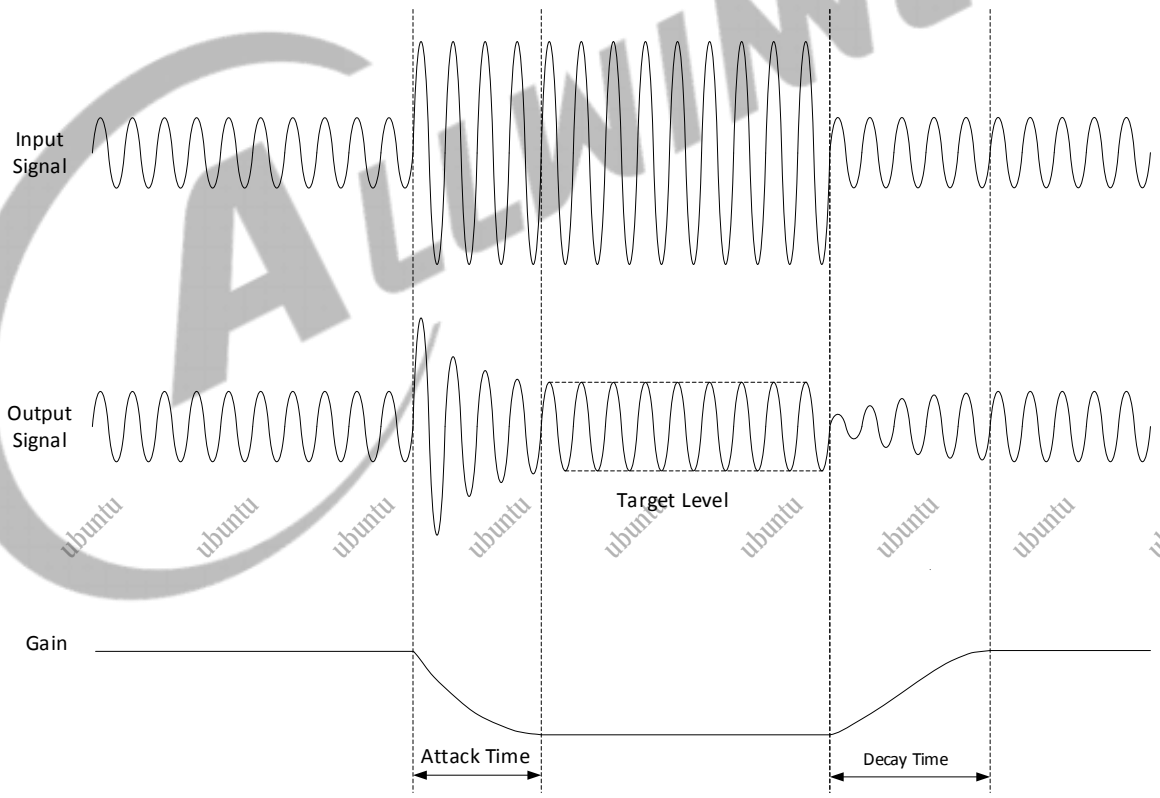
  For example, it is desired to set to 2:1, then the Kc requires to set to 2:1, is Kc = 1/2 = 0.5, Kc is entered as a 32-bit number in 8.24 format.

  Therefore, Kc = 0.5 = 0000 0000.1000 0000 0000 0000 0000 0000 = 0x0080 0000 in 8.24 format.

- Gain Smooth Filter

  The Gain Smooth Filter is to smooth the gain and control the ratio of gain increase and decrease. The decay time and attack are shown in Figure 5-30. The structure of the Gain Smooth filter is also the Alpha filter, so the rise time computation is the same as the Energy filter which is $\alpha = 1 - e^{-2.2Ts/ta}$.

Figure 5-30 Gain Smooth Filter



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.4 Programming Guidelines

5.4.4.1 Recording

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

Step 1 Codec initialization: configure **AUDIO_CODEC_BGR_REG** to open the audio codec bus clock gating and de-assert bus reset; configure **AUDIO_CODEC_ADC_CLK_REG** and **PLL_AUDIO0_CTRL_REG** to configure PLL_Audio0 frequency and enable PLL_Audio0. For details, refer to section 3.3 "CCU".

Step 2 Configure the sample rate and data transfer format, then open the ADC.

Step 3 Configure the DMA and DMA request.

Step 4 Enable the ADC DRQ and DMA.

5.4.4.2 Playing

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

Step 1 Codec initialization: configure **AUDIO_CODEC_BGR_REG** to open the audio codec bus clock gating and de-assert bus reset; configure **AUDIO_CODEC_DAC_CLK_REG** and **PLL_AUDIO1_CTRL_REG** to configure PLL_Audio1 frequency and enable PLL_Audio1. For details, refer to section 3.3 "CCU".

Step 2 Configure the sample rate and data transfer format, then open the DAC.

Step 3 Configure the DMA and DMA request.

Step 4 Enable the DAC DRQ and DMA.

5.4.5 Register List

Module Name	Base Address
Audio Codec	0x07032000

Register Name	Offset	Description
AC_DAC_DPC	0x0000	DAC Digital Part Control Register
DAC_VOL_CTRL	0x0004	DAC Volume Control Register
AC_DAC_FIFOC	0x0010	DAC FIFO Control Register
AC_DAC_FIFOS	0x0014	DAC FIFO Status Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
AC_DAC_TXDATA	0x0020	DAC TX DATA Register
AC_DAC_CNT	0x0024	DAC TX FIFO Counter Register
AC_DAC_DG	0x0028	DAC Debug Register
AC_ADC_FIFOC	0x0030	ADC FIFO Control Register
ADC_VOL_CTRL1	0x0034	ADC Volume Control1 Register
AC_ADC_FIFOS	0x0038	ADC FIFO Status Register
ADC_VOL_CTRL2	0x003C	ADC Volume Control2 Register
AC_ADC_RXDATA	0x0040	ADC RX Data Register
AC_ADC_CNT	0x0044	ADC RX Counter Register
AC_ADC_DG	0x004C	ADC Debug Register
ADC_DIG_CTRL	0x0050	ADC Digtial Control Register
VRA1SPEEDUP_DOWN_CTRL	0x0054	VRA1 Speedup Down Control Register
AC_DAC_DAP_CTRL	0x00F0	DAC DAP Control Register
AC_ADC_DAP_CTR	0x00F8	ADC DAP Control Register
AC_DAC_DRC_HHPFC	0x0100	DAC DRC High HPF Coef Register
AC_DAC_DRC_LHPFC	0x0104	DAC DRC Low HPF Coef Register
AC_DAC_DRC_CTRL	0x0108	DAC DRC Control Register
AC_DAC_DRC_LPFHAT	0x010C	DAC DRC Left Peak Filter High Attack Time Coef Register
AC_DAC_DRC_LPFLAT	0x0110	DAC DRC Left Peak Filter Low Attack Time Coef Register
AC_DAC_DRC_RPFLAT	0x0118	DAC DRC Peak Filter Low Attack Time Coef Register
AC_DAC_DRC_LPFHRT	0x011C	DAC DRC Left Peak Filter High Release Time Coef Register
AC_DAC_DRC_LPFLRT	0x0120	DAC DRC Left Peak Filter Low Release Time Coef Register
AC_DAC_DRC_LRMSHAT	0x012C	DAC DRC Left RMS Filter High Coef Register
AC_DAC_DRC_LRMSLAT	0x0130	DAC DRC Left RMS Filter Low Coef Register
AC_DAC_DRC_HCT	0x013C	DAC DRC Compressor Threshold High Setting Register
AC_DAC_DRC_LCT	0x0140	DAC DRC Compressor Slope High Setting Register
AC_DAC_DRC_HKC	0x0144	DAC DRC Compressor Slope High Setting Register
AC_DAC_DRC_LKC	0x0148	DAC DRC Compressor Slope Low Setting Register
AC_DAC_DRC_HOPC	0x014C	DAC DRC Compressor High Output at Compressor Threshold Register
AC_DAC_DRC_LOPC	0x0150	DAC DRC Compressor Low Output at Compressor Threshold Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
AC_DAC_DRC_HLT	0x0154	DAC DRC Limiter Threshold High Setting Register
AC_DAC_DRC_LLT	0x0158	DAC DRC Limiter Threshold Low Setting Register
AC_DAC_DRC_HKl	0x015C	DAC DRC Limiter Slope High Setting Register
AC_DAC_DRC_LKl	0x0160	DAC DRC Limiter Slope Low Setting Register
AC_DAC_DRC_HOPL	0x0164	DAC DRC Limiter High Output at Limiter Threshold
AC_DAC_DRC_LOPL	0x0168	DAC DRC Limiter Low Output at Limiter Threshold
AC_DAC_DRC_HET	0x016C	DAC DRC Expander Threshold High Setting Register
AC_DAC_DRC_LET	0x0170	DAC DRC Expander Threshold Low Setting Register
AC_DAC_DRC_HKE	0x0174	DAC DRC Expander Slope High Setting Register
AC_DAC_DRC_LKE	0x0178	DAC DRC Expander Slope Low Setting Register
AC_DAC_DRC_HOPE	0x017C	DAC DRC Expander High Output at Expander Threshold
AC_DAC_DRC_LOPE	0x0180	DAC DRC Expander Low Output at Expander Threshold
AC_DAC_DRC_HKN	0x0184	DAC DRC Linear Slope High Setting Register
AC_DAC_DRC_LKN	0x0188	DAC DRC Linear Slope Low Setting Register
AC_DAC_DRC_SFHAT	0x018C	DAC DRC Smooth filter Gain High Attack Time Coef Register
AC_DAC_DRC_SFLAT	0x0190	DAC DRC Smooth filter Gain Low Attack Time Coef Register
AC_DAC_DRC_SFHRT	0x0194	DAC DRC Smooth filter Gain High Release Time Coef Register
AC_DAC_DRC_SFLRT	0x0198	DAC DRC Smooth filter Gain Low Release Time Coef Register
AC_DAC_DRC_MXGHS	0x019C	DAC DRC MAX Gain High Setting Register
AC_DAC_DRC_MXGLS	0x01A0	DAC DRC MAX Gain Low Setting Register
AC_DAC_DRC_MNGHS	0x01A4	DAC DRC MIN Gain High Setting Register
AC_DAC_DRC_MNGLS	0x01A8	DAC DRC MIN Gain Low Setting Register
AC_DAC_DRC_EPSHC	0x01AC	DAC DRC Expander Smooth Time High Coef Register
AC_DAC_DRC_EPSLC	0x01B0	DAC DRC Expander Smooth Time Low Coef Register
AC_DAC_DRC_HPFHGAIN	0x01B8	DAC DRC HPF Gain High Coef Register
AC_DAC_DRC_HPFLGAIN	0x01BC	DAC DRC HPF Gain Low Coef Register
AC_ADC_DRC_HHPFC	0x0200	ADC DRC High HPF Coef Register
AC_ADC_DRC_LHPFC	0x0204	ADC DRC Low HPF Coef Register
AC_ADC_DRC_CTRL	0x0208	ADC DRC Control Register
AC_ADC_DRC_LPFHAT	0x020C	ADC DRC Left Peak Filter High Attack Time Coef Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
AC_ADC_DRC_LPFLAT	0x0210	ADC DRC Left Peak Filter Low Attack Time Coef Register
AC_ADC_DRC_LPFHRT	0x021C	ADC DRC Left Peak Filter High Release Time Coef Register
AC_ADC_DRC_LPFLRT	0x0220	ADC DRC Left Peak Filter Low Release Time Coef Register
AC_ADC_DRC_LRMSHAT	0x022C	ADC DRC Left RMS Filter High Coef Register
AC_ADC_DRC_LRMSLAT	0x0230	ADC DRC Left RMS Filter Low Coef Register
AC_ADC_DRC_HCT	0x023C	ADC DRC Compressor Threshold High Setting Register
AC_ADC_DRC_LCT	0x0240	ADC DRC Compressor Slope High Setting Register
AC_ADC_DRC_HKC	0x0244	ADC DRC Compressor Slope High Setting Register
AC_ADC_DRC_LKC	0x0248	ADC DRC Compressor Slope Low Setting Register
AC_ADC_DRC_HOPC	0x024C	ADC DRC Compressor High Output at Compressor Threshold Register
AC_ADC_DRC_LOPC	0x0250	ADC DRC Compressor Low Output at Compressor Threshold Register
AC_ADC_DRC_HLT	0x0254	ADC DRC Limiter Threshold High Setting Register
AC_ADC_DRC_LLT	0x0258	ADC DRC Limiter Threshold Low Setting Register
AC_ADC_DRC_HKl	0x025C	ADC DRC Limiter Slope High Setting Register
AC_ADC_DRC_LKl	0x0260	ADC DRC Limiter Slope Low Setting Register
AC_ADC_DRC_HOPL	0x0264	ADC DRC Limiter High Output at Limiter Threshold
AC_ADC_DRC_LOPL	0x0268	ADC DRC Limiter Low Output at Limiter Threshold
AC_ADC_DRC_HET	0x026C	ADC DRC Expander Threshold High Setting Register
AC_ADC_DRC_LET	0x0270	ADC DRC Expander Threshold Low Setting Register
AC_ADC_DRC_HKE	0x0274	ADC DRC Expander Slope High Setting Register
AC_ADC_DRC_LKE	0x0278	ADC DRC Expander Slope Low Setting Register
AC_ADC_DRC_HOPE	0x027C	ADC DRC Expander High Output at Expander Threshold
AC_ADC_DRC_LOPE	0x0280	ADC DRC Expander Low Output at Expander Threshold
AC_ADC_DRC_HKN	0x0284	ADC DRC Linear Slope High Setting Register
AC_ADC_DRC_LKN	0x0288	ADC DRC Linear Slope Low Setting Register
AC_ADC_DRC_SFHAT	0x028C	ADC DRC Smooth filter Gain High Attack Time Coef Register
AC_ADC_DRC_SFLAT	0x0290	ADC DRC Smooth filter Gain Low Attack Time Coef Register
AC_ADC_DRC_SFHRT	0x0294	ADC DRC Smooth filter Gain High Release Time Coef Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
AC_ADC_DRC_SFLRT	0x0298	ADC DRC Smooth filter Gain Low Release Time Coef Register
AC_ADC_DRC_MXGHS	0x029C	ADC DRC MAX Gain High Setting Register
AC_ADC_DRC_MXGLS	0x02A0	ADC DRC MAX Gain Low Setting Register
AC_ADC_DRC_MNGLS	0x02A4	ADC DRC MIN Gain High Setting Register
AC_ADC_DRC_MXGLS	0x02A8	ADC DRC MIN Gain Low Setting Register
AC_ADC_DRC_EPSHC	0x02AC	ADC DRC Expander Smooth Time High Coef Register
AC_ADC_DRC_EPSLC	0x02B0	ADC DRC Expander Smooth Time Low Coef Register
AC_ADC_DRC_HPFHGAIN	0x02B8	ADC DRC HPF Gain High Coef Register
AC_ADC_DRC_HPFLGAIN	0x02BC	ADC DRC HPF Gain Low Coef Register
Analog Domain Register		
ADC1_REG	0x0300	ADC1 Analog Control Register
ADC2_REG	0x0304	ADC2 Analog Control Register
ADC3_REG	0x0308	ADC3 Analog Control Register
ADC4_REG	0x030C	ADC4 Analog Control Register
DAC_REG	0x0310	DAC Analog Control Register
MICBIAS_REG	0x0318	MICBIAS Analog Control Register
RAMP_REG	0x031C	BIAS Analog Control Register
BIAS_REG	0x0320	BIAS Analog Control Register
ADC5_REG	0x0330	ADC5 Analog Control Register

5.4.6 Register Description

5.4.6.1 0x0000 DAC Digital Part Control Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: AC_DAC_DPC
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	EN_DA DAC Digital Part Enable 0: Disabled 1: Enabled
30:29	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: AC_DAC_DPC
Bit	Read/Write	Default/Hex	Description
28:25	R/W	0x0	MODQU Internal DAC Quantization Levels Levels = [7*(21 + MODQU[3:0])]/128 Default levels = 7*21/128 = 1.15
24	R/W	0x0	DWA DWA Function Disable 0: Enabled 1: Disabled
23:19	/	/	/
18	R/W	0x0	HPF_EN High Pass Filter Enable 0: Disabled 1: Enabled
17:12	R/W	0x0	DVOL Digital volume control: DVC, ATT = DVC[5:0]*(-1.16 dB) 64 steps, -1.16 dB/step
11:1	/	/	/
0	R/W	0x0	HUB_EN Audio Hub Enable The bit takes effect only when the EN_DA is set to 1. CPUX Domain: I2S0/I2S1/OWA TXFIFO Hub Enable. CPUS Domain: R_I2S0/Audio codec TXFIFO Hub Enable. 0: Disabled 1: Enabled

5.4.6.2 0x0004 DAC Volume Control Register (Default Value: 0x0000_A0A0)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.3 0x0010 DAC FIFO Control Register (Default Value: 0x0000_4000)

Offset: 0x0010			Register Name: AC_DAC_FIFOC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	R/W	0x0	DAC_FS Sample Rate of DAC 000: 48 kHz 010: 24 kHz 100: 12 kHz 110: 192 kHz 001: 32 kHz 011: 16 kHz 101: 8 kHz 111: 96 kHz 44.1 kHz/22.05 kHz/11.025 kHz can be supported by Audio PLL Configure Bit
28	R/W	0x0	FIR_VER FIR Version 0: 64-Tap FIR 1: 32-Tap FIR
27	/	/	/
26	R/W	0x0	SEND_LASAT Audio sample select when TX FIFO underrun 0: Sending zero 1: Sending the last audio sample
25:24	R/W	0x0	FIFO_MODE For 20-bit transmitted audio sample: 00/10: FIFO_I[19:0] = {TXDATA[31:12]} 01/11: FIFO_I[19:0] = {TXDATA[19:0]} For 16-bit transmitted audio sample: 00/10: FIFO_I[19:0] = {TXDATA[31:16], 4'b0} 01/11: FIFO_I[19:0] = {TXDATA[15:0], 4'b0}
23	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0010			Register Name: AC_DAC_FIFOC
Bit	**Read/Write**	**Default/Hex**	**Description**
22:21	R/W	0x0	DAC_DRQ_CLR_CNT When TX FIFO available room is less than or equal N, DRQ Request will be de-asserted. N is defined here: 00: IRQ/DRQ de-asserted when WLEVEL > TXTL 01: 4 10: 8 11: 16
20:15	/	/	/
14:8	R/W	0x40	TX_TRIG_LEVEL TX FIFO Empty Trigger Level (TXTL[12:0]) Interrupt and DMA request trigger level for TX FIFO normal condition. IRQ/DRQ generated when WLEVEL ≤ TXTL **Note: WLEVEL represents the number of valid samples in the TX FIFO. Only TXTL[6:0] valid when TXMODE = 0**
7	/	/	/
6	R/W	0x0	DAC_MONO_EN DAC Mono Enable 0: Stereo, 64 levels FIFO 1: Mono, 128 levels FIFO When enabled, L & R channel send same data.
5	R/W	0x0	TX_SAMPLE_BITS Transmitting Audio Sample Resolution 0: 16 bits 1: 20 bits
4	R/W	0x0	DAC_DRQ_EN DAC FIFO Empty DRQ Enable 0: Disabled 1: Enabled
3	R/W	0x0	DAC_IRQ_EN DAC FIFO Empty IRQ Enable 0: Disabled 1: Enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

5.4.6.4 0x0014 DAC FIFO Status Register (Default Value: 0x0080_8008)

Offset: 0x0014			Register Name: AC_DAC_FIFOS
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23	R	0x1	TX_EMPTY TX FIFO Empty 0: No room for new sample in TX FIFO 1: More than one room for new sample in TX FIFO (>= 1 word)
22:8	R	0x80	TXE_CNT TX FIFO Empty Space Word Counter
7:4	/	/	/
3	R/W1C	0x1	TXE_INT TX FIFO Empty Pending Interrupt 0: No Pending IRQ 1: FIFO Empty Pending Interrupt Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.
2	R/W1C	0x0	TXU_INT TX FIFO Underrun Pending Interrupt 0: No Pending Interrupt 1: FIFO Underrun Pending Interrupt Write '1' to clear this interrupt

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

5.4.6.5 0x0020 DAC TX DATA Register (Default Value: 0x0000_0000)

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

5.4.6.6 0x0024 DAC TX Counter Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.7 0x0028 DAC Debug Register (Default Value: 0x0000_0000)

Offset: 0x0028			Register Name: AC_DAC_DG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:12	/	/	/
11	R/W	0x0	DAC_MODU_SELECT DAC Modulator Debug 0: DAC Modulator Normal Mode 1: DAC Modulator Debug Mode
10:9	R/W	0x0	DAC_PATTERN_SELECT DAC Pattern Select 00: Normal (Audio Sample from TX FIFO) 01: -6 dB Sin wave 10: -60 dB Sin wave 11: Silent wave
8	R/W	0x0	CODEC_CLK_SELECT CODEC Clock Source Select 0: CODEC Clock from PLL 1: CODEC Clock from OSC (for Debug)
7	/	/	/
6	R/W	0x0	DA_SWP DAC Output Channel Swap Enable 0: Disabled 1: Enabled
5:3	/	/	/
2:0	R/W	0x0	ADDA_LOOP_MODE ADDA Loop Mode Select 000: Disabled 001: ADDA LOOP MODE DACL/DACR is connected to ADC1/ADC2. 010: ADDA LOOP MODE DACL/DACR is connected to ADC3/ADC4. 011: ADDA LOOP MODE DACL is connected to ADC5. 1xx: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.8 0x0030 ADC FIFO Control Register (Default Value: 0x0000_0400)

Offset: 0x0030			Register Name: AC_ADC_FIFOC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	R/W	0x0	ADFS Sample Rate of ADC 000: 48 kHz 010: 24 kHz 100: 12 kHz 110: Reserved 001: 32 kHz 011: 16 kHz 101: 8 kHz 111: Reserved 44.1 kHz/22.05 kHz/11.025 kHz can be supported by Audio PLL Configure Bit.
28	R/W	0x0	EN_AD ADC Digital Part Enable 0: Disabled 1: Enabled
27:26	R/W	0x0	ADCFDT ADC FIFO delay time for writing data after EN_AD 00:5ms 01:10ms 10:20ms 11:30ms
25	R/W	0x0	ADCDFEN ADC FIFO delay function for writing data after EN_AD 0: Disabled 1: Enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0030			Register Name: AC_ADC_FIFOC
Bit	**Read/Write**	**Default/Hex**	**Description**
24	R/W	0x0	RX_FIFO_MODE RX FIFO Output Mode (Mode 0, 1) 0: Expanding '0' at LSB of TX FIFO register 1: Expanding received sample sign bit at MSB of TX FIFO register For 20-bit received audio sample: Mode 0: RXDATA[31:0] = {FIFO_O[19:0], 12'h0} Mode 1: RXDATA[31:0] = {12{FIFO_O[19]}, FIFO_O[19:0]} For 16-bit received audio sample: Mode 0: RXDATA[31:0] = {FIFO_O[19:4], 16'h0} Mode 1: RXDATA[31:0] = {16{FIFO_O[19]}, FIFO_O[19:4]}
23	/	/	/
22	R/W	0x0	Reserved
21	R/W	0x0	RX_SYNC_EN_START The bit takes effect only when RX_SYNC_EN is set to 1. CPUX Domain: I2S0/I2S1/OWA RX Synchronize Enable Start. CPUS Domain: R_I2S0/Audiocodec/DMIC RX Synchronize Enable Start. 0: Disabled 1: Enabled
20	R/W	0x0	RX_SYNC_EN Audiocodec RX Synchronize Enable 0: Disabled 1: Enabled
19:17	/	/	/
16	R/W	0x0	RX_SAMPLE_BITS Receiving Audio Sample Resolution 0: 16 bits 1: 20 bits
15:12	/	/	/
11:4	R/W	0x40	RX_FIFO_TRG_LEVEL RX FIFO Trigger Level (RXTL[5:0]) Interrupt and DMA request trigger level for RX FIFO normal condition IRQ/DRQ generated when WLEVEL > RXTL[5:0] **Note: WLEVEL represents the number of valid samples in the RX FIFO.**

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0030			Register Name: AC_ADC_FIFOC
Bit	**Read/Write**	**Default/Hex**	**Description**
3	R/W	0x0	ADC_DRQ_EN ADC FIFO Data Available DRQ Enable 0: Disabled 1: Enabled
2	R/W	0x0	ADC_IRQ_EN ADC FIFO Data Available IRQ Enable 0: Disabled 1: Enabled
1	R/W	0x0	ADC_OVERRUN_IRQ_EN ADC FIFO Overrun IRQ Enable 0: Disabled 1: Enabled
0	R/WC	0x0	ADC_FIFO_FLUSH ADC FIFO Flush Write '1' to flush TX FIFO, self clear to '0'.

5.4.6.9 0x0034 ADC Volume Control1 Register (Default Value: 0xA0A0_A0A0)

Offset: 0x0034			Register Name: ADC_VOL_CTRL1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	R/W	0xA0	ADC4_VOL ADC4 channel volume (-119.25 dB To 71.25 dB, 0.75 dB/Step) 0x00: Mute 0x01: -119.25 dB ... 0x9F: -0.75 dB 0xA0: 0 dB 0xA1: 0.75 dB ... 0xFF: 71.25 dB

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.10 0x0038 ADC FIFO Status Register (Default Value: 0x0000_0001)

Offset: 0x0038			Register Name: AC_ADC_FIFOS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	/	/	/
23	R	0x0	RXA RX FIFO Available 0: No available data in RX FIFO 1: More than one sample in RX FIFO (>= 1 word)
22:17	/	/	/
16:8	R	0x0	RXA_CNT RX FIFO Available Sample Word Counter
7:4	/	/	/
3	R/W1C	0x0	RXA_INT RX FIFO Data Available Pending Interrupt 0: No Pending IRQ 1: Data Available Pending IRQ Write '1' to clear this interrupt or automatic clear if the interrupt condition fails.
2	/	/	/
1	R/W1C	0x0	RXO_INT RX FIFO Overrun Pending Interrupt 0: No Pending IRQ 1: FIFO Overrun Pending IRQ Write '1' to clear this interrupt.
0	R	0x1	Reserved

5.4.6.11 0x003C ADC Volume Control2 Register (Default Value: 0x0000_00A0)

Offset: 0x003C			Register Name: ADC_VOL_CTRL2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x003C			Register Name: ADC_VOL_CTRL2
Bit	**Read/Write**	**Default/Hex**	**Description**
7:0	R/W	0xA0	ADC5_VOL ADC5 channel volume (-119.25 dB to 71.25 dB, 0.75 dB/Step) 0x00: Mute 0x01: -119.25 dB ... 0x9F: -0.75 dB 0xA0: 0 dB 0xA1: 0.75 dB ... 0xFF: 71.25 dB

5.4.6.12 0x0040 ADC RX DATA Register (Default Value: 0x0000_0000)

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

5.4.6.13 0x0044 ADC RX Counter Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.14 0x004C ADC Debug Register (Default Value: 0x0000_0000)

Offset: 0x004C			Register Name: AC_ADC_DG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	/	/	/
26	R/W	0x0	AD_SWP3 ADC output channel swap enable (for digital filter) 0: Disabled 1: Enabled **Note: ADC5 and ADC6 (2`b0) swap data.**
25	R/W	0x0	AD_SWP2 ADC output channel swap enable (for digital filter) 0: Disabled 1: Enabled **Note: ADC3 and ADC4 swap data.**
24	R/W	0x0	AD_SWP1 ADC output channel swap enable (for digital filter) 0: Disabled 1: EnabledNote **Note: ADC1 and ADC2 swap data.**
23:0	/	/	/

5.4.6.15 0x0050 ADC Digtial Control Register (Default Value: 0x0000_0000)

Offset: 0x0050			Register Name: ADC_DIG_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:19	/	/	/
18	R/W	0x0	ADC5_VOL_EN ADC5 Volume Control Enable 0: Disabled 1: Enabled
17	R/W	0x0	ADC3_4_VOL_EN ADC3/4 Volume Control Enable 0: Disabled 1: Enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0050			Register Name: ADC_DIG_CTRL
Bit	Read/Write	Default/Hex	Description
16	R/W	0x0	ADC1_2_VOL_EN ADC1/2 Volume Control Enable 0: Disabled 1: Enabled
15:8	/	/	/
7:0	R/W	0x0	ADC_CHANNEL_EN Bit 7: ADC8 enabled Bit 6: ADC7 enabled Bit 5: ADC6 enabled Bit 4: ADC5 enabled Bit 3: ADC4 enabled Bit 2: ADC3 enabled Bit 1: ADC2 enabled Bit 0: ADC1 enabled

5.4.6.16 0x0054 VRA1 Speedup Down Control Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0054			Register Name: VRA1SPEEDUP_DOWN_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
0	R/W	0x0	VRA1SPEEDUP_DOWN_RST_CTRL VAR1Speedup Down RST Manual Control Enable 0: Disabled, VAR1Speedup Down convert 1 after bus rst release 32 ms. 1: Enabled, VAR1Speedup Down reset 0 immediately.

5.4.6.17 0x00F0 DAC DAP Control Register (Default Value: 0x0000_0000)

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

5.4.6.18 0x00F8 ADC DAP Control Register (Default Value: 0x0000_0000)

Offset: 0x00F8			Register Name: AC_ADC_DAP_CTR
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	ADC_DAP0_EN (control the DAP of ADC1/2) DAP for ADC enable 0:Bypassed 1: Enabled
30	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x00F8			Register Name: AC_ADC_DAP_CTR
Bit	**Read/Write**	**Default/Hex**	**Description**
29	R/W	0x0	ADC_DRC0_EN ADC DRC0 enable control 0: Disabled 1: Enabled
28	R/W	0x0	ADC_HPF0_EN ADC HPF0 enable control 0: Disabled 1: Enabled
27	R/W	0x0	ADC_DAP1_EN (control the DAP of ADC3/4) The DAP controls the DAC of ADCX/Y.
26	/	/	/
25	R/W	0x0	ADC_DRC1_EN ADC DRC1 enable control 0: Disabled 1: Enabled
24	R/W	0x0	ADC_HPF1_EN ADC HPF1 enable control 0: Disabled 1: Enabled
23	R/W	0x0	ADC_DAP2_EN (control the DAP of ADC5)
22	/	/	/
21	R/W	0x0	ADC_DRC2_EN ADC DRC2 enable control 0: Disabled 1: Enabled
20	R/W	0x0	ADC_HPF2_EN ADC HPF2 enable control 0: Disabled 1: Enabled
19:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.19 0x0100 DAC DRC High HPF Coef Register (Default Value: 0x0000_00FF)

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

5.4.6.20 0x0104 DAC DRC Low HPF Coef Register (Default Value: 0x0000_FAC1)

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

5.4.6.21 0x0108 DAC DRC Control Register (Default Value: 0x0000_0080)

Offset: 0x0108			Register Name: AC_DAC_DRC_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R	0x0	DRC delay buffer data output state when The DRC delay function is enabled and the DRC function is disabled. After disabling the DRC function and this bit goes to 0, write the DRC delay function bit to 0. 0: Not completed 1: Completed
14:10	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0108			Register Name: AC_DAC_DRC_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
13:8	R/W	0x0	Signal delay time setting 6'h00: (8x1) fs 6'h01: (8x2) fs 6'h02: (8x3) fs ---------- 6'h2e: (8*47) fs 6'h2f: (8*48) fs 6'h30 -- 6'h3f: (8*48) fs Delay time = 8*(n + 1) fs, n<6'h30; When the delay function is disabled, the signal delay time is unused.
7	R/W	0x1	The delay buffer use or not when the DRC is disabled and the DRC buffer data output completely. 0: Do not use the buffer. 1: Use the buffer.
6	R/W	0x0	DRC gain max limit enable 0: Disabled 1: Enabled
5	R/W	0x0	DRC gain min limit enable When this function is enabled, it will overwrite the noise detect function. 0: Disabled 1: Enabled
4	R/W	0x0	Control the DRC to detect noise when ET is enabled. 0: Disabled 1: Enabled
3	R/W	0x0	Signal function select 0: RMS filter 1: Peak filter When the signal function selects the Peak filter, the RMS parameter is unused. (AC_DRC_LRMSHAT, AC_DRC_LRMSLAT, AC_DRC_LRMSHAT, AC_DRC_LRMSLAT) When the signal function selects the RMS filter, the Peak filter parameter is unused. (AC_DRC_LPFHAT, AC_DRC_LPFLAT, AC_DRC_RPFHAT, AC_DRC_RPFLAT, AC_DRC_LPFHRT, AC_DRC_LPFLRT, AC_DRC_RPFHRT, and AC_DRC_RPFLRT)

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0108			Register Name: AC_DAC_DRC_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
2	R/W	0x0	Delay function enable 0: Disabled 1: Enabled When the bit is disabled, the signal delay time is unused.
1	R/W	0x0	DRC LT enable 0: Disabled 1: Enabled When the bit is disabled, Kl and OPL parameter is unused.
0	R/W	0x0	DRC ET enable 0: Disabled 1: Enabled When the bit is disabled, Ke and OPE parameter is unused.

5.4.6.22 0x010C DAC DRC Left Peak Filter High Attack Time Coef Register (Default Value: 0x0000_000B)

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

5.4.6.23 0x0110 DAC DRC Left Peak Filter Low Attack Time Coef Register (Default Value: 0x0000_77BF)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.24 0x0118 DAC DRC Left Peak Filter Low Attack Time Coef Register (Default Value: 0x0000_77BF)

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

5.4.6.25 0x011C DAC DRC Left Peak Filter High Release Time Coef Register (Default Value: 0x0000_00FF)

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

5.4.6.26 0x0120 DAC DRC Left Peak Filter Low Release Time Coef Register (Default Value: 0x0000_E1F8)

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

5.4.6.27 0x012C DAC DRC Left RMS Filter High Coef Register (Default Value: 0x0000_0001)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.28 0x0130 DAC DRC Left RMS Filter Low Coef Register (Default Value: 0x0000_2BAF)

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

5.4.6.29 0x013C DAC DRC Compressor Theshold High Setting Register (Default Value: 0x0000_06A4)

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

5.4.6.30 0x0140 DAC DRC Compressor Slope High Setting Register (Default Value: 0x0000_D3C0)

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

5.4.6.31 0x0144 DAC DRC Compressor Slope High Setting Register (Default Value: 0x0000_0080)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

5.4.6.32 0x0148 DAC DRC Compressor Slope Low Setting Register (Default Value: 0x0000_0000)

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

5.4.6.33 0x014C DAC DRC Compressor High Output at Compressor Threshold Register (Default Value: 0x0000_F95B)

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

5.4.6.34 0x0150 DAC DRC Compressor Low Output at Compressor Threshold Register (Default Value: 0x0000_2C3F)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.35 0x0154 DAC DRC Limiter Theshold High Setting Register (Default Value: 0x0000_01A9)

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

5.4.6.36 0x0158 DAC DRC Limiter Theshold Low Setting Register (Default Value: 0x0000_34F0)

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

5.4.6.37 0x015C DAC DRC Limiter Slope High Setting Register (Default Value: 0x0000_0005)

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

5.4.6.38 0x0160 DAC DRC Limiter Slope Low Setting Register (Default Value: 0x0000_1EB8)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.39 0x0164 DAC DRC Limiter High Output at Limiter Threshold Register (Default Value: 0x0000_FBD8)

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

5.4.6.40 0x0168 DAC DRC Limiter Low Output at Limiter Threshold Register (Default Value: 0x0000_FBA7)

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

5.4.6.41 0x016C DAC DRC Expander Theshold High Setting Register (Default Value: 0x0000_0BA0)

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

5.4.6.42 0x0170 DAC DRC Expander Theshold Low Setting Register (Default Value: 0x0000_7291)

Offset: 0x0170			Register Name: AC_DAC_DRC_LET
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x7291	The expander threshold setting, which is set by the equation that ETin = -ET/6.0206. The format is 8.24. (The default value is -70 dB)

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.43 0x0174 DAC DRC Expander Slope High Setting Register (Default Value: 0x0000_0500)

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

5.4.6.44 0x0178 DAC DRC Expander Slope Low Setting Register (Default Value: 0x0000_0000)

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

5.4.6.45 0x017C DAC DRC Expander High Output at Expander Threshold Register (Default Value: 0x0000_F45F)

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

5.4.6.46 0x0180 DAC DRC Expander Low Output at Expander Threshold Register (Default Value: 0x0000_8D6E)

Offset: 0x0180			Register Name: AC_DAC_DRC_LOPE
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x8D6E	The output of the expander which determine by equation OPE/6.0206. The format is 8.24. (The default value is -70 dB)

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.47 0x0184 DAC DRC Linear Slope High Setting Register (Default Value: 0x0000_0100)

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

5.4.6.48 0x0188 DAC DRC Linear Slope Low Setting Register (Default Value: 0x0000_0000)

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

5.4.6.49 0x018C DAC DRC Smooth Filter Gain High Attack Time Coef Register (Default Value: 0x0000_0002)

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

5.4.6.50 0x0190 DAC DRC Smooth Filter Gain Low Attack Time Coef Register (Default Value: 0x0000_5600)

Offset: 0x0190			Register Name: AC_DAC_DRC_SFLAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x5600	The smooth filter attack time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/tr). The format is 3.24. (The default value is 5 ms)

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.51 0x0194 DAC DRC Smooth Filter Gain High Release Time Coef Register (Default Value: 0x0000_0000)

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

5.4.6.52 0x0198 DAC DRC Smooth Filter Gain Low Release Time Coef Register (Default Value: 0x0000_0F04)

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

5.4.6.53 0x019C DAC DRC MAX Gain High Setting Register (Default Value: 0x0000_FE56)

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

5.4.6.54 0x01A0 DAC DRC MAX Gain Low Setting Register (Default Value: 0x0000_CB0F)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x01A0			Register Name: AC_DAC_DRC_MXGLS
Bit	Read/Write	Default/Hex	Description
15:0	R/W	0xCB0F	The max gain setting, which is determined by equation MXG/6.0206. The format is 8.24 and must -20 dB < MXG < 30 dB (The default value is -10 dB)

5.4.6.55 0x01A4 DAC DRC MIN Gain High Setting Register (Default Value: 0x0000_F95B)

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

5.4.6.56 0x01A8 DAC DRC MIN Gain Low Setting Register (Default Value: 0x0000_2C3F)

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

5.4.6.57 0x01AC DAC DRC Expander Smooth Time High Coef Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.58 0x01B0 DAC DRC Expander Smooth Time Low Coef Register (Default Value: 0x0000_640C)

Offset: 0x01B0			Register Name: AC_DAC_DRC_EPSLC
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x640C	The gain smooth filter release and attack time parameter setting in expander region, which are determined by the equation that RT = 1-exp (-2.2Ts/tr). The format is 3.24. (The default value is 30 ms)

5.4.6.59 0x01B8 DAC DRC HPF Gain High Coef Register (Default Value: 0x0000_0100)

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

5.4.6.60 0x01BC DAC DRC HPF Gain Low Coef Register (Default Value: 0x0000_0000)

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

5.4.6.61 0x0200 ADC DRC High HPF Coef Register (Default Value: 0x0000_00FF)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.62 0x0204 ADC DRC Low HPF Coef Register (Default Value: 0x0000_FAC1)

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

5.4.6.63 0x0208 ADC DRC Control Register (Default Value: 0x0000_0080)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0208			Register Name: AC_ADC_DRC_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
5	R/W	0x0	DRC gain min limit enable. When this fuction is enabled, it will overwrite the noise detect function. 0: Disabled 1: Enabled
4	R/W	0x0	Control the DRC to detect noise when ET is enabled 0: Disabled 1: Enabled
3	R/W	0x0	Signal function select 0: RMS filter 1: Peak filter When the signal function selects the Peak filter, the RMS parameter is unused. (AC_DRC_LRMSHAT, AC_DRC_LRMSLAT, AC_DRC_LRMSHAT, and AC_DRC_LRMSLAT) When the signal function selects the RMS filter, the Peak filter parameter is unused. (AC_DRC_LPFHAT, AC_DRC_LPFLAT, AC_DRC_RPFHAT, AC_DRC_RPFLAT, AC_DRC_LPFHRT, AC_DRC_LPFLRT, AC_DRC_RPFHRT, and AC_DRC_RPFLRT)
2	R/W	0x0	Delay function enable 0: Disabled 1: Enabled When the bit is disabled, the signal delay time is unused.
1	R/W	0x0	DRC LT enable 0: Disabled 1: Enabled When the bit is disabled, Kl and OPL parameter is unused.
0	R/W	0x0	DRC ET enable 0: Disabled 1: Enabled When the bit is disabled, Ke and OPE parameter is unused.

5.4.6.64 0x020C ADC DRC Left Peak Filter High Attack Time Coef Register (Default Value: 0x0000_000B)

Offset: 0x020C			Register Name: AC_ADC_DRC_LPFHAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

5.4.6.65 0x0210 ADC DRC Left Peak Filter Low Attack Time Coef Register (Default Value: 0x0000_77BF)

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

5.4.6.66 0x021C ADC DRC Left Peak Filter High Release Time Coef Register (Default Value: 0x0000_00FF)

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

5.4.6.67 0x0220 ADC DRC Left Peak Filter Low Release Time Coef Register (Default Value: 0x0000_E1F8)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.68 0x022C ADC DRC Left RMS Filter High Coef Register (Default Value: 0x0000_0001)

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

5.4.6.69 0x0230 ADC DRC Left RMS Filter Low Coef Register (Default Value: 0x0000_2BAF)

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

5.4.6.70 0x023C ADC DRC Compressor Theshold High Setting Register (Default Value: 0x0000_06A4)

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

5.4.6.71 0x0240 ADC DRC Compressor Slope High Setting Register (Default Value: 0x0000_D3C0)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.72 0x0244 ADC DRC Compressor Slope High Setting Register (Default Value: 0x0000_0080)

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

5.4.6.73 0x0248 ADC DRC Compressor Slope Low Setting Register (Default Value: 0x0000_0000)

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

5.4.6.74 0x024C ADC DRC Compressor High Output at Compressor Threshold Register (Default Value: 0x0000_F95B)

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

5.4.6.75 0x0250 ADC DRC Compressor Low Output at Compressor Threshold Register (Default Value: 0x0000_2C3F)

Offset: 0x0250			Register Name: AC_ADC_DRC_LOPC
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

5.4.6.76 0x0254 ADC DRC Limiter Theshold High Setting Register (Default Value: 0x0000_01A9)

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

5.4.6.77 0x0258 ADC DRC Limiter Theshold Low Setting Register (Default Value: 0x0000_34F0)

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

5.4.6.78 0x025C ADC DRC Limiter Slope High Setting Register (Default Value: 0x0000_0005)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.79 0x0260 ADC DRC Limiter Slope Low Setting Register (Default Value: 0x0000_1EB8)

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

5.4.6.80 0x0264 ADC DRC Limiter High Output at Limiter Threshold Register (Default Value: 0x0000_FBD8)

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

5.4.6.81 0x0268 ADC DRC Limiter Low Output at Limiter Threshold Register (Default Value: 0x0000_FBA7)

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

5.4.6.82 0x026C ADC DRC Expander Theshold High Setting Register (Default Value: 0x0000_0BA0)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.83 0x0270 ADC DRC Expander Theshold Low Setting Register (Default Value: 0x0000_7291)

Offset: 0x0270			Register Name: AC_ADC_DRC_LET
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x7291	The expander threshold setting, which is set by the equation that ETin = -ET/6.0206, The format is 8.24. (The default value is -70 dB)

5.4.6.84 0x0274 ADC DRC Expander Slope High Setting Register (Default Value: 0x0000_0500)

Offset: 0x0274			Register Name: AC_ADC_DRC_HKE
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13:0	R/W	0x0500	The slope of the expander, which is determined by the equation that Ke = 1/R. R is the ratio of the expander, which is always an integer and the ke must larger than 50. The format is 8.24. (The default value is <1:5>)

5.4.6.85 0x0278 ADC DRC Expander Slope Low Setting Register (Default Value: 0x0000_0000)

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

5.4.6.86 0x027C ADC DRC Expander High Output at Expander Threshold Register (Default Value: 0x0000_F45F)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.87 0x0280 ADC DRC Expander Low Output at Expander Threshold Register (Default Value: 0x0000_8D6E)

Offset: 0x0280			Register Name: AC_ADC_DRC_LOPE
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x8D6E	The output of the expander, which is determined by equation OPE/6.0206. The format is 8.24. (The default value is -70 dB)

5.4.6.88 0x0284 ADC DRC Linear Slope High Setting Register (Default Value: 0x0000_0100)

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

5.4.6.89 0x0288 ADC DRC Linear Slope Low Setting Register (Default Value: 0x0000_0000)

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

5.4.6.90 0x028C ADC DRC Smooth Filter Gain High Attack Time Coef Register (Default Value: 0x0000_0002)

Offset: 0x028C			Register Name: AC_ADC_DRC_SFHAT
Bit	Read/Write	Default/Hex	Description
31:11	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

5.4.6.91 0x0290 ADC DRC Smooth Filter Gain Low Attack Time Coef Register (Default Value: 0x0000_5600)

Offset: 0x0290			Register Name: AC_ADC_DRC_SFLAT
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x5600	The smooth filter attack time parameter setting, which is determined by the equation that AT = 1-exp (-2.2Ts/tr). The format is 3.24. (The default value is 5 ms)

5.4.6.92 0x0294 ADC DRC Smooth Filter Gain High Release Time Coef Register (Default Value: 0x0000_0000)

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

5.4.6.93 0x0298 ADC DRC Smooth Filter Gain Low Release Time Coef Register (Default Value: 0x0000_0F04)

Offset: 0x0298			Register Name: AC_ADC_DRC_SFLRT
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x0F04	The gain smooth filter release time parameter setting, which is determined by the equation that RT = 1-exp (-2.2Ts/tr). The format is 3.24. (The default value is 200 ms)

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.94 0x029C ADC DRC MAX Gain High Setting Register (Default Value: 0x0000_FE56)

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

5.4.6.95 0x02A0 ADC DRC MAX Gain Low Setting Register (Default Value: 0x0000_CB0F)

Offset: 0x02A0			Register Name: AC_ADC_DRC_MXGLS
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0xCB0F	The max gain setting, which is determined by equation MXG/6.0206. The format is 8.24 and must -20 dB < MXG < 30 dB (The default value is -10 dB)

5.4.6.96 0x02A4 ADC DRC MIN Gain High Setting Register (Default Value: 0x0000_F95B)

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

5.4.6.97 0x02A8 ADC DRC MIN Gain Low Setting Register (Default Value: 0x0000_2C3F)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.98 0x02AC ADC DAP Expander Smooth Time High Coef Register (Default Value: 0x0000_0000)

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

5.4.6.99 0x02B0 ADC DRC Expander Smooth Time Low Coef Register (Default Value: 0x0000_640C)

Offset: 0x02B0			Register Name: AC_ADC_DRC_EPSLC
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x640C	The gain smooth filter release and attack time parameter setting in expander region, which is determined by the equation that RT = 1-exp (-2.2Ts/tr). The format is 3.24. (The default value is 30 ms)

5.4.6.100 0x02B8 ADC DRC HPF Gain High Coef Register (Default Value: 0x0000_0100)

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

5.4.6.101 0x02BC ADC DRC HPF Gain Low Coef Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.102 0x0300 ADC1 Analog Control Register (Default Value: 0x000C_C055)

Offset: 0x0300			Register Name: ADC1_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	ADC1_EN ADC1 Channel Enable 0: Disabled 1: Enabled
30	R/W	0x0	MIC1_PGA_EN MIC1 PGA Enable 0: Disabled 1: Enabled
29	R/W	0x0	ADC1 Dither Control 0: New Dither Off 1: New Dither On
28:26	/	/	/
25:24	R/W	0x0	DSM_DITHER_LVL Dither Level Control (Dither level is positive ralated to the ctrl bits) 0: No Level 1: Min Level 2: Middle Level 3: Max Level
23:20	/	/	/
19:18	R/W	0x3	ADC1_PGA_CTRL_RCM ADC1 PGA Common Mode Input Impedance Control for MICIN 0: 100 kΩ 1: 75 kΩ 2: 50 kΩ 3: 25 kΩ
17:16	R/W	0x0	ADC1_PGA_IN_VCM_CTRL ADC1 PGA Common-Mode Voltage Control 0: 900 mV 1: 800 mV 2: 750 mV 3: 700 mV

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0300			Register Name: ADC1_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
15:14	R/W	0x3	IOPADC ADC1-ADC5 Bias Current Select 00: 1 uA 01: 2 uA 10: 3 uA 11: 4 uA
13	/	/	/
12:8	R/W	0x0	ADC1_PGA_GAIN_CTRL ADC1 PGA gain settings: 0x0: 0 dB 0x10: 21 dB 0x1: 6 dB 0x11: 22 dB 0x2: 6 dB 0x12: 23 dB 0x3: 6 dB 0x13: 24 dB 0x4: 9 dB 0x14: 25 dB 0x5: 10 dB 0x15: 26 dB 0x6: 11 dB 0x16: 27 dB 0x7: 12 dB 0x17: 28 dB 0x8: 13 dB 0x18: 29 dB 0x9: 14 dB 0x19: 30 dB 0xA: 15 dB 0x1A: 31 dB 0xB: 16 dB 0x1B: 32 dB 0xC: 17 dB 0x1C: 33 dB 0xD: 18 dB 0x1D: 34 dB 0xE: 19 dB 0x1E: 35 dB 0xF: 20 dB 0x1F: 36 dB

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0300			Register Name: ADC1_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
7:6	R/W	0x1	ADC1_IOPAAF ADC1 OP AAF Bias Current Select 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA For example: ADC1_REG<15:14> = 11，IOPADC = 4 uA 00: 1.50*4 uA = 6 uA 01: 1.75*4 uA = 7 uA 10: 2.00*4 uA = 8 uA 11: 2.25*4 uA = 9 uA
5:4	R/W	0x1	ADC1_IOPSDM1 ADC1 OP SDM Bias Current Select 1 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA
3:2	R/W	0x1	ADC1_IOPSDM2 ADC1 OP SDM Bias Current Select 2 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA
1:0	R/W	0x1	ADC1_IOPMIC ADC1 OP MIC Bias Current Select 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.103 0x0304 ADC2 Analog Control Register (Default Value: 0x000C_0055)

Offset: 0x0304			Register Name: ADC2_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	ADC2_EN ADC2 Channel Enable 0: Disabled 1: Enabled
30	R/W	0x0	MIC2_PGA_EN MIC2 PGA Enable 0: Disabled 1: Enabled
29	R/W	0x0	ADC2 Dither Control 0: New Dither Off 1: New Dither On
28:26	/	/	/
25:24	R/W	0x0	DSM_DITHER_LVL Dither Level Control (Dither level is positive ralated to the ctrl bits) 00: No Level 01: Min Level 10: Middle Level 11: Max Level
23:20	/	/	/
19:18	R/W	0x3	ADC2_PGA_CTRL_RCM ADC2 PGA Common Mode Input Impedance Control for MICIN 00: 100 kΩ 01: 75 kΩ 10: 50 kΩ 11: 25 kΩ
17:16	R/W	0x0	ADC2_PGA_IN_VCM_CTRL ADC2 PGA Common-Mode Voltage Control 00: 900 mV 01: 800 mV 10: 750 mV 11: 700 mV
15:13	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0304			Register Name: ADC2_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
12:8	R/W	0x0	ADC2_PGA_GAIN_CTRL ADC2 PGA gain settings: 0x0: 0 dB 0x10: 21 dB 0x1: 6 dB 0x11: 22 dB 0x2: 6 dB 0x12: 23 dB 0x3: 6 dB 0x13: 24 dB 0x4: 9 dB 0x14: 25 dB 0x5: 10 dB 0x15: 26 dB 0x6: 11 dB 0x16: 27 dB 0x7: 12 dB 0x17: 28 dB 0x8: 13 dB 0x18: 29 dB 0x9: 14 dB 0x19: 30 dB 0xA: 15 dB 0x1A: 31 dB 0xB: 16 dB 0x1B: 32 dB 0xC: 17 dB 0x1C: 33 dB 0xD: 18 dB 0x1D: 34 dB 0xE: 19 dB 0x1E: 35 dB 0xF: 20 dB 0x1F: 36 dB
7:6	R/W	0x1	ADC2_IOPAAF ADC2 OP AAF Bias Current Select 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA
5:4	R/W	0x1	ADC2_IOPSDM1 ADC2 OP SDM Bias Current Select 1 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

5.4.6.104 0x0308 ADC3 Analog Control Register (Default Value: 0x000C_0055)

Offset: 0x0308			Register Name: ADC3_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	ADC3_EN ADC3 Channel Enable 0: Disabled 1: Enabled
30	R/W	0x0	MIC3_PGA_EN MIC3 PGA Enable 0: Disabled 1: Enabled
29	R/W	0x0	ADC3 Dither Control 0: New Dither Off 1: New Dither On
28:26	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0308			Register Name: ADC3_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
25:24	R/W	0x0	DSM_DITHER_LVL Dither Level Control (Dither level is positive ralated to the ctrl bits) 00: No Level 01: Min Level 10: Middle Level 11: Max Level
23:20	/	/	/
19:18	R/W	0x3	ADC3_PGA_CTRL_RCM ADC3 PGA Common Mode Input Impedance Control for MICIN 00: 100 kΩ 01: 75 kΩ 10: 50 kΩ 11: 25 kΩ
17:16	R/W	0x0	ADC3_PGA_IN_VCM_CTRL ADC3 PGA Common-Mode Voltage Control 00: 900 mV 01: 800 mV 10: 750 mV 11: 700 mV
15:13	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0308			Register Name: ADC3_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
12:8	R/W	0x0	ADC3_PGA_GAIN_CTRL ADC3 PGA gain settings: 0x0: 0 dB 0x10: 21 dB 0x1: 6 dB 0x11: 22 dB 0x2: 6 dB 0x12: 23 dB 0x3: 6 dB 0x13: 24 dB 0x4: 9 dB 0x14: 25 dB 0x5: 10 dB 0x15: 26 dB 0x6: 11 dB 0x16: 27 dB 0x7: 12 dB 0x17: 28 dB 0x8: 13 dB 0x18: 29 dB 0x9: 14 dB 0x19: 30 dB 0xA: 15 dB 0x1A: 31 dB 0xB: 16 dB 0x1B: 32 dB 0xC: 17 dB 0x1C: 33 dB 0xD: 18 dB 0x1D: 34 dB 0xE: 19 dB 0x1E: 35 dB 0xF: 20 dB 0x1F: 36 dB
7:6	R/W	0x1	ADC3_IOPAAF ADC3 OP AAF Bias Current Select 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA
5:4	R/W	0x1	ADC3_IOPSDM1 ADC3 OP SDM Bias Current Select 1 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0308			Register Name: ADC3_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
3:2	R/W	0x1	ADC3_IOPSDM2 ADC3 OP SDM Bias Current Select 2 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA
1:0	R/W	0x1	ADC3_IOPMIC ADC3 OP MIC Bias Current Select 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA

5.4.6.105 0x030C ADC4 Analog Control Register (Default Value: 0x000C_0055)

Default:0x030C			Register Name: ADC4_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	ADC4_EN ADC4 Channel Enable 0: Disabled 1: Enabled
30	R/W	0x0	MIC4_PGA_EN MIC4 PGA Enable 0: Disabled 1: Enabled
29	R/W	0x0	ADC4 Dither Control 0: New Dither Off 1: New Dither On
28:26	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Default:0x030C			Register Name: ADC4_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
25:24	R/W	0x0	DSM_DITHER_LVL Dither Level Control (Dither level is positive ralated to the ctrl bits) 00: No Level 01: Min Level 10: Middle Level 11: Max Level
23:20	/	/	/
19:18	R/W	0x3	ADC4_PGA_CTRL_RCM ADC4 PGA Common Mode Input Impedance Control for MICIN 00: 100 kΩ 01: 75 kΩ 10: 50 kΩ 11: 25 kΩ
17:16	R/W	0x0	ADC4_PGA_IN_VCM_CTRL ADC4 PGA Common-Mode Voltage Control 00: 900 mV 01: 800 mV 10: 750 mV 11: 700 mV
15:13	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Default:0x030C			Register Name: ADC4_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
12:8	R/W	0x0	ADC4_PGA_GAIN_CTRL ADC4 PGA gain settings: 0x0: 0 dB 0x10: 21 dB 0x1: 6 dB 0x11: 22 dB 0x2: 6 dB 0x12: 23 dB 0x3: 6 dB 0x13: 24 dB 0x4: 9 dB 0x14: 25 dB 0x5: 10 dB 0x15: 26 dB 0x6: 11 dB 0x16: 27 dB 0x7: 12 dB 0x17: 28 dB 0x8: 13 dB 0x18: 29 dB 0x9: 14 dB 0x19: 30 dB 0xA: 15 dB 0x1A: 31 dB 0xB: 16 dB 0x1B: 32 dB 0xC: 17 dB 0x1C: 33 dB 0xD: 18 dB 0x1D: 34 dB 0xE: 19 dB 0x1E: 35 dB 0xF: 20 dB 0x1F: 36 dB
7:6	R/W	0x1	ADC4_IOPAAF ADC4 OP AAF Bias Current Select 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA
5:4	R/W	0x1	ADC4_IOPSDM1 ADC4 OP SDM Bias Current Select 1 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Default:0x030C			Register Name: ADC4_REG
Bit	Read/Write	Default/Hex	Description
3:2	R/W	0x1	ADC4_IOPSDM2 ADC4 OP SDM Bias Current Select 2 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA
1:0	R/W	0x1	ADC4_IOPMIC ADC4 OP MIC Bias Current Select 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA

5.4.6.106 0x0310 DAC Analog Control Register (Default Value: 0x0015_0000)

Offset: 0x0310			Register Name: DAC_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23	R/W	0x0	CURRENT_TEST_SELECT Internal Current Sink Test Enable (from MICIN1N pin) 0: Normal 1: For Debug
22	R/W	0x0	VAR1SPEEDUP_DOWN_Further_CTRL VRA1Speedup Down Further Control In Adda Analog 0: Digital Signal Interface Pin l_vra1speedup (vra1_speedup_down) Normally Control VRA1 Speedup down 1: Manual Control Finish VRA1 Speedup down, Ignore Digital Signal Interface Pin l_vra1speedup (vra1_speedup_down) Control

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0310			Register Name: DAC_REG
Bit	Read/Write	Default/Hex	Description
21:20	R/W	0x1	IOPVRS VRA2 Buffer OP Bias Current Select 00: 6 uA 01: 7 uA 10: 8 uA 11: 9 uA
19:18	R/W	0x1	ISPKOUTAMPS SPKOUTL/R AMP Bias Current Select 00: 6 uA 01: 7 uA 10: 8 uA 11: 9 uA
17:16	R/W	0x1	IOPDACS OPDAC Bias Current Select 00: 6 uA 01: 7 uA 10: 8 uA 11: 9 uA
15	R/W	0x0	DACL_EN DACL Enable 0: Disabled 1: Enabled
14	R/W	0x0	DACR_EN DACR Enable 0: Disabled 1: Enabled
13	R/W	0x0	SPKOUTL_EN Left Channel SPKOUT Enable 0: Disabled 1: Enabled
12	R/W	0x0	LMUTE DACL to Left Channel SPKOUT Mute Control 0: Mute 1: Not mute

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0310			Register Name: DAC_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
11	R/W	0x0	SPKOUTR_EN Right Channel SPKOUT Enable 0: Disabled 1: Enabled
10	R/W	0x0	RMUTE DACR to Right Channel SPKOUT Mute Control 0: Mute 1: Not mute
9	R/W	0x0	RSWITCH 0: SPKOUT OUTPUT of RAMP_DAC 1: VRA1
8	R/W	0x0	RAMPEN Ramp DAC Enable 0: Disabled 1: Enabled
7	/	/	/
6	R/W	0x0	SPKOUTL_DIFF_EN Left Channel SPKOUT Output Control 0: Single-End 1: Differential
5	R/W	0x0	SPKOUTR_DIFF_EN Right Channel SPKOUT Output Control 0: Single-End 1: Differential
4:0	R/W	0x0	SPKOUT Volume Control, Total 30 level from 0x1F to 0x02 with the volume 0 dB to -43.5 dB, -1.5 dB/step, mute when 00000 & 00001.

5.4.6.107 0x0318 MICBIAS Analog Control Register (Default Value: 0x0000_0030)

Offset: 0x0318			Register Name: MICBIAS_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

5.4.6.108 0x031C Ramp Control Register (Default Value: 0x0000_0000)

Offset: 0x031C			Register Name: RAMP_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:7	/	/	/
6:4	R/W	0x0	RS Ramp Step 000: 3 us 001: 6 us 010: 12 us 011: 24 us 100: 48 us 101: 60 us 110: 96 us 111: 120 us

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x031C			Register Name: RAMP_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
3	R/W	0x0	RMD_EN Ramp Manual Down Enable 0: Disabled 1: Enabled
2	R/W	0x0	RMU_EN Ramp Manual Up Enable 0: Disabled 1: Enabled
1	R/W	0x0	RMC_EN Ramp Manual Control Enable 0: Disabled 1: Enabled
0	R/W	0x0	RD_EN Ramp Digital Enable 0: Disabled 1: Enabled

5.4.6.109 0x0320 BIAS Analog Control Register (Default Value: 0x0000_0080)

Offset: 0x0320			Register Name: BIAS_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R/W	0x80	BIASDATA Bias Current Register Setting Data The register is not controlled by the reset of Audio Codec, only by system bus reset.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

5.4.6.110 0x0330 ADC5 Analog Control Register (Default Value: 0x000C_0055)

Offset: 0x0330			Register Name: ADC5_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	ADC5_EN ADC5 Channel Enable 0: Disabled 1: Enabled
30	R/W	0x0	MIC5_PGA_EN MIC5 PGA Enable 0: Disabled 1: Enabled
29	R/W	0x0	ADC5 Dither Control 0: New Dither Off 1: New Dither On
28:26	/	/	/
25:24	R/W	0x0	DSM_DITHER_LVL Dither Level Control (Dither level is positive ralated to the ctrl bits) 00: No Level 01: Min Level 10: Middle Level 11: Max Level
23:20	/	/	/
19:18	R/W	0x3	ADC5_PGA_CTRL_RCM ADC5 PGA Common Mode Input Impedance Control for MICIN 00: 100 kΩ 01: 75 kΩ 10: 50 kΩ 11: 25 kΩ
17:16	R/W	0x0	ADC5_PGA_IN_VCM_CTRL ADC5 PGA Common-Mode Voltage Control 00: 900 mV 01: 800 mV 10: 750 mV 11: 700 mV
15:13	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0330			Register Name: ADC5_REG
Bit	Read/Write	Default/Hex	Description
12:8	R/W	0x0	ADC5_PGA_GAIN_CTRL ADC5 PGA gain settings: 0x0: 0 dB 0x10: 21 dB 0x1: 6 dB 0x11: 22 dB 0x2: 6 dB 0x12: 23 dB 0x3: 6 dB 0x13: 24 dB 0x4: 9 dB 0x14: 25 dB 0x5: 10 dB 0x15: 26 dB 0x6: 11 dB 0x16: 27 dB 0x7: 12 dB 0x17: 28 dB 0x8: 13 dB 0x18: 29 dB 0x9: 14 dB 0x19: 30 dB 0xA: 15 dB 0x1A: 31 dB 0xB: 16 dB 0x1B: 32 dB 0xC: 17 dB 0x1C: 33 dB 0xD: 18 dB 0x1D: 34 dB 0xE: 19 dB 0x1E: 35 dB 0xF: 20 dB 0x1F: 36 dB
7:6	R/W	0x1	ADC5_IOPAAF ADC5 OP AAF Bias Current Select 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA
5:4	R/W	0x1	ADC5_IOPSDM1 ADC5 OP SDM Bias Current Select 1 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0330			Register Name: ADC5_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
3:2	R/W	0x1	ADC5_IOPSDM2 ADC5 OP SDM Bias Current Select 2 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA
1:0	R/W	0x1	ADC5_IOPMIC ADC5 OP MIC Bias Current Select 00: 1.50*IOPADC 01: 1.75*IOPADC 10: 2.00*IOPADC 11: 2.25*IOPADC IOPADC is defined by ADC1_REG<15:14> from 1 uA to 4 uA

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Contents

6 AIPU 555
6.1 Overview 555
6.2 Block Diagram 555
6.3 Programming Guidelines 556
6.4 Register List 556
6.5 Register Description 557
6.5.1 0x0000 AIPU Control Register (Default Value: 0x0000_0000) 557
6.5.2 0x0004 AIPU Status Register (Default Value: 0x0007_0000) 558
6.5.3 0x0008 AIPU Start PC Register (Default Value: 0x0000_0000) 559
6.5.4 0x000C AIPU Interrupt PC Register (Default Value: 0x0000_0010) 560
6.5.5 0x0010 AIPU IPI Control Register (Default Value: 0x0000_0000) 560
6.5.6 0x0014 AIPU Data Address 0 Register (Default Value: 0x0000_0000) 560
6.5.7 0x0018 AIPU Data Address 1 Register (Default Value: 0x0000_0000) 560
6.5.8 0x003C AIPU Clock Control Register (Default Value: 0x0000_0000) 561
6.5.9 0x0400 AIPU AXI PMU Enable Register (Default Value: 0x0000_0000) 561
6.5.10 0x0404 AIPU AXI PMU Clear Register (Default Value: 0x0000_0000) 561
6.5.11 0x0408 AIPU AXI PMU Period Register (Default Value: 0x0000_0000) 562
6.5.12 0x040C AIPU AXI PMU Read Latency Register (Default Value: 0x0000_0000) 562
6.5.13 0x0410 AIPU AXI PMU Write Latency Register (Default Value: 0x0000_0000) 562
6.5.14 0x0414 AIPU AXI PMU Read Request Register (Default Value: 0x0000_0000) 562
6.5.15 0x0418 AIPU AXI PMU Write Request Register (Default Value: 0x0000_0000) 562
6.5.16 0x041C AIPU AXI PMU Read Bandwidth Register (Default Value: 0x0000_0000) 563
6.5.17 0x0420 AIPU AXI PMU Write Request Register (Default Value: 0x0000_0000) 563

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

6 AIPU

6.1 Overview

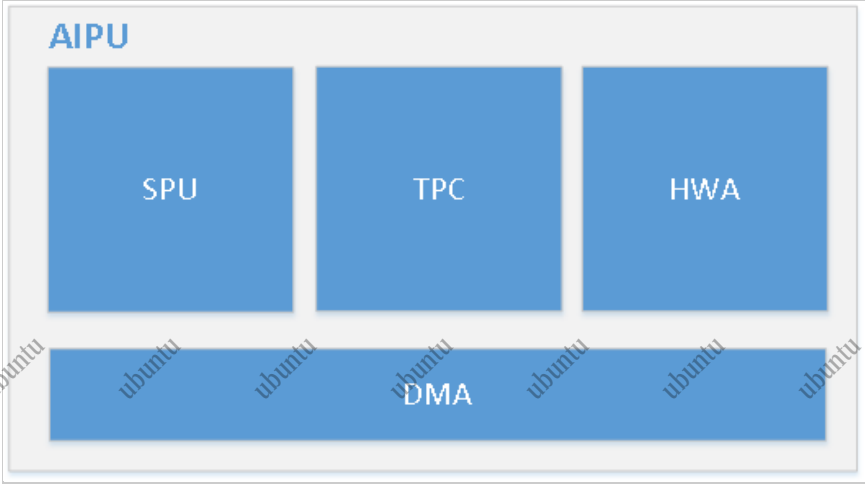
The AI processing unit (AIPU) is a dedicated acceleration engine for deep learning. It is used for key word spotting (KWS), speech recognition, text to speech (TTS), and noise reduction in AI applications. It has a single clock source FCLK operating at the maximum 600 MHz frequency.

NOTE

R329-S3 does not support AIPU.

6.2 Block Diagram

Figure 6-1 AIPU Block Diagram



The AIPU contains the following sub-blocks: scalar processing unit (SPU), tensor processing unit (TPC), neural network hardware accelerator (HWA), and direct memory access (DMA).

Table 6-1 AIPU Sub-blocks

Sub-block	Description
SPU	The SPU is a microprocessor based on the RISC VLIW architecture. It controls the whole AIPU system and provides instructions like scalar, vector, and tensor.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Sub-block	Description
TPC	The TPC is the universal computing unit of AIPU. It supports INT8 and INT16 vector operations like multiply-accumulate (MAC) operation and other flexible operations with unfixed operators.
HWA	The HWA is a dedicated hardware accelerator for accelerating artificial neural networks. It can efficiently execute the CNN and DNN operations like convolution, pooling, linear activation, nonlinear activation, and element-wise operation.
DMA	The DMA provides AXI4 bus interfaces to do the read-write access to external storage. For example, reading the instructions, data, and writing back the intermediate and final result to the external DRAM.

6.3 Programming Guidelines

The SPU controls the whole execution of the neural network edge inference. Follow the steps below to run a trained neural network application on AIPU.

Step 1 Quantify the trained neural network and compile it to a binary file (.bin) with the provided software development kit (SDK).

Step 2 Load the .bin file, data, and parameters to the buffer of DRAM or on-chip SRAM before you run the AIPU.

Step 3 The main CPUX or the CPUS (when in low-power consumption mode) assigns addresses for the corresponding AIPU registers.

Step 4 When enabled, the AIPU automatically runs and writes back the result to the specified address of external storage, and generates the interrupt notification or exception.

For how to program in SPU, refer to *AI Platform Software Programming Guide*.

6.4 Register List

Module Name	Base Address
AIPU	0x03050000

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
AIPU_CONTROL	0x0000	AIPU Control Register
AIPU_STATUS	0x0004	AIPU Status Register
AIPU_START_PC	0x0008	AIPU Start PC Register
AIPU_INTERRUPT_PC	0x000C	AIPU Interrupt PC Register
AIPU_IPI_CONTROL	0x0010	AIPU IPI Control Register
AIPU_DATA_ ADDR0	0x0014	AIPU Data Address 0 Register
AIPU_DATA_ ADDR1	0x0018	AIPU Data Address 1 Register
AIPU_CLOCK_CONTROL	0x003C	AIPU Clock Control Register
AIPU_AXI_PMU_EN	0x0400	AIPU AXI PMU Enable Register
AIPU_AXI_PMU_CLR	0x0404	AIPU AXI PMU Clear Register
AIPU_AXI_PMU_PRD	0x0408	AIPU AXI PMU Period Register
AIPU_AXI_PMU_LAT_RD	0x040C	AIPU AXI PMU Read Latency Register
AIPU_AXI_PMU_LAT_WR	0x0410	AIPU AXI PMU Write Latency Register
AIPU_AXI_PMU_REQ_RD	0x0414	AIPU AXI PMU Read Request Register
AIPU_AXI_PMU_REQ_WR	0x0418	AIPU AXI PMU Write Request Register
AIPU_AXI_PMU_BW_RD	0x041C	AIPU AXI PMU Read Bandwidth Register
AIPU_AXI_PMU_BW_WR	0x0420	AIPU AXI PMU Write Bandwidth Register

6.5 Register Description

6.5.1 0x0000 AIPU Control Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: AIPU_CONTROL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	R/W	/	/
2	R/W	0x0	EIE Interrupt enable when an AIPU exception occurs. 0: Disabled. 1: Enabled.
1	R/W	0x0	DIE Interrupt enable when the AIPU is done. 0: Disabled. 1: Enabled.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: AIPU_CONTROL
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	QEIE Interrupt enable when the AIPU queue is empty. 0: Disabled. 1: Enabled.

6.5.2 0x0004 AIPU Status Register (Default Value: 0x0007_0000)

Offset: 0x0004			Register Name: AIPU_STATUS
Bit	Read/Write	Default/Hex	Description
31:19	/	/	/
18	R	0x1	BIF AIPU bus handshake idle. 0: The bus transaction is ongoing. 1: No bus transaction. The host CPU can hard reset the AIPU if no transaction is ongoing, but the software should ensure that no new transaction is initiated.
17	R	0x1	IF AIPU idle flag. 0: The AIPU is busy 1: The AIPU is idle. Note that the AIPU will flush the cache automatically after reset, so this bit will be 0 and last for a short time. It will become 1 when the AIPU finishes the cache refresh.
16	R	0x1	EF Task queue empty flag. 0: The queue is not empty. 1: The queue is empty.
15:3	/	/	/
2	R/W1C	0x0	EIF AIPU Exception. 0: No interrupt. 1: AIPU exception. It is set by the hardware when the AIPU goes into interrupt. Writing 1 clears it.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0004			Register Name: AIPU_STATUS
Bit	Read/Write	Default/Hex	Description
1	R/W1C	0x0	DIF Interrupt enable when the AIPU is done. 0: Disabled. 1: Enabled. It is set by the hardware when AIPU is done. Writing 1 clears it.
0	R/W1C	0x0	QEIF Task queue empty interrupt flag. 0: No interrupt. 1: Task queue empty. It is set by the hardware when the queue goes from non-idle to idle. Writing 1 clears it.

6.5.3 0x0008 AIPU Start PC Register (Default Value: 0x0000_0000)

Offset: 0x0008			Register Name: AIPU_START_PC
Bit	Read/Write	Default/Hex	Description
31:4	R/W	0x0	SPC High 28 bits of the start PC.
3	R/W	0x0	ICF Instruction cache flush. 0: No operation. 1: Invalidate.
2:1	R/W	0x0	DCF Data cache flush. 00: No operation. 01: Write-back. 10: Write-back and invalidate. 11: Invalidate.
0	R/W	0x0	EN AIPU start enable. 0: No operation. 1: AIPU start, cleared by the hardware when the AIPU accepts the task.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

6.5.4 0x000C AIPU Interrupt PC Register (Default Value: 0x0000_0010)

Offset: 0x000C			Register Name: AIPU_INTERRUPT_PC
Bit	Read/Write	Default/Hex	Description
31:4	R	0x1	IPC Interrupt PC.
3:0	/	/	/

6.5.5 0x0010 AIPU IPI Control Register (Default Value: 0x0000_0000)

Offset: 0x0010			Register Name: AIPU_IPI_CONTROL
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
3:0	R/W	0x0	Mode IPI mode. It is cleared by the hardware when the AIPU accepts IPI. 0 No operation. 1 IPI. Others are reserved.

6.5.6 0x0014 AIPU Data Address 0 Register (Default Value: 0x0000_0000)

Offset: 0x0014			Register Name: AIPU_DATA_ ADDR0
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	Info Read-write by the host CPU and read by the AIPU.

6.5.7 0x0018 AIPU Data Address 1 Register (Default Value: 0x0000_0000)

Offset: 0x0018			Register Name: AIPU_DATA_ ADDR1
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	Info Read-write by the host CPU and read by the AIPU.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

6.5.8 0x003C AIPU Clock Control Register (Default Value: 0x0000_0000)

Offset: 0x003C			Register Name: AIPU_CLOCK_CONTROL
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	Disen Clock gating disable register. 0: Clock gating enabled. 1: Clock gating disabled.

6.5.9 0x0400 AIPU AXI PMU Enable Register (Default Value: 0x0000_0000)

Offset: 0x0400			Register Name: AIPU_AXI_PMU_EN
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	EN PMU Enable register. 0: PMU disabled. 1: PMU Enabled.

6.5.10 0x0404 AIPU AXI PMU Clear Register (Default Value: 0x0000_0000)

Offset: 0x0404			Register Name: AIPU_AXI_PMU_CLR
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	CLR PMU Clear register. 0: No operation. 1: PMU cleared.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

6.5.11 0x0408 AIPU AXI PMU Period Register (Default Value: 0x0000_0000)

Offset: 0x0408			Register Name: AIPU_AXI_PMU_PRD
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	PRD Monitor period based on the AXI cycle. The period is recommended to be calculated by milliseconds.

6.5.12 0x040C AIPU AXI PMU Read Latency Register (Default Value: 0x0000_0000)

Offset: 0x040C			Register Name: AIPU_AXI_PMU_LAT_RD
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	LAT_RD Monitor the total delay of the read channel in a period.

6.5.13 0x0410 AIPU AXI PMU Write Latency Register (Default Value: 0x0000_0000)

Offset: 0x0410			Register Name: AIPU_AXI_PMU_LAT_WR
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	LAT_WR Monitor the total delay of the write channel in a period.

6.5.14 0x0414 AIPU AXI PMU Read Request Register (Default Value: 0x0000_0000)

Offset: 0x0414			Register Name: AIPU_AXI_PMU_REQ_RD
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	REQ_RD Monitor the total command number of reading channels in a period.

6.5.15 0x0418 AIPU AXI PMU Write Request Register (Default Value: 0x0000_0000)

Offset: 0x0418			Register Name: AIPU_AXI_PMU_REQ_WR
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	REQ_WR Monitor the total command number of writing channels in a period.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

6.5.16 0x041C AIPU AXI PMU Read Bandwidth Register (Default Value: 0x0000_0000)

Offset: 0x041C			Register Name: AIPU_AXI_PMU_BW_RD
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	BW_RD Monitor the total data volume (KB) of reading channels in a period.

6.5.17 0x0420 AIPU AXI PMU Write Request Register (Default Value: 0x0000_0000)

Offset: 0x0420			Register Name: AIPU_AXI_PMU_BW_WR
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	BW_WR Monitor the total data volume (KB) of writing channels in a period.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Contents

7 Interfaces 574
7.1 TWI 574
7.1.1 Overview 574
7.1.2 Block Diagram 574
7.1.3 Functional Descriptions 576
7.1.4 Programming Guidelines 582
7.1.5 Register List 585
7.1.6 Register Description 586
7.2 UART 601
7.2.1 Overview 601
7.2.2 Block Diagram 601
7.2.3 Functional Descriptions 602
7.2.4 Programming Guidelines 608
7.2.5 Register List 611
7.2.6 Register Description 613
7.3 SPI 642
7.3.1 Overview 642
7.3.2 Block Diagram 642
7.3.3 Functional Descriptions 644
7.3.4 Programming Guidelines 652
7.3.5 Register List 655
7.3.6 Register Description 656
7.4 SPI_DBI 677
7.4.1 Overview 677
7.4.2 Block Diagram 677
7.4.3 Functional Descriptions 679
7.4.4 Programming Guidelines 699
7.4.5 Register List 707
7.4.6 Register Description 708
7.5 USB2.0 DRD 738

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.5.1 Overview 738
7.5.2 Block Diagram 739
7.5.3 Functional Descriptions 739
7.6 USB1.1 HOST 741
7.6.1 Overview 741
7.6.2 Block Diagram 741
7.6.3 Functional Descriptions 742
7.6.4 Register List 742
7.6.5 Register Description 744
7.7 GPIO 766
7.7.1 Overview 766
7.7.2 Block Diagram 766
7.7.3 Functional Descriptions 767
7.7.4 PB/PC/PF/PG/PH Register List 774
7.7.5 PB/PC/PF/PG/PH Register Description 775
7.7.6 PL/PM/PN Register List 834
7.7.7 PL/PM/PN Register Description 835
7.8 GPADC 886
7.8.1 Overview 886
7.8.2 Block Diagram 886
7.8.3 Functional Descriptions 887
7.8.4 Programming Guidelines 889
7.8.5 Register List 890
7.8.6 Register Description 891
7.9 LRADC 902
7.9.1 Overview 902
7.9.2 Block Diagram 902
7.9.3 Functional Descriptions 903
7.9.4 Programming Guidelines 904
7.9.5 Register List 906
7.9.6 Register Description 906
7.10 PWM 911
7.10.1 Overview 911

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.10.2 Block Diagram 912
7.10.3 Functional Descriptions 912
7.10.4 Programming Guidelines 922
7.10.5 Register List 923
7.10.6 Register Description 925
7.11 LEDC 954
7.11.1 Overview 954
7.11.2 Block Diagram 954
7.11.3 Functional Descriptions 955
7.11.4 Programming Guidelines 959
7.11.5 Register List 964
7.11.6 Register Description 964
7.12 EMAC 975
7.12.1 Overview 975
7.12.2 Block Diagram 975
7.12.3 Functional Descriptions 976
7.12.4 Programming Guidelines 983
7.12.5 Register List 984
7.12.6 Register Description 985
7.13 CIR Receiver 1003
7.13.1 Overview 1003
7.13.2 Block Diagram 1003
7.13.3 Functional Descriptions 1004
7.13.4 Programming Guidelines 1008
7.13.5 Register List 1009
7.13.6 Register Description 1009
7.14 CIR Transmitter 1015
7.14.1 Overview 1015
7.14.2 Block Diagram 1015
7.14.3 Functional Descriptions 1016
7.14.4 Programming Guidelines 1019
7.14.5 Register List 1020
7.14.6 Register Description 1020

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.15 SCR 1028
7.15.1 Overview 1028
7.15.2 Block Diagram 1028
7.15.3 Functional Descriptions 1029
7.15.4 Register List 1031
7.15.5 Register Description 1032

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figures

Figure 7-1 TWI Block Diagram 575
Figure 7-2 Write Timing in 7-bit Standard Addressing Mode 577
Figure 7-3 Read Timing in 7-bit Standard Addressing Mode 577
Figure 7-4 Write Timing in 10-bit Extended Addressing Mode 578
Figure 7-5 Read Timing in 10-bit Extended Addressing Mode 578
Figure 7-6 TWI Driver Write Packet Transmission 579
Figure 7-7 TWI Driver Read Packet Transmission 580
Figure 7-8 TWI Programming State Diagram 582
Figure 7-9 UART Block Diagram 602
Figure 7-10 UART Serial Data Format 603
Figure 7-11 Application Diagram for RTS/CTS Autoflow Control 604
Figure 7-12 RTS/CTS Autoflow Control Data Format 604
Figure 7-13 Application Diagram for IrDA Transceiver 604
Figure 7-14 Serial IrDA Data Format 605
Figure 7-15 Application Diagram for RS-485 Transceiver 605
Figure 7-16 RS-485 Data Format 605
Figure 7-17 SPI Block Diagram 643
Figure 7-18 SPI Application Block Diagram 645
Figure 7-19 SPI Phase 0 Timing Diagram 646
Figure 7-20 SPI Phase 1 Timing Diagram 646
Figure 7-21 SPI 3-Wire Mode 647
Figure 7-22 SPI Dual-Input/Dual-Output Mode 648
Figure 7-23 SPI Dual I/O Mode 648
Figure 7-24 SPI Quad-Input/Quad-Output Mode 649
Figure 7-25 SPI Write/Read Data in CPU Mode 653
Figure 7-26 SPI Write/Read Data in DMA Mode 654
Figure 7-27 SPI_DBI Block Diagram 678
Figure 7-28 SPI Application Block Diagram 681
Figure 7-29 DBI Application Block Diagram 681
Figure 7-30 SPI Phase 0 Timing Diagram 682

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-31 SPI Phase 1 Timing Diagram 682
Figure 7-32 SPI 3-Wire Mode 683
Figure 7-33 SPI Dual-Input/Dual-Output Mode 684
Figure 7-34 SPI Dual I/O Mode 684
Figure 7-35 SPI Quad-Input/Quad-Output Mode 685
Figure 7-36 DBI 3-Line Display Bus Serial Interface Writing Operation Format 687
Figure 7-37 DBI 3-Line Display Bus Serial Interface 8-bit Reading Operation Format 688
Figure 7-38 DBI 3-Line Display Bus Serial Interface 24-bit Reading Operation Format 688
Figure 7-39 DBI 3-Line Display Bus Serial Interface 32-bit Reading Operation Format 689
Figure 7-40 DBI 4-Line Display Bus Serial Interface Writing Operation Format 690
Figure 7-41 DBI 4-Line Display Bus Serial Interface 8-bit Reading Operation Format 690
Figure 7-42 DBI 4-Line Display Bus Serial Interface 24-bit Reading Operation Format 691
Figure 7-43 DBI 4-Line Display Bus Serial Interface 32-bit Reading Operation Format 691
Figure 7-44 RGB111 3-Line Interface Transmit Video Format 692
Figure 7-45 RGB444 3-Line Interface Transmit Video Format 692
Figure 7-46 RGB565 3-Line Interface Transmit Video Format 693
Figure 7-47 RGB666 3-Line Interface Transmit Video Format 693
Figure 7-48 RGB111 4-Line Interface Transmit Video Format 694
Figure 7-49 RGB444 4-Line Interface Transmit Video Format 694
Figure 7-50 RGB565 4-Line Interface Transmit Video Format 695
Figure 7-51 RGB666 4-Line Interface Transmit Video Format 695
Figure 7-52 RGB444 2 Data Lane Interface Transmit Video Format 696
Figure 7-53 RGB565 2 Data Lane Interface Transmit Video Format 697
Figure 7-54 RGB666 2 Data Lane Interface Transmit Video Format 0 698
Figure 7-55 RGB666 2 Data Lane Interface Transmit Video Format 1 (ilitek) 698
Figure 7-56 RGB666 2 Data Lane Interface Transmit Video Format 2 (New vision) 698
Figure 7-57 RGB888 2 Data Lane Interface Transmit Video Format 699
Figure 7-58 SPI Write/Read Data in CPU Mode 700
Figure 7-59 SPI Write/Read Data in DMA Mode 701
Figure 7-60 USB2.0 DRD Controller Block Diagram 739
Figure 7-61 USB2.0 DRD Controller and PHY Connection Diagram 740
Figure 7-62 USB1.1 Host Controller Block Diagram 741
Figure 7-63 USB1.1 Host Controller and PHY Connection Diagram 742

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-64 GPIO Block Diagram 767
Figure 7-65 Pull up/down Logic 772
Figure 7-66 IO Buffer Strength Diagram 773
Figure 7-67 GPADC Block Diagram 887
Figure 7-68 GPADC Clock and Timing Requirement 888
Figure 7-69 GPADC Initial Process 889
Figure 7-70 LRADC Block Diagram 902
Figure 7-71 LRADC Interrupt 904
Figure 7- 72 PWM Block Diagram 912
Figure 7-73 PWM01 Clock Controller Diagram 913
Figure 7-74 PWM01 Output Logic Module Diagram 914
Figure 7-75 PWM0 High Level Active State 915
Figure 7-76 PWM0 Low Level Active State 916
Figure 7-77 Phase of PWM0 High Level Active State 916
Figure 7-78 PWM0 Output Waveform in Pulse Mode and Cycle Mode 917
Figure 7-79 PWM01 Complementary Pair Output 918
Figure 7-80 Dead-time Output Waveform 919
Figure 7-81 Group 0~3 PWM Signal Output 920
Figure 7-82 PWM01 Capture Logic Module Diagram 920
Figure 7-83 PWM0 Channel Capture Timing 921
Figure 7-84 LEDC Block Diagram 954
Figure 7-85 LEDC Package Output Timing Diagram 955
Figure 7-86 LEDC 1-frame Output Timing Diagram 956
Figure 7-87 LEDC Input Data Structure 956
Figure 7-88 LEDC Typical Circuit 956
Figure 7-89 LEDC Data Input Code 957
Figure 7-90 LEDC Data Transfer Mode 957
Figure 7-91 LEDC Normal Configuration Process 960
Figure 7-92 LEDC Timeout Abnormal Processing Flow 962
Figure 7-93 FIFO Overflow Abnormal Processing Flow 963
Figure 7-94 EMAC Block Diagram 976
Figure 7-95 EMAC Typical Application 978
Figure 7-96 EMAC RX/TX Descriptor List 979

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-97 CIR Receiver Block Diagram 1003
Figure 7-98 CIR Receiver Clock 1004
Figure 7-99 CIR Receiver Application Diagram 1004
Figure 7-100 NEC Protocol 1005
Figure 7-101 Logical '0' and Logical '1' of NEC Protocol 1005
Figure 7-102 ATHR Definition 1006
Figure 7-103 ITHR Definition 1006
Figure 7-104 NTHR Definition 1006
Figure 7-105 APAM Definition 1007
Figure 7-106 CIR Receiver Process 1008
Figure 7-107 CIR_TX Block Diagram 1016
Figure 7-108 CIR_TX Clock Description 1017
Figure 7-109 Definitions of Logical "1" and Logical "0" 1018
Figure 7-110 CIR Message Timing Diagram 1018
Figure 7-111 CIR Transmitter Process 1019
Figure 7-112 SCR Block Diagram 1029
Figure 7-113 SCR IO Pad Configuration Diagram 1031

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Tables

Table 7-1 TWI Sub-blocks 575
Table 7-2 TWI External Signals 576
Table 7-3 TWI Clock Sources 576
Table 7-4 UART External Signals 602
Table 7-5 UART Clock Sources 603
Table 7-6 UART Mode Baud and Error Rates 606
Table 7-7 IrDA Mode Baud and Error Rates 607
Table 7-8 RS485 Mode Baud and Error Rates 607
Table 7-9 SPI Sub-blocks 643
Table 7-10 SPI External Signals 644
Table 7-11 SPI Clock Sources 644
Table 7-12 SPI Transmit Format 645
Table 7-13 SPI Old Sample Mode and Run Clock 650
Table 7-14 SPI New Sample Mode 650
Table 7-15 SPI_DBI Sub-blocks 678
Table 7-16 SPI_DBI External Signals 679
Table 7-17 SPI_DBI Clock Sources 680
Table 7-18 SPI Transmit Format 682
Table 7-19 SPI Old Sample Mode and Run Clock 686
Table 7-20 SPI New Sample Mode 686
Table 7-21 USB2.0 DRD External Signals 739
Table 7-23 USB1.1 Host External Signals 742
Table 7-24 Multi-function Port 767
Table 7-25 PB Multiplex Function 768
Table 7-26 PC Multiplex Function 768
Table 7-27 PF Multiplex Function 769
Table 7-28 PG Multiplex Function 769
Table 7-29 PH Multiplex Function 770
Table 7-30 PL Multiplex Function 770
Table 7-31 PM Multiplex Function 770

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Table 7-32 PN Multiplex Function....771
Table 7-33 Port Function....772
Table 7-34 GPADC External Signals....887
Table 7-35 GPADC Clock Sources....887
Table 7-36 LRADC External Signals....903
Table 7-37 LRADC Clock Sources....903
Table 7-38 PWM External Signals....912
Table 7-39 LEDC Sub-blocks....954
Table 7-40 LEDC External Signals....955
Table 7-41 LEDC Clock Sources....955
Table 7-42 EMAC Pin Mapping....976
Table 7-43 EMAC RGMII Pin List....977
Table 7-44 EMAC RMII Pin List....977
Table 7-45 EMAC Clock Characteristics....978
Table 7-46 EMAC_EPHY_CLK_REG0 Configuration Value....983
Table 7-47 CIR Receiver External Signals....1004
Table 7-48 CIR_TX External Signals....1016
Table 7-49 SCR External Signals....1029
Table 7-50 SCR Clock Sources....1029

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7 Interfaces

7.1 TWI

7.1.1 Overview

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

The TWI has the following features:

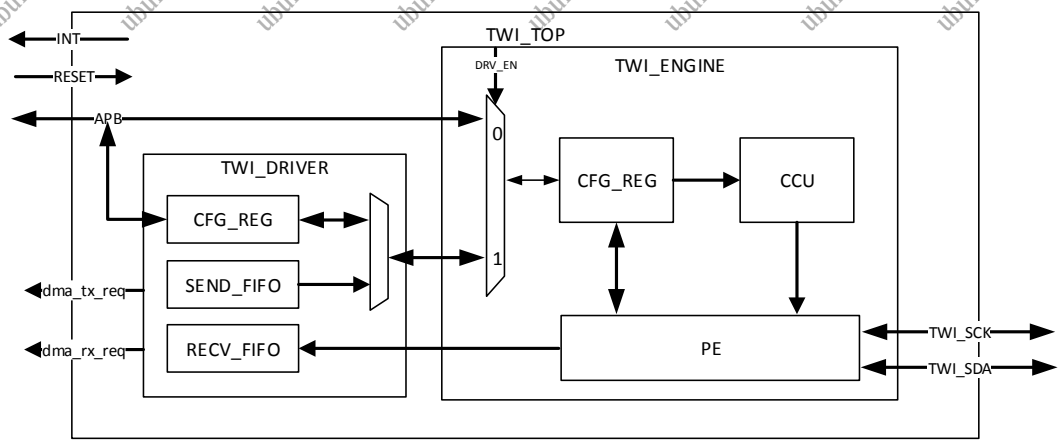
- Compliant with I2C bus standard
- Software programmability for slave or master
- Supports standard mode (up to 100 kbit/s) and fast mode (up to 400 kbit/s)
- Supports 7-bit and 10-bit device addressing modes
- Supports repeated START signal
- Performs arbitration and clock synchronization
- Own address and general call address detection
- Interrupt on address detection
- Byte-by-byte data transfer
- Allows operation from a wide range of input clock frequencies
- The TWI controller includes one TWI engine and one TWI driver. And the TWI driver supports packet transmission and DMA mode when TWI works in master mode

7.1.2 Block Diagram

Figure 7-1 shows the block diagram of TWI.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-1 TWI Block Diagram



TWI contains the following sub-blocks:

Table 7-1 TWI Sub-blocks

Sub-block	Description
RESET	Module reset signal
INT	Module output interrupt signal
CFG_REG	Module configuration register in TWI
PE	Packet encoding/decoding
CCU	Module clock controller unit
SEND_FIFO	The register address bytes and the written data bytes are buffered in SEND_FIFO
RECV_FIFO	The read data bytes are buffered in RECV_FIFO

The controller includes TWI engine and TWI driver. Each time the TWI engine sends a START signal, a STOP signal, or a BYTE data, or a corresponding ACK, the TWI engine will generate an interrupt, and wait for the CPU to process and clear the interrupt before the next START, STOP, or BYTE, ACK transmission can be performed. Therefore, when a device communication is completed, many interrupts will be generated, and the CPU needs to wait for the previous interrupt before it can configure the next one. The TWI driver defines each communication with the device as a packet transmission. The CPU can directly configure the slave address, register address and data transmission for one or more package transmissions without waiting for interruption, then start the TWI driver, and the TWI driver can control the TWI engine to complete a pre-configured communication, and report an interrupt to the CPU after completion.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.1.3 Functional Descriptions

7.1.3.1 External Signals

The TWI controller has 3 TWI modules called TWI0, TWI1, and S-TWI0. The S-TWI0 module is located in the CPUS domain. The following table describes the external signals of the TWI. The TWIn-SCK and TWIn-SDA are bidirectional I/O, when the TWI is configured as a master device, the TWIn-SCK is an output pin; when the TWI is configurable as a slave device, the TWIn-SCK is an input pin. When using TWI, the corresponding PADs are selected as TWI function via section 7.7 "GPIO".

Table 7-2 TWI External Signals

Signal	Description	Type
TWI0-SCK	TWI0 Clock Signal	I/O, OD
TWI0-SDA	TWI0 Serial Data	I/O, OD
TWI1-SCK	TWI1 Clock Signal	I/O, OD
TWI1-SDA	TWI1 Serial Data	I/O, OD
S-TWI0-SCK	S-TWI0 Serial Clock Signal	I/O, OD
S-TWI0-SDA	S-TWI0 Serial Data Signal	I/O, OD

7.1.3.2 Clock Sources

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

For more details on the clock setting, configuration, and gating information, see section 3.3 "CCU".

Table 7-3 TWI Clock Sources

Clock Sources	Description
APB1 Bus	TWI clock source. Refer to CCU for details on APB1.
APBS1 Bus	S-TWI clock source. Refer to PRCM for details on APBS1.

7.1.3.3 Write/Read Timing in Standard and Extended Addressing Mode

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-2 describes the write timing in 7-bit standard addressing mode.

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

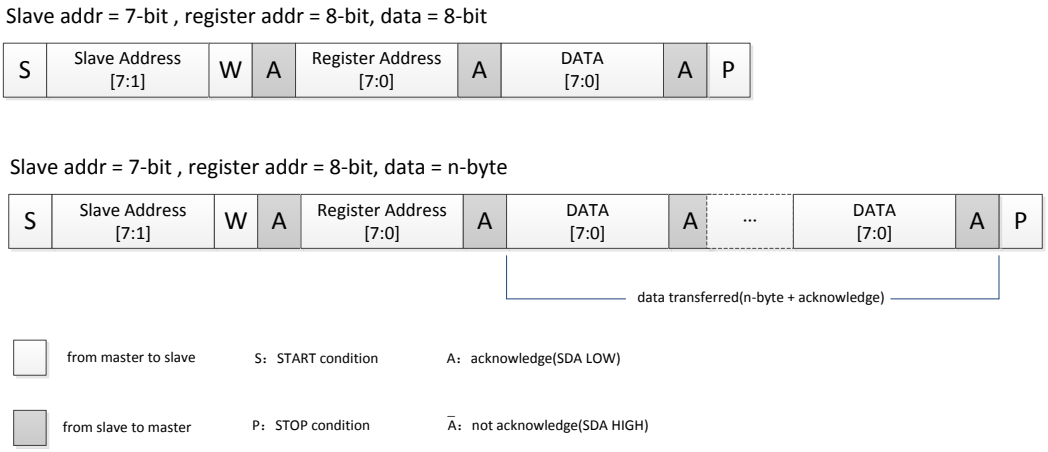


Figure 7-3 describes the read timing in 7-bit standard address mode.

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

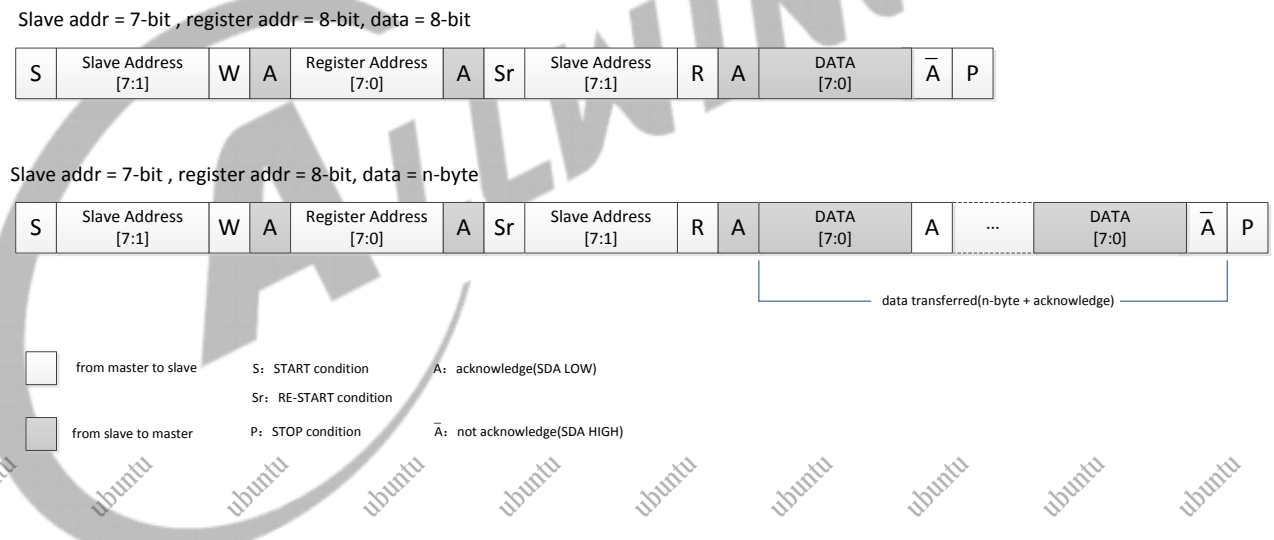


Figure 7-4 describes the write timing in 10-bit extended address mode.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-4 Write Timing in 10-bit Extended Addressing Mode

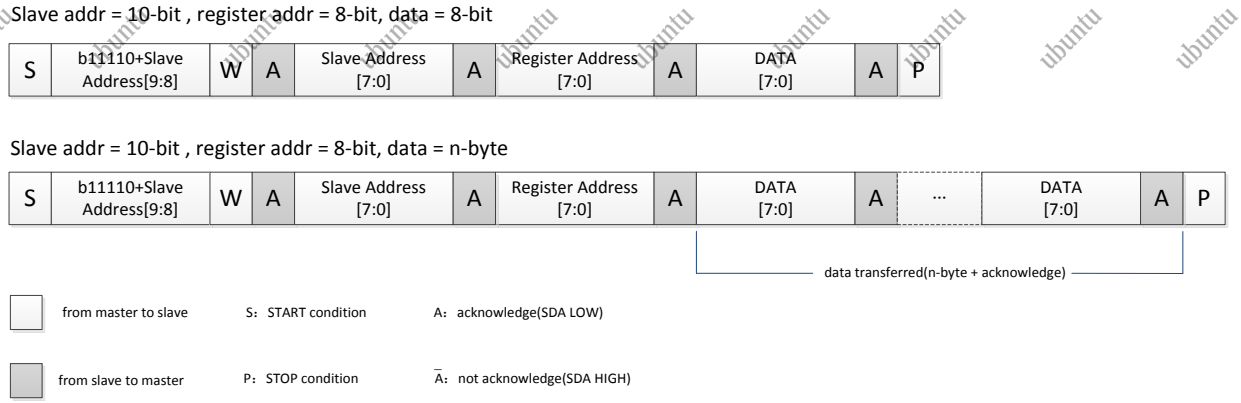
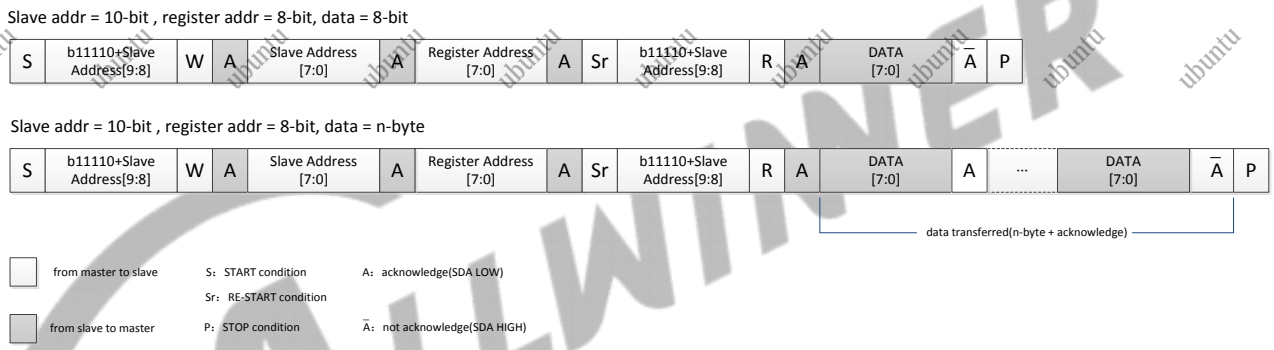


Figure 7-5 describes the read timing in 10-bit extended address mode.

Figure 7-5 Read Timing in 10-bit Extended Addressing Mode

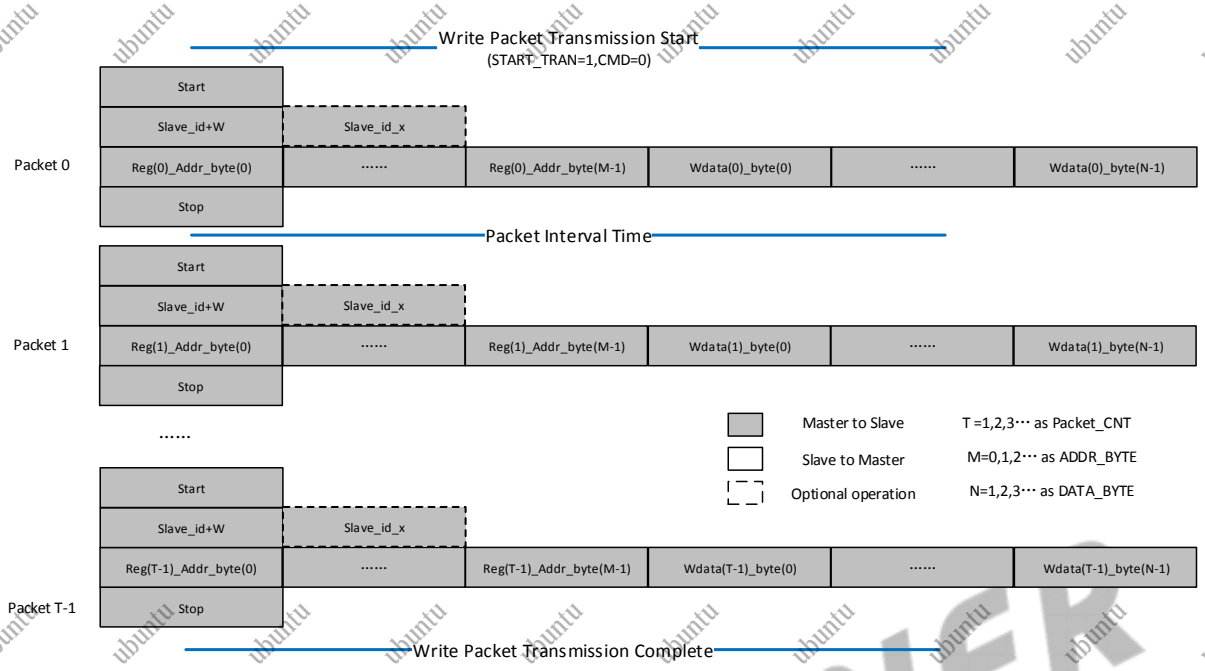


7.1.3.4 Write/Read Packet Transmission of TWI Driver

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

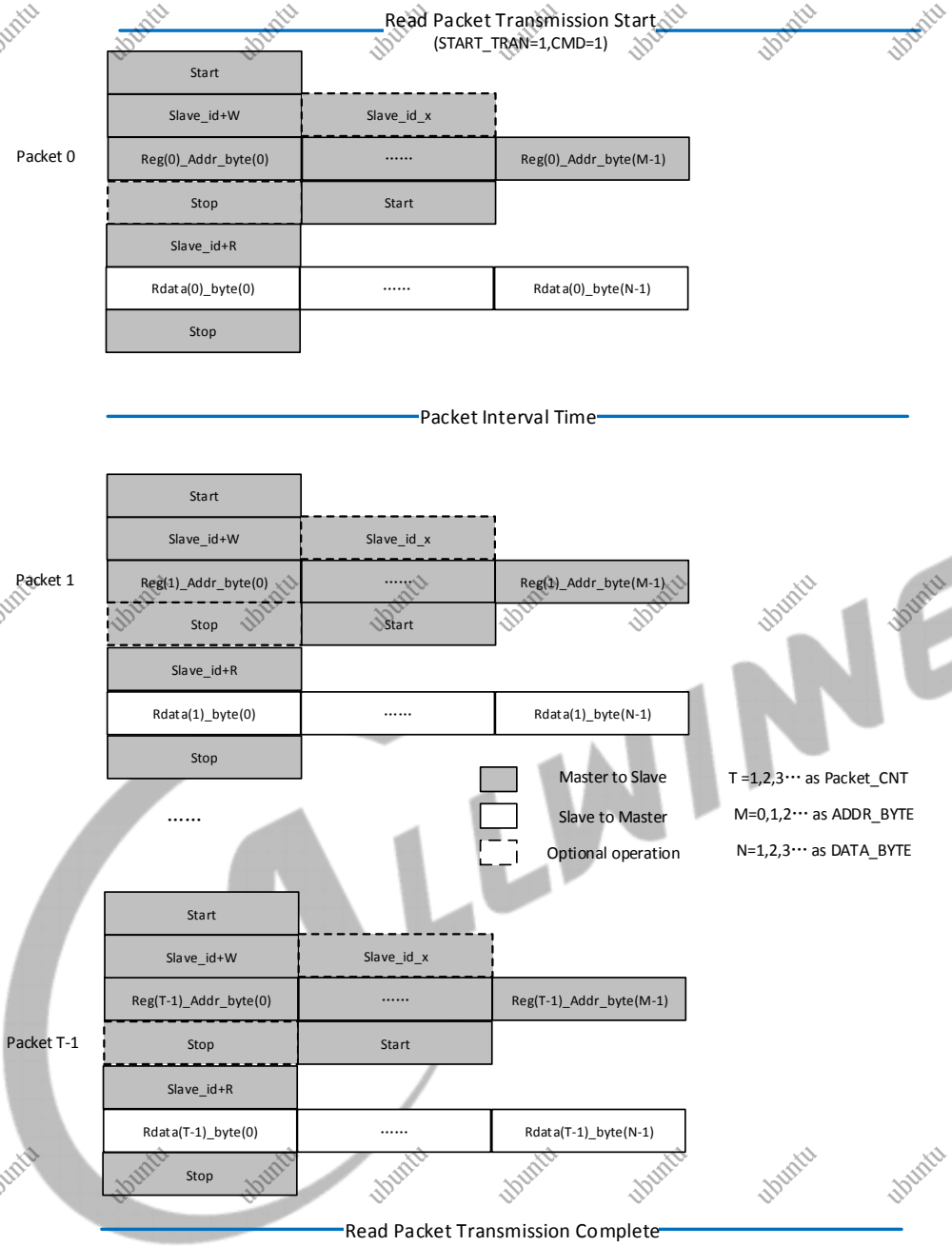
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-6 TWI Driver Write Packet Transmission



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-7 TWI Driver Read Packet Transmission



7.1.3.5 Master and Slave Mode of TWI Engine

In Master mode, the CPU host controls the TWI engine by writing command and data to its registers. The TWI engine transmits an interrupt to CPU when each time a byte transfer is done or a START/STOP command is detected. The CPU host can poll the status register if the interrupt mechanism is not disabled by the CPU host.

When the CPU host wants to start a bus transfer, it initiates a bus START to enter the master mode by setting **TWI_CNTR**[M_STA] to high. The TWI engine will assert the INT line and **TWI_CNTR**[INT_FLAG] to indicate a completion for the START command and each consequent byte transfer. At each interrupt, the CPU host needs

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

to check the current state by the TWI_STAT register. A transfer must conclude with the STOP command by setting TWI_CNTR[M_STP] to high.

In Slave mode, the TWI engine also constantly samples the bus and look for its own slave address during addressing cycles. Once a match is found, it is addressed, and the TWI engine interrupts the CPU host with the corresponding status. Upon request, the CPU host should read the status, read/write the TWI_DATA register, and set the TWI_CNTR register. After each byte transfer, a slave device always stops the operation of the remote master by holding the next low pulse on the SCL line until the CPU host responds to the status of the previous byte transfer or START command.

7.1.3.6 Generation of Repeated Start

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

7.1.3.7 Programming State Diagram

Figure 7-8 shows the TWI programming state diagram. For the value between two states, see the TWI_STAT register in section 7.1.6.5.

M_SEND_S: master sends START signal;

M_SEND_ADDR: master sends slave address;

M_SEND_XADD: master sends slave extended address;

M_SEND_SR: master repeated start;

M_SEND_DATA: master sends data;

M_SEND_P: master sends STOP signal;

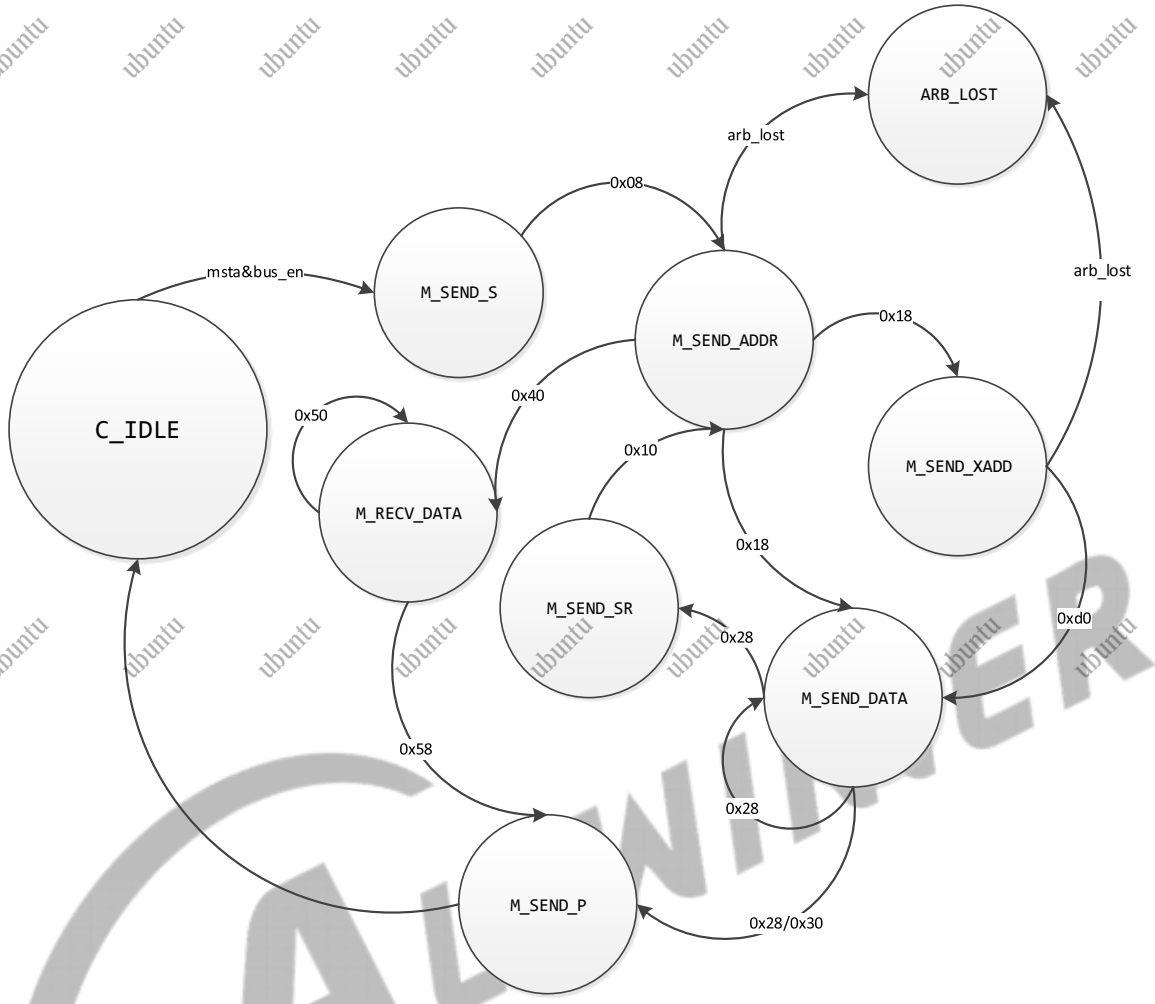
M_RECV_DATA: master receives data;

ARB_LOST: Arbitration lost;

C_IDLE: Idle.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-8 TWI Programming State Diagram



7.1.4 Programming Guidelines

The TWI controller operates in an 8-bit data format. The data on the TWI_SDA line is always 8 bits long. At first, the TWI controller sends a start condition. When in the addressing formats of 7-bit, the TWI sends out an 8-bit message which includes 7 MSB slave address and 1 LSB read/write flag. The least significant of the salve address indicates the direction of transmission. When the TWI works in 10-bit slave address mode, the operation will be divided into two steps, for details on the operation, refer to register description in Section 7.1.6.1 and 7.1.6.2.

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

7.1.4.1 Initialization for TWI Engine

To initialize the TWI engine, perform the following steps:

Step 1 Configure corresponding GPIO multiplex function as TWI mode.

Step 2 For TWIn, set **TWI_BGR_REG**[TWIn_GATING] in CCU module to 0 to close TWIn clock.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 3 For TWIn, set **TWI_BGR_REG**[TWIn_RST] in CCU module to 0, then set to 1 to reset TWIn.

Step 4 For TWIn, set **TWI_BGR_REG**[TWIn_GATING] in CCU module to 1 to open TWIn clock.

Step 5 Configure **TWI_CCR**[CLK_M] and **TWI_CCR**[CLK_N] to get the needed rate (The clock source of TWI is from APB1).

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

7.1.4.2 Writing Data Operation for TWI Engine

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

Step 1 Clear **TWI_EFR** register, and configure **TWI_CNTR**[M_STA] to 1 to transmit the START signal.

Step 2 After the START signal is transmitted, the first interrupt is triggered, then write device ID to **TWI_DATA** (For a 10-bit device ID, firstly write the first byte ID, secondly write the second byte ID in the next interrupt).

Step 3 The Interrupt is triggered again after device ID transmission completes, write device data address to be read to **TWI_DATA** (For a 16-bit address, firstly write the first-byte address, secondly write the second-byte address).

Step 4 Interrupt is triggered after data address transmission completes, write data to be transmitted to **TWI_DATA** (For consecutive write data operation, every byte transmission completion triggers interrupt, during interrupt write the next byte data to **TWI_DATA**).

Step 5 After transmission completes, write **TWI_CNTR**[M_STP] to 1 to transmit the STOP signal and end this write-operation.

7.1.4.3 Reading Data Operation for TWI Engine

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

Step 1 Clear **TWI_EFR** register, and set **TWI_CNTR**[A_ACK] to 1, and configure **TWI_CNTR**[M_STA] to 1 to transmit the START signal.

Step 2 After the START signal is transmitted, the first interrupt is triggered, then write device ID to **TWI_DATA** (For a 10-bit device ID, firstly write the first-byte ID, secondly write the second-byte ID in the next interrupt).

Step 3 The Interrupt is triggered again after device ID transmission completes, write device data address to be read to **TWI_DATA** (For a 16-bit address, firstly write the first-byte address, secondly write the second-byte address).

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 4 The Interrupt is triggered after data address transmission completes, write **TWI_CNTR**[M_STA] to 1 to transmit new START signal, and after interrupt triggers, write device ID to **TWI_DATA** to start read-operation.

Step 5 After device address transmission completes, each receive completion will trigger an interrupt, in turn, read **TWI_DATA** to get data, when receiving the previous interrupt of the last byte data, clear [A_ACK] to stop acknowledge signal of the last byte.

Step 6 Write **TWI_CNTR**[M_STP] to 1 to transmit the STOP signal and end this read-operation.

7.1.4.4 Initialization for TWI Driver

To initialize the TWI driver, perform the following steps:

Step 1 Configure corresponding GPIO multiplex function as TWI mode.

Step 2 For TWIn, set **TWI_BGR_REG**[TWIn_GATING] in CCU module to 0 to close TWIn clock.

Step 3 For TWIn, set **TWI_BGR_REG**[TWIn_RST] in CCU module to 0, then set to 1 to reset TWIn.

Step 4 For TWIn, set **TWI_BGR_REG**[TWIn_GATING] in CCU module to 1 to open TWIn clock.

Step 5 Set **TWI_DRV_CTRL**[TWI_DRV_EN] to 1 to enable the TWI driver.

Step 6 Configure **TWI_DRV_BUS_CTRL**[CLK_M] and **TWI_DRV_BUS_CTRL**[CLK_N] to get the needed rate (The clock source of TWI is from APB1).

Step 7 Set **TWI_DRV_CTRL**[RESTART_MODE] to 0 and [READ_TRAN_MODE] to 1, set **TWI_DRV_INT_CTRL**[TRAN_COM_INT_EN] to 1.

Step 8 When using DMA for data transmission, set **TWI_DRV_DMA_CFG**[DMA_RX_EN] and **TWI_DRV_DMA_CFG**[DMA_TX_EN] to 1, and configure **TWI_DRV_DMA_CFG**[RX_TRIG] and **TWI_DRV_DMA_CFG**[TX_TRIG] to set the thresholds of RXFIFO and TXFIFO.

7.1.4.5 Writing Packet Transmission for TWI Driver

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

Step 1 Configure **TWI_DRV_SLV**[SLV_ID] to set the device ID, and configure **TWI_DRV_SLV**[CMD] to 0 to set the write operation.

Step 2 Configure **TWI_DRV_FMT**[ADDR_BYTE] according to the address width of the device register, and **TWI_DRV_FMT**[DATA_BYTE] according to the written data count in a packet.

Step 3 Configure **TWI_DRV_CFG**[PACKET_CNT] to set the written packet number.

Step 4 Configure DMA channel, including TWI TXFIFO, device register address, and the written data.

Step 5 Set [START_TRAN] to 1 to start TWI Driver transmission.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 6 When TWI driver transmission completes, the interrupt is triggered, it indicates that the write packet transmission ends.

7.1.4.6 Reading Packet Transmission for TWI Driver

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

Step 1 Configure **TWI_DRV_SLV**[SLV_ID] to set the device ID, and configure **TWI_DRV_SLV**[CMD] to 1 to set the read operation.

Step 2 Configure **TWI_DRV_FMT**[ADDR_BYTE] according to the address width of the device register, and **TWI_DRV_FMT**[DATA_BYTE] according to the read data count in a packet.

Step 3 Configure **TWI_DRV_CFG**[PACKET_CNT] to set the read packet number.

Step 4 Configure DMA channel, including TWI TXFIFO, TWI RXFIFO, device register address and the read data.

Step 5 Set [START_TRAN] to 1 to start TWI Driver transmission.

Step 6 When TWI driver transmission completes, the interrupt is triggered, it indicates that the read packet transmission ends.

7.1.5 Register List

Module Name	Base Address
TWI0	0x02502000
TWI1	0x02502400
R_TWI0	0x07081400

Register Name	Offset	Description
TWI_ADDR	0x0000	TWI Slave Address Register
TWI_XADDR	0x0004	TWI Extended Slave Address Register
TWI_DATA	0x0008	TWI Data Byte Register
TWI_CNTR	0x000C	TWI Control Register
TWI_STAT	0x0010	TWI Status Register
TWI_CCR	0x0014	TWI Clock Control Register
TWI_SRST	0x0018	TWI Software Reset Register
TWI_EFR	0x001C	TWI Enhance Feature Register
TWI_LCR	0x0020	TWI Line Control Register
TWI_DRV_CTRL	0x0200	TWI_DRV Control Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
TWI_DRV_CFG	0x0204	TWI_DRV Transmission Configuration Register
TWI_DRV_SLV	0x0208	TWI_DRV Slave ID Register
TWI_DRV_FMT	0x020C	TWI_DRV Packet Format Register
TWI_DRV_BUS_CTRL	0x0210	TWI_DRV Bus Control Register
TWI_DRV_INT_CTRL	0x0214	TWI_DRV Interrupt Control Register
TWI_DRV_DMA_CFG	0x0218	TWI_DRV DMA Configure Register
TWI_DRV_FIFO_CON	0x021C	TWI_DRV FIFO Content Register
TWI_DRV_SEND_FIFO_ACC	0x0300	TWI_DRV Send Data FIFO Access Register
TWI_DRV_RECV_FIFO_ACC	0x0304	TWI_DRV Receive Data FIFO Access Register

7.1.6 Register Description

7.1.6.1 0x0000 TWI Slave Address Register (Default Value: 0x0000_0000)

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

NOTE

For 7-bit addressing:

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

For 10-bit addressing:

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

7.1.6.2 0x0004 TWI Extend Address Register (Default Value: 0x0000_0000)

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

7.1.6.3 0x0008 TWI Data Register (Default Value: 0x0000_0000)

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

7.1.6.4 0x000C TWI Control Register (Default Value: 0x0000_0000)

Offset: 0x000C			Register Name: TWI_CNTR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R/W	0x0	INT_EN Interrupt Enable 0: The interrupt line always low 1: The interrupt line will go high when INT_FLAG is set.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x000C			Register Name: TWI_CNTR
Bit	**Read/Write**	**Default/Hex**	**Description**
6	R/W	0x0	BUS_EN TWI Bus Enable 0: The TWI bus SDA/SCL is ignored and the TWI controller will not respond to any address on the bus. 1: The TWI will respond to call to its slave address – and to the general call address if the GCE bit in the ADDR register is set. **Note: In master operation mode, this bit should be set to '1'.**
5	R/WAC	0x0	M_STA Master Mode Start When the M_STA is set to '1', the TWI controller enters master mode and will transmit a START condition on the bus when the bus is free. If the M_STA bit is set to '1' when the TWI controller is already in master mode and one or more bytes have been transmitted, then a repeated START condition will be sent. If the M_STA bit is set to '1' when the TWI is accessed in slave mode, the TWI will complete the data transfer in slave mode then enter master mode when the bus has been released. The M_STA bit is cleared automatically after a START condition has been sent. Writing a '0' to this bit has no effect.
4	R/W1C	0x0	M_STP Master Mode Stop If the M_STP is set to '1' in master mode, a STOP condition is transmitted on the TWI bus. If the M_STP bit is set to '1' in slave mode, the TWI will indicate if a STOP condition has been received, but no STOP condition will be transmitted on the TWI bus. If both M_STA and M_STP bits are set, the TWI will first transmit the STOP condition (if in master mode), then transmit the START condition. The M_STP bit is cleared automatically. Writing a '0' to this bit has no effect.
3	R/W1C	0x0	INT_FLAG Interrupt Flag The INT_FLAG is automatically set to '1' when any of the 28 (out of the possible 29) states is entered (see 'STAT Register' below). The state that does not set INT_FLAG is state F8h. If the INT_EN bit is set, the interrupt line goes high when INT_FLAG is set to '1'. If the TWI is operating in slave mode, the data transfer is suspended when INT_FLAG is set and the low period of the TWI bus clock line (SCL) is stretched until '1' is written to INT_FLAG. The TWI clock line is then released and the interrupt line goes low.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x000C			Register Name: TWI_CNTR
Bit	Read/Write	Default/Hex	Description
2	R/W	0x0	A_ACK Assert Acknowledge When A_ACK is set to '1', an Acknowledge (low level on SDA) will be sent during the acknowledge clock pulse on the TWI bus if: (1). Either the whole of a matching 7-bit slave address or the first or the second byte of a matching 10-bit slave address has been received. (2). The general call address has been received and the GCE bit in the ADDR register is set to '1'. (3). A data byte has been received in master or slave mode. When A_ACK is '0', a Not Acknowledge (high level on SDA) will be sent when a data byte is received in master or slave mode. If A_ACK is cleared to '0' in slave transmitter mode, the byte in the DATA register is assumed to be the 'last byte'. After this byte has been transmitted, the TWI will enter state C8h then return to the idle state (status code F8h) when INT_FLAG is cleared. The TWI will not respond as a slave unless A_ACK is set.
1	/	/	/
0	R/W	0x0	CLK_COUNT_MODE 0: scl clock high period count on oscl 1: scl clock high period count on iscl

7.1.6.5 0x0010 TWI Status Register (Default Value: 0x0000_00F8)

Offset: 0x0010			Register Name: TWI_STAT
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R	0xF8	STA Status Information Byte Code Status 0x00: Bus error 0x08: START condition transmitted 0x10: Repeated START condition transmitted 0x18: Address + Write bit transmitted, ACK received 0x20: Address + Write bit transmitted, ACK not received 0x28: Data byte transmitted in master mode, ACK received

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0010			Register Name: TWI_STAT
Bit	**Read/Write**	**Default/Hex**	**Description**
			0x30: Data byte transmitted in master mode, ACK not received 0x38: Arbitration lost in address or data byte 0x40: Address + Read bit transmitted, ACK received 0x48: Address + Read bit transmitted, ACK not received 0x50: Data byte received in master mode, ACK transmitted 0x58: Data byte received in master mode, not ACK transmitted 0x60: Slave address + Write bit received, ACK transmitted 0x68: Arbitration lost in the address as master, slave address + Write bit received, ACK transmitted 0x70: General Call address received, ACK transmitted 0x78: Arbitration lost in the address as master, General Call address received, ACK transmitted 0x80: Data byte received after slave address received, ACK transmitted 0x88: Data byte received after slave address received, not ACK transmitted 0x90: Data byte received after General Call received, ACK transmitted 0x98: Data byte received after General Call received, not ACK transmitted 0xA0: STOP or repeated START condition received in slave mode 0xA8: Slave address + Read bit received, ACK transmitted 0xB0: Arbitration lost in the address as master, slave address + Read bit received, ACK transmitted 0xB8: Data byte transmitted in slave mode, ACK received 0xC0: Data byte transmitted in slave mode, ACK not received 0xC8: The Last byte transmitted in slave mode, ACK received 0xD0: Second Address byte + Write bit transmitted, ACK received 0xD8: Second Address byte + Write bit transmitted, ACK not received 0xF8: No relevant status information, INT_FLAG=0 Others: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.1.6.6 0x0014 TWI Clock Register (Default Value: 0x0000_0080)

Offset: 0x0014			Register Name: TWI_CCR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R/W	0x1	CLK_DUTY Setting duty cycle of Clock as Master 0: 50% 1: 40%
6:3	R/W	0x0	CLK_M The TWI SCL output frequency, in master mode, is F1/10: F1 = F0/(CLK_M + 1) Fscl = F1/10 = Fin/(2^CLK_N * (CLK_M + 1)*10) Specially, Fscl = F1/11 when CLK_M=0 and CLK_DUTY=40% due to the delay of SCL sample debounce.
2:0	R/W	0x0	CLK_N The TWI bus is sampled by the TWI at the frequency defined by F0: Fsamp = F0 = Fin/2^CLK_N The TWI SCL output frequency, in master mode, is F1/10: F1 = F0/(CLK_M + 1) Fscl = F1/10 = Fin/(2^CLK_N * (CLK_M + 1)*10) Specially, Fscl = F1/11 when CLK_M=0 and CLK_DUTY=40% due to the delay of SCL sample debounce. For Example: Fin = 48 MHz (APB clock input) For 400 kHz full speed 2-wire, CLK_N = 2, CLK_M = 2 F0 = 48 MHz/2^2 = 12 MHz, F1 = F0/(10*(2+1)) = 0.4 MHz For 100 kHz standard speed 2-wire, CLK_N = 2, CLK_M = 11 F0 = 48 MHz/2^2 = 12 MHz, F1 = F0/(10*(11+1)) = 0.1 MHz

7.1.6.7 0x0018 TWI Soft Reset Register (Default Value: 0x0000_0000)

Offset: 0x0018			Register Name: TWI_SRST
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.1.6.8 0x001C TWI Enhance Feature Register (Default Value: 0x0000_0000)

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

7.1.6.9 0x0020 TWI Line Control Register (Default Value: 0x0000_003A)

Offset: 0x0020			Register Name: TWI_LCR
Bit	Read/Write	Default/Hex	Description
31:6	/	/	/
5	R	0x1	SCL_STATE Current State of TWI_SCL 0: Low 1: High
4	R	0x1	SDA_STATE Current State of TWI_SDA 0: Low 1: High

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0020			Register Name: TWI_LCR
Bit	**Read/Write**	**Default/Hex**	**Description**
3	R/W	0x1	SCL_CTL TWI_SCL Line State Control Bit When the line control mode is enabled (bit[2] is set), this bit decides the output level of TWI_SCL. 0: Output low level 1: Output high level
2	R/W	0x0	SCL_CTL_EN TWI_SCL Line State Control Enable When this bit is set, the state of TWI_SCL is controlled by the value of bit[3]. 0: Disable TWI_SCL line control mode 1: Enable TWI_SCL line control mode
1	R/W	0x1	SDA_CTL TWI_SDA Line State Control Bit When the line control mode is enabled (bit[0] is set), this bit decides the output level of TWI_SDA. 0: Output low level 1: Output high level
0	R/W	0x0	SDA_CTL_EN TWI_SDA Line State Control Enable When this bit is set, the state of TWI_SDA is controlled by the value of bit[1]. 0: Disable TWI_SDA line control mode 1: Enable TWI_SDA line control mode

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.1.6.10 0x0200 TWI_DRV Control Register (Default Value: 0x00F8_1000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.1.6.11 0x0204 TWI_DRV Transmission Configuration Register (Default Value: 0x0000_0001)

Offset: 0x0204			Register Name: TWI_DRV_CFG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	R/W	0x0	PKT_INTERVAL Define the interval between each packet for PKT_INTERVAL F_{SCL} cycles.
15:0	R/W	0x1	PACKET_CNT The FIFO data is transmitted as PACKET_CNT packets in current format.

7.1.6.12 0x0208 TWI_DRV Slave ID Register (Default Value: 0x0000_0000)

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

7.1.6.13 0x020C TWI_DRV Packet Format Register (Default Value: 0x0001_0001)

Offset: 0x020C			Register Name: TWI_DRV_FMT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.1.6.14 0x0210 TWI_DRV Bus Control Register (Default Value: 0x0000_80C0)

Offset: 0x0210			Register Name: TWI_DRV_BUS_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	W	0x0	CLK_COUNT_MODE Clock count mode 0: SCL clock high period count on oscl 1: SCL clock high period count on iscl
15	R/W	0x1	CLK_DUTY Setting duty cycle of clock as Master 0: 50% 1: 40%
14:12	R/W	0x0	CLK_N TWI_DRV bus sampling clock F0=24MHz/2^CLK_N
11:8	R/W	0x0	CLK_M TWI_DRV output SCL frequency is F_{SCL}=F1/10=(F0/(CLK_M+1))/10 Specially, Foscl = F1/11 when CLK_M=0 and CLK_DUTY=40% due to the delay of SCL sample debounce.
7	R	0x1	SCL_STA SCL current status
6	R	0x1	SDA_STA SDA current status
5:4	/	/	/
3	R/W	0x0	SCL_MOV SCL manual output value

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0210			Register Name: TWI_DRV_BUS_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
2	R/W	0x0	SDA_MOV SDA manual output value
1	R/W	0x0	SCL_MOE SCL manual output enable
0	R/W	0x0	SDA_MOE SDA manual output enable

7.1.6.15 0x0214 TWI_DRV Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0214			Register Name: TWI_DRV_INT_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
19	R/W	0x0	RX_REQ_INT_EN If set, an interrupt is sent when RX_REQ_PD sets.
18	R/W	0x0	TX_REQ_INT_EN If set, an interrupt is sent when TX_REQ_PD sets.
17	R/W	0x0	TRAN_ERR_INT_EN If set, an interrupt is sent when TRAN_ERR_PD sets.
16	R/W	0x0	TRAN_COM_INT_EN If set, an interrupt is sent when TRAN_COM_PD sets.
15:4	/	/	/
3	R/W1C	0x0	RX_REQ_PD Set when the data byte number in RECV_FIFO reaches RX_TRIG.
2	R/W1C	0x0	TX_REQ_PD Set when there is no less than DMA_TX_TRIG empty byte number in SEND_FIFO.
1	R/W1C	0x0	TRAN_ERR_PD Packet transmission failure pending
0	R/W1C	0x0	TRAN_COM_PD Packet transmission completion pending

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.1.6.16 0x0218 TWI_DRV DMA Configure Register (Default Value: 0x0010_0010)

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

7.1.6.17 0x021C TWI_DRV FIFO Content Register (Default Value: 0x0000_0000)

Offset: 0x021C			Register Name: TWI_DRV_FIFO_CON
Bit	**Read/Write**	**Default/Hex**	**Description**
31:23	/	/	/
22	R/WAC	0x0	RECV_FIFO_CLEAR Set this bit to clear RECV_FIFO pointer, and this bit is cleared automatically.
21:16	R	0x0	RECV_FIFO_CONTENT The number of data in RECV_FIFO
15:7	/	/	/
6	R/WAC	0x0	SEND_FIFO_CLEAR Set this bit to clear SEND_FIFO pointer, and this bit is cleared automatically.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x021C			Register Name: TWI_DRV_FIFO_CON
Bit	Read/Write	Default/Hex	Description
5:0	R	0x0	SEND_FIFO_CONTENT The number of data in SEND_FIFO

7.1.6.18 0x0300 TWI_DRV Send Data FIFO Access Register (Default Value: 0x0000_0000)

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

7.1.6.19 0x0304 TWI_DRV Receive Data FIFO Access Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.2 UART

7.2.1 Overview

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

The UART has the following features:

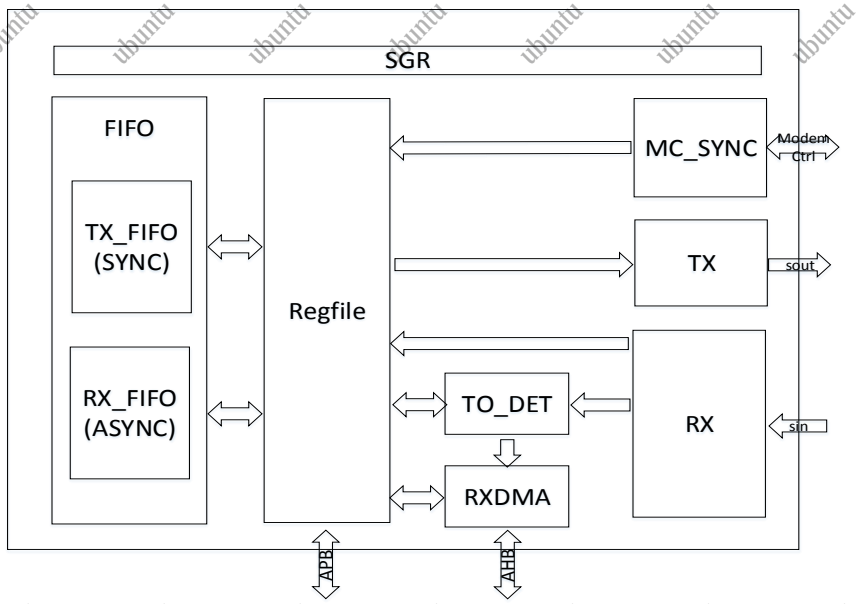
- Compatible with industry-standard 16450/16550 UARTs
- Supports IrDA-compatible slow infrared (SIR) format
- Supports RS-485 compatible 9-bit data format
- Two separate FIFOs: one is RX FIFO, and the other is TX FIFO
  - Each of them is 64 bytes (For UART0 and S-UART0)
  - Each of them is 256 bytes (For UART1, UART2 and UART3)
- The working reference clock is from the APB bus clock
  - Speed up to 4 Mbit/s with 64 MHz APB clock
  - Speed up to 1.5 Mbit/s with 24 MHz APB clock
- 5 to 8 data bits for RS-232 characters, or 9 bits RS-485 format
- 1, 1.5 or 2 stop bits
- Programmable parity (even, odd, or no parity)
- Supports TX/RX DMA slave controller interface
- Supports software/hardware flow control
- Supports RX DMA Master interface (Only for UART1)
- Supports auto-flow by using CTS & RTS (Only for UART1/2/3)

7.2.2 Block Diagram

Figure 7-9 shows a block diagram of the UART.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-9 UART Block Diagram



7.2.3 Functional Descriptions

7.2.3.1 External Signals

The following table describes the external signals of UART.

Table 7-4 UART External Signals

Signal	Description	Type
UART0-TX	UART0 Data Transmit	O
UART0-RX	UART0 Data Receive	I
UART1-TX	UART1 Data Transmit	O
UART1-RX	UART1 Data Receive	I
UART1-CTS	UART1 Data Clear to Send	I
UART1-RTS	UART1 Data Request to Send	O
UART2-TX	UART2 Data Transmit	O
UART2-RX	UART2 Data Receive	I
UART2-CTS	UART2 Data Clear to Send	I
UART2-RTS	UART2 Data Request to Send	O
UART3-TX	UART3 Data Transmit	O
UART3-RX	UART3 Data Receive	I
UART3-CTS	UART3 Data Clear to Send	I

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Signal	Description	Type
UART3-RTS	UART3 Data Request to Send	O
S-UART0-TX	S-UART0 Data Transmit	O
S-UART0-RX	S-UART0 Data Receive	I

7.2.3.2 Clock Sources

The following table describes the clock sources of UART.

Table 7-5 UART Clock Sources

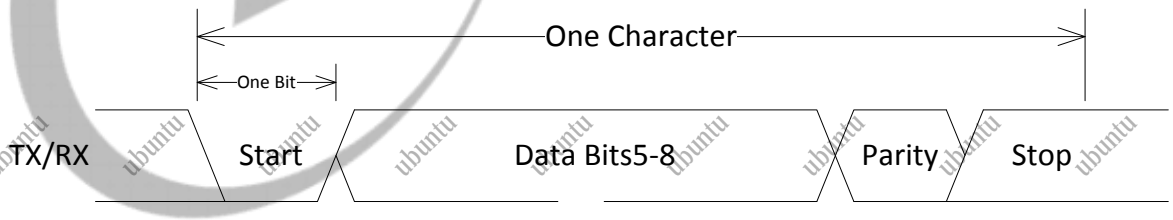
Clock Sources	Description
APB1 Bus	UART clock source. Refer to CCU for details on APB1.
APBS1 Bus	S-UART clock source. Refer to PRCM for details on APBS1.

7.2.3.3 Typical Applications and Timing Diagram

UART Serial Data Format

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

Figure 7-10 UART Serial Data Format



Using UART for RTS/CTS Autoflow Control

Figure 7-11 shows the typical application diagram for RTS/CTS autoflow control. Figure 7-12 shows the data format of the RTS/CTS autoflow control.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-11 Application Diagram for RTS/CTS Autoflow Control

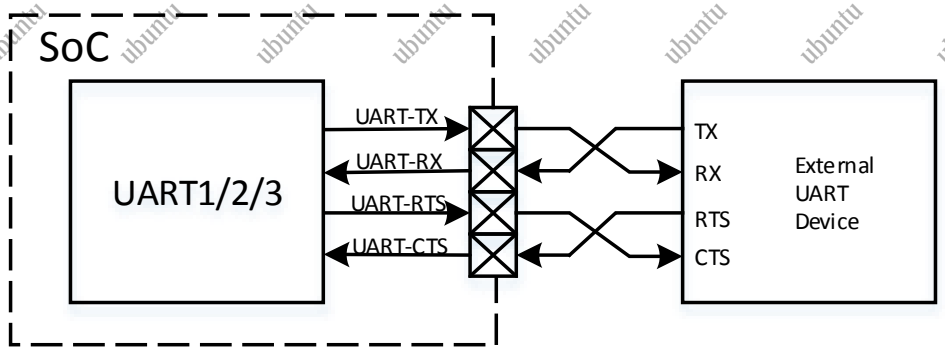
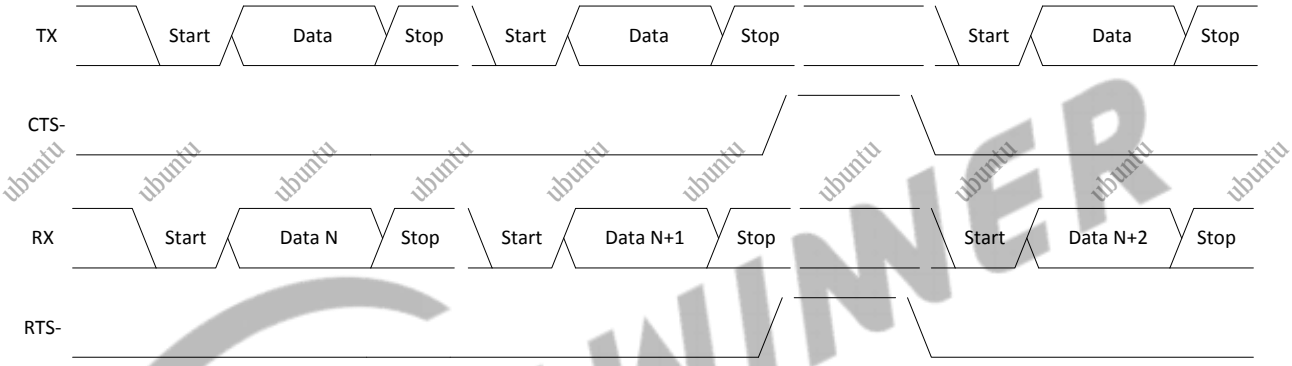


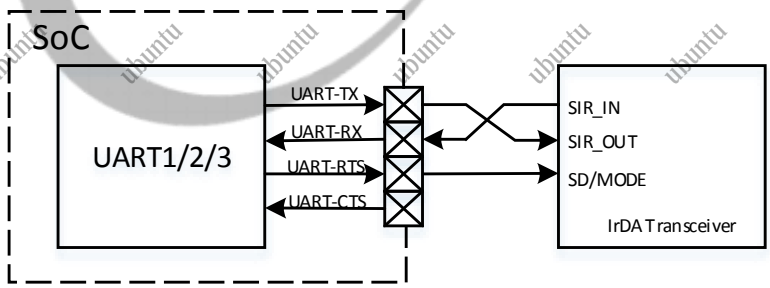
Figure 7-12 RTS/CTS Autoflow Control Data Format



Using UART for Serial IrDA

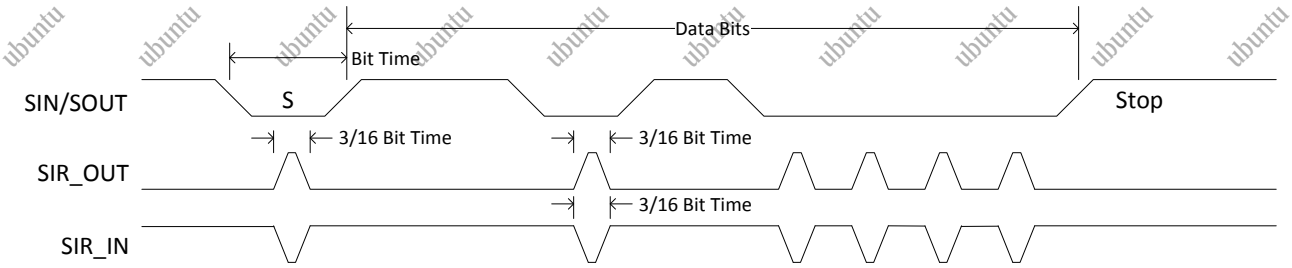
Figure 7-13 shows the application diagram for the IrDA transceiver. Figure 7-14 shows the data format of the serial IrDA.

Figure 7-13 Application Diagram for IrDA Transceiver



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-14 Serial IrDA Data Format



Using UART for RS-485

Figure 7-15 shows the application diagram for the RS-485 transceiver. Figure 7-16 shows the data format of the RS-485.

Figure 7-15 Application Diagram for RS-485 Transceiver

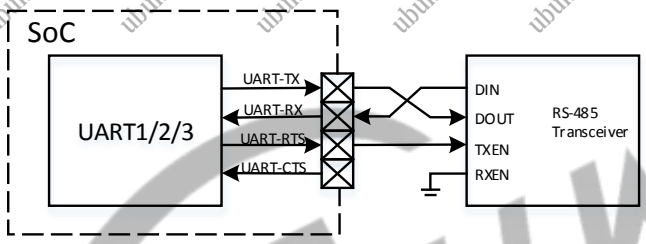
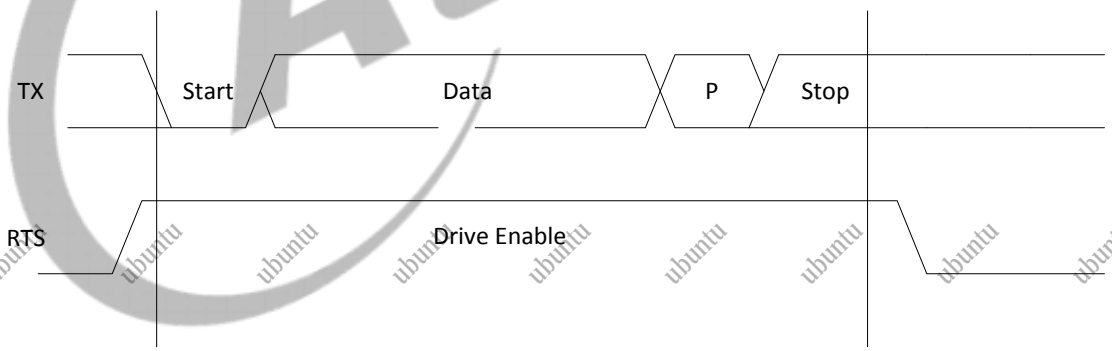


Figure 7-16 RS-485 Data Format



7.2.3.4 UART Operating Mode

Data Frame Format

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

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

Baud and Error Rates

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

The SCLK is usually APB1 and can be set in section 3.3 "CCU".

The divisor is frequency divider of UART. The frequency divider has 16-bit, the low 8-bit is in the **UART_DLL** register, the high 8-bit is in the **UART_DLH** register.

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

Table 7-6 UART Mode Baud and Error Rates

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Clock source	Divisor	Baud rate	Over sampling	Error (%)
48000000	3	1000000	16	0
24000000	1	1500000	16	0
48000000	1	3000000	16	0
64000000	1	4000000	16	0

Table 7-7 IrDA Mode Baud and Error Rates

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

Table 7-8 RS485 Mode Baud and Error Rates

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

DLAB Definition

The DLAB control bit (UART_LCR[7]) is the access control bit of the divisor Latch register.

If DLAB is 0, then the 0x00 offset address is the UART_RBR/UART_THR (RX/TX FIFO) register, and the 0x04 offset address is the UART_IER register.

If DLAB is 1, then the 0x00 offset address is the UART_DLL register, and the 0x04 offset address is the UART_DLH register.

When the UART initials, the divisor needs to be set. That is, writing 1 to DLAB can access the UART_DLL and UART_DLH register, after finished the configuration, writing 0 to DLAB can access the UART_RBR/UART_THR register.

CHCFG_AT_BUSY Definition

The function of the CHCFG_AT_BUSY (UART_HALT[1]) and CHANGE_UPDATE (UART_HALT[2]) are as follows.

CHCFG_AT_BUSY: Enable the bit, the software can also set the UART controller when UART is busy, such as the UART_LCR, UART_DLH, UART_DLL register.

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

Setting divisor performs the following steps:

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

Step 2 Write 1 to DLAB (UART_LCR[7]) and set the UART_DLH and UART_DLL registers.

Step 3 Write 1 to CHANGE_UPDATE to update the configuration. The bit is cleared to 0 automatically after completing the update.

UART Busy Flag

The UART_USR[0] is a busy flag of the UART controller.

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

7.2.4 Programming Guidelines

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.2.4.1 Initialization

Step 1 System Initialization

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

Step 2 UART Controller Initialization

- IO configuration: Configure GPIO multiplex as UART function, and set UART pins to internal pull-up mode (For detail, see the description in section 7.7 "GPIO").
- Baud-rate configuration:
  - Set UART baud-rate (refer to section 7.2.3.4);
  - Write **UART_FCR**[FIFOE] to 1 to enable TX/RX FIFO;
  - Write **UART_HALT**[HALT_TX] to 1 to disable TX transfer;
  - Set **UART_LCR**[DLAB] to 1, remain default configuration for other bits; set 0x00 offset address to the **UART_DLL** register, set 0x04 offset address to the **UART_DLH** register;
  - Write the high 8-bit of divisor to the **UART_DLH** register, and write the low 8-bit of divisor to the **UART_DLL** register;
  - Set **UART_LCR**[DLAB] to 0, remain default configuration for other bits; set 0x00 offset address to the **UART_RBR**/**UART_THR** register, set 0x04 offset address to the **UART_IER** register;
  - Set **UART_HALT**[HALT_TX] to 0 to enable TX transfer.

Step 3 Controller Parameter Configuration

- Set data width, stop bits, and even/odd parity type by writing the **UART_LCR** register.
- Reset, enable FIFO and set FIFO trigger condition by writing the **UART_FCR** register.
- Set the flow control parameter by writing the **UART_MCR** register.

Step 4 Interrupt Configuration

- Configure UART interrupt vector number to request UART interrupt (Refer to section 3.8 "GIC" module for interrupt vector number).
- In DMA mode, write **UART_IER** to 0 to disable interrupt; write **UART_HSK**[Handshake configuration] to 0xE5 to set DMA handshake mode; write **UART_FCR**[DMAM] to 1 to set DMA transmission/reception mode; set DMA parameter and request DMA interrupt according to DMA configuration process.
- In Interrupt mode, configure **UART_IER** to enable the corresponding interrupt according to requirements: such as transmit (TX) interrupt, receive (RX) interrupt, receive line status interrupt, RS48 interrupt, etc. (Here TX/RX interrupt is usually used).

7.2.4.2 Transferring/Receiving Data in Query Mode

Data transfer

Step 1 Write data to **UART_THR** to start data transfer.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 2 Check TX_FIFO status by reading **UART_USR**[TFNF]. If the bit is 1, data can continue to be written; if the bit is 0, wait for data transfer, and data cannot continue to write until FIFO is not full.

Data receive

Step 1 Check RX_FIFO status by reading **UART_USR**[RFNE].

Step 2 Read data from **UART_RBR** if RX_FIFO is not empty.

Step 3 If **UART_USR**[RFNE] is 0, data is received completely.

7.2.4.3 Transferring/Receiving Data in Interrupt Mode

Data transfer

Step 1 Set **UART_IER**[ETBEI] to 1 to enable the UART transmission interrupt.

Step 2 Write the data to be transmitted to **UART_THR**.

Step 3 When the data of TX_FIFO meets trigger condition (such as FIFO/2, FIFO/4), the UART transfer interrupt is generated.

Step 4 Check **UART_USR**[TFE] and determine whether TX_FIFO is empty. If **UART_USR**[TFE] is 1, it indicates that the data in TX_FIFO is transmitted completely.

Step 5 Clear **UART_IER**[ETBEI] to 0 to disable transfer interrupt.

Data receive

Step 1 Set **UART_IER**[ERBFI] to 1 to enable the UART reception interrupt.

Step 2 When the received data from RX_FIFO meets trigger condition (such as FIFO/2, FIFO/4), the UART receive interrupt is generated.

Step 3 Read data from **UART_RBR**.

Step 4 Check RX_FIFO status by reading **UART_USR**[RFNE] and determine whether to read data. If the bit is 1, continue to read data from **UART_RBR** until **UART_USR**[RFNE] is cleared to 0, which indicates data is received completely.

7.2.4.4 Transferring/Receiving Data in DMA Mode

Data transfer

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 1 Configure the UART DMA interrupt according to the initialization process.

Step 2 Configure DMA data channel, including the transfer source address, the transfer destination address, the number of data to be transferred, and the transfer type, and so on. (For details, see section 3.9 "DMA").

Step 3 Enable the DMA transfer function of the UART by setting the register of the DMA module.

Step 4 Determine whether UART data is transferred completely based on the DMA status. If all data is transferred completely, disable the DMA transfer function of the UART.

Data receive

Step 1 Configure DMA data channel, including the transfer source address, the transfer destination address, the number of data to be transferred, and the transfer type, and so on. (For details, see section 3.9 "DMA").

Step 2 Enable the DMA receive function of the UART by setting the register of the DMA module.

Step 3 Determine whether UART data is received completely based on the DMA status. If all data is received completely, disable the DMA receive function of the UART.

7.2.5 Register List

Module Name	Base Address
UART0	0x02500000
UART1	0x02500400
UART2	0x02500800
UART3	0x02500C00
R_UART0	0x07080000

Register Name	Offset	Description
UART_RBR	0x0000	UART Receive Buffer Register
UART_THR	0x0000	UART Transmit Holding Register
UART_DLL	0x0000	UART Divisor Latch Low Register
UART_DLH	0x0004	UART Divisor Latch High Register
UART_IER	0x0004	UART Interrupt Enable Register
UART_IIR	0x0008	UART Interrupt Identity Register
UART_FCR	0x0008	UART FIFO Control Register
UART_LCR	0x000C	UART Line Control Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
UART_MCR	0x0010	UART Modem Control Register
UART_LSR	0x0014	UART Line Status Register
UART_MSR	0x0018	UART Modem Status Register
UART_SCH	0x001C	UART Scratch Register
UART_USR	0x007C	UART Status Register
UART_TFL	0x0080	UART TX FIFO Level Register
UART_RFL	0x0084	UART RX FIFO Level Register
UART_HSK	0x0088	UART DMA Handshake Configuration Register
UART_DMA_REQ_EN	0x008C	UART DMA Request Enable Register
UART_HALT	0x00A4	UART Halt TX Register
UART_DBG_DLL	0x00B0	UART Debug DLL Register
UART_DBG_DLH	0x00B4	UART Debug DLH Register
UART_A_FCC	0x00F0	UART FIFO Clock Control Register
UART_A_RXDMA_CTRL	0x0100	UART RXDMA Control Register
UART_A_RXDMA_STR	0x0104	UART RXDMA Start Register
UART_A_RXDMA_STA	0x0108	UART RXDMA Status Register
UART_A_RXDMA_LMT	0x010C	UART RXDMA Limit Register
UART_A_RXDMA_SADDRL	0x0110	UART RXDMA Buffer Start Address Low Register
UART_A_RXDMA_SADDRH	0x0114	UART RXDMA Buffer Start Address High Register
UART_A_RXDMA_BL	0x0118	UART RXDMA Buffer Length Register
UART_A_RXDMA_IE	0x0120	UART RXDMA Interrupt Enable Register
UART_A_RXDMA_IS	0x0124	UART RXDMA Interrupt Status Register
UART_A_RXDMA_WADDRL	0x0128	UART RXDMA Write Address Low Register
UART_A_RXDMA_WADDRH	0x012C	UART RXDMA Write Address high Register
UART_A_RXDMA_RADDRL	0x0130	UART RXDMA Read Address Low Register
UART_A_RXDMA_RADDRH	0x0134	UART RXDMA Read Address high Register
UART_A_RXDMA_DCNT	0x0138	UART RADMA Data Count Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.2.6 Register Description

7.2.6.1 0x0000 UART Receiver Buffer Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: UART_RBR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R	0x0	RBR Receiver Buffer Register Data byte received on the serial input port (sin) in UART mode, or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in *UART_LCR* is set. If in FIFO mode and FIFOs are enabled (The *UART_FCR*[0] is set to 1), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register can not read before the next data character arrives, then the data already in the FIFO are preserved, but any incoming data are lost and an overrun error occurs.

7.2.6.2 0x0000 UART Transmit Holding Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.2.6.3 0x0000 UART Divisor Latch Low Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: UART_DLL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R/W	0x0	DLL Divisor Latch Low Lower 8 bits of 16 bits, read/write, Divisor Latch Register contains the baud rate divisor for the UART. This register may only be accessed when the DLAB bit (*UART_LCR*[7]) is set and the UART is not busy (*UART_USR*[0] is 0). 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 when the Divisor Latch Registers (*UART_DLL* and *UART_DLH*) are set to 0, the baud clock is disabled and no serial communications occur. Also, once the DLL is set, at least 8 clock cycles of the slowest UART clock should be allowed to pass before transmitting or receiving data.

7.2.6.4 0x0004 UART Divisor Latch High Register (Default Value: 0x0000_0000)

Offset: 0x0004			Register Name: UART_DLH
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R/W	0x0	DLH Divisor Latch High Upper 8 bits of 16 bits, read/write, Divisor Latch Register contains the baud rate divisor for the UART. This register may only be accessed when the DLAB bit (*UART_LCR*[7]) is set and the UART is not busy (*UART_USR*[0] is 0). 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 when the Divisor Latch Registers (*UART_DLL* and *UART_DLH*) is set to 0, the baud clock is disabled and no serial communications occur. Also, once the DLH is set, at least 8 clock cycles of the slowest UART clock should be allowed to pass before transmitting or receiving data.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.2.6.5 0x0004 UART Interrupt Enable Register (Default Value: 0x0000_0000)

Offset: 0x0004			Register Name: UART_IER
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R/W	0x0	PTIME Programmable THRE Interrupt Mode Enable This is used to enable/disable the generation of THRE Interrupt. 0: Disabled 1: Enabled
6:5	/	/	/
4	R/W	0x0	RS485_INT_EN RS485 Interrupt Enable 0: Disabled 1: Enabled
3	R/W	0x0	EDSSI Enable Modem Status Interrupt This is used to enable/disable the generation of Modem Status Interrupt. This is the fourth highest priority interrupt. 0: Disabled 1: Enabled
2	R/W	0x0	ELSI Enable Receiver Line Status Interrupt This is used to enable/disable the generation of Receiver Line Status Interrupt. This is the highest priority interrupt. 0: Disabled 1: Enabled
1	R/W	0x0	ETBEI Enable Transmit Holding Register Empty Interrupt This is used to enable/disable the generation of Transmitter Holding Register Empty Interrupt. This is the third-highest priority interrupt. 0: Disabled 1: Enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0004			Register Name: UART_IER
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	ERBFI Enable Received Data Available Interrupt This is used to enable/disable the generation of Received Data Available Interrupt and the Character Timeout Interrupt (if in FIFO mode and FIFOs enabled). These are the second-highest priority interrupt. 0: Disabled 1: Enabled

7.2.6.6 0x0008 UART Interrupt Identity Register (Default Value: 0x0000_0001)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Interrupt ID	Priority Level	Interrupt Type	Interrupt Source	Interrupt Reset
0001	-	None	None	-
0110	Highest	Receiver line status	Overrun/parity/framing errors or break interrupt	Reading the line status register
0011	Second	RS485 Interrupt	In RS485 mode, receives address data and match setting address	Writes 1 to addr flag to reset
0100	Third	Received data available	Receiver data available (non-FIFO mode or FIFOs disabled) or RCVR FIFO trigger level reached (FIFO mode and FIFOs enabled)	Reading the receiver buffer register (non-FIFO mode or FIFOs disabled) or the FIFO drops below the trigger level (FIFO mode and FIFOs enabled)
1100	Fourth	Character timeout indication	No characters in or out of the RCVR FIFO during the last 4 character times and there is at least 1character in it during This time	Reading the receiver buffer register
0010	Fifth	Transmit holding register empty	Transmitter holding register empty (Program THRE mode disabled) or XMIT FIFO at or below threshold (Program THRE mode enabled)	Reading the IIR register (if the source of interrupt); or, writing into THR (FIFOs or THRE mode not selected or disabled) or XMIT FIFO above threshold (FIFOs and THRE mode selected and enabled).
0000	Sixth	Modem status	Clear to send or data set ready or ring indicator or data carrier detect. Note that if the autoflow control mode is enabled, a change in CTS (that is, DCTS set) does not cause an interrupt.	Reading the Modem status register
0111	Seventh	Busy detect indication	UART_16550_COMPATIBLE = NO and master has tried to write to the Line Control Register while the UART is busy (USR[0] is set to one).	Reading the UART status register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.2.6.7 0x0008 UART FIFO Control Register (Default Value: 0x0000_0000)

Offset: 0x0008			Register Name: UART_FCR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:6	W	0x0	RT RCVR Trigger This is used to select the trigger level in the receiver FIFO when the Received Data Available Interrupt is generated. In the autoflow control mode, it is used to determine when the rts_n signal is de-asserted. It also determines when the dma_rx_req_n signal is asserted in certain modes of operation. 00: 1 character in the FIFO 01: FIFO ¼ full 10: FIFO ½ full 11: FIFO-2 less than full
5:4	W	0x0	TFT TX Empty Trigger This is used to select the empty threshold level when the THRE Interrupts are generated and the mode is active. It also determines when the dma_tx_req_n signal is asserted in certain modes of operation. 00: FIFO empty 01: 2 characters in the FIFO 10: FIFO ¼ full 11: FIFO ½ full

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0008			Register Name: UART_FCR
Bit	**Read/Write**	**Default/Hex**	**Description**
3	W	0x0	DMAM DMA Mode 0: Mode 0 In this mode, when the PTE in *UART_HALT* is high and TX FIFO is enabled, the TX DMA request will be set when the TFL in *UART_TFL* is less than or equal to FIFO Trigger Level (otherwise it will be cleared). When the PTE is high and TX FIFO is disabled, the TX DMA request will be set only if the THR in *UART_THR* is empty. If the PTE is low, the TX DMA request will be set only if the TX FIFO (TX FIFO enabled) or THR (TX FIFO disabled) is empty. When the DMA_PTE_RX in *UART_HALT* is high and RX FIFO is enabled, the RX DRQ will be set only if the RFL in *UART_RFL* is equal to or more than FIFO Trigger Level, otherwise, it will be cleared. 1: Mode 1 In this mode, TX FIFO should be enabled. If the PTE in *UART_HALT* is high, the TX DMA request will be set when the TFL in *UART_TFL* is less than or equal to FIFO Trigger Level; If the PTE is low, the TX DMA request will be set when TX FIFO is empty. Once the request is set, it is cleared only when TX FIFO is full. If the RFL in *UART_RFL* is equal to or more than FIFO Trigger Level or there is a character timeout, the RX DRQ will be set; Once the RX DRQ is set, it is cleared only when RX FIFO (RX FIFO enabled) or RBR (RX FIFO disabled) is empty.
2	W	0x0	XFIFOR XMIT FIFO Reset The bit resets the control part of the transfer FIFO and treats the FIFO as empty. This also de-asserts the DMA TX request. It is 'self-cleared'. It is not necessary to clear this bit.
1	W	0x0	RFIFOR RCVR FIFO Reset The bit resets the control part of the receive FIFO and treats the FIFO as empty. This also de-asserts the DMA RX request. It is 'self-cleared'. It is not necessary to clear this bit.
0	W	0x0	FIFOE Enable FIFOs The bit enables/disables the transmitting (XMIT) and receiving (RCVR) FIFOs. Whenever the value of this bit is changed, both the XMIT and RCVR controller part of FIFOs is reset.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.2.6.8 0x000C UART Line Control Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.2.6.9 0x0010 UART Modem Control Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.2.6.10 0x0014 UART Line Status Register (Default Value: 0x0000_0060)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.2.6.11 0x0018 UART Modem Status Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.2.6.12 0x001C UART Scratch Register (Default Value: 0x0000_0000)

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

7.2.6.13 0x007C UART Status Register (Default Value: 0x0000_0006)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.2.6.14 0x0080 UART TX FIFO Level Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.2.6.15 0x0084 UART RX FIFO Level Register (Default Value: 0x0000_0000)

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

7.2.6.16 0x0088 UART DMA Handshake Configuration Register (Default Value: 0x0000_00A5)

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

7.2.6.17 0x008C UART DMA Request Enable Register(Default Value: 0x0000_0003)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.2.6.18 0x00A4 UART Halt TX Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.2.6.19 0x00B0 UART DBG DLL Register (Default Value: 0x0000_0000)

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

7.2.6.20 0x00B4 UART DBG DLH Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.2.6.21 0x00F0 UART FIFO Clock Control Register (Default Value: 0x0000_0003)

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

7.2.6.22 0x0100 UART RXDMA Control Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.2.6.23 0x0104 UART RXDMA Start Register (Default Value: 0x0000_0000)

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

7.2.6.24 0x0108 UART RXDMA Status Register (Default Value: 0x0000_0000)

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

7.2.6.25 0x010C UART RXDMA Limit Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.2.6.26 0x0110 UART RXDMA Buffer Start Address Low Register (Default Value: 0x0000_0000)

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

7.2.6.27 0x0114 UART RXDMA Buffer Start Address High Register (Default Value: 0x0000_0000)

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

7.2.6.28 0x0118 UART RXDMA Buffer Length Register (Default Value: 0x0000_0000)

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

7.2.6.29 0x0120 UART RXDMA Interrupt Enable Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.2.6.30 0x0124 UART RXDMA Interrupt Status Register (Default Value: 0x0000_0000)

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

7.2.6.31 0x0128 UART RXDMA Write Address Low Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.2.6.32 0x012C UART RXDMA Write Address High Register (Default Value: 0x0000_0000)

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

7.2.6.33 0x0130 UART RXDMA Read Address Low Register (Default Value: 0x0000_0000)

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

7.2.6.34 0x0134 UART RXDMA Read Address High Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.2.6.35 0x0138 UART RXDMA Data Count Register(Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.3 SPI

7.3.1 Overview

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

The SPI has the following features:

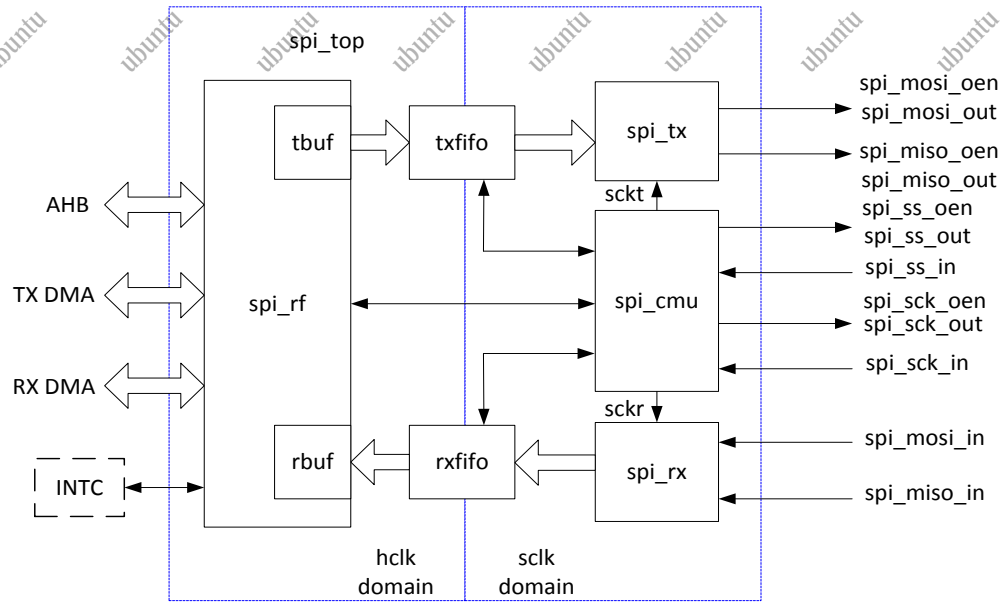
- Full-duplex synchronous serial interface
- Master/slave configurable
- 8-bit wide by 64-entry FIFO for both transmit and receive data
- Polarity and phase of the Chip Select (SPI_SS) and SPI Clock (SPI_SCLK) are configurable
- Supports interrupts and DMA
- Supports mode0, mode1, mode2, and mode3
-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

7.3.2 Block Diagram

Figure 7-17 shows a block diagram of the SPI.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-17 SPI Block Diagram



SPI contains the following sub-blocks:

Table 7-9 SPI Sub-blocks

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.3.3 Functional Descriptions

7.3.3.1 External Signals

The following table describes the external signals of SPI. The MOSI and MISO are bidirectional I/O, when SPI is as a master device, the CLK and CS are the output pin; when SPI is as a slave device, the CLK and CS are the input pin. When using SPI, the corresponding PADs are selected as SPI function via section 7.7 "GPIO".

Table 7-10 SPI External Signals

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

7.3.3.2 Clock Sources

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

Table 7-11 SPI Clock Sources

Clock Sources	Description
DCXO24M	24 MHz Crystal

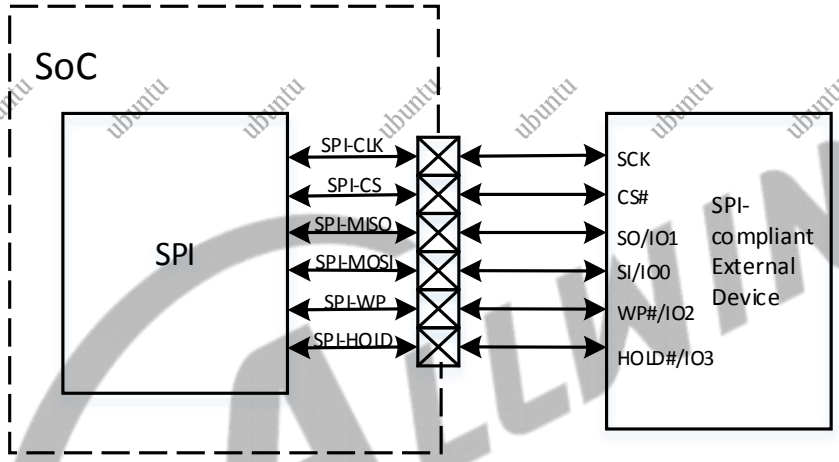
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.3.3.3 Typical Application

Figure 7-18 shows the application block diagram when the SPI master device is connected to a slave device.

Figure 7-18 SPI Application Block Diagram



7.3.3.4 SPI Transmit Format

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

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

Table 7-12 SPI Transmit Format

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Figure 7-19 and Figure 7-20 describe four waveforms for SPI_SCLK.

Figure 7-19 SPI Phase 0 Timing Diagram

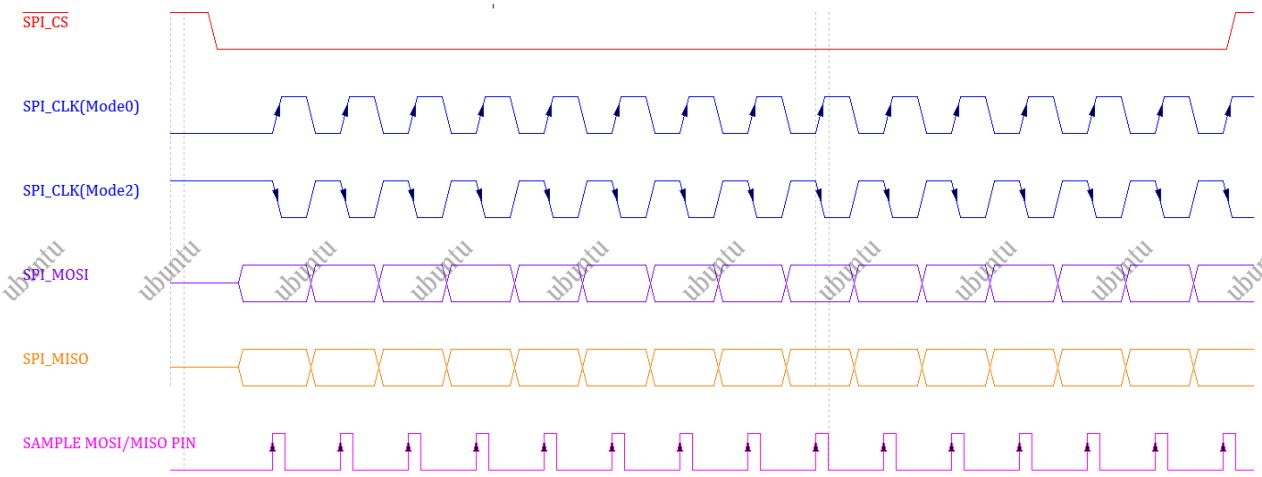
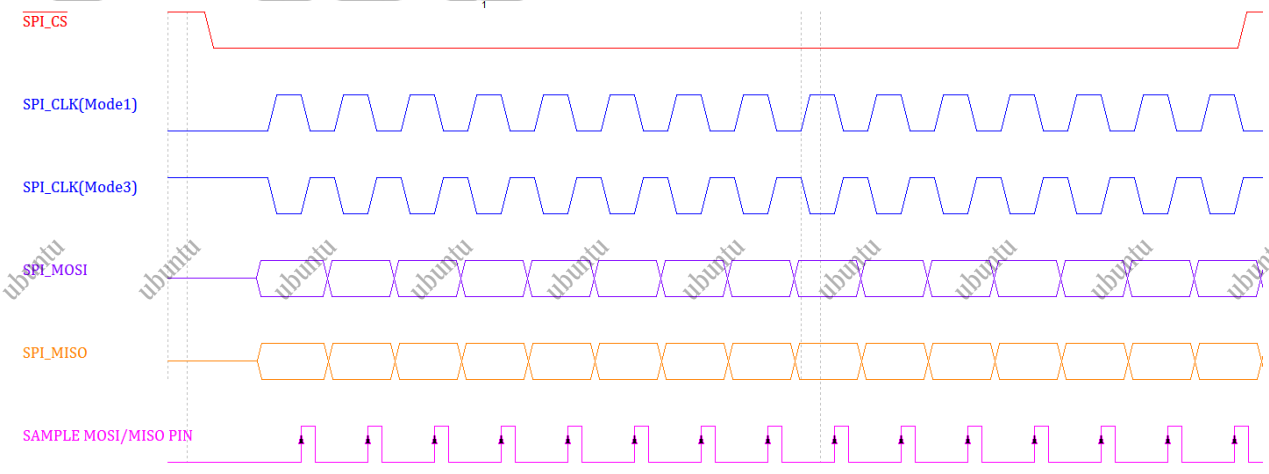


Figure 7-20 SPI Phase 1 Timing Diagram



7.3.3.5 SPI Master and Slave Mode

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

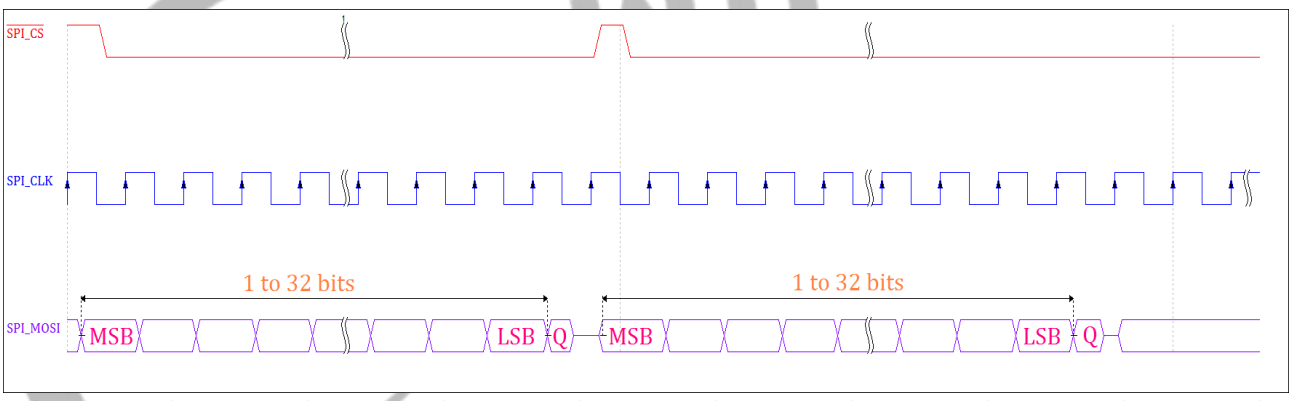
In master mode, the SPI_CLK is generated and transmitted to the external device, and the data from the TX FIFO is transmitted on the MOSI pin, the data from the slave device is received on the MISO pin and sent to RX FIFO. The Chip Select (SPI_SS) is an active low signal, and it must be set low before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

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

7.3.3.6 SPI 3-Wire Mode

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

Figure 7-21 SPI 3-Wire Mode

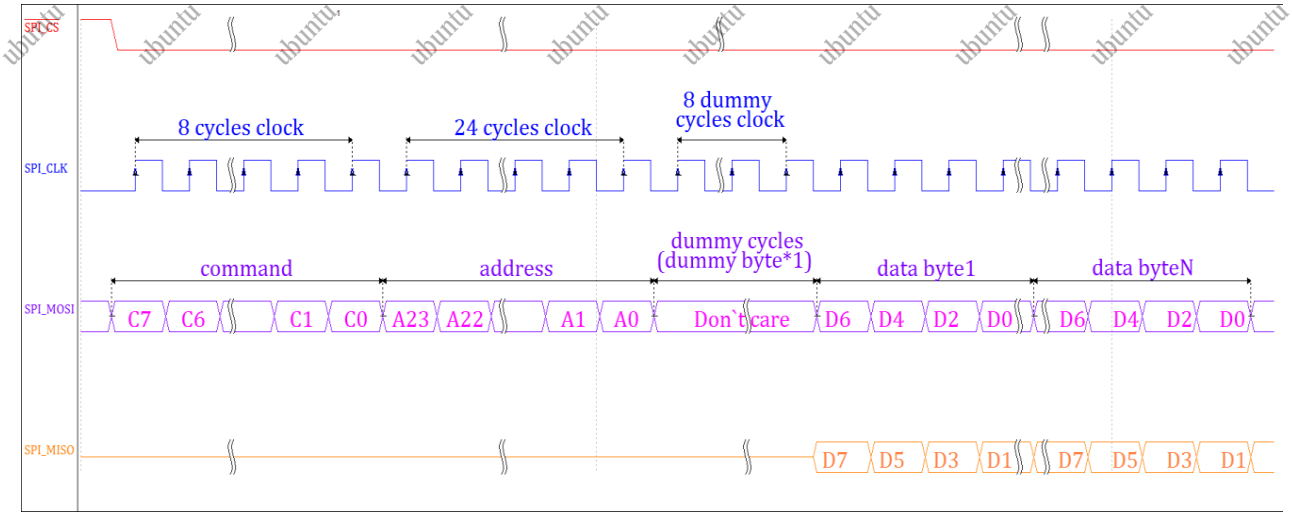


7.3.3.7 SPI Dual-Input/Dual-Output and Dual I/O Mode

The dual read mode (SPI x2) is selected when the *DRM* is set in **SPI_BCC**[28]. Using the dual mode allows data to be transferred to or from the device at double the rate of standard single mode, the data can be read at fast speed using two data bits (MOSI and MISO) at a time. The following figure describes the dual-input/dual-output SPI (Figure 7-22) and the dual I/O SPI (Figure 7-23).

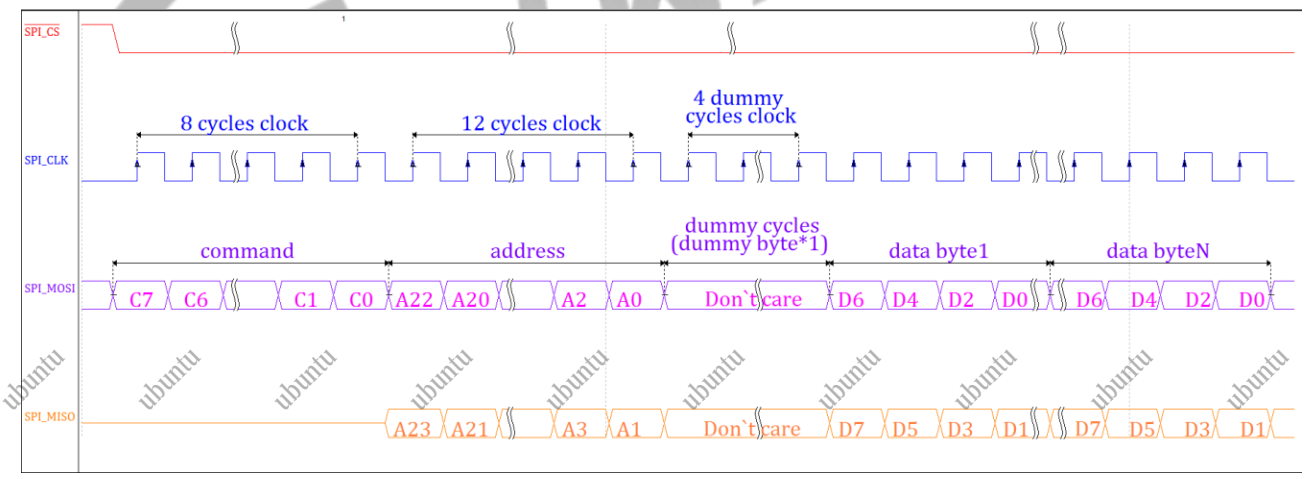
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-22 SPI Dual-Input/Dual-Output Mode



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

Figure 7-23 SPI Dual I/O Mode



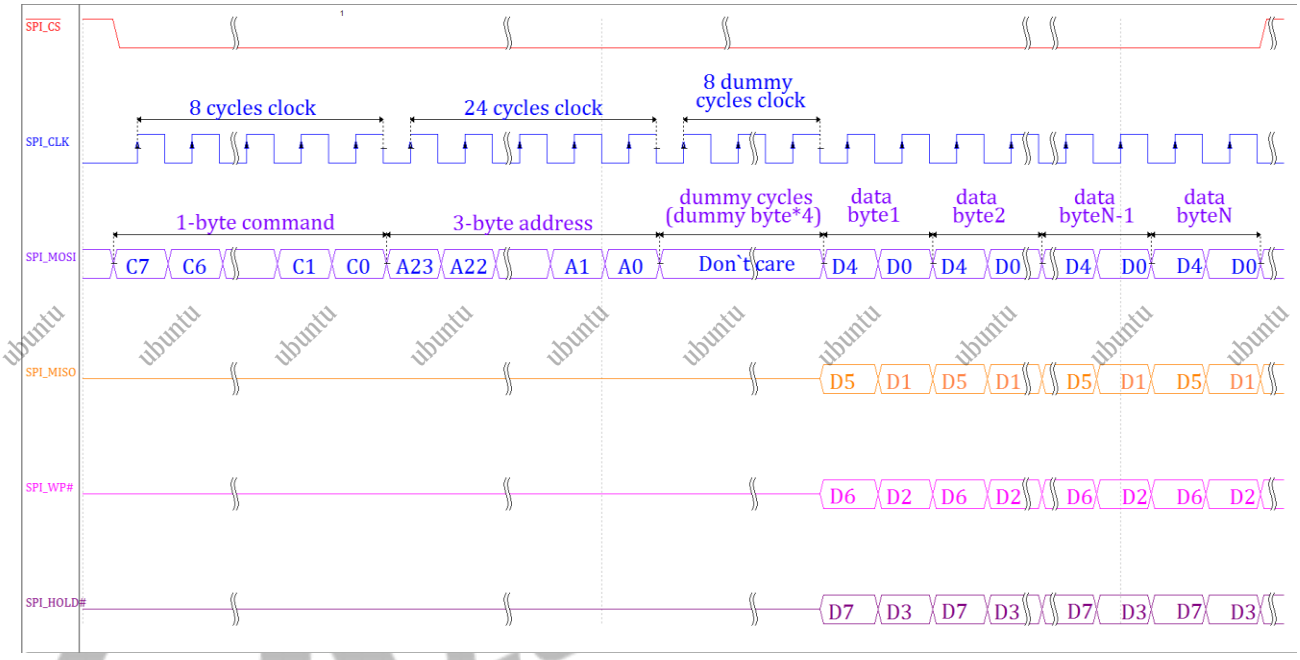
In the dual I/O SPI mode, only the command bytes output in a unit of a single bit in serial mode through the SPI_MOSI line. The address bytes and the dummy bytes output in a unit of dual bits through the SPI_MOSI and SPI_MISO. And the data bytes output (write) and input (read) in a unit of dual bits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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.3.3.8 SPI Quad-Input/Quad-Output Mode

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

Figure 7-24 SPI Quad-Input/Quad-Output Mode



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

7.3.3.9 Transmission/Reception Bursts in Master Mode

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.3.3.10 SPI Sample Mode and Run Clock Configuration

The SPI controller runs at 3 kHz~100 MHz at its interface to external SPI devices. The internal SPI clock should run at the same frequency as the outgoing clock in the master mode. The SPI clock is selected from different clock sources, the SPI must configure different work mode. There are three work modes: normal sample mode, delay half-cycle sample mode, delay one-cycle sample mode. The Delay half-cycle sample mode is the default mode of the SPI controller. When the SPI runs at 40 MHz or below 40 MHz, the SPI can work at normal sample mode or delay half-cycle sample mode. When the SPI runs over 80 MHz, setting the SDC bit in the SPI Transfer Control Register to '1' makes the internal read sample point with a half-cycle delay of SPI_CLK, which is used in high speed read operation to reduce the error caused by the time delay of SPI_CLK between master and slave. Table 7-13 and Table 7-14 show the different configurations of the SPI sample mode.

Table 7-13 SPI Old Sample Mode and Run Clock

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

CAUTION

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

Table 7-14 SPI New Sample Mode

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.3.3.11 SPI Error Conditions

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

1. **TX_FIFO Underrun**

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

2. **TX_FIFO Overflow**

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

3. **RX_FIFO Underrun**

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

4. **RX_FIFO Overflow**

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.3.4 Programming Guidelines

7.3.4.1 Writing/Reading Data Process

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

Write Data: The CPU or DMA must write data on the **SPI_TXD** register, the data on the register are automatically moved to TX FIFO.

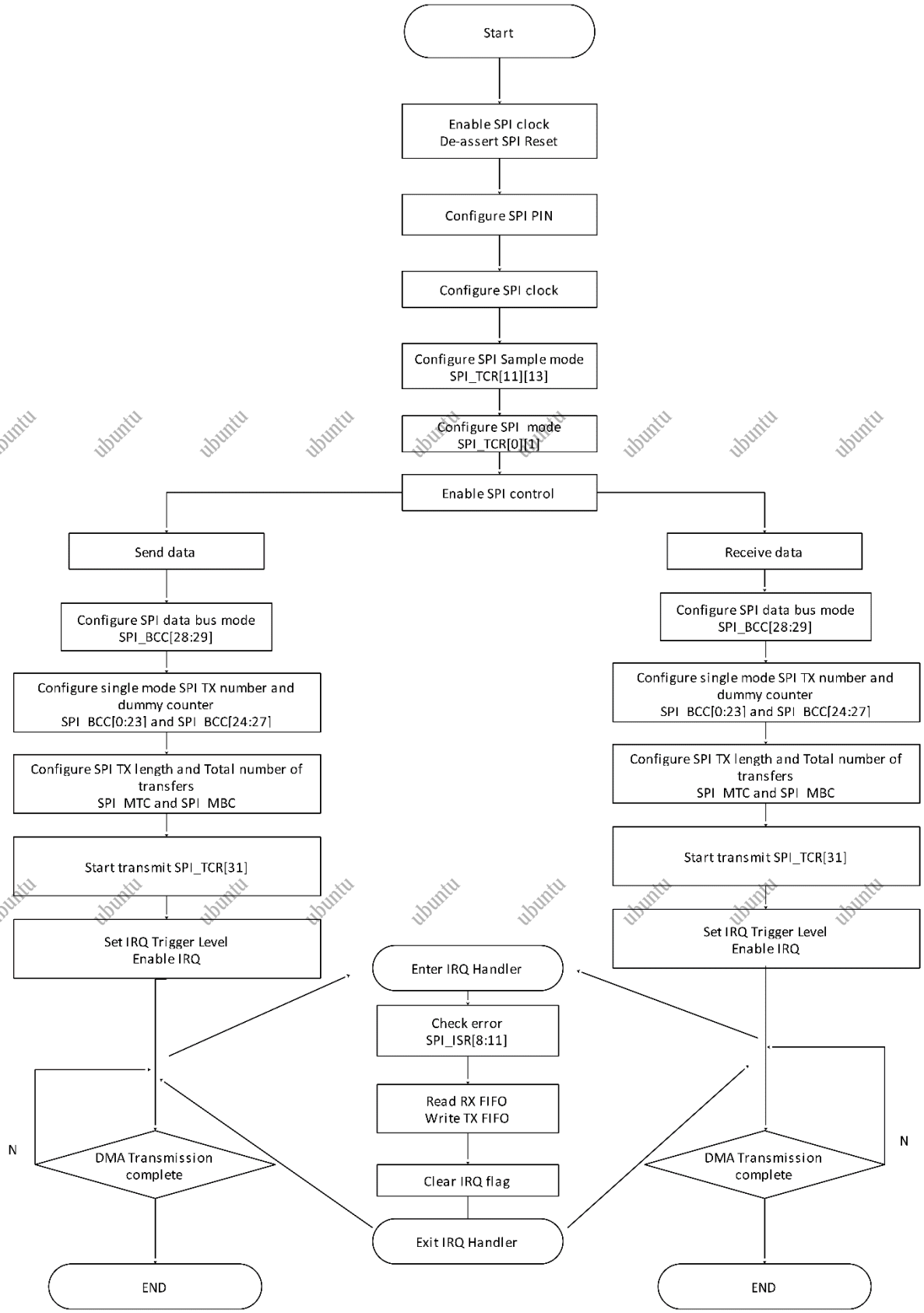
Read Data: To read data from RX FIFO, the CPU or DMA must access the **SPI_RXD** register and the data are automatically sent to the **SPI_RXD** register.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

CPU Mode

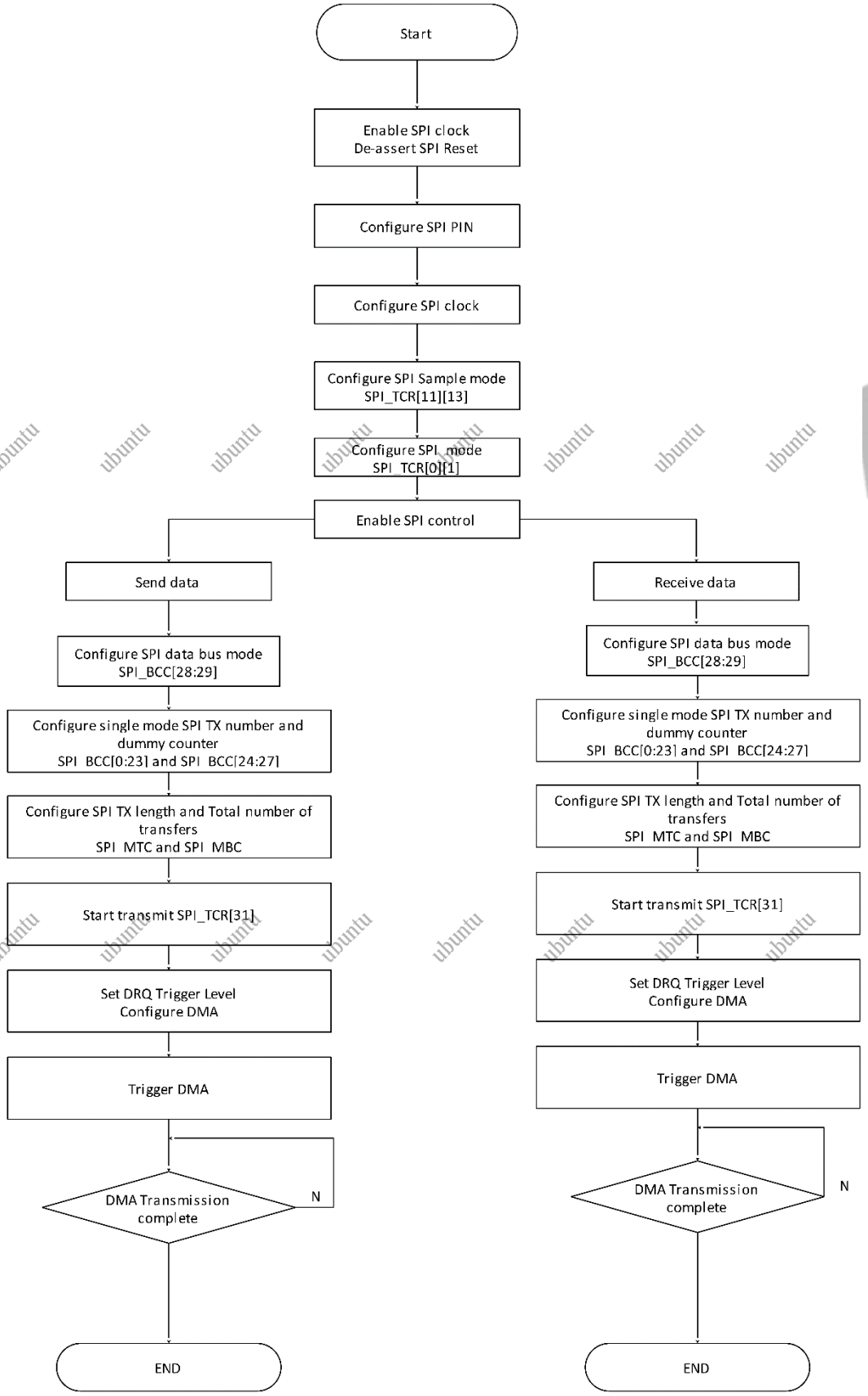
Figure 7-25 SPI Write/Read Data in CPU Mode



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

DMA Mode

Figure 7-26 SPI Write/Read Data in DMA Mode



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.3.4.2 Calibrating Delay Chain

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

The steps to calibrate delay chain are as follows:

Step 1 Enable SPI. To calibrate the delay chain by the operation registers in SPI, the SPI must be enabled through AHB reset and AHB clock gating control registers.

Step 2 Configure a proper clock for SPI. The calibration delay chain is based on the clock for SPI from CCU.

Step 3 Set proper initial delay value. Write 0xA0 to the SPI Sample Delay Control Register to set initial delay value 0x20 to delay chain. Then write 0x0 to the SPI Sample Delay Control Register to clear this value.

Step 4 Write 0x8000 to the SPI Sample Delay Control Register to start to calibrate the delay chain.

Step 5 Wait until the flag (bit14 in the SPI Sample Delay Control Register) of calibration done is set. The number of delay cells is shown at the bit[13:8] of the SPI Sample Delay Control Register. The delay time generated by these delay cells is equal to the cycle of the SPI clock nearly. This value is the result of calibration.

Step 6 Calculate the delay time of one delay cell according to the cycle of the SPI clock and the result of calibration.

7.3.5 Register List

Module Name	Base Address
SPI0	0x04025000

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.3.6 Register Description

7.3.6.1 0x0004 SPI Global Control Register (Default Value: 0x0000_0080)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.3.6.2 0x0008 SPI Transfer Control Register (Default Value: 0x0000_0087)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.3.6.3 0x0010 SPI Interrupt Control Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.3.6.4 0x0014 SPI Interrupt Status Register (Default Value: 0x0000_0032)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.3.6.5 0x0018 SPI FIFO Control Register (Default Value: 0x0040_0001)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.3.6.6 0x001C SPI FIFO Status Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.3.6.7 0x0020 SPI Wait Clock Register (Default Value: 0x0000_0000)

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

7.3.6.8 0x0028 SPI Sample Delay Control Register (Default Value: 0x0000_2000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.3.6.9 0x0030 SPI Master Burst Counter Register (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: SPI_MBC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	/	/	/
23:0	R/W	0x0	MBC Master Burst Counter In master mode, this field specifies the total burst number. The total transfer data include the TXD, RXD, and dummy burst. 0: 0 burst 1: 1 burst … N: N bursts Cannot be written when XCH=1.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.3.6.10 0x0034 SPI Master Transmit Counter Register (Default Value: 0x0000_0000)

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

7.3.6.11 0x0038 SPI Master Burst Control Counter Register (Default Value: 0x0000_0000)

Offset: 0x0038			Register Name: SPI_BCC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/
29	R/W	0x0	Quad_EN Quad Mode Enable Quad mode includes Quad-Input and Quad-Output. 0: Quad mode disabled 1: Quad mode enabled Cannot be written when XCH=1.
28	R/W	0x0	DRM Master Dual Mode RX Enable It is only valid when Quad_Mode_EN=0. 0: RX uses the single-bit mode 1: RX uses the dual-bit mode Cannot be written when XCH=1.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0038			Register Name: SPI_BCC
Bit	Read/Write	Default/Hex	Description
27:24	R/W	0x0	DBC Master Dummy Burst Counter In master mode, this field specifies the burst number that should be sent before receiving in dual SPI mode. The data does not care by the device. 0: 0 burst 1: 1 burst … N: N bursts Cannot be written when XCH=1.
23:0	R/W	0x0	STC Master Single Mode Transmit Counter In master mode, this field specifies the burst number that should be sent in the single mode before automatically sending dummy bursts. This is the first transmit counter in all bursts. 0: 0 burst 1: 1 burst … N: N bursts Cannot be written when XCH=1.

7.3.6.12 0x0040 SPI Bit-Aligned Transfer Configure Register (Default Value: 0x0000_00A0)

Offset: 0x0040			Register Name: SPI_BATC
Bit	Read/Write	Default/Hex	Description
31	R/WAC	0x0	TCE Transfer Control Enable In master mode, it is used to start to transfer the serial bit frame, it is only valid when *Work Mode Select*==0x10/0x11. 0: Idle 1: Initiates transfer Writing “1” to this bit will start to transfer serial bit frame (the value comes from the *SPI TX Bit Register* or *SPI RX Bit Register*), and will auto-clear after the bursts transfer completely. Writing ‘0’ to this bit has no effect.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0040			Register Name: SPI_BATC
Bit	Read/Write	Default/Hex	Description
30	R/W	0x0	MSMS Master Sample Standard 0: Delay Sample Mode 1: Standard Sample Mode In Standard Sample Mode, the SPI master samples the data at the standard rising edge of SCLK for each SPI mode; In Delay Sample Mode, the SPI master samples data at the edge that is half cycle delayed by the standard rising edge of SCLK defined in respective SPI mode.
29:26	/	/	/
25	R/W1C	0x0	TBC Transfer Bits Completed When set, this bit indicates that the last bit of the serial data frame in *SPI TX Bit Register* (or *SPI RX Bit Register*) has been transferred completely. Writing 1 to this bit clears it. 0: Busy 1: Transfer Completed It is only valid when *Work Mode Select*==0x10/0x11.
24	R/W	0x0	TBC_INT_EN Transfer Bits Completed Interrupt Enable 0: Disable 1: Enable It is only valid when *Work Mode Select*==0x10/0x11.
23:22	/	/	/
21:16	R/W	0x00	Configure the length of serial data frame (burst) of RX 000000: 0 bit 000001: 1 bit … 100000: 32 bits Other values: reserved It is only valid when *Work Mode Select*==0x10/0x11, and cannot be written when TCE (bit31) is 1.
15:14	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0040			Register Name: SPI_BATC
Bit	**Read/Write**	**Default/Hex**	**Description**
13:8	R/W	0x00	Configure the length of serial data frame (burst) of TX 000000: 0 bit 000001: 1 bit … 100000: 32 bits Other values: reserved It is only valid when *Work Mode Select*==0x10/0x11, and cannot be written when TCE (bit31) is 1.
7	R/W	0x1	SS_LEVEL When control SS signal manually, set this bit to '1' or '0' to control the level of SS signal. 0: Set SS to low 1: Set SS to high It is only valid when *Work Mode Select*==0x10/0x11, and only work in Mode0, cannot be written when TCE (bit31) is 1.
6	R/W	0x0	SS_OWNER SS Output Owner Select Usually, the controller sends the SS signal automatically with data together. When this bit is set to 1, the software must manually write *SS_LEVEL* to 1 or 0 to control the level of the SS signal. 0: SPI controller 1: Software It is only valid when *Work Mode Select*==0x10/0x11, and only work in Mode0, cannot be written when TCE (bit31) is 1.
5	R/W	0x1	SPOL SPI Chip Select Signal Polarity Control 0: Active high polarity (0 = Idle) 1: Active low polarity (1 = Idle) It is only valid when *Work Mode Select*==0x10/0x11, and only work in Mode0, cannot be written when TCE (bit31) is 1.
4	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.3.6.13 0x0044 SPI Bit-Aligned Clock Configuration Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.3.6.14 0x0048 SPI TX Bit Register (Default Value: 0x0000_0000)

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

7.3.6.15 0x004C SPI RX Bit Register (Default Value: 0x0000_0000)

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

7.3.6.16 0x0088 SPI Normal DMA Mode Control Register (Default Value: 0x0000_00E5)

Offset: 0x0088			Register Name: SPI_NDMA_MODE_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:6	R/W	0x11	ACK SELECT 00: dma_active is low 01: dma_active is high 10: dma_active is controlled by dma_request (DRQ) 11: dma_active is controlled by controller
5	R/W	0x1	ACK CARE 0: active fall do not care ack 1: active fall must after detect ack is high

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.3.6.17 0x0200 SPI TX Data Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.3.6.18 0x0300 SPI RX Data Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.4 SPI_DBI

7.4.1 Overview

The R329 provides a 3/4 line SPI display bus interface (SPI_DBI) for video data transmission. It supports DBI mode or SPI mode. The DBI mode is compatible with multiple video data formats at the same time. The SPI mode is used for low-cost display schemes.

The SPI mode has the following features:

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

The DBI mode has the following features:

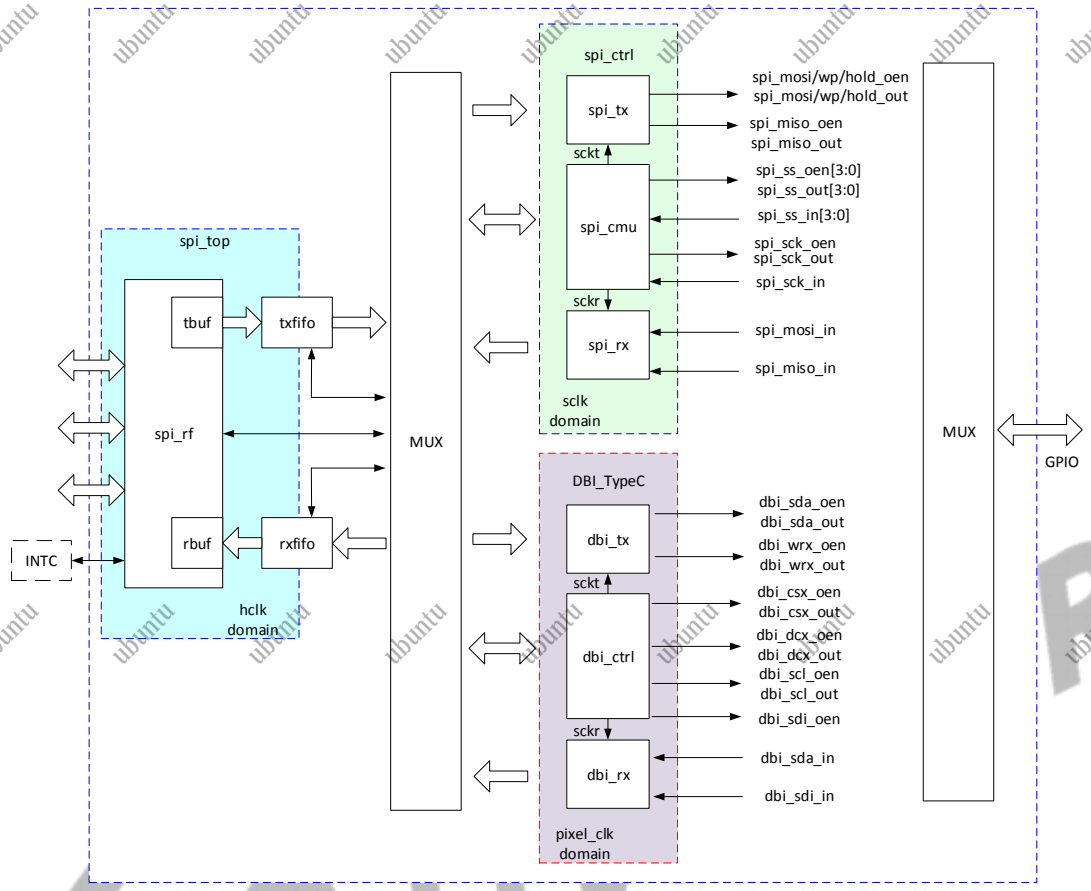
- Supports DBI Type C 3 Line/4 Line Interface Mode
- Supports 2 Data Lane Interface Mode
- Supports data source from CPU or DMA
- Supports RGB111/444/565/666/888 video format
- Maximum resolution of RGB666 240 x 320@30Hz with single data lane
- Maximum resolution of RGB888 240 x 320@60Hz or 320 x 480@30Hz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

7.4.2 Block Diagram

Figure 7-27 shows a block diagram of the SPI_DBI.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-27 SPI_DBI Block Diagram



SPI_DBI contains the following sub-blocks:

Table 7-15 SPI_DBI Sub-blocks

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.4.3 Functional Descriptions

7.4.3.1 External Signals

The following table describes the external signals of SPI_DBI. When using SPI_DBI, the corresponding PADs are selected as SPI_DBI function via section 7.7 "GPIO".

Table 7-16 SPI_DBI External Signals

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.4.3.2 Clock Sources

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

Table 7-17 SPI_DBI Clock Sources

Clock Sources	Description
DCXO24M	24 MHz Crystal
PLL_PERI(1X)	Peripheral Clock, the default value is 600 MHz
PLL_PERI(2X)	Peripheral Clock, the default value is 1200 MHz
PLL_AUDIO0(DIV2)	Audio Clock, the default value is 1536 MHz
PLL_AUDIO0(DIV5)	Audio Clock, the default value is 614.4 MHz

7.4.3.3 Typical Application

Figure 7-28 shows the application block diagram when the SPI master device is connected to a slave device.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-28 SPI Application Block Diagram

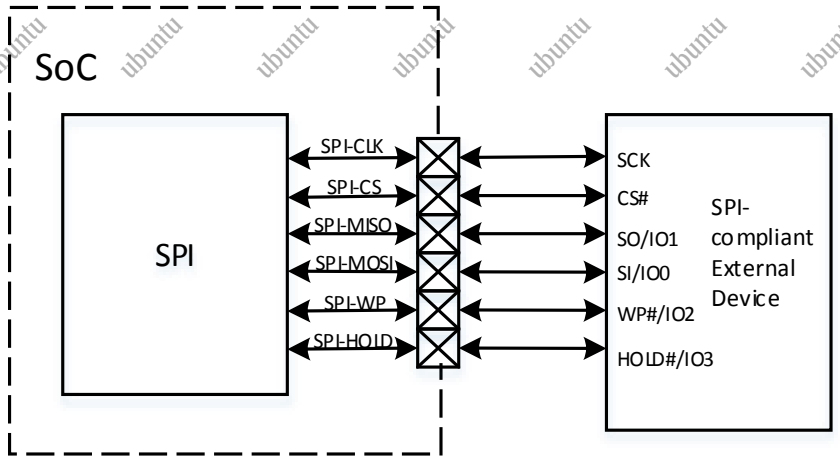
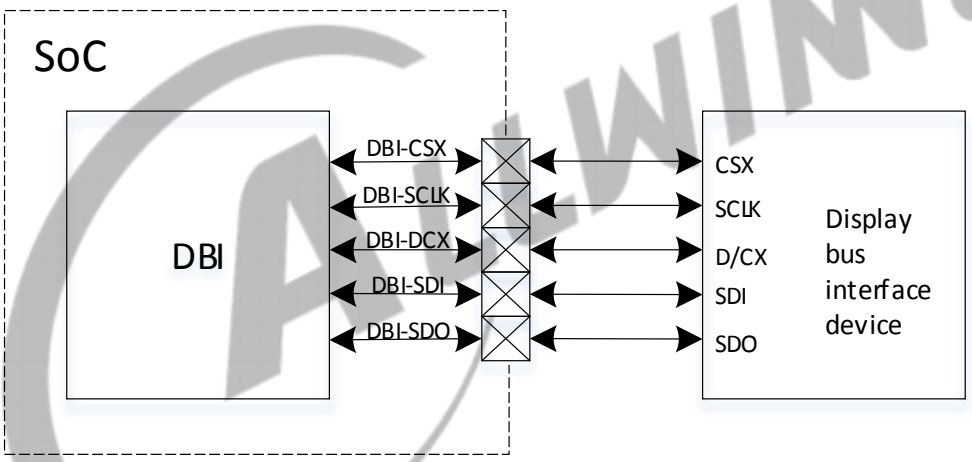


Figure 7-29 shows the application block diagram when the DBI master device is connected to a display bus interface device.

Figure 7-29 DBI Application Block Diagram



7.4.3.4 SPI Transmission Format

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

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Table 7-18 SPI Transmit Format

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

Figure 7-30 and Figure 7-31 describe four waveforms for SPI_SCLK.

Figure 7-30 SPI Phase 0 Timing Diagram

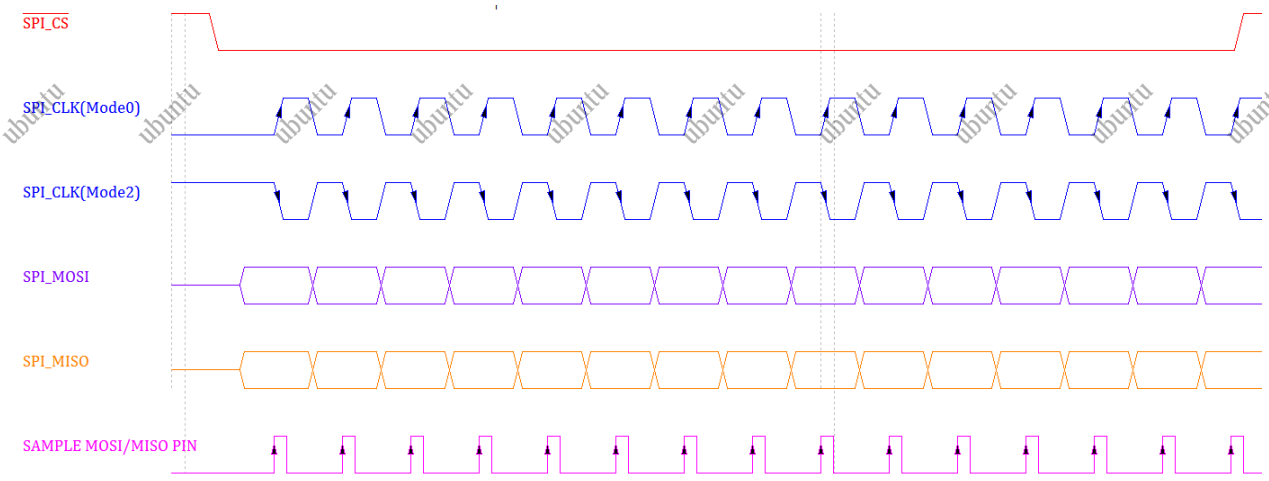
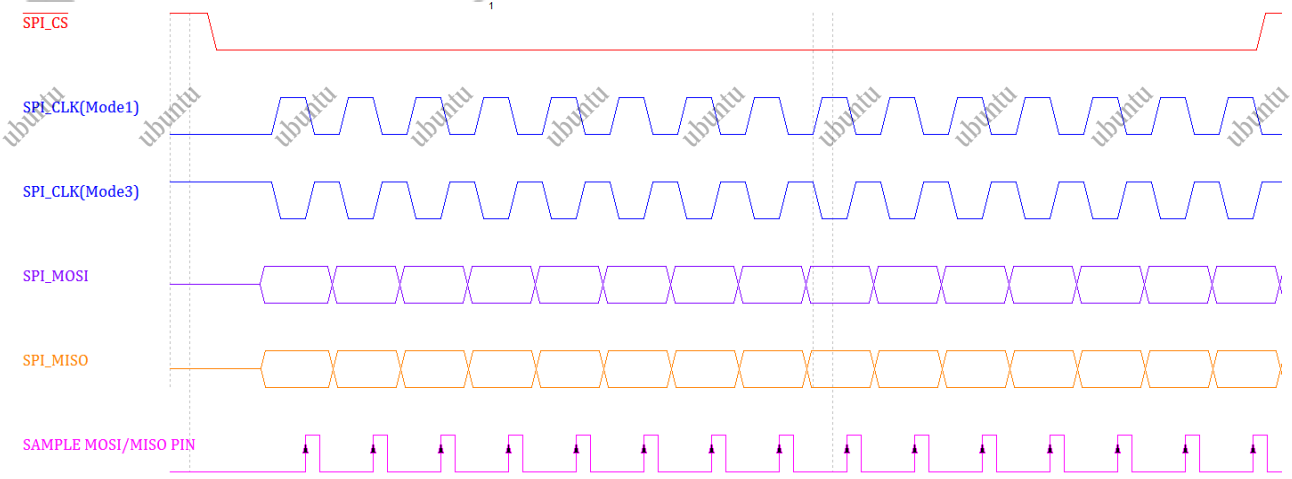


Figure 7-31 SPI Phase 1 Timing Diagram



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.4.3.5 SPI Master and Slave Mode

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

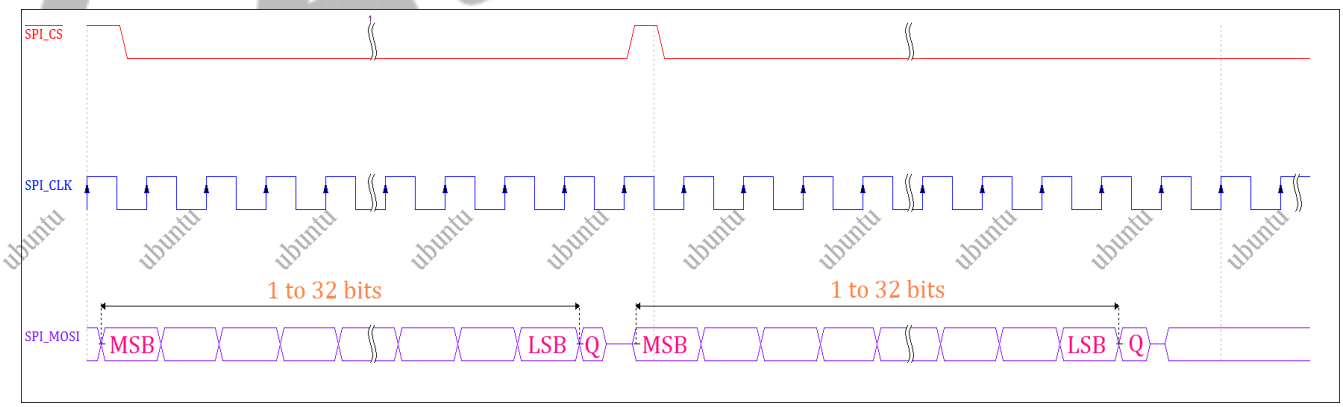
In master mode, the SPI_CLK is generated and transmitted to the external device, and the data from the TX FIFO is transmitted on the MOSI pin, the data from the slave is received on the MISO pin and sent to RX FIFO. The Chip Select (SPI_SS) is an active low signal, and it must be set low before the data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

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

7.4.3.6 SPI 3-Wire Mode

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

Figure 7-32 SPI 3-Wire Mode

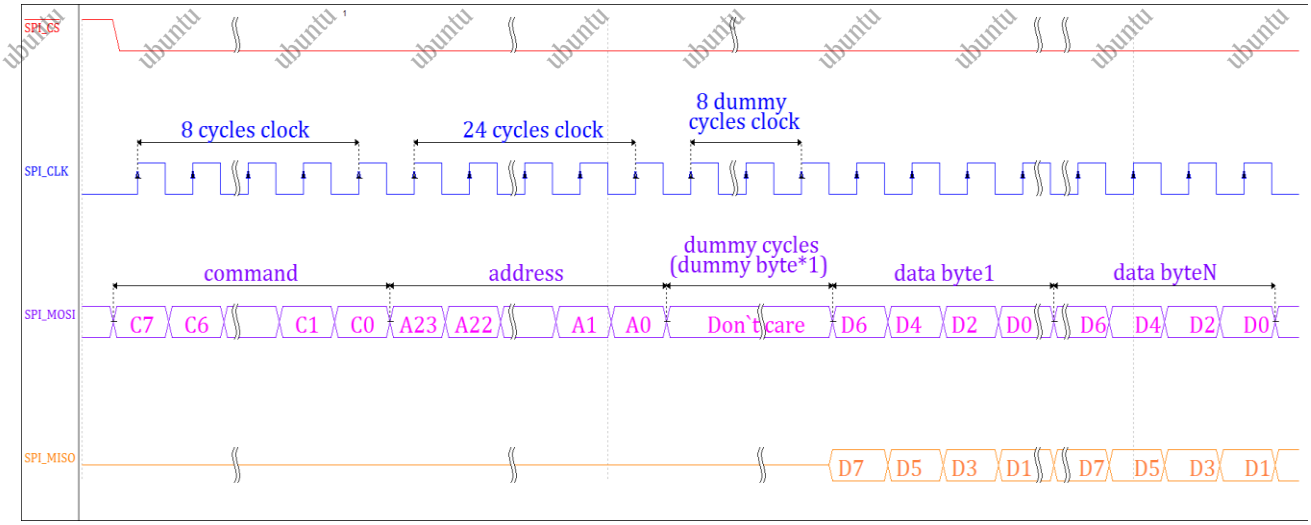


7.4.3.7 SPI Dual-Input/Dual-Output and Dual I/O Mode

The dual read mode (SPI x2) is selected when the *DRM* is set in **SPI_BCC**[28]. Using the dual mode allows data to be transferred to or from the device at double the rate of standard single mode SPI devices, the data can be read at fast speed using two data bits (MOSI and MISO) at a time. The following figure describes the dual-input/dual-output SPI (Figure 7-33) and the dual I/O SPI (Figure 7-34).

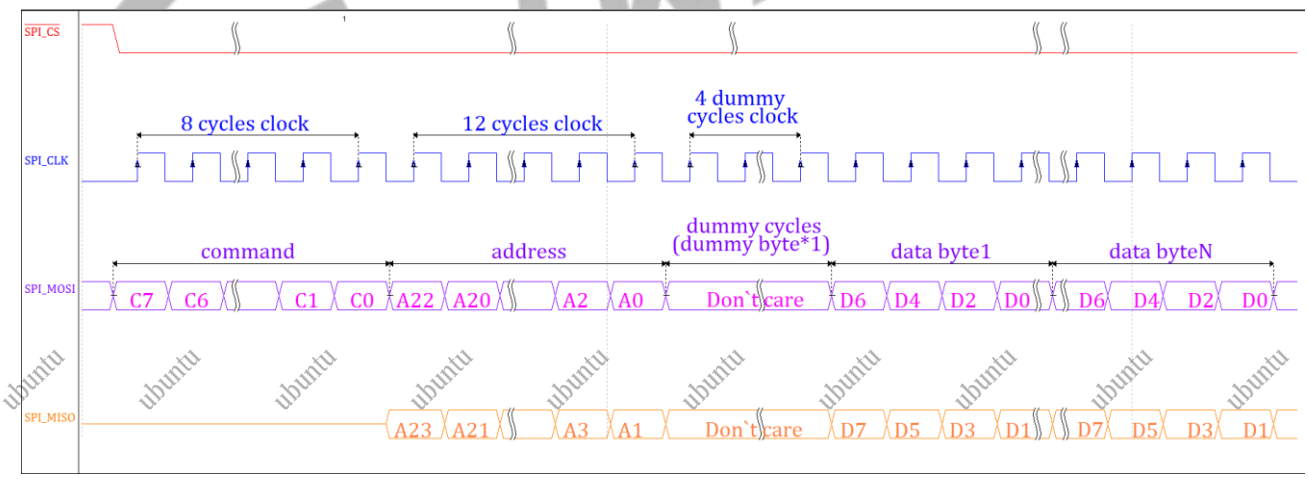
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-33 SPI Dual-Input/Dual-Output Mode



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

Figure 7-34 SPI Dual I/O Mode



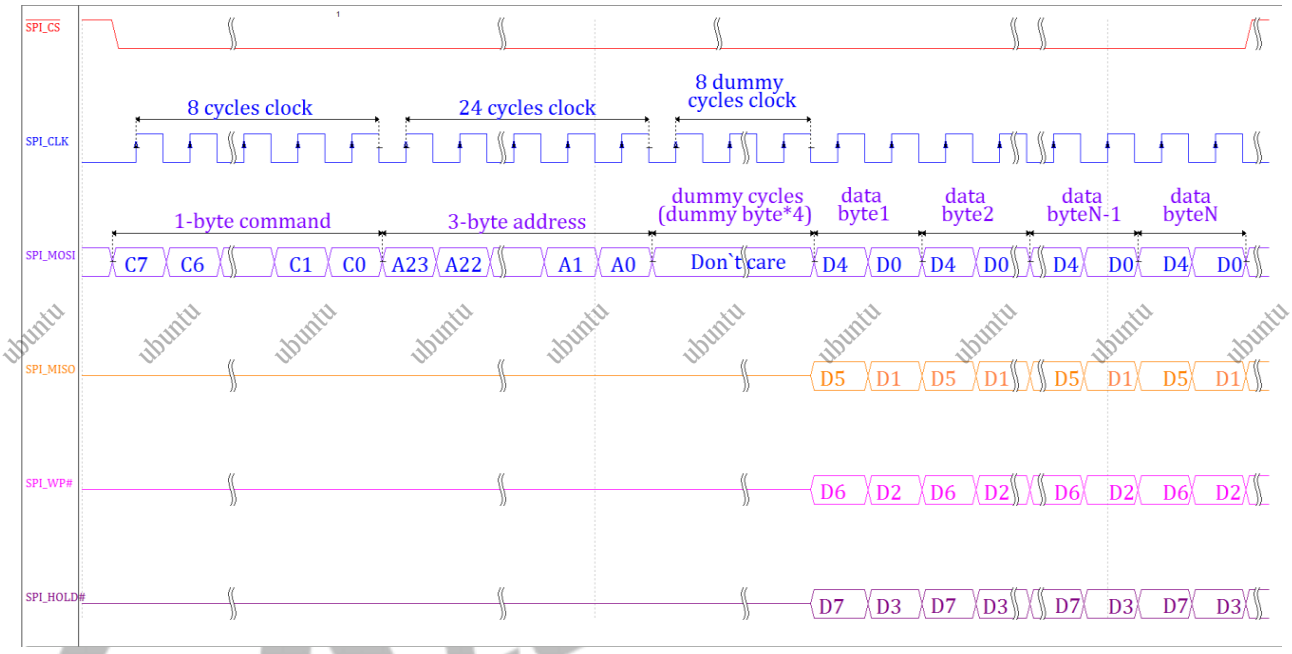
In the dual I/O SPI mode, only the command bytes output in a unit of a single bit in serial mode through the SPI_MOSI line. The address bytes and the dummy bytes output in a unit of dual bits through the SPI_MOSI and SPI_MISO. And the data bytes output (write) and input (read) in a unit of dual bits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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.4.3.8 SPI Quad-Input/Quad-Output Mode

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

Figure 7-35 SPI Quad-Input/Quad-Output Mode



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

7.4.3.9 Transmission/Reception Bursts in Master Mode

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.4.3.10 SPI Sample Mode and Run Clock Configuration

The SPI controller runs at 3 kHz~100 MHz at its interface to external SPI devices. The internal SPI clock should run at the same frequency as the outgoing clock in the master mode. The SPI clock is selected from different clock sources, the SPI must configure different work mode. There are three work modes: normal sample mode, delay half-cycle sample mode, delay one-cycle sample mode. Delay half-cycle sample mode is the default mode of the SPI controller. When the SPI runs at 40 MHz or below 40 MHz, the SPI can work at normal sample mode or delay half-cycle sample mode. When the SPI runs over 80 MHz, setting the **SDC** bit in the SPI Transfer Control Register to '1' makes the internal read sample point with a half-cycle delay of SPI_CLK, which is used in high speed read operation to reduce the error caused by the time delay of SPI_CLK between master and slave. Table 7-19 and Table 7-20 show the different configurations of the SPI sample mode.

Table 7-19 SPI Old Sample Mode and Run Clock

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

CAUTION

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

Table 7-20 SPI New Sample Mode

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.4.3.11 DBI 3-Line Interface Writing and Reading Timing

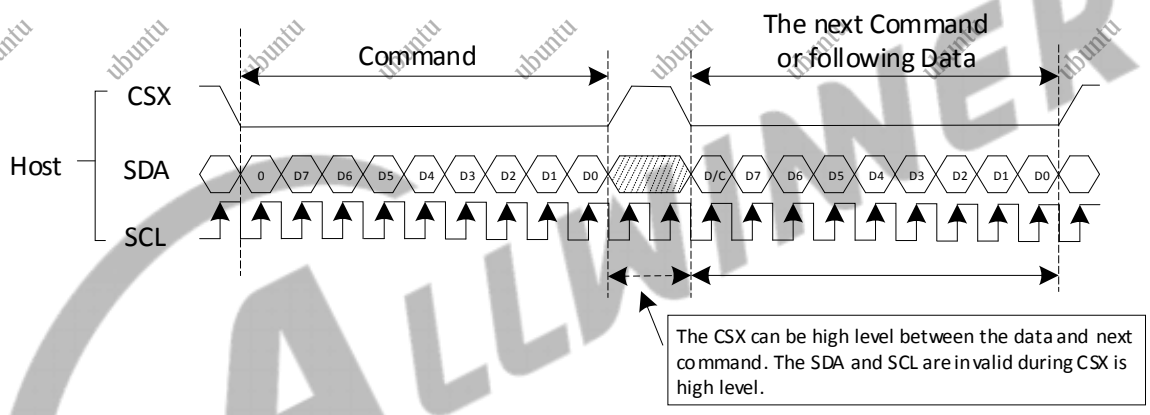
The 3-line DBI Interface I contains CSX, SDA, and SCL, where SDA shares this port for bidirectional port data input and output.

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

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

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

Figure 7-36 DBI 3-Line Display Bus Serial Interface Writing Operation Format

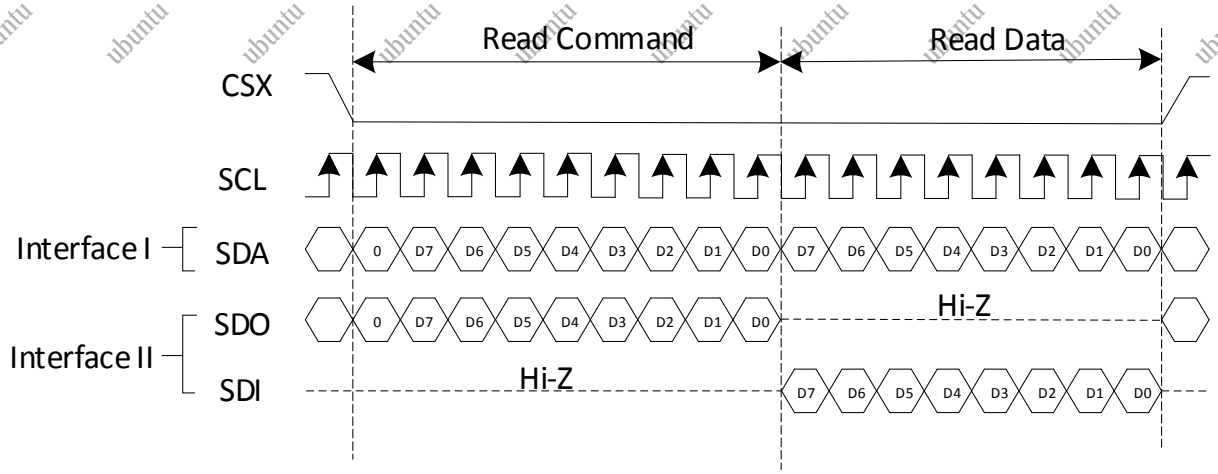


The 3-line DBI Interface I uses the SDA port as bidirectional data input and output port. There are only three cases of data reading volume, 8bits/24bits/32bits, and the first data sampled is high.

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

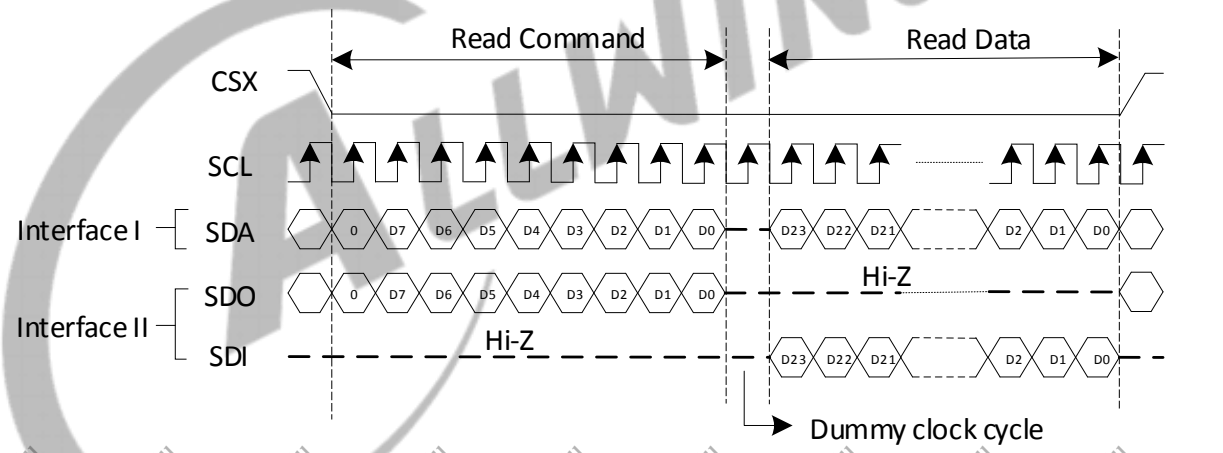
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-37 DBI 3-Line Display Bus Serial Interface 8-bit Reading Operation Format



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

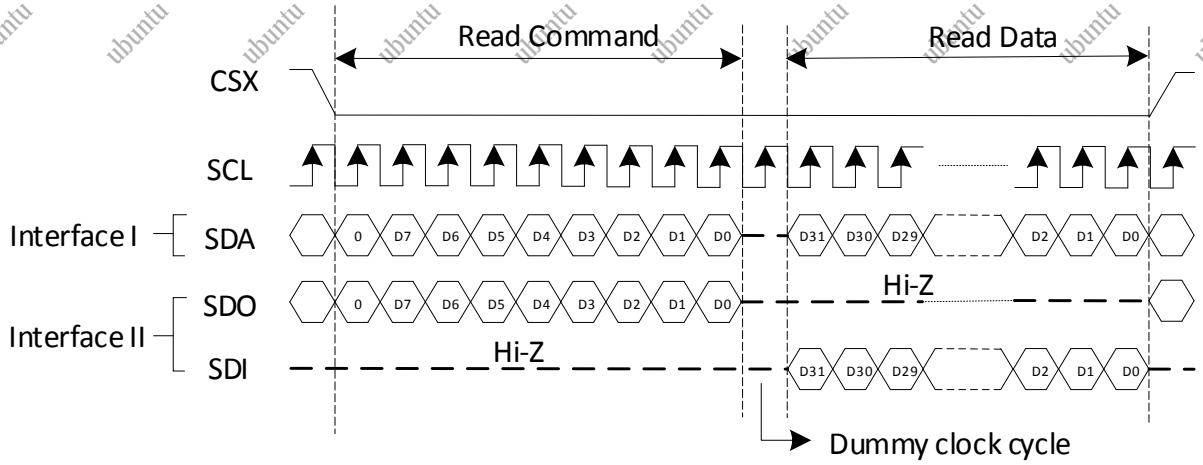
Figure 7-38 DBI 3-Line Display Bus Serial Interface 24-bit Reading Operation Format



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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-39 DBI 3-Line Display Bus Serial Interface 32-bit Reading Operation Format



7.4.3.12 DBI 4-Line Interface Writing and Reading Timing

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

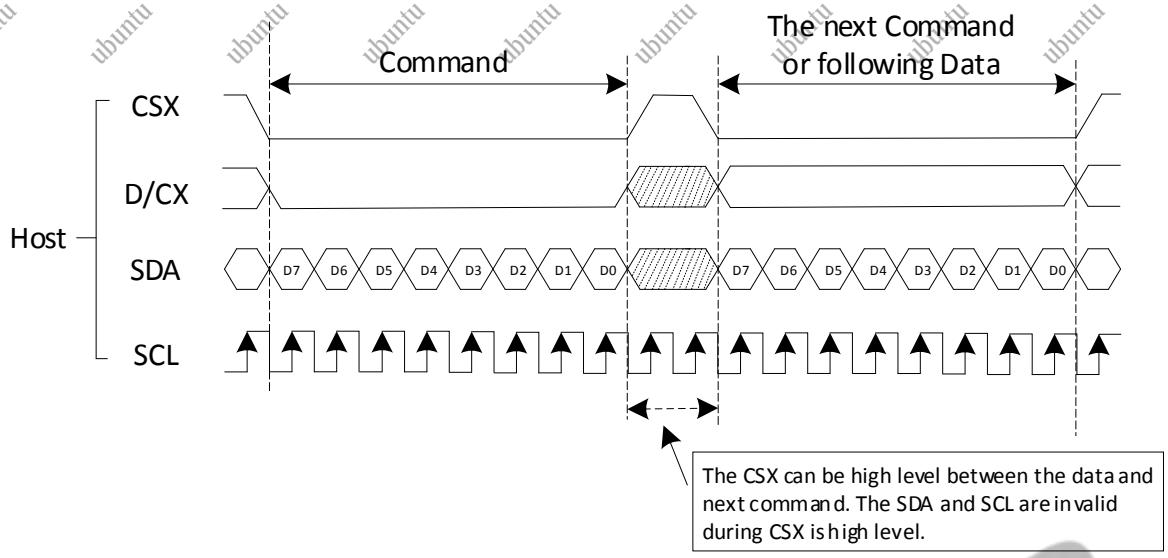
The 4-line DBI Interface II contains CSX, D/CX, SDA, SCL, and SDI; Data input and output ports are independent of each other.

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

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

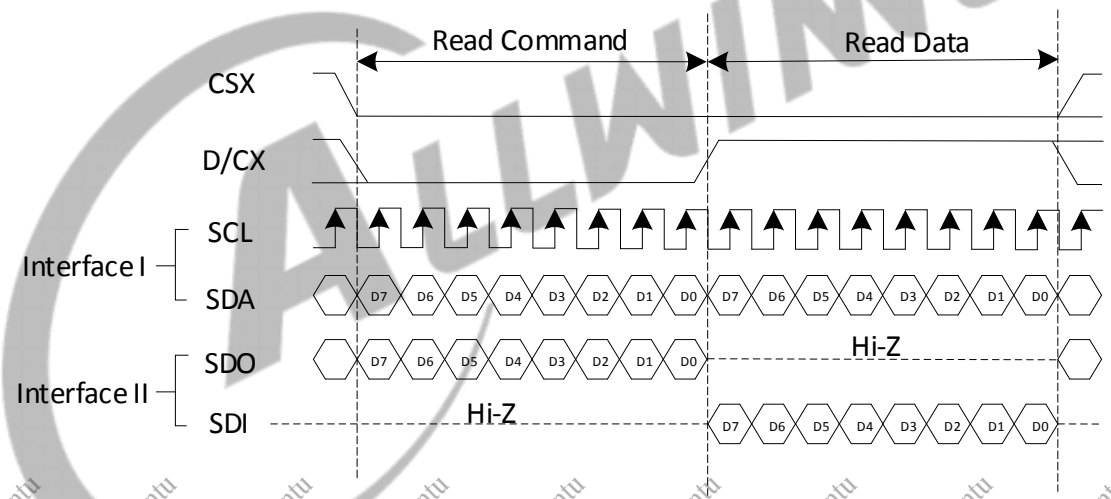
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-40 DBI 4-Line Display Bus Serial Interface Writing Operation Format



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

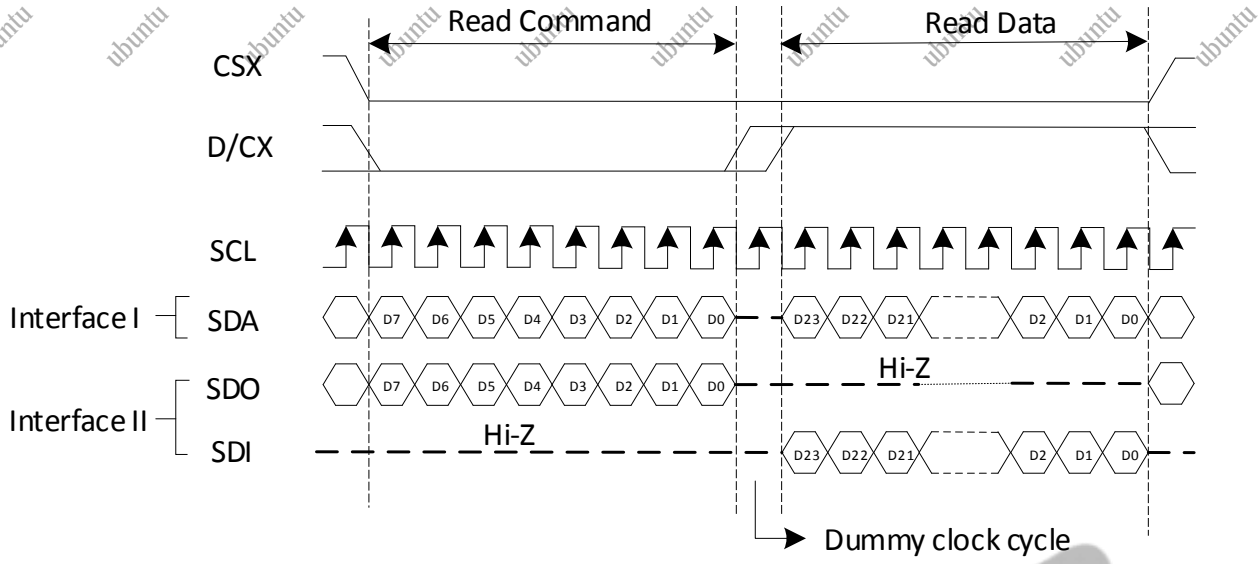
Figure 7-41 DBI 4-Line Display Bus Serial Interface 8-bit Reading Operation Format



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

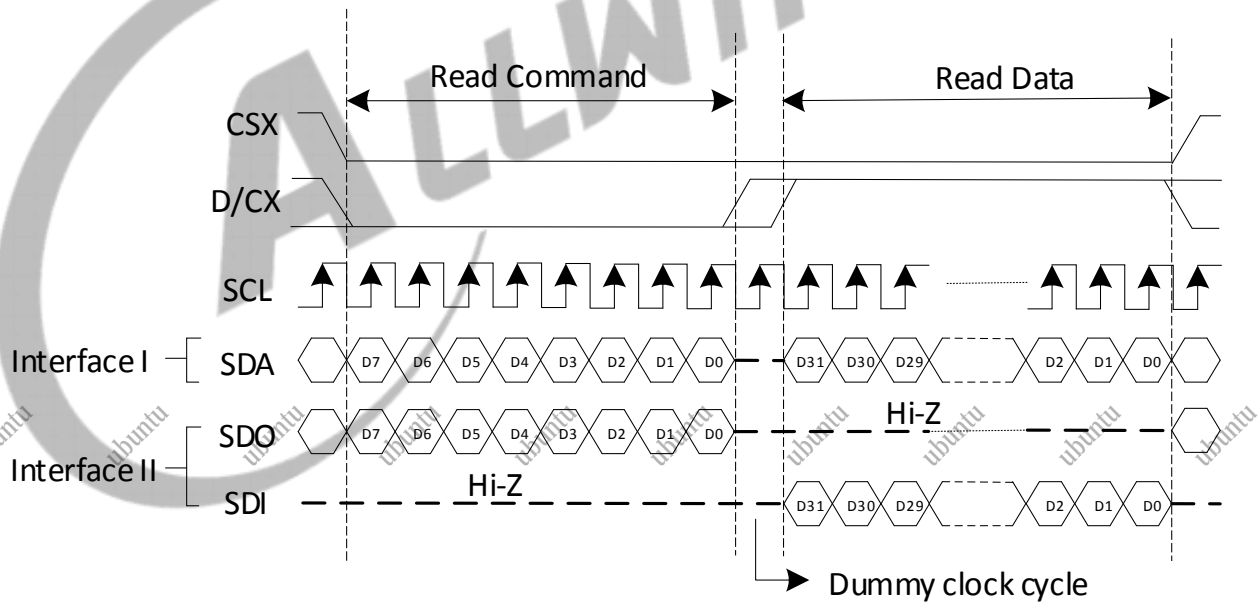
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-42 DBI 4-Line Display Bus Serial Interface 24-bit Reading Operation Format



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

Figure 7-43 DBI 4-Line Display Bus Serial Interface 32-bit Reading Operation Format



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.4.3.13 DBI 3-Line Interface Transmit Video Format

Figure 7-44 RGB111 3-Line Interface Transmit Video Format

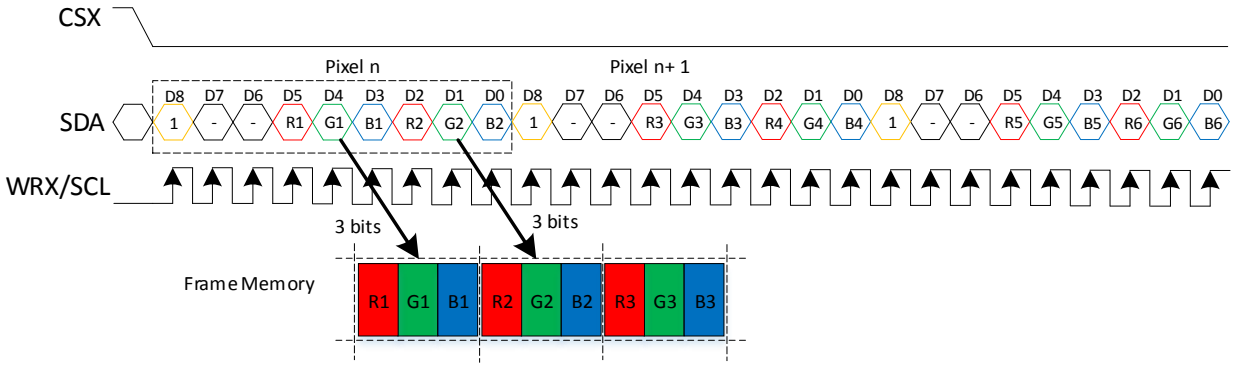
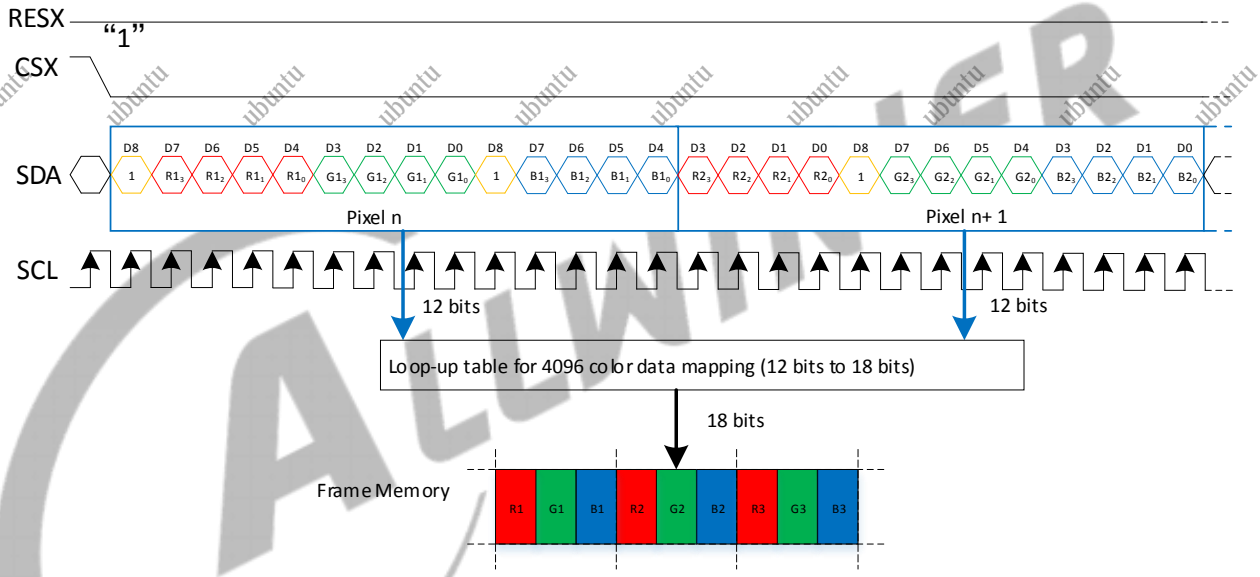


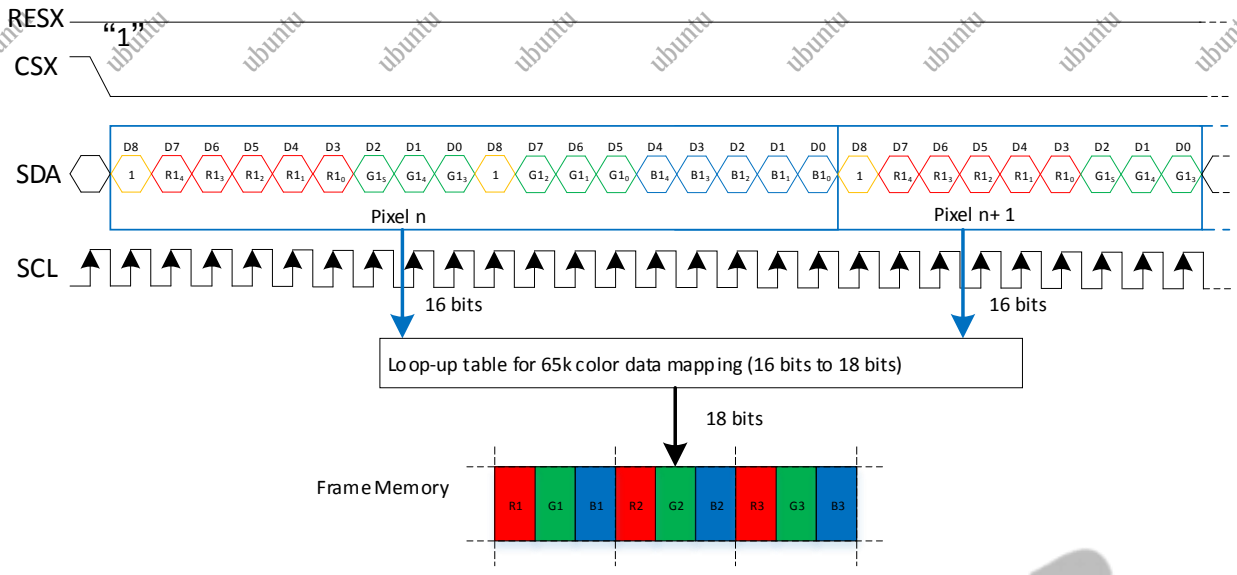
Figure 7-45 RGB444 3-Line Interface Transmit Video Format



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

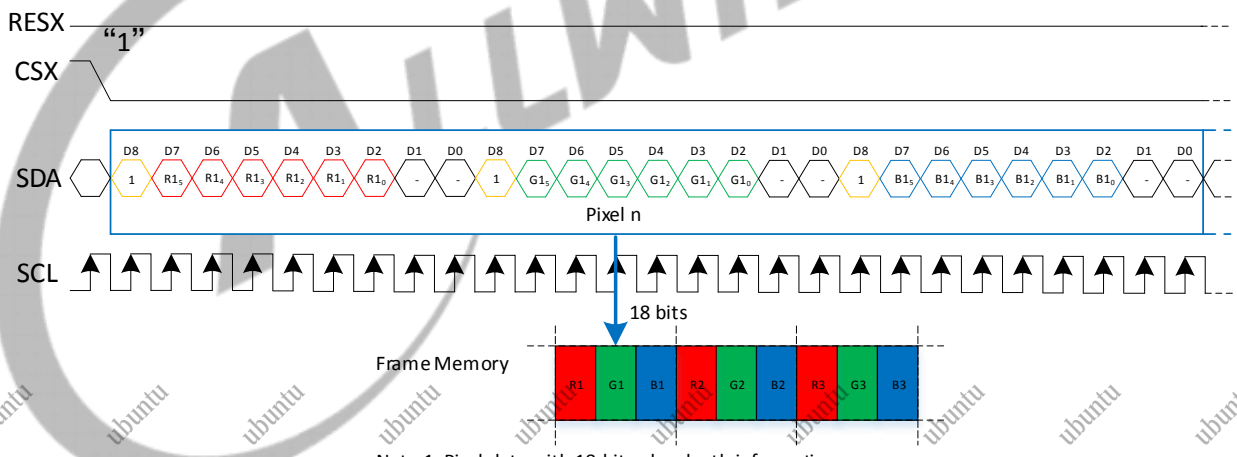
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-46 RGB565 3-Line Interface Transmit Video Format



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

Figure 7-47 RGB666 3-Line Interface Transmit Video Format



Note 1. Pixel data with 18-bit color depth information
Note 2. The most significant bits are: Rx5, Gx5 and Bx5
Note 3. The least significant bits are: Rx0, Gx0 and Bx0

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.4.3.14 DBI 4-Line Interface Transmit Video Format

Figure 7-48 RGB111 4-Line Interface Transmit Video Format

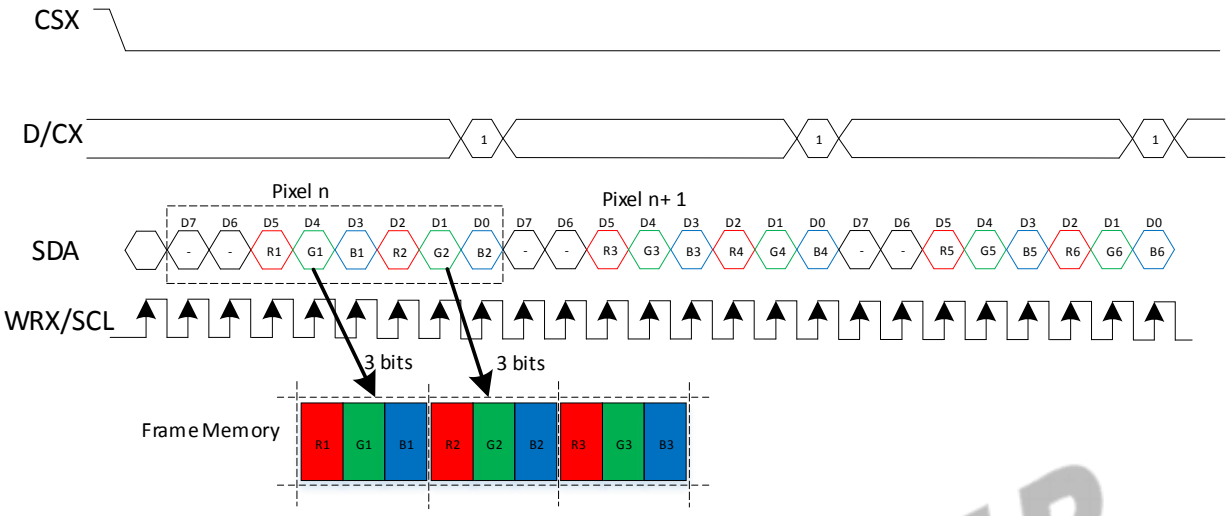
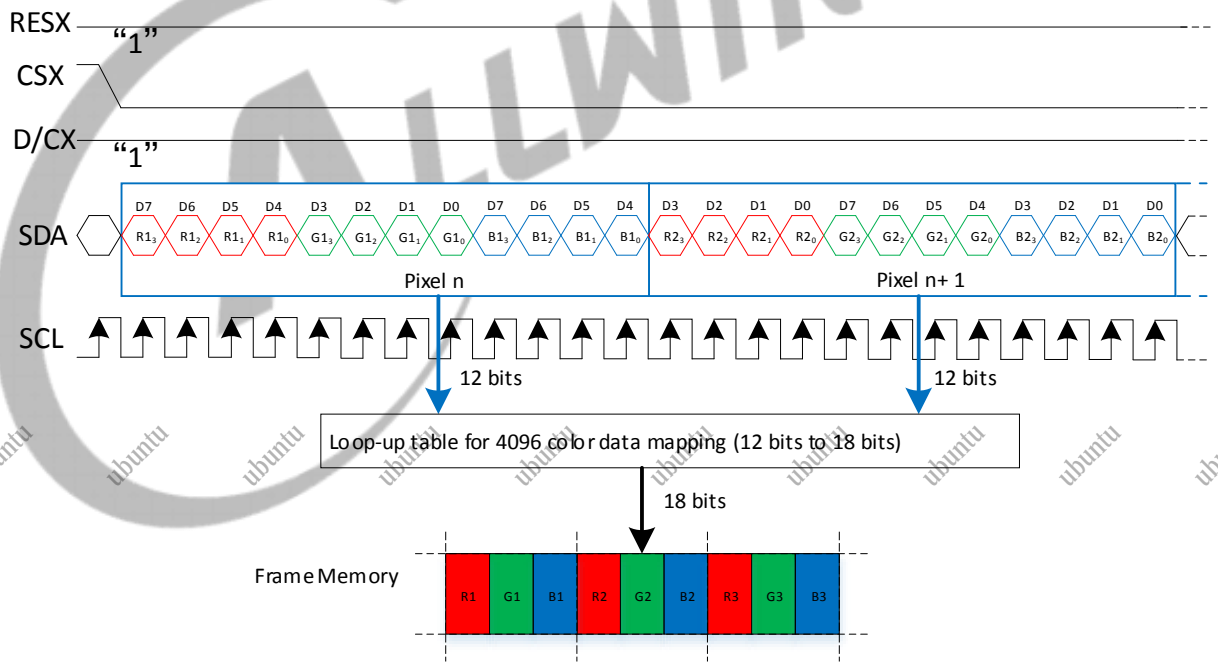


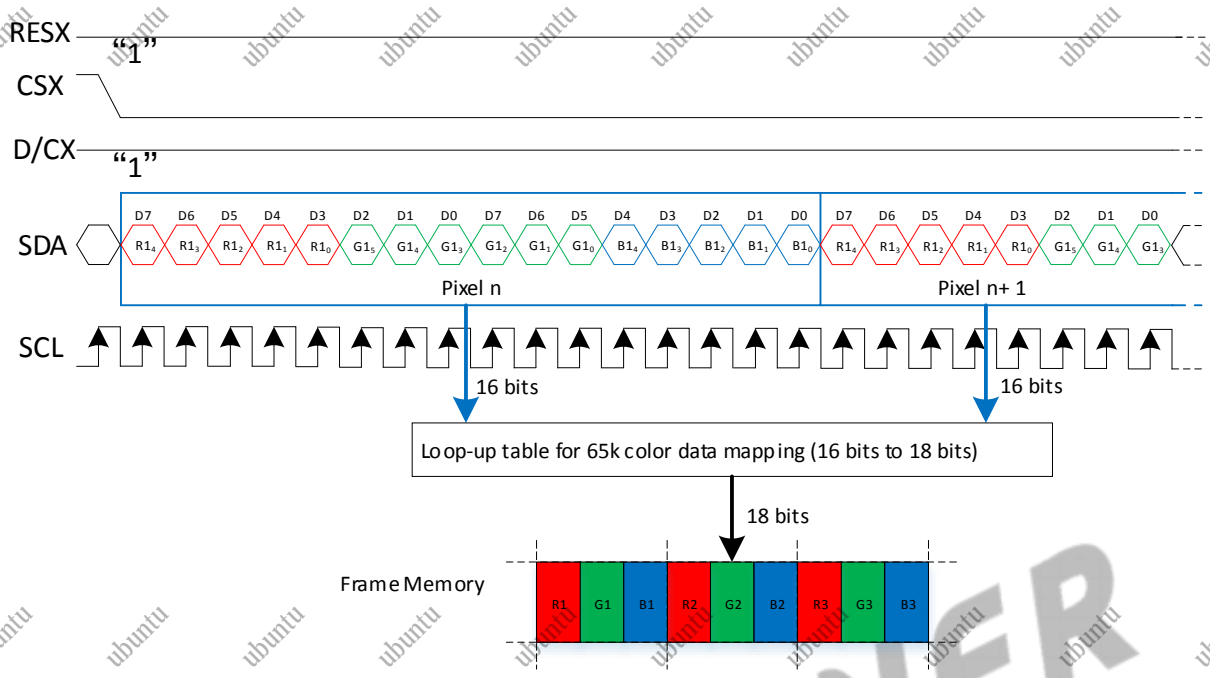
Figure 7-49 RGB444 4-Line Interface Transmit Video Format



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

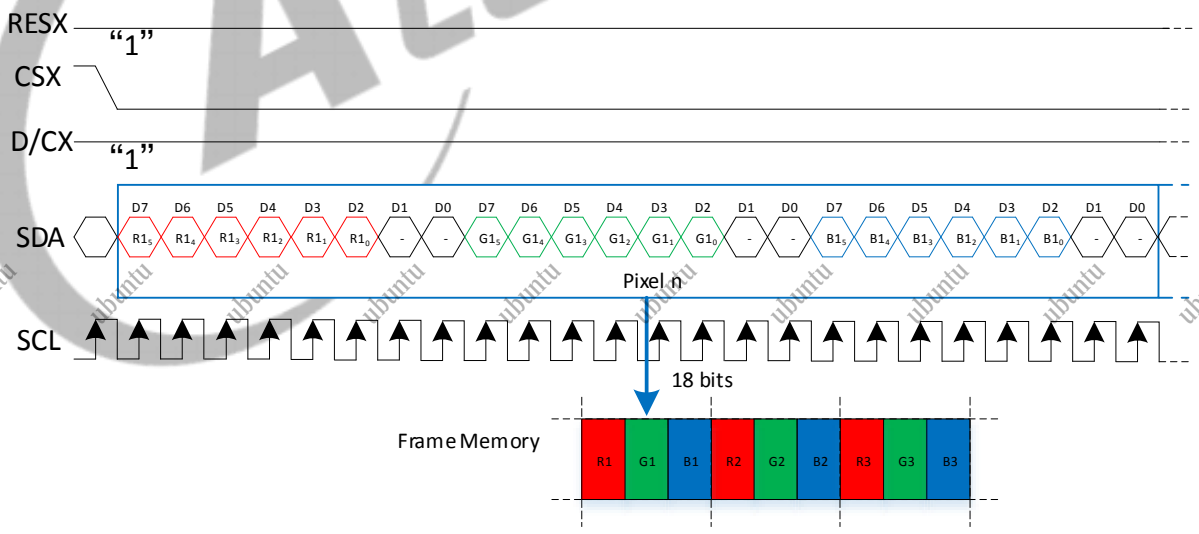
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-50 RGB565 4-Line Interface Transmit Video Format



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

Figure 7-51 RGB666 4-Line Interface Transmit Video Format



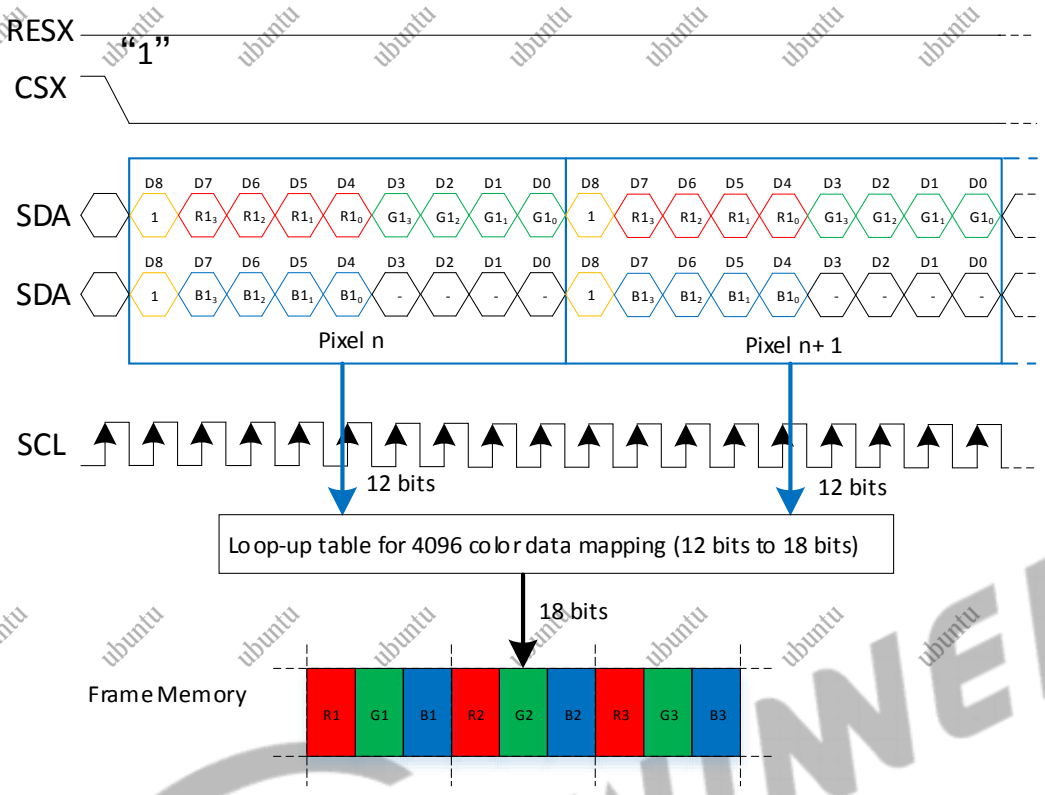
Note 1. Pixel data with 18-bit color depth information
Note 2. The most significant bits are: Rx5, Gx5 and Bx5
Note 3. The least significant bits are: Rx0, Gx0 and Bx0

7.4.3.15 DBI 2 Data Lane Interface Transmit Video Format

For RGB444:

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

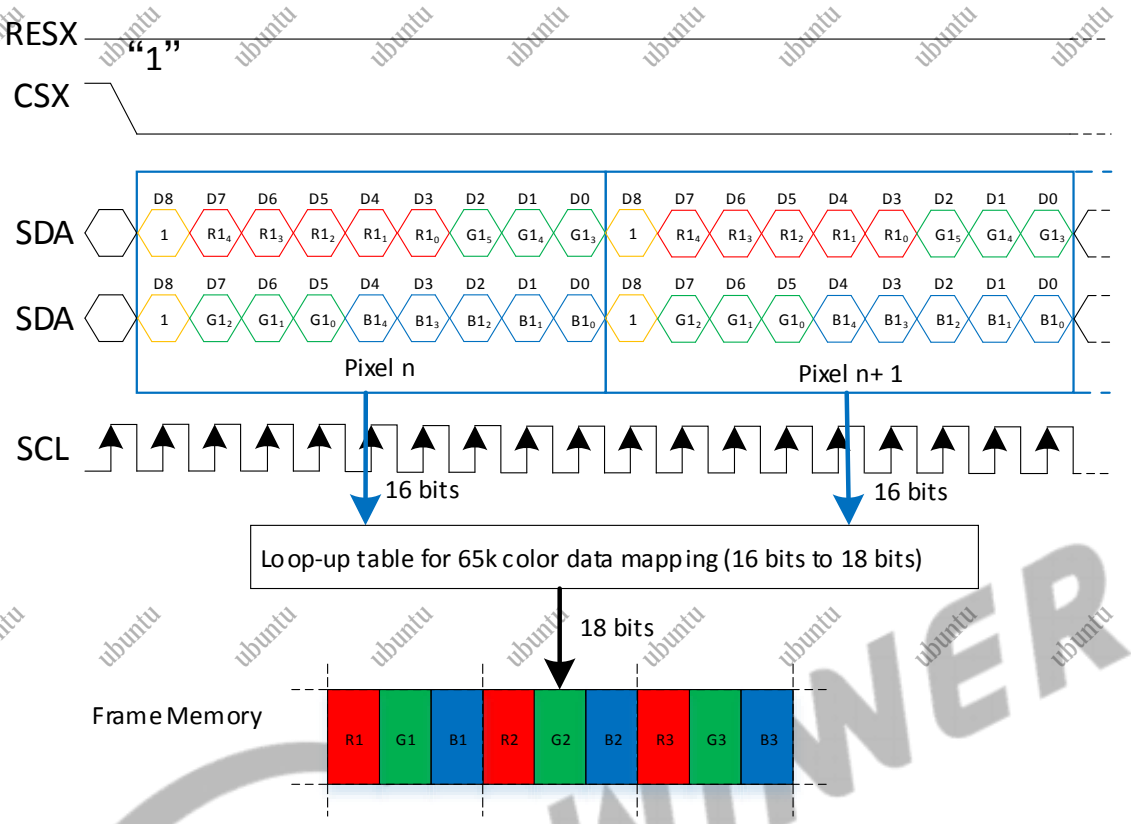
Figure 7-52 RGB444 2 Data Lane Interface Transmit Video Format



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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

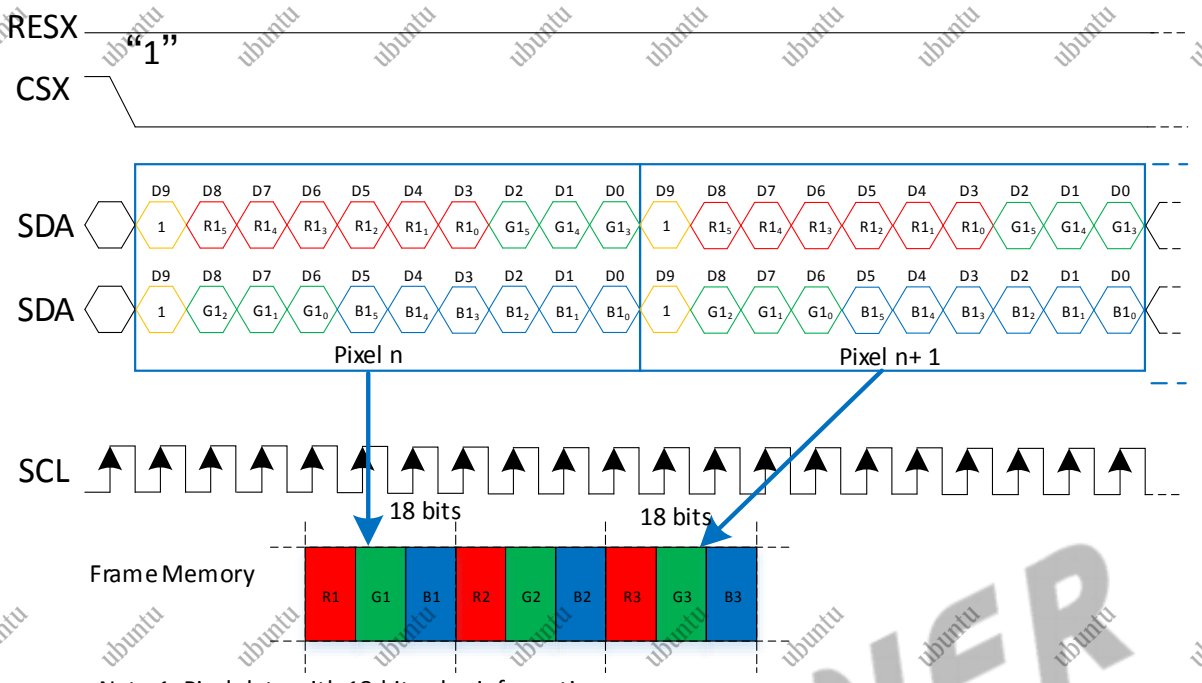
Figure 7-53 RGB565 2 Data Lane Interface Transmit Video Format



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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-54 RGB666 2 Data Lane Interface Transmit Video Format 0



Note 1. Pixel data with 18-bit color information
Note 2. The most significant bits are: Rx5, Gx5 and Bx5
Note 3. The least significant bits are: Rx0, Gx0 and Bx0

Figure 7-55 RGB666 2 Data Lane Interface Transmit Video Format 1 (ilitek)

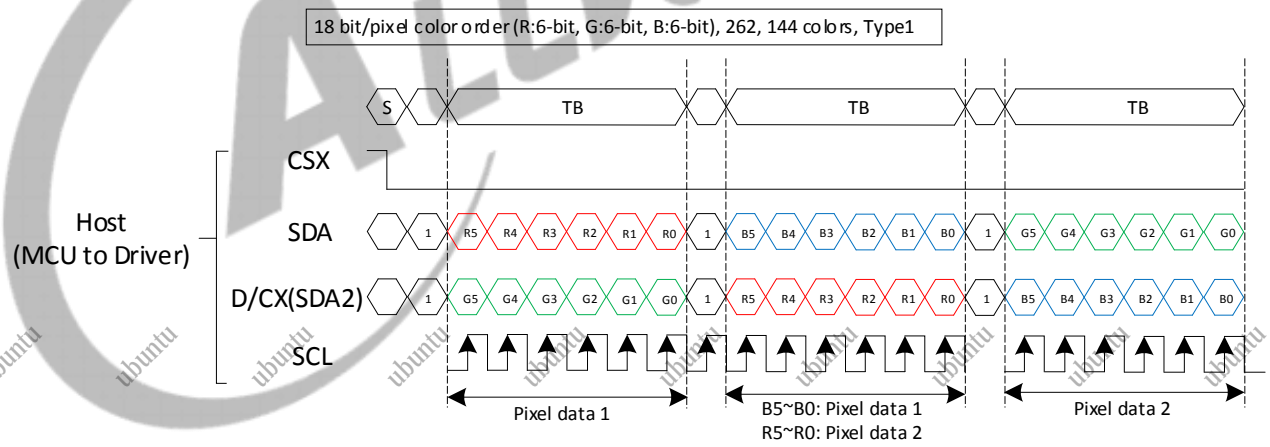
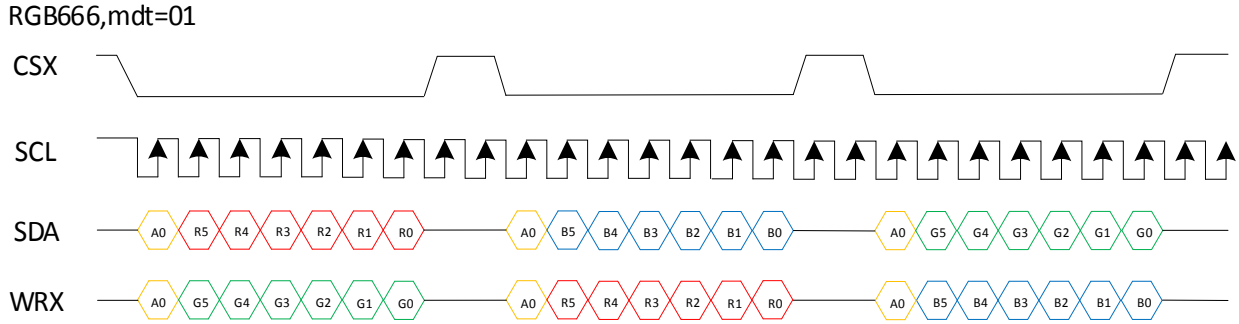
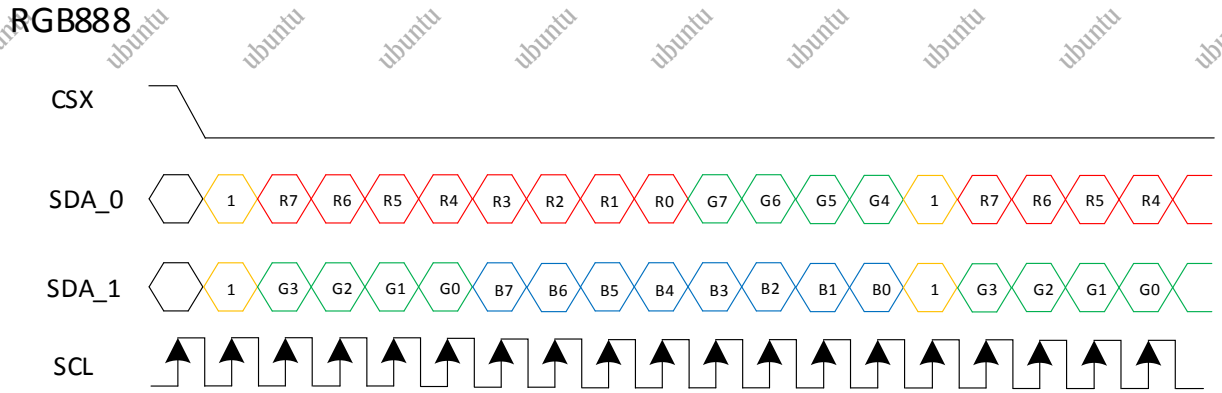


Figure 7-56 RGB666 2 Data Lane Interface Transmit Video Format 2 (New vision)



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-57 RGB888 2 Data Lane Interface Transmit Video Format



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

7.4.4 Programming Guidelines

7.4.4.1 Writing/Reading Data Process by Using SPI Mode

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

Write Data: CPU or DMA must write data on the **SPI_TXD** register, the data on the register are automatically moved to TX FIFO.

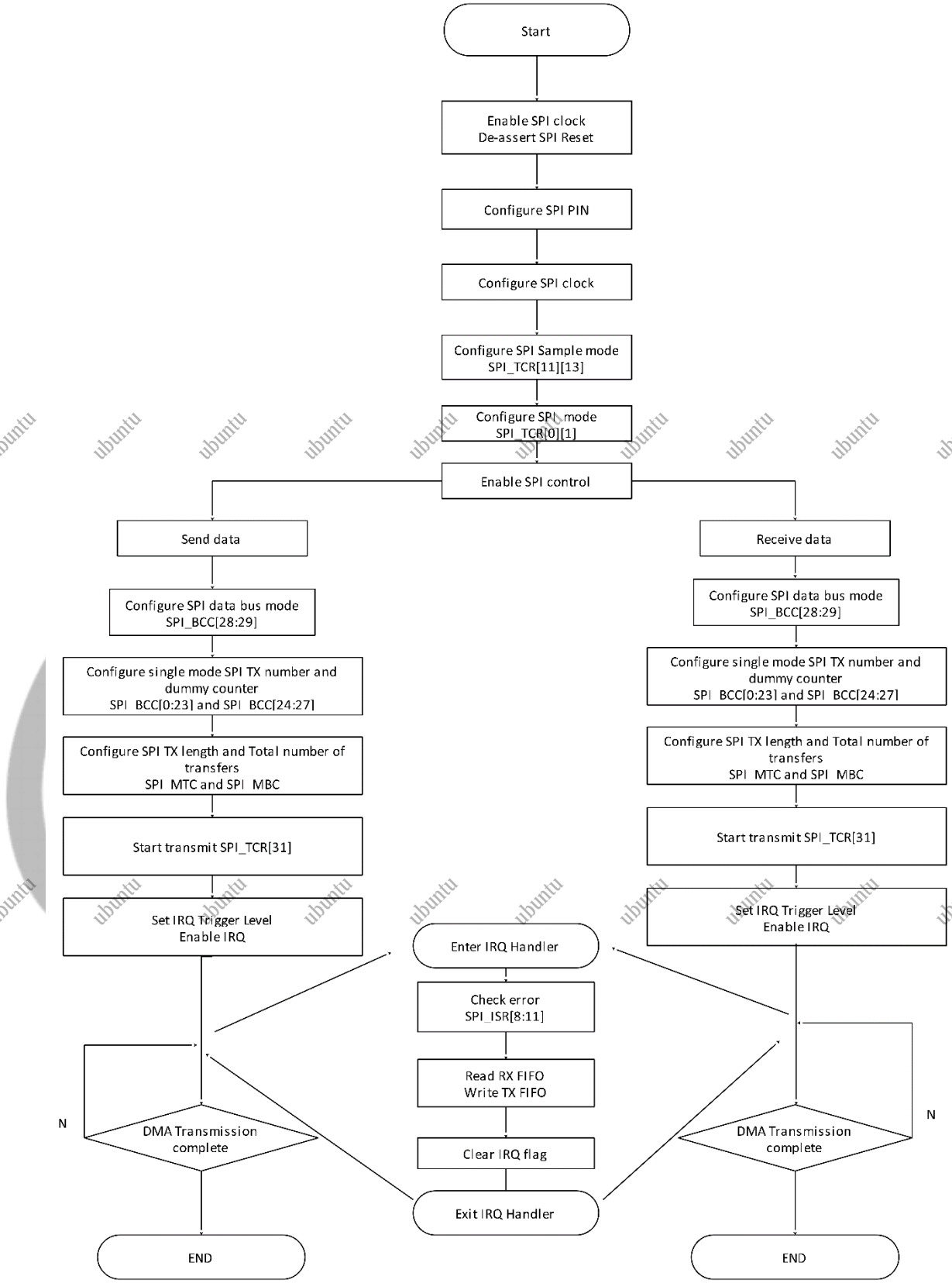
Read Data: To read data from RX FIFO, CPU or DMA must access the register **SPI_RXD** and data are automatically sent to the register **SPI_RXD**.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

CPU Mode

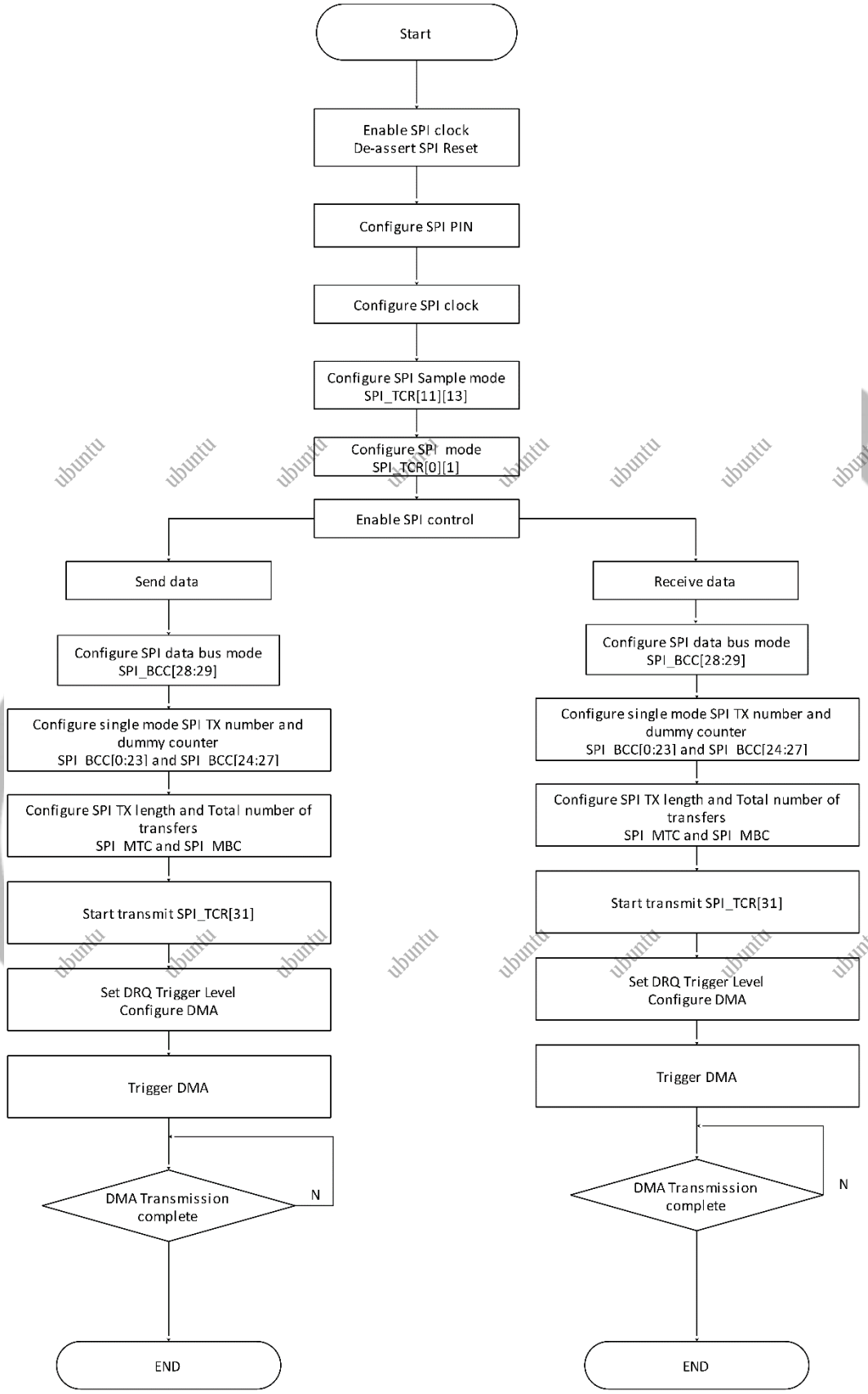
Figure 7-58 SPI Write/Read Data in CPU Mode



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

DMA Mode

Figure 7-59 SPI Write/Read Data in DMA Mode



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.4.4.2 Calibrating Delay Chain by Using SPI Mode

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

The steps to calibrate delay chain are as follows:

Step 1 Enable SPI. To calibrate the delay chain by operation registers in SPI, the SPI must be enabled through AHB reset and AHB clock gating control registers.

Step 2 Configure a proper clock for SPI. The calibration delay chain is based on the clock for SPI from CCU.

Step 3 Set proper initial delay value. Write 0xA0 to the SPI Sample Delay Control Register to set initial delay value 0x20 to delay chain. Then write 0x0 to the SPI Sample Delay Control Register to clear this value.

Step 4 Write 0x8000 to the SPI Sample Delay Control Register to start to calibrate the delay chain.

Step 5 Wait until the flag (Bit14 in the SPI Sample Delay Control Register) of calibration done is set. The number of delay cells is shown at Bit[13:8] in SPI Sample Delay Control Register. The delay time generated by these delay cells is equal to the cycle of SPI's clock nearly. This value is the result of calibration.

Step 6 Calculate the delay time of one delay cell according to the cycle of the SPI clock and the result of calibration.

7.4.4.3 Transmitting Write Command by Using DBI Mode

Step 1 Set the SPI_DBI_MODE_SEL (bit3) of **SPI_GCR** (0x0004) to 1 to select DBI mode.

Step 2 Set the DBI EN MODE SEL (bit[30:29]) of **DBI_CTL_1** (0x0104) to 0 to select the trigger mode of DBI.

Step 3 Configure the **DBI_CTL_0** (0x0100).

- Set **DBI_CTL_0**[Command Type] (bit31) to 0 to configure the writing command.
- Set **DBI_CTL_0**[Write Command Dummy Cycles] (bit[30:20]) to configure the number of dummy cycles between commands.
- Set **DBI_CTL_0**[Output Data Sequence] (bit19) to select the MSB or LSB.
- Set **DBI_CTL_0**[Transmit Mode] (bit15) to 0 to select the command path.
- Set **DBI_CTL_0**[Output Data Format] (bit[14:12]) to 0 to transmit the command.
- Set **DBI_CTL_0**[DBI interface Select] (bit[10:8]) to select the DBI interface type.
- The remaining values of the **DBI_CTL_0** register remain the default value.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 4 Set **DBI_CTL_1**[DCX_DATA] (bit22) to 0 to send the command.

Step 5 DMA Path: Configure the **SPI_FCR** register (0x0018).

- Set **SPI_FCR**[TF_DRQ_EN] (bit24) to 1 to enable TXFIFO DMA.
- Set **SPI_FCR**[TX_TRIG_LEVEL] (bit[23:16]) to 255. It indicates the controller requests data from DMA if the remaining space of TX FIFO is greater than 255.

CPU Path: Write the command to be sent to the 0x200 address.

Step 6 Set **SPI_GCR**[DBI_EN] (bit4) to 1 to start transmitting the command.

Step 7 Wait until the TX FIFO underrun interrupt (**SPI_ISR**[TF_UDF]) is 1. It indicates that the command written to the TX FIFO is transmitted completely.

7.4.4.4 Transmitting Parameter by Using DBI Mode

Step 1 Set the SPI_DBI_MODE_SEL (bit3) of **SPI_GCR** (0x0004) to 1 to select DBI mode.

Step 2 Set the DBI_EN_MODE_SEL (bit[30:29]) of **DBI_CTL_1** (0x0104) to 0 to select the trigger mode of DBI.

Step 3 Configure the **DBI_CTL_0** register (0x0100).

- Set **DBI_CTL_0**[Command Type] (bit31) to 0 to configure the writing command.
- Set **DBI_CTL_0**[Write Command Dummy Cycles] (bit[30:20]) to configure the number of dummy cycles between commands.
- Set **DBI_CTL_0**[Output Data Sequence] (bit19) to select the MSB or LSB.
- Set **DBI_CTL_0**[Transmit Mode] (bit15) to 0 to select the command path.
- Set **DBI_CTL_0**[Output Data Format] (bit[14:12]) to 0 to transmit the command.
- Set **DBI_CTL_0**[DBI interface Select] (bit[10:8]) to select the DBI interface type.
- The remaining values of the **DBI_CTL_0** register remain the default value.

Step 4 Set **DBI_CTL_1**[DCX_DATA] (bit22) to 1 to send the parameter.

Step 5 DMA Path: Configure the register (0x0018).

- Set **SPI_FCR**[TF_DRQ_EN] (bit24) to 1 to enable TXFIFO DMA.
- Set **SPI_FCR**[TX_TRIG_LEVEL] (bit[23:16]) to 255. It indicates the controller requests data from DMA if the remaining space of TX FIFO is greater than 255.

CPU Path: Write the command to be sent to the 0x0200 address.

Step 6 Set **SPI_GCR**[DBI_EN] (bit4) to 1 to start transmitting the command.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 7 Wait until the TX FIFO underrun interrupt (**SPI_ISR**[TF_UDF]) is 1. It indicates that the command written to the TX FIFO is transmitted completely.

7.4.4.5 Transmitting Video by Using DBI Mode

Set the SPI_DBI_MODE_SEL (bit3) of **SPI_GCR** (0x0004) to 1 to select DBI mode.

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

If the data is from the DMA path, configure **DBI_CTL_1**[DBI_FIFO_DRQ_EN] (bit15) to 1 and **DBI_CTL_1**[TX_TRIG_LEVEL] (bit[14:8]) to 64, which indicates the controller requests data from DMA if the remaining space of TX FIFO is greater than 64.

Software Trigger Mode

The software enables DBI_en_trigger when the edge interrupt of TE is detected.

After transmitting each frame image, the controller clears automatically the line_cnt, pixel_cnt and stops transmitting data.

Wait for the edge interrupt of TE, the software needs to enable DBI_en_trigger, in circulation.

The operation process is as follows.

Step 1 Set the SPI_DBI_MODE_SEL (bit3) of **SPI_GCR** (0x0004) to 1 to select DBI mode.

Step 2 Set the DBI EN MODE SEL (bit[30:29]) of **DBI_CTL_1** (0x0104) to 1 to select the software trigger mode.

Step 3 Configure the **DBI_CTL_0** register (0x0100).

- Set **DBI_CTL_0**[Command Type] (bit31) to 0 to set the writing command.
- Set **DBI_CTL_0**[Write Command Dummy Cycles] (bit[30:20]) to configure the number of dummy cycles between commands.
- Set **DBI_CTL_0**[Output Data Sequence] (bit19) to select the MSB or LSB.
- Set **DBI_CTL_0**[Transmit Mode] (bit15) to 1 to select the image path.
- Set **DBI_CTL_0**[Output Data Format] (bit[14:12]) to select RGB111//444/565/666/888.
- Set **DBI_CTL_0**[DBI interface Select] (bit[10:8]) to select the DBI interface type.
- The remaining values of the **DBI_CTL_0** register remain the default value.

Step 4 Set **DBI_CTL_1**[DCX_DATA] (bit22) to 0 to send the image data.

Step 5 Configure **DBI_Video_Size** (0x110) according to the sent image size.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 6 Configure **DBI_CTL_2** (0x0108) to set the TE-related parameter.

Step 7 Detect the TE interrupt of the **DBI_INT** (0x0120) register.

Step 8 Configure **DBI_CTL_1**[DBI_soft_trigger] to 1.

Timer Trigger Mode

The software configures timer_en to enable timer counting, and when the counter reaches the specified value, the DBI_EN automatically can be enabled to start transmitting data.

After transmitting each frame image, the controller clears automatically the line_cnt, pixel_cnt, and stops transmitting data.

The timer starts counting again. When the counter reaches the specified value, the controller automatically enables DBI_EN, and in circulation until the software turns off the timer_en.

The operation process is as follows.

Step 1 Set the SPI_DBI_MODE_SEL (bit3) of **SPI_GCR** (0x0004) to 1 to select DBI mode.

Step 2 Set the DBI EN MODE SEL (bit30:29) of **DBI_CTL_1** (0x0104) to 2 to select the timer trigger mode.

Step 3 Configure the **DBI_CTL_0** register (0x0100).

- Set **DBI_CTL_0**[Command Type] (bit31) to 0 to set the writing command.
- Set **DBI_CTL_0**[Write Command Dummy Cycles] (bit[30:20]) to configure the number of dummy cycles between commands.
- Set **DBI_CTL_0**[Output Data Sequence] (bit19) to select the MSB or LSB.
- Set **DBI_CTL_0**[Transmit Mode] (bit15) to 1 to select the image path.
- Set **DBI_CTL_0**[Output Data Format] (bit[14:12]) to select RGB111/444/565/666/888.
- Set **DBI_CTL_0**[DBI interface Select] (bit[10:8]) to select the DBI interface type.
- The remaining values of the **DBI_CTL_0** register remain the default value.

Step 4 Set **DBI_CTL_1**[DCX_DATA] (bit22) to 0 to send the image data.

Step 5 Configure **DBI_Video_Size** (0x110) to transmit the image size.

Step 6 Configure the related parameter of **DBI_Timer** (0x10C).

TE Trigger Mode

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

After transmitting each frame image, the controller clears automatically the line_cnt, pixel_cnt, and stops transmitting data.

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

The operation process is as follows.

Step 1 Set the SPI_DBI_MODE_SEL (bit3) of **SPI_GCR** (0x0004) to 1 to select DBI mode.

Step 2 Set the DBI EN MODE SEL (bit30:29) of **DBI_CTL_1** (0x0104) to 3 to select the TE trigger mode.

Step 3 Configure the **DBI_CTL_0** register (0x0100).

- Set **DBI_CTL_0**[Command Type] (bit31) to 0 to set the writing command.
- Set **DBI_CTL_0**[Write Command Dummy Cycles] (bit[30:20]) to configure the number of dummy cycles between commands.
- Set **DBI_CTL_0**[Output Data Sequence] (bit19) to select the MSB or LSB.
- Set **DBI_CTL_0**[Transmit Mode] (bit15) to 1 to select the image path.
- Set **DBI_CTL_0**[Output Data Format] (bit[14:12]) to select RGB111/444/565/666/888.
- Set **DBI_CTL_0**[DBI interface Select] (bit[10:8]) to select the DBI interface type.
- The remaining values of the **DBI_CTL_0** register remain the default value.

Step 4 Configure **DBI_CTL_1**[DCX_DATA] (bit22) to 0 to send the image data.

Step 5 Configure **DBI_Video_Size** (0x0110) to transmit the image size.

Step 6 Configure **DBI_CTL_2** (0x0108) to set the TE-related parameter.

7.4.4.6 Transmitting Read Command and Read Data by Using DBI Mode

Step 1 Set the SPI_DBI_MODE_SEL (bit3) of **SPI_GCR** (0x0004) to 1 to select DBI mode.

Step 2 Set the DBI EN MODE SEL (bit[30:29]) of **DBI_CTL_1** (0x0104) to 0.

Step 3 Configure the **DBI_CTL_0** register (0x0100).

- Set **DBI_CTL_0**[Command Type] (bit31) to 0 to set the reading command.
- Set **DBI_CTL_0**[Output Data Sequence] (bit19) to select the MSB or LSB.
- Set **DBI_CTL_0**[Transmit Mode] (bit15) to 0 to select the command path.
- Set **DBI_CTL_0**[Output Data Format] (bit[14:12]) to 0.
- Set **DBI_CTL_0**[DBI interface Select] (bit[10:8]) to select the DBI interface type.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

- The remaining values of the DBI_CTL_0 register remain the default value.

Step 4 Configure the DBI_CTL_1 register (0x0104).

- Configure DBI_CTL_1[DCX_DATA] (bit22) to 0 to send the command.
- Configure DBI_CTL_1[Read_MSB_First] (bit20) to select whether the first bit of the read data is the highest or lowest bit of data.
- Configure DBI_CTL_1[Read Data Number of Bytes] to set the byte number to be read.
- Configure DBI_CTL_1[Read Command Dummy Cycles] to set the dummy cycle between the read command and the read data, when the dummy cycle is complete, the data starts to be sampled.

Step 5 DMA Path: Configure the SPI_FCR register (0x0018).

- Set SPI_FCR[RF_DRQ_EN] (bit8) to 1 to enable RXFIFO DMA.
- Set SPI_FCR[RX_TRIG_LEVEL] (bit[7:0]) to 32, which indicates the controller requests receiving data from DMA if the data of the RX FIFO is greater than 64.

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

Step 6 Set SPI_GCR[DBI_EN] (bit4) to 1 to start transmitting command.

Step 7 Wait until DBI_INT[RD_DONE_INT] is 1. It indicates that the data is read completely.

7.4.5 Register List

Module Name	Base Address
SPI_DBI	0x04026000

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.4.6 Register Description

7.4.6.1 0x0004 SPI Global Control Register (Default Value: 0x0000_0080)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.4.6.2 0x0008 SPI Transfer Control Register (Default Value: 0x0000_0087)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.4.6.3 0x0010 SPI Interrupt Control Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.4.6.4 0x0014 SPI Interrupt Status Register (Default Value: 0x0000_0032)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.4.6.5 0x0018 SPI FIFO Control Register (Default Value: 0x0040_0001)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.4.6.6 0x001C SPI FIFO Status Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.4.6.7 0x0020 SPI Wait Clock Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.4.6.8 0x0028 SPI Sample Delay Control Register (Default Value: 0x0000_2000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.4.6.9 0x0030 SPI Master Burst Counter Register (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: SPI_MBC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	/	/	/
23:0	R/W	0x0	MBC Master Burst Counter In master mode, this field specifies the total burst number which includes the TXD, RXD, and dummy burst. 0: 0 burst 1: 1 burst … N: N bursts Cannot be written when XCH=1.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.4.6.10 0x0034 SPI Master Transmit Counter Register (Default Value: 0x0000_0000)

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

7.4.6.11 0x0038 SPI Master Burst Control Counter Register (Default Value: 0x0000_0000)

Offset: 0x0038			Register Name: SPI_BCC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/
29	R/W	0x0	Quad_EN Quad_Mode_EN The Quad mode includes Quad-Input and Quad-Output. 0: Quad mode disabled 1: Quad mode enabled Cannot be written when XCH=1.
28	R/W	0x0	DRM Master Dual Mode RX Enable It is only valid when *Quad_Mode_EN*=0. 0: RX uses the single-bit mode 1: RX uses the dual mode Cannot be written when XCH=1.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0038			Register Name: SPI_BCC
Bit	Read/Write	Default/Hex	Description
27:24	R/W	0x0	DBC Master Dummy Burst Counter In master mode, this field specifies the burst number that should be sent before receiving in dual SPI mode. The data does not care by the device. 0: 0 burst 1: 1 burst … N: N bursts Cannot be written when XCH=1
23:0	R/W	0x0	STC Master Single Mode Transmit Counter In master mode, this field specifies the burst number that should be sent in the single mode before automatically sending dummy bursts. This is the first transmit counter in all bursts. 0: 0 burst 1: 1 burst … N: N bursts Cannot be written when XCH=1

7.4.6.12 0x0040 SPI Bit-Aligned Transfer Configure Register (Default Value: 0x0000_00A0)

Offset: 0x0040			Register Name: SPI_BATC
Bit	Read/Write	Default/Hex	Description
31	R/WAC	0x0	TCE Transfer Control Enable In master mode, it is used to start to transfer the serial bits frame, it is only valid when *Work Mode Select*==0x10/0x11. 0: Idle 1: Initiates transfer Writing “1” to this bit will start to transfer serial bits frame (the value comes from the *SPI TX Bit Register* or *SPI RX Bit Register*), and will auto-clear after the bursts transfer completely. Writing ‘0’ to this bit has no effect.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0040			Register Name: SPI_BATC
Bit	**Read/Write**	**Default/Hex**	**Description**
30	R/W	0x0	MSMS Master Sample Standard 0: Delay Sample Mode 1: Standard Sample Mode In Standard Sample Mode, the SPI master samples the data at the standard rising edge of SCLK for each SPI mode; In Delay Sample Mode, the SPI master samples data at the edge that is half cycle delayed by the standard rising edge of SCLK defined in respective SPI mode.
29:26	/	/	/
25	R/W1C	0x0	TBC Transfer Bits Completed When set, this bit indicates that the last bit of the serial data frame in *SPI TX Bit Register* (or *SPI RX Bit Register*) has been transferred completely. Writing 1 to this bit clears it. 0: Busy 1: Transfer Completed It is only valid when *Work Mode Select*==0x10/0x11.
24	R/W	0x0	TBC_INT_EN Transfer Bits Completed Interrupt Enable 0: Disable 1: Enable It is only valid when *Work Mode Select*==0x10/0x11.
23:22	/	/	/
21:16	R/W	0x00	Configure the length of serial data frame (burst) of RX 000000: 0 bit 000001: 1 bit … 100000: 32 bits Other values: reserved It is only valid when *Work Mode Select*==0x10/0x11, and cannot be written when TCE (*SPI_BATC[31]*)=1.
15:14	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0040			Register Name: SPI_BATC
Bit	**Read/Write**	**Default/Hex**	**Description**
13:8	R/W	0x00	Configure the length of serial data frame (burst) of TX 000000: 0 bit 000001: 1 bit … 100000: 32 bits Other values: reserved It is only valid when *Work Mode Select*==0x10/0x11, and cannot be written when TCE=1.
7	R/W	0x1	SS_LEVEL When control SS signal manually, set this bit to '1' or '0' to control the level of SS signal. 0: Set SS to low 1: Set SS to high It is only valid when *Work Mode Select*==0x10/0x11, and only work in Mode0, cannot be written when TCE=1.
6	R/W	0x0	SS_OWNER SS Output Owner Select Usually, the controller sends the SS signal automatically with data together. When this bit is set to 1, the software must manually write SS_LEVEL (*SPI_BATC[7]*) to 1 or 0 to control the level of the SS signal. 0: SPI controller 1: Software It is only valid when *Work Mode Select*==0x10/0x11, and only work in Mode0, cannot be written when TCE=1.
5	R/W	0x1	SPOL SPI Chip Select Signal Polarity Control 0: Active high polarity (0 = Idle) 1: Active low polarity (1 = Idle) It is only valid when *Work Mode Select*==0x10/0x11, and only work in Mode0, cannot be written when TCE=1.
4	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0040			Register Name: SPI_BATC
Bit	**Read/Write**	**Default/Hex**	**Description**
3:2	R/W	0x0	SS_SEL SPI Chip Select Select one of four external SPI Master/Slave Devices 00: SPI_SS0 will be asserted 01: SPI_SS1 will be asserted 10: SPI_SS2 will be asserted 11: SPI_SS3 will be asserted It is only valid when *Work Mode Select*==0x10/0x11, and only work in Mode0, cannot be written when TCE=1.
1:0	R/W	0x0	Work Mode Select 00: Data frame is byte aligned in standard SPI, dual-output/dual input SPI, dual IO SPI, and quad-output/quad-input SPI 01: Reserved 10: Data frame is bit aligned in 3-wire SPI 11: Data frame is bit aligned in standard SPI

7.4.6.13 0x0044 SPI Bit-Aligned Clock Configuration Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.4.6.14 0x0048 SPI TX Bit Register (Default Value: 0x0000_0000)

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

7.4.6.15 0x004C SPI RX Bit Register (Default Value: 0x0000_0000)

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

7.4.6.16 0x0088 SPI Normal DMA Mode Control Register (Default Value: 0x0000_00E5)

Offset: 0x0088			Register Name: SPI_NDMA_MODE_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:6	R/W	0x11	ACK SELECT 00: dma_active is low 01: dma_active is high 10: dma_active is controlled by dma_request (DRQ) 11: dma_active is controlled by controller
5	R/W	0x1	ACK CARE 0: active fall do not care ack 1: active fall must after detect ack is high

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.4.6.17 0x0100 DBI Control Register 0 (Default Value: 0x0010_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0100			Register Name: DBI_CTL_0
Bit	**Read/Write**	**Default/Hex**	**Description**
14:12	R/W	0x0	Output Data Format 000: RGB111 001: RGB444 010: RGB565 011: RGB666 100: RGB888 (only for 2 Data Lane Interface) 101~111: Reserved
11	/	/	/
10:8	R/W	0x0	DBI Interface 000: 3 Line Interface I 001: 3 Line Interface II 010: 4 Line Interface I 011: 4 Line Interface II 100: 2 Data Lane Interface
7:4	R/W	0x0	RGB_Source_Format When *video_source_type* is RGB32 (DBI_CTL_0[bit0] = 0) 0000: RGB 0001: RBG 0010: GRB 0011: GBR 0100: BRG 0101: BGR Others: Reserved When *video_source_type* is RGB16 (DBI_CTL_0[bit0] = 1) 0000: RGB 0001~0100: Reserved 0101: BGR 0110: GRBG_0 {G[5:3]R[4:0]B[4:0]G[2:0]} 0111: GBRG_0 {G[5:3]B[4:0]R[4:0]G[2:0]} 1000: GRBG_1 {G[2:0]R[4:0]B[4:0]G[5:3]} 1001: GBRG_1 {G[2:0]B[4:0]R[4:0]G[5:3]} Others: Reserved
3	R/W	0x0	Dummy Cycle Value Output Value During Dummy Cycle

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0100			Register Name: DBI_CTL_0
Bit	**Read/Write**	**Default/Hex**	**Description**
2	R/W	0x0	RGB_Bit_Order 0: Remain the sequence of RGB data 1: Swap the higher bit and the lower bit for each component of DRAM RGB
1	R/W	0x0	Element_A_Position Only for RGB32 Data Format 0: A component is in the bit[31:24] of data source 1: A component is in the bit[7:0] of data source
0	R/W	0x0	Video_Source_Type 0: RGB32 1: RGB16

7.4.6.18 0x0104 DBI Control Register 1 (Default Value: 0x0000_0000)

Offset: 0x0104			Register Name: DBI_CTL_1
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/WAC	0x0	DBI_soft_trigger It is only available for software trigger mode. Writing '1' to this bit will start DBI TX module and auto clear to '0' when completing start operation, writing '0' to this bit has no effect.
30:29	R/W	0x0	DBI EN MODE SEL 00: Always on DBI mode 01: Software trigger mode 10: Timer trigger mode 11: TE trigger mode
28	/	/	/
27:26	R/W	0x0	2 Data Lane RGB666 Format 00: Normal Format 01: Special Format for ILITEK 10: Special Format for New Vision
25	R/W	0x0	DBI rx clock inv 0: Sample data by using the positive edge of the output clock 1: Sample data by using the negative edge of the output clock

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0104			Register Name: DBI_CTL_1
Bit	**Read/Write**	**Default/Hex**	**Description**
24	R/W	0x0	DBI output clock mode 0: DBI clock always on (DCX Setup/hold equals one clock cycle) 1: DBI clock auto gating (DCX Setup/hold equals to a half clock cycle)
23	R/W	0x0	DBI clock output Inv **When the bit24 (DBI output clock mode) is 0.** 0: The falling edge releases the CSX signal, and the falling edge releases data 1: The rising edge releases the CSX signal, and the rising edge releases data **When the bit24 (DBI output clock mode) is 1.** 0: The rising edge releases the CSX signal, and the falling edge releases data 1: The falling edge releases the CSX signal, and the rising edge releases data
22	R/W	0x0	DCX_DATA 0: DCX Value equal to 0 1: DCX Value equal to 1
21	R/W	0x0	RGB 16 Data Source Select 0: Pixel1 is stored in the higher bit of address, and Pixel0 is stored in the lower bit of address 1: Pixel0 is stored in the higher bit of address, and Pixel1 is stored in the lower bit of address
20	R/W	0x0	Read_MSB_First 0: A reading data is the higher bit 1: A reading data is the lower bit
19:16	/	/	/
15:8	R/W	0x0	Read Command Dummy Cycles The dummy cycle between the read command and read data Reading 1-byte (8 bits) data has not dummy cycle.
7:0	R/W	0x0	Read Data Number of Bytes Sample Bytes data based on configuration.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.4.6.19 0x0108 DBI Control Register 2 (Default Value: 0x0000_4000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.4.6.20 0x010C DBI Timer Register (Default Value: 0x0000_0000)

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

7.4.6.21 0x0110 DBI Video Size Register (Default Value: 0x01E0_0140)

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

7.4.6.22 0x0120 DBI Interrupt Register (Default Value: 0x0000_0000)

Offset: 0x0120			Register Name: DBI_INT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:15	/	/	/
14	W1C	0x0	DBI_FIFO_EMPTY_INT 0: DBI_FIFO is not empty 1: DBI_FIFO is empty

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0120			Register Name: DBI_INT
Bit	Read/Write	Default/Hex	Description
13	W1C	0x0	DBI_FIFO_FULL_INT 0: DBI_FIFO is not full 1: DBI_FIFO is full
12	W1C	0x0	TIMER_INT It indicates that the timer has been count sclk cycles to the value of *DBI_Timer Register[30:0]*. Writing 1 to this bit clears it. 0: Timer has not been achieved the objective 1: Timer has been achieved the objective
11	W1C	0x0	RD_DONE_INT It indicates that the number of byte setting in *DBI_Control Register 1[19:8]* has been read. Writing 1 to this bit clears it. 0: All data has not been read 1: All data has been read
10	W1C	0x0	TE_INT It indicates that the TE signal has been changed. Writing 1 to this bit clears it. 0: TE signal has not been changed 1: TE signal has been changed
9	W1C	0x0	FRAM_DONE_INT It indicates that a frame video data has been sent. Writing 1 to this bit clears it. 0: A frame video has not been sent 1: A frame video has been sent
0x8	W1C	0x0	LINE_DONE_INT It indicates that a line of video data has been sent. Writing 1 to this bit clears it. 0: A line of video data has not been sent 1: A line of video data has been sent
7	/	/	/
6	R/W	0x0	DBI_FIFO_EMPTY_INT_EN 0: Disable 1: Enable
5	R/W	0x0	DBI_FIFO_FULL_INT_EN 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0120			Register Name: DBI_INT
Bit	**Read/Write**	**Default/Hex**	**Description**
4	R/W	0x0	TIMER_INT_EN Timer Interrupt Enable 0: Disable 1: Enable
3	R/W	0x0	RD_DONE_INT_EN Read Done Interrupt Enable 0: Disable 1: Enable
2	R/W	0x0	TE_INT_EN TE Interrupt Enable 0: Disable 1: Enable
1	R/W	0x0	FRAM_DONE_INT_EN Frame Done Interrupt Enable 0: Disable 1: Enable
0	R/W	0x0	LINE_DONE_INT_EN Line Done Interrupt Enable 0: Disable 1: Enable

7.4.6.23 0x0124 DBI Debug Register 0 (Default Value: 0x0000_0000)

Offset: 0x0124			Register Name: DBI_Debug_0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:23	/	/	/
22:16	R	0x0	DBI_FIFO ROOM VALID 0~127 Words
15:13	/	/	/
12	R	0x0	TE input 0: TE not Trigger 1: TE Trigger
11:8	R	0x0	dbi_rxcs RX_BS0 ~ RX_BS6 , Gray - Code

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0124			Register Name: DBI_Debug_0
Bit	**Read/Write**	**Default/Hex**	**Description**
7:4	R	0x0	sh_cs 0~11 : SH0~SH11
3:2	/	/	dbi_txcs 00: IDLE 01: SHIF 10: DUMY 11: READ
1:0	R	0x0	tx_mem_cs 00: IDLE_FRM 01: FRM_POS 10: FRM_RDY

7.4.6.24 0x0128 DBI Debug Register 1 (Default Value: 0x0000_0000)

Offset: 0x0128			Register Name: DBI_Debug_1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:26	/	/	/
25:16	R	0x0	line_cnt The number of pixel lines that are currently sent
15:12	/	/	/
11:0	R	0x0	component_cnt The number of RGB components that are currently sent The field is equal to pixel_cnt *3.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.4.6.25 0x0200 SPI TX Data Register (Default Value: 0x0000_0000)

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

7.4.6.26 0x0300 SPI RX Data Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.5 USB2.0 DRD

7.5.1 Overview

The USB2.0 dual-role device (USB2.0 DRD) supports both device and host functions which can also be configured as a Host-only or Device-only controller. It complies with the USB2.0 Specification. The DRD module supports high-speed (HS, 480 Mbit/s), full-speed (FS, 12 Mbit/s), and low-speed (LS, 1.5 Mbit/s) transfers in Host mode. It also supports high-speed (HS, 480 Mbit/s), and full-speed (FS, 12 Mbit/s) in Device mode. The standard USB transceiver can be used through its UTMI+PHY Level3 interface. The UTMI+PHY interface is bidirectional with an 8-bit data bus. For saving CPU bandwidth, the DMA interface of the DRD module can also support the external DMA controller to do the data transfer between the memory and the DRD FIFO. The DRD core also supports USB power saving functions.

The USB2.0 DRD has the following features:

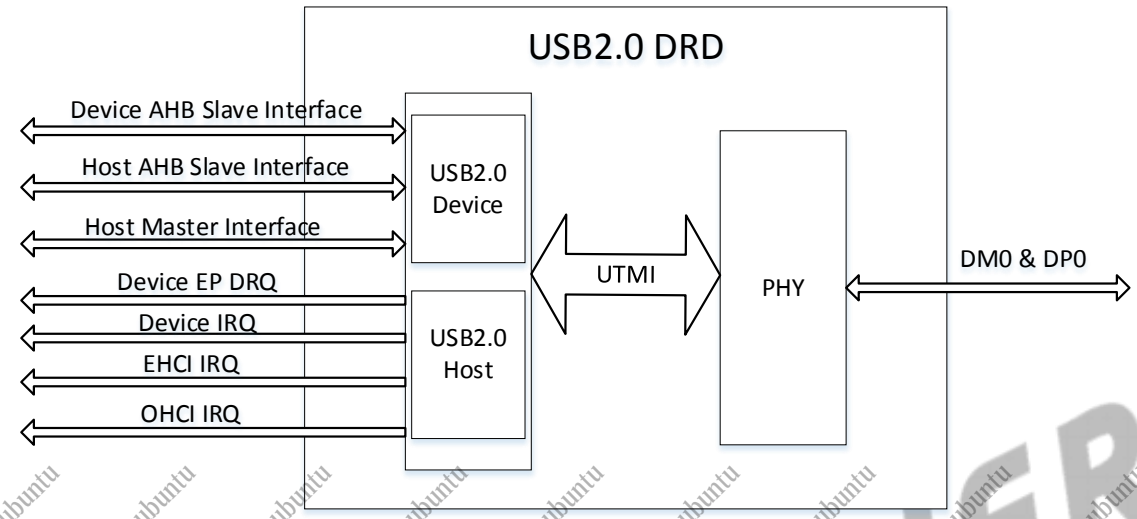
- Complies with USB2.0 Specification
- Supports High-Speed (HS, 480 Mbit/s), Full-Speed (FS, 12 Mbit/s), and Low-Speed (LS, 1.5 Mbit/s) in Host mode
- Supports High-Speed (HS, 480 Mbit/s), Full-Speed (FS, 12 Mbit/s) in Device mode
- Supports the UTMI+ Level 3 interface. The 8-bit bidirectional data buses are used
- Supports bi-directional endpoint0 (EP0) for Control transfer
- Up to 8 user-configurable endpoints (EP1 +, EP1 -, EP2 +, EP2 -, EP3 +, EP3 -, EP4 +, EP4 -) for Bulk transfer, Isochronous transfer and Interrupt transfer
- Up to (4 KB + 64 Bytes) FIFO for all EPs (including EP0)
- Supports High-Bandwidth Isochronous & Interrupt transfers
- Automated splitting/combining of packets for Bulk transfers
- Supports point-to-point and point-to-multipoint transfer in both Host and Peripheral modes
- Includes automatic ping capabilities
- Soft connect/disconnect function
- Performs all transaction scheduling in hardware
- Power optimization and power management capabilities
- Includes interface to an external Normal DMA controller for every EP
- Device and host controller share a 4K SRAM and a physical PHY

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.5.2 Block Diagram

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

Figure 7-60 USB2.0 DRD Controller Block Diagram



7.5.3 Functional Descriptions

7.5.3.1 External Signals

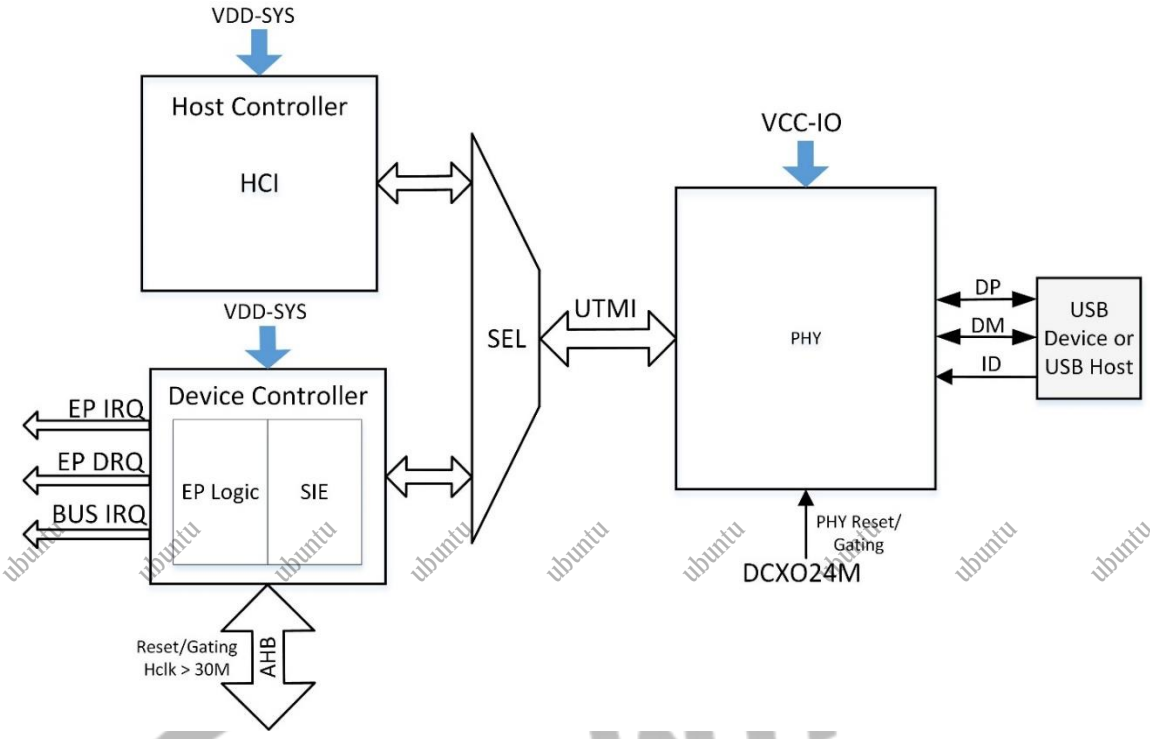
Table 7-21 USB2.0 DRD External Signals

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.5.3.2 Controller and PHY Connection Diagram

Figure 7-61 USB2.0 DRD Controller and PHY Connection Diagram



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.6 USB1.1 HOST

7.6.1 Overview

The USB Host Controller is fully compliant with USB 1.1 Specification and Open Host Controller Interface (OHCI) Specification Release 1.0a. The controller supports full-speed (12 Mbit/s) and low-speed (1.5 Mbit/s) data transfer modes through one or more integrated OHCI Host Controllers.

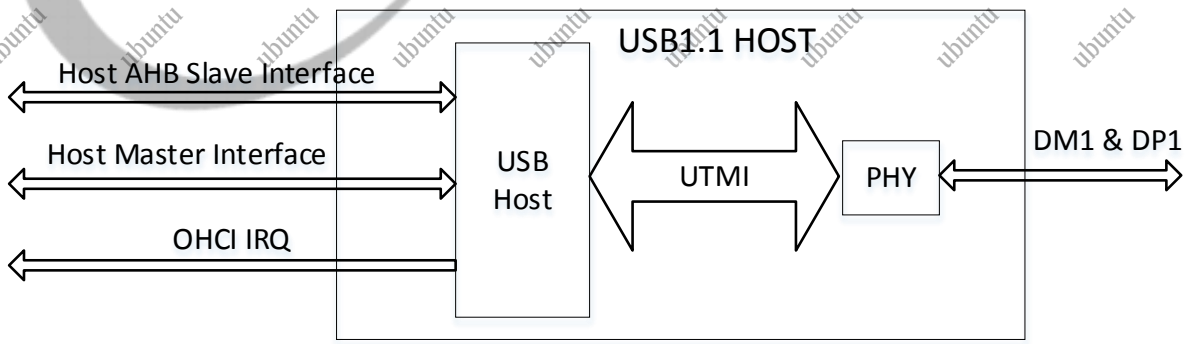
The USB1.1 host controller includes the following features:

- Complies with USB1.1 and the Open Host Controller Interface (OHCI) Specification, Version 1.0a
- Supports industry-standard AMBA High-Performance Bus (AHB) and it is fully compliant with the AMBA Specification, Revision 2.0
- Supports 32-bit Little Endian AMBA AHB Slave Bus for Register Access
- Supports 32-bit Little Endian AMBA AHB Master Bus for Memory Access
- An internal DMA Controller for data transfer with memory
- Supports Full-Speed (FS, 12 Mbit/s) and Low-Speed (LS, 1.5 Mbit/s) Device
- Supports the UTMI+ Level 3 interface. The 8-bit bidirectional data buses are used
- Supports only 1 USB Root port for OHCI

7.6.2 Block Diagram

The following figure shows the block diagram of USB1.1 Host Controller.

Figure 7-62 USB1.1 Host Controller Block Diagram



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.6.3 Functional Descriptions

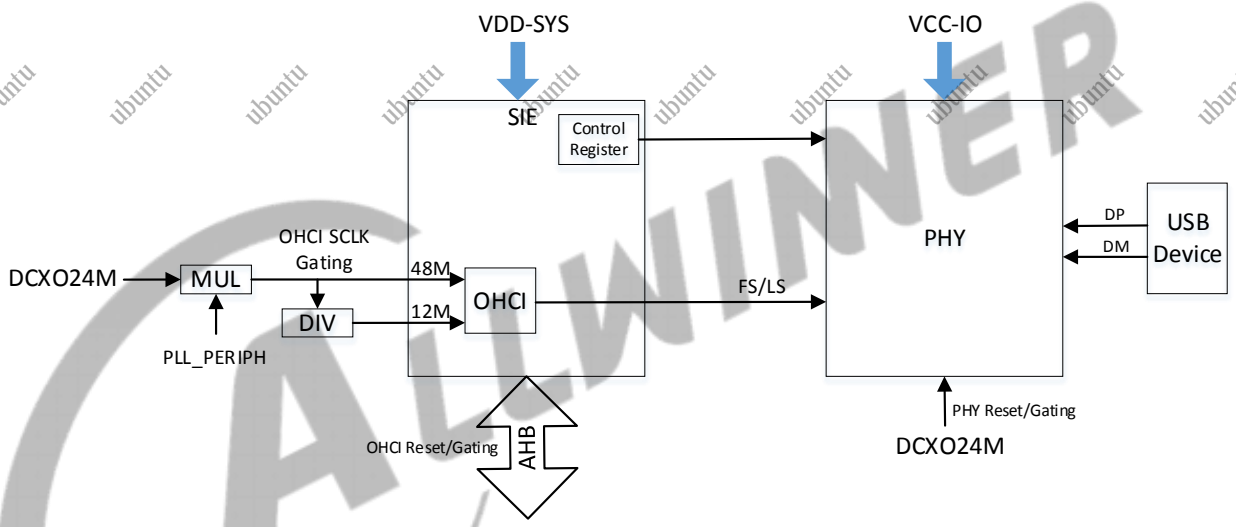
7.6.3.1 External Signals

Table 7-22 USB1.1 Host External Signals

Signal	Description	Type
USB1-DP	USB1.1 Host differential signal positive	AI/O
USB1-DM	USB1.1 Host differential signal negative	AI/O

7.6.3.2 Controller and PHY Connection Diagram

Figure 7-63 USB1.1 Host Controller and PHY Connection Diagram



7.6.3.3 USB1.1 Host Timing

Refer to USB1.1 Specification and Open Host Controller Interface (OHCI) Specification, Version 1.0a.

7.6.4 Register List

Module Name	Base Address
USB1	0x04201000

Register Name	Offset	Description
OHCI Control and Status Partition Register		
O_HcRevision	0x0400	OHCI Revision Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
O_HcControl	0x0404	OHCI Control Register
O_HcCommandStatus	0x0408	OHCI Command Status Register
O_HcInterruptStatus	0x040C	OHCI Interrupt Status Register
O_HcInterruptEnable	0x0410	OHCI Interrupt Enable Register
O_HcInterruptDisable	0x0414	OHCI Interrupt Disable Register
OHCI Memory Pointer Partition Register		
O_HcHCCA	0x0418	OHCI HCCA Base
O_HcPeriodCurrentED	0x041C	OHCI Period Current ED Base
O_HcControlHeadED	0x0420	OHCI Control Head ED Base
O_HcControlCurrentED	0x0424	OHCI Control Current ED Base
O_HcBulkHeadED	0x0428	OHCI Bulk Head ED Base
O_HcBulkCurrentED	0x042C	OHCI Bulk Current ED Base
O_HcDoneHead	0x0430	OHCI Done Head Base
OHCI Frame Counter Partition Register		
O_HcFmInterval	0x0434	OHCI Frame Interval Register
O_HcFmRemaining	0x0438	OHCI Frame Remaining Register
O_HcFmNumber	0x043C	OHCI Frame Number Register
O_HcPerioddicStart	0x0440	OHCI Periodic Start Register
O_HcLSThreshold	0x0444	OHCI LS Threshold Register
OHCI Root Hub Partition Register		
O_HcRhDescriptorA	0x0448	OHCI Root Hub Descriptor Register A
O_HcRhDesriptorB	0x044C	OHCI Root Hub Descriptor Register B
O_HcRhStatus	0x0450	OHCI Root Hub Status Register
O_HcRhPortStatus	0x0454	OHCI Root Hub Port Status Register
HCI Controller and PHY Interface Register		
HCI_Interface	0x0800	HCI Interface Register
HCI_CTRL3	0x0808	HCI Control Register
PHY_Control	0x0810	PHY Control Register
PHY_STATUS	0x0824	PHY Status Register
HCI SIE Port Disable Control	0x0828	HCI SIE Port Disable Control Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.6.5 Register Description

7.6.5.1 0x0400 HcRevision Register (Default Value: 0x0000_0010)

Offset: 0x0400				Register Name: HcRevision
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:8	/	/	/	/
7:0	R	R	0x10	Revision This read-only field contains the BCD representation of the version of the HCI specification that is implemented by this HC. For example, a value of 0x11 corresponds to version 1.1. All of the HC implementations that are compliant with this specification will have a value of 0x10.

7.6.5.2 0x0404 HcControl Register (Default Value: 0x0000_0000)

Offset: 0x0404				Register Name: HcRevision
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:11	/	/	/	/
10	R/W	R	0x0	**RemoteWakeupEnable** This bit is used by HCD to enable or disable the remote wakeup feature upon the detection of upstream resume signaling. When this bit is set and the **ResumeDetected** bit in ***HcInterruptStatus*** is set, a remote wakeup is signaled to the host system. Setting this bit has no impact on the generation of hardware interrupt.
9	R/W	R/W	0x0	**RemoteWakeupConnected** This bit indicates whether HC supports remote wakeup signaling. If remote wakeup is supported and used by the system, it is the responsibility of system firmware to set this bit during POST. HC clear the bit upon a hardware reset but does not alter it upon a software reset. Remote wakeup signaling of the host system is host-bus-specific and is not described in this specification.
8	R/W	R	0x0	**InterruptRouting** This bit determines the routing of interrupts generated by events registered in HcInterruptStatus. If clear, all interrupt are routed to the normal host bus interrupt mechanism. If set interrupts are routed to the System Management Interrupt. HCD clears this bit upon a hardware reset, but it does not alter this bit upon a software reset. HCD uses this bit as a tag to indicate the ownership of HC.
7:6	R/W	R/W	0x0	HostControllerFunctionalState for USB

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

<table>
<tr><th colspan="3">Offset: 0x0404</th><th colspan="2">Register Name: HcRevision</th></tr>
<tr><th rowspan="2">Bit</th><th colspan="2">Read/Write</th><th rowspan="2">Default/Hex</th><th rowspan="2">Description</th></tr>
<tr><th>HCD</th><th>HC</th></tr>
<tr><td></td><td></td><td></td><td></td><td>
<table>
<tr><td>00b</td><td>USBReset</td></tr>
<tr><td>01b</td><td>USBResume</td></tr>
<tr><td>10b</td><td>USBOperational</td></tr>
<tr><td>11b</td><td>USBSuspend</td></tr>
</table>
A transition to USBOperational from another state causes SOF generation to begin 1 ms later. HCD may determine whether HC has begun sending SOFs by reading the StartoFrame field of HcInterruptStatus.
This field may be changed by HC only when in the USBSUSPEND state. HC may move from the USBSUSPEND state to the USBRESUME state after detecting the resume signaling from a downstream port.
HC enters USBSUSPEND after a software reset, whereas it enters USBRESET after a hardware reset. The latter also resets the Root Hub and asserts subsequent reset signaling to downstream ports.</td></tr>
<tr><td>5</td><td>R/W</td><td>R</td><td>0x0</td><td>BulkListEnable
This bit is set to enable the processing of the Bulk list in the next Frame. If cleared by HCD, the processing of the Bulk list does not occur after the next SOF. HC checks this bit whenever it determines to process the list. When disabled, HCD may modify the list. If HcBulkCurrentED is pointing to an ED to be removed, HCD must advance the pointer by updating HcBulkCurrentED before re-enabling processing of the list.</td></tr>
<tr><td>4</td><td>R/W</td><td>R</td><td>0x0</td><td>ControlListEnable
This bit is set to enable the processing of the Control list in the next Frame. If cleared by HCD, the processing of the Control list does not occur after the next SOF. HC must check this bit whenever it determines to process the list. When disabled, HCD may modify the list. If HcControlCurrentED is pointing to an ED to be removed, HCD must advance the pointer by updating HcControlCurrentED before re-enabling processing of the list.</td></tr>
<tr><td>3</td><td>R/W</td><td>R</td><td>0x0</td><td>IsochronousEnable
This bit is used by HCD to enable/disable processing of isochronous EDs. While processing the periodic list in a Frame, HC checks the status of this bit when it finds an Isochronous ED (F=1). If set (enabled), HC continues processing the EDs. If cleared (disabled), HC halts processing of the periodic list (which now contains only isochronous EDs) and begins processing the Bulk/Control lists.
Setting this bit is guaranteed to take effect in the next Frame (not the current Frame).</td></tr>
<tr><td>2</td><td>R/W</td><td>R</td><td>0x0</td><td>PeriodicListEnable</td></tr>
</table>

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

<table>
<tr><th colspan="4">Offset: 0x0404</th><th>Register Name: HcRevision</th></tr>
<tr><th rowspan="2">Bit</th><th colspan="2">Read/Write</th><th rowspan="2">Default/Hex</th><th rowspan="2">Description</th></tr>
<tr><th>HCD</th><th>HC</th></tr>
<tr><td></td><td></td><td></td><td></td><td>This bit is set to enable the processing of periodic list in the next Frame. If cleared by HCD, processing of the periodic list does not occur after the next SOF. HC must check this bit before it starts processing the list.</td></tr>
<tr><td>1:0</td><td>R/W</td><td>R</td><td>0x0</td><td>ControlBulkServiceRatio
This specifies the service ratio between Control and Bulk EDs. Before processing any of the nonperiodic lists, HC must compare the ratio specified with its internal count on how many nonempty Control EDs have been processed, in determining whether to continue serving another Control ED or switching to Bulk EDs. The internal count will be retained when crossing the frame boundary. In case of reset, HCD is responsible for restoring this value.
<table>
<tr><td>CBSR</td><td>No. of Control EDs Over Bulk EDs Served</td></tr>
<tr><td>0</td><td>1:1</td></tr>
<tr><td>1</td><td>2:1</td></tr>
<tr><td>2</td><td>3:1</td></tr>
<tr><td>3</td><td>4:1</td></tr>
</table>
The default value is 0x0.</td></tr>
</table>

7.6.5.3 0x0408 HcCommandStatus Register (Default Value: 0x0000_0000)

<table>
<tr><th colspan="4">Offset: 0x0408</th><th>Register Name: HcCommandStatus</th></tr>
<tr><th rowspan="2">Bit</th><th colspan="2">Read/Write</th><th rowspan="2">Default/Hex</th><th rowspan="2">Description</th></tr>
<tr><th>HCD</th><th>HC</th></tr>
<tr><td>31:18</td><td>/</td><td>/</td><td>0x0</td><td>Reserved</td></tr>
<tr><td>17:16</td><td>R</td><td>R/W</td><td>0x0</td><td>SchedulingOverrunCount
These bits are incremented on each scheduling overrun error. It is initialized to 00b and wraps around at 11b. This will be incremented when a scheduling overrun is detected even if SchedulingOverrun in HcInterruptStatus has already been set. This is used by HCD to monitor any persistent scheduling problem.</td></tr>
<tr><td>15:4</td><td>/</td><td>/</td><td>/</td><td>/</td></tr>
<tr><td>3</td><td>R/W</td><td>R/W</td><td>0x0</td><td>OwershipChangeRequest
This bit is set by an OS HCD to request a change of control of the HC. When set HC will set the OwnershipChange field in HcInterruptStatus. After the changeover, this bit is cleared and remains so until the next request from OS HCD.</td></tr>
<tr><td>2</td><td>R/W</td><td>R/W</td><td>0x0</td><td>BulklListFilled
This bit is used to indicate whether there are any TDs on the Bulk list. It is set by HCD whenever it adds a TD to an ED in the Bulk list.</td></tr>
</table>

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0408				Register Name: HcCommandStatus
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
				When HC begins to process the head of the Bulk list, it checks BLF. As long as **BulkListFilled** is 0, HC will not start processing the Bulk list. If **BulkListFilled** is 1, HC will start processing the Bulk list and will set BF to 0. If HC finds a TD on the list, then HC will set **BulkListFilled** to 1 causing the Bulk list processing to continue. If no TD is found on the Bulk list, and if HCD does not set **BulkListFilled**, then **BulkListFilled** will still be 0 when HC completes processing the Bulk list and Bulk list processing will stop.
1	R/W	R/W	0x0	**ControlListFilled** This bit is used to indicate whether there are any TDs on the Control list. It is set by HCD whenever it adds a TD to an ED in the Control list. When HC begins to process the head of the Control list, it checks CLF. As long as **ControlListFilled** is 0, HC will not start processing the Control list. If CF is 1, HC will start processing the Control list and will set **ControlListFilled** to 0. If HC finds a TD on the list, then HC will set **ControlListFilled** to 1 causing the Control list processing to continue. If no TD is found on the Control list, and if the HCD does not set **ControlListFilled**, then **ControlListFilled** will still be 0 when HC completes processing the Control list and Control list processing will stop.
0	R/W	R/E	0x0	**HostControllerReset** This bit is by HCD to initiate a software reset of HC. Regardless of the functional state of HC, it moves to the USBSuspend state in which most of the operational registers are reset except those stated otherwise; e.g, the InteruptRouting field of HcControl, and no Host bus accesses are allowed. This bit is cleared by HC upon the completion of the reset operation. The reset operation must be completed within 10 ms. This bit, when set, should not cause a reset to the Root Hub and no subsequent reset signaling should be asserted to its downstream ports.

7.6.5.4 0x040C HcInterruptStatus Register (Default Value: 0x0000_0000)

Offset: 0x040C				Register Name: HcInterruptStatus
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:7	/	/	/	/
6	R/W	R/W	0x0	**RootHubStatusChange**

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x040C				Register Name: HcInterruptStatus
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
				This bit is set when the content of *HcRhStatus* or the content of any of *HcRhPortStatus*[**NumberofDownstreamPort**] has changed.
5	R/W	R/W	0x0	**FrameNumberOverflow** This bit is set when the MSb of *HcFmNumber* (bit 15) changes value, from 0 to 1 or from 1 to 0, and after *HccaFrameNumber* has been updated.
4	R/W	R/W	0x0	**UnrecoverableError** This bit is set when HC detects a system error not related to USB. HC should not proceed with any processing nor signaling before the system error has been corrected. HCD clears this bit after HC has been reset.
3	R/W	R/W	0x0	**ResumeDetected** This bit is set when HC detects that a device on the USB is asserting resume signaling. It is the transition from no resume signaling to resume signaling causing this bit to be set. This bit is not set when HCD sets the USBRseume state.
2	R/W	R/W	0x0	**StartofFrame** This bit is set by HC at each start of frame and after the update of HccaFrameNumber. HC also generates a SOF token at the same time.
1	R/W	R/W	0x0	**WritebackDoneHead** This bit is set immediately after HC has written *HcDoneHead* to *HccaDoneHead*. Further updates of the *HccaDoneHead* will not occur until this bit has been cleared. HCD should only clear this bit after it has saved the content of *HccaDoneHead*.
0	R/W	R/W	0x0	**SchedulingOverrun** This bit is set when the USB schedule for the current Frame overruns and after the update of *HccaFrameNumber*. A scheduling overrun will also cause the **SchedulingOverrunCount** of *HcCommandStatus* to be incremented.

7.6.5.5 0x0410 HcInterruptEnable Register (Default Value: 0x0000_0000)

Offset: 0x0410				Register Name: HcInterruptEnable Register
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31	R/W	R	0x0	**MasterInterruptEnable** A '0' writtern to this field is ignored by HC. A '1' written to this field enables interrupt generation due to events specified in the other bits of this register. This is used by HCD as Master Interrupt Enable.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0410				Register Name: HcInterruptEnable Register
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
30:7	/	/	/	/
6	R/W	R	0x0	**RootHubStatusChange Interrupt Enable** 0 Ignore; 1 Enable interrupt generation due to Root Hub Status Change;
5	R/W	R	0x0	**FrameNumberOverflow Interrupt Enable** 0 Ignore; 1 Enable interrupt generation due to Frame Number Over Flow;
4	R/W	R	0x0	**UnrecoverableError Interrupt Enable** 0 Ignore; 1 Enable interrupt generation due to Unrecoverable Error;
3	R/W	R	0x0	**ResumeDetected Interrupt Enable** 0 Ignore; 1 Enable interrupt generation due to Resume Detected;
2	R/W	R	0x0	**StartofFrame Interrupt Enable** 0 Ignore; 1 Enable interrupt generation due to Start of Flame;
1	R/W	R	0x0	**WritebackDoneHead Interrupt Enable** 0 Ignore; 1 Enable interrupt generation due to Write back Done Head;
0	R/W	R	0x0	**SchedulingOverrun Interrupt Enable** 0 Ignore; 1 Enable interrupt generation due to Scheduling Overrun;

7.6.5.6 0x0414 HcInterruptDisable Register (Default Value: 0x0000_0000)

Offset: 0x0414				Register Name: HcInterruptDisable Register
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31	R/W	R	0x0	**MasterInterruptEnable** A written '0' to this field is ignored by HC. A '1' written to this field disables interrupt generation due events specified in the other bits of this register. This field is set after a hardware or software reset.
30:7	/	/	/	/
6	R/W	R	0x0	**RootHubStatusChange Interrupt Disable** 0 Ignore; 1 Disable interrupt generation due to Root Hub Status Change;

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0414				Register Name: HcInterruptDisable Register
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
5	R/W	R	0x0	**FrameNumberOverflow Interrupt Disable** 0: Ignore; 1: Disable interrupt generation due to Frame Number Over Flow;
4	R/W	R	0x0	**UnrecoverableError Interrupt Disable** 0: Ignore; 1: Disable interrupt generation due to Unrecoverable Error;
3	R/W	R	0x0	**ResumeDetected Interrupt Disable** 0: Ignore; 1: Disable interrupt generation due to Resume Detected;
2	R/W	R	0x0	**StartofFrame Interrupt Disable** 0: Ignore; 1: Disable interrupt generation due to Start of Flame;
1	R/W	R	0x0	**WritebackDoneHead Interrupt Disable** 0: Ignore; 1: Disable interrupt generation due to Write back Done Head;
0	R/w	R	0x0	**SchedulingOverrun Interrupt Disable** 0: Ignore; 1: Disable interrupt generation due to Scheduling Overrun;

7.6.5.7 0x0418 HcHCCA Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.6.5.8 0x041C HcPeriodCurrentED Register (Default Value: 0x0000_0000)

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

7.6.5.9 0x0420 HcControlHeadED Register (Default Value: 0x0000_0000)

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

7.6.5.10 0x0424 HcControlCurrentED Register (Default Value: 0x0000_0000)

Offset: 0x0424				Register Name: HcControlCurrentED[CCED]
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:4	R/W	R/W	0x0	**CCED[31:4]** The pointer is advanced to the next ED after serving the present one. HC will continue processing the list from where it left off in the last Frame. When it reaches the end of the Control list, HC checks the ControlListFilled of in HcCommandStatus. If set, it copies the content of HcControlHeadED to HcControlCurrentED

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0424				Register Name: HcControlCurrentED[CCED]
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
				and clears the bit. If not set, it does nothing. HCD is allowed to modify this register only when the ControlListEnable of HcControl is cleared. When set, HCD only reads the instantaneous value of this register. Initially, this is set to zero to indicate the end of the Control list.
3:0	R	R	0x0	**CCED[3:0]** Because the general TD length is 16 bytes, the memory structure for the TD must be aligned to a 16-byte boundary. So the lower bits in the PCED, through bit 0 to bit 3 must be zero in this field.

7.6.5.11 0x0428 HcBulkHeadED Register (Default Value: 0x0000_0000)

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

7.6.5.12 0x042C HcBulkCurrentED Register (Default Value: 0x0000_0000)

Offset: 0x042C				Register Name: HcBulkCurrentED [BCED]
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:4	R/W	R/W	0x0	**BulkCurrentED[31:4]** This is advanced to the next ED after the HC has served the present one. HC continues processing the list from where it left off in the last Frame. When it reaches the end of the Bulk list, HC checks the **ControlListFilled** of HcControl. If set, it copies the content of *HcBulkHeadED* to *HcBulkCurrentED* and clears the bit. If it is not set, it does nothing. HCD is only allowed to modify this register when the **BulkListEnable** of *HcControl* is cleared. When set, the HCD only reads the instantaneous value of this register. This is

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x042C				Register Name: HcBulkCurrentED [BCED]
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
				initially set to zero to indicate the end of the Bulk list.
3:0	R	R	0x0	**BulkCurrentED [3:0]** Because the general TD length is 16 bytes, the memory structure for the TD must be aligned to a 16-byte boundary. So the lower bits in the PCED, through bit 0 to bit 3 must be zero in this field.

7.6.5.13 0x0430 HcDoneHead Register (Default Value: 0x0000_0000)

Offset: 0x0430				Register Name: HcDoneHead
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:4	R	R/W	0x0	**HcDoneHead[31:4]** When a TD is completed, HC writes the content of *HcDoneHead* to the NextTD field of the TD. HC then overwrites the content of *HcDoneHead* with the address of this TD. This is set to zero whenever HC writes the content of this register to HCCA. It also sets the **WritebackDoneHead** of *HcInterruptStatus*.
3:0	R	R	0x0	**HcDoneHead[3:0]** Because the general TD length is 16 bytes, the memory structure for the TD must be aligned to a 16-byte boundary. So the lower bits in the PCED, through bit 0 to bit 3 must be zero in this field.

7.6.5.14 0x0434 HcFmInterval Register (Default Value: 0x0000_2EDF)

Offset: 0x0434				Register Name: HcFmInterval Register
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31	R/W	R	0x0	**FrameIntervalToggler** HCD toggles this bit whenever it loads a new value to **FrameInterval**.
30:16	R/W	R	0x0	**FSLargestDataPacket** This field specifies a value which is loaded into the Largest Data Packet Counter at the beginning of each frame. The counter value represents the largest amount of data in bits which can be sent or received by the HC in a single transaction at any given time without causing scheduling overrun. The field value is calculated by the HCD.
15:14	/	/	/	/
13:0	R/W	R	0x2edf	**FrameInterval**

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0434				Register Name: HcFmInterval Register
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
				This specifies the interval between two consecutive SOFs in bit times. The nominal value is set to be 11,999. HCD should store the current value of this field before resetting HC. By setting the **HostControllerReset** field of *HcCommandStatus* as this will cause the HC to reset this field to its nominal value. HCD may choose to restore the stored value upon the completion of the Reset sequence.

7.6.5.15 0x0438 HcFmRemaining Register (Default Value: 0x0000_0000)

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

7.6.5.16 0x043C HcFmNumber Register (Default Value: 0x0000_0000)

Offset: 0x043C				Register Name: HcFmNumber
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:16	/	/	/	/
15:0	R	R/W	0x0	**FrameNumber** This is incremented when *HcFmRemaining* is re-loaded. It will be rolled over to 0x0 after 0x0ffff. When entering the USBOPERATIONAL state, this will be incremented automatically. The content will be written to HCCA after HC has incremented the **FrameNumber** at each frame boundary and sent a SOF but before

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x043C				Register Name: HcFmNumber
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
				HC reads the first ED in that Frame. After writing to HCCA, HC will set the **StartofFrame** in *HcInterruptStatus.*

7.6.5.17 0x0440 HcPeriodicStart Register (Default Value: 0x0000_0000)

Offset: 0x0440				Register Name: HcPeriodicStatus
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:14	/	/	/	/
13:0	R/W	R	0x0	**PeriodicStart** After a hardware reset, this field is cleared. This is then set by HCD during the HC initialization. The value is calculated roughly as 10% off from *HcFmInterval*. A typical value will be 0x2A3F (0x3e67??). When *HcFmRemaining* reaches the value specified, processing of the periodic lists will have priority over Control/Bulk processing. HC will therefore start processing the Interrupt list after completing the current Control or Bulk transaction that is in progress.

7.6.5.18 0x0444 HcLSThreshold Register (Default Value: 0x0000_0628)

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

7.6.5.19 0x0448 HcRhDescriptorA Register (Default Value: 0x0200_1201)

Offset: 0x0448				Register Name: HcRhDescriptorA
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:24	R/W	R	0x2	**PowerOnToPowerGoodTime[POTPGT]**

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0448				Register Name: HcRhDescriptorA
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
				This byte specifies the duration HCD has to wait before accessing a powered-on port of the Root Hub. It is implementation-specific. The unit of time is 2 ms. The duration is calculated as POTPGT * 2ms.
23:13	/	/	/	/
12	R/W	R	0x1	**NoOverCurrentProtection** This bit describes how the overcurrent status for the Root Hub ports are reported. When this bit is cleared, the **OverCurrentProtectionMode** field specifies global or per-port reporting. **0** Over-current status is reported collectively for all downstream ports. **1** No overcurrent protection supported.
11	R/W	R	0x0	**OverCurrentProtectionMode** This bit describes how the overcurrent status for the Root Hub ports are reported. At reset, these fields should reflect the same mode as **PowerSwitchingMode**. This field is valid only if the **NoOverCurrentProtection** field is cleared. 0 Over-current status is reported collectively for all downstream ports. 1 Over-current status is reported on per-port basis.
10	R	R	0x0	**Device Type** This bit specifies that the Root Hub is not a compound device. The Root Hub is not permitted to be a compound device. This field should always read/write 0.
9	R/W	R	0x1	**PowerSwitchingMode** This bit is used to specify how the power switching of the Root Hub ports is controlled. It is implementation-specific. This field is only valid if the **NoPowerSwitching** field is cleared. 0 All ports are powered at the same time. 1 Each port is powered individually. This mode allows port power to be controlled by either the global switch or per-port switching. If the **PortPowerControlMask** bit is set, the port responds only to port power commands (**Set/ClearPortPower**). If the port mask is cleared, then the port is controlled only by the global power switch (**Set/ClearGlobalPower**).

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0448				Register Name: HcRhDescriptorA
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
8	R/W	R	0x0	**NoPowerSwithcing** These bits are used to specify whether power switching is supported or ports are always powered. It is implementation-specific. When this bit is cleared, the **PowerSwitchingMode** specifies global or per-port switching. **0** Ports are power switched. **1** Ports are always powered on when the HC is powered on.
7:0	R	R	0x01	**NumberDownstreamPorts** These bits specify the number of downstream ports supported by the Root Hub. It is implementation-specific. The minimum number of ports is 1. The maximum number of ports supported.

7.6.5.20 0x044C HcRhDescriptorB Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.6.5.21 0x0450 HcRhStatus Register (Default Value: 0x0000_0000)

Offset: 0x0450				Register Name: HcRhStatus Register
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31	W	R	0x0	**(write)ClearRemoteWakeupEnable** Write a '1' clears **DeviceRemoteWakeupEnable.** Writing a '0' has no effect.
30:18	/	/	/	/
17	R/W	R	0x0	**OverCurrentIndicatorChang** This bit is set by hardware when a change has occurred to the **OverCurrentIndicator** field of this register. The HCD clears this bit by writing a '1'. Writing a '0' has no effect.
16	R/W	R	0x0	**(read)LocalPowerStartusChange** The Root Hub does not support the local power status features, thus, this bit is always read as '0'. **(write)SetGlobalPower** In global power mode (**PowerSwitchingMode**=0), This bit is written to '1' to turn on power to all ports (clear **PortPowerStatus**). In per-port power mode, it sets **PortPowerStatus** only on ports whose **PortPowerControlMask** bit is not set. Writing a '0' has no effect.
15	R/W	R	0x0	**(read)DeviceRemoteWakeupEnable** This bit enables a **ConnectStatusChange** bit as a resume event, causing a USBSUSPEND to USBRESUME state transition and setting the **ResumeDetected** interrupt. 0 **ConnectStatusChange** is not a remote wakeup event. 1 **ConnectStatusChange** is a remote wakeup event. **(write)SetRemoteWakeupEnable** Writing a '1' sets DeviceRemoveWakeupEnable. Writing a '0' has no effect.
14:2	/	/	/	/
1	R	R/W	0x0	**OverCurrentIndicator** This bit reports overcurrent conditions when the global reporting is implemented. When set, an overcurrent condition exists. When cleared, all power operations are normal. If per-port overcurrent protection is implemented this bit is always '0'
0	R/W	R	0x0	**(Read)LocalPowerStatus** When read, this bit returns the LocalPowerStatus of the Root Hub. The Root Hub does not support the local power status feature;

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0450				Register Name: HcRhStatus Register
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
				thus, this bit is always read as '0'. **(Write)ClearGlobalPower** When write, this bit is operated as the ClearGlobalPower. In global power mode (**PowerSwitchingMode**=0), This bit is written to '1' to turn off power to all ports (clear **PortPowerStatus**). In per-port power mode, it clears **PortPowerStatus** only on ports whose **PortPowerControlMask** bit is not set. Writing a '0' has no effect.

7.6.5.22 0x0454 HcRhPortStatus Register (Default Value: 0x0000_0100)

Offset: 0x0454				Register Name: HcRhPortStatus
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:21	/	/	/	/
20	R/W	R/W	0x0	**PortResetStatusChange** This bit is set at the end of the 10-ms port reset signal. The HCD writes a '1' to clear this bit. Writing a '0' has no effect. 0 port reset is not complete 1 port reset is complete
19	R/W	R/W	0x0	**PortOverCurrentIndicatorChange** This bit is valid only if overcurrent conditions are reported on a per-port basis. This bit is set when Root Hub changes the **PortOverCurrentIndicator** bit. The HCD writes a '1' to clear this bit. Writing a '0' has no effect. 0 no change in **PortOverCurrentIndicator** 1 **PortOverCurrentIndicator** has changed
18	R/W	R/W	0x0	**PortSuspendStatusChange** This bit is set when the full resume sequence has been completed. This sequence includes the 20-s resume pulse, LS EOP, and 3-ms resychronization delay. The HCD writes a '1' to clear this bit. Writing a '0' has no effect. This bit is also cleared when **ResetStatusChange** is set. 0 resume is not completed 1 resume completed
17	R/W	R/W	0x0	**PortEnableStatusChange** This bit is set when hardware events cause the **PortEnableStatus** bit to be cleared. Changes from HCD writes do not set this bit. The HCD writes a '1' to clear this bit. Writing a '0' has no effect. 0 no change in **PortEnableStatus** 1 change in **PortEnableStatus**

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

<table>
<tr><td colspan="3">Offset: 0x0454</td><td colspan="2">Register Name: HcRhPortStatus</td></tr>
<tr><td rowspan="2">Bit</td><td colspan="2">Read/Write</td><td rowspan="2">Default/Hex</td><td rowspan="2">Description</td></tr>
<tr><td>HCD</td><td>HC</td></tr>
<tr><td>16</td><td>R/W</td><td>R/W</td><td>0x0</td><td>ConnectStatusChange
This bit is set whenever a connect or disconnect event occurs. The HCD writes a ‘1’ to clear this bit. Writing a ‘0’ has no effect. If CurrentConnectStatus is cleared when a SetPortReset,SetPortEnable, or SetPortSuspend write occurs, this bit is set to force the driver to re-evaluate the connection status since these writes should not occur if the port is disconnected.
0 | no change in PortEnableStatus
1 | change in PortEnableStatus
Note: If the DeviceRemovable[NDP] bit is set, this bit is set only after a Root Hub reset to inform the system that the device is attached.</td></tr>
<tr><td>15:10</td><td>/</td><td>/</td><td>/</td><td>/</td></tr>
<tr><td>9</td><td>R/W</td><td>R/W</td><td>0x0</td><td>(read)LowSpeedDeviceAttached
This bit indicates the speed of the device attached to this port. When set, a Low Speed device is attached to this port. When clear, a Full Speed device is attached to this port. This field is valid only when the CurrentConnectStatus is set.
0 | full speed device attached
1 | low speed device attached
(write)ClearPortPower
The HCD clears the PortPowerStatus bit by writing a ‘1’ to this bit. Writing a ‘0’ has no effect.</td></tr>
<tr><td>8</td><td>R/W</td><td>R/W</td><td>0x1</td><td>(read)PortPowerStatus
This bit reflects the port’s power status, regardless of the type of power switching implemented. This bit is cleared if an overcurrent condition is detected. HCD sets this bit by writing SetPortPower or SetGlobalPower. HCD clears this bit by writing ClearPortPower or ClearGlobalPower. Which power control switches are enabled is determined by PowerSwitchingMode and PortPortControlMask[NumberDownstreamPort]. In global switching mode(PowerSwitchingMode=0), only Set/ClearGlobalPower controls
this bit. In per-port power switching (PowerSwitchingMode=1), if the PortPowerControlMask[NDP] bit for the port is set, only Set/ClearPortPower commands are enabled. If the mask is not set, only Set/ClearGlobalPower commands are enabled. When port power is disabled, CurrentConnectStatus, PortEnableStatus,</td></tr>
</table>

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

<table>
<tr><td colspan="3">Offset: 0x0454</td><td colspan="2">Register Name: HcRhPortStatus</td></tr>
<tr><td rowspan="2">Bit</td><td colspan="2">Read/Write</td><td rowspan="2">Default/Hex</td><td rowspan="2">Description</td></tr>
<tr><td>HCD</td><td>HC</td></tr>
<tr><td></td><td></td><td></td><td></td><td>PortSuspendStatus, and PortResetStatus should be reset.
<table><tr><td>0</td><td>port power is off</td></tr><tr><td>1</td><td>port power is on</td></tr></table>
(write)SetPortPower
The HCD writes a ‘1’ to set the PortPowerStatus bit. Writing a ‘0’ has no effect.

Note: This bit is always reads ‘1b’ if power switching is not supported.</td></tr>
<tr><td>7:5</td><td>/</td><td>/</td><td>/</td><td>/</td></tr>
<tr><td>4</td><td>R/W</td><td>R/W</td><td>0x0</td><td>(read)PortResetStatus
When this bit is set by a write to SetPortReset, port reset signaling is asserted. When reset is completed, this bit is cleared when PortResetStatusChange is set. This bit cannot be set if CurrentConnectStatus is cleared.
<table><tr><td>0</td><td>port reset signal is not active</td></tr><tr><td>1</td><td>port reset signal is active</td></tr></table>
(write)SetPortReset
The HCD sets the port reset signaling by writing a ‘1’ to this bit. Writing a ‘0’ has no effect. If CurrentConnectStatus is cleared, this write does not set PortResetStatus, but instead sets ConnectStatusChange. This informs the driver that it attempted to reset a disconnected port.</td></tr>
<tr><td>3</td><td>R/W</td><td>R/W</td><td>0x0</td><td>(read)PortOverCurrentIndicator
This bit is only valid when the Root Hub is configured in such a way that overcurrent conditions are reported on a per-port basis. If per-port overcurrent reporting is not supported, this bit is set to 0. If cleared, all power operations are normal for this port. If set, an overcurrent condition exists on this port. This bit always reflects the overcurrent input signal.
<table><tr><td>0</td><td>no overcurrent condition.</td></tr><tr><td>1</td><td>overcurrent condition detected.</td></tr></table>
(write)ClearSuspendStatus
The HCD writes a ‘1’ to initiate a resume. Writing a ‘0’ has no effect. A resume is initiated only if PortSuspendStatus is set.</td></tr>
<tr><td>2</td><td>R/W</td><td>R/W</td><td>0x0</td><td>(read)PortSuspendStatus
This bit indicates the port is suspended or in the resume sequence. It is set by a SetSuspendState write and cleared when</td></tr>
</table>

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

<table>
<tr><td colspan="3">Offset: 0x0454</td><td colspan="2">Register Name: HcRhPortStatus</td></tr>
<tr><td rowspan="2">Bit</td><td colspan="2">Read/Write</td><td rowspan="2">Default/Hex</td><td rowspan="2">Description</td></tr>
<tr><td>HCD</td><td>HC</td></tr>
<tr><td></td><td></td><td></td><td></td><td>PortSuspendStatusChange is set at the end of the resume interval. This bit cannot be set if CurrentConnectStatus is cleared. This bit is also cleared when PortResetStatusChange is set at the end of the port reset or when the HC is placed in the USBRESUME state. If an upstream resume is in progress, it should propagate to the HC.
<table><tr><td>0</td><td>port is not suspended</td></tr><tr><td>1</td><td>port is suspended</td></tr></table>
(write)SetPortSuspend
The HCD sets the PortSuspendStatus bit by writing a ‘1’ to this bit. Writing a ‘0’ has no effect. If CurrentConnectStatus is cleared, this write does not set PortSuspendStatus; instead it sets ConnectStatusChange. This informs the driver that it attempted to suspend a disconnected port.</td></tr>
<tr><td>1</td><td>R/W</td><td>R/W</td><td>0x0</td><td>(read)PortEnableStatus
This bit indicates whether the port is enabled or disabled. The Root Hub may clear this bit when an overcurrent condition, disconnect event, switched-off power, or operational bus error such as babble is detected. This change also causes PortEnabledStatusChange to be set. HCD sets this bit by writing SetPortEnable and clears it by writing ClearPortEnable. This bit cannot be set when CurrentConnectStatus is cleared. This bit is also set, if not already, at the completion of a port reset when ResetStatusChange is set or port suspend when SuspendStatusChange is set.
<table><tr><td>0</td><td>port is disabled</td></tr><tr><td>1</td><td>port is enabled</td></tr></table>
(write)SetPortEnable
The HCD sets PortEnableStatus by writing a ‘1’. Writing a ‘0’ has no effect. If CurrentConnectStatus is cleared, this write does not set PortEnableStatus, but instead sets ConnectStatusChange. This informs the driver that it attempted to enable a disconnected Port.</td></tr>
<tr><td>0</td><td>R/W</td><td>R/W</td><td>0x0</td><td>(read)CurrentConnectStatus
This bit reflects the current state of the downstream port.
<table><tr><td>0</td><td>No device connected</td></tr><tr><td>1</td><td>Device connected</td></tr></table>
(write)ClearPortEnable
The HCD writes a ‘1’ to clear the PortEnableStatus bit. Writing ‘0’</td></tr>
</table>

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0454				Register Name: HcRhPortStatus
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
				to this bit has no effect. The CurrentConnectStatus is not affected by any write. **Note: This bit is always read '1' when the attached device is nonremovalble (DviceRemoveable[NumberDownstreamPort]).**

7.6.5.23 0x0800 HCI Interface Register (Default Value: 0x1000_0000)

Offset: 0x0800			Register Name: USB_CTRL
Bit	Read/Write	Default/Hex	Description
31:29	/	/	Reserved
28	R	1	DMA Transfer Status Enable 0: Disable 1: Enable
27:26	/	/	/
25	R/W	0	OHCI count select 1: Simulation mode. The counters will be much shorter then real time 0: Normal mode. The counters will count full time
24:19	/	/	/
18	R/W	0	1: Within 2 us of the resume-K to SE0 transition 0: Random time value of the resume-K to SE0 transition
17:13	/	/	/
12	R/W	0	PP2VBUS 1: ULPI wrapper interface will automatically set or clear DrvVbus register in ULPI PHY according to the port power status form the root hub 0: ULPI wrapper will ignore the difference between power status of root hub and ULPI PHY
11	R/W	0	AHB Master interface INCR16 enable 1: Use INCR16 when appropriate 0: Do not use INCR16, use other enabled INCRX or unspecified length burst INCR
10	R/W	0	AHB Master interface INCR8 enable 1: Use INCR8 when appropriate 0: Do not use INCR8, use other enabled INCRX or unspecified length burst INCR
9	R/W	0	AHB Master interface burst type INCR4 enable 1: Use INCR4 when appropriate 0: Do not use INCR4, use other enabled INCRX or unspecified length burst INCR

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0800			Register Name: USB_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
8	R/W	0	AHB Master interface INCRX align enable 1: Start INCRx burst only on burst x-align address 0: Start burst on any double word boundary Note: This bit must enable if any bit of bit[11:9] is enabled
7:1	/	/	/
0	R/W	0	ULPI bypass enable 1: Enable UTMI interface, disable ULPI interface 0: Enable ULPI interface, disable UTMI interface

7.6.5.24 0x0808 HCI Control 3 Register (Default Value: 0x0001_0000)

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

7.6.5.25 0x0810 PHY Control Register (Default Value: 0x0000_0008)

Offset: 0x0810			Register Name: PHY_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0	bist_en_a
15:8	R/W	0	vc_addr
7	R/W	0	vc_di
6:4	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0810			Register Name: PHY_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
3	R/W	0x1	SIDDQ 1: Write 1 to disable phy 0: Write 0 to enable phy
2:1	/	/	/
0	R/W	0x0	vc_clk

7.6.5.26 0x0824 PHY Status Register (Default Value: 0x0000_0000)

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

7.6.5.27 0x0828 HCI SIE Port Disable Control Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7 GPIO

7.7.1 Overview

The general purpose input/output (GPIO) is one of the blocks controlling the chip multiplexing pins. The R329 supports 8 groups of GPIO pins. Each pin can be configured as input or output and these pins are used to generate input signals or output signals for special purposes.

The Port Controller has the following features:

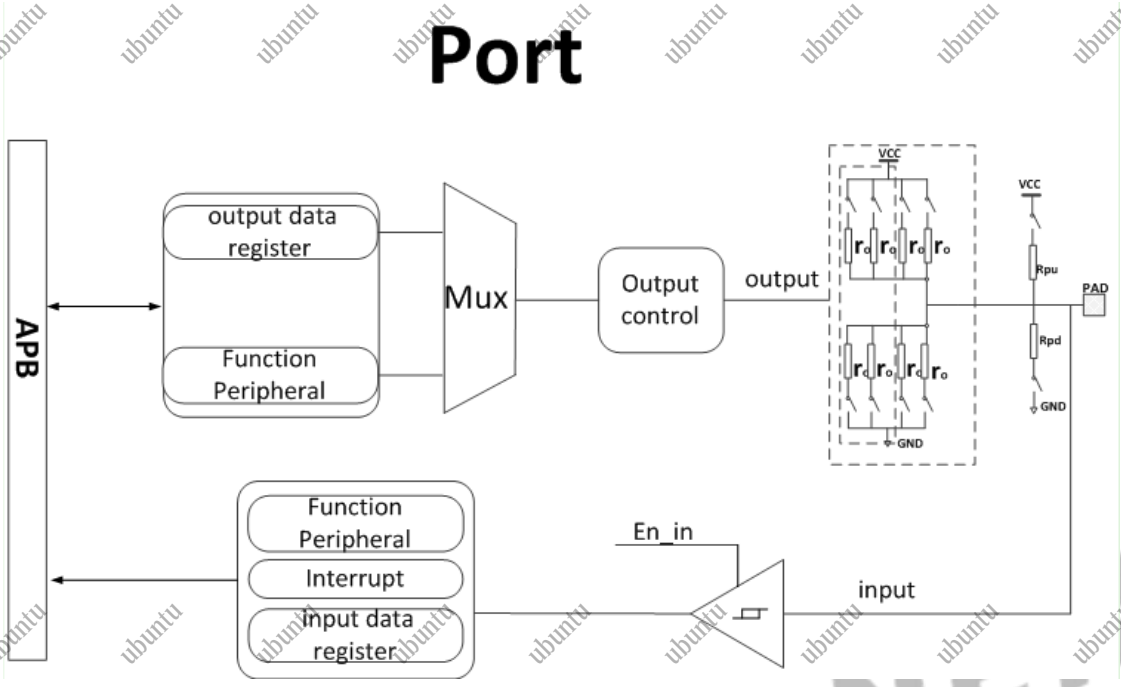
- 8 groups of ports (PB, PC, PF, PG, PH, PL, PM, PN)
- Software control for each signal pin
- Data input (capture)/output (drive)
- Each GPIO peripheral can produce an interrupt
- Pull-up/Pull-down/no-Pull register control
- Control the direction of every signal
- 4 drive strengths in each operating mode
- Up to 85 interrupts
- Configurable interrupt edges

7.7.2 Block Diagram

The following figure shows the block diagram of the GPIO.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-64 GPIO Block Diagram



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

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

7.7.3 Functional Descriptions

7.7.3.1 Multi-function Port

The R329 includes 93 multi-functional input/output port pins. There are 8 ports as listed below.

Table 7-23 Multi-function Port

Port Name	Number of Pins	Input Driver	Output Driver	Multiplex Pins	Power
PB	9	Schmitt	CMOS	UART/IR/PWM/I2S/JTAG/LEDC/TWI	3.3 V
PC	8	Schmitt	CMOS	NAND/SMHC/SPI/BOOT_SEL	1.8 V/3.3 V
PF	7	Schmitt	CMOS	NAND/SCR/UART/JTAG/OWA/SMHC	1.8 V/3.3 V
PG	15	Schmitt	CMOS	SMHC/UART/TWI/I2S/DBI/SPI	1.8 V/3.3 V

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Port Name	Number of Pins	Input Driver	Output Driver	Multiplex Pins	Power
PH	10	Schmitt	CMOS	TWI/UART/LEDC/OWA/SPI/DBI/TWI/ PWM/IR	3.3 V
PL	11	Schmitt	CMOS	R_I2S/R_TWI/R_UART/R_IR/R_DIMC/ X32KFOUT/R_PWM	1.8 V/3.3 V
PM	9	Schmitt	CMOS	R_UART/R_TWI/X32KFOUT/NMI/ R_JTAG/R_IR	1.8 V/3.3 V
PN	24	Schmitt	CMOS	EMAC	1.8 V/3.3 V

7.7.3.2 GPIO Multiplex Function

Table 7-24 to Table 7-31 show the multiplex function pins of the R329.

Table 7-24 PB Multiplex Function

GPIO Port	Function 2	Function 3	Function 4	Function 5	Function 6
PB0	UART2-TX	PWM0	JTAG-MS	LEDC-DO	PB-EINT0
PB1	UART2-RX	PWM1	JTAG-CK	I2S0-MCLK	PB-EINT1
PB2	UART2-RTS	PWM2	JTAG-DO	I2S0-LRCK	PB-EINT2
PB3	UART2-CTS	PWM3	JTAG-DI	I2S0-BCLK	PB-EINT3
PB4	UART0-TX	PWM4	I2S0-DOUT0	I2S0-DIN1	PB-EINT4
PB5	UART0-RX	PWM5	I2S0-DOUT1	I2S0-DIN0	PB-EINT5
PB6	IR-RX	PWM6	I2S0-DOUT2	TWI0-SCK	PB-EINT6
PB7	IR-TX	PWM7	I2S0-DOUT3	TWI0-SDA	PB-EINT7
PB8	IR-TX	PWM8	IR-RX	LEDC-DO	PB-EINT8

Table 7-25 PC Multiplex Function

GPIO Port	Function 2	Function 3	Function 4	Function 5	Function 6
PC0	NAND-RB0	SDC0-CLK	SPI0-CS0		
PC1	NAND-RE	SDC0-CMD	SPI0-MISO		
PC2	NAND-CE0	SDC0-D2	SPI0-WP		
PC3	NAND-CLE	SDC0-D1	SPI0-MOSI		
PC4	NAND-ALE	SDC0-D0	SPI0-CLK		
PC5	NAND-WE	SDC0-D3	SPI0-HOLD		
PC6	NAND-DQ0	SDC0-RST			

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

GPIO Port	Function 2	Function 3	Function 4	Function 5	Function 6
PC7	NAND-DQ1			BOOT-SEL	

Table 7-26 PF Multiplex Function

GPIO Port	Function 2	Function 3	Function 4	Function 5	Function 6
PF0	NAND-DQ7	SIM0-VPPEN	JTAG-MS	SDC0-D1	PF-EINT0
PF1	NAND-DQ6	SIM0-VPPPP	JTAG-DI	SDC0-D0	PF-EINT1
PF2	NAND-DQ5	SIM0-PWREN	UART0-TX	SDC0-CLK	PF-EINT2
PF3	NAND-DQ4	SIM0-CLK	JTAG-DO	SDC0-CMD	PF-EINT3
PF4	NAND-DQS	SIM0-DATA	UART0-RX	SDC0-D3	PF-EINT4
PF5	NAND-DQ3	SIM0-RST	JTAG-CK	SDC0-D2	PF-EINT5
PF6	NAND-DQ2	SIM0-DET	OWA-IN	OWA-OUT	PF-EINT6

Table 7-27 PG Multiplex Function

GPIO Port	Function 2	Function 3	Function 4	Function 5	Function 6
PG0	SDC1-CLK	SDC1-D2	SDC1-D2		PG-EINT0
PG1	SDC1-CMD	SDC1-D3	SDC1-CLK		PG-EINT1
PG2	SDC1-D0	SDC1-CMD	SDC1-D3	PLL-LOCK-DBG	PG-EINT2
PG3	SDC1-D1	SDC1-CLK	SDC1-D1		PG-EINT3
PG4	SDC1-D2	SDC1-D0	SDC1-D0		PG-EINT4
PG5	SDC1-D3	SDC1-D1	SDC1-CMD		PG-EINT5
PG6	UART1-TX	TWI0-SCK			PG-EINT6
PG7	UART1-RX	TWI0-SDA			PG-EINT7
PG8	UART1-RTS	TWI1-SCK		SPI1-HOLD/DBI-DCX/DBI-WRX	PG-EINT8
PG9	UART1-CTS	TWI1-SDA		SPI1-WP/DBI-TE	PG-EINT9
PG10		I2S1-MCLK	LEDC-DO		PG-EINT10
PG11	UART3-TX	I2S1-LRCK		SPI1-CS/DBI-CSX	PG-EINT11
PG12	UART3-RX	I2S1-BCLK		SPI1-CLK/DBI-SCLK	PG-EINT12
PG13	UART3-RTS	I2S1-DOUT0	I2S1-DIN1	SPI1-MOSI/DBI-SDO	PG-EINT13
PG14	UART3-CTS	I2S1-DOUT1	I2S1-DIN0	SPI1-MISO/DBI-SDI/DBI-TE/DBI-DCX	PG-EINT14

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Table 7-28 PH Multiplex Function

GPIO Port	Function 2	Function 3	Function 4	Function 5	Function 6
PH0	TWI0-SCK	UART0-TX	SPI1-CS/DBI-CSX	PWM0	PH-EINT0
PH1	TWI0-SDA	UART0-RX	SPI1-CLK/DBI-SCLK	PWM1	PH-EINT1
PH2	TWI1-SCK	LEDC-DO	SPI1-MOSI/DBI-SDO	IR-RX	PH-EINT2
PH3	TWI1-SDA	OWA-OUT	SPI1-MISO/ DBI-SDI/ DBI-TE/DBI-DCX	IR-TX	PH-EINT3
PH4	UART3-TX	SPI1-CS/DBI-CSX	SPI1-HOLD/ DBI-DCX/DBI-WRX	PWM2	PH-EINT4
PH5	UART3-RX	SPI1-CLK/DBI-SCLK	SPI1-WP/DBI-TE	PWM3	PH-EINT5
PH6	UART3-RTS	SPI1-MOSI/DBI-SDO	TWI0-SCK	PWM4	PH-EINT6
PH7	UART3-CTS	SPI1-MISO/DBI-SDI/ DBI-TE/DBI-DCX	TWI0-SDA	PWM5	PH-EINT7
PH8	TWI1-SCK	SPI1-HOLD/DBI-DCX/DBI-WRX	OWA-IN	IR-RX	PH-EINT8
PH9	TWI1-SDA	SPI1-WP/DBI-TE	LEDC-DO	IR-TX	PH-EINT9

Table 7-29 PL Multiplex Function

GPIO Port	Function 2	Function 3	Function 4	Function 5	Function 6
PL0	S-I2S0-LRCK		S-DMIC-DATA3	S-PWM0	PL-EINT0
PL1	S-I2S0-BCLK		S-DMIC-DATA2	S-PWM1	PL-EINT1
PL2	S-I2S0-DOUT0	S-I2S0-DIN1	S-DMIC-DATA1	S-PWM2	PL-EINT2
PL3	S-I2S0-DOUT1	S-I2S0-DIN0	S-DMIC-DATA0	S-TWI0-SDA	PL-EINT3
PL4	S-I2S0-MCLK	S-IR-RX	S-DMIC-CLK	S-TWI0-SCK	PL-EINT4
PL5	S-TWI0-SDA			S-PWM3	PL-EINT5
PL6	S-TWI0-SCK			S-PWM4	PL-EINT6
PL7	S-IR-RX		X32KFOUT	S-PWM5	PL-EINT7
PL8	S-UART0-TX	S-TWI0-SDA	S-IR-RX		PL-EINT8
PL9	S-UART0-RX	S-TWI0-SCK	X32KFOUT		PL-EINT9
PL10					PL-EINT10

Table 7-30 PM Multiplex Function

GPIO Port	Function 2	Function 3	Function 4	Function 5	Function 6
PM0	S-UART0-TX	S-JTAG-MS			PM-EINT0

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

GPIO Port	Function 2	Function 3	Function 4	Function 5	Function 6
PM1	S-UART0-RX	S-JTAG-CK			PM-EINT1
PM2		S-JTAG-DO	S-TWI0-SDA	S-IR-RX	PM-EINT2
PM3	S-TWI0-SDA	S-IR-RX			PM-EINT3
PM4	S-TWI0-SCK				PM-EINT4
PM5	X32KFOUT	S-JTAG-DI	S-TWI0-SCK		PM-EINT5
PM6	NMI	S-IR-RX	X32KFOUT		PM-EINT6
PM7	S-IR-RX	X32KFOUT			PM-EINT7
PM8					PM-EINT8

Table 7-31 PN Multiplex Function

GPIO Port	Function 2	Function 3	Function 4	Function 5	Function 6
PN0					PN-EINT0
PN1		MDC			PN-EINT1
PN2		MDIO			PN-EINT2
PN3		RGMII-TXD3			PN-EINT3
PN4		RGMII-TXCK/RMII-TXCK			PN-EINT4
PN5		RGMII-RXD2			PN-EINT5
PN6		RGMII-RXD0/RMII-RXD0			PN-EINT6
PN7		RGMII-RXCK			PN-EINT7
PN8		RMII-RXER			PN-EINT8
PN9		RGMII-TXCTL/RMII-TXEN			PN-EINT9
PN10		RGMII-RXD3			PN-EINT10
PN11		RGMII-RXD1/RMII-RXD1			PN-EINT11
PN12		RGMII-RXCTL/RMII-CRS-DV			PN-EINT12
PN13		RGMII-TXD2			PN-EINT13
PN14		RGMII-TXD1/RMII-TXD1			PN-EINT14
PN15		RGMII-TXD0/RMII-TXD0			PN-EINT15
PN16		EPHY-25M			PN-EINT16
PN17		RGMII-CLKIN			PN-EINT17
PN18					PN-EINT18
PN19					PN-EINT19
PN20					PN-EINT20
PN21					PN-EINT21
PN22					PN-EINT22
PN23					PN-EINT23

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.3.3 Port Function

The Port Controller supports 8 GPIOs, every GPIO can configure as Input, Output, Function Peripheral, IO disable or Interrupt function. The configuration instruction of every function is as follows.

Table 7-32 Port Function

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

/: non-configure, configuration is invalid

Y: configure

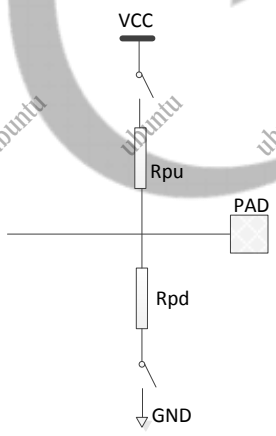
X: Select configuration according to the actual situation

N: Forbid to configure

7.7.3.4 Pull Up/Down and High-Impedance Logic

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

Figure 7-65 Pull up/down Logic



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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

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

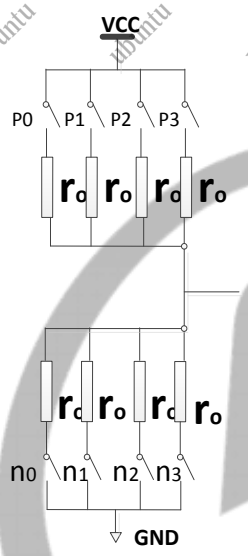
The pull-up/down of each IO is weak pull-up/down.

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

7.7.3.5 Buffer Strength

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

Figure 7-66 IO Buffer Strength Diagram



When output high level, the n0, n1, n2, n3 of NMOS is off, the p0, p1, p2, p3 of PMOS is on. When the buffer strength is set to 0 (buffer strength is weakest), only the p0 is on, the output impedance is maximum, the impedance value is r0. When the buffer strength is set to 1, only the p0 and p1 is on, the output impedance is equivalent to two r0 in parallel, the impedance value is r0/2. When the buffer strength is 2, only the p0, p1, and p2 is on, the output impedance is equivalent to three r0 in parallel, the impedance value is r0/3. When buffer strength is 3, the p0, p1, p2, and p3 is on, the output impedance is equivalent to four r0 in parallel, the impedance value is r0/4.

When output low level, the p0, p1, p2, p3 of PMOS is off, the n0, n1, n2, n3 of NMOS is on. When the buffer strength is set to 0 (buffer strength is weakest), only the n0 is on, the output impedance is maximum, the impedance value is r0. When the buffer strength is set to 1, only the n0 and n1 is on, the output impedance is equivalent to two r0 in parallel, the impedance value is r0/2. When the buffer strength is 2, only the n0, n1, and n2 is on, the output impedance is equivalent to three r0 in parallel, the impedance value is r0/3. When the

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

buffer strength is 3, the n0, n1, n2, and n3 is on, the output impedance is equivalent to four r0 in parallel, the impedance value is r0/4.

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

NOTE

The typical value of r0 is 180Ω.

7.7.3.6 Interrupt

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

The interrupt trigger of GPIO supports the following trigger types.

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

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

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

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

7.7.4 PB/PC/PF/PG/PH Register List

Module Name	Base Address
GPIO	0x02000400

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
Pn_CFG0	n*0x24+0x00	Port n Configure Register 0 (n from 1 to 9)
Pn_CFG1	n*0x24+0x04	Port n Configure Register 1 (n from 1 to 9)
Pn_CFG2	n*0x24+0x08	Port n Configure Register 2 (n from 1 to 9)
Pn_CFG3	n*0x24+0x0C	Port n Configure Register 3 (n from 1 to 9)
Pn_DAT	n*0x24+0x10	Port n Data Register (n from 1 to 9)
Pn_DRV0	n*0x24+0x14	Port n Multi_Driving Register 0 (n from 1 to 9)
Pn_DRV1	n*0x24+0x18	Port n Multi_Driving Register 1 (n from 1 to 9)
Pn_PUL0	n*0x24+0x1C	Port n Pull Register 0 (n from 1 to 9)
Pn_PUL1	n*0x24+0x20	Port n Pull Register 1 (n from 1 to 9)
Pn_INT_CFG0	0x200+n*0x20+0x00	PIO Interrupt Configure Register 0
Pn _INT_CFG1	0x200+n*0x20+0x04	PIO Interrupt Configure Register 1
Pn _INT_CFG2	0x200+n*0x20+0x08	PIO Interrupt Configure Register 2
Pn _INT_CFG3	0x200+n*0x20+0x0C	PIO Interrupt Configure Register 3
Pn _INT_CTL	0x200+n*0x20+0x10	PIO Interrupt Control Register
Pn _INT_STA	0x200+n*0x20+0x14	PIO Interrupt Status Register
Pn _INT_DEB	0x200+n*0x20+0x18	PIO Interrupt Debounce Register
PIO_POW_MOD_SEL	0x0340	PIO Group Withstand Voltage Mode Select Register
PIO_POW_MS_CTL	0x0344	PIO Group Withstand Voltage Mode Select Control Register
PIO_POW_Val	0x0348	PIO Group Power Value Register
PIO_POW_VOL_SET_CTL	0x0350	PIO Group Power Voltage Select Control Register

7.7.5 PB/PC/PF/PG/PH Register Description

7.7.5.1 0x0024 PB Configure Register 0 (Default Value: 0x7777_7777)

Offset: 0x0024			Register Name: PB_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31	/	/	/
30:28	R/W	0x7	PB7_SELECT 000: Input 001:Output 010: IR-TX 011: PWM7 100: I2S0-DOUT3 101: TWI0-SDA 110: PB-EINT7 111: IO Disable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0024			Register Name: PB_CFG0	
Bit	**Read/Write**	**Default/Hex**	**Description**	
27	/	/	/	
26:24	R/W	0x7	PB6_SELECT 000: Input 010: IR-RX 100: I2S0-DOUT2 110: PB-EINT6	 001:Output 011: PWM6 101: TWI0-SCK 111: IO Disable
23	/	/	/	
22:20	R/W	0x7	PB5_SELECT 000: Input 010: UART0-RX 100: I2S0-DOUT1 110: PB-EINT5	 001:Output 011: PWM5 101: I2S0-DIN0 111: IO Disable
19	/	/	/	
18:16	R/W	0x7	PB4_SELECT 000: Input 010: UART0-TX 100: I2S0-DOUT0 110: PB-EINT4	 001:Output 011: PWM4 101: I2S0-DIN1 111: IO Disable
15	/	/	/	
14:12	R/W	0x7	PB3_SELECT 000: Input 010: UART2-CTS 100: JTAG-DI 110: PB-EINT3	 001:Output 011: PWM3 101: I2S0-BCLK 111: IO Disable
11	/	/	/	
10:8	R/W	0x7	PB2_SELECT 000: Input 010: UART2-RTS 100: JTAG-DO 110: PB-EINT2	 001: Output 011: PWM2 101: I2S0-LRCK 111: IO Disable
7	/	/	/	
6:4	R/W	0x7	PB1_SELECT 000: Input 010: UART2-RX 100: JTAG-CK 110: PB-EINT1	 001:Output 011: PWM1 101: I2S0-MCLK 111: IO Disable
3	/	/	/	

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0024			Register Name: PB_CFG0	
Bit	Read/Write	Default/Hex	Description	
2:0	R/W	0x7	PB0_SELECT 000: Input 010: UART2-TX 100: JTAG-MS 110: PB-EINT0	001: Output 011: PWM0 101: LEDC-DO 111: IO Disable

7.7.5.2 0x0028 PB Configure Register 1 (Default Value: 0x0000_0007)

Offset: 0x0028			Register Name: PB_CFG1	
Bit	Read/Write	Default/Hex	Description	
31:3	/	/	/	
2:0	R/W	0x7	PB8_SELECT 000: Input 010: IR-TX 100: IR-RX 110: PB-EINT8	001: Output 011: PWM8 101: LEDC-DO 111: IO Disable

7.7.5.3 0x002C PB Configure Register 2 (Default Value: 0x0000_0000)

Offset: 0x002C			Register Name: PB_CFG2
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.5.4 0x0030 PB Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: PB_CFG3
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.5 0x0034 PB Data Register (Default Value: 0x0000_0000)

Offset: 0x0034			Register Name: PB_DAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:9	/	/	/
8:0	R/W	0x0	PB_DAT If the port is configured as the input function, the corresponding bit is the pin state. If the port is configured as the output function, the pin state is the same as the corresponding bit. The read bit value is the value set up by software. If the port is configured as a functional pin, the undefined value will be read.

7.7.5.6 0x0038 PB Multi_Driving Register 0 (Default Value: 0x0001_5555)

Offset: 0x0038			Register Name: PB_DRV0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17:16	R/W	0x1	PB8_DRV PB8 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
15:14	R/W	0x1	PB7_DRV PB7 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
13:12	R/W	0x1	PB6_DRV PB6 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
11:10	R/W	0x1	PB5_DRV PB5 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
9:8	R/W	0x1	PB4_DRV PB4 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0038			Register Name: PB_DRV0
Bit	**Read/Write**	**Default/Hex**	**Description**
7:6	R/W	0x1	PB3_DRV PB3 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
5:4	R/W	0x1	PB2_DRV PB2 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
3:2	R/W	0x1	PB1_DRV PB1 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
1:0	R/W	0x1	PB0_DRV PB0 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

7.7.5.7 0x003C PB Multi_Driving Register 1 (Default Value: 0x0000_0000)

Offset: 0x003C			Register Name: PB_DRV1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

7.7.5.8 0x0040 PB Pull Register 0 (Default Value: 0x0000_0000)

Offset: 0x0040			Register Name: PB_PULL0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17:16	R/W	0x0	PB8_PULL PB8 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0040			Register Name: PB_PULL0
Bit	**Read/Write**	**Default/Hex**	**Description**
15:14	R/W	0x0	PB7_PULL PB7 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
13:12	R/W	0x0	PB6_PULL PB6 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
11:10	R/W	0x0	PB5_PULL PB5 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
9:8	R/W	0x0	PB4_PULL PB4 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
7:6	R/W	0x0	PB3_PULL PB3 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
5:4	R/W	0x0	PB2_PULL PB2 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
3:2	R/W	0x0	PB1_PULL PB1 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
1:0	R/W	0x0	PB0_PULL PB0 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.9 0x0044 PB Pull Register 1 (Default Value: 0x0000_0000)

Offset: 0x0044			Register Name: PB_PULL1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

7.7.5.10 0x0048 PC Configure Register 0 (Default Value: 0x7777_7777)

Offset: 0x0048			Register Name: PC_CFG0	
Bit	**Read/Write**	**Default/Hex**	**Description**	
31	/	/	/	
30:28	R/W	0x7	PC7_SELECT 000: Input 010: NAND-DQ1 100: Reserved 110: Reserved	 001: Output 011: Reserved 101: BOOT-SEL 111: IO Disable
27	/	/	/	
26:24	R/W	0x7	PC6_SELECT 000: Input 010: NAND-DQ0 100: Reserved 110: Reserved	 001: Output 011: SDC0-RST 101: Reserved 111: IO Disable
23	/	/	/	
22:20	R/W	0x7	PC5_SELECT 000: Input 010: NAND-WE 100: SPI0-HOLD 110: Reserved	 001: Output 011: SDC0-D3 101: Reserved 111: IO Disable
19	/	/	/	
18:16	R/W	0x7	PC4_SELECT 000: Input 010: NAND-ALE 100: SPI0-CLK 110: Reserved	 001: Output 011: SDC0-D0 101: Reserved 111: IO Disable
15	/	/	/	

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0048			Register Name: PC_CFG0	
Bit	**Read/Write**	**Default/Hex**	**Description**	
14:12	R/W	0x7	PC3_SELECT 000: Input 010: NAND-CLE 100: SPI0-MOSI 110: Reserved	001: Output 011: SDC0-D1 101: Reserved 111: IO Disable
11	/	/	/	
10:8	R/W	0x7	PC2_SELECT 000: Input 010: NAND-CE0 100: SPI0-WP 110: Reserved	001:Output 011: SDC0-D2 101: Reserved 111: IO Disable
7	/	/	/	
6:4	R/W	0x7	PC1_SELECT 000: Input 010: NAND-RE 100: SPI0-MISO 110: Reserved	001:Output 011: SDC0-CMD 101: Reserved 111: IO Disable
3	/	/	/	
2:0	R/W	0x7	PC0_SELECT 000: Input 010:NAND-RB0 100: SPI0-CS0 110: Reserved	001: Output 011: SDC0-CLK 101: Reserved 111: IO Disable

7.7.5.11 0x004C PC Configure Register 1 (Default Value: 0x0000_0000)

Offset: 0x004C			Register Name: PC_CFG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

7.7.5.12 0x0050 PC Configure Register 2 (Default Value: 0x0000_0000)

Offset: 0x0050			Register Name: PC_CFG2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.13 0x0054 PC Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x0054			Register Name: PC_CFG3
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

7.7.5.14 0x0058 PC Data Register (Default Value: 0x0000_0000)

Offset: 0x0058			Register Name: PC_DAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R/W	0x0	PC_DAT If the port is configured as the input function, the corresponding bit is the pin state. If the port is configured as the output function, the pin state is the same as the corresponding bit. The read bit value is the value set up by software. If the port is configured as a functional pin, the undefined value will be read.

7.7.5.15 0x005C PC Multi_Driving Register 0 (Default Value: 0x0000_5555)

Offset: 0x005C			Register Name: PC_DRV0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:14	R/W	0x1	PC7_DRV PC7 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
13:12	R/W	0x1	PC6_DRV PC6 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x005C			Register Name: PC_DRV0
Bit	**Read/Write**	**Default/Hex**	**Description**
11:10	R/W	0x1	PC5_DRV PC5 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
9:8	R/W	0x1	PC4_DRV PC4 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
7:6	R/W	0x1	PC3_DRV PC3 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
5:4	R/W	0x1	PC2_DRV PC2 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
3:2	R/W	0x1	PC1_DRV PC1 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
1:0	R/W	0x1	PC0_DRV PC0 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

7.7.5.16 0x0060 PC Multi_Driving Register 1 (Default Value: 0x0000_0000)

Offset: 0x0060			Register Name: PC_DRV1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.17 0x0064 PC Pull Register 0 (Default Value: 0x0000_4411)

Offset: 0x0064			Register Name: PC_PULL0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:14	R/W	0x1	PC7_PULL PC7 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
13:12	R/W	0x0	PC6_PULL PC6 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
11:10	R/W	0x1	PC5_PULL PC5 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
9:8	R/W	0x0	PC4_PULL PC4 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
7:6	R/W	0x0	PC3_PULL PC3 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
5:4	R/W	0x1	PC2_PULL PC2 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
3:2	R/W	0x0	PC1_PULL PC Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
1:0	R/W	0x1	PC0_PULL PC0 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.18 0x0068 PC Pull Register 1 (Default Value: 0x0000_0000)

Offset: 0x0068			Register Name: PC_PULL1
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.5.19 0x00B4 PF Configure Register 0 (Default Value: 0x0777_7777)

Offset: 0x00B4			Register Name: PF_CFG0
Bit	Read/Write	Default/Hex	Description
31:27	/	/	/
26:24	R/W	0x7	PF6_SELECT 000: Input 001: Output 010: NAND-DQ2 011: SIM0-DET 100: OWA-IN 101: OWA-OUT 110: PF-EINT6 111: IO Disable
23	/	/	/
22:20	R/W	0x7	PF5_SELECT 000: Input 001: Output 010: NAND-DQ3 011: SIM0-RST 100: JTAG-CK 101: SDC0-D2 110: PF-EINT5 111: IO Disable
19	/	/	/
18:16	R/W	0x7	PF4_SELECT 000: Input 001: Output 010: NAND-DQS 011: SIM0-DATA 100: UART0-RX 101: SDC0-D3 110: PF-EINT4 111: IO Disable
15	/	/	/
14:12	R/W	0x7	PF3_SELECT 000: Input 001: Output 010: NAND-DQ4 011: SIM0-CLK 100: JTAG-DO 101: SDC0-CMD 110: PF-EINT3 111: IO Disable
11	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x00B4			Register Name: PF_CFG0
Bit	Read/Write	Default/Hex	Description
10:8	R/W	0x7	PF2_SELECT 000: Input 001: Output 010: NAND-DQ5 011: SIM0-PWREN 100: UART0-TX 101: SDC0-CLK 110: PF-EINT2 111: IO Disable
7	/	/	/
6:4	R/W	0x7	PF1_SELECT 000: Input 001: Output 010: NAND-DQ6 011: SIM0-VPPPP 100: JTAG-DI 101: SDC0-D0 110: PF-EINT1 111: IO Disable
3	/	/	/
2:0	R/W	0x7	PF0_SELECT 000: Input 001: Output 010: NAND-DQ7 011: SIM0-VPPEN 100: JTAG-MS 101: SDC0-D1 110: PF-EINT0 111: IO Disable

7.7.5.20 0x00B8 PF Configure Register 1 (Default Value: 0x0000_0000)

Offset: 0x00B8			Register Name: PF_CFG1
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.5.21 0x00BC PF Configure Register 2 (Default Value: 0x0000_0000)

Offset: 0x00BC			Register Name: PF_CFG2
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.22 0x00C0 PF Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x00C0			Register Name: PF_CFG3
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.5.23 0x00C4 PF Data Register (Default Value: 0x0000_0000)

Offset: 0x00C4			Register Name: PF_DAT
Bit	Read/Write	Default/Hex	Description
31:7	/	/	/
6:0	R/W	0x0	PF_DAT If the port is configured as the input function, the corresponding bit is the pin state. If the port is configured as the output function, the pin state is the same as the corresponding bit. The read bit value is the value set up by software. If the port is configured as a functional pin, the undefined value will be read.

7.7.5.24 0x00C8 PF Multi_Driving Register 0 (Default Value: 0x0000_1555)

Offset: 0x00C8			Register Name: PF_DRV0
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13:12	R/W	0x1	PF6_DRV PF6 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
11:10	R/W	0x1	PF5_DRV PF5 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
9:8	R/W	0x1	PF4_DRV PF4 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x00C8			Register Name: PF_DRV0
Bit	Read/Write	Default/Hex	Description
7:6	R/W	0x1	PF3_DRV PF3 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
5:4	R/W	0x1	PF2_DRV PF2 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
3:2	R/W	0x1	PF1_DRV PF1 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
1:0	R/W	0x1	PF0_DRV PF0 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

7.7.5.25 0x00CC PF Multi_Driving Register 1 (Default Value: 0x0000_0000)

Offset: 0x00CC			Register Name: PF_DRV1
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.5.26 0x00D0 PF Pull Register 0 (Default Value: 0x0000_0000)

Offset: 0x00D0			Register Name: PF_PULL0
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13:12	R/W	0x0	PF6_PULL PF6 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x00D0			Register Name: PF_PULL0
Bit	Read/Write	Default/Hex	Description
11:10	R/W	0x0	PF5_PULL PF5 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
9:8	R/W	0x0	PF4_PULL PF4 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
7:6	R/W	0x0	PF3_PULL PF3 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
5:4	R/W	0x0	PF2_PULL PF2 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
3:2	R/W	0x0	PF1_PULL PF1 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
1:0	R/W	0x0	PF0_PULL PF0 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

7.7.5.27 0x00D4 PF Pull Register 1 (Default Value: 0x0000_0000)

Offset: 0x00D4			Register Name: PF_PULL1
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.28 0x00D8 PG Configure Register 0 (Default Value: 0x7777_7777)

Offset: 0x00D8			Register Name: PG_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31	/	/	/
30:28	R/W	0x7	PG7_SELECT 000: Input 001: Output 010: UART1-RX 011: TWI0-SDA 100: Reserved 101: Reserved 110: PG-EINT7 111: IO Disable
27	/	/	/
26:24	R/W	0x7	PG6_SELECT 000: Input 001: Output 010: UART1-TX 011: TWI0-SCK 100: Reserved 101: Reserved 110: PG-EINT6 111: IO Disable
23	/	/	/
22:20	R/W	0x7	PG5_SELECT 000: Input 001: Output 010: SDC1-D3 011: SDC1-D1 100: SDC1-CMD 101: Reserved 110: PG-EINT5 111: IO Disable
19	/	/	/
18:16	R/W	0x7	PG4_SELECT 000: Input 001: Output 010: SDC1-D2 011: SDC1-D0 100: SDC1-D0 101: Reserved 110: PG-EINT4 111: IO Disable
15	/	/	/
14:12	R/W	0x7	PG3_SELECT 000: Input 001: Output 010: SDC1-D1 011: SDC1-CLK 100: SDC1-D1 101: Reserved 110: PG-EINT3 111: IO Disable
11	/	/	/
10:8	R/W	0x7	PG2_SELECT 000: Input 001: Output 010: SDC1-D0 011: SDC1-CMD 100: SDC1-D3 101: PLL-LOCK-DBG 110: PG-EINT2 111: IO Disable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x00D8			Register Name: PG_CFG0
Bit	Read/Write	Default/Hex	Description
7	/	/	/
6:4	R/W	0x7	PG1_SELECT 000: Input 001: Output 010: SDC1-CMD 011: SDC1-D3 100: SDC1-CLK 101: Reserved 110: PG-EINT1 111: IO Disable
3	/	/	/
2:0	R/W	0x7	PG0_SELECT 000: Input 001: Output 010: SDC1-CLK 011: SDC1-D2 100: SDC1-D2 101: Reserved 110: PG-EINT0 111: IO Disable

7.7.5.29 0x00DC PG Configure Register 1 (Default Value: 0x0777_7777)

Offset: 0x00DC			Register Name: PG_CFG1
Bit	Read/Write	Default/Hex	Description
31:26	/	/	/
25:24	R/W	0x7	PG14_SELECT 000: Input 001: Output 010: UART3-CTS 011: I2S1-DOUT1 100: I2S1-DIN0 101: SPI1-MISO/DBI-SDI/DBI-TE/DBI-DCX 110: PG_EINT13 111: IO Disable
23	/	/	/
22:20	R/W	0x7	PG13_SELECT 000: Input 001: Output 010: UART3-RTS 011: I2S1-DOUT0 100: I2S1-DIN1 101: SPI1-MOSI/DBI-SDO 110: PG-EINT13 111: IO Disable
19	/	/	/
18:16	R/W	0x7	PG12_SELECT 000: Input 001: Output 010: UART3-RX 011: I2S1-BCLK 100: Reserved 101: SPI1-CLK/DBI-SCLK 110: PG-EINT12 111: IO Disable
15	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x00DC			Register Name: PG_CFG1
Bit	Read/Write	Default/Hex	Description
14:12	R/W	0x7	PG11_SELECT 000: Input 001: Output 010: UART3-TX 011: I2S1-LRCK 100: Reserved 101: SPI1-CS/DBI-CSX 110: PG-EINT11 111: IO Disable
11	/	/	/
10:8	R/W	0x7	PG10_SELECT 000: Input 001: Output 010: Reserved 011: I2S1-MCLK 100: LEDC-DO 101: Reserved 110: PG-EINT10 111: IO Disable
7	/	/	/
6:4	R/W	0x7	PG9_SELECT 000: Input 001: Output 010: UART1-CTS 011: TWI1-SDA 100: Reserved 101: SPI1-WP/DBI-TE 110: PG-EINT9 111: IO Disable
3	/	/	/
2:0	R/W	0x7	PG8_SELECT 000: Input 001: Output 010: UART1-RTS 011: TWI1-SCK 100: Reserved 101: SPI1-HOLD/DBI-DCX/DBI-WRX 110: PG-EINT8 111: IO Disable

7.7.5.30 0x00E0 PG Configure Register 2 (Default Value: 0x0000_0000)

Offset: 0x00E0			Register Name: PG_CFG2
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.5.31 0x00E4 PG Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x00E4			Register Name: PG_CFG3
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.32 0x00E8 PG Data Register (Default Value: 0x0000_0000)

Offset: 0x00E8			Register Name: PG_DAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:15	/	/	/
14:0	R/W	0x0	PG_DAT If the port is configured as the input function, the corresponding bit is the pin state. If the port is configured as the output function, the pin state is the same as the corresponding bit. The read bit value is the value set up by software. If the port is configured as a functional pin, the undefined value will be read.

7.7.5.33 0x00EC PG Multi_Driving Register 0 (Default Value: 0x1555_5555)

Offset: 0x00EC			Register Name: PG_DRV0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/
29:28	R/W	0x1	PG14_DRV PG14 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
27:26	R/W	0x1	PG13_DRV PG13 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
25:24	R/W	0x1	PG12_DRV PG12 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
23:22	R/W	0x1	PG11_DRV PG11 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x00EC			Register Name: PG_DRV0
Bit	**Read/Write**	**Default/Hex**	**Description**
21:20	R/W	0x1	PG10_DRV PG10 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
19:18	R/W	0x1	PG9_DRV PG9 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
17:16	R/W	0x1	PG8_DRV PG8 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
15:14	R/W	0x1	PG7_DRV PG7 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
13:12	R/W	0x1	PG6_DRV PG6 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
11:10	R/W	0x1	PG5_DRV PG5 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
9:8	R/W	0x1	PG4_DRV PG4 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
7:6	R/W	0x1	PG3_DRV PG3 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
5:4	R/W	0x1	PG2_DRV PG2 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x00EC			Register Name: PG_DRV0
Bit	**Read/Write**	**Default/Hex**	**Description**
3:2	R/W	0x1	PG1_DRV PG1 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
1:0	R/W	0x1	PG0_DRV PG0 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

7.7.5.34 0x00F0 PG Multi_Driving Register 1 (Default Value: 0x0000_0000)

Offset: 0x00F0			Register Name: PG_DRV1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

7.7.5.35 0x00F4 PG Pull Register 0 (Default Value: 0x0000_0000)

Offset: 0x00F4			Register Name: PG_PULL0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/
29:28	R/W	0x0	PG14_PULL PG14 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
27:26	R/W	0x0	PG13_PULL PG13 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
25:24	R/W	0x0	PG12_PULL PG12 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x00F4			Register Name: PG_PULL0
Bit	**Read/Write**	**Default/Hex**	**Description**
23:22	R/W	0x0	PG11_PULL PG11 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
21:20	R/W	0x0	PG10_PULL PG10 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
19:18	R/W	0x0	PG9_PULL PG9 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
17:16	R/W	0x0	PG8_PULL PG8 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
15:14	R/W	0x0	PG7_PULL PG7 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
13:12	R/W	0x0	PG6_PULL PG6 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
11:10	R/W	0x0	PG5_PULL PG5 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
9:8	R/W	0x0	PG4_PULL PG4 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
7:6	R/W	0x0	PG3_PULL PG3 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x00F4			Register Name: PG_PULL0
Bit	Read/Write	Default/Hex	Description
5:4	R/W	0x0	PG2_PULL PG2 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
3:2	R/W	0x0	PG1_PULL PG1 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
1:0	R/W	0x0	PG0_PULL PG0 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

7.7.5.36 0x00F8 PG Pull Register 1 (Default Value: 0x0000_0000)

Offset: 0x00F8			Register Name: PG_PULL1
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.5.37 0x00FC PH Configure Register 0 (Default Value: 0x7777_7777)

Offset: 0x00FC			Register Name: PH_CFG0
Bit	Read/Write	Default/Hex	Description
31	/	/	/
30:28	R/W	0x7	PH7_SELECT 000: Input 001: Output 010: UART3-CTS 011: SPI1-MISO/DBI-SDI/DBI-TE/DBI-DCX 100: TWI0-SDA 101: PWM5 110: PH-EINT7 111: IO Disable
27	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x00FC			Register Name: PH_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
26:24	R/W	0x7	PH6_SELECT 000: Input 001: Output 010: UART3-RTS 011: SPI1-MOSI/DBI-SDO 100: TWI0-SCK 101: PWM4 110: PH-EINT6 111: IO Disable
23	/	/	/
22:20	R/W	0x7	PH5_SELECT 000: Input 001: Output 010: UART3-RX 011: SPI1-CLK/DBI-SCLK 100: SPI1-WP/DBI-TE 101: PWM3 110: PH-EINT5 111: IO Disable
19	/	/	/
18:16	R/W	0x7	PH4_SELECT 000: Input 001: Output 010: UART3-TX 011: SPI1-CS/DBI-CSX 100: SPI1-HOLD/DBI-DCX/DBI-WRX 101: PWM2 110: PH-EINT4 111: IO Disable
15	/	/	/
14:12	R/W	0x7	PH3_SELECT 000: Input 001: Output 010:TWI1-SDA 011: OWA-OUT 100: SPI1-MISO/DBI-SDI/DBI-TE/DBI-DCX 101: IR-TX 110: PH-EINT3 111: IO Disable
11	/	/	/
10:8	R/W	0x7	PH2_SELECT 000: Input 001: Output 010: TWI1-SCK 011: LEDC-DO 100: SPI1-MOSI/DBI-SDO 101: IR-RX 110: PH-EINT2 111: IO Disable
7	/	/	/
6:4	R/W	0x7	PH1_SELECT 000: Input 001: Output 010: TWI0-SDA 011: UART0-RX 100: SPI1-CLK/DBI-SCLK 101: PWM1 110: PH-EINT1 111: IO Disable
3	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x00FC			Register Name: PH_CFG0
Bit	Read/Write	Default/Hex	Description
2:0	R/W	0x7	PH0_SELECT 000: Input 001: Output 010: TWI0-SCK 011: UART0-TX 100: SPI1-CS/DBI-CSX 101: PWM0 110: PH-EINT0 111: IO Disable

7.7.5.38 0x0100 PH Configure Register 1 (Default Value: 0x0000_0077)

Offset: 0x0100			Register Name: PH_CFG1
Bit	Read/Write	Default/Hex	Description
31:7	/	/	/
6:4	R/W	0x7	PH9_SELECT 000: Input 001: Output 010: TWI1-SDA 011: SPI1-WP/DBI-TE 100: LEDC-DO 101: IR-TX 110: PH-EINT9 111: IO Disable
3	/	/	/
2:0	R/W	0x7	PH8_SELECT 000: Input 001: Output 010: TWI1-SCK 011: SPI1-HOLD/DBI-DCX/DBI-RX 100: OWA-IN 101: IR-RX 110: PH-EINT8 111: IO Disable

7.7.5.39 0x0104 PH Configure Register 2 (Default Value: 0x0000_0000)

Offset: 0x0104			Register Name: PH_CFG2
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.5.40 0x0108 PH Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x108			Register Name: PH_CFG3
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.41 0x010C PH Data Register (Default Value: 0x0000_0000)

Offset: 0x10C			Register Name: PH_DAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:10	/	/	/
9:0	R/W	0x0	PH_DAT If the port is configured as the input function, the corresponding bit is the pin state. If the port is configured as the output function, the pin state is the same as the corresponding bit. The read bit value is the value set up by software. If the port is configured as a functional pin, the undefined value will be read.

7.7.5.42 0x0110 PH Multi_Driving Register 0 (Default Value: 0x0005_5555)

Offset: 0x0110			Register Name: PH_DRV0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:20	/	/	/
19:18	R/W	0x1	PH9_DRV PH9 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
17:16	R/W	0x1	PH8_DRV PH8 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
15:14	R/W	0x1	PH7_DRV PH7 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
13:12	R/W	0x1	PH6_DRV PH6 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0110			Register Name: PH_DRV0
Bit	Read/Write	Default/Hex	Description
11:10	R/W	0x1	PH5_DRV PH5 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
9:8	R/W	0x1	PH4_DRV PH4 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
7:6	R/W	0x1	PH3_DRV PH3 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
5:4	R/W	0x1	PH2_DRV PH2 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
3:2	R/W	0x1	PH1_DRV PH1 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
1:0	R/W	0x1	PH0_DRV PH0 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

7.7.5.43 0x0114 PH Multi_Driving Register 1 (Default Value: 0x0000_0000)

Offset: 0x0114			Register Name: PH_DRV1
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.44 0x0118 PH Pull Register 0 (Default Value: 0x0000_0000)

Offset: 0x0118			Register Name: PH_PULL0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:20	/	/	/
19:18	R/W	0x0	PH9_PULL PH9 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
17:16	R/W	0x0	PH8_PULL PH8 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
15:14	R/W	0x0	PH7_PULL PH7 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
13:12	R/W	0x0	PH6_PULL PH6 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
11:10	R/W	0x0	PH5_PULL PH5 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
9:8	R/W	0x0	PH4_PULL PH4 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
7:6	R/W	0x0	PH3_PULL PH3 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
5:4	R/W	0x0	PH2_PULL PH2 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0118			Register Name: PH_PULL0
Bit	**Read/Write**	**Default/Hex**	**Description**
3:2	R/W	0x0	PH1_PULL PH1 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
1:0	R/W	0x0	PH0_PULL PH0 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

7.7.5.45 0x011C PH Pull Register 1 (Default Value: 0x0000_0000)

Offset: 0x011C			Register Name: PH_PULL1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

7.7.5.46 0x0220 PB External Interrupt Configure Register 0 (Default Value: 0x0000_0000)

Offset: 0x0220			Register Name:PB_EINT_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x0	EINT7_CFG External INT7 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0220			Register Name:PB_EINT_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
27:24	R/W	0x0	EINT6_CFG External INT6 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
23:20	R/W	0x0	EINT5_CFG External INT5 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
19:16	R/W	0x0	EINT4_CFG External INT4 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
15:12	R/W	0x0	EINT3_CFG External INT3 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0220			Register Name:PB_EINT_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
11:8	R/W	0x0	EINT2_CFG External INT2 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
7:4	R/W	0x0	EINT1_CFG External INT1 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
3:0	R/W	0x0	EINT0_CFG External INT0 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

7.7.5.47 0x0224 PB External Interrupt Configure Register 1 (Default Value: 0x0000_0000)

Offset: 0x0224			Register Name: PB_EINT_CFG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:4	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0224			Register Name: PB_EINT_CFG1
Bit	Read/Write	Default/Hex	Description
3:0	R/W	0x0	EINT8_CFG External INT8 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

7.7.5.48 0x0228 PB External Interrupt Configure Register 2 (Default Value: 0x0000_0000)

Offset: 0x0228			Register Name: PB_EINT_CFG2
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.5.49 0x022C PB External Interrupt Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x022C			Register Name: PB_EINT_CFG3
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.5.50 0x0230 PB External Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0230			Register Name: PB_EINT_CTL
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8	R/W	0x0	EINT8_CTL External INT8 Enable 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0230			Register Name: PB_EINT_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
7	R/W	0x0	EINT7_CTL External INT7 Enable 0: Disable 1: Enable
6	R/W	0x0	EINT6_CTL External INT6 Enable 0: Disable 1: Enable
5	R/W	0x0	EINT5_CTL External INT5 Enable 0: Disable 1: Enable
4	R/W	0x0	EINT4_CTL External INT4 Enable 0: Disable 1: Enable
3	R/W	0x0	EINT3_CTL External INT3 Enable 0: Disable 1: Enable
2	R/W	0x0	EINT2_CTL External INT2 Enable 0: Disable 1: Enable
1	R/W	0x0	EINT1_CTL External INT1 Enable 0: Disable 1: Enable
0	R/W	0x0	EINT0_CTL External INT0 Enable 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.51 0x0234 PB External Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0234			Register Name: PB_EINT_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:9	/	/	/
8	R/W	0x0	EINT8_STATUS External INT8 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
7	R/W	0x0	EINT7_STATUS External INT7 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
6	R/W	0x0	EINT6_STATUS External INT6 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
5	R/W	0x0	EINT5_STATUS External INT5 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
4	R/W	0x0	EINT4_STATUS External INT4 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
3	R/W	0x0	EINT3_STATUS External INT3 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0234			Register Name: PB_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
2	R/W	0x0	EINT2_STATUS External INT2 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
1	R/W	0x0	EINT1_STATUS External INT1 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
0	R/W	0x0	EINT0_STATUS External INT0 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

7.7.5.52 0x0238 PB External Interrupt Debounce Register (Default Value: 0x0000_0000)

Offset: 0x0238			Register Name: PB_EINT_DEB
Bit	Read/Write	Default/Hex	Description
31:7	/	/	/
6:4	R/W	0x0	DEB_CLK_PRE_SCALE Debounce Clock Pre_scale n The selected clock source is prescaled by 2^n.
3:1	/	/	/
0	R/W	0x0	PIO_INT_CLK_SELECT PIO Interrupt Clock Select 0: LOSC 32KHz 1: HOSC 24MHz

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.53 0x02A0 PF External Interrupt Configure Register 0 (Default Value: 0x0000_0000)

Offset: 0x02A0			Register Name:PF_EINT_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	/	/	/
27:24	R/W	0x0	EINT6_CFG External INT6 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
23:20	R/W	0x0	EINT5_CFG External INT5 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
19:16	R/W	0x0	EINT4_CFG External INT4 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
15:12	R/W	0x0	EINT3_CFG External INT3 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x02A0			Register Name:PF_EINT_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
11:8	R/W	0x0	EINT2_CFG External INT2 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
7:4	R/W	0x0	EINT1_CFG External INT1 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
3:0	R/W	0x0	EINT0_CFG External INT0 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

7.7.5.54 0x02A4 PF External Interrupt Configure Register 1 (Default Value: 0x0000_0000)

Offset: 0x02A4			Register Name: PF_EINT_CFG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.55 0x02A8 PF External Interrupt Configure Register 2 (Default Value: 0x0000_0000)

Offset: 0x02A8			Register Name: PF_EINT_CFG2
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.5.56 0x02AC PF External Interrupt Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x02AC			Register Name: PF_EINT_CFG3
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.5.57 0x02B0 PF External Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x02B0			Register Name: PF_EINT_CTL
Bit	Read/Write	Default/Hex	Description
31:7	/	/	/
6	R/W	0x0	EINT6_CTL External INT6 Enable 0: Disable 1: Enable
5	R/W	0x0	EINT5_CTL External INT5 Enable 0: Disable 1: Enable
4	R/W	0x0	EINT4_CTL External INT4 Enable 0: Disable 1: Enable
3	R/W	0x0	EINT3_CTL External INT3 Enable 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x02B0			Register Name: PF_EINT_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
2	R/W	0x0	EINT2_CTL External INT2 Enable 0: Disable 1: Enable
1	R/W	0x0	EINT1_CTL External INT1 Enable 0: Disable 1: Enable
0	R/W	0x0	EINT0_CTL External INT0 Enable 0: Disable 1: Enable

7.7.5.58 0x02B4 PF External Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x02B4			Register Name: PF_EINT_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:7	/	/	/
6	R/W	0x0	EINT6_STATUS External INT6 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
5	R/W	0x0	EINT5_STATUS External INT5 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
4	R/W	0x0	EINT4_STATUS External INT4 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x02B4			Register Name: PF_EINT_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
3	R/W	0x0	EINT3_STATUS External INT3 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
2	R/W	0x0	EINT2_STATUS External INT2 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
1	R/W	0x0	EINT1_STATUS External INT1 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
0	R/W	0x0	EINT0_STATUS External INT0 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

7.7.5.59 0x02B8 PF External Interrupt Debounce Register (Default Value: 0x0000_0000)

Offset: 0x02B8			Register Name: PF_EINT_DEB
Bit	**Read/Write**	**Default/Hex**	**Description**
31:7	/	/	/
6:4	R/W	0x0	DEB_CLK_PRE_SCALE Debounce Clock Pre_scale n The selected clock source is prescaled by 2^n.
3:1	/	/	/
0	R/W	0x0	PIO_INT_CLK_SELECT PIO Interrupt Clock Select 0: LOSC 32KHz 1: HOSC 24MHz

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.60 0x02C0 PG External Interrupt Configure Register 0 (Default Value: 0x0000_0000)

Offset: 0x02C0			Register Name:PG_EINT_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x0	EINT7_CFG External INT7 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
27:24	R/W	0x0	EINT6_CFG External INT6 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
23:20	R/W	0x0	EINT5_CFG External INT5 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
19:16	R/W	0x0	EINT4_CFG External INT4 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x02C0			Register Name:PG_EINT_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
15:12	R/W	0x0	EINT3_CFG External INT3 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
11:8	R/W	0x0	EINT2_CFG External INT2 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
7:4	R/W	0x0	EINT1_CFG External INT1 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
3:0	R/W	0x0	EINT0_CFG External INT0 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.61 0x02C4 PG External Interrupt Configure Register 1 (Default Value: 0x0000_0000)

Offset: 0x02C4			Register Name: PG_EINT_CFG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	/	/	/
27:24	R/W	0x0	EINT14_CFG External INT14 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
23:20	R/W	0x0	EINT13_CFG External INT13 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
19:16	R/W	0x0	EINT12_CFG External INT12 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
15:12	R/W	0x0	EINT11_CFG External INT11 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x02C4			Register Name: PG_EINT_CFG1
Bit	**Read/Write**	**Default/Hex**	**Description**
11:8	R/W	0x0	EINT10_CFG External INT10 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
7:4	R/W	0x0	EINT9_CFG External INT9 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
3:0	R/W	0x0	EINT8_CFG External INT8 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

7.7.5.62 0x02C8 PG External Interrupt Configure Register 2 (Default Value: 0x0000_0000)

Offset: 0x02C8			Register Name: PG_EINT_CFG2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.63 0x02CC PG External Interrupt Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x02CC			Register Name: PG_EINT_CFG3
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.5.64 0x02D0 PG External Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x02D0			Register Name: PG_EINT_CTL
Bit	Read/Write	Default/Hex	Description
31:15	/	/	/
14	R/W	0x0	EINT14_CTL External INT14 Enable 0: Disable 1: Enable
13	R/W	0x0	EINT13_CTL External INT13 Enable 0: Disable 1: Enable
12	R/W	0x0	EINT12_CTL External INT12 Enable 0: Disable 1: Enable
11	R/W	0x0	EINT11_CTL External INT11 Enable 0: Disable 1: Enable
10	R/W	0x0	EINT10_CTL External INT10 Enable 0: Disable 1: Enable
9	R/W	0x0	EINT9_CTL External INT9 Enable 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x02D0			Register Name: PG_EINT_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
8	R/W	0x0	EINT8_CTL External INT8 Enable 0: Disable 1: Enable
7	R/W	0x0	EINT7_CTL External INT7 Enable 0: Disable 1: Enable
6	R/W	0x0	EINT6_CTL External INT6 Enable 0: Disable 1: Enable
5	R/W	0x0	EINT5_CTL External INT5 Enable 0: Disable 1: Enable
4	R/W	0x0	EINT4_CTL External INT4 Enable 0: Disable 1: Enable
3	R/W	0x0	EINT3_CTL External INT3 Enable 0: Disable 1: Enable
2	R/W	0x0	EINT2_CTL External INT2 Enable 0: Disable 1: Enable
1	R/W	0x0	EINT1_CTL External INT1 Enable 0: Disable 1: Enable
0	R/W	0x0	EINT0_CTL External INT0 Enable 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.65 0x02D4 PG External Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x02D4			Register Name: PG_EINT_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:15	/	/	/
14	R/W	0x0	EINT14_STATUS External INT14 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
13	R/W	0x0	EINT13_STATUS External INT13 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
12	R/W	0x0	EINT12_STATUS External INT12 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
11	R/W	0x0	EINT11_STATUS External INT11 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
10	R/W	0x0	EINT10_STATUS External INT10 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
9	R/W	0x0	EINT9_STATUS External INT9 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x02D4			Register Name: PG_EINT_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
8	R/W	0x0	EINT8_STATUS External INT8 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
7	R/W	0x0	EINT7_STATUS External INT7 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
6	R/W	0x0	EINT6_STATUS External INT6 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
5	R/W	0x0	EINT5_STATUS External INT5 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
4	R/W	0x0	EINT4_STATUS External INT4 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
3	R/W	0x0	EINT3_STATUS External INT3 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
2	R/W	0x0	EINT2_STATUS External INT2 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x02D4			Register Name: PG_EINT_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
1	R/W	0x0	EINT1_STATUS External INT1 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
0	R/W	0x0	EINT0_STATUS External INT0 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

7.7.5.66 0x02D8 PG External Interrupt Debounce Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.67 0x02E0 PH External Interrupt Configure Register 0 (Default Value: 0x0000_0000)

Offset: 0x02E0			Register Name:PH_EINT_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x0	EINT7_CFG External INT7 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
27:24	R/W	0x0	EINT6_CFG External INT6 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
23:20	R/W	0x0	EINT5_CFG External INT5 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
19:16	R/W	0x0	EINT4_CFG External INT4 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x02E0			Register Name:PH_EINT_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
15:12	R/W	0x0	EINT3_CFG External INT3 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
11:8	R/W	0x0	EINT2_CFG External INT2 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
7:4	R/W	0x0	EINT1_CFG External INT1 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
3:0	R/W	0x0	EINT0_CFG External INT0 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.68 0x02E4 PH External Interrupt Configure Register 1 (Default Value: 0x0000_0000)

Offset: 0x02E4			Register Name: PH_EINT_CFG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:4	R/W	0x0	EINT9_CFG External INT9 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
3:0	R/W	0x0	EINT8_CFG External INT8 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

7.7.5.69 0x02E8 PH External Interrupt Configure Register 2 (Default Value: 0x0000_0000)

Offset: 0x02E8			Register Name: PH_EINT_CFG2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

7.7.5.70 0x02EC PH External Interrupt Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x02EC			Register Name: PH_EINT_CFG3
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.71 0x02F0 PH External Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x02F0			Register Name: PH_EINT_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:10	/	/	/
9	R/W	0x0	EINT9_CTL External INT9 Enable 0: Disable 1: Enable
8	R/W	0x0	EINT8_CTL External INT8 Enable 0: Disable 1: Enable
7	R/W	0x0	EINT7_CTL External INT7 Enable 0: Disable 1: Enable
6	R/W	0x0	EINT6_CTL External INT6 Enable 0: Disable 1: Enable
5	R/W	0x0	EINT5_CTL External INT5 Enable 0: Disable 1: Enable
4	R/W	0x0	EINT4_CTL External INT4 Enable 0: Disable 1: Enable
3	R/W	0x0	EINT3_CTL External INT3 Enable 0: Disable 1: Enable
2	R/W	0x0	EINT2_CTL External INT2 Enable 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x02F0			Register Name: PH_EINT_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
1	R/W	0x0	EINT1_CTL External INT1 Enable 0: Disable 1: Enable
0	R/W	0x0	EINT0_CTL External INT0 Enable 0: Disable 1: Enable

7.7.5.72 0x02F4 PH External Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x02F4			Register Name: PH_EINT_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:10	/	/	/
9	R/W	0x0	EINT9_STATUS External INT9 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
8	R/W	0x0	EINT8_STATUS External INT8 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
7	R/W	0x0	EINT7_STATUS External INT7 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
6	R/W	0x0	EINT6_STATUS External INT6 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x02F4			Register Name: PH_EINT_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
5	R/W	0x0	EINT5_STATUS External INT5 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
4	R/W	0x0	EINT4_STATUS External INT4 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
3	R/W	0x0	EINT3_STATUS External INT3 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
2	R/W	0x0	EINT2_STATUS External INT2 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
1	R/W	0x0	EINT1_STATUS External INT1 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
0	R/W	0x0	EINT0_STATUS External INT0 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.5.73 0x02F8 PH External Interrupt Debounce Register (Default Value: 0x0000_0000)

Offset: 0x02F8			Register Name: PH_EINT_DEB
Bit	**Read/Write**	**Default/Hex**	**Description**
31:7	/	/	/
6:4	R/W	0x0	DEB_CLK_PRE_SCALE Debounce Clock Pre_scale n The selected clock source is prescaled by 2^n.
3:1	/	/	/
0	R/W	0x0	PIO_INT_CLK_SELECT PIO Interrupt Clock Select 0: LOSC 32KHz 1: HOSC 24MHz

7.7.5.74 0x0340 PIO Group Withstand Voltage Mode Select Register (Default Value: 0x0000_0000)

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

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

Offset: 0x0340			Register Name: PIO_POW_MOD_SEL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:13	/	/	/
12	R/W	0x0	VCC_IO POWER MODE Select 0: 3.3V 1: 1.8V
11:7	/	/	/
6	R/W	0x0	PG_POWER MODE Select 0: 3.3V 1: 1.8V If PG_Port Power Source selects VCC_IO, this bit is invalid
5	R/W	0x0	PF_POWER MODE Select 0: 3.3V 1: 1.8V If PF_Port Power Source selects VCC_IO, this bit is invalid
4:3	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0340			Register Name: PIO_POW_MOD_SEL
Bit	Read/Write	Default/Hex	Description
2	R/W	0x0	PC_POWER MODE Select 0: 3.3V 1: 1.8V If PC_Port Power Source selects VCC_IO, this bit is invalid
1	R/W	0x0	PB_POWER MODE Select 0: 3.3V 1: 1.8V If PB_Port Power Source selects VCC_IO, this bit is invalid
0	/	/	/

7.7.5.75 0x0344 PIO Group Withstand Voltage Mode Select Control Register (Default Value: 0x0000_0000)

For 1.8 V and 3.3 V power, the withstand function is enabled by default, the corresponding bit in the 0x0344 register is set to 0. For 2.5 V power, the withstand function is disabled, the corresponding bit in the 0x0344 register is set to 1, and the corresponding withstand voltage in the 0x0340 register needs to be set to 3.3 V.

Offset: 0x0344			Register Name: PIO_POW_MS_CTL
Bit	Read/Write	Default/Hex	Description
31:13	/	/	/
12	R/W	0x0	VCC_IO Withstand Voltage Mode Select Control 0: Enable 1: Disable
11:7	/	/	/
6	R/W	0x0	VCC_PG Withstand Voltage Mode Select Control 0: Enable 1: Disable
5	R/W	0x0	VCC_PF Withstand Voltage Mode Select Control 0: Enable 1: Disable
4:3	/	/	/
2	R/W	0x0	VCC_PC Withstand Voltage Mode Select Control 0: Enable 1: Disable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.7.5.76 0x0348 PIO Group Power Value Register (Default Value: 0x0000_0000)

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

Offset: 0x0348			Register Name: PIO_VER
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R	0x0	VCC_IO Power Value
15:7	/	/	/
6	R	0x0	PG_Port Power Value If PG_Port Power Source select VCC_IO, this bit is invalid
5	R	0x0	PF_Port Power Value If PF_Port Power Source select VCC_IO, this bit is invalid
4:3	/	/	/
2	R	0x0	PC_Port Power Value If PC_Port Power Source select VCC_IO, this bit is invalid
1	R	0x0	PB_Port Power Value If PB_Port Power Source select VCC_IO, this bit is invalid
0	/	/	/

7.7.5.77 0x0350 PIO Group Power Voltage Select Control Register (Default Value: 0x0000_0001)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.6 PL/PM/PN Register List

Module Name	Base Address
R_GPIO	0x07022000

Register Name	Offset	Description
Pn_CFG0	n*0x24+0x00	Port n Configure Register 0 (n =0, 1, 2)
Pn_CFG1	n*0x24+0x04	Port n Configure Register 1 (n =0, 1, 2)
Pn_CFG2	n*0x24+0x08	Port n Configure Register 2 (n =0, 1, 2)
Pn_CFG3	n*0x24+0x0C	Port n Configure Register 3 (n =0, 1, 2)
Pn_DAT	n*0x24+0x10	Port n Data Register (n =0, 1, 2)
Pn_DRV0	n*0x24+0x14	Port n Multi_Driving Register 0 (n =0, 1, 2)
Pn_DRV1	n*0x24+0x18	Port n Multi_Driving Register 1 (n =0, 1, 2)
Pn_PUL0	n*0x24+0x1C	Port n Pull Register 0 (n =0, 1, 2)
Pn_PUL1	n*0x24+0x20	Port n Pull Register 1 (n =0, 1, 2)
Pn_INT_CFG0	0x200+n*0x20+0x00	PIO Interrupt Configure Register 0 (n =0, 1, 2)
Pn _INT_CFG1	0x200+n*0x20+0x04	PIO Interrupt Configure Register 1 (n =0, 1, 2)
Pn _INT_CFG2	0x200+n*0x20+0x08	PIO Interrupt Configure Register 2 (n =0, 1, 2)
Pn _INT_CFG3	0x200+n*0x20+0x0C	PIO Interrupt Configure Register 3 (n =0, 1, 2)
Pn _INT_CTL	0x200+n*0x20+0x10	PIO Interrupt Control Register (n =0, 1, 2)
Pn _INT_STA	0x200+n*0x20+0x14	PIO Interrupt Status Register (n =0, 1, 2)
Pn _INT_DEB	0x200+n*0x20+0x18	PIO Interrupt Debounce Register (n =0, 1, 2)
PIO_POW_MOD_SEL	0x0340	PIO Group Withstand Voltage Mode Select Register
PIO_POW_MS_CTL	0x0344	PIO Group Withstand Voltage Mode Select Control Register
PIO_POW_VAL	0x0348	PIO Group Power Value Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.7 PL/PM/PN Register Description

7.7.7.1 0x0000 PL Configure Register 0 (Default Value: 0x7777_7777)

Offset: 0x0000			Register Name: PL_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31	/	/	/
30:28	R/W	0x7	PL7_SELECT 000: Input 001: Output 010: S-IR-RX 011: Reserved 100: X32KFOUT 101: S-PWM5 110: S-PL-EINT7 111: IO Disable
27	/	/	/
26:24	R/W	0x7	PL6_SELECT 000: Input 001: Output 010: S-TWI0-SCK 011: Reserved 100: Reserved 101: S-PWM4 110: S-PL-EINT6 111: IO Disable
23	/	/	/
22:20	R/W	0x7	PL5_SELECT 000: Input 001: Output 010: S-TWI0-SDA 011: Reserved 100: Reserved 101: S-PWM3 110: S-PL-EINT5 111: IO Disable
19	/	/	/
18:16	R/W	0x7	PL4_SELECT 000: Input 001: Output 010: S-I2S0-MCLK 011: S-IR-RX 100: S-DMIC-CLK 101: S-TWI0-SCK 110: S-PL-EINT4 111: IO Disable
15	/	/	/
14:12	R/W	0x7	PL3_SELECT 000: Input 001: Output 010: S-I2S0-DOUT1 011: S-I2S0-DIN0 100: S-DMIC-DATA0 101: S-TWI0-SDA 110: S-PL-EINT3 111: IO Disable
11	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: PL_CFG0	
Bit	**Read/Write**	**Default/Hex**	**Description**	
10:8	R/W	0x7	PL2_SELECT 000: Input 010: S-I2S0-DOUT0 100: S-DMIC-DATA1 110: S-PL-EINT2	 001: Output 011: S-I2S0-DIN1 101: S-PWM2 111: IO Disable
7	/	/	/	
6:4	R/W	0x7	PL1_SELECT 000: Input 010: S-I2S0-BCLK 100: S-DMIC-DATA2 110: S-PL-EINT1	 001: Output 011: Reserved 101: S-PWM1 111: IO Disable
3	/	/	/	
2:0	R/W	0x7	PL0_SELECT 000: Input 010: S-I2S0-LRCK 100: S-DMIC-DATA3 110: S-PL-EINT0	 001: Output 011: Reserved 101: S-PWM0 111: IO Disable

7.7.7.2 0x0004 PL Configure Register 1 (Default Value: 0x0000_0777)

Offset: 0x0004			Register Name: PL_CFG1	
Bit	**Read/Write**	**Default/Hex**	**Description**	
31:11	/	/	/	
10:8	R/W	0x7	PL10_SELECT 000: Input 010: Reserved 100: Reserved 110: S-PL-EINT10	 001: Output 011: Reserved 101: Reserved 111: IO Disable
7	/	/	/	
6:4	R/W	0x7	PL9_SELECT 000: Input 010: S-UART0-RX 100: X32KFOUT 110: S-PL-EINT9	 001: Output 011: S-TWI0-SCK 101: Reserved 111: IO Disable
3	/	/	/	

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0004			Register Name: PL_CFG1
Bit	Read/Write	Default/Hex	Description
2:0	R/W	0x7	PL8_SELECT 000: Input 001: Output 010: S-UART0-TX 011: S-TWI0-SDA 100: S-IR-RX 101: Reserved 110: S-PL-EINT8 111: IO Disable

7.7.7.3 0x0008 PL Configure Register 2 (Default Value: 0x0000_0000)

Offset: 0x0008			Register Name: PL_CFG2
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.7.4 0x000C PL Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x000C			Register Name: PL_CFG3
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.7.5 0x0010 PL Data Register (Default Value: 0x0000_0000)

Offset: 0x0010			Register Name: PL_DAT
Bit	Read/Write	Default/Hex	Description
31:11	/	/	/
10:0	R/W	0x0	PL_DAT If the port is configured as the input function, the corresponding bit is the pin state. If the port is configured as the output function, the pin state is the same as the corresponding bit. The read bit value is the value set up by software. If the port is configured as a functional pin, the undefined value will be read.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.7.6 0x0014 PL Multi_Driving Register 0 (Default Value: 0x0015_5555)

Offset: 0x0014			Register Name: PL_DRV0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:22	/	/	/
21:20	R/W	0x1	PL10_DRV PL10 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
19:18	R/W	0x1	PL9_DRV PL9 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
17:16	R/W	0x1	PL8_DRV PL8 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
15:14	R/W	0x1	PL7_DRV PL7 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
13:12	R/W	0x1	PL6_DRV PL6 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
11:10	R/W	0x1	PL5_DRV PL5 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
9:8	R/W	0x1	PL4_DRV PL4 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
7:6	R/W	0x1	PL3_DRV PL3 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0014			Register Name: PL_DRV0
Bit	Read/Write	Default/Hex	Description
5:4	R/W	0x1	PL2_DRV PL2 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
3:2	R/W	0x1	PL1_DRV PL1 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
1:0	R/W	0x1	PL0_DRV PL0 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

7.7.7.7 0x0018 PL Multi_Driving Register 1 (Default Value: 0x0000_0000)

Offset: 0x0018			Register Name: PL_DRV1
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.7.8 0x001C PL Pull Register 0 (Default Value: 0x0000_0000)

Offset: 0x001C			Register Name: PL_PULL0
Bit	Read/Write	Default/Hex	Description
31:22	/	/	/
21:20	R/W	0x0	PL10_PULL PL10 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
19:18	R/W	0x0	PL9_PULL PL9 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x001C			Register Name: PL_PULL0
Bit	**Read/Write**	**Default/Hex**	**Description**
17:16	R/W	0x0	PL8_PULL PL8 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
15:14	R/W	0x0	PL7_PULL PL7 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
13:12	R/W	0x0	PL6_PULL PL6 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
11:10	R/W	0x0	PL5_PULL PL5 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
9:8	R/W	0x0	PL4_PULL PL4 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
7:6	R/W	0x0	PL3_PULL PL3 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
5:4	R/W	0x0	PL2_PULL PL2 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
3:2	R/W	0x0	PL1_PULL PL1 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
1:0	R/W	0x0	PL0_PULL PL0 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.7.9 0x0020 PL Pull Register 1 (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: PL_PULL1
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.7.10 0x0024 PM Configure Register 0 (Default Value: 0x7777_7777)

Offset: 0x0024			Register Name: PM_CFG0
Bit	Read/Write	Default/Hex	Description
31	/	/	/
30:28	R/W	0x7	PM7_SELECT 000: Input 001: Output 010: S-IR-RX 011: X32KFOUT 100: Reserved 101: Reserved 110: PM-EINT7 111: IO Disable
27	/	/	/
26:24	R/W	0x7	PM6_SELECT 000: Input 001: Output 010: NMI 011: S-IR-RX 100: X32KFOUT 101: Reserved 110: PM-EINT6 111: IO Disable
23	/	/	/
22:20	R/W	0x7	PM5_SELECT 000: Input 001: Output 010: X32KFOUT 011: S-JTAG-DI 100: S-TWI0-SCK 101: Reserved 110: PM-EINT5 111: IO Disable
19	/	/	/
18:16	R/W	0x7	PM4_SELECT 000: Input 001: Output 010: S-TWI0-SCK 011: Reserved 100: Reserved 101: Reserved 110: PM-EINT4 111: IO Disable
15	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0024			Register Name: PM_CFG0	
Bit	**Read/Write**	**Default/Hex**	**Description**	
14:12	R/W	0x7	PM3_SELECT 000: Input 010: S-TWI0-SDA 100: Reserved 110: PM-EINT3	 001: Output 011: S-IR-RX 101: Reserved 111: IO Disable
11	/	/	/	
10:8	R/W	0x7	PM2_SELECT 000: Input 010: Reserved 100: S-TWI0-SDA 110: PM-EINT2	 001: Output 011: S-JTAG-DO 101: S-IR-RX 111: IO Disable
7	/	/	/	
6:4	R/W	0x7	PM1_SELECT 000: Input 010: S-UART0-RX 100: Reserved 110: PM-EINT1	 001: Output 011: S-JTAG-CK 101: Reserved 111: IO Disable
3	/	/	/	
2:0	R/W	0x7	PM0_SELECT 000: Input 010: S-UART0-TX 100: Reserved 110: PM-EINT0	 001: Output 011: S-JTAG-MS 101: Reserved 111: IO Disable

7.7.7.11 0x0028 PM Configure Register 1 (Default Value: 0x0000_0000)

Offset: 0x0028			Register Name: PM_CFG1	
Bit	**Read/Write**	**Default/Hex**	**Description**	
31:3	/	/	/	
2:0	R/W	0x7	PM8_SELECT 000: Input 010: Reserved 100: Reserved 110: PM-EINT8	 001: Output 011: Reserved 101: Reserved 111: IO Disable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.7.12 0x002C PM Configure Register 2 (Default Value: 0x0000_0000)

Offset: 0x002C			Register Name: PM_CFG2
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.7.13 0x0030 PM Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: PM_CFG3
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.7.14 0x0034 PM Data Register (Default Value: 0x0000_0000)

Offset: 0x0034			Register Name: PM_DAT
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8:0	R/W	0x0	PM_DAT If the port is configured as the input function, the corresponding bit is the pin state. If the port is configured as the output function, the pin state is the same as the corresponding bit. The read bit value is the value set up by software. If the port is configured as a functional pin, the undefined value will be read.

7.7.7.15 0x0038 PM Multi_Driving Register 0 (Default Value: 0x0001_5555)

Offset: 0x0038			Register Name: PM_DRV0
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17:16	R/W	0x1	PM8_DRV PM8 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0038			Register Name: PM_DRV0
Bit	**Read/Write**	**Default/Hex**	**Description**
15:14	R/W	0x1	PM7_DRV PM7 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
13:12	R/W	0x1	PM6_DRV PM6 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3 **Note: The configuration is invalid.**
11:10	R/W	0x1	PM5_DRV PM5 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
9:8	R/W	0x1	PM4_DRV PM4 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
7:6	R/W	0x1	PM3_DRV PM3 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
5:4	R/W	0x1	PM2_DRV PM2 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
3:2	R/W	0x1	PM1_DRV PM1 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
1:0	R/W	0x1	PM0_DRV PM0 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.7.16 0x003C PM Multi_Driving Register 1 (Default Value: 0x0000_0000)

Offset: 0x003C			Register Name: PM_DRV1
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.7.17 0x0040 PM Pull Register 0 (Default Value: 0x0000_0000)

Offset: 0x0040			Register Name: PM_PULL0
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17:16	R/W	0x0	PM8_PULL PM8 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
15:14	R/W	0x0	PM7_PULL PM7 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
13:12	R/W	0x0	PM6_PULL PM6 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved **Note: The configuration is invalid.**
11:10	R/W	0x0	PM5_PULL PM5 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
9:8	R/W	0x0	PM4_PULL PM4 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
7:6	R/W	0x0	PM3_PULL PM3 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0040			Register Name: PM_PULL0
Bit	Read/Write	Default/Hex	Description
5:4	R/W	0x0	PM2_PULL PM2 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
3:2	R/W	0x0	PM1_PULL PM1 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
1:0	R/W	0x0	PM0_PULL PM0 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

7.7.7.18 0x0044 PM Pull Register 1 (Default Value: 0x0000_0000)

Offset: 0x0044			Register Name: PM_PULL1
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.7.19 0x0048 PN Configure Register 0 (Default Value: 0x7777_7777)

Offset: 0x0048			Register Name: PN_CFG0
Bit	Read/Write	Default/Hex	Description
31	/	/	/
30:28	R/W	0x7	PN7_SELECT 000:Input 001:Output 010:Reserved 011:RGMII-RXCK 100:Reserved 101:Reserved 110:PN-EINT7 111:IO Disable
27	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0048			Register Name: PN_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
26:24	R/W	0x7	PN6_SELECT 000:Input 001:Output 010:Reserved 011:RGMII-RXD0/RMII-RXD0 100:Reserved 101:Reserved 110:PN-EINT6 111:IO Disable
23	/	/	/
22:20	R/W	0x7	PN5_SELECT 000:Input 001:Output 010:Reserved 011:RGMII-RXD2 100:Reserved 101:Reserved 110:PN-EINT5 111:IO Disable
19	/	/	/
18:16	R/W	0x7	PN4_SELECT 000:Input 001:Output 010:Reserved 011:RGMII-TXCK/RMII-TXCK 100:Reserved 101:Reserved 110:PN-EINT4 111:IO Disable
15	/	/	/
14:12	R/W	0x7	PN3_SELECT 000:Input 001:Output 010:Reserved 011:RGMII-RXD3 100:Reserved 101:Reserved 110:PN-EINT3 111:IO Disable
11	/	/	/
10:8	R/W	0x7	PN2_SELECT 000:Input 001:Output 010:Reserved 011:MDIO 100:Reserved 101:Reserved 110:PN-EINT2 111:IO Disable
7	/	/	/
6:4	R/W	0x7	PN1_SELECT 000:Input 001:Output 010:Reserved 011:MDC 100:Reserved 101:Reserved 110:PN-EINT1 111:IO Disable
3	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0048			Register Name: PN_CFG0
Bit	Read/Write	Default/Hex	Description
2:0	R/W	0x7	PN0_SELECT 000:Input 001:Output 010:Reserved 011:Reserved 100:Reserved 101:Reserved 110:PN-EINT0 111:IO Disable

7.7.7.20 0x004C PN Configure Register 1 (Default Value: 0x7777_7777)

Offset: 0x004C			Register Name: PN_CFG1
Bit	Read/Write	Default/Hex	Description
31	/	/	/
30:28	R/W	0x7	PN15_SELECT 000:Input 001:Output 010:Reserved 011:RGMII-TXD0/RMII-TXD0 100:Reserved 101:Reserved 110:PN-EINT15 111:IO Disable
27	/	/	/
26:24	R/W	0x7	PN14_SELECT 000:Input 001:Output 010:Reserved 011:RGMII-TXD1/RMII-TXD1 100:Reserved 101:Reserved 110:PN-EINT14 111:IO Disable
23	/	/	/
22:20	R/W	0x7	PN13_SELECT 000:Input 001:Output 010:Reserved 011:RGMII-TXD2 100:Reserved 101:Reserved 110:PN-EINT13 111:IO Disable
19	/	/	/
18:16	R/W	0x7	PN12_SELECT 000:Input 001:Output 010:Reserved 011:RGMII-RXCTL/RMII-CRS-DV 100:Reserved 101:Reserved 110:PN-EINT12 111:IO Disable
15	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x004C			Register Name: PN_CFG1	
Bit	**Read/Write**	**Default/Hex**	**Description**	
14:12	R/W	0x7	PN11_SELECT 000:Input 010:Reserved 100:Reserved 110:PN-EINT11	 001:Output 011:RGMII-RXD1/RMII-RXD1 101:Reserved 111:IO Disable
11	/	/	/	
10:8	R/W	0x7	PN10_SELECT 000:Input 010:Reserved 100:Reserved 110:PN-EINT10	 001:Output 011:RGMII-RXD3 101:Reserved 111:IO Disable
7	/	/	/	
6:4	R/W	0x7	PN9_SELECT 000:Input 010:Reserved 100:Reserved 110:PN-EINT9	 001:Output 011:RGMII-TXCTL/RMII-TXEN 101:Reserved 111:IO Disable
3	/	/	/	
2:0	R/W	0x7	PN8_SELECT 000:Input 010:Reserved 100:Reserved 110:PN-EINT8	 001:Output 011:RMII-RXER 101:Reserved 111:IO Disable

7.7.7.21 0x0050 PN Configure Register 2 (Default Value: 0x7777_7777)

Offset: 0x0050			Register Name: PN_CFG2	
Bit	**Read/Write**	**Default/Hex**	**Description**	
31	/	/	/	
30:28	R/W	0x7	PN23_SELECT 000:Input 010:Reserved 100:Reserved 110:PN-EINT23	 001:Output 011:Reserved 101:Reserved 111:IO Disable
27	/	/	/	

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0050			Register Name: PN_CFG2
Bit	**Read/Write**	**Default/Hex**	**Description**
26:24	R/W	0x7	PN22_SELECT 000:Input 001:Output 010:Reserved 011:Reserved 100:Reserved 101:Reserved 110:PN-EINT22 111:IO Disable
23	/	/	/
22:20	R/W	0x7	PN21_SELECT 000:Input 001:Output 010:Reserved 011: Reserved 100:Reserved 101: Reserved 110:PN-EINT21 111:IO Disable
19	/	/	/
18:16	R/W	0x7	PN20_SELECT 000:Input 001:Output 010:Reserved 011:Reserved 100:Reserved 101:Reserved 110:PN-EINT20 111:IO Disable
15	/	/	/
14:12	R/W	0x7	PN19_SELECT 000:Input 001:Output 010:Reserved 011: Reserved 100:Reserved 101: Reserved 110:PN-EINT19 111:IO Disable
11	/	/	/
10:8	R/W	0x7	PN18_SELECT 000:Input 001:Output 010:Reserved 011: Reserved 100:Reserved 101: Reserved 110:PN-EINT18 111:IO Disable
7	/	/	/
6:4	R/W	0x7	PN17_SELECT 000:Input 001:Output 010:Reserved 011:RGMII-CLKIN 100:Reserved 101:Reserved 110:PN-EINT17 111:IO Disable
3	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0050			Register Name: PN_CFG2
Bit	Read/Write	Default/Hex	Description
2:0	R/W	0x7	PN16_SELECT 000:Input 001:Output 010:Reserved 011:EPHY-25M 100:Reserved 101:Reserved 110:PN-EINT16 111:IO Disable

7.7.7.22 0x0054 PN Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x0054			Register Name: PN_CFG3
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.7.23 0x0058 PN Data Register (Default Value: 0x0000_0000)

Offset: 0x0058			Register Name: PN_DAT
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:0	R/W	0x0	PN_DAT If the port is configured as the input function, the corresponding bit is the pin state. If the port is configured as the output function, the pin state is the same as the corresponding bit. The read bit value is the value set up by software. If the port is configured as a functional pin, the undefined value will be read.

7.7.7.24 0x005C PN Multi_Driving Register 0 (Default Value: 0x5555_5555)

Offset: 0x005C			Register Name: PN_DRV0
Bit	Read/Write	Default/Hex	Description
31:30	R/W	0x1	PN15_DRV PN15 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x005C			Register Name: PN_DRV0
Bit	**Read/Write**	**Default/Hex**	**Description**
29:28	R/W	0x1	PN14_DRV PN14 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
27:26	R/W	0x1	PN13_DRV PN13 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
25:24	R/W	0x1	PN12_DRV PN12 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
23:22	R/W	0x1	PN11_DRV PN11 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
21:20	R/W	0x1	PN10_DRV PN10 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
19:18	R/W	0x1	PN9_DRV PN9 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
17:16	R/W	0x1	PN8_DRV PN8 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
15:14	R/W	0x1	PN7_DRV PN7 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
13:12	R/W	0x1	PN6_DRV PN6 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x005C			Register Name: PN_DRV0
Bit	**Read/Write**	**Default/Hex**	**Description**
11:10	R/W	0x1	PN5_DRV PN5 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
9:8	R/W	0x1	PN4_DRV PN4 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
7:6	R/W	0x1	PN3_DRV PN3 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
5:4	R/W	0x1	PN2_DRV PN2 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
3:2	R/W	0x1	PN1_DRV PN1 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
1:0	R/W	0x1	PN0_DRV PN0 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

7.7.7.25 0x0060 PN Multi_Driving Register 1 (Default Value: 0x0000_5555)

Offset: 0x0060			Register Name: PN_DRV1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:14	R/W	0x1	PN23_DRV PN23 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0060			Register Name: PN_DRV1
Bit	**Read/Write**	**Default/Hex**	**Description**
13:12	R/W	0x1	PN22_DRV PN22 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
11:10	R/W	0x1	PN21_DRV PN21 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
9:8	R/W	0x1	PN20_DRV PN20 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
7:6	R/W	0x1	PN19_DRV PN19 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
5:4	R/W	0x1	PN18_DRV PN18 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
3:2	R/W	0x1	PN17_DRV PN17 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
1:0	R/W	0x1	PN16_DRV PN16 Multi_Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.7.26 0x0064 PN Pull Register 0 (Default Value: 0x0000_0000)

Offset: 0x0064			Register Name: PN_PULL0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	R/W	0x0	PN15_PULL PN15 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
29:28	R/W	0x0	PN14_PULL PN14 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
27:26	R/W	0x0	PN13_PULL PN13 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
25:24	R/W	0x0	PN12_PULL PN12 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
23:22	R/W	0x0	PN11_PULL PN11 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
21:20	R/W	0x0	PN10_PULL PN10 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
19:18	R/W	0x0	PN9_PULL PN9 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
17:16	R/W	0x0	PN8_PULL PN8 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0064			Register Name: PN_PULL0
Bit	**Read/Write**	**Default/Hex**	**Description**
15:14	R/W	0x0	PN7_PULL PN7 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
13:12	R/W	0x0	PN6_PULL PN6 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
11:10	R/W	0x0	PN5_PULL PN5 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
9:8	R/W	0x0	PN4_PULL PN4 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
7:6	R/W	0x0	PN3_PULL PN3 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
5:4	R/W	0x0	PN2_PULL PN2 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
3:2	R/W	0x0	PN1_PULL PN1 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
1:0	R/W	0x0	PN0_PULL PN0 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.7.27 0x0068 PN Pull Register 1 (Default Value: 0x0000_0000)

Offset: 0x0068			Register Name: PN_PULL1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:14	R/W	0x0	PN23_PULL PN23 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
13:12	R/W	0x0	PN22_PULL PN22 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
11:10	R/W	0x0	PN21_PULL PN21 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
9:8	R/W	0x0	PN20_PULL PN20 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
7:6	R/W	0x0	PN19_PULL PN19 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
5:4	R/W	0x0	PN18_PULL PN18 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
3:2	R/W	0x0	PN17_PULL PN17 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved
1:0	R/W	0x0	PN16_PULL PN16 Pull_up/down Select 00: Pull_up/down disable 01: Pull_up 10: Pull_down 11: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.7.28 0x0200 PL External Interrupt Configure Register 0 (Default Value: 0x0000_0000)

Offset: 0x0200			Register Name:PL_EINT_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x0	EINT7_CFG External INT7 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
27:24	R/W	0x0	EINT6_CFG External INT6 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
23:20	R/W	0x0	EINT5_CFG External INT5 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
19:16	R/W	0x0	EINT4_CFG External INT4 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0200			Register Name:PL_EINT_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
15:12	R/W	0x0	EINT3_CFG External INT3 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
11:8	R/W	0x0	EINT2_CFG External INT2 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
7:4	R/W	0x0	EINT1_CFG External INT1 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
3:0	R/W	0x0	EINT0_CFG External INT0 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.7.29 0x0204 PL External Interrupt Configure Register 1 (Default Value: 0x0000_0000)

Offset: 0x0204			Register Name: PL_EINT_CFG1
Bit	**Read/Write**	**Default**	**Description**
31:12	/	/	/
11:8	R/W	0x0	EINT10_CFG External INT10 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
7:4	R/W	0x0	EINT9_CFG External INT9 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
3:0	R/W	0x0	EINT8_CFG External INT8 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

7.7.7.30 0x0208 PL External Interrupt Configure Register 2 (Default Value: 0x0000_0000)

Offset: 0x0208			Register Name: PL_EINT_CFG2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.7.31 0x020C PL External Interrupt Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x020C			Register Name: PL_EINT_CFG3
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.7.32 0x0210 PL External Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0210			Register Name: PL_EINT_CTL
Bit	Read/Write	Default/Hex	Description
31:11	/	/	/
10	R/W	0x0	EINT10_CTL External INT10 Enable 0: Disabl 1: Enable
9	R/W	0x0	EINT9_CTL External INT9 Enable 0: Disabl 1: Enable
8	R/W	0x0	EINT8_CTL External INT8 Enable 0: Disable 1: Enable
7	R/W	0x0	EINT7_CTL External INT7 Enable 0: Disable 1: Enable
6	R/W	0x0	EINT6_CTL External INT6 Enable 0: Disable 1: Enable
5	R/W	0x0	EINT5_CTL External INT5 Enable 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0210			Register Name: PL_EINT_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
4	R/W	0x0	EINT4_CTL External INT4 Enable 0: Disable 1: Enable
3	R/W	0x0	EINT3_CTL External INT3 Enable 0: Disable 1: Enable
2	R/W	0x0	EINT2_CTL External INT2 Enable 0: Disable 1: Enable
1	R/W	0x0	EINT1_CTL External INT1 Enable 0: Disable 1: Enable
0	R/W	0x0	EINT0_CTL External INT0 Enable 0: Disable 1: Enable

7.7.7.33 0x0214 PL External Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0214			Register Name: PL_EINT_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10	R/W	0x0	EINT10_STATUS External INT10 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0214			Register Name: PL_EINT_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
9	R/W	0x0	EINT9_STATUS External INT9 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
8	R/W	0x0	EINT8_STATUS External INT8 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
7	R/W	0x0	EINT7_STATUS External INT7 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
6	R/W	0x0	EINT6_STATUS External INT6 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
5	R/W	0x0	EINT5_STATUS External INT5 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
4	R/W	0x0	EINT4_STATUS External INT4 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
3	R/W	0x0	EINT3_STATUS External INT3 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0214			Register Name: PL_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
2	R/W	0x0	EINT2_STATUS External INT2 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
1	R/W	0x0	EINT1_STATUS External INT1 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
0	R/W	0x0	EINT0_STATUS External INT0 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

7.7.7.34 0x0218 PL External Interrupt Debounce Register (Default Value: 0x0000_0000)

Offset: 0x0218			Register Name: PL_EINT_DEB
Bit	Read/Write	Default	Description
31:7	/	/	/
6:4	R/W	0x0	DEB_CLK_PRE_SCALE Debounce Clock Pre_scale n The selected clock source is prescaled by 2^n.
3:1	/	/	/
0	R/W	0x0	PIO_INT_CLK_SELECT PIO Interrupt Clock Select 0: LOSC 32KHz 1: HOSC 24MHz

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.7.35 0x0220 PM External Interrupt Configure Register 0 (Default Value: 0x0000_0000)

Offset: 0x0220			Register Name:PM_EINT_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x0	EINT7_CFG External INT7 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
27:24	R/W	0x0	EINT6_CFG External INT6 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
23:20	R/W	0x0	EINT5_CFG External INT5 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
19:16	R/W	0x0	EINT4_CFG External INT4 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0220			Register Name:PM_EINT_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
15:12	R/W	0x0	EINT3_CFG External INT3 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
11:8	R/W	0x0	EINT2_CFG External INT2 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
7:4	R/W	0x0	EINT1_CFG External INT1 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
3:0	R/W	0x0	EINT0_CFG External INT0 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.7.36 0x0224 PM External Interrupt Configure Register 1 (Default Value: 0x0000_0000)

Offset: 0x0224			Register Name: PM_EINT_CFG1
Bit	**Read/Write**	**Default**	**Description**
31:4	/	/	/
3:0	R/W	0x0	EINT8_CFG External INT8 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

7.7.7.37 0x0228 PM External Interrupt Configure Register 2 (Default Value: 0x0000_0000)

Offset: 0x0228			Register Name: PM_EINT_CFG2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

7.7.7.38 0x022C PM External Interrupt Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x022C			Register Name: PM_EINT_CFG3
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

7.7.7.39 0x0230 PM External Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0230			Register Name: PM_EINT_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:9	/	/	/
8	R/W	0x0	EINT8_CTL External INT8 Enable 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0230			Register Name: PM_EINT_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
7	R/W	0x0	EINT7_CTL External INT7 Enable 0: Disable 1: Enable
6	R/W	0x0	EINT6_CTL External INT6 Enable 0: Disable 1: Enable
5	R/W	0x0	EINT5_CTL External INT5 Enable 0: Disable 1: Enable
4	R/W	0x0	EINT4_CTL External INT4 Enable 0: Disable 1: Enable
3	R/W	0x0	EINT3_CTL External INT3 Enable 0: Disable 1: Enable
2	R/W	0x0	EINT2_CTL External INT2 Enable 0: Disable 1: Enable
1	R/W	0x0	EINT1_CTL External INT1 Enable 0: Disable 1: Enable
0	R/W	0x0	EINT0_CTL External INT0 Enable 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.7.40 0x0234 PM External Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0234			Register Name: PM_EINT_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:9	/	/	/
8	R/W	0x0	EINT8_STATUS External INT8 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
7	R/W	0x0	EINT7_STATUS External INT7 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
6	R/W	0x0	EINT6_STATUS External INT6 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
5	R/W	0x0	EINT5_STATUS External INT5 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
4	R/W	0x0	EINT4_STATUS External INT4 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
3	R/W	0x0	EINT3_STATUS External INT3 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0234			Register Name: PM_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
2	R/W	0x0	EINT2_STATUS External INT2 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
1	R/W	0x0	EINT1_STATUS External INT1 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
0	R/W	0x0	EINT0_STATUS External INT0 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

7.7.7.41 0x0238 PM External Interrupt Debounce Register (Default Value: 0x0000_0000)

Offset: 0x0238			Register Name: PM_EINT_DEB
Bit	Read/Write	Default	Description
31:7	/	/	/
6:4	R/W	0x0	DEB_CLK_PRE_SCALE Debounce Clock Pre_scale n The selected clock source is prescaled by 2^n.
3:1	/	/	/
0	R/W	0x0	PIO_INT_CLK_SELECT PIO Interrupt Clock Select 0: LOSC 32KHz 1: HOSC 24MHz

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.7.42 0x0240 PN External Interrupt Configure Register 0 (Default Value: 0x0000_0000)

Offset: 0x0240			Register Name:PN_EINT_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x0	EINT7_CFG External INT7 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
27:24	R/W	0x0	EINT6_CFG External INT6 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
23:20	R/W	0x0	EINT5_CFG External INT5 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
19:16	R/W	0x0	EINT4_CFG External INT4 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0240			Register Name:PN_EINT_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
15:12	R/W	0x0	EINT3_CFG External INT3 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
11:8	R/W	0x0	EINT2_CFG External INT2 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
7:4	R/W	0x0	EINT1_CFG External INT1 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
3:0	R/W	0x0	EINT0_CFG External INT0 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.7.43 0x0244 PN External Interrupt Configure Register 1 (Default Value: 0x0000_0000)

Offset: 0x0244			Register Name: PN_EINT_CFG1
Bit	**Read/Write**	**Default**	**Description**
31:28	R/W	0x0	EINT15_CFG External INT15 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
27:24	R/W	0x0	EINT14_CFG External INT14 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
23:20	R/W	0x0	EINT13_CFG External INT13 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
19:16	R/W	0x0	EINT12_CFG External INT12 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0244			Register Name: PN_EINT_CFG1
Bit	**Read/Write**	**Default**	**Description**
15:12	R/W	0x0	EINT11_CFG External INT11 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
11:8	R/W	0x0	EINT10_CFG External INT10 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
7:4	R/W	0x0	EINT9_CFG External INT9 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
3:0	R/W	0x0	EINT8_CFG External INT8 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.7.44 0x0248 PN External Interrupt Configure Register 2 (Default Value: 0x0000_0000)

Offset: 0x0248			Register Name: PN_EINT_CFG2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x0	EINT23_CFG External INT23 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
27:24	R/W	0x0	EINT22_CFG External INT22 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
23:20	R/W	0x0	EINT21_CFG External INT21 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
19:16	R/W	0x0	EINT20_CFG External INT20 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0248			Register Name: PN_EINT_CFG2
Bit	**Read/Write**	**Default/Hex**	**Description**
15:12	R/W	0x0	EINT19_CFG External INT19 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
11:8	R/W	0x0	EINT18_CFG External INT18 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
7:4	R/W	0x0	EINT17_CFG External INT17 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved
3:0	R/W	0x0	EINT16_CFG External INT16 Mode 0x0: Positive Edge 0x1: Negative Edge 0x2: High Level 0x3: Low Level 0x4: Double Edge (Positive/Negative) Others: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.7.45 0x024C PN External Interrupt Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x024C			Register Name: PN_EINT_CFG3
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

7.7.7.46 0x0250 PN External Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0250			Register Name: PN_EINT_CTL
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23	R/W	0x0	EINT23_CTL External INT23 Enable 0: Disable 1: Enable
22	R/W	0x0	EINT22_CTL External INT22 Enable 0: Disable 1: Enable
21	R/W	0x0	EINT21_CTL External INT21 Enable 0: Disable 1: Enable
20	R/W	0x0	EINT20_CTL External INT20 Enable 0: Disable 1: Enable
19	R/W	0x0	EINT19_CTL External INT19 Enable 0: Disable 1: Enable
18	R/W	0x0	EINT18_CTL External INT18 Enable 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0250			Register Name: PN_EINT_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
17	R/W	0x0	EINT17_CTL External INT17 Enable 0: Disable 1: Enable
16	R/W	0x0	EINT16_CTL External INT16 Enable 0: Disable 1: Enable
15	R/W	0x0	EINT15_CTL External INT15 Enable 0: Disable 1: Enable
14	R/W	0x0	EINT14_CTL External INT14 Enable 0: Disable 1: Enable
13	R/W	0x0	EINT13_CTL External INT13 Enable 0: Disable 1: Enable
12	R/W	0x0	EINT12_CTL External INT12 Enable 0: Disable 1: Enable
11	R/W	0x0	EINT11_CTL External INT11 Enable 0: Disable 1: Enable
10	R/W	0x0	EINT10_CTL External INT10 Enable 0: Disable 1: Enable
9	R/W	0x0	EINT9_CTL External INT9 Enable 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0250			Register Name: PN_EINT_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
8	R/W	0x0	EINT8_CTL External INT8 Enable 0: Disable 1: Enable
7	R/W	0x0	EINT7_CTL External INT7 Enable 0: Disable 1: Enable
6	R/W	0x0	EINT6_CTL External INT6 Enable 0: Disable 1: Enable
5	R/W	0x0	EINT5_CTL External INT5 Enable 0: Disable 1: Enable
4	R/W	0x0	EINT4_CTL External INT4 Enable 0: Disable 1: Enable
3	R/W	0x0	EINT3_CTL External INT3 Enable 0: Disable 1: Enable
2	R/W	0x0	EINT2_CTL External INT2 Enable 0: Disable 1: Enable
1	R/W	0x0	EINT1_CTL External INT1 Enable 0: Disable 1: Enable
0	R/W	0x0	EINT0_CTL External INT0 Enable 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.7.47 0x0254 PN External Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0254			Register Name: PN_EINT_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	/	/	/
23	R/W	0x0	EINT23_STATUS External INT23 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
22	R/W	0x0	EINT22_STATUS External INT22 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
21	R/W	0x0	EINT21_STATUS External INT21 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
20	R/W	0x0	EINT20_STATUS External INT20 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
19	R/W	0x0	EINT19_STATUS External INT19 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
18	R/W	0x0	EINT18_STATUS External INT18 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0254			Register Name: PN_EINT_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
17	R/W	0x0	EINT17_STATUS External INT17 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
16	R/W	0x0	EINT16_STATUS External INT16 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
15	R/W	0x0	EINT15_STATUS External INT15 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
14	R/W	0x0	EINT14_STATUS External INT14 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
13	R/W	0x0	EINT13_STATUS External INT13 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
12	R/W	0x0	EINT12_STATUS External INT12 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
11	R/W	0x0	EINT11_STATUS External INT11 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0254			Register Name: PN_EINT_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
10	R/W	0x0	EINT10_STATUS External INT10 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
9	R/W	0x0	EINT9_STATUS External INT9 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
8	R/W	0x0	EINT8_STATUS External INT8 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
7	R/W	0x0	EINT7_STATUS External INT7 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
6	R/W	0x0	EINT6_STATUS External INT6 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
5	R/W	0x0	EINT5_STATUS External INT5 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
4	R/W	0x0	EINT4_STATUS External INT4 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0254			Register Name: PN_EINT_STATUS
Bit	Read/Write	Default/Hex	Description
3	R/W	0x0	EINT3_STATUS External INT3 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
2	R/W	0x0	EINT2_STATUS External INT2 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
1	R/W	0x0	EINT1_STATUS External INT1 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
0	R/W	0x0	EINT0_STATUS External INT0 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

7.7.7.48 0x0258 PN External Interrupt Debounce Register (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.7.7.49 0x0340 PIO Group Withstand Voltage Mode Select Register (Default Value: 0x0000_0000)

When the power domain of GPIO is larger than 1.8 V, the withstand voltage is set to 3.3 V mode, the corresponding value in the 0x0340 register is set to 0. When the power domain of GPIO is 1.8 V, the withstand voltage is set to 1.8 V mode, the corresponding value in the 0x0340 register is set to 1.

Offset: 0x0340			Register Name: PIO_POW_MOD_SEL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:13	/	/	/
12	R/W	0x0	VCC_IO POWER MODE Select 0: 3.3 V 1: 1.8 V
11:2	/	/	/
2	R/W	0x0	PN_POWER MODE Select 0: 3.3 V 1: 1.8 V If PN_Port Power Source selects VCC_IO, this bit is invalid
1	R/W	0x0	PM_POWER MODE Select 0: 3.3 V 1: 1.8 V If PM_Port Power Source selects VCC_IO, this bit is invalid
0	R/W	0x0	PL_POWER MODE Select 0: 3.3 V 1: 1.8 V If PL_Port Power Source selects VCC_IO, this bit is invalid

7.7.7.50 0x0344 PIO Group Withstand Voltage Mode Select Control Register (Default Value: 0x0000_0000)

For 1.8 V and 3.3 V power, the withstand function is enabled by default, the corresponding bit in the 0x0344 register is set to 0. For 2.5 V power, the withstand function is disabled, the corresponding bit in the 0x0344 register is set to 1, and the corresponding withstand voltage in the 0x0340 register needs to be set to 3.3 V.

Offset: 0x0344			Register Name: PIO_POW_MS_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2	R/W	0x0	VCC_PN Withstand Voltage Mode Select Control 0: Enable 1: Disable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0344			Register Name: PIO_POW_MS_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
1	R/W	0x0	VCC_PM Withstand Voltage Mode Select Control 0: Enable 1: Disable
0	R/W	0x0	VCC_PL Withstand Voltage Mode Select Control 0: Enable 1: Disable

7.7.7.51 0x0348 PIO Group Power Value Register (Default Value: 0x0000_0000)

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

Offset: 0x0348			Register Name: PIO_POW_VOL_SET_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R	0x0	VCC_IO Power Value
15:3	/	/	/
2	R	0x0	PN_Port Power Value If PN_Port Power Source selects VCC_IO, this bit is invalid
1	R	0x0	PM_Port Power Value If PM_Port Power Source selects VCC_IO, this bit is invalid
0	R	0x0	PL_Port Power Value If PL_Port Power Source selects VCC_IO, this bit is invalid

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.8 GPADC

7.8.1 Overview

The General Purpose ADC (GPADC) can convert the external signal into a certain proportion of digital value, to realize the measurement of analog signal, which can be applied to power detection and key detection. This ADC is a 4-channel successive approximation register (SAR) A/D converter.

The GPADC has the following features:

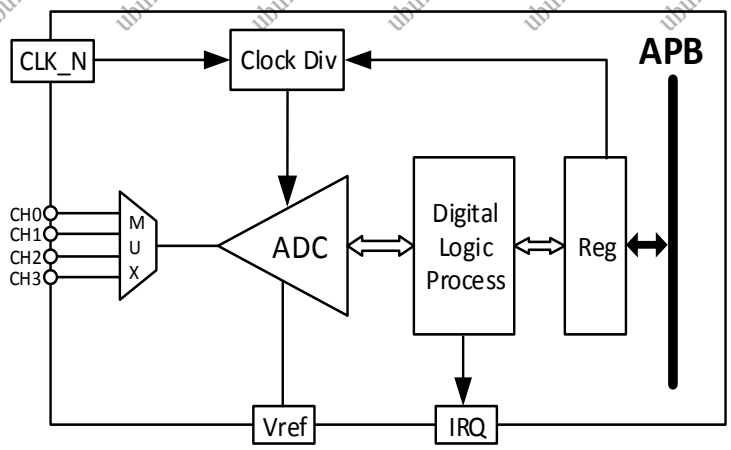
- Four independent channels
- 12-bit sampling resolution and 8-bit precision
- 64 FIFO depth of data register
- Power reference voltage: AVCC, and analog input voltage range: 0 to AVCC
- Maximum sampling frequency: 1 MHz
- Supports data compare and interrupt
- Supports DMA transport
- Supports three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

7.8.2 Block Diagram

Figure 7-67 shows the block diagram of the GPADC.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-67 GPADC Block Diagram



7.8.3 Functional Descriptions

7.8.3.1 External Signals

The following table describes the external signals of the GPADC.

Table 7-33 GPADC External Signals

Signal	Description	Type
GPADC0	ADC Input Channel0	AI
GPADC1	ADC Input Channel1	AI
GPADC2	ADC Input Channel2	AI
GPADC3	ADC Input Channel3	AI

7.8.3.2 Clock Sources

The GPADC has one clock source. The following table describes the clock source for GPADC. Users can see section 3.3 "CCU" for clock setting, configuration, and gating information.

Table 7-34 GPADC Clock Sources

Clock Sources	Description
DCXO24M	24 MHz

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.8.3.3 GPADC Work Mode

- **Single conversion mode**

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

- **Continuous conversion mode**

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

- **Burst conversion mode**

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

7.8.3.4 Clock and Timing Requirements

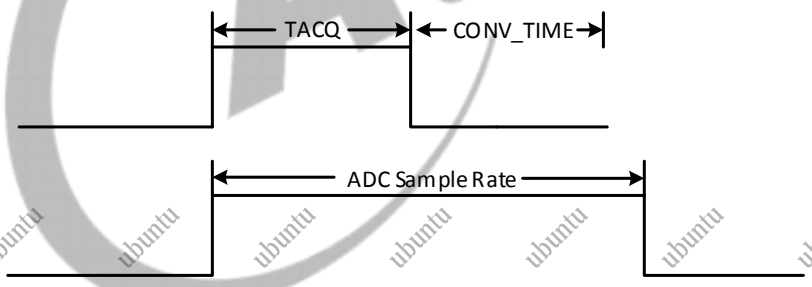
CLK_IN = 24 MHz

CONV_TIME(Conversion Time) = 1/(24MHz/14Cycles) =0.583 (us)

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

ADC Sample Frequency > TACQ+CONV_TIME

Figure 7-68 GPADC Clock and Timing Requirement



7.8.3.5 GPADC Calculate Formula

GPADC calculate formula: GPADC_DATA = Vin/V_{REF}*4096

Where:

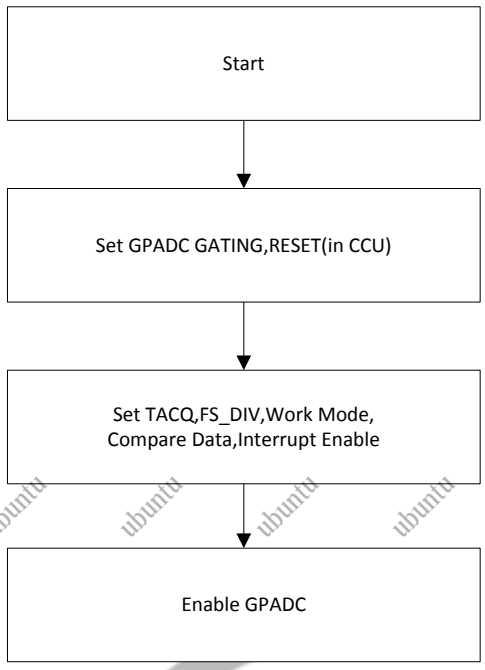
V_{REF} = 1.8 V

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.8.4 Programming Guidelines

The GPADC initial process is as follows.

Figure 7-69 GPADC Initial Process



(1).Query Mode

Step 1 Write 0x1 to the bit[16] of **GPADC_BGR_REG** to dessert reset.

Step 2 Write 0x1 to the bit[0] of **GPADC_BGR_REG** to enable the GPADC clock.

Step 3 Write 0x2F to the bit[15:0] of **GP_SR_CON** to set the acquiring time of ADC.

Step 4 Write 0x1DF to the bit[31:16] of **GP_SR_CON** to set the ADC sample frequency divider.

Step 5 Write 0x2 to the bit[19:18] of **GP_CTRL** to set the continuous conversion mode.

Step 6 Write 0x1 to the bit[0] of **GP_CS_EN** to enable the analog input channel.

Step 7 Write 0x1 to the bit[16] of **GP_CTRL** to enable the ADC function.

Step 8 Read the bit[0] of **GP_DATA_INTS**, if the bit is 1, then data conversion is complete.

Step 9 Read the bit[11:0] of **GP_CH0_DATA**, and calculate voltage value based on GPADC formula.

(2).Interrupt Mode

Step 1 Write 0x1 to the bit[16] of **GPADC_BGR_REG** to dessert reset.

Step 2 Write 0x1 to the bit[0] of **GPADC_BGR_REG** to enable the GPADC clock.

Step 3 Write 0x2F to the bit[15:0] of **GP_SR_CON** to set the acquiring time of ADC.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 4 Write 0x1DF to the bit[31:16] of **GP_SR_CON** to set the ADC sample frequency divider.

Step 5 Write 0x2 to the bit[19:18] of **GP_CTRL** to set the continuous conversion mode.

Step 6 Write 0x1 to the bit[0] of **GP_CS_EN** to enable the analog input channel.

Step 7 Write 0x1 to the bit[0] of **GP_DATA_INTC** to enable the GPADC data interrupt.

Step 8 Set the GIC module based on the IRQ 148.

Step 9 Put interrupt handler address into interrupt vector table based on the IRQ 148.

Step 10 Write 0x1 to the bit16 of **GP_CTRL** to enable the ADC function.

Step 11 Read the bit[11:0] of **GP_CH0_DATA** from the interrupt handler, calculate voltage value based on GPADC formula.

7.8.5 Register List

Module Name	Base Address
GPADC	0x07030000

Register Name	Offset	Description
GP_SR_CON	0x0000	GPADC Sample Rate Configure Register
GP_CTRL	0x0004	GPADC Control Register
GP_CS_EN	0x0008	GPADC Compare and Select Enable Register
GP_FIFO_INTC	0x000C	GPADC FIFO Interrupt Control Register
GP_FIFO_INTS	0x0010	GPADC FIFO Interrupt Status Register
GP_FIFO_DATA	0x0014	GPADC FIFO Data Register
GP_CDATA	0x0018	GPADC Calibration Data Register
GP_DATAL_INTC	0x0020	GPADC Data Low Interrupt Configure Register
GP_DATAH_INTC	0x0024	GPADC Data High Interrupt Configure Register
GP_DATA_INTC	0x0028	GPADC Data Interrupt Configure Register
GP_DATAL_INTS	0x0030	GPADC Data Low Interrupt Status Register
GP_DATAH_INTS	0x0034	GPADC Data High Interrupt Status Register
GP_DATA_INTS	0x0038	GPADC Data Interrupt Status Register
GP_CH0_CMP_DATA	0x0040	GPADC CH0 Compare Data Register
GP_CH1_CMP_DATA	0x0044	GPADC CH1 Compare Data Register
GP_CH2_CMP_DATA	0x0048	GPADC CH2 Compare Data Register
GP_CH3_CMP_DATA	0x004C	GPADC CH3 Compare Data Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
GP_CH0_DATA	0x0080	GPADC CH0 Data Register
GP_CH1_DATA	0x0084	GPADC CH1 Data Register
GP_CH2_DATA	0x0088	GPADC CH2 Data Register
GP_CH3_DATA	0x008C	GPADC CH3 Data Register

7.8.6 Register Description

7.8.6.1 0x0000 GPADC Sample Rate Configure Register (Default Value: 0x01DF_002F)

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

7.8.6.2 0x0004 GPADC Control Register (Default Value: 0x0080_0000)

Offset: 0x0004			Register Name: GP_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	R/ W	0x0	ADC_FIRST_DLY ADC First Convert Delay Setting ADC conversion of each channel is delayed by N samples.
23	R/ W	0x1	ADC_AUTOCALI_EN ADC Auto Calibration
22	/	/	/
21:20	R/W	0x0	ADC_OP_BIAS ADC OP Bias Adjust the bandwidth of the ADC amplifier

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0004			Register Name: GP_CTRL
Bit	Read/Write	Default/Hex	Description
19:18	R/W	0x0	GPADC Work Mode 00: Single conversion mode 01: Reserved 10: Continuous conversion mode 11: Burst conversion mode
17	R/W	0x0	ADC_CALI_EN ADC Calibration 1: Start Calibration, it is cleared to 0 after calibration
16	R/W	0x0	ADC_EN ADC Function Enable Before the bit is enabled, configure ADC parameters including the work mode and channel number, etc. 0: Disable 1: Enable **Note: When selecting a single conversion mode, the bit can be cleared automatically after the switch is completed.**
15:0	/	/	/

7.8.6.3 0x0008 GPADC Compare and Select Enable Register (Default Value: 0x0000_0000)

Offset: 0x0008			Register Name: GP_CS_EN
Bit	Read/Write	Default/Hex	Description
31:20	/	/	/
19	R/W	0x0	ADC_CH3_CMP_EN Channel 3 Compare Enable 0: Disable 1: Enable
18	R/W	0x0	ADC_CH2_CMP_EN Channel 2 Compare Enable 0: Disable 1: Enable
17	R/W	0x0	ADC_CH1_CMP_EN Channel 1 Compare Enable 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0008			Register Name: GP_CS_EN
Bit	**Read/Write**	**Default/Hex**	**Description**
16	R/W	0x0	ADC_CH0_CMP_EN Channel 0 Compare Enable 0: Disable 1: Enable
15:4	/	/	/
3	R/W	0x0	ADC_CH3_SELECT Analog Input Channel 3 Select 0: Disable 1: Enable
2	R/W	0x0	ADC_CH2_SELECT Analog Input Channel 2 Select 0: Disable 1: Enable
1	R/W	0x0	ADC_CH1_SELECT Analog Input Channel 1 Select 0: Disable 1: Enable
0	R/W	0x0	ADC_CH0_SELECT Analog Input Channel 0 Select 0: Disable 1: Enable

7.8.6.4 0x000C GPADC FIFO Interrupt Control Register (Default Value: 0x0000_1F00)

Offset: 0x000C			Register Name: GP_FIFO_INTC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:19	/	/	/
18	R/W	0x0	FIFO_DATA_DRQ_EN. ADC FIFO Date DRQ Enable 0: Disable 1: Enable
17	R/W	0x0	FIFO_OVERRUN_IRQ_EN ADC FIFO Overrun IRQ Enable 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

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

7.8.6.5 0x0010 GPADC FIFO Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0010			Register Name: GP_FIFO_INTS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17	R/W1C	0x0	FIFO_OVERRUN_PENDING ADC FIFO Overrun IRQ Pending 0: No Pending IRQ 1: FIFO Overrun Pending IRQ Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.
16	R/W1C	0x0	FIFO_DATA_PENDING ADC FIFO Data Available Pending Bit 0: NO Pending IRQ 1: FIFO Available Pending IRQ Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.
15:14	/	/	/
13:8	R	0x0	RXA_CNT ADC FIFO available sample word counter
7:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.8.6.6 0x0014 GPADC FIFO Data Register

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

7.8.6.7 0x0018 GPADC Calibration Data Register (Default Value: 0x0000_0000)

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

7.8.6.8 0x0020 GPADC Low Interrupt Configure Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: GP_DATAL_INTC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:4	/	/	/
3	R/W	0x0	CH3_LOW_IRQ_EN 0: Disable 1: Enable
2	R/W	0x0	CH2_LOW_IRQ_EN 0: Disable 1: Enable
1	R/W	0x0	CH1_LOW_IRQ_EN 0: Disable 1: Enable
0	R/W	0x0	CH0_LOW_IRQ_EN 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.8.6.9 0x0024 GPADC High Interrupt Configure Register (Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: GP_DATAH_INTC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:4	/	/	/
3	R/W	0x0	CH3_HIG_IRQ_EN 0: Disable 1: Enable
2	R/W	0x0	CH2_HIG_IRQ_EN 0: Disable 1: Enable
1	R/W	0x0	CH1_HIG_IRQ_EN 0: Disable 1: Enable
0	R/W	0x0	CH0_HIG_IRQ_EN 0: Disable 1: Enable

7.8.6.10 0x0028 GPADC DATA Interrupt Configure Register (Default Value: 0x0000_0000)

Offset: 0x0028			Register Name: GP_DATA_INTC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:4	/	/	/
3	R/W	0x0	CH3_DATA_IRQ_EN 0: Disable 1: Enable
2	R/W	0x0	CH2_DATA_IRQ_EN 0: Disable 1: Enable
1	R/W	0x0	CH1_DATA_IRQ_EN 0: Disable 1: Enable
0	R/W	0x0	CH0_DATA_IRQ_EN 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.8.6.11 0x0030 GPADC Low Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: GP_DATAL_INTS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:4	/	/	/
3	R/W1C	0x0	CH3_LOW_PENGDING 1: Channel 3 Voltage Low Available Pending IRQ Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.
2	R/W1C	0x0	CH2_LOW_PENGDING 1: Channel 2 Voltage Low Available Pending IRQ Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.
1	R/W1C	0x0	CH1_LOW_PENGDING 1: Channel 1 Voltage Low Available Pending IRQ Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.
0	R/W1C	0x0	CH0_LOW_PENGDING 1: Channel 0 Voltage Low Available Pending IRQ Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.

7.8.6.12 0x0034 GPADC High Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0034			Register Name: GP_DATAH_INTS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:4	/	/	/
3	R/W1C	0x0	CH3_HIG_PENGDING 0: No Pending IRQ 1: Channel 3 Voltage High Available Pending IRQ Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.
2	R/W1C	0x0	CH2_HIG_PENGDING 0: No Pending IRQ 1: Channel 2 Voltage High Available Pending IRQ Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0034			Register Name: GP_DATAH_INTS
Bit	**Read/Write**	**Default/Hex**	**Description**
1	R/W1C	0x0	CH1_HIG_PENGDING 0: No Pending IRQ 1: Channel 1 Voltage High Available Pending IRQ Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.
0	R/W1C	0x0	CH0_HIG_PENGDING 0: No Pending IRQ 1: Channel 0 Voltage High Available Pending IRQ Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.

7.8.6.13 0x0038 GPADC Data Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0038			Register Name: GP_DATA_INTS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:4	/	/	/
3	R/W1C	0x0	CH3_DATA_PENGDING 0: No Pending IRQ 1: Channel 3 Data Available Pending IRQ Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.
2	R/W1C	0x0	CH2_DATA_PENGDING 0: No Pending IRQ 1: Channel 2 Data Available Pending IRQ Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.
1	R/W1C	0x0	CH1_DATA_PENGDING 0: No Pending IRQ 1: Channel 1 Data Available Pending IRQ Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.
0	R/W1C	0x0	CH0_DATA_PENGDING 0: No Pending IRQ 1: Channel 0 Data Available Pending IRQ Write '1' to clear this interrupt or automatically clear if the interrupt condition fails.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.8.6.14 0x0040 GPADC CH0 Compare Data Register (Default Value: 0x0BFF_0400)

Offset: 0x0040			Register Name: GP_CH0_CMP_DATA
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0xBFF	CH0_CMP_HIG_DATA Channel 0 Voltage High Value
15:12	/	/	/
11:0	R/W	0x400	CH0_CMP_LOW_DATA Channel 0 Voltage Low Value

7.8.6.15 0x0044 GPADC CH1 Compare Data Register (Default Value: 0x0BFF_0400)

Offset: 0x0044			Register Name: GP_CH1_CMP_DATA
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0xBFF	CH1_CMP_HIG_DATA Channel 1 Voltage High Value
15:12	/	/	/
11:0	R/W	0x400	CH1_CMP_LOW_DATA Channel 1 Voltage Low Value

7.8.6.16 0x0048 GPADC CH2 Compare Data Register (Default Value: 0x0BFF_0400)

Offset: 0x0048			Register Name: GP_CH2_CMP_DATA
Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0xBFF	CH2_CMP_HIG_DATA Channel 2 Voltage High Value
15:12	/	/	/
11:0	R/W	0x400	CH2_CMP_LOW_DATA Channel 2 Voltage Low Value

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.8.6.17 0x004C GPADC CH3 Compare Data Register (Default Value: 0x0BFF_0400)

Offset: 0x004C			Register Name: GP_CH3_CMP_DATA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	/	/	/
27:16	R/W	0xBFF	CH3_CMP_HIG_DATA Channel 3 Voltage High Value
15:12	/	/	/
11:0	R/W	0x400	CH3_CMP_LOW_DATA Channel 3 Voltage Low Value

7.8.6.18 0x0080 GPADC CH0 Data Register (Default Value: 0x0000_0000)

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

7.8.6.19 0x0084 GPADC CH1 Data Register (Default Value: 0x0000_0000)

Offset: 0x0084			Register Name: GP_CH1_DATA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:12	/	/	/
11:0	R	0x000	GP_CH1_DATA Channel 1 Data

7.8.6.20 0x0088 GPADC CH2 Data Register (Default Value: 0x0000_0000)

Offset: 0x0088			Register Name: GP_CH2_DATA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:12	/	/	/
11:0	R	0x000	GP_CH2_DATA Channel 2 Data

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.8.6.21 0x008C GPADC CH3 Data Register (Default Value: 0x0000_0000)

Offset: 0x008C			Register Name: GP_CH3_DATA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:12	/	/	/
11:0	R	0x000	GP_CH3_DATA Channel 3 Data

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.9 LRADC

7.9.1 Overview

The low rate analog-to-digital converter (LRADC) can convert the external signal into a certain proportion of digital value, to realize the measurement of analog signal, which can be applied to power detection and key detection.

The LRADC has the following features:

- 6-bit sampling resolution and 5-bit precision
- Sample rate up to 2 kHz
- Supports hold Key and general Key
- Supports normal, continuous and single working mode
- Power supply voltage: AVCC, and power reference voltage: 0.75*AVCC, analog input and detected voltage range: 0 to 1.266 V
- Supports interrupt

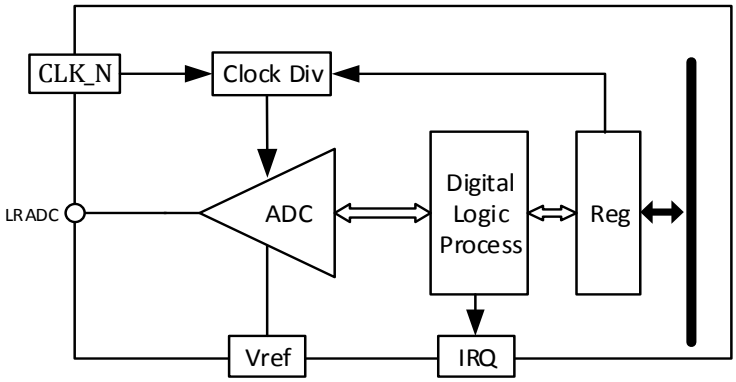
NOTE

The LRADC has a 6-bit resolution, 1-bit offset error, and 1-bit precision error. After the LRADC calibrates 1-bit offset error, the LRADC has 5-bit precision.

7.9.2 Block Diagram

Figure 7-70 shows the block diagram of the LRADC.

Figure 7-70 LRADC Block Diagram



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.9.3 Functional Descriptions

7.9.3.1 External Signals

The following table describes the external signals of the LRADC. The LRADC pin is the analog input signal.

Table 7-35 LRADC External Signals

Signal	Description	Type
LRADC	Analog Input Channel0	AI

7.9.3.2 Clock Sources

The LRADC has one clock source. The following table describes the clock source for LRADC.

Table 7-36 LRADC Clock Sources

Clock Sources	Description
LOSC	32.768 kHz LOSC

7.9.3.3 LRADC Working Mode

- **Normal Mode**

  The LRADC gathers 8 samples, the average value of these 8 samples is updated in the data register, and the data interrupt sign is enabled. It is sampled repeatedly according to this mode until the LRADC is disabled.

- **Continuous Mode**

  The LRADC gathers 8 samples every other 8*(N+1) sample cycle. The average value of every 8 samples is updated in the data register, and the data interrupt sign is enabled. (N is defined in the bit[19:16] of LRADC_CTRL).

- **Single Mode**

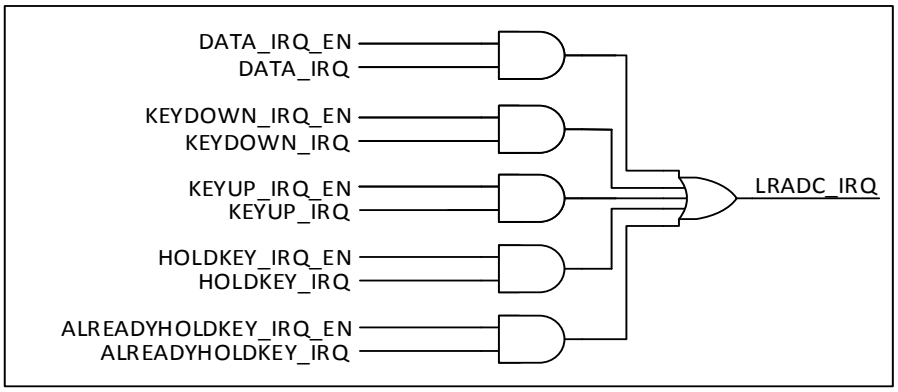
  The LRADC gathers 8 samples, and the average value of these 8 samples is updated in the data register, and the data interrupt sign is enabled at the same time, then the LRADC stops sample.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.9.3.4 Interrupt

Each LRADC channel has five interrupt sources and five interrupt enable controls.

Figure 7-71 LRADC Interrupt



When the input voltage is between LEVEL A (1.35V) and LEVEL B (control by the bit[5:4] of **LRADC_CTRL**), the IRQ1 can be generated. When the input voltage is lower than LEVEL B, the IRQ2 can be generated.

If the controller receives IRQ1 and does not receive IRQ2 at the same time, then the controller will generate Hold Key Pending, otherwise Data IRQ Pending.

The Hold KEY usually is used for the self-locking key. When the self-locking key holds a locking status, the controller receives the IRQ2, then the controller will generate Already Hold Pending.

7.9.3.5 Calculation Formula

Calculation formula: LRADC_DATA = $Vin/V_{REF}*64$, V_{REF}=1.35 V

7.9.4 Programming Guidelines

7.9.4.1 Normal Detecting

Perform the following steps for normal detecting mode:

Step 1 Configure **LRADC_BGR_REG**[LRADC_GATING] to 0 to disable the clock of LRADC.

Step 2 Configure **LRADC_BGR_REG**[LRADC_RST] to 1 to deassert the reset of LRADC.

Step 3 Configure **LRADC_BGR_REG**[LRADC_GATING] to 1 to enable the clock of LRADC.

Step 4 Configure **LRADC_CTRL**[LRADC_SAMPLE_RATE] to set the appropriate sampling frequency.

Step 5 Configure **LRADC_CTRL**[LEVELB_VOL] to set the appropriate voltage threshold.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 6 Configure **LRADC_CTRL**[FIRST_CONVER_DLY] and **LRADC_CTRL**[LEVELA_B_CNT] to set the appropriate debounce value.

Step 7 Configure **LRADC_CTRL**[LRADC_HOLD_KEY_EN] to 1.

Step 8 Configure **LRADC_CTRL**[KEY_MODE_SELECT] to 0 to set the normal mode.

Step 9 Configure **LRADC_INTC** to enable the corresponding interrupt.

Step 10 Configure **LRADC_CTRL**[LRADC_HOLD_KEY_EN] to 1.

Step 11 Read the corresponding key voltage value from **LRADC_DATA** when the CPU receives the LRADC interrupt.

7.9.4.2 Single Detecting

Perform the following steps for the single detecting mode:

Step 1 Configure **LRADC_BGR_REG**[LRADC_GATING] to 0 to disable the clock of LRADC.

Step 2 Configure **LRADC_BGR_REG**[LRADC_RST] to 1 to deassert the reset of LRADC.

Step 3 Configure **LRADC_BGR_REG**[LRADC_GATING] to 1 to enable the clock of LRADC.

Step 4 Configure **LRADC_CTRL**[LRADC_SAMPLE_RATE] to set the appropriate sampling frequency.

Step 5 Configure **LRADC_CTRL**[LEVELB_VOL] to set the appropriate voltage threshold.

Step 6 Configure **LRADC_CTRL**[FIRST_CONVER_DLY] and **LRADC_CTRL**[LEVELA_B_CNT] to set the appropriate debounce value.

Step 7 Configure **LRADC_CTRL**[LRADC_HOLD_KEY_EN] to 1.

Step 8 Configure **LRADC_CTRL**[KEY_MODE_SELECT] to 1 to set the single mode.

Step 9 Configure **LRADC_INTC** to enable the corresponding interrupt.

Step 10 Configure **LRADC_CTRL**[LRADC_HOLD_KEY_EN] to 1.

Step 11 Read the corresponding key voltage value from **LRADC_DATA** when the CPU receives the LRADC interrupt.

7.9.4.3 Continuous Detecting

Perform the following steps for continuous detecting mode:

Step 1 Configure **LRADC_BGR_REG**[LRADC_GATING] to 0 to disable the clock of LRADC.

Step 2 Configure **LRADC_BGR_REG**[LRADC_RST] to 1 to deassert the reset of LRADC.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 3 Configure **LRADC_BGR_REG**[LRADC_GATING] to 1 to enable the clock of LRADC.

Step 4 Configure **LRADC_CTRL**[LRADC_SAMPLE_RATE] to set the appropriate sampling frequency.

Step 5 Configure **LRADC_CTRL**[LEVELB_VOL] to set the appropriate voltage threshold.

Step 6 Configure **LRADC_CTRL**[FIRST_CONVER_DLY] and **LRADC_CTRL**[LEVELA_B_CNT] to set the appropriate debounce value.

Step 7 Configure **LRADC_CTRL**[LRADC_HOLD_KEY_EN] to 1.

Step 8 Configure **LRADC_CTRL**[KEY_MODE_SELECT] to 2 to set the continuous mode, and configure **LRADC_CTRL**[CONTINUE_TIME_SELECT] to set a sampling interval.

Step 9 Configure **LRADC_INTC** to enable the corresponding interrupt.

Step 10 Configure **LRADC_CTRL**[LRADC_HOLD_KEY_EN] to 1.

Step 11 Read the corresponding key voltage value from **LRADC_DATA** when the CPU receives the LRADC interrupt.

7.9.5 Register List

Module Name	Base Address
LRADC	0x07030800

Register Name	Offset	Description
LRADC_CTRL	0x0000	LRADC Control Register
LRADC_INTC	0x0004	LRADC Interrupt Control Register
LRADC_INTS	0x0008	LRADC Interrupt Status Register
LRADC_DATA	0x000C	LRADC Data Register

7.9.6 Register Description

7.9.6.1 0x0000 LRADC Control Register (Default Value: 0x0100_0168)

Offset: 0x0000			Register Name: LRADC_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	R/W	0x1	FIRST_CONVERT_DLY ADC First Convert Delay Setting ADC conversion is delayed by n samples.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: LRADC_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
23:20	/	/	/
19:16	R/W	0x0	CONTINUE_TIME_SELECT Continuous Mode Time Select One of 8*(N+1) sample as a valuable sample data.
15:14	/	/	/
13:12	R/W	0x0	KEY_MODE_SELECT Key Mode Select 00: Normal Mode 01: Single Mode 10: Continuous Mode
11:8	R/W	0x1	LEVELA_B_CNT Level A to Level B time threshold select Judge the ADC convert value from level A to level B in n+1 samples.
7	R/W	0x0	LRADC_HOLD_KEY_EN LRADC Hold KEY Enable 0: Disable 1: Enable
6	R/W	0x1	LRADC_CHANNEL_EN LRADC Channel Enable 0: Disable 1: Enable
5:4	R/W	0x2	LEVELB_VOL Level B Corresponding Data Value Setting (the real voltage value) 00: Reserved 01: 0x39 (1.221 V) 10: 0x36 (1.157 V) 11: 0x33 (1.093 V)
3:2	R/W	0x2	LRADC_SAMPLE_RATE LRADC Sample Rate 00: 2 kHz 01: 1 kHz 10: 500 Hz 11: 250 Hz
1	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: LRADC_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
0	R/W	0x0	LRADC_EN LRADC Enable 0: Disable 1: Enable

7.9.6.2 0x0004 LRADC Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0004			Register Name: LRADC_INTC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:5	/	/	/
4	R/W	0x0	ADC0_KEYUP_IRQ_EN ADC0 Key Up IRQ Enable 0: Disable 1: Enable
3	R/W	0x0	ADC0_ALRDY_HOLD_IRQ_EN ADC0 Already Hold Key IRQ Enable 0: Disable 1: Enable
2	R/W	0x0	ADC0_HOLD_IRQ_EN ADC0 Hold Key IRQ Enable 0: Disable 1: Enable
1	R/W	0x0	ADC0_KEYDOWN_EN ADC0 Key Down Enable 0: Disable 1: Enable
0	R/W	0x0	ADC0_DATA_IRQ_EN ADC0 Data IRQ Enable 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.9.6.3 0x0008 LRADC Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0008			Register Name: LRADC_INTS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:5	/	/	/
4	R/W1C	0x0	ADC0_KEYUP_PENDING ADC0 Key up Pending Bit When the general key is pulled up, and the corresponding interrupt is enabled, the status bit is set. 0: No IRQ 1: IRQ Pending Writing 1 to the bit will clear it and its corresponding interrupt if the interrupt is enabled.
3	R/W1C	0x0	ADC0_ALRDY_HOLD_PENDING ADC0 Already Hold Pending Bit When the hold key is pulled down and the general key is pulled down, and the corresponding interrupt is enabled. 0: No IRQ 1: IRQ Pending Writing 1 to the bit will clear it and its corresponding interrupt if the interrupt is enabled.
2	R/W1C	0x0	ADC0_HOLDKEY_PENDING ADC0 Hold Key Pending Bit When the hold key is pulled down, and the corresponding interrupt is enabled, the status bit is set and the interrupt line is set. 0: NO IRQ 1: IRQ Pending Writing 1 to the bit will clear it and its corresponding interrupt if the interrupt is enabled.
1	R/W1C	0x0	ADC0_KEYDOWN_PENDING ADC0 Key Down IRQ Pending Bit When the general key is pulled down, and the corresponding interrupt is enabled, the status bit is set and the interrupt line is set. 0: No IRQ 1: IRQ Pending Writing 1 to the bit will clear it and its corresponding interrupt if the interrupt is enabled.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0008			Register Name: LRADC_INTS
Bit	**Read/Write**	**Default/Hex**	**Description**
0	R/W1C	0x0	ADC0_DATA_PENDING ADC0 Data IRQ Pending Bit 0: No IRQ 1: IRQ Pending Writing 1 to the bit will clear it and its corresponding interrupt if the interrupt is enabled.

7.9.6.4 0x000C LRADC Data Register (Default Value: 0x0000_003F)

Offset: 0x000C			Register Name: LRADC_DATA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:6	/	/	/
5:0	R	0x3F	LRADC_DATA LRADC Data

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.10 PWM

7.10.1 Overview

The PWM controller has 9 PWM channels in CPUX domain (PWM0 to PWM8), and divides to 4 PWM pairs: PWM01 pair (PWM0 + PWM1), PWM23 pair (PWM2 + PWM3), PWM45 pair (PWM4 + PWM5), PWM67 pair (PWM6 + PWM7). The PWM8 is a single channel and has no pair function. The PWM controller has 6 S-PWM channels in CPUS domain (S-PWM0 to S-PWM5) and divides to 3 S-PWM pairs.

The PWM has the following features:

- 9 PWM channels in CPUX domain (4 PWM pairs + 1 PWM single channel), and 6 S-PWM channels in CPUS domain (3 PWM pairs)
- Supports pulse, cycle and complementary pair output
- Supports capture input
- Programming dead-zone output
- Build-in the programmable dead-time generator, controllable dead-time
- Three kinds of output waveform: continuous waveform, pulse waveform, and complementary pair
- Output frequency range: 0 to 24 MHz or 100 MHz
- Various duty-cycle: 0% to 100%
- Minimum resolution: 1/65536
- Interrupt generation of PWM output and capture input

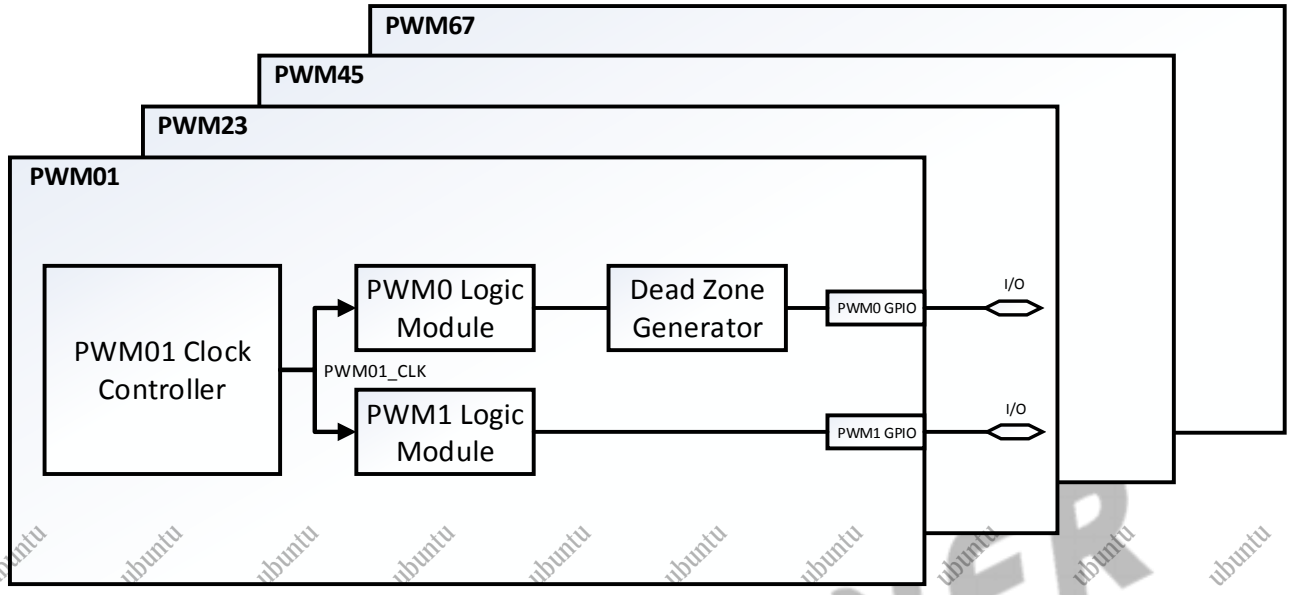
NOTE

The following description focuses on the PWM of the CPUX domain. The PWM module in CPUS domain has all characteristics of the PWM module in CPUX domain.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.10.2 Block Diagram

Figure 7-72 PWM Block Diagram



Each PWM pair consists of 1 clock module, 2 timer logic module, and 1 programmable dead-zone generator.

7.10.3 Functional Descriptions

7.10.3.1 External Signals

The following table describes the external signals of the PWM.

Table 7-37 PWM External Signals

Signal	Description	Type
PWM0	Pulse Width Module Channel0	I/O
PWM1	Pulse Width Module Channel1	I/O
PWM2	Pulse Width Module Channel2	I/O
PWM3	Pulse Width Module Channel3	I/O
PWM4	Pulse Width Module Channel4	I/O
PWM5	Pulse Width Module Channel5	I/O
PWM6	Pulse Width Module Channel6	I/O
PWM7	Pulse Width Module Channel7	I/O
PWM8	Pulse Width Module Channel8	I/O
S-PWM0	Pulse Width Module Channel0 (CPUS domain)	I/O
S-PWM1	Pulse Width Module Channel1 (CPUS domain)	I/O

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Signal	Description	Type
S-PWM2	Pulse Width Module Channel2 (CPUS domain)	I/O
S-PWM3	Pulse Width Module Channel3 (CPUS domain)	I/O
S-PWM4	Pulse Width Module Channel4 (CPUS domain)	I/O
S-PWM5	Pulse Width Module Channel5 (CPUS domain)	I/O

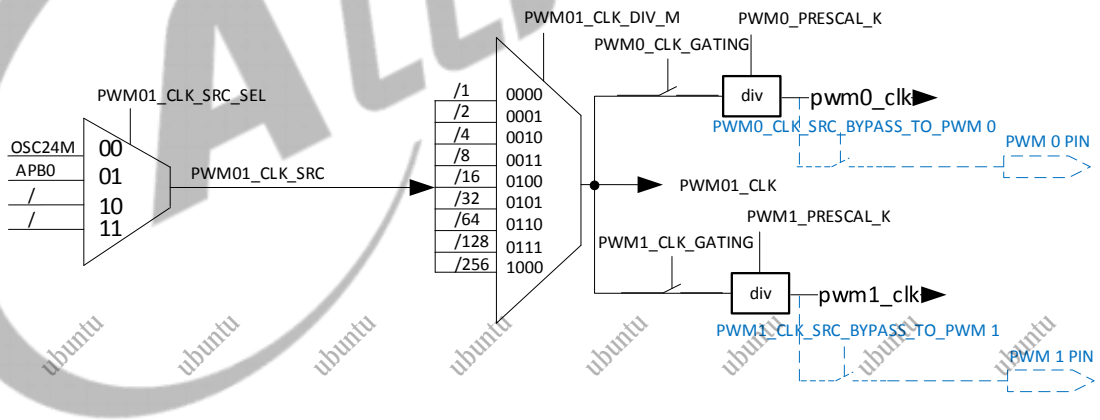
7.10.3.2 Typical Application

- Suitable for display device, such as LCD
- Suitable for electric motor control

7.10.3.3 Clock Controller

Using PWM01 as an example. The other PWM pairs are the same as PWM01.

Figure 7-73 PWM01 Clock Controller Diagram



The clock controller includes clock source select (PWM01_CLK_SRC), 1~256 scaler (PWM01_CLK_DIV_M), frequency division (PWM_PRESCAL_K), clock source bypass (PWMx_CLK_BYPASS) and clock switch (PWMx_CLK_GATING).

The clock sources have OSC24M and APB0 (For S-PWM in CPUS domain, the clock sources have OSC24M and APBS0). The OSC24M comes from the external high-frequency oscillator; the APB0 is APB0 bus clock.

The bypass function of the clock source is that the clock source directly accesses PWM output, the PWM output waveform is the waveform of the clock controller output. The BYPASS gridlines in the above figure indicate the

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

bypass function of the clock source, see Figure 7-73 for the details about implement. At last, the output clock of the clock controller is sent to the PWM logic module.

7.10.3.4 PWM Output

Taking PWM01 as an example, Figure 7-74 indicates the PWM01 output logic diagram. The logic diagrams of other PWM pairs are the same as PWM01.

The timer logic module of PWM consists of one 16-bit up-counter (**PCNTR**) and three 16-bit parameters (PWM_ENTIRE_CYCLE, PWM_ACT_CYCLE, PWM_COUNTER_START). The PWM_ENTIRE_CYCLE is used to control the PWM cycle, the PWM_ACT_CYCLE is used to control the duty-cycle, the PWM_COUNTER_START is used to control the output phase (multi-channel synchronization work requirements).

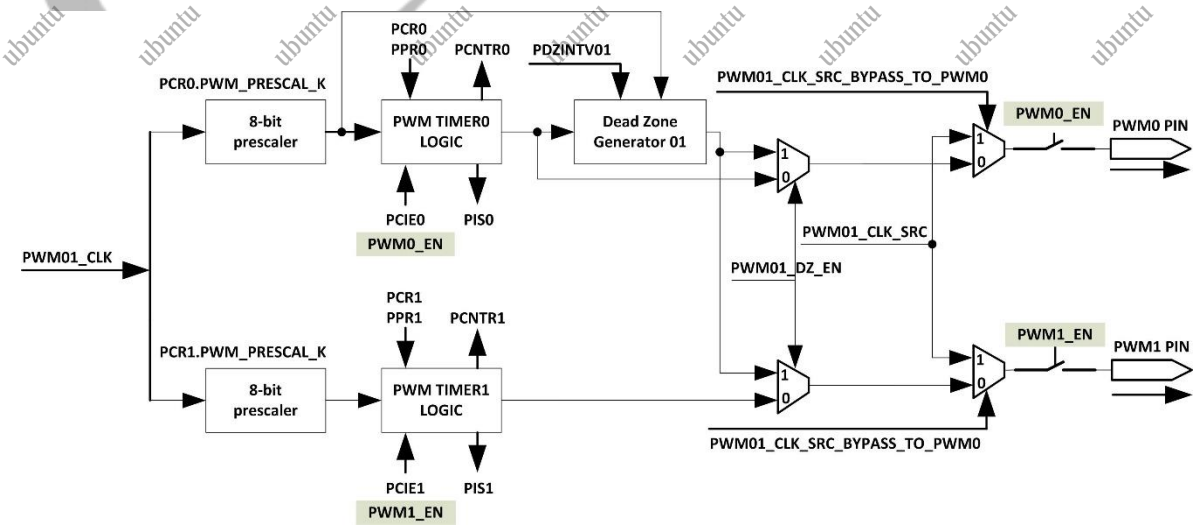
The PWM_ENTIRE_CYCLE and the PWM_ACT_CYCLE support the cache load, after PWM output is enabled, the register values of the PWM_ENTIRE_CYCLE and the PWM_ACT_CYCLE can be changed anytime, the changed value caches into the cache register. When the PCNTR counter outputs a period of PWM waveform, the value of the cache register can be updated for the PCNTR control. The purpose of the cache load is to avoid the unstable PWM output waveform with the burred feature when updating the values of the PWM_ENTIRE_CYCLE and PWM_ACT_CYCLE.

The PWM supports cycle and pulse waveform output.

Cycle mode: The PWM outputs the setting PWM waveform continually, that is, the output waveform is a continuous PWM square wave.

Pulse mode: After setting the PWM_PUL_NUM parameter, the PWM outputs (PWM_PULNUM+1) periods of PWM waveform, that is, the waveform with several pulses are output.

Figure 7-74 PWM01 Output Logic Module Diagram



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.10.3.5 Up-Counter and Comparator

The period, duty-cycle, and phase of PWM output waveform are decided by the **PCNTR**, PWM_ENTIRE_CYCLE, PWM_ACT_CYCLE, and PWM_COUNTER_START. The rules are as follows.

- PCNTR= (PCNTR==PWM_ENTIRE_CYCLE)?0 : PCNTR + 1
- PCNTR starts to count by PWM_COUNTER_START, the counter of a PWM period is (PWM_ENTIRE_CYCLE+1).
- PCNTR > (PWM_ENTIRE_CYCLE - PWM_ACT_CYCLE), output "active state"
- PCNTR <= (PWM_ENTIRE_CYCLE - PWM_ACT_CYCLE), output "~ (active state)"

Active state of PWM0 channel is high level (PCR0. PWM_ACT_STA = 1)

When PCNTR0 > (PPR0. PWM_ENTIRE_CYCLE - PPR0.PWM_ACT_CYCLE), then PWM0 outputs 1 (high level).

When PCNTR0 <= (PPR0. PWM_ENTIRE_CYCLE - PPR0.PWM_ACT_CYCLE), then PWM0 outputs 0 (low level).

The formula of the output period and the duty-cycle for PWM are as follows.

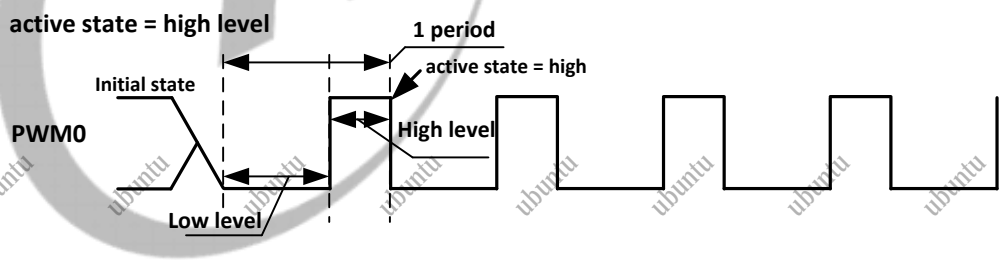
T_{period} = (PWM01_CLK / PWM0_PRESCALE_K)$^{-1}$ * (PPR0.PWM_ENTIRE_CYCLE + 1)

$T_{high\text{-}level}$ = (PWM01_CLK / PWM0_PRESCALE_K)$^{-1}$ * PPR0.PWM_ACT_CYCLE

$T_{low\text{-}level}$ = (PWM01_CLK / PWM0_PRESCALE_K)$^{-1}$ * (PPR0.PWM_ENTIRE_CYCLE + 1 - PPR0.PWM_ACT_CYCLE)

Duty-cycle = (high level time) / (1 period time) = $T_{high\text{-}level}$ / T_{period}

Figure 7-75 PWM0 High Level Active State



Active state of PWM0 channel is low level (PCR0. PWM_ACT_STA = 0)

When PCNTR0 > (PPR0.PWM_ENTIRE_CYCLE - PPR0.PWM_ACT_CYCLE), then PWM0 outputs 0.

When PCNTR0 <= (PPR0.PWM_ENTIRE_CYCLE - PPR0.PWM_ACT_CYCLE), then PWM0 outputs 1.

The formula of the output period and the duty-cycle for PWM are as follows.

T_{period} = (PWM01_CLK / PWM0_PRESCALE_K)$^{-1}$ * (PPR0.PWM_ENTIRE_CYCLE + 1)

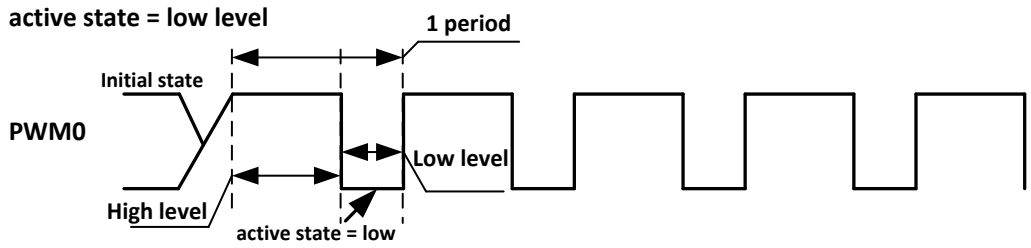
$T_{high\text{-}level}$ = (PWM01_CLK / PWM0_PRESCALE_K)$^{-1}$ * (PPR0.PWM_ENTIRE_CYCLE + 1 - PPR0.PWM_ACT_CYCLE)

$T_{low\text{-}level}$ = (PWM01_CLK / PWM0_PRESCALE_K)$^{-1}$ * PPR0.PWM_ACT_CYCLE

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

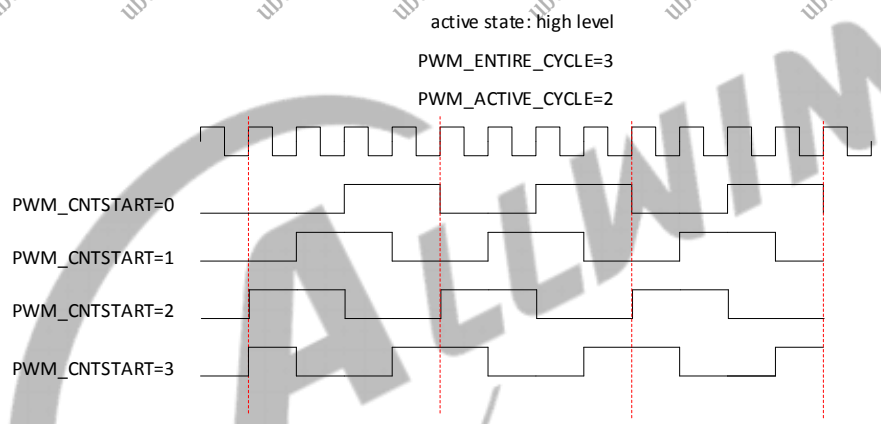
Duty-cycle = (low level time) / (1 period time) = $T_{low\text{-}level}$ / T_{period}

Figure 7-76 PWM0 Low Level Active State



The counter of PCNTR starts from 0 by default, it can output the pulse control of the waveform by setting PWM_COUNTER_START. The figure is as follows.

Figure 7-77 Phase of PWM0 High Level Active State

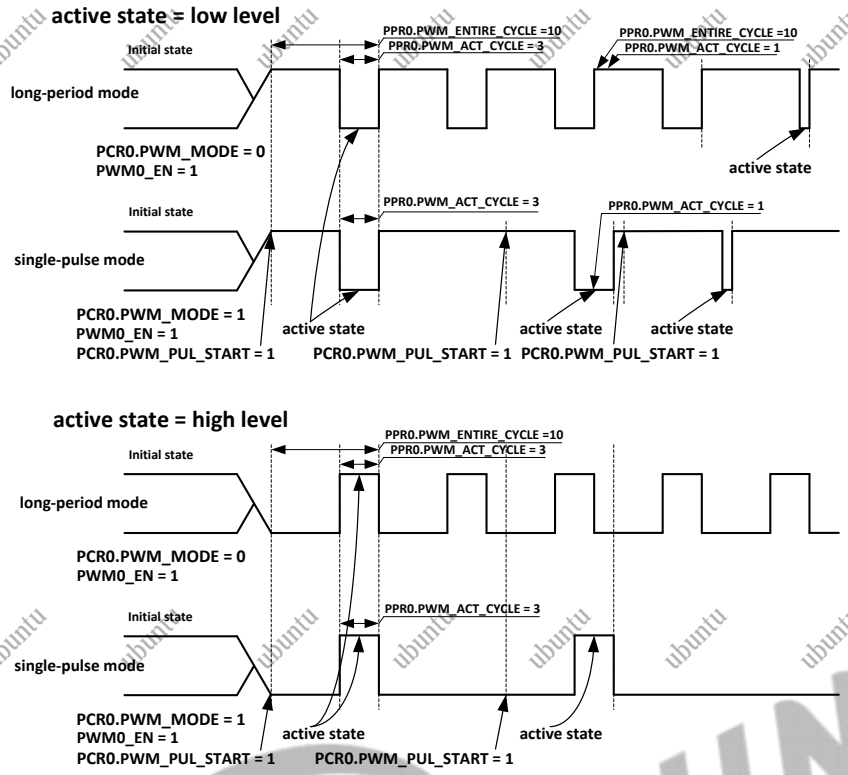


7.10.3.6 Pulse Mode and Cycle Mode

The PWM output supports pulse mode and cycle mode. PWM in pulse mode outputs one pulse waveform, but PWM in cycle mode outputs continuous waveform. Figure 7-78 shows the PWM output waveform in pulse mode and cycle mode.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-78 PWM0 Output Waveform in Pulse Mode and Cycle Mode



Each channel of the PWM module supports the PWM output of pulse mode and cycle mode, the active state of the PWM output waveform can be programmed to control.

When **PCR**0[PWM_MODE] is 0, the PWM0 outputs in cycle mode. When **PCR**0[PWM_MODE] is 1, the PWM0 outputs in pulse mode.

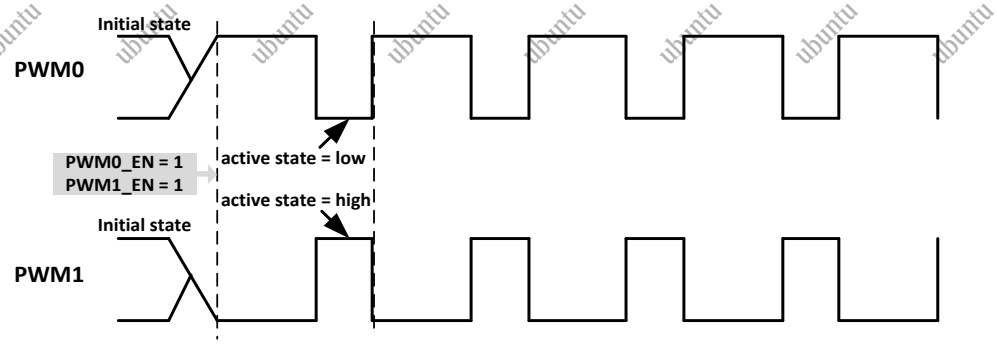
Specifically, in pulse mode, after the PWM0 channel enabled, **PCR**0[PWM_PUL_START] needs to be set to 1 when the PWM0 needs to output pulse waveform, after completed the output, **PCR**0[PWM_PUL_START] can be cleared to 0 by hardware. The next setting 1 can be operated after **PCR**0[PWM_PUL_START] is cleared.

7.10.3.7 Complementary Pair Output

Every PWM pair supports complementary pair output and PWM pair with dead-time. Figure 7-79 shows the complementary pair output of PWM01.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-79 PWM01 Complementary Pair Output



The complementary pair output needs to satisfy the following conditions:

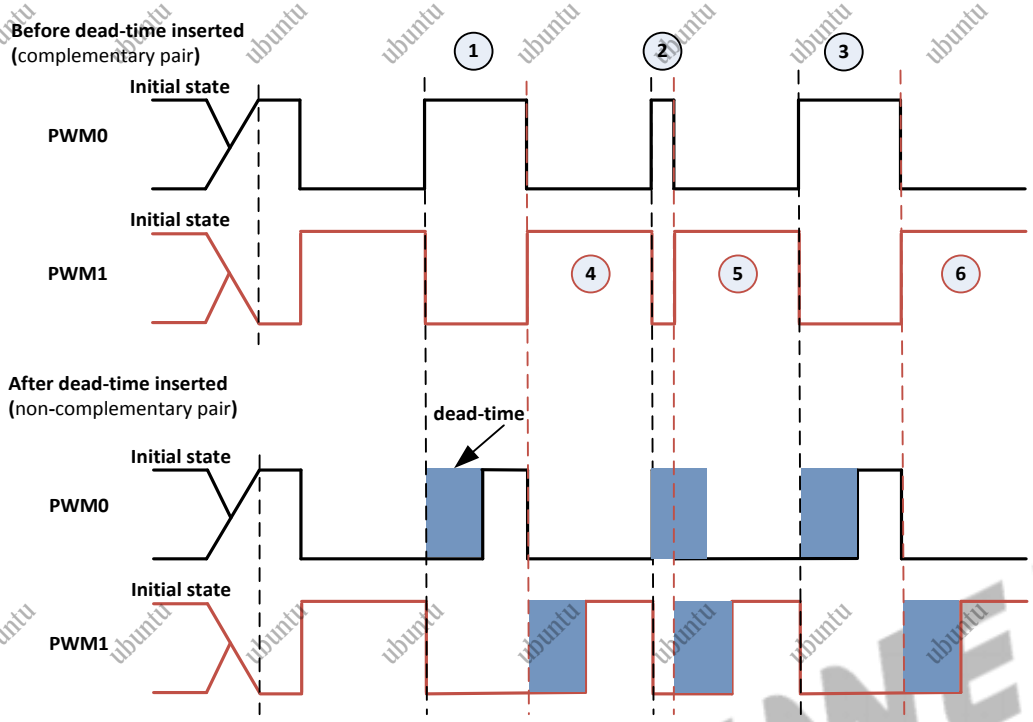
- PWM0 and PWM1 have the same clock divider, frequency, duty-cycle, pulse
- PWM0 and PWM1 have an opposite active state
- Enable the clock gating of PWM0 and PWM1 at the same time
- Enable the waveform output of PWM0 and PWM1 at the same time

7.10.3.8 Dead-time Generator

Every PWM pair has a programmable dead-time generator. When the dead-time function of the PWM pair enabled, the PWM01 output waveform is decided by PWM timer logic and DeadZone Generator. Figure 7-80 shows the output waveform.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-80 Dead-time Output Waveform



The PWM waveform before the insertion of dead-time indicates a complementary waveform pair of non-inserted dead-time in Dead Zone Generator 01.

The PWM waveform after the insertion of dead-time indicates a non-complementary PWM waveform pair inserted dead-time in a complementary waveform pair of Dead Zone Generator 01. The PWM waveform pair at last outputs to PWM0 pin and PWM1 pin.

For the complementary pair of Dead Zone Generator 01, the principle of inserting dead-time is that to insert dead-time as soon as the rising edge came. If the high level time for mark② in the above figure is less than dead-time, then dead-time will override the high level. The setting of dead-time needs to consider the period and the duty-cycle of the output waveform. The dead-time formula is defined as follows:

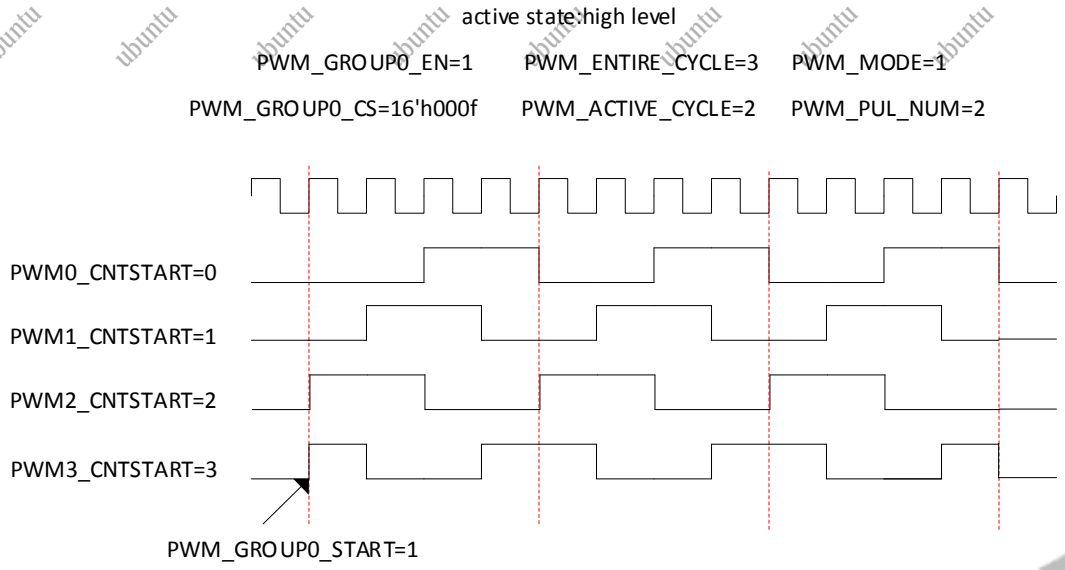
Dead-time = (PWM01_CLK / PWM0_PRESCALE_K)$^{-1}$ * PDZINTV01

7.10.3.9 PWM Group Mode

Taking PWM Group0 as an example. The same group of PWM channel is selected to work by PGR0.CS; the same PWM_ENTIRE_CYCLE, PWM_ACT_CYCLE are set by the same clock configuration; the different PWM_COUNTER_START can output PWM group signals with the same duty-cycle and the different phase.

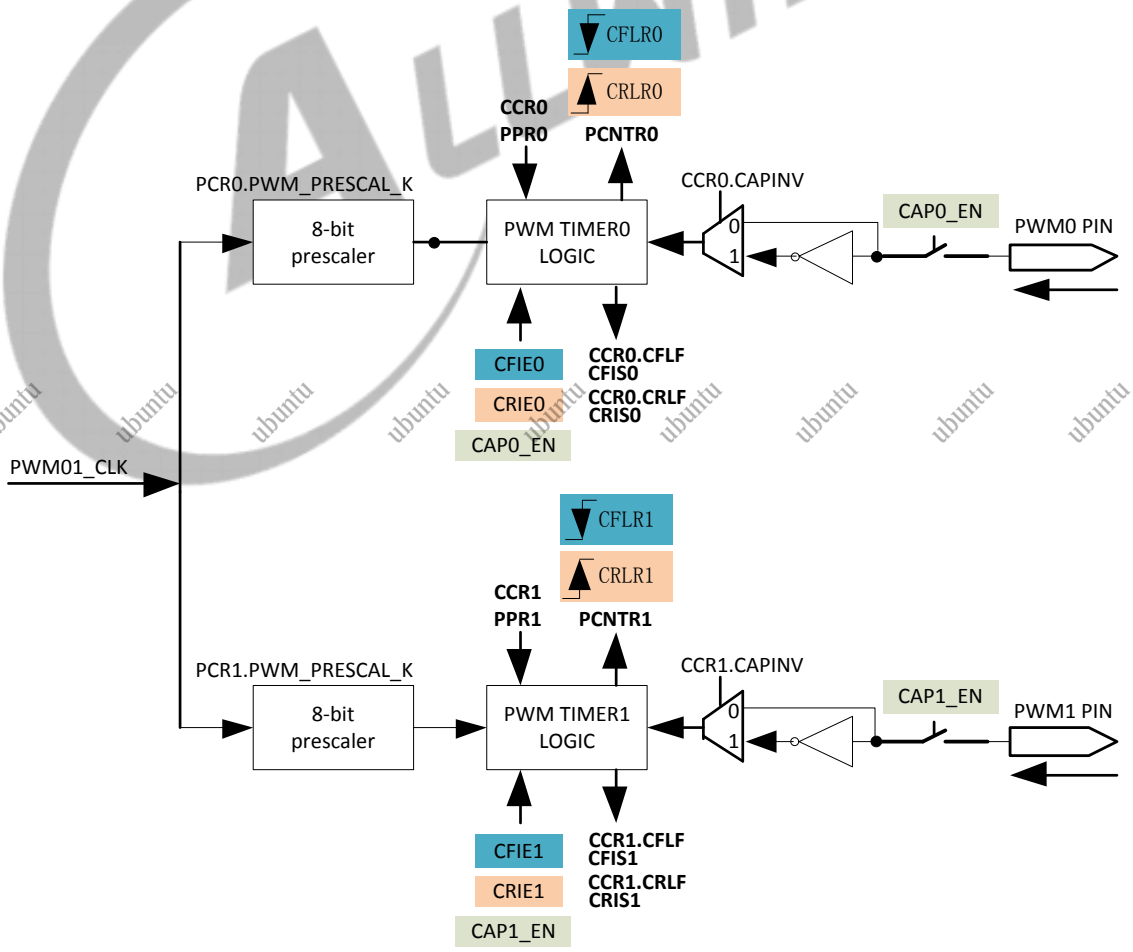
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-81 Group 0~3 PWM Signal Output



7.10.3.10 Capture Input

Figure 7-82 PWM01 Capture Logic Module Diagram



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

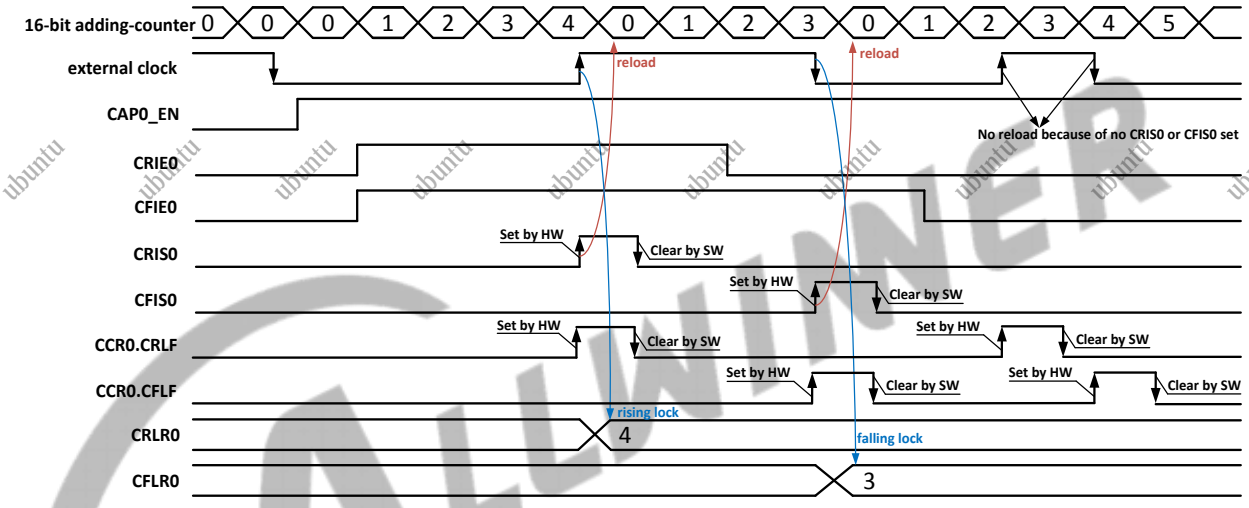
Besides the timer logic module of every PWM channel generates PWM output, it can be used to capture the rising edge and the falling edge of the external clock. Using the PWM0 channel as an example, the PWM0 channel has one **CFLR0** and one **CRLR0** for capturing up-counter value on the falling edge and rising edge, respectively. You can calculate the period of the external clock by **CFLR0** and **CRLR0**.

$T_{high\text{-}level}$ = (PWM01_CLK / PWM0_PRESCALE_K)$^{-1}$ * CRLR0

$T_{low\text{-}level}$ = (PWM01_CLK / PWM0_PRESCALE_K)$^{-1}$ * CFLR0

T_{period} = $T_{high\text{-}level}$ + $T_{low\text{-}level}$

Figure 7-83 PWM0 Channel Capture Timing



When the capture input function of the PWM0 channel is enabled, the PCNTR of the PWM0 channel starts to work.

When the timer logic module of PWM0 captures a rising edge, the current value of the up-counter is locked to **CRLR**0 and **CCR**0[CRLF] is set to 1. If CRIE0 is 1, then CRIS0 is set to 1, the PWM0 channel sends interrupt requests, and the up-counter is loaded to 0 and continues to count. If CRIE0 is 0, the timer logic module of PWM0 captures a rising edge, CRIS0 cannot be set to 1, the up-counter is not loaded to 0.

When the timer logic module of PWM0 captures one falling edge, the current value of PCNTR is locked to **CFLR**0 and **CCR**0[CFLF] is set to 1. If CFIE0 is 1, then CFIS0 is set to 1, the PWM0 channel sends interrupt requests, and the up-counter is loaded to 0 and continues to count. If CFIE0 is 0, the timer logic module of PWM0 captures a falling edge, CFIS0 cannot be set to 1, the up-counter is not loaded to 0.

7.10.3.11 Interrupt

The PWM supports an interrupt generation when configuring the PWM channel to PWM output or capturing input.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

For PWM output function, when the controller outputs one period of PWM waveform in cycle mode, the PIS of the corresponding PWM channel is set to 1; when the controller outputs (PWM_PULNUM+1) periods of PWM waveform in pulse mode, the PIS of the corresponding PWM channel is set to 1.

NOTE

The PIS bit is set to 1 automatically by hardware and cleared by software.

For capturing input function, when the timer logic module of the capture channel0 captures rising edge, and CRIE0 is 1, then CRIS0 is set to 1; when the timer logic module of the capture channel0 captures falling edge, and CFIE0 is 1, then CFIS0 is set to 1.

7.10.4 Programming Guidelines

The following working mode takes PWM01 as an example, other PWM pairs and PWM01 are consistent.

7.10.4.1 Configuring Clock

1. PWM gating: When using PWM, write 1 to **PCGR**[PWMx_CLK_GATING].
2. PWM clock source select: Set **PCCR01**[PWM01_CLK_SRC] to select OSC24M or APB0 clock.
3. PWM clock divider: Set **PCCR01**[PWM01_CLK_DIV_M] to select different frequency division coefficient (1/2/4/8/16/32/64/128/256).
4. PWM clock bypass: Set **PCGR**[PWM_CLK_SRC_BYPASS_TO_PWM] to 1, output the PWM clock after the secondary frequency division to the corresponding PWM output pin.
5. PWM internal clock configuration: Set **PCR**[PWM_PRESCAL_K] to select any frequency division coefficient from 1 to 256.

NOTE

For the channel of complementary output and group mode, firstly, set the same clock configurations (clock source selects APB0, clock division configures the same division factor); secondly, open clock gating at the same time; thirdly, configure PWM parameters; finally, enable PWM output at the same time to ensure each channel sync.

We suggest that the two channels of the same PWM pair cannot subject to two groups because of they have the same first level clock division and gating. If must allocate based on this way, the first level of clock division of the channel used by all groups needs to set to the same coefficient and open gating at the same time. And the total module needs to be reset when the group mode regroups.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.10.4.2 Configuring PWM

1. PWM mode: Set PCR[PWM_MODE] to select cycle mode or pulse mode, if pulse mode, PCR[PWM_PUL_NUM] needs to be configured.
2. PWM active level: Set PCR[PWM_ACT_STA] to select a low level or high level.
3. PWM duty-cycle: Configure PPR[PWM_ENTIRE_CYCLE] and PPR[PWM_ACT_CYCLE] after clock gating is opened.
4. Enable PWM: Configure PER to select the corresponding PWM enable bit; when selecting pulse mode, PCR[PWM_PUL_START] needs to be enabled.

7.10.4.3 Configuring Deadzone

1. Set initial value: set [PDZINTV01].
2. Enable Deadzone: set [PWM01_DZ_CN].

7.10.4.4 Configuring Capture Input

1. Enable capture: Configure CER to enable the corresponding channel.
2. Capture mode: Configure CCR[CRLF] and CCR[CFLF] to select rising edge capture or falling edge capture, configure CCR[CAPINV] to select whether the input signal does reverse processing.

7.10.5 Register List

Module Name	Base Address
PWM (in CPUX domain)	0x02000C00
R_PWM (in CPUS domain)	0x07020C00

Register Name	Offset	Description
PIER	0x0000	PWM IRQ Enable Register
PISR	0x0004	PWM IRQ Status Register
CIER	0x0010	Capture IRQ Enable Register
CISR	0x0014	Capture IRQ Status Register
PCCR01	0x0020	PWM01 Clock Configuration Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
PCCR23	0x0024	PWM23 Clock Configuration Register
PCCR45	0x0028	PWM45 Clock Configuration Register
PCCR67	0x002C	PWM67 Clock Configuration Register
PCCR89	0x0030	PWM89 Clock Configuration Register
PCGR	0x0040	PWM Clock Gating Register
PDZCR01	0x0060	PWM01 Dead Zone Control Register
PDZCR23	0x0064	PWM23 Dead Zone Control Register
PDZCR45	0x0068	PWM45 Dead Zone Control Register
PDZCR67	0x006C	PWM67 Dead Zone Control Register
PER	0x0080	PWM Enable Register
PGR0	0x0090	PWM Group0 Register
PGR1	0x0094	PWM Group1 Register
PGR2	0x0098	PWM Group2 Register
CER	0x00C0	Capture Enable Register
PCR	0x0100+0x0000+N*0x0020 (N= 0~8)	PWM Control Register
PPR	0x0100+0x0004+N*0x0020 (N= 0~8)	PWM Period Register
PCNTR	0x0100+0x0008+N*0x0020 (N= 0~8)	PWM Count Register
PPCNTR	0x0100+0x000C+N*0x0020 (N= 0~8)	PWM Pulse Count Register
CCR	0x0100+0x0010+N*0x0020 (N= 0~8)	Capture Control Register
CRLR	0x0100+0x0014+N*0x0020 (N= 0~8)	Capture Rise Lock Register
CFLR	0x0100+0x0018+N*0x0020 (N= 0~8)	Capture Fall Lock Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.10.6 Register Description

7.10.6.1 0x0000 PWM IRQ Enable Register (Default Value: 0x0000_0000)

Offset:0x0000			Register Name: PIER
Bit	**Read/Write**	**Default/Hex**	**Description**
31:19	/	/	/
18	R/W	0x0	PGIE2 PWM Group 2 Interrupt Enable 0: Disable 1: Enable
17	R/W	0x0	PGIE1 PWM Group 1 Interrupt Enable 0: Disable 1: Enable
16	R/W	0x0	PGIE0 PWM Group 0 Interrupt Enable 0: Disable 1: Enable
15:9	/	/	/
8	R/W	0x0	PCIE8 PWM Channel 8 Interrupt Enable 0: PWM Channel 8 Interrupt Disable 1: PWM Channel 8 Interrupt Enable
7	R/W	0x0	PCIE7 PWM Channel 7 Interrupt Enable 0: PWM Channel 7 Interrupt Disable 1: PWM Channel 7 Interrupt Enable
6	R/W	0x0	PCIE6 PWM Channel 6 Interrupt Enable 0: PWM Channel 6 Interrupt Disable 1: PWM Channel 6 Interrupt Enable
5	R/W	0x0	PCIE5 PWM Channel 5 Interrupt Enable 0: PWM Channel 5 Interrupt Disable 1: PWM Channel 5 Interrupt Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0000			Register Name: PIER
Bit	Read/Write	Default/Hex	Description
4	R/W	0x0	PCIE4 PWM Channel 4 Interrupt Enable 0: PWM Channel 4 Interrupt Disable 1: PWM Channel 4 Interrupt Enable
3	R/W	0x0	PCIE3 PWM Channel 3 Interrupt Enable 0: PWM Channel 3 Interrupt Disable 1: PWM Channel 3 Interrupt Enable
2	R/W	0x0	PCIE2 PWM Channel 2 Interrupt Enable 0: PWM Channel 2 Interrupt Disable 1: PWM Channel 2 Interrupt Enable
1	R/W	0x0	PCIE1 PWM Channel 1 Interrupt Enable 0: PWM Channel 1 Interrupt Disable 1: PWM Channel 1 Interrupt Enable
0	R/W	0x0	PCIE0 PWM Channel 0 Interrupt Enable 0: PWM Channel 0 Interrupt Disable 1: PWM Channel 0 Interrupt Enable

7.10.6.2 0x0004 PWM IRQ Status Register (Default Value: 0x0000_0000)

Offset:0x0004			Register Name: PISR
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
18	R/W1C	0x0	PGIS2 PWM Group 2 Interrupt Status
17	R/W1C	0x0	PGIS1 PWM Group 1 Interrupt Status
16	R/W1C	0x0	PGIS0 PWM Group 0 Interrupt Status
15:9	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0004			Register Name: PISR
Bit	**Read/Write**	**Default/Hex**	**Description**
8	R/W1C	0x0	PIS8 PWM Channel 8 Interrupt Status When the PWM channel 8 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit. Reads 0: PWM channel 8 interrupt is not pending. Reads 1: PWM channel 8 interrupt is pending. Writes 0: No effect. Writes 1: Clear PWM channel 8 interrupt status.
7	R/W1C	0x0	PIS7 PWM Channel 7 Interrupt Status When the PWM channel 7 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit. Reads 0: PWM channel 7 interrupt is not pending. Reads 1: PWM channel 7 interrupt is pending. Writes 0: No effect. Writes 1: Clear PWM channel 7 interrupt status.
6	R/W1C	0x0	PIS6 PWM Channel 6 Interrupt Status When the PWM channel 6 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit. Reads 0: PWM channel 6 interrupt is not pending. Reads 1: PWM channel 6 interrupt is pending. Writes 0: No effect. Writes 1: Clear PWM channel 6 interrupt status.
5	R/W1C	0x0	PIS5 PWM Channel 5 Interrupt Status When the PWM channel 5 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit. Reads 0: PWM channel 5 interrupt is not pending. Reads 1: PWM channel 5 interrupt is pending. Writes 0: No effect. Writes 1: Clear PWM channel 5 interrupt status.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0004			Register Name: PISR
Bit	**Read/Write**	**Default/Hex**	**Description**
4	R/W1C	0x0	PIS4 PWM Channel 4 Interrupt Status When the PWM channel 4 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit. Reads 0: PWM channel 4 interrupt is not pending. Reads 1: PWM channel 4 interrupt is pending. Writes 0: No effect. Writes 1: Clear PWM channel 4 interrupt status.
3	R/W1C	0x0	PIS3 PWM Channel 3 Interrupt Status When the PWM channel 3 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit. Reads 0: PWM channel 3 interrupt is not pending. Reads 1: PWM channel 3 interrupt is pending. Writes 0: No effect. Writes 1: Clear PWM channel 3 interrupt status.
2	R/W1C	0x0	PIS2 PWM Channel 2 Interrupt Status When the PWM channel 2 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit. Reads 0: PWM channel 2 interrupt is not pending. Reads 1: PWM channel 2 interrupt is pending. Writes 0: No effect. Writes 1: Clear PWM channel 2 interrupt status.
1	R/W1C	0x0	PIS1 PWM Channel 1 Interrupt Status When the PWM channel 1 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit. Reads 0: PWM channel 1 interrupt is not pending. Reads 1: PWM channel 1 interrupt is pending. Writes 0: No effect. Writes 1: Clear PWM channel 1 interrupt status.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0004			Register Name: PISR
Bit	**Read/Write**	**Default/Hex**	**Description**
0	R/W1C	0x0	PIS0 PWM Channel 0 Interrupt Status When the PWM channel 0 counter reaches the Entire Cycle Value, this bit is set 1 by hardware. Writing 1 to clear this bit. Reads 0: PWM channel 0 interrupt is not pending. Reads 1: PWM channel 0 interrupt is pending. Writes 0: No effect. Writes 1: Clear PWM channel 0 interrupt status.

7.10.6.3 0x0010 PWM Capture IRQ Enable Register (Default Value: 0x0000_0000)

Offset:0x0010			Register Name: CIER
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17	R/W	0x0	CFIE8 If this enable bit is set 1, when the capture channel 8 captures falling edge, it generates a capture channel 8 pending. 0: Capture channel 8 fall lock Interrupt disable 1: Capture channel 8 fall lock Interrupt enable
16	R/W	0x0	CRIE8 If this enable bit is set 1, when the capture channel 8 captures rising edge, it generates a capture channel 8 pending. 0: Capture channel 8 rise lock Interrupt disable 1: Capture channel 8 rise lock Interrupt enable
15	R/W	0x0	CFIE7 If the enable bit is set 1, when the capture channel 7 captures falling edge, it generates a capture channel 7 pending. 0: Capture channel 7 fall lock interrupt disable 1: Capture channel 7 fall lock interrupt enable
14	R/W	0x0	CRIE7 If the enable bit is set 1, when the capture channel 7 captures rising edge, it generates a capture channel 7 pending. 0: Capture channel 7 rise lock interrupt disable 1: Capture channel 7 rise lock interrupt enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0010			Register Name: CIER
Bit	**Read/Write**	**Default/Hex**	**Description**
13	R/W	0x0	CFIE6 If the enable bit is set 1, when the capture channel 6 captures falling edge, it generates a capture channel 6 pending. 0: Capture channel 6 fall lock interrupt disable 1: Capture channel 6 fall lock interrupt enable
12	R/W	0x0	CRIE6 If the enable bit is set 1, when the capture channel 6 captures rising edge, it generates a capture channel 6 pending. 0: Capture channel 6 rise lock interrupt disable 1: Capture channel 6 rise lock interrupt enable
11	R/W	0x0	CFIE5 If the enable bit is set 1, when the capture channel 5 captures falling edge, it generates a capture channel 5 pending. 0: Capture channel 5 fall lock interrupt disable 1: Capture channel 5 fall lock interrupt enable
10	R/W	0x0	CRIE5 If the enable bit is set 1, when the capture channel 5 captures rising edge, it generates a capture channel 5 pending. 0: Capture channel 5 rise lock interrupt disable 1: Capture channel 5 rise lock interrupt enable
9	R/W	0x0	CFIE4 If the enable bit is set 1, when the capture channel 4 captures falling edge, it generates a capture channel 4 pending. 0: Capture channel 4 fall lock interrupt disable 1: Capture channel 4 fall lock interrupt enable
8	R/W	0x0	CRIE4 If the enable bit is set 1, when the capture channel 4 captures rising edge, it generates a capture channel 4 pending. 0: Capture channel 4 rise lock interrupt disable 1: Capture channel 4 rise lock interrupt enable
7	R/W	0x0	CFIE3 If the enable bit is set 1, when the capture channel 3 captures falling edge, it generates a capture channel 3 pending. 0: Capture channel 3 fall lock interrupt disable 1: Capture channel 3 fall lock interrupt enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0010			Register Name: CIER
Bit	**Read/Write**	**Default/Hex**	**Description**
6	R/W	0x0	CRIE3 If the enable bit is set 1, when the capture channel 3 captures rising edge, it generates a capture channel 3 pending. 0: Capture channel 3 rise lock interrupt disable 1: Capture channel 3 rise lock interrupt enable
5	R/W	0x0	CFIE2 If the enable bit is set 1, when the capture channel 2 captures falling edge, it generates a capture channel 2 pending. 0: Capture channel 2 fall lock interrupt disable 1: Capture channel 2 fall lock interrupt enable
4	R/W	0x0	CRIE2 If the enable bit is set 1, when the capture channel 2 captures rising edge, it generates a capture channel 2 pending. 0: Capture channel 2 rise lock interrupt disable 1: Capture channel 2 rise lock interrupt enable
3	R/W	0x0	CFIE1 If the enable bit is set 1, when the capture channel 1 captures falling edge, it generates a capture channel 1 pending. 0: Capture channel 1 fall lock interrupt disable 1: Capture channel 1 fall lock interrupt enable
2	R/W	0x0	CRIE1 If the enable bit is set 1, when the capture channel 1 captures rising edge, it generates a capture channel 1 pending. 0: Capture channel 1 rise lock interrupt disable 1: Capture channel 1 rise lock interrupt enable
1	R/W	0x0	CFIE0 If the enable bit is set 1, when the capture channel 0 captures falling edge, it generates a capture channel 0 pending. 0: Capture channel 0 fall lock interrupt disable 1: Capture channel 0 fall lock interrupt enable
0	R/W	0x0	CRIE0 If the enable bit is set 1, when the capture channel 0 captures rising edge, it generates a capture channel 0 pending. 0: Capture channel 0 rise lock interrupt disable 1: Capture channel 0 rise lock interrupt enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.10.6.4 0x0014 PWM Capture IRQ Status Register (Default Value: 0x0000_0000)

Offset:0x0014			Register Name: CISR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17	R/W1C	0x0	CFIS8 Status of the capture channel 8 falling lock interrupt When the capture channel 8 captures falling edge, if the fall lock interrupt (CFIE8) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 8 interrupt is not pending. Reads 1: The capture channel 8 interrupt is pending. Writes 0: No effect. Writes 1: Clear the status of the capture channel 8 interrupt.
16	R/W1C	0x0	CRIS8 Status of the capturing channel 8 rising lock interrupt When the capture channel 8 captures rising edge, if the rise lock interrupt (CRIE8) is enabled, this bit is set to 1 by hardware. Write 1 to clear this bit. Reads 0: The capture channel 8 interrupt is not pending. Reads 1: The capture channel 8 interrupt is pending. Writes 0: No effect. Writes 1: Clear the status of the capture channel 8 interrupt.
15	R/W1C	0x0	CFIS7 Status of the capture channel 7 falling lock interrupt When the capture channel 7 captures falling edge, if the fall lock interrupt (CFIE7) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 7 interrupt is not pending. Reads 1: The capture channel 7 interrupt is pending. Writes 0: No effect. Writes 1: Clear the status of the capture channel 7 interrupt.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0014			Register Name: CISR
Bit	**Read/Write**	**Default/Hex**	**Description**
14	R/W1C	0x0	CRIS7 Status of the capture channel 7 rising lock interrupt When the capture channel 7 captures rising edge, if the rise lock interrupt (CRIE7) is enabled, this bit is set to 1 by hardware. Write 1 to clear this bit. Reads 0: The capture channel 7 interrupt is not pending. Reads 1: The capture channel 7 interrupt is pending. Writes 0: No effect. Writes 1: Clear the status of the capture channel 7 interrupt.
13	R/W1C	0x0	CFIS6 Status of the capture channel 6 falling lock interrupt When the capture channel 6 captures falling edge, if the fall lock interrupt (CFIE6) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 6 interrupt is not pending. Reads 1: The capture channel 6 interrupt is pending. Writes 0: No effect. Writes 1: Clear the status of the capture channel 6 interrupt.
12	R/W1C	0x0	CRIS6 Status of the capture channel 6 rising lock interrupt. When the capture channel 6 captures rising edge, if the rise lock interrupt (CRIE6) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 6 interrupt is not pending. Reads 1: The capture channel 6 interrupt is pending. Writes 0: No effect. Writes 1: Clear the status of the capture channel 6 interrupt.
11	R/W1C	0x0	CFIS5 Status of the capture channel 5 falling lock interrupt When the capturing channel 5 captures falling edge, if the fall lock interrupt (CFIE5) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 5 interrupt is not pending. Reads 1: The capture channel 5 interrupt is pending. Writes 0: No effect. Reads 1: Clear the status of the capture channel 5 interrupt.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0014			Register Name: CISR
Bit	**Read/Write**	**Default/Hex**	**Description**
10	R/W1C	0x0	CRIS5 Status of the capture channel 5 rising lock interrupt When the capture channel 5 captures rising edge, if the rise lock interrupt (CRIE5) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 5 interrupt is not pending. Reads 1: The capture channel 5 interrupt is pending. Writes 0: No effect. Writes 1: Clear the status of the capture channel 5 interrupt.
9	R/W1C	0x0	CFIS4 Status of the capture channel 4 falling lock interrupt When the capture channel 4 captures falling edge, if the fall lock interrupt (CFIE4) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 4 interrupt is not pending. Reads 1: The capture channel 4 interrupt is pending. Writes 0: No effect. Writes 1: Clear the status of the capture channel 4 interrupt.
8	R/W1C	0x0	CRIS4 Status of the capture channel 4 rising lock interrupt. When the capture channel 4 captures rising edge, if the rise lock interrupt (CRIE4) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 4 interrupt is not pending. Reads 1: The capture channel 4 interrupt is pending. Writes 0: No effect. Writes 1: Clear the status of the capture channel 4 interrupt status.
7	R/W1C	0x0	CFIS3 Status of the capture channel 3 falling lock interrupt. When the capture channel 3 captures falling edge, if the fall lock interrupt (CFIE3) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 3 interrupt is not pending. Reads 1: The capture channel 3 interrupt is pending. Writes 0: no effect. Writes 1: Clear the status of the capture channel 3 interrupt.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0014			Register Name: CISR
Bit	**Read/Write**	**Default/Hex**	**Description**
6	R/W1C	0x0	CRIS3 Status of the capture channel 3 rising lock interrupt When the capture channel 3 captures rising edge, if the rise lock interrupt (CRIE3) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 3 interrupt is not pending. Reads 1: The capture channel 3 interrupt is pending. Writes 0: no effect. Writes 1: Clear the status of the capture channel 3 interrupt.
5	R/W1C	0x0	CFIS2 Status of the capture channel 2 falling lock interrupt When the capture channel 2 captures falling edge, if the fall lock interrupt (CFIE2) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 2 interrupt is not pending. Reads 1: The capture channel 2 interrupt is pending. Writes 0: no effect. Writes 1: Clear the status of the capture channel 2 interrupt.
4	R/W1C	0x0	CRIS2 Status of the capture channel 2 rising lock interrupt. When the capture channel 2 captures rising edge, if the rise lock interrupt (CRIE2) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 2 interrupt is not pending. Reads 1: The capture channel 2 interrupt is pending. Writes 0: no effect. Writes 1: Clear the status of the capture channel 2 interrupt.
3	R/W1C	0x0	CFIS1 Status of the capture channel 1 falling lock interrupt When the capture channel 1 captures falling edge, if the fall lock interrupt (CFIE1) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 1 interrupt is not pending. Reads 1: The capture channel 1 interrupt is pending. Writes 0: no effect. Writes 1: Clear the status of the capture channel 1 interrupt.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0014			Register Name: CISR
Bit	**Read/Write**	**Default/Hex**	**Description**
2	R/W1C	0x0	CRIS1 Status of the capture channel 1 rising lock interrupt. When the capture channel 1 captures rising edge, if the rise lock interrupt (CRIE1) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 1 interrupt is not pending. Reads 1: The capture channel 1 interrupt is pending. Writes 0: no effect. Writes 1: Clear the status of the capture channel 1 interrupt.
1	R/W1C	0x0	CFIS0 Status of the capture channel 0 falling lock interrupt When the capture channel 0 captures falling edge, if the fall lock interrupt (CFIE0) is enabled, this bit is set to 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 0 interrupt is not pending. Reads 1: The capture channel 0 interrupt is pending. Writes 0: no effect. Writes 1: Clear the status of the capture channel 0 interrupt.
0	R/W1C	0x0	CRIS0 Status of the capture channel 0 rising lock interrupt When the capture channel 0 captures rising edge, if the rise lock interrupt (CRIE0) is enabled, this bit is set 1 by hardware. Writing 1 to clear this bit. Reads 0: The capture channel 0 interrupt is not pending. Reads 1: The capture channel 0 interrupt is pending. Writes 0: no effect. Writes 1: Clear the status of the capture channel 0 interrupt.

7.10.6.5 0x0020 PWM01 Clock Configuration Register (Default Value: 0x0000_0000)

Offset:0x0020			Register Name: PCCR01
Bit	**Read/Write**	**Default/Hex**	**Description**
31:9	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0020			Register Name: PCCR01
Bit	Read/Write	Default/Hex	Description
8:7	R/W	0x0	PWM01_CLK_SRC Select PWM01 Clock Source 00: OSC24M 01: APB0 (It is APBS0 for S-PWM01) Others: Reserved
6:4	/	/	/
3:0	R/W	0x0	PWM01_CLK_DIV_M PWM01 Clock Divide M 0000: /1 0001: /2 0010: /4 0011: /8 0100: /16 0101: /32 0110: /64 0111: /128 1000: /256 Others: Reserved

7.10.6.6 0x0024 PWM23 Clock Configuration Register (Default Value: 0x0000_0000)

Offset:0x0024			Register Name: PCCR23
Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8:7	R/W	0x0	PWM23_CLK_SRC_SEL Select PWM23 Clock Source 00: OSC24M 01: APB0 (It is APBS0 for S-PWM23) Others: Reserved
6:4	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0024			Register Name: PCCR23
Bit	**Read/Write**	**Default/Hex**	**Description**
3:0	R/W	0x0	PWM23_CLK_DIV_M PWM23 Clock Divide M 0000: /1 0001: /2 0010: /4 0011: /8 0100: /16 0101: /32 0110: /64 0111: /128 1000: /256 Others: Reserved

7.10.6.7 0x0028 PWM45 Clock Configuration Register (Default Value: 0x0000_0000)

Offset:0x0028			Register Name: PCCR45
Bit	**Read/Write**	**Default/Hex**	**Description**
31:9	/	/	/
8:7	R/W	0x0	PWM45_CLK_SRC_SEL Select PWM45 Clock Source 00: OSC24M 01: APB0 (It is APBS0 for S-PWM45) Others: Reserved
6:4	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0028			Register Name: PCCR45
Bit	**Read/Write**	**Default/Hex**	**Description**
3:0	R/W	0x0	PWM45_CLK_DIV_M PWM45 Clock Divide M 0000: /1 0001: /2 0010: /4 0011: /8 0100: /16 0101: /32 0110: /64 0111: /128 1000: /256 Others: Reserved

7.10.6.8 0x002C PWM67 Clock Configuration Register (Default Value: 0x0000_0000)

Offset:0x002C			Register Name: PCCR67
Bit	**Read/Write**	**Default/Hex**	**Description**
31:9	/	/	/
8:7	R/W	0x0	PWM67_CLK_SRC_SEL Select PWM67 Clock Source 00: OSC24M 01: APB0 Others: Reserved
6:4	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x002C			Register Name: PCCR67
Bit	**Read/Write**	**Default/Hex**	**Description**
3:0	R/W	0x0	PWM67_CLK_DIV_M PWM67 Clock Divide M 0000: /1 0001: /2 0010: /4 0011: /8 0100: /16 0101: /32 0110: /64 0111: /128 1000: /256 Others: Reserved

7.10.6.9 0x0030 PWM8 Clock Configuration Register (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: PCCR8
Bit	**Read/Write**	**Default/Hex**	**Description**
31:9	/	/	/
8:7	R/W	0x00	PWM8_CLK_SRC_SEL Select PWM89 clock source 00: OSC24M 01: APB0 Others: /
6:4	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0030			Register Name: PCCR8
Bit	Read/Write	Default/Hex	Description
3:0	R/W	0x0000	PWM8_CLK_DIV_M PWM89 clock divide M 0000: /1 0001: /2 0010: /4 0011: /8 0100: /16 0101: /32 0110: /64 0111: /128 1000: /256 Others: /

7.10.6.10 0x0040 PWM Clock Gating Register (Default Value: 0x0000_0000)

Offset: 0x0040			Register Name: PCGR
Bit	Read/Write	Default/Hex	Description
31:25	/	/	/
24	R/W	0x0	PWM8_CLK_BYPASS Bypass clock source (after pre-scale) to PWM8 output 0: not bypass 1: bypass
23	R/W	0x0	PWM7_CLK_BYPASS Bypass clock source (after pre-scale) to PWM7 output 0: not bypass 1: bypass
22	R/W	0x0	PWM6_CLK_BYPASS Bypass clock source (after pre-scale) to PWM6 output 0: not bypass 1: bypass
21	R/W	0x0	PWM5_CLK_BYPASS Bypass clock source (after pre-scale) to PWM5 output 0: not bypass 1: bypass

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0040			Register Name: PCGR
Bit	**Read/Write**	**Default/Hex**	**Description**
20	R/W	0x0	PWM4_CLK_BYPASS Bypass clock source (after pre-scale) to PWM4 output 0: not bypass 1: bypass
19	R/W	0x0	PWM3_CLK_BYPASS Bypass clock source (after pre-scale) to PWM3 output 0: not bypass 1: bypass
18	R/W	0x0	PWM2_CLK_BYPASS Bypass clock source (after pre-scale) to PWM2 output 0: not bypass 1: bypass
17	R/W	0x0	PWM1_CLK_BYPASS Bypass clock source (after pre-scale) to PWM1 output 0: not bypass 1: bypass
16	R/W	0x0	PWM0_CLK_BYPASS Bypass clock source (after pre-scale) to PWM0 output 0: not bypass 1: bypass
15	R/W	0x0	PWMF_CLK_GATING Gating clock for PWMF 0: Mask 1: Pass
14	R/W	0x0	PWME_CLK_GATING Gating clock for PWME 0: Mask 1: Pass
13	R/W	0x0	PWMD_CLK_GATING Gating clock for PWMD 0: Mask 1: Pass
12	R/W	0x0	PWMC_CLK_GATING Gating clock for PWMC 0: Mask 1: Pass

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0040			Register Name: PCGR
Bit	**Read/Write**	**Default/Hex**	**Description**
11	R/W	0x0	PWMB_CLK_GATING Gating clock for PWMB 0: Mask 1: Pass
10	R/W	0x0	PWMA_CLK_GATING Gating clock for PWMA 0: Mask 1: Pass
9	R/W	0x0	PWM9_CLK_GATING Gating clock for PWM9 0: Mask 1: Pass
8	R/W	0x0	PWM8_CLK_GATING Gating clock for PWM8 0: Mask 1: Pass
7	R/W	0x0	PWM7_CLK_GATING Gating clock for PWM7 0: Mask 1: Pass
6	R/W	0x0	PWM6_CLK_GATING Gating clock for PWM6 0: Mask 1: Pass
5	R/W	0x0	PWM5_CLK_GATING Gating clock for PWM5 0: Mask 1: Pass
4	R/W	0x0	PWM4_CLK_GATING Gating clock for PWM4 0: Mask 1: Pass
3	R/W	0x0	PWM3_CLK_GATING Gating clock for PWM3 0: Mask 1: Pass

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0040			Register Name: PCGR
Bit	Read/Write	Default/Hex	Description
2	R/W	0x0	PWM2_CLK_GATING Gating clock for PWM2 0: Mask 1: Pass
1	R/W	0x0	PWM1_CLK_GATING Gating clock for PWM1 0: Mask 1: Pass
0	R/W	0x0	PWM0_CLK_GATING Gating clock for PWM0 0: Mask 1: Pass

7.10.6.11 0x0060 PWM01 Dead Zone Control Register (Default Value: 0x0000_0000)

Offset:0x0060			Register Name: ***PDZCR01***
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:8	R/W	0x0	PWM01_DZ_INTV PWM01 Dead Zone Interval Value
7:1	/	/	/
0	R/W	0x0	PWM01_DZ_EN PWM01 Dead Zone Enable 0: Dead Zone disabled 1: Dead Zone enabled

7.10.6.12 0x0064 PWM23 Dead Zone Control Register (Default Value: 0x0000_0000)

Offset:0x0064			Register Name: ***PDZCR23***
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:8	R/W	0x0	PWM23_DZ_INTV PWM23 Dead Zone Interval Value

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0064			Register Name: ***PDZCR23***
Bit	**Read/Write**	**Default/Hex**	**Description**
7:1	/	/	/
0	R/W	0x0	PWM23_DZ_EN PWM23 Dead Zone Enable 0: Dead Zone disabled 1: Dead Zone enabled

7.10.6.13 0x0068 PWM45 Dead Zone Control Register (Default Value: 0x0000_0000)

Offset:0x0068			Register Name: ***PDZCR45***
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:8	R/W	0x0	PWM45_DZ_INTV PWM45 Dead Zone Interval Value
7:1	/	/	/
0	R/W	0x0	PWM45_DZ_EN PWM45 Dead Zone Enable 0: Dead Zone disabled 1: Dead Zone enabled

7.10.6.14 0x006C PWM67 Dead Zone Control Register (Default Value: 0x0000_0000)

Offset:0x006C			Register Name: ***PDZCR67***
Bit	**Read/Write**	**Default/Hex**	**Description**
15:8	R/W	0x0	PWM67_DZ_INTV PWM67 Dead Zone Interval Value
7:1	/	/	/
0	R/W	0x0	PWM67_DZ_EN PWM67 Dead Zone Enable 0: Dead Zone disabled 1: Dead Zone enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.10.6.15 0x0080 PWM Enable Register (Default Value: 0x0000_0000)

Offset:0x0080			Register Name: PER
Bit	**Read/Write**	**Default/Hex**	**Description**
31:9	/	/	/
8	R/W	0x0	PWM8_EN When PWM is enabled, the 16-bit up-counter starts working and PWM channel8 is permitted to output PWM waveform. 0: PWM disabled 1: PWM enabled
7	R/W	0x0	PWM7_EN When PWM is enabled, the 16-bit up-counter starts working and PWM channel7 is permitted to output PWM waveform. 0: PWM disabled 1: PWM enabled
6	R/W	0x0	PWM6_EN When PWM is enabled, the 16-bit up-counter starts working and PWM channel6 is permitted to output PWM waveform. 0: PWM disabled 1: PWM enabled
5	R/W	0x0	PWM5_EN When PWM is enabled, the 16-bit up-counter starts working and PWM channel5 is permitted to output PWM waveform. 0: PWM disabled 1: PWM enabled
4	R/W	0x0	PWM4_EN When PWM is enabled, the 16-bit up-counter starts working and PWM channel4 is permitted to output PWM waveform. 0: PWM disabled 1: PWM enabled
3	R/W	0x0	PWM3_EN When PWM is enabled, the 16-bit up-counter starts working and PWM channel3 is permitted to output PWM waveform. 0: PWM disabled 1: PWM enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0080			Register Name: PER
Bit	**Read/Write**	**Default/Hex**	**Description**
2	R/W	0x0	PWM2_EN When PWM is enabled, the 16-bit up-counter starts working and PWM channel2 is permitted to output PWM waveform. 0: PWM disabled 1: PWM enabled
1	R/W	0x0	PWM1_EN When PWM is enabled, the 16-bit up-counter starts working and PWM channel1 is permitted to output PWM waveform. 0: PWM disabled 1: PWM enabled
0	R/W	0x0	PWM0_EN When PWM is enabled, the 16-bit up-counter starts working and PWM channel0 is permitted to output PWM waveform. 0: PWM disabled 1: PWM enabled

7.10.6.16 0x0090 PWM Group0 Register (Default Value: 0x0000_0000)

Offset: 0x0090			Register Name: PGR0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17	R/WAC	0x0	PWMG0_START PWM channels selected in PWMG0_CS start to output PWM waveform at the same time.
16	R/W	0x0	PWMG0_EN PWM Group0 Enable.
15:0	R/W	0x0	PWMG0_CS If bit[i] is set, the PWM i is selected as one channel of PWM Group0.

7.10.6.17 0x0094 PWM Group1 Register (Default Value: 0x0000_0000)

Offset: 0x0094			Register Name: PGR1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0094			Register Name: PGR1
Bit	**Read/Write**	**Default/Hex**	**Description**
17	R/WAC	0x0	PWMG1_START PWM channels selected in PWMG1_CS start to output PWM waveform at the same time.
16	R/W	0x0	PWMG1_EN PWM Group1 Enable.
15:0	R/W	0x0	PWMG1_CS If bit[i] is set, the PWM i is selected as one channel of PWM Group1.

7.10.6.18 0x0098 PWM Group2 Register (Default Value: 0x0000_0000)

Offset: 0x0098			Register Name: PGR2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17	R/WAC	0x0	PWMG2_START PWM channels selected in PWMG2_CS start to output PWM waveform at the same time.
16	R/W	0x0	PWMG2_EN PWM Group2 Enable.
15:0	R/W	0x0	PWMG2_CS If bit[i] is set, the PWM i is selected as one channel of PWM Group2.

7.10.6.19 0x00C0 Capture Enable Register (Default Value: 0x0000_0000)

Offset: 0x00C0			Register Name: CER
Bit	**Read/Write**	**Default/Hex**	**Description**
31:9	/	/	/
8	R/W	0x0	CAP8_EN When enabling the capture function, the 16-bit up-counter starts working, and the capture channel8 is permitted to capture external falling edge or rising edge. 0: Capture disabled 1: Capture enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x00C0			Register Name: CER
Bit	**Read/Write**	**Default/Hex**	**Description**
7	R/W	0x0	CAP7_EN When enabling the capture function, the 16-bit up-counter starts working, and the capture channel7 is permitted to capture external falling edge or rising edge. 0: Capture disabled 1: Capture enabled
6	R/W	0x0	CAP6_EN When enabling the capture function, the 16-bit up-counter starts working, and the capture channel6 is permitted to capture external falling edge or rising edge. 0: Capture disabled 1: Capture enabled
5	R/W	0x0	CAP5_EN When enabling the capture function, the 16-bit up-counter starts working, and the capture channel5 is permitted to capture external falling edge or rising edge. 0: Capture disabled 1: Capture enabled
4	R/W	0x0	CAP4_EN When enabling the capture function, the 16-bit up-counter starts working, and the capture channel4 is permitted to capture external falling edge or rising edge. 0: Capture disabled 1: Capture enabled
3	R/W	0x0	CAP3_EN When enabling the capture function, the 16-bit up-counter starts working, and the capture channel3 is permitted to capture external falling edge or rising edge. 0: Capture disabled 1: Capture enabled
2	R/W	0x0	CAP2_EN When enabling the capture function, the 16-bit up-counter starts working, and the capture channel2 is permitted to capture external falling edge or rising edge. 0: Capture disabled 1: Capture enabled

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x00C0			Register Name: CER
Bit	**Read/Write**	**Default/Hex**	**Description**
1	R/W	0x0	CAP1_EN When enabling the capture function, the 16-bit up-counter starts working, and the capture channel1 is permitted to capture external falling edge or rising edge. 0: Capture disabled 1: Capture enabled
0	R/W	0x0	CAP0_EN When enabling the capture function, the 16-bit up-counter starts working, and the capture channel is permitted to capture external falling edge or rising edge. 0: Capture disabled 1: Capture enabled

7.10.6.20 0x0100 + N*0x20 PWM Control Register (Default Value: 0x0000_0000)

Offset:0x0100+0x0+N*0x20 (N=0~8)			Register Name: PCR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	R/W	0x0	PWM_PUL_NUM In pulse mode, PWM outputs pulse for PWM_CYCLE_NUM+1 times and then stops.
15:12	/	/	/
11	R	0x0	PWM_PERIOD_RDY PWM Period Register Ready 0: PWM period register is ready to write 1: PWM period register is busy
10	R/W1S	0x0	PWM_PUL_START PWM Pulse Output Start 0: No effect 1: Output pulse for PWM_CYCLE_NUM+1. After finishing configuration for the output pulse, set this bit once , then PWM would output waveform. After the waveform is finished, the bit will be cleared automatically.
9	R/W	0x0	PWM_MODE PWM Output Mode Select 0: Cycle mode 1: Pulse mode

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0100+0x0+N*0x20 (N=0~8)			Register Name: PCR
Bit	**Read/Write**	**Default/Hex**	**Description**
8	R/W	0x0	PWM_ACT_STA PWM Active State 0: Low Level 1: High Level
7:0	R/W	0x0	PWM_PRESCAL_K PWM pre-scale K, actual pre-scale is (K+1). K = 0, actual pre-scale: 1 K = 1, actual pre-scale: 2 K = 2, actual pre-scale: 3 K = 3, actual pre-scale: 4 ... K = 255, actual pre-scale: 256

7.10.6.21 0x0104 + N*0x20 PWM Period Register (Default Value: 0x0000_0000)

Offset:0x0100+0x04+N*0x20 (N=0~8)			Register Name: PPR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	R/W	0x0	PWM_ENTIRE_CYCLE Number of the entire cycles in the PWM clock. 0: 1 cycle 1: 2 cycles … N: N+1 cycles If the register needs to be modified dynamically, the PCLK should be faster than the PWM CLK.
15:0	R/W	0x0	PWM_ACT_CYCLE Number of the active cycles in the PWM clock. 0: 0 cycle 1: 1 cycle … N: N cycles

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.10.6.22 0x0108 + N*0x20 PWM Counter Register (Default Value: 0x0000_0000)

Offset:0x0100+0x08+N*0x20 (N=0~8)			Register Name: PCNTR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	R/W	0x0	PWM_COUNTER_START PWM counter value is set for phase control.
15:0	R	0x0	PWM_COUNTER_STATUS On PWM output or capture input, reading this register could get the current value of the PWM 16-bit up-counter.

7.10.6.23 0x010C + N*0x20 PWM Pulse Counter Register (Default Value: 0x0000_0000)

Offset: 0x0100+0x0C+N*0x20 (N=0~8)			Register Name: PPCNTR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R	0x0	PWM_PUL_COUNTER_STATUS On PWM output, reading this register could get the current value of the PWM pulse counter.

7.10.6.24 0x0110 + N*0x20 PWM Capture Control Register (Default Value: 0x0000_0000)

Offset:0x0100+0x10+N*0x20 (N=0~8)			Register Name: CCR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:5	/	/	/
4	R/W1C	0x0	CRLF When the capture channel captures a rising edge, the current value of the 16-bit up-counter is latched to CRLR, and this bit is set 1 by hardware. Write 1 to clear this bit.
3	R/W1C	0x0	CFLF When the capture channel captures a falling edge, the current value of the 16-bit up-counter is latched to CFLR, and this bit is set 1 by hardware. Write 1 to clear this bit.
2	R/W	0x0	CRTE Rising edge capture trigger enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset:0x0100+0x10+N*0x20 (N=0~8)			Register Name: CCR
Bit	**Read/Write**	**Default/Hex**	**Description**
1	R/W	0x0	CFTE Falling edge capture trigger enable
0	R/W	0x0	CAPINV Inverse the signal input from capture channel before 16-bit counter of capture channel. 0: not inverse 1: inverse

7.10.6.25 0x0114 + N*0x20 PWM Capture Rise Lock Register (Default Value: 0x0000_0000)

Offset:0x0100+0x14+N*0x20 (N=0~8)			Register Name: CRLR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R	0x0	When the capture channel captures a rising edge, the current value of the 16-bit up-counter is latched to the register.

7.10.6.26 0x0118 + N*0x20 PWM Capture Fall Lock Register (Default Value: 0x0000_0000)

Offset:0x0100+0x18+N*0x20 (N=0~8)			Register Name: CFLR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R	0x0	When the capture channel captures a falling edge, the current value of the 16-bit up-counter is latched to the register.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.11 LEDC

7.11.1 Overview

The LEDC is used to control the external LED lamp.

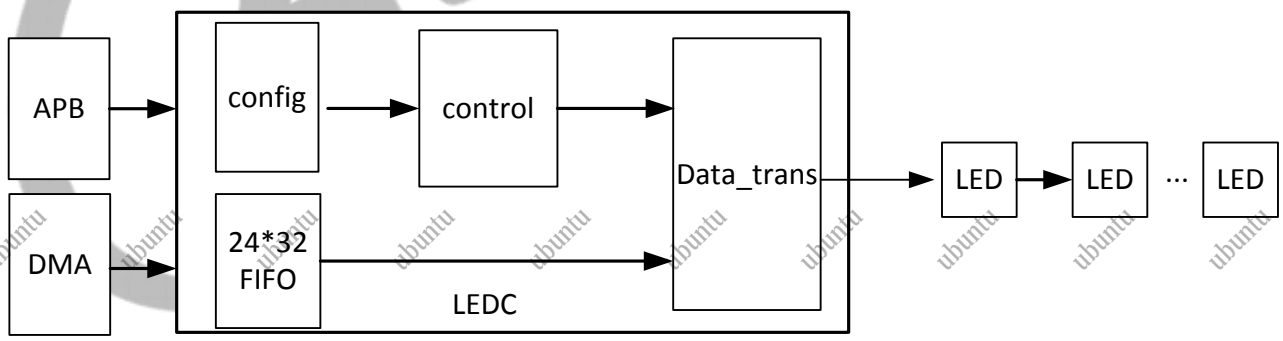
The LEDC has the following features:

- Configurable LED output high-/low-level width
- Configurable LED reset time
- LEDC data supports DMA configuration mode and CPU configuration mode
- Maximum 1024 LED serial connect
- LED data transfer rate up to 800 kbit/s
- Configurable RGB display mode
- The default level of non-data output is configurable

7.11.2 Block Diagram

The following figure shows a block diagram of the LEDC.

Figure 7-84 LEDC Block Diagram



LEDC contains the following sub-blocks:

Table 7-38 LEDC Sub-blocks

Sub-block	Description
config	register configuration
control	LEDC timing control and status control
FIFO	24-bit width x 32 depth
Data_trans	Convert input data to 0 and 1 characters

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.11.3 Functional Descriptions

7.11.3.1 External Signals

The following table describes the external signals of the LEDC.

Table 7-39 LEDC External Signals

Signal	Description	Type
LEDC-DO	Intelligent Control LED Signal Output	O

7.11.3.2 Clock Sources

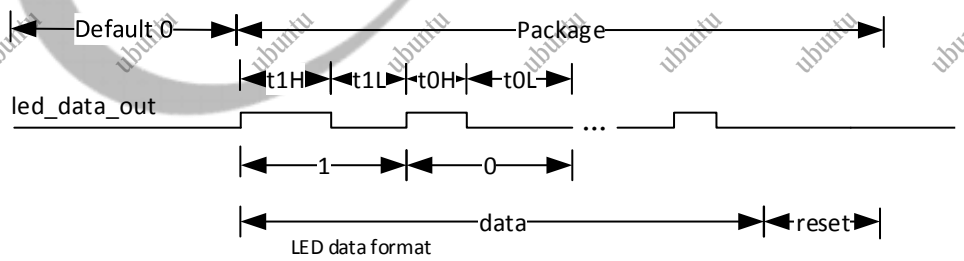
The following table describes the clock sources of the LEDC.

Table 7-40 LEDC Clock Sources

Clock Sources	Description
DCXO24M	24 MHz
PLL_PERI(1X)	Peripheral Clock. The default value is 600 MHz

7.11.3.3 LEDC Timing

Figure 7-85 LEDC Package Output Timing Diagram



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-86 LEDC 1-frame Output Timing Diagram

7.11.3.4 LEDC Input Data Structure

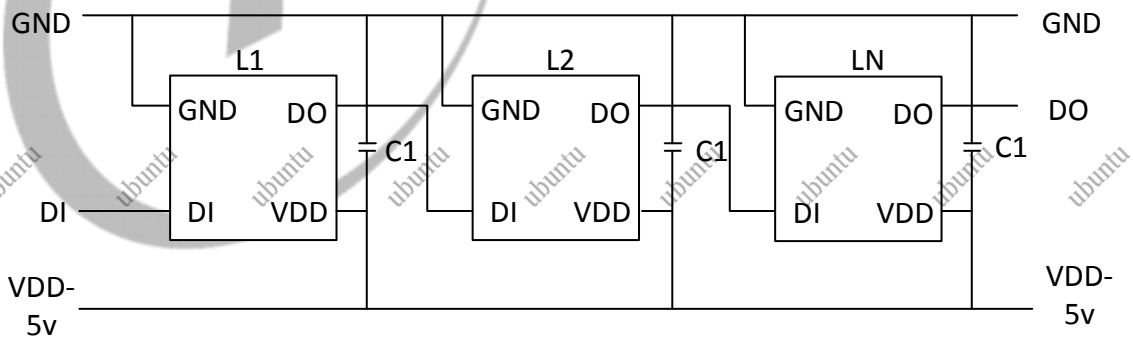
The RGB mode of LEDC data is configurable. By default, the data is sent in GRB order, and the higher bit is transmitted first.

Figure 7-87 LEDC Input Data Structure

G7	G6	G5	G4	G3	G2	G1	G0	R7	R6	R5	R4	R3	R2	R1	R0	B7	B6	B5	B4	B3	B2	B1	B0

7.11.3.5 LEDC Typical Circuit

Figure 7-88 LEDC Typical Circuit

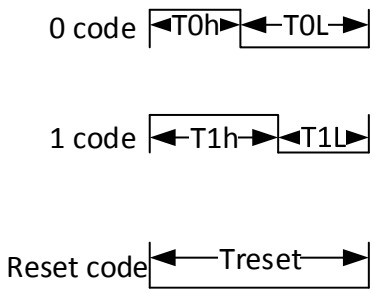


C1: filter capacitor of LED， usually is 100NF

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.11.3.6 LEDC Data Input Code

Figure 7-89 LEDC Data Input Code



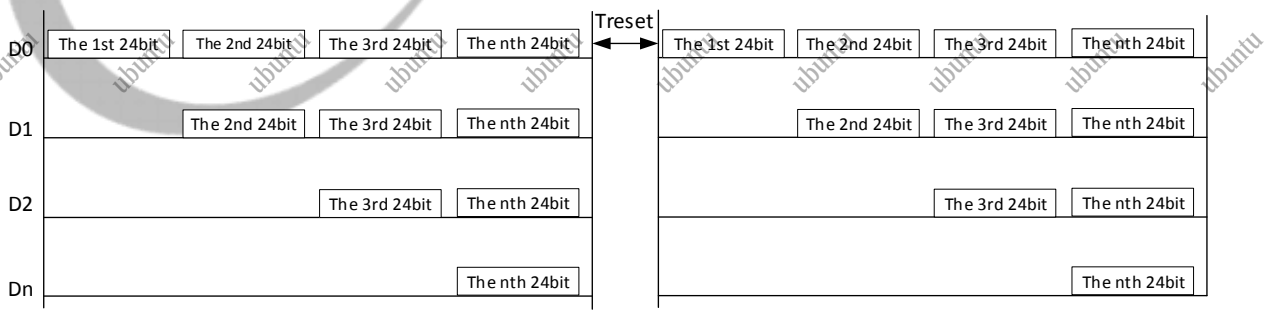
7.11.3.7 LEDC Data Transfer Time

The time parameter of the typical LED specification shows as follows.

T0H	0 code, high-level time	220 ns to 380 ns
T0L	0 code, low-level time	580 ns to 1.6 us
T1H	1 code, high-level time	580 ns to 1.6 us
T1L	1 code, low-level time	220 ns to 420 ns
RESET	Frame unit, low-level time	> 280 us

7.11.3.8 LEDC Data Transfer Mode

Figure 7-90 LEDC Data Transfer Mode



7.11.3.9 LEDC Parameter

1. PAD rate > 800 kbit/s
2. LED number supported:

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

$T_{0\text{-}code}$: 800 ns to 1980 ns, $T_{1\text{-}code}$: 800 ns to 2020 ns

When the LED refresh rate is 30 frame/s, LED number supported is (1 s/30-280 us)/((800 ns to 2020 ns)*24) =1024 ↓to 681.

When the LED refresh rate is 60 frame/s, LED number supported is (1 s/60-280 us)/((800 ns to 2020 ns)*24) =853 ↓to 338.

7.11.3.10 LEDC Data Transfer

The LEDC supports DMA data transfer mode or CPU data transfer mode. The DMA data transfer mode is set by *LEDC_DMA_EN.*

Data transfer in DMA mode

When the valid space of internal FIFO is greater than the setting FIFO free space threshold, the LEDC sends DMA_REQ to require DMA to transfer data from DRAM to LEDC. The maximum data transfer size in DMA mode is 16 words. (The internal FIFO level is 32.)

Data transfer in CPU mode

When the valid space of internal FIFO is greater than the setting FIFO free space threshold, the LEDC sends LEDC_CPUREQ_INT to require CPU to transfer data to LEDC. The transfer data size in CPU mode is controlled by software. The internal FIFO destination address is 0x06700014. The data width is 32-bit. (The lower 24-bit is valid.)

7.11.3.11 LEDC Interrupt

Module Name	Description
FIFO_OVERFLOW_INT	FIFO overflow interrupt. The data written by external is more than the maximum storage space of LED FIFO, the LEDC will be in data loss state. At this time, software needs to deal with the abnormal situation. The processing mode is as follows. The software can query LED_FIFO_DATA_REG to determine which data has been stored in the internal FIFO of LEDC. The LEDC performs soft_reset operation to refresh all data.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Module Name	Description
WAITDATA_TIMEOUT_INT	Wait for data timeout interrupt When internal FIFO of LEDC cannot get data because of some abnormal situation, the timeout interrupt is set after led_wait_data_time, now the LEDC is in WAIT_DATA state, and the LEDC outputs a level state configured by LED_POLARITY; in the course of wait_data, if the new data arrives, the LEDC will continue to send data, at this time the software needs to notice whether the waiting time of the LEDC exceeds the operation time of reset. If the waiting time of the LEDC exceeds the operation time of reset (this is equivalent to reset operation sent by LEDC), the LED may enter in refresh state, the data has not been sent.
FIFO_CPUREQ_INT	FIFO request CPU data interrupt When FIFO data is less than a threshold, the interrupt will be reported to the CPU.
LEDC_TRANS_FINISH_INT	Data transfer complete interrupt The value indicates that the data configured as total_data_length has been transferred completely.

LEDC interrupt usage scenario:

CPU mode

The software can enable GLOBAL_INT_EN, FIFO_CPUREQ_INT_EN, WAITDATA_TIMEOUT_INT_EN, FIFO_OVERFLOW_INT_EN, LEDC_TRANS_FINISH_INT_EN, and cooperate with LEDC_FIFO_TRIG_LEVEL to use. When FIFO_CPUREQ_INT is set to 1, the software can configure data of LEDC_FIFO_TRIG_LEVEL to LEDC.

DMA mode

The software can enable GLOBAL_INT_EN, WAITDATA_TIMEOUT_INT_EN, FIFO_OVERFLOW_INT_EN, LEDC_TRANS_FINISH_INT_EN, and cooperate with LEDC_FIFO_TRIG_LEVEL to use. When DMA receives LEDC DMA_REQ, DMA can transfer data of LEDC_FIFO_TRIG_LEVEL to LEDC.

7.11.4 Programming Guidelines

7.11.4.1 Configuring LEDC Normal Process

Step 1 Configure LEDC_CLK and bus pclk.

Step 2 Configure the written LEDC data.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

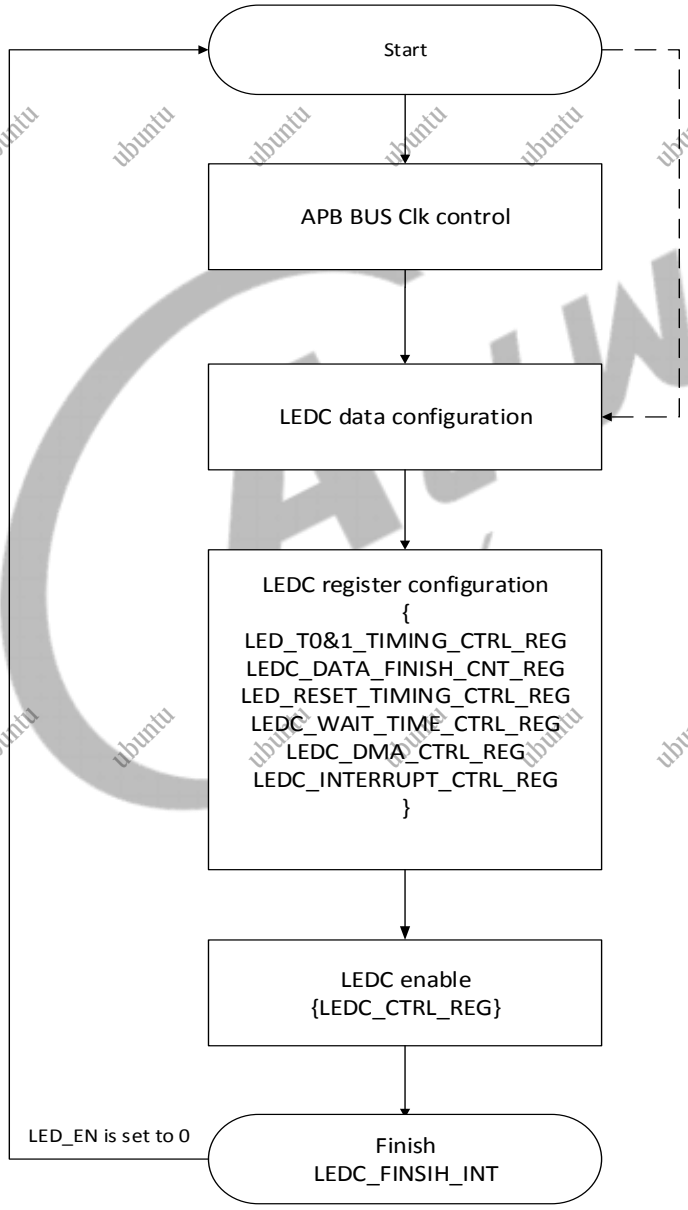
Step 3 Configure LED_T01_TIMING_CTRL_REG, LEDC_DATA_FINISH_CNT_REG, LED_RESET_TIMING_CTRL_REG, LEDC_WAIT_TIME0_CTRL_REG, LEDC_DMA_CTRL_REG, LEDC_INTERRUPT_CTRL_REG. Configure 0-code, 1-code, reset time, LEDC waiting time, and the number of external connected LEDC and the threshold of DMA transfer data.

Step 4 Configure LEDC_CTRL_REG to enable LEDC_EN, the LEDC will start to output data.

Step 5 When the LEDC interrupt is pulled up, it indicates the configured data has transferred complete, at this time LED_EN will be set to 0, and the read/write point of LEDC FIFO is cleared to 0.

Step 6 Repeat step1, 2, 3, 4 to re-execute a new round of configuration, enable LEDC_EN, the LEDC will start new data transfer.

Figure 7-91 LEDC Normal Configuration Process



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.11.4.2 Processing LEDC Abnormal Scene

WAITDATA_TIMEOUT Abnormal Status

Step 1 When WAITDATA_TIMEOUT_INT appears, it indicates the internal FIFO data request of LEDC cannot obtain a response, at this time if the default output level is low, then the external LED may think there was a reset operation and cause LED data to be flushed incorrectly.

Step 2 The LEDC needs to be performed soft_reset operation (LEDC_SOFT_RESET=1); after soft_reset, the LEDC_EN will be pulled-down automatically, all internal status register and control state machine will return to the idle state, the LEDC FIFO read & write point is cleared to 0, the LEDC interrupt is cleared.

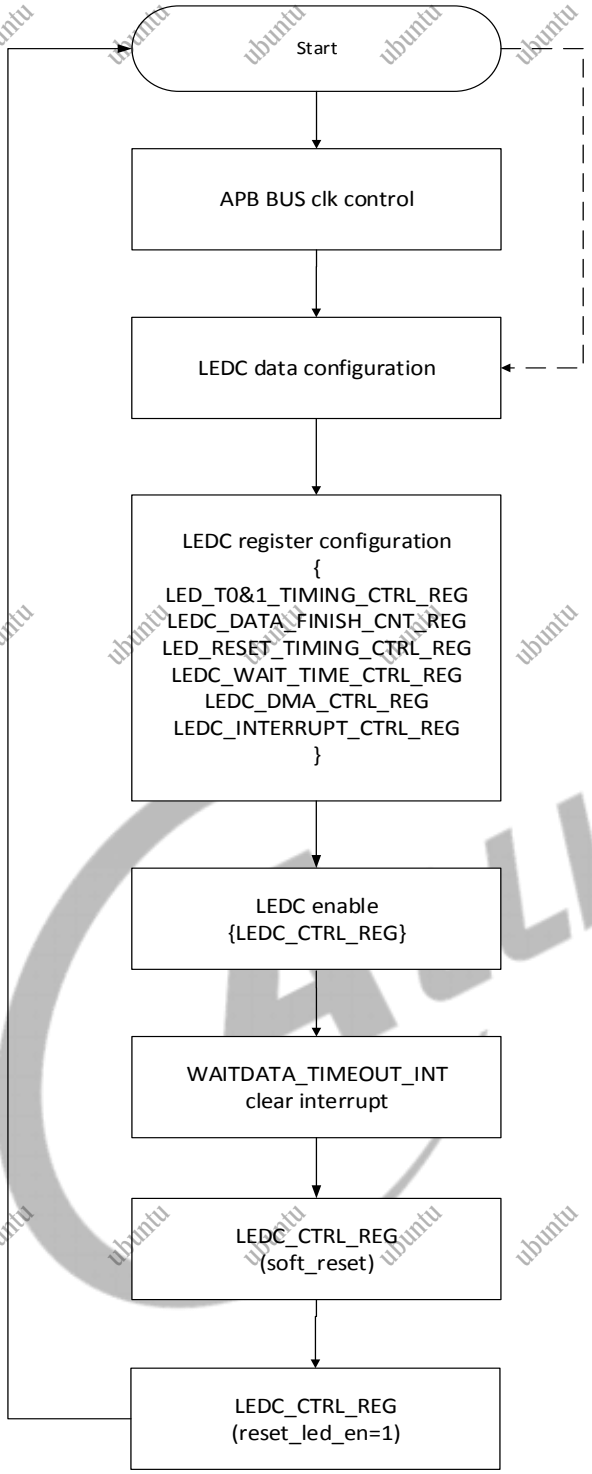
Step 3 Setting reset_led_en to 1 indicates LEDC can actively send a reset operation to ensure the external LED lamp in the right state.

Step 4 The software reads the status of reset_led_en, when the status value is 1, it indicates LEDC does not perform the transmission of LED reset operation; when the status value is 0, the LEDC completes the transmission of LED reset operation.

Step 5 When LEDC reset operation finishes, the LEDC data and register configuration need to be re-operated to start re-transmission data operation.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-92 LEDC Timeout Abnormal Processing Flow



FIFO Overflow Abnormal Status

Step 1 When FIFO_OVERFLOW_INT appears, it indicates the data configured by software exceeds the LEDC FIFO space, at this time the redundant data will be lost.

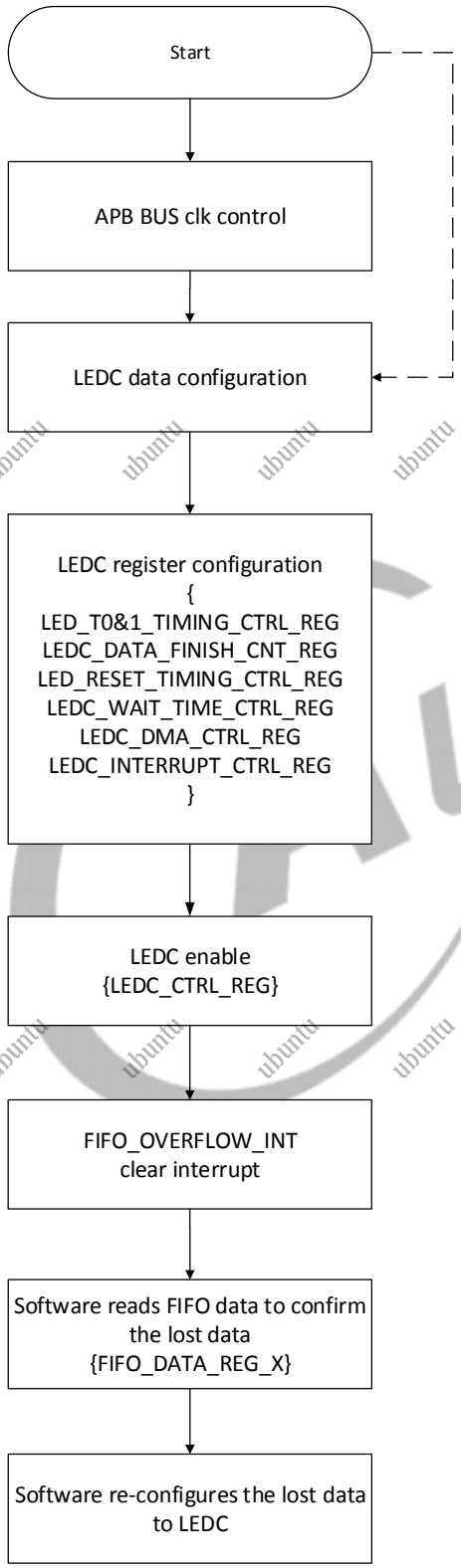
Step 2 The software needs to read data in **LEDC_FIFO_DATA_X** to confirm the lost data.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 3 The software re-configures the lost data to the LEDC.

Step 4 If the software uses the soft_reset operation, the operation is the same with the timeout abnormal processing flow.

Figure 7-93 FIFO Overflow Abnormal Processing Flow



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.11.5 Register List

Module Name	Base Address
LEDC	0x02002800

Register Name	Offset	Description
LEDC_CTRL_REG	0x0000	LEDC Control Register
LED_T01_TIMING_CTRL_REG	0x0004	LEDC T0 & T1 Timing Control Register
LEDC_DATA_FINISH_CNT_REG	0x0008	LEDC Data Finish Counter Register
LED_RESET_TIMING_CTRL_REG	0x000C	LEDC Reset Timing Control Register
LEDC_WAIT_TIME0_CTRL_REG	0x0010	Wait Time0 Control Register
LEDC_DATA_REG	0x0014	LEDC Data Register
LEDC_DMA_CTRL_REG	0x0018	DMA Control Register
LEDC_INT_CTRL_REG	0x001C	Interrupt Control Register
LEDC_INT_STS_REG	0x0020	Interrupt Status Register
LEDC_WAIT_TIME1_CTRL_REG	0x0028	Wait Time1 Control Register
LEDC_FIFO_DATA_REG	0x0030+4*N	LEDC FIFO Data Register

7.11.6 Register Description

7.11.6.1 0x0000 LEDC Control Register (Default Value: 0x0000_003C)

Offset: 0x0000			Register Name: LEDC_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	0x0	TOTAL_DATA_LENGTH Total length of transfer data(range: 0 to 8K, unit: 32-bit, only low 24-bit is valid) The field is recommended to be set to an integer multiple of (LED_NUM+1). If TOTAL_DATA_LENGTH is greater than (LED_NUM+1), but non-integer multiple, the last frame of data will transfer data less than (LED_NUM+1).
15:11	/	/	/
10	R/W	0x0	RESET_LED_EN Write operation:

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

<table>
<tr><td colspan="3">Offset: 0x0000</td><td>Register Name: LEDC_CTRL_REG</td></tr>
<tr><td>Bit</td><td>Read/Write</td><td>Default/Hex</td><td>Description</td></tr>
<tr><td></td><td></td><td></td><td>The software writes 1 to the bit, the CPU triggers LEDC to transfer a reset to LED.
Only when LEDC is in IDLE status, the reset can be performed. After reset finished, the control state machine returns to the IDLE status. To return LEDC to the IDLE status, it also needs to be used with SOFT_RESET.
When the software sets the bit, the software can read the bit to check if the reset is complete.
Read operation:
0: LEDC completes the transmission of the LED reset operation
1: LEDC does not complete the transmission of the LED reset operation</td></tr>
<tr><td>9</td><td>/</td><td>/</td><td>/</td></tr>
<tr><td>8:6</td><td>R/W</td><td>0x0</td><td>LED_RGB_MODE
000 GRB (bypass)
001 GBR
010 RGB
011 RBG
100 BGR
101 BRG
By default, the software configures data to LEDC according to GRB (MSB) mode, the LEDC internal combines data to output to the external LED.
Other modes configure as follows.
<table>
<tr><td>Software Input Mode</td><td>Configuration</td><td>LEDC Output Mode</td></tr>
<tr><td rowspan="6">GRB</td><td>000</td><td>GRB</td></tr>
<tr><td>001</td><td>GBR</td></tr>
<tr><td>010</td><td>RGB</td></tr>
<tr><td>011</td><td>RBG</td></tr>
<tr><td>100</td><td>BGR</td></tr>
<tr><td>101</td><td>BRG</td></tr>
<tr><td rowspan="4">GBR</td><td>000</td><td>GBR</td></tr>
<tr><td>001</td><td>GRB</td></tr>
<tr><td>010</td><td>BGR</td></tr>
<tr><td>011</td><td>BRG</td></tr>
</table></td></tr>
</table>

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: LEDC_CTRL_REG		
Bit	Read/Write	Default/Hex	Description		
				100	RGB
				101	RBG
			RGB	000	RGB
				001	RBG
				010	GRB
				011	GBR
				100	BRG
				101	BGR
			RBG	000	RBG
				001	RGB
				010	BRG
				011	BGR
				100	GRB
				101	GBR
			BGR	000	BGR
				001	BRG
				010	GBR
				011	GRB
				100	RBG
				101	RGB
			BRG	000	BRG
				001	BGR
				010	RBG
				011	RGB
				100	GBR
				101	GRB
5	R/W	0x1	LED_MSB_TOP Adjust sequence of the combined GRB data 0: LSB 1: MSB		
4	R/W	0x1	LED_MSB_G MSB control for Green data 0: LSB		

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: LEDC_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
			1: MSB
3	R/W	0x1	LED_MSB_R MSB control for Red data 0: LSB 1: MSB
2	R/W	0x1	LED_MSB_B MSB control for Blue data 0: LSB 1: MSB
1	R/W1C	0x0	LEDC_SOFT_RESET LEDC soft reset Write 1 to clear it automatically. The ranges of LEDC soft reset include the following points: all internal status registers, the control state machine returns to in idle status, the LEDC FIFO read & write point is cleared to 0, the LEDC interrupt is cleared; and the affected registers are followed. 1.LEDC_CTRL_REG(LEDC_EN is cleared to 0); 2. PLL_T0&1_TIMING_CTRL_REG remains unchanged; 3. LEDC_DATA_FINISH_CNT_REG(LEDC_DATA_FINISH_CNT is cleared to 0) 4.LED_RESET_TIMING_CTRL_REG remains unchanged; 5. LEDC_WAIT_TIME_CTRL_REG remains unchanged; 6. LEDC_DMA_CTRL_REG remains unchanged; 7. LEDC_INTERRUPT_CTRL_REG remains unchanged; 8.LEDC_INT_STS _REG is cleared to 0; 9. LEDC_CLK_GATING_REG remains unchanged; 10.LEDC_FIFO_DATA_REG remains unchanged;
0	R/W	0x0	LEDC_EN LEDC Enable 0: Disable 1: Enable That the bit is enabled indicates LEDC can be started when LEDC data finished transmission or LEDC_EN is cleared to 0 by hardware in LEDC_SOFT_RESET situation.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.11.6.2 0x0004 LEDC T0 & T1 Timing Control Register (Default Value: 0x0286_01D3)

Offset: 0x0004			Register Name: LED_T01_TIMING_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	/	/	/
26:21	R/W	0x14	T1H_TIME LED T1H time Unit: cycle (24 MHz), T1H_TIME =42 ns*(N+1) The default value is 882 ns, the range is 80 ns~2560 ns. N: 1~3F When is 0, T1H_TIME = 3F
20:16	R/W	0x6	T1L_TIME LED T1L time Unit: cycle (24 MHz), T1L_TIME =42 ns*(N+1) The default value is 294 ns, the range is 80 ns~1280 ns. N: 1~1F When is 0, T1L_TIME = 1F
15:11	/	/	/
10:6	R/W	0x7	T0H_TIME LED T0h time Unit: cycle (24 MHz), T0H_TIME =42 ns*(N+1) The default value is 336 ns, the range is 80 ns~1280 ns. N: 1~1F When is 0, T0H_TIME = 1F
5:0	R/W	0x13	T0L_TIME LED T0l time Unit: cycle (24 MHz), T0L_TIME =42 ns*(N+1) The default value is 840 ns, the range is 80 ns~2560 ns. N: 1~3F When is 0, T0L_TIME = 3F

7.11.6.3 0x0008 LEDC Data Finish Counter Register (Default Value: 0x1D4C_0000)

Offset: 0x0008			Register Name: LEDC_DATA_FINISH_CNT_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0008			Register Name: LEDC_DATA_FINISH_CNT_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
29:16	R/W	0x1D4C	LED_WAIT_DATA_TIME The value is the time that internal FIFO in LEDC is waiting for data. When the time is exceeded, the LEDC will send the wait_data_timeout_int interrupt. (This is an abnormal situation, software needs to reset LEDC.) The value is 300 us by default. The adjust range is from 80 ns to 655 us, led_wait_data_time=42ns*(N+1), N: 1~1FFF When the field is 0, LEDC_WAIT_DATA_TIME=1FFF
15:13	/	/	/
12:0	R	0x0	LED_DATA_FINISH_CNT The value is the total LED data that have been sent. (Range: 0~8k)

7.11.6.4 0x000C LEDC Reset Timing Control Register (Default Value: 0x1D4C_0000)

Offset: 0x000C			Register Name: LED_RESET_TIMING_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	/	/	/
28:16	R/W	0x1D4C	TR_TIME Reset time control of LED lamp Unit: cycle(24 MHz), tr_time=42 ns*(N+1) The default value is 300 us. The adjust range is from 80 ns to 327 us. N: 1~1FFF
15:10	/	/	/
9:0	R/W	0x0	LED_NUM The value is the number of external LED lamp. Maximum up to 1024. The default value 0 indicates that 1 LED lamp is external connected. The range is from 0 to 1023.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.11.6.5 0x0010 LEDC Wait Time 0 Control Register (Default Value: 0x0000_00FF)

Offset: 0x0010			Register Name: LEDC_WAIT_TIME0_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:9	/	/	/
8	R/W	0x0	WAIT_TIM0_EN WAIT_TIME0 enable When it is 1, the controller automatically inserts waiting time before LED package data. 0: Disable 1: Enable
7:0	R/W	0xFF	TOTAL_WAIT_TIME0 Waiting time between 2 LED data. The LEDC output is low level. The adjust range is from 80 ns to 10 us. wait_time0=42 ns*(N+1) Unit: cycle(24 MHz) N: 1~FF

7.11.6.6 0x0014 LEDC Data Register (Default Value: 0x0000_0000)

Offset: 0x0014			Register Name: LEDC_DATA_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	W	0x0	LEDC DATA LED display data (the lower 24-bit is valid)

7.11.6.7 0x0018 LEDC DMA Control Register (Default Value: 0x0000_002F)

Offset: 0x0018			Register Name: LEDC_DMA_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:6	/	/	/
5	R/W	0x1	LEDC_DMA_EN LEDC DMA request enable 0: Disable request of DMA transfer data 1: Enable request of DMA transfer data

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0018			Register Name: LEDC_DMA_CTRL_REG
Bit	Read/Write	Default/Hex	Description
4:0	R/W	0x0F	LEDC_FIFO_TRIG_LEVEL The remaining space of internal FIFO in LEDC The internal FIFO in LEDC is 24*32. When the remaining space of internal FIFO in LEDC is more than or equal to LEDFIFO_TRIG_LEVEL, DMA, or the CPU request will generate. The default value is 15. The adjusted value is from 1 to 31. The recommended configuration is 7 or 15. When the configuration value is 0, LEDFIFO_TRIG_LEVEL=F.

7.11.6.8 0x001C LEDC Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x001C			Register Name: LEDC_INTERRUPT_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:6	/	/	/
5	R/W	0x0	GLOBAL_INT_EN Global interrupt enable 0: Disable 1: Enable
4	R/W	0x0	FIFO_OVERFLOW_INT_EN FIFO overflow interrupt enable When the data written by the software is more than the internal FIFO level of LEDC, the LEDC is in the data loss state. 0: Disable 1: Enable
3	R/W	0x0	WAITDATA_TIMEOUT_INT_EN The internal FIFO in LEDC cannot get data because of some abnormal situation, after the time of led_wait_data_time, the interrupt will be enabled. 0: Disable 1: Enable
2	/	/	/
1	R/W	0x0	FIFO_CPUREQ_INT_EN FIFO request CPU data interrupt enable 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x001C			Register Name: LEDC_INTERRUPT_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
0	R/W	0x0	LED_TRANS_FINISH_INT_EN Data transmission complete interrupt enable 0: Disable 1: Enable

7.11.6.9 0x0020 LEDC Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: LEDC_INT_STS _REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17	R	0x1	FIFO_EMPTY FIFO empty status flag
16	R	0x0	FIFO_FULL FIFO full status flag
15:10	R	0x0	FIFO_WLW FIFO internal valid data depth It indicates the space FIFO has been occupied.
9:5	/	/	/
4	R/W1C	0x0	FIFO_OVERFLOW_INT FIFO overflow interrupt The data written by external is more than the maximum storage space of LED FIFO, the LEDC will be in the data loss state. At this time, the software needs to deal with the abnormal situation. The processing mode is as follows. The software can query LED_FIFO_DATA_REG to determine which data has been stored in the internal FIFO of LEDC. The LEDC performs soft_reset operation to refresh all data. 0: FIFO not overflow 1: FIFO overflow

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0020			Register Name: LEDC_INT_STS _REG
Bit	**Read/Write**	**Default/Hex**	**Description**
3	R/W1C	0x0	WAITDATA_TIMEOUT_INT When internal FIFO of LEDC cannot get data because of some abnormal situation, after led_wait_data_time, the timeout interrupt is set, the LEDC is in WAIT_DATA state, the LEDC outputs a level state configured by LED_POLARITY; in the course of wait_data, if new data arrives, the LEDC will continue to send data, at this time software needs to notice whether the waiting time of LEDC exceeds the operation time of reset. If the waiting time of LEDC exceeds the operation time of reset (this is equivalent to reset operation sent by LEDC), the LED may enter in refresh state, the data has not been sent. 0: LEDC not timeout 1: LEDC timeout
2	/	/	/
1	R/W1C	0x0	FIFO_CPUREQ_INT FIFO request CPU data interrupt When FIFO data is less than the threshold, the interrupt will be reported to the CPU. 0: FIFO does not request that CPU transfers data 1: FIFO requests that CPU transfers data
0	R/W1C	0x0	LED_TRANS_FINISH_INT Data transfer complete interrupt The value indicates that the data configured as total_data_length is transferred completely. 0: Data is not transferred completely 1: Data is transferred completely

7.11.6.10 0x0028 LEDC Wait Time 1 Control Register (Default Value: 0x1FF_FFFF)

Offset: 0x0028			Register Name: LEDC_WAIT_TIME1_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	WAIT_TIM1_EN 0: Disable 1: Enable WAIT_TIME1 enable When the bit is 1, the controller automatically inserts the waiting time before the LED frame data.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0028			Register Name: LEDC_WAIT_TIME1_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
30:0	R/W	0x1FF_FFFF	TOTAL_WAIT_TIME1 Waiting time between 2 frame data. The LEDC output is low level. The adjust range is from 80 ns to 85 s. wait_time1=42 ns*(N+1) Unit: cycle(24 MHz) N: 80~7FFF_FFFF

7.11.6.11 0x0030+N*0x04 LEDC FIFO Data Register X (Default Value: 0x0000_0000)

Offset: 0x0030+N*0x04 (N=0~31)			Register Name: LEDC_FIFO_DATA_X
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	LEDC_FIFO_DATA_X Internal FIFO data of LEDC The lower 24-bit is valid.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.12 EMAC

7.12.1 Overview

The Ethernet Medium Access Controller (EMAC) enables a host to transmit and receive data over Ethernet in compliance with the IEEE 802.3-2002 standard. It supports 10/100/1000 Mbit/s external PHY with RMII/RGMII interface in full-duplex and half-duplex modes. The internal DMA is designed for packet-oriented data transfers based on a linked list of descriptors.

The EMAC has the following features:

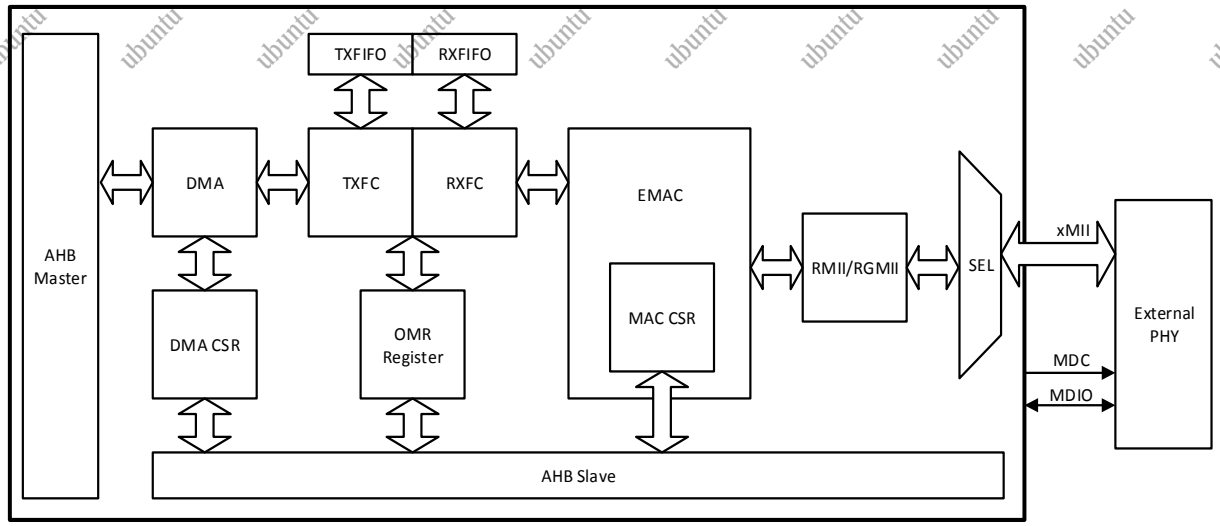
- One 10/100/1000 Mbit/s Ethernet port with reduced gigabit media independent interface (RGMII) and reduced media independent interface (RMII) interfaces, for connecting the external EPHY
- Compliant with IEEE 802.3-2002 standard
- Provides the management data input/output (MDIO) interface for PHY device configuration and management with configurable clock frequencies
- Supports both full-duplex and half-duplex operation
- Programmable frame length to support Standard or Jumbo Ethernet frames with sizes up to 16 KB
- Supports a variety of flexible address filtering modes
- Separate 32-bit status returned for transmission and reception packets
- Comprehensive status reporting for normal operation and transfers with errors
- Optimization for packet-oriented DMA transfers with frame delimiters
- Supports flow control, cyclic redundancy check (CRC) pad & stripping, and address filtering
- Supports linked-list descriptor list structure
- Descriptor architecture, allowing large blocks of data transfer with minimum CPU intervention; each descriptor can transfer up to 4 KB of data
- 4 KB TXFIFO for transmission packets and 16 KB RXFIFO for reception packets
- Programmable interrupt options for different operational conditions

7.12.2 Block Diagram

The following figure shows the block diagram of EMAC.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-94 EMAC Block Diagram



7.12.3 Functional Descriptions

7.12.3.1 External Signals

The following table describes the pin mapping of EMAC.

Table 7-41 EMAC Pin Mapping

Pin Name	RGMII	RMII
RGMII-RXD3	RXD3	/
RGMII-RXD2	RXD2	/
RGMII-RXD1/RMII-RXD1	RXD1	RXD1
RGMII-RXD0/RMII-RXD0	RXD0	RXD0
RGMII-RXCK	RXCK	/
RGMII-RXCTL/RMII-CRS-DV	RXCTL	CRS_DV
RMII-RXER	/	RXER
RGMII-TXD3	TXD3	/
RGMII-TXD2	TXD2	/
RGMII-TXD1/RMII-TXD1	TXD1	TXD1
RGMII-TXD0/RMII-TXD0	TXD0	TXD0
RGMII-TXCK/RMII-TXCK	TXCK	TXCK
RGMII-TXCTL/RMII-TXEN	TXCTL	TXEN
RGMII-CLKIN	CLKIN	/
MDC	MDC	MDC

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Pin Name	RGMII	RMII
MDIO	MDIO	MDIO
EPHY-25M	EPHY-25M	EPHY-25M

The following table describes the pin list of RGMII.

Table 7-42 EMAC RGMII Pin List

Pin Name	Description	Type
RGMII-TXD[3:0]	EMAC RGMII transmit data	O
RGMII-TXCTL	EMAC RGMII transmit control	O
RGMII-TXCK	EMAC RGMII transmit clock	O
RGMII-RXD[3:0]	EMAC RGMII receive data	I
RGMII-RXCTL	EMAC RGMII receive control	I
RGMII-RXCK	EMAC RGMII receive clock	I
RGMII-CKIN	EMAC RGMII 125M reference clock input	I
MDC	EMAC management data clock	O
MDIO	EMAC management data input output	I/O
EPHY-25M	25 MHz output for EMAC PHY	O

The following table describes the pin list of RMII.

Table 7-43 EMAC RMII Pin List

Pin Name	Description	Type
RMII-TXD[1:0]	EMAC RMII transmit data	O
RMII-TXEN	EMAC RMII transmit enable	O
RMII-TXCK	EMAC RMII transmit clock	I
RMII-RXD[1:0]	EMAC RMII receive data	I
RMII-CRS_DV	EMAC RMII receive data valid	I
RMII-RXER	EMAC RMII receive error	I
MDC	EMAC management data clock	O
MDIO	EMAC management data input output	I/O
EPHY-25M	25 MHz output for EMAC PHY	O

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.12.3.2 Clock Characteristics

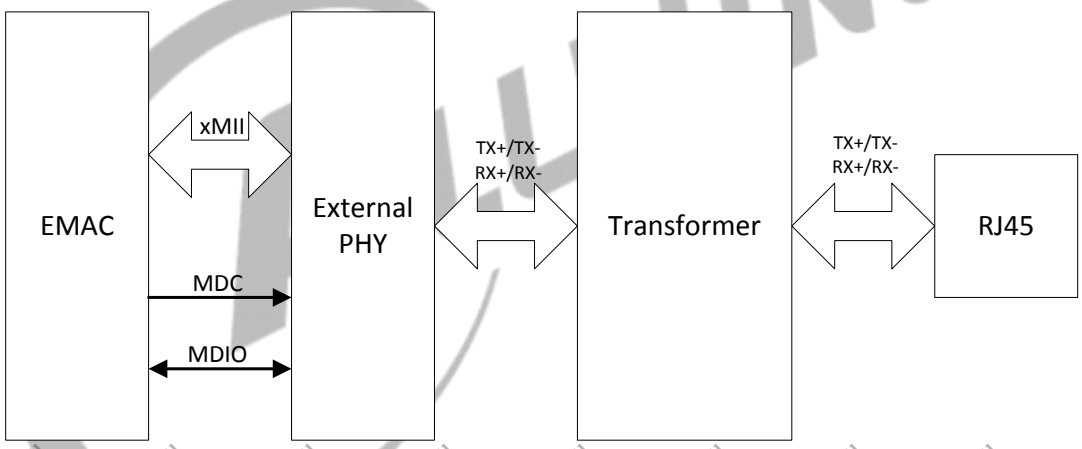
The following table describes the clock of EMAC.

Table 7-44 EMAC Clock Characteristics

Clock Name	Description	Type
RGMII0-TXCK/RMII0-TXCK	In RGMII mode, output 2.5 MHz/25 MHz/125 MHz. In RMII mode, input 5 MHz/50 MHz.	O/I
RGMII0-RXCK	In RGMII mode, input 2.5 MHz/25 MHz/125 MHz. In RMII mode, no input.	I
RGMII0-CLKIN	In RGMII mode, input 125M Reference Clock. In RMII mode, no clock.	I

7.12.3.3 Typical Application

Figure 7-95 EMAC Typical Application

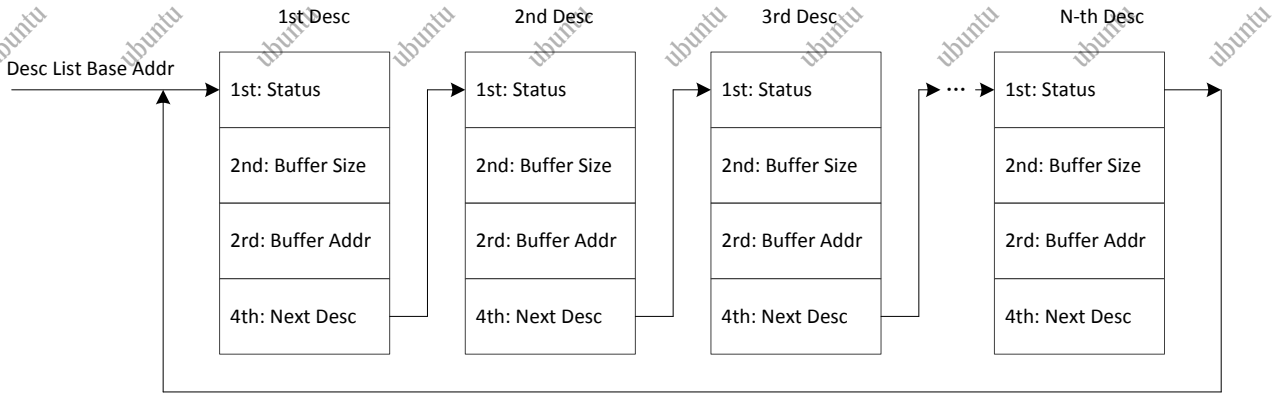


7.12.3.4 EMAC RX/TX Descriptor

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-96 EMAC RX/TX Descriptor List



7.12.3.5 TX Descriptor

1st Word of TX Descriptor

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Bits	Description
6:3	TX_COL_CNT The number of collisions before transmission.
2	TX_DEFER_ERR When set, the frame is aborted because of too much deferral.
1	TX_UNDERFLOW_ERR When set, the frame is aborted because of the TX FIFO underflow error.
0	TX_DEFER When set in Half-Duplex mode, the EMAC defers the frame transmission.

2nd Word of TX Descriptor

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

3rd Word of TX Descriptor

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

4th Word of TX Descriptor

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

7.12.3.6 RX Descriptor

1st Word of RX Descriptor

Bits	Description
31	RX_DESC_CTL When setting the bit, the current descriptor can be used by DMA. This bit is cleared by DMA when the complete frame is received or the buffer of the current descriptor is full.
30	RX_DAF_FAIL When setting the bit, the current frame does not pass the DA filter.
29:16	RX_FRM_LEN When LAST_DESC is not set and no error bit is set, this field is the length of received data for the current frame. When LAST_DESC is set, RX_OVERFLOW_ERR and RX_NO_ENOUGH_BUF_ERR are not set, this field is the length of the received frame.
15	Reserved
14	RX_NO_ENOUGH_BUF_ERR When setting the bit, the current frame is clipped because of no enough buffer.
13	RX_SAF_FAIL When setting the bit, the current fame does not pass the SA filter.
12	Reserved
11	RX_OVERFLOW_ERR When setting the bit, a buffer overflow error occurred and the current frame is wrong.
10	Reserved
9	FIR_DESC When setting the bit, the current descriptor is the first descriptor of the current frame.
8	LAST_DESC When setting the bit, the current descriptor is the last descriptor of the current frame.
7	RX_HEADER_ERR When setting the bit, the checksum of the frame header is wrong.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Bits	Description
6	RX_COL_ERR When setting the bit, there is a late collision during the reception in half-duplex mode.
5	Reserved
4	RX_LENGTH_ERR When setting the bit, the length of the current frame is wrong.
3	RX_PHY_ERR When setting the bit, the receive error signal from PHY is asserted during the reception.
2	Reserved
1	RX_CRC_ERR When setting the bit, the CRC field of the received frame is wrong.
0	RX_PAYLOAD_ERR When setting the bit, the checksum or length of the payload for the received frame is wrong.

2nd Word of RX Descriptor

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

3rd Word of RX Descriptor

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

4th Word of RX Descriptor

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.12.4 Programming Guidelines

7.12.4.1 Configuring EMAC Clock

Perform the following steps:

Step 1 Write 0 to EMAC_BGR_REG[bit16] to assert the module reset.

Step 2 Write 1 to EMAC_BGR_REG[bit16] to deassert the module reset.

Step 3 Write 1 to EMAC_BGR_REG[bit0] to enable the bus clock of the module.

Step 4 Configure the pin interfaces of EMAC by setting GPIO module.

Step 5 Configure EMAC_EPHY_CLK_REG0 to set the transmission clock source of RGMII/RMII.

For RGMII RXCLK/CLK125M:

In RGMII mode, in addition to the configuration of the transmission clock source, it is generally necessary to adjust the timing by configuring the transmission clock delay, reception clock delay, transmission clock reverse, reception clock reverse.

- Write 0 to the bit[13] and write 1 to the bit[2] to select the RGMII interface.
- If selecting RXCLK as the clock source of RGMII, write 2 to the bit[1:0]; if selecting CLK125M as the clock source of RGMII, write 1 to the bit[1:0].
- Write 0 to the bit[3], write 0 to the bit[4], write 31 to the bit[9:5], and write 7 to the bit[12:10] to transmit the reception sequence adjustment.

For RMII TXCLK:

- Write 1 to the bit[13] and write 0 to the bit[2] to select the RMII interface.
- Write 0 to the bit[0] to select TXCLK as the clock source of RMII.

The configuration value of EMAC_EPHY_CLK_REG0 can refer to the following table.

Table 7-45 EMAC_EPHY_CLK_REG0 Configuration Value

EMAC_EPHY_CLK_REG0	PHY_SEL	RMII_EN	ETXDC	ERXDC	ERXIE	ETXIE	MII/RGMII	ETCS
	Bit15	Bit13	Bit[12:10]	Bit[9:5]	Bit4	Bit3	Bit2	Bit[1:0]
RGMII	0	0	7	31	0	0	1	1/2
RMII	0	1	0	0	0	0	0	0

7.12.4.2 Initializing EMAC

Step 1 Write 1 to EMAC_BASIC_CTL1[bit0] to perform the software reset.

Step 2 Write 1 to EMAC_BASIC_CTL1[bit1] to set the DMA priority of TX/RX.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 3 Configure EMAC_TX_CTL1 and EMAC_RX_CTL1 to set the configuration of DMA TX and DMA RX.

Step 4 Configure EMAC_INT_EN to set the corresponding interrupts and shield the needless interrupts.

Step 5 Configure EMAC_TX_DMA_LIST and EMAC_RX_DMA_LIST to set the first address of the TX descriptor and the RX descriptor, respectively.

Step 6 Configure EMAC_TX_CTL0 and EMAC_RX_CTL0 to set the TX and RX parameters. Configure EMAC_BASIC_CTL0 to set the speed, duplex mode, loopback configuration. (If enabled the auto-negotiation, the configuration is performed as a result of the negotiation)

Step 7 Configure EMAC_RX_FRM_FLT to set the RX frame filter.

Step 8 Configure EMAC_TX_FLOW_CTL and EMAC_RX_CTL0 to set the control mechanism of TX and RX.

Step 9 Clear all interrupt flags.

Step 10 Write 1 to EMAC_TX_CTL0[bit31] and write 1 to EMAC_RX_CTL0[bit31] to enable the TX and RX functions.

7.12.5 Register List

Module Name	Base Address
EMAC	0x04500000

Register Name	Offset	Description
EMAC_BASIC_CTL0	0x0000	EMAC Basic Control Register0
EMAC_BASIC_CTL1	0x0004	EMAC Basic Control Register1
EMAC_INT_STA	0x0008	EMAC Interrupt Status Register
EMAC_INT_EN	0x000C	EMAC Interrupt Enable Register
EMAC_TX_CTL0	0x0010	EMAC Transmit Control Register0
EMAC_TX_CTL1	0x0014	EMAC Transmit Control Register1
EMAC_TX_FLOW_CTL	0x001C	EMAC Transmit Flow Control Register
EMAC_TX_DMA_DESC_LIST	0x0020	EMAC Transmit Descriptor List Address Register
EMAC_RX_CTL0	0x0024	EMAC Receive Control Register0
EMAC_RX_CTL1	0x0028	EMAC Receive Control Register1
EMAC_RX_DMA_DESC_LIST	0x0034	EMAC Receive Descriptor List Address Register
EMAC_RX_FRM_FLT	0x0038	EMAC Receive Frame Filter Register
EMAC_RX_HASH0	0x0040	EMAC Hash Table Register0
EMAC_RX_HASH1	0x0044	EMAC Hash Table Register1

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
EMAC_MII_CMD	0x0048	EMAC Management Interface Command Register
EMAC_MII_DATA	0x004C	EMAC Management Interface Data Register
EMAC_ADDR_HIGH0	0x0050	EMAC MAC Address High Register0
EMAC_ADDR_LOW0	0x0054	EMAC MAC Address Low Register0
EMAC_ADDR_HIGHN	0x0050+0x08*N (N=1~7)	EMAC MAC Address High Register N (N=1~7)
EMAC_ADDR_LOWN	0x0054+0x08*N (N=1~7)	EMAC MAC Address Low Register N (N=1~7)
EMAC_TX_DMA_STA	0x00B0	EMAC Transmit DMA Status Register
EMAC_TX_CUR_DESC	0x00B4	EMAC Current Transmit Descriptor Register
EMAC_TX_CUR_BUF	0x00B8	EMAC Current Transmit Buffer Address Register
EMAC_RX_DMA_STA	0x00C0	EMAC Receive DMA Status Register
EMAC_RX_CUR_DESC	0x00C4	EMAC Current Receive Descriptor Register
EMAC_RX_CUR_BUF	0x00C8	EMAC Current Receive Buffer Address Register
EMAC_RGMII_STA	0x00D0	EMAC RGMII Status Register

7.12.6 Register Description

7.12.6.1 0x0000 EMAC Basic Control Register0 (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: EMAC_BASIC_CTL0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:4	/	/	/
3:2	R/W	0x0	SPEED 00: 1000 Mbit/s 01: Reserved 10: 10 Mbit/s 11: 100 Mbit/s
1	R/W	0x0	LOOPBACK 0: Disable 1: Enable
0	R/W	0x0	DUPLEX 0: Half-duplex 1: Full-duplex

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.12.6.2 0x0004 EMAC Basic Control Register1 (Default Value: 0x0800_0000)

Offset: 0x0004			Register Name: EMAC_BASIC_CTL1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/
29:24	R/W	0x8	BURST_LEN The burst length of RX and TX DMA transfer.
23:2	/	/	/
1	R/W	0x0	RX_TX_PRI RX TX DMA Priority 0: Same priority 1: RX priority over TX
0	R/W	0x0	SOFT_RST Soft Reset all Registers and Logic 0: No valid 1: Reset All clock inputs must be valid before soft reset. This bit is cleared internally when the reset operation is completed fully. Before writing any register, this bit should read a 0.

7.12.6.3 0x0008 EMAC Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0008			Register Name: EMAC_INT_STA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W1C	0x0	RGMII_LINK_STA_P RMII Link Status Changed Interrupt Pending 0: No Pending 1: Pending Write '1' to clear it.
15:14	/	/	/
13	R/W1C	0x0	RX_EARLY_P RX DMA Filled First Data Buffer of the Receive Frame Interrupt Pending 0: No Pending 1: Pending Write '1' to clear it.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0008			Register Name: EMAC_INT_STA
Bit	Read/Write	Default/Hex	Description
12	R/W1C	0x0	RX_OVERFLOW_P RX FIFO Overflow Error Interrupt Pending 0: No Pending 1: Pending Write '1' to clear it.
11	R/W1C	0x0	RX_TIMEOUT_P RX Timeout Interrupt Pending 0: No Pending 1: Pending Write '1' to clear it. When this bit asserted, the length of the received frame is greater than 2048 bytes (10240 when JUMBO_FRM_EN is set)
10	R/W1C	0x0	RX_DMA_STOPPED_P When this bit asserted, the RX DMA FSM is stopped.
9	R/W1C	0x0	RX_BUF_UA _P RX Buffer UA Interrupt Pending 0: No Pending 1: Pending Write '1' to clear it. When this asserted, the RX DMA cannot acquire the next RX descriptor and RX DMA FSM is suspended. The ownership of the next RX descriptor should be changed to RX DMA. The RX DMA FSM will resume when writing to RX_DMA_START bit or next receive frame is coming.
8	R/W1C	0x0	RX_P Frame RX Completed Interrupt Pending 0: No Pending 1: Pending Write '1' to clear it. When this bit is asserted, a frame reception is completed. The RX DMA FSM remains running.
7:6	/	/	/
5	R/W1C	0x0	TX_EARLY_P The frame is transmitted to FIFO Total Interrupt Pending 0: No Pending 1: Pending Write '1' to clear it.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0008			Register Name: EMAC_INT_STA
Bit	**Read/Write**	**Default/Hex**	**Description**
4	R/W1C	0x0	TX_UNDERFLOW_P TX FIFO Underflow Interrupt Pending 0: No Pending 1: Pending Write '1' to clear it.
3	R/W1C	0x0	TX_TIMEOUT_P Transmitter Timeout Interrupt Pending 0: No Pending 1: Pending Write '1' to clear it.
2	R/W1C	0x0	TX_BUF_UA_P TX Buffer UA Interrupt Pending 0: No Pending 1: Pending When this asserted, the TX DMA can not acquire the next TX descriptor and the TX DMA FSM is suspended. The ownership of the next TX descriptor should be changed to TX DMA. The TX DMA FSM will resume when writing to TX_DMA_START bit.
1	R/W1C	0x0	TX_DMA_STOPPED_P Transmission DMA Stopped Interrupt Pending 0: No Pending 1: Pending Write '1' to clear it.
0	R/W1C	0x0	TX_P Frame Transmission Interrupt Pending 0: No Pending 1: Pending Write '1' to clear it.

7.12.6.4 0x000C EMAC Interrupt Enable Register (Default Value: 0x0000_0000)

Offset: 0x000C			Register Name: EMAC_INT_EN
Bit	**Read/Write**	**Default/Hex**	**Description**
31:14	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x000C			Register Name: EMAC_INT_EN
Bit	**Read/Write**	**Default/Hex**	**Description**
13	R/W	0x0	RX_EARLY_INT_EN Early Receive Interrupt 0: Disable 1: Enable
12	R/W	0x0	RX_OVERFLOW_INT_EN Receive Overflow Interrupt 0: Disable 1: Enable
11	R/W	0x0	RX_TIMEOUT_INT_EN Receive Timeout Interrupt 0: Disable 1: Enable
10	R/W	0x0	RX_DMA_STOPPED_INT_EN Receive DMA FSM Stopped Interrupt 0: Disable 1: Enable
9	R/W	0x0	RX_BUF_UA_INT_EN Receive Buffer Unavailable Interrupt 0: Disable 1: Enable
8	R/W	0x0	RX_INT_EN Receive Interrupt 0: Disable 1: Enable
7:6	/	/	/
5	R/W	0x0	TX_EARLY_INT_EN Early Transmit Interrupt 0: Disable 1: Enable
4	R/W	0x0	TX_UNDERFLOW_INT_EN Transmit Underflow Interrupt 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x000C			Register Name: EMAC_INT_EN
Bit	Read/Write	Default/Hex	Description
3	R/W	0x0	TX_TIMEOUT_INT_EN Transmit Timeout Interrupt 0: Disable 1: Enable
2	R/W	0x0	TX_BUF_UA_INT_EN Transmit Buffer Available Interrupt 0: Disable 1: Enable
1	R/W	0x0	TX_DMA_STOPPED_INT_EN Transmit DMA FSM Stopped Interrupt 0: Disable 1: Enable
0	R/W	0x0	TX_INT_EN Transmit Interrupt 0: Disable 1: Enable

7.12.6.5 0x0010 EMAC Transmit Control Register0 (Default Value: 0x0000_0000)

Offset: 0x0010			Register Name: EMAC_TX_CTL0
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	TX_EN Enable Transmitter 0: Disable 1: Enable When disabled, the transmission will continue until the current transmission finishes.
30	R/W	0x0	TX_FRM_LEN_CTL Frame Transmit Length Control 0: Up to 2,048 bytes (JUMBO_FRM_EN==0) Up to 10,240 bytes (JUMBO_FRM_EN==1) 1: Up to 16,384 bytes Any bytes after that is cut off.
29:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.12.6.6 0x0014 EMAC Transmit Control Register1 (Default Value: 0x0000_0000)

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

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.12.6.7 0x001C EMAC Transmit Flow Control Register (Default Value: 0x0000_0000)

Offset: 0x001C			Register Name: EMAC_TX_FLOW_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	TX_FLOW_CTL_STA This bit indicates a pause frame transmission is in progress. When the configuration of flow control is ready, set this bit to transmit a pause frame in full-duplex mode or activate the backpressure function. After completion of transmission, this bit will be cleared automatically. Before writing register TX_FLOW_CTRL, this bit must be read as 0.
30:22	/	/	/
21:20	R/W	0x0	TX_PAUSE_FRM_SLOT The threshold of the pause timer at which the input flow control signal is checked for automatic re-transmission of the pause frame. The threshold values should be always less than PAUSE_TIME.
19:4	R/W	0x0	PAUSE_TIME The pause time field in the transmitted control frame.
3:2	/	/	/
1	R/W	0x0	ZQP_FRM_EN 0: Disabled 1: Enabled When set, enable the functionality to generate the Zero-Quanta Pause control frame.
0	R/W	0x0	TX_FLOW_CTL_EN TX Flow Control Enable 0: Disabled 1: Enabled When set, enable flow control operation to transmit pause frames in full-duplex mode, or enable the back-pressure operation in half-duplex mode.

7.12.6.8 0x0020 EMAC Transmit DMA Descriptor List Address Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: EMAC_TX_DMA_LIST
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	TX_DESC_LIST The base address of the transmission descriptor list. It must be 32-bit aligned.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.12.6.9 0x0024 EMAC Receive Control Register0 (Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: EMAC_RX_CTL0
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	RX_EN Enable Receiver 0: Disable receiver after current reception 1: Enable
30	R/W	0x0	RX_FRM_LEN_CTL Frame Receive Length Control 0: Up to 2048 bytes (JUMBO_FRM_EN==0) Up to 10240 bytes (JUMBO_FRM_EN==1) 1: Up to 16384 bytes Any bytes after that is cut off
29	R/W	0x0	JUMBO_FRM_EN Jumbo Frame Enable 0: Disable 1: Enable Jumbo frames of 9,018 bytes without reporting a giant
28	R/W	0x0	STRIP_FCS When set, strip the Pad/FCS field on received frames only when the length of field value is less than or equal to 1500 bytes.
27	R/W	0x0	CHECK_CRC Check CRC Enable 0: Disable 1: Calculate CRC and check the IPv4 Header Checksum
26:18	/	/	/
17	R/W	0x0	RX_PAUSE_FRM_MD 0: Only detect multicast pause frame specified in the 802.3x standard. 1: In addition to detect multicast pause frame specified in the 802.3x standard, also detect unicast pause frame with the address specified in MAC Address 0 High Register and MAC address 0 Low Register.
16	R/W	0x0	RX_FLOW_CTL_EN When set, enable the functionality that decodes the received pause frame and disable its transmitter for a specified time by pause frame.
15:0	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.12.6.10 0x0028 EMAC Receive Control Register1 (Default Value: 0x0000_0000)

Offset: 0x0028			Register Name: EMAC_RX_CTL1
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	RX_DMA_START When set, the RX DMA will not work. It is cleared internally and always read a 0.
30	R/W	0x0	RX_DMA_EN Receive DMA Enable 0: Stop RX DMA after finishing the received current frame 1: Start and run RX DMA
29:25	/	/	/
24	R/W	0x0	RX_FIFO_FLOW_CTL Receive FIFO Flow Control Enable 0: Disable 1: Enable, base on RX_FLOW_CTL_TH_DEACT and RX_FLOW_CTL_TH_ACT
23:22	R/W	0x0	RX_FLOW_CTL_TH_DEACT Threshold for Deactivating Flow Control 00: Full minus 1 KB 01: Full minus 2 KB 10: Full minus 3 KB 11: Full minus 4 KB Valid in both half-duplex mode and full-duplex mode.
21:20	R/W	0x0	RX_FLOW_CTL_TH_ACT Threshold for Activating Flow Control 00: Full minus 1 KB 01: Full minus 2 KB 10: Full minus 3 KB 11: Full minus 4 KB Valid in both half-duplex mode and full-duplex mode.
19:6	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0028			Register Name: EMAC_RX_CTL1
Bit	Read/Write	Default/Hex	Description
5:4	R/W	0x0	RX_TH Threshold for RX DMA FIFO Start 00: 64 01: 32 10: 96 11: 128 Only valid when RX_MD == 0, the full frames with a length less than the threshold are transferred automatically.
3	R/W	0x0	RX_ERR_FRM 0: RX DMA drops frames with error 1: RX DMA forwards frames with error
2	R/W	0x0	RX_RUNT_FRM When setting, forward undersized frames with no error and length less than 64 bytes.
1	R/W	0x0	RX_MD Receive Mode 0: RX starts to read after the RX DMA FIFO byte is greater than RX_TH 1: RX starts to read after the RX DMA FIFO is located a full frame
0	R/W	0x0	FLUSH_RX_FRM Flush Receive Frames 0: Enable when receive descriptors/buffers is unavailable 1: Disable

7.12.6.11 0x0034 EMAC Receive DMA Descriptor List Address Register (Default Value: 0x0000_0000)

Offset: 0x0034			Register Name: EMAC_RX_DMA_LIST
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	RX_DESC_LIST The base address of the received descriptor list. It must be 32-bit aligned.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.12.6.12 0x0038 EMAC Receive Frame Filter Register (Default Value: 0x0000_0000)

Offset: 0x0038			Register Name: EMAC_RX_FRM_FLT
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	DIS_ADDR_FILTER Disable Address Filter 0: Enable 1: Disable
30:18	/	/	/
17	R/W	0x0	DIS_BROADCAST Disable Receive Broadcast Frames 0: Receive 1: Drop
16	R/W	0x0	RX_ALL_MULTICAST Receive All Multicast Frames Filter 0: Filter according to HASH_MULTICAST 1: Receive all
15:14	/	/	/
13:12	R/W	0x0	CTL_FRM_FILTER Receive Control Frames Filter 00: Drop all control frames 01: Drop all control frames 10: Receive all control frames 11: Receive all control frames when passing the address filter
11:10	/	/	/
9	R/W	0x0	HASH_MULTICAST Filter Multicast Frames Set 0: By comparing the DA field in DA MAC address registers 1: According to the hash table
8	R/W	0x0	HASH_UNICAST Filter Unicast Frames Set 0: By comparing the DA field in DA MAC address registers 1: According to the hash table
7	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0038			Register Name: EMAC_RX_FRM_FLT
Bit	Read/Write	Default/Hex	Description
6	R/W	0x0	SA_FILTER_EN Receive SA Filter Enable 0: Receive frames and update the result of SA filter 1: Update the result of the SA filter. In addition, if the SA field of the received frame does not match the values in SA MAC address registers, drop this frame.
5	R/W	0x0	SA_INV_FILTER Receive SA Invert Filter Set 0: Pass frames whose SA field matches SA MAC address registers 1: Pass frames whose SA field not matches SA MAC address registers
4	R/W	0x0	DA_INV_FILTER 0: Normal filtering of frames is performed 1: Filter both unicast and multicast frames by comparing DA field in inverse filtering mode
3:2	/	/	/
1	R/W	0x0	FLT_MD 0: If the HASH_MULTICAST or HASH_UNICAST is set, the frame is passed only when it matches the Hash filter 1: Receive the frame when it passes the address register filter or the hash filter(set by HASH_MULTICAST or HASH_UNICAST)
0	R/W	0x0	RX_ALL Receive All Frame Enable 0: Receive the frames that pass the SA/DA address filter 1: Receive all frames and update the result of address filter(pass or fail) in the receive status word

7.12.6.13 0x0040 EMAC Receive Hash Table Register0 (Default Value: 0x0000_0000)

Offset: 0x0040			Register Name: EMAC_RX_HASH0
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	HASH_TAB0 The upper 32 bits of Hash table for the received frame filter.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.12.6.14 0x0044 EMAC Receive Hash Table Register1 (Default Value: 0x0000_0000)

Offset: 0x0044			Register Name: EMAC_RX_HASH1
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	HASH_TAB1 The lower 32 bits of Hash table for the received frame filter.

7.12.6.15 0x0048 EMAC MII Command Register (Default Value: 0x0000_0000)

Offset: 0x0048			Register Name: EMAC_MII_CMD
Bit	Read/Write	Default/Hex	Description
31:23	/	/	/
22:20	R/W	0x0	MDC_DIV_RATIO_M MDC Clock Divider Ratio The MDC Clock is divided from the AHB clock. 000: 16 001: 32 010: 64 011: 128 Others: Reserved
19:17	/	/	/
16:12	R/W	0x0	PHY_ADDR PHY Address
11:9	/	/	/
8:4	R/W	0x0	PHY_REG_ADDR PHY Register Address
3:2	/	/	/
1	R/W	0x0	MII_WR MII Write and Read 0: Read 1: Write
0	R/W	0x0	MII_BUSY 0: Writing 0 is no valid, reading 0 indicates the read or write operation is finished 1: Write 1 to start the read or write operation, reading 1 indicates busy.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.12.6.16 0x004C EMAC MII Data Register (Default Value: 0x0000_0000)

Offset: 0x004C			Register Name: EMAC_MII_DATA
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x0	MII_DATA Write to or read from the register in the selected PHY.

7.12.6.17 0x0050 EMAC MAC Address High Register0 (Default Value: 0x0000_FFFF)

Offset: 0x0050			Register Name: EMAC_ADDR_HIGH0
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0xFFFF	MAC_ADDR_HIGH0 The upper 16 bits of the 1st MAC address.

7.12.6.18 0x0054 EMAC MAC Address Low Register0 (Default Value: 0xFFFF_FFFF)

Offset: 0x0054			Register Name: EMAC_ADDR_LOW0
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0xFFFFFFFF	MAC_ADDR_LOW0 The lower 32 bits of 1st MAC address.

7.12.6.19 0x0050+0x08*N EMAC MAC Address High Register N (Default Value: 0x0000_0000)

Offset: 0x0050+0x08*N (N=1~7)			Register Name: EMAC_ADDR_HIGHN
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	MAC_ADDR_CTL MAC Address Valid 0: Not valid 1: Valid
30	R/W	0x0	MAC_ADDR_TYPE MAC Address Type 0: Used to compare with the destination address of the received frame 1: Used to compare with the source address of the received frame

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0050+0x08*N (N=1~7)			Register Name: EMAC_ADDR_HIGHN
Bit	Read/Write	Default/Hex	Description
29:24	R/W	0x0	MAC_ADDR_BYTE_CTL MAC Address Byte Control Mask The lower bit of mask controls the lower byte of the MAC address. When the bit of mask is 1, do not compare the corresponding byte.
23:16	/	/	/
15:0	R/W	0x0	MAC_ADDR_HIGH The upper 16 bits of the MAC address.

7.12.6.20 0x0054+0x08*N EMAC MAC Address Low Register N (Default Value: 0x0000_0000)

Offset: 0x0054+0x08*N (N=1~7)			Register Name: EMAC_ADDR_LOWN
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	MAC_ADDR_LOWN The lower 32 bits of MAC address N (N: 1~7).

7.12.6.21 0x00B0 EMAC Transmit DMA Status Register (Default Value: 0x0000_0000)

Offset: 0x00B0			Register Name: EMAC_TX_DMA_STA
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/
2:0	R	0x0	TX_DMA_STA The State of Transmit DMA FSM 000: STOP, when reset or disable TX DMA 001: RUN_FETCH_DESC, fetching TX DMA descriptor 010: RUN_WAIT_STA, waiting for the status of TX frame 011: RUN_TRANS_DATA, passing the frame from host memory to TX DMA FIFO 100: Reserved 101: Reserved 111: RUN_CLOSE_DESC, closing TX descriptor 110: SUSPEND, TX descriptor unavailable or TX DMA FIFO underflow

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.12.6.22 0x00B4 EMAC Transmit DMA Current Descriptor Register (Default Value: 0x0000_0000)

Offset: 0x00B4			Register Name: EMAC_TX_DMA_CUR_DESC
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	The address of current transmit descriptor.

7.12.6.23 0x00B8 EMAC Transmit DMA Current Buffer Address Register (Default Value: 0x0000_0000)

Offset: 0x00B8			Register Name: EMAC_TX_DMA_CUR_BUF
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	The address of current transmit DMA buffer.

7.12.6.24 0x00C0 EMAC Receive DMA Status Register (Default Value: 0x0000_0000)

Offset: 0x00C0			Register Name: EMAC_RX_DMA_STA
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/
2:0	R	0x0	RX_DMA_STA The State of RX DMA FSM 000: STOP, when reset or disable RX DMA 001: RUN_FETCH_DESC, fetching RX DMA descriptor 010: Reserved 011: RUN_WAIT_FRM, waiting for the frame 100: SUSPEND, RX descriptor unavailable 101: RUN_CLOSE_DESC, closing RX descriptor 110: Reserved 111: RUN_TRANS_DATA, passing the frame from host memory to RX DMA FIFO

7.12.6.25 0x00C4 EMAC Receive DMA Current Descriptor Register (Default Value: 0x0000_0000)

Offset: 0x00C4			Register Name: EMAC_RX_DMA_CUR_DESC
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	The address of current receive descriptor

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.12.6.26 0x00C8 EMAC Receive DMA Current Buffer Address Register (Default Value: 0x0000_0000)

Offset: 0x00C8			Register Name: EMAC_RX_DMA_CUR_BUF
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	The address of current receive DMA buffer

7.12.6.27 0x00D0 EMAC RGMII Status Register (Default Value: 0x0000_0000)

Offset: 0x00D0			Register Name: EMAC_RGMII_STA
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/
3	R	0x0	RGMII_LINK The link status of the RGMII interface 0: Down 1: Up
2:1	R	0x0	RGMII_LINK_SPD The link speed of the RGMII interface 00: 2.5 MHz 01: 25 MHz 10: 125 MHz 11: Reserved
0	R	0x0	RGMII_LINK_MD The link mode of the RGMII interface 0: Half-Duplex 1: Full-Duplex

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.13 CIR Receiver

7.13.1 Overview

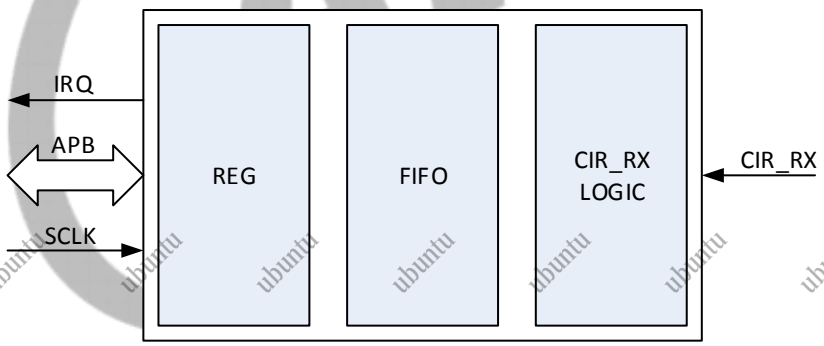
The Consumer Infrared (CIR) receiver captures pulse from the IR Receiver module and uses the Run-Length Code (RLC) to encode the pulse.

The CIR receiver has the following features:

- Full physical layer implementation
- Supports CIR for remote control or wireless keyboard
- Supports NEC IR protocol
- 64x8 bits FIFO for data buffer
- Programmable FIFO thresholds
- Supports interrupt
- Sample clock up to 1 MHz

7.13.2 Block Diagram

Figure 7-97 CIR Receiver Block Diagram



The CIR receiver samples the input signal on the programmable frequency and records these samples into RX FIFO when one CIR signal is found on the air. The CIR receiver uses Run-Length Code (RLC) to encode pulse width. The encoded data is buffered in 64 levels and 8-bit width RX FIFO; the MSB bit is used to record the polarity of the receiving CIR signal, the rest 7 bits are used for the length of RLC. The maximum length of the RLC is 128. If the duration of one level (high or low level) is more than 128, another byte is used.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.13.3 Functional Descriptions

7.13.3.1 External Signals

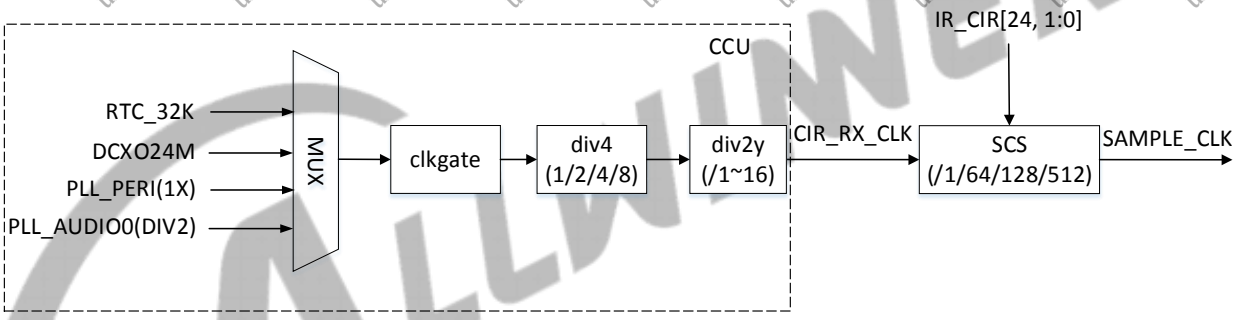
The following table describes the external signals of CIR Receiver.

Table 7-46 CIR Receiver External Signals

Signal	Description	Type
IR-RX	Consumer Infrared Receiver in CPUX Domain	I
S-IR-RX	Consumer Infrared Receiver in CPUS Domain	I

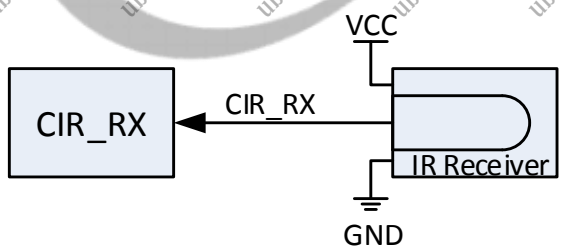
7.13.3.2 Clock Sources

Figure 7-98 CIR Receiver Clock



7.13.3.3 Typical Application

Figure 7-99 CIR Receiver Application Diagram



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.13.3.4 NEC Protocol Format

Figure 7-100 NEC Protocol

The CIR receiver module is a timer with a capture function.

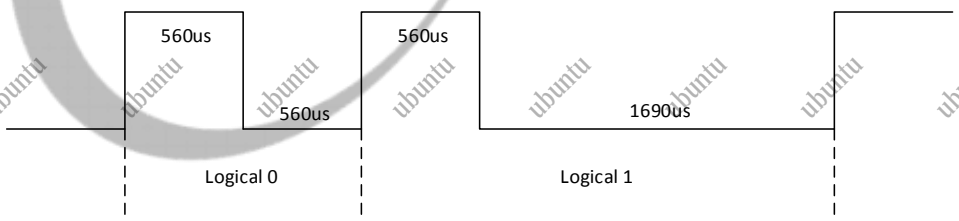
When CIR_RX signals satisfy the Active Threshold (ATHR), the CIR receiver can start to capture. In the process, the signal is ignored if the pulse width of the signal is less than NTHR. When CIR_RX signals satisfy ITHR (Idle Threshold), the capture process is stopped and the *Receiver Packet End interrupt* is generated, then the *Receiver Packet End Flag* is asserted.

In a capture process, every effective pulse is buffered to FIFO in bytes according to the form of the Run-Length Code. The MSB bit of a byte is the polarity of pulse, and the rest 7 bits is pulse width by taking Sample Clock as a basic unit. This is the code form of the RLC-Byte. When the level changes or the pulse width counter overflows, the RLC-Byte is buffered to FIFO. The CIR_RX module receives the infrared signals transmitted by the infrared remote control, the software decodes the signals.

7.13.3.5 Operating Mode

Sample Clock

Figure 7-101 Logical '0' and Logical '1' of NEC Protocol



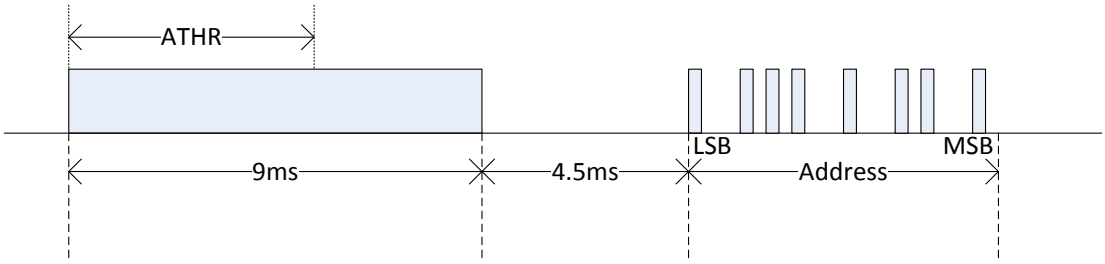
For NEC protocol, a logical "1" takes 2.25 ms (560 us+1680 us) to transmit, while a logical "0" is only half of that, being 1.12 ms (560 us+560 us). For example, if the sample clock is 31.25 kHz, a sample cycle is 32 us, then 18 sample cycles are 560 us. So the RLC of 560 us low level is 0x12 (b'00010010), the RLC of 560 us high level is 0x92 (b'10010010). Then a logical "1" takes code 0x12 (b'00010010) and code 0xb5 (b'10110101) to transmit, a logical "0" takes code 0x12 and code 0x92 to transmit.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Active Threshold (ATHR)

When the CIR receiver is in Idle state, if the electrical level of the CIR_RX signal changes (positive jump or negative jump), and the duration reaches this threshold, then the CIR receiver takes the starting of the signal as a lead code, and the CIR receiver turns into an active state and starts to capture CIR_RX signals.

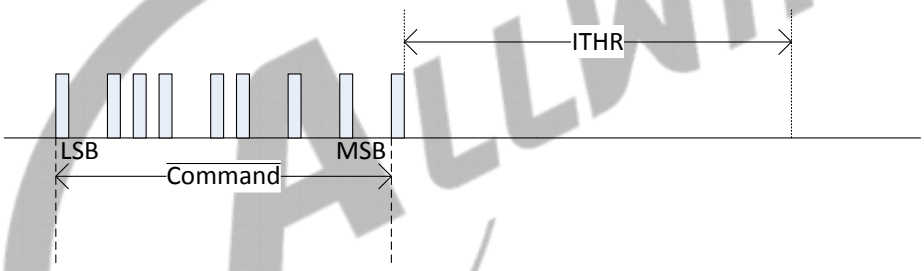
Figure 7-102 ATHR Definition



Idle Threshold (ITHR)

If the electrical level of CIR_RX signals has no change, and the duration reaches this threshold, then the CIR receiver enters into Idle state and ends this capture.

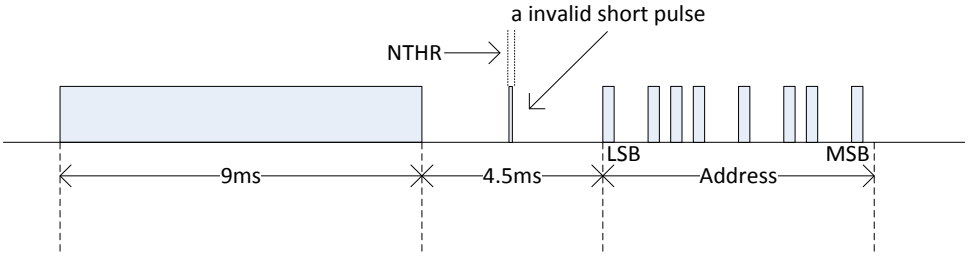
Figure 7-103 ITHR Definition



Noise Threshold (NTHR)

In the capture process, the pulse is ignored if the pulse width is less than the Noise Threshold.

Figure 7-104 NTHR Definition

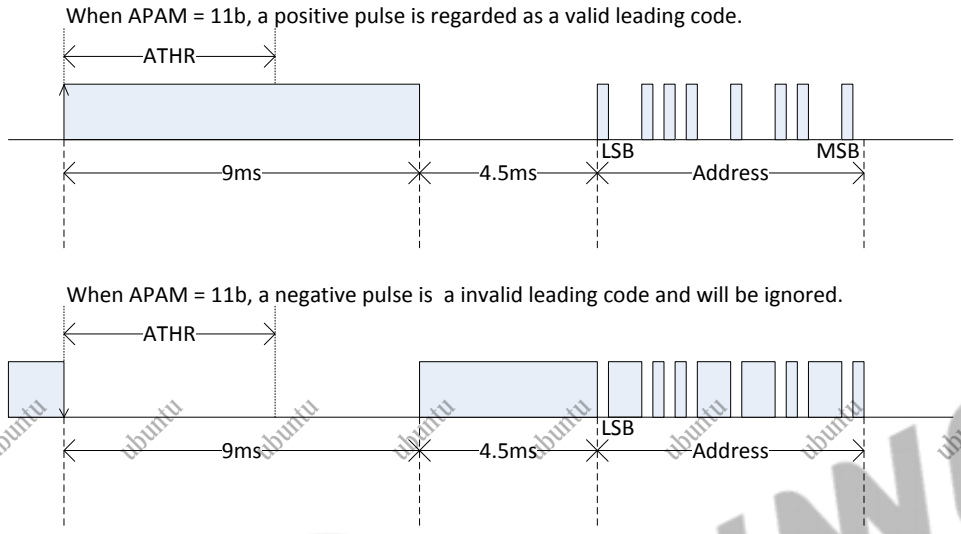


Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Active Pulse Accept Mode (APAM)

The APAM is used to fit the type of lead code. If a pulse does not fit the type of lead code, it is not regarded as a lead code even if the pulse width reaches ATHR.

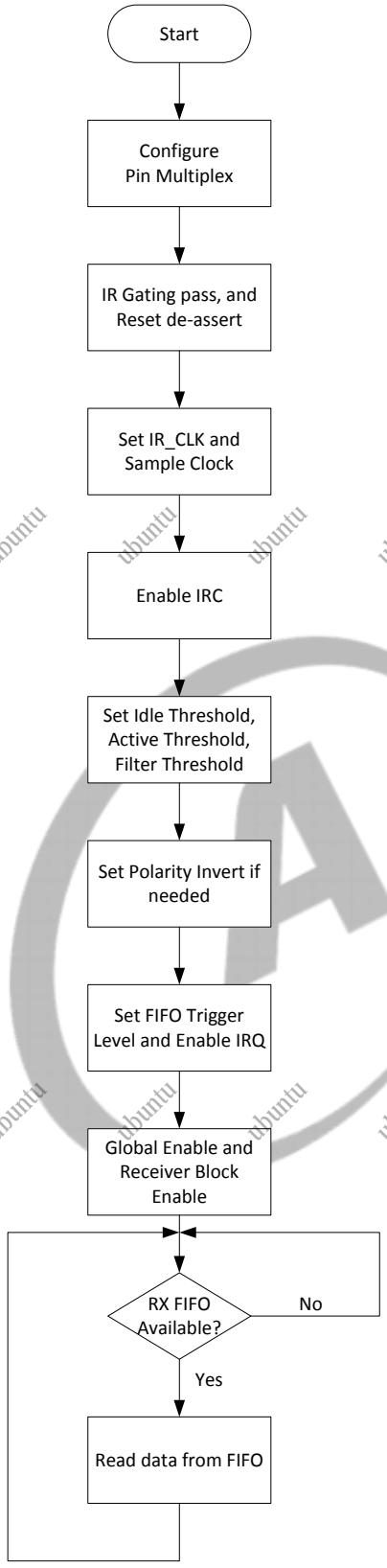
Figure 7-105 APAM Definition



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.13.4 Programming Guidelines

Figure 7-106 CIR Receiver Process



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.13.5 Register List

Module Name	Base Address
CIR_RX	0x02003400
R_CIR_RX	0x07040000

Register Name	Offset	Description
CIR_CTL	0x0000	CIR Control Register
CIR_RXPCFG	0x0010	CIR Receiver Pulse Configure Register
CIR_RXFIFO	0x0020	CIR Receiver FIFO Register
CIR_RXINT	0x002C	CIR Receiver Interrupt Control Register
CIR_RXSTA	0x0030	CIR Receiver Status Register
CIR_RXCFG	0x0034	CIR Receiver Configure Register

7.13.6 Register Description

7.13.6.1 0x0000 CIR Receiver Control Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: CIR_CTL
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:6	R/W	0x0	Active Pulse Accept Mode 00, 01: Both positive and negative pulses are valid as a leading code 10: Only negative pulse is valid as a leading code 11: Only positive pulse is valid as a leading code
5:4	R/W	0x0	CIR ENABLE 00~10: Reserved 11: CIR mode enable
3:2	/	/	/
1	R/W	0x0	RXEN Receiver Block Enable 0: Disable 1: Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: CIR_CTL
Bit	Read/Write	Default/Hex	Description
0	R/W	0x0	GEN Global Enable A disable on this bit overrides any other block or channel enables and flushes all FIFOs. 0: Disabled 1: Enabled

7.13.6.2 0x0010 CIR Receiver Pulse Configure Register (Default Value: 0x0000_0004)

Offset: 0x0010			Register Name: CIR_RXPCFG
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/
2	R/W	0x1	RPPI Receiver Pulse Polarity Invert 0: Do not invert receiver signal 1: Invert receiver signal
1:0	/	/	/

7.13.6.3 0x0020 CIR Receiver FIFO Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: CIR_RXFIFO
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R	0x0	Receiver Byte FIFO

7.13.6.4 0x002C CIR Receiver Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x002C			Register Name: CIR_RXINT
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x002C			Register Name: CIR_RXINT
Bit	**Read/Write**	**Default/Hex**	**Description**
13:8	R/W	0x0	RAL RX FIFO available received byte level for interrupt and DMA request TRIGGER_LEVEL = RAL + 1
5	R/W	0x0	DRQ_EN RX FIFO DMA Enable 0: Disable 1: Enable When set to '1', the Receiver FIFO DRQ is asserted if reaching RAL. The DRQ is de-asserted when the condition fails.
4	R/W	0x0	RAI_EN RX FIFO Available Interrupt Enable 0: Disable 1: Enable When set to '1', the Receiver FIFO IRQ is asserted if reaching RAL. The IRQ is de-asserted when the condition fails.
3:2	/	/	/
1	R/W	0x0	RPEI_EN Receiver Packet End Interrupt Enable 0: Disable 1: Enable
0	R/W	0x0	ROI_EN Receiver FIFO Overrun Interrupt Enable 0: Disable 1: Enable

7.13.6.5 0x0030 CIR Receiver Status Register (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: CIR_RXSTA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:15	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0030			Register Name: CIR_RXSTA
Bit	**Read/Write**	**Default/Hex**	**Description**
14:8	R	0x0	RAC RX FIFO Available Counter 0: No available data in RX FIFO 1: 1-byte available data in RX FIFO 2: 2-bytes available data in RX FIFO … 64: 64-bytes available data in RX FIFO
7	R	0x0	STAT Status of CIR 0: Idle 1: Busy
6:5	/	/	/
4	R/W1C	0x0	RA RX FIFO Available 0: RX FIFO not available according to its level 1: RX FIFO available according to its level Writing 1 clears this bit.
3:2	/	/	/
1	R/W1C	0x0	RPE Receiver Packet End Flag 0: STO was not detected. In CIR mode, one CIR symbol is receiving or not detected. 1: STO field or packet abort symbol (7'b0000,000 and 8'b0000,0000 for MIR and FIR) is detected. In CIR mode, one CIR symbol is received. Writing 1 clears this bit.
0	R/W1C	0x0	ROI Receiver FIFO Overrun 0: Receiver FIFO not overrun 1: Receiver FIFO overrun Writing 1 clears this bit.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.13.6.6 0x0034 CIR Receiver Configure Register (Default Value: 0x0000_1828)

Offset: 0x0034			Register Name: CIR_RXCFG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:25	/	/	/
24	R/W	0x0	SCS2 Bit2 of Sample Clock Select for CIR This bit is defined by SCS bits below.
23	R/W	0x0	ATHC Active Threshold Control for CIR 0: ATHR in a unit of (Sample Clock) 1: ATHR in a unit of (128*Sample Clocks)
22:16	R/W	0x0	ATHR Active Threshold for CIR These bits control the duration of CIR from the idle to the active state. The duration can be calculated by ((ATHR + 1)*(ATHC? Sample Clock: 128*Sample Clock)).
15:8	R/W	0x18	ITHR Idle Threshold for CIR The Receiver uses it to decide whether the CIR command is received. If there is no CIR signal on the air, the receiver is staying in IDLE status. One active pulse will bring the receiver from IDLE status to Receiving status. After the CIR receiver ends, the inputting signal will keep the specified level (high or low level) for a long time. The receiver can use this idle signal duration to decide that it has received the CIR command. The corresponding flag is asserted. If the corresponding interrupt is enabled, the interrupt line is asserted to the CPU. When the duration of the signal keeps one status (high or low level) for the specified duration ((ITHR + 1)*128 sample_clk), this means that the previous CIR command is finished.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

<table>
<tr><td colspan="3">Offset: 0x0034</td><td>Register Name: CIR_RXCFG</td></tr>
<tr><td>Bit</td><td>Read/Write</td><td>Default/Hex</td><td>Description</td></tr>
<tr><td>7:2</td><td>R/W</td><td>0xA</td><td>NTHR
Noise Threshold for CIR
When the duration of the signal pulse (high or low level) is less than NTHR, the pulse is taken as noise and should be discarded by hardware.
0: All samples are recorded into RX FIFO
1: If the signal is only one sample duration, it is taken as noise and discarded.
2: If the signal is less than (<=) two sample duration, it is taken as noise and discarded.
…
61: If the signal is less than (<=) sixty-one sample duration, it is taken as noise and discarded.</td></tr>
<tr><td>1:0</td><td>R/W</td><td>0x0</td><td>SCS
Sample Clock Select for CIR
<table>
<tr><td>SCS2</td><td>SCS[1]</td><td>SCS[0]</td><td>Sample Clock</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>CIR_CLK/64</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>CIR_CLK/128</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>CIR_CLK/256</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>CIR_CLK/512</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>CIR_CLK</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>Reserved</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>Reserved</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>Reserved</td></tr>
</table></td></tr>
</table>

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.14 CIR Transmitter

7.14.1 Overview

The CIR transmitter (CIR_TX) can transfer arbitrary waves which can be modulated with configurable carrier waves such as 38 kHz. It only uses the low 8-bit of a 32-bit register, so it stores a 16-bit number by using two 32-bit registers, one register stores the high 8-bit, the other register stores the low 8-bit.

The CIR_TX has the following features:

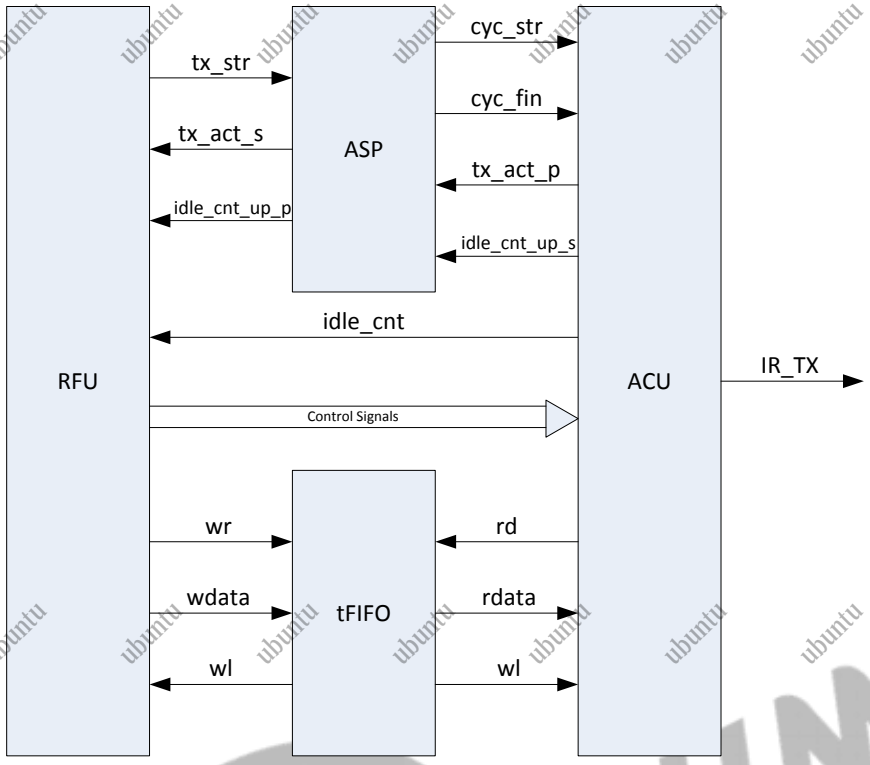
- Supports industry-standard AMBA Peripheral Bus (APB)
- Fully compliant with the AMBA Specification, Revision 2.0
- 128 bytes FIFO for data buffer
- Configurable carrier frequency
- Supports Interrupt and DMA
- Supports handshake mode and waiting mode of DMA
- Supports arbitrary wave generator

7.14.2 Block Diagram

The following figure shows a block diagram of the CIR_TX.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-107 CIR_TX Block Diagram



7.14.3 Functional Descriptions

7.14.3.1 External Signals

The following table describes the external signals of CIR_TX.

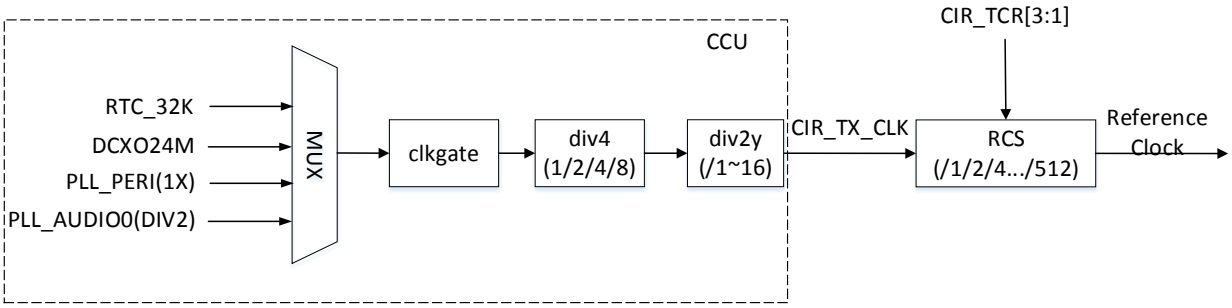
Table 7-47 CIR_TX External Signals

Signal	Description	Type
CIR-TX	Consumer infrared transmitter	O

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.14.3.2 Clock Sources

Figure 7-108 CIR_TX Clock Description



7.14.3.3 Function Implementation

The CIR_TX is used to generate a waveform of arbitrary length, arbitrary shape, and no high-speed requirement, and it can change the data into the high-/low-level sequence of a certain length. Every transmitting data is in bytes, the Bit[7] of a byte means whether the level of a transmitting wave is high or low, the Bit[6:0] is the length of this wave. If the current transmitting frequency-division is 1, 0x88 is a high level of 8 cycles, 0x08 is a low level of 8 cycles. If the current transmitting frequency-division is 4, 0x88 is a high level of 32 cycles, 0x08 is a low level of 32 cycles.

The CIR_TX has two transmission modes: non-cycle transmission, and cycle transmission.

The non-cycle transmission is to transmit all the data in TX_FIFO until the FIFO is empty.

The cycle transmission is to transmit all the data in TX_FIFO, after the transmission completion, wait for a certain time to recover the data in TX_FIFO and then send it until a stop signal is detected. The data recovery in FIFO is implemented by clearing the read pointer.

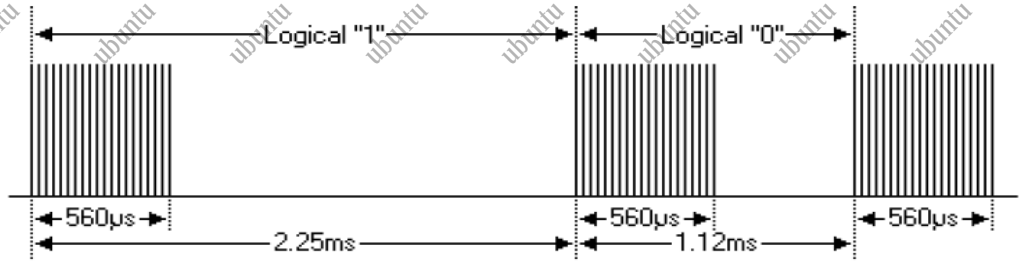
7.14.3.4 Timing Diagram

The CIR remote control contains many protocols designed by different manufacturers. Here to NEC protocol as an example, the CIR-TX module uses a variable pulse-width modulation technique to encompass the various formats of infrared encoding for remote-control applications. A message is started by a 9 ms AGC burst, which is used to set the gain of the earlier CIR receivers. This AGC burst is then followed by a 4.5 ms space, which is then followed by the address and command.

Bit definition: the logical “1” takes 2.25 ms to transmit, while a logical “0” is only 1.12 ms.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-109 Definitions of Logical “1” and Logical “0”



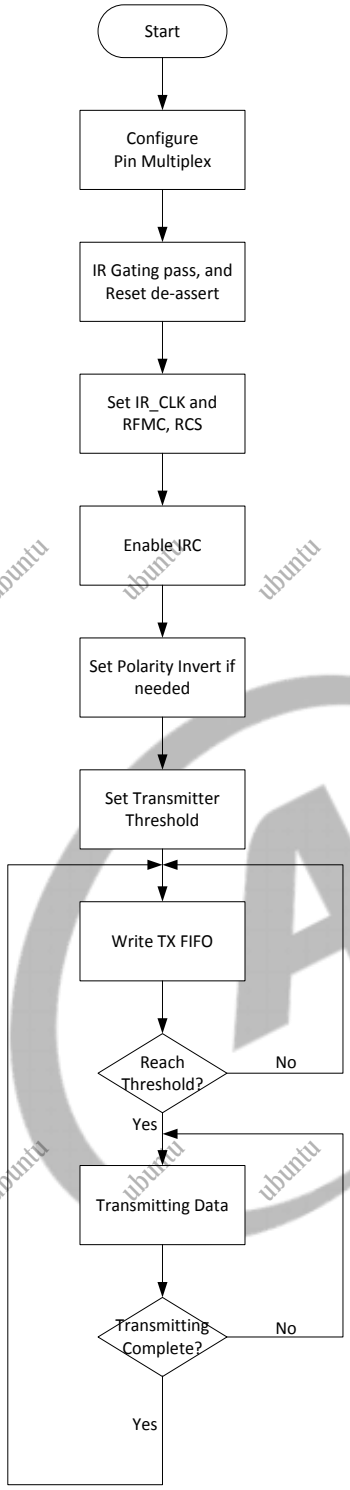
Timing for a message:

Figure 7-110 CIR Message Timing Diagram

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.14.4 Programming Guidelines

Figure 7-111 CIR Transmitter Process



Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.14.5 Register List

Module Name	Base Address
CIR_TX	0x02003000

Register Name	Offset	Description
CIR_TGLR	0x0000	CIR Transmit Global Register
CIR_TMCR	0x0004	CIR Transmit Modulation Control Register
CIR_TCR	0x0008	CIR Transmit Control Register
CIR_IDC_H	0x000C	CIR Transmit Idle Duration Threshold High Bit Register
CIR_IDC_L	0x0010	CIR Transmit Idle Duration Threshold Low Bit Register
CIR_TICR_H	0x0014	CIR Transmit Idle Counter High Bit Register
CIR_TICR_L	0x0018	CIR Transmit Idle Counter Low Bit Register
CIR_TEL	0x0020	CIR TX FIFO Empty Level Register
CIR_TXINT	0x0024	CIR Transmit Interrupt Control Register
CIR_TAC	0x0028	CIR Transmit FIFO Available Counter Register
CIR_TXSTA	0x002C	CIR Transmit Status Register
CIR_TXT	0x0030	CIR Transmit Threshold Register
CIR_DMA	0x0034	CIR DMA Control Register
CIR_TXFIFO	0x0080	CIR Transmit FIFO Data Register

7.14.6 Register Description

7.14.6.1 0x0000 CIR Transmitter Global Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: CIR_TGLR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R/W	0x0	IMS Internal Modulation Select 0: The transmitting signal is not modulated 1: The transmitting signal is modulated internally

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: CIR_TGLR
Bit	**Read/Write**	**Default/Hex**	**Description**
6:5	R/W	0x0	DRMC Duty ratio of modulated carrier is high level/low level. 00: Low level is equal to high level 01: Low level is the double of high level 10: Low level is the triple of high level 11: Reserved
4:3	/	/	/
2	R/W	0x0	TPPI Transmit Pulse Polarity Invert 0: Not invert transmit pulse 1: Invert transmit pulse
1	R/W	0x0	TR Transmit Reset When this bit is set, the transmitting is reset. The FIFO will be flushed, the TIC filed and the CSS field will be cleared during Transmit Reset. This field will automatically be cleared when the Transmit Reset is finished, and the CIR transmitter will state Idle.
0	R/W	0x0	TXEN Transmit Block Enable 0: Disable the CIR Transmitter 1: Enable the CIR Transmitter

7.14.6.2 0x0004 CIR Transmitter Modulation Control Register (Default Value: 0x0000_009E)

Offset: 0x0004			Register Name: CIR_TMCR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R/W	0x9E	RFMC Reference Frequency of modulated carrier. Reference Frequency of modulated carrier based on a division of a fixed functional clock (FCLK). The range of the modulated carrier is usually 30 kHz to 60 kHz. Most consumer electronics is 38 kHz. The default modulated carrier is 38 kHz when FCLK is 12 MHz. RFMC= FCLK/((N+1)*(DRMC+2)).

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.14.6.3 0x0008 CIR Transmitter Control Register (Default Value: 0x0000_0000)

Offset: 0x0008			Register Name: CIR_TCR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R/W	0x0	CSS Cyclical Pulse Start/Stop Control 0: Stop when cleared to '0'. From start to stop, all data in FIFO must be transmitted. 1: Start. Start to transmit when it is set to '1'.
6:4	/	/	/
3:1	R/W	0x0	RCS Reference Clock Select for CIR Transmit The data in TX_FIFO is used to describe the pulse in Run-Length Code. The basic unit of pulse width is Reference Clock. 000: CIR Transmit reference clock is ir_clk 001: CIR Transmit reference clock is ir_clk/2 010: CIR Transmit reference clock is ir_clk/4 011: CIR Transmit reference clock is ir_clk/8 100: CIR Transmit reference clock is ir_clk/64 101: CIR Transmit reference clock is ir_clk/128 110: CIR Transmit reference clock is ir_clk/256 111: CIR Transmit reference clock is ir_clk/512
0	R/W	0x0	TTS Type of the transmission signal 0: The transmitting wave is a single non-cyclical pulse. 1: The transmitting wave is a cyclical short-pulse.

7.14.6.4 0x000C CIR Transmitter Idle Duration Counter High Bit Register (Default Value: 0x0000_0000)

Offset: 0x000C			Register Name: CIR_IDC_H
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x000C			Register Name: CIR_IDC_H
Bit	Read/Write	Default/Hex	Description
3:0	R/W	0x0	IDC_H Idle Duration Counter Threshold (High 4 bits) Idle Duration = 128*IDC*Ts (IDC = 0~4095) It is used in cyclical transmission mode. When all the data in FIFO is transmitted, the signals can be transmitted after a specific time.

7.14.6.5 0x0010 CIR Transmitter Idle Duration Counter Low Bit Register (Default Value: 0x0000_0000)

Offset: 0x0010			Register Name: CIR_IDC_L
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
3:0	R/W	0x0	IDC_L Idle Duration Counter Threshold (Low 8 bits) Idle Duration = 128*IDC*Ts (IDC = 0~4095) It is used in cyclical transmission mode. When all the data in FIFO is transmitted, the signals can be transmitted after a specific time.

7.14.6.6 0x0014 CIR Transmitter Idle Counter High Bit Register (Default Value: 0x0000_0000)

Offset: 0x0014			Register Name: CIR_TICR_H
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R	0x0	TIC_H Transmit Idle Counter_H (High 8 bits) It is used to count the idle duration of CIR transmitter by software. Count in 128*Ts (Sample Duration, 1/Fs) when the transmitter is idle, and it should be reset when the transmitter is active. When this counter reaches the maximum value (0xFFFF), it will stop automatically, and should not be cleared to zero.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.14.6.7 0x0018 CIR Transmitter Idle Counter Low Bit Register (Default Value: 0x0000_0000)

Offset: 0x0018			Register Name: CIR_TICR_L
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R	0x0	TIC_L Transmit Idle Counter_L (Low 8 bits) It is used to count the idle duration of CIR transmitter by software. Count in 128*Ts (Sample Duration, 1/Fs) when the transmitter is idle, and it should be reset when the transmitter is active. When this counter reaches the maximum value (0xFFFF), it will stop automatically, and should not be cleared to zero.

7.14.6.8 0x0020 CIR Transmitter FIFO Empty Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: CIR_TEL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R/W	0x0	TEL TX FIFO empty Level for DRQ and IRQ. TRIGGER_LEVEL = TEL + 1

7.14.6.9 0x0024 CIR Transmitter Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: CIR_TXINT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2	R/W	0x0	DRQ_EN TX FIFO DMA Enable 0: Disable 1: Enable When set to '1', the TX FIFO DRQ is asserted if the number of the transmitting data in the FIFO is less than the RAL. The DRQ is de-asserted when the condition fails.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0024			Register Name: CIR_TXINT
Bit	Read/Write	Default/Hex	Description
1	R/W	0x0	TAI_EN TX FIFO Available Interrupt Enable 0: Disable 1: Enable
0	R/W	0x0	TPEI_EN Transmit Packet End Interrupt Enable for Cyclical Pulse 0: Disable 1: Enable TUI_EN Transmitter FIFO Underrun Interrupt Enable for Non-cyclical Pulse 0: Disable 1: Enable

7.14.6.10 0x0028 CIR Transmitter FIFO Available Counter Register (Default Value: 0x0000_0080)

Offset: 0x0028			Register Name: CIR_TAC
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:0	R	0x80	TAC TX FIFO Available Space Counter 0x00: No available space in TX FIFO 0x01: 1 byte available space in TX FIFO 0x02: 2 bytes available space in TX FIFO … 0x80: 128 bytes available space in TX FIFO

7.14.6.11 0x002C CIR Transmitter Status Register (Default Value: 0x0000_0002)

Offset: 0x002C			Register Name: CIR_TXSTA
Bit	Read/Write	Default/Hex	Description
31:4	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x002C			Register Name: CIR_TXSTA
Bit	**Read/Write**	**Default/Hex**	**Description**
3	R	0x0	STCT Status of CIR Transmitter 0: Idle 1: Active This bit will automatically set when the controller begins to transmit the data in the FIFO. The “1” will last when the data in the FIFO. It will automatically be cleared to “0” when all data in the FIFO is transmitted. The bit is for debugging. The output Level of Idle state is determined by the level of the last data output.
2	R	0x0	DRQ DMA Request Flag When set to ‘1’, the TX FIFO DRQ is asserted if the number of the transmission data in the FIFO is less than the RAL. The DRQ is de-asserted when the condition fails. This bit is for debugging.
1	R/W	0x1	TAI TX FIFO Available Interrupt Flag 0: TX FIFO not available by its level 1: TX FIFO available by its level Writing 1 clears this bit.
0	R/W	0x0	TPE Transmitter Packet End Flag for Cyclical Pulse 0: Transmissions of address, control and data fields not completed 1: Transmissions of address, control and data fields completed TUR Transmitter FIFO Underrun Flag for Non-cyclical Pulse 0: No transmitter FIFO underrun 1: Transmitter FIFO underrun Writing 1 clears this bit.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.14.6.12 0x0030 CIR Transmitter Threshold Register (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: CIR_TXT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R/W	0x0	NCTT Non-cyclical Pulse Transmit Threshold The controller will trigger transmitting the data in the FIFO when the data byte number has reached the Transmit Threshold set in this field.

7.14.6.13 0x0034 CIR Transmitter DMA Control Register (Default Value: 0x0000_00A5)

Offset: 0x0034			Register Name: CIR_DMA_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R/W	0xA5	DMA Handshake Configuration 0xA5: DMA waiting cycle mode 0xEA: DMA handshake mode

7.14.6.14 0x0080 CIR Transmitter FIFO Data Register (Default Value: 0x0000_0000)

Offset: 0x0080			Register Name: CIR_TXFIFO
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	W	0x0	Transmit Byte FIFO When the transmission is triggered, the data in the FIFO will be transmitted until the data number is transmitted completely.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.15 SCR

7.15.1 Overview

The Smart Card Reader (SCR) is a communication controller that transmits data between the system and the Smart Card. The controller can perform a complete smart card session, including card activation, card deactivation, cold/warm reset, Answer to Reset (ATR) response reception, data transfers, etc.

Features:

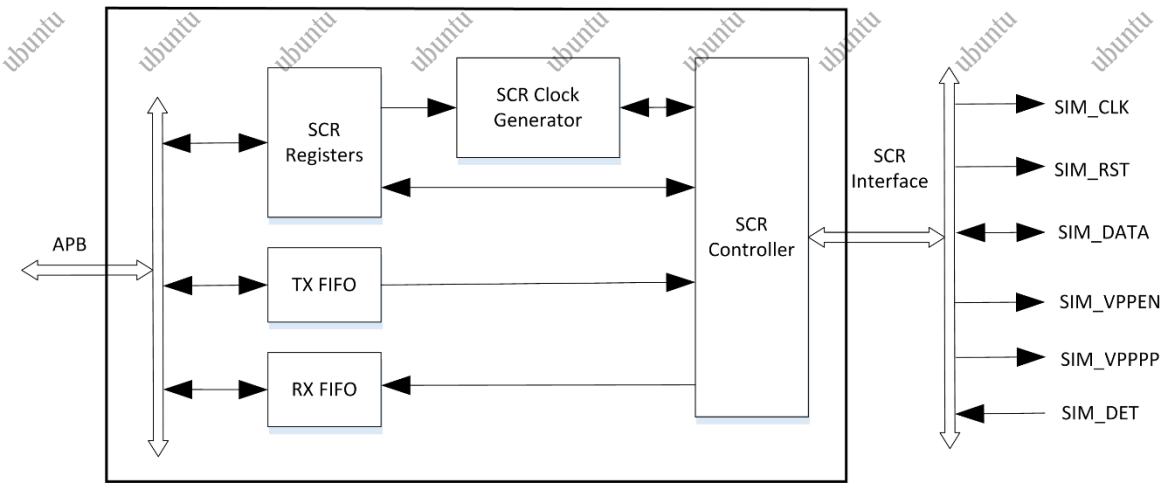
- Supports the ISO/IEC 7816-3:1997(E) and EMV2000 (4.0) Specifications
- Performs functions needed for complete smart card sessions, including:
  - Card activation and deactivation
  - Cold/warm reset
  - Answer to Reset (ATR) response reception
  - Data transfers to and from the card
- Supports adjustable clock rate and bit rate
- Configurable automatic byte repetition
- Supports commonly used communication protocols:
  - T=0 for asynchronous half-duplex character transmission
  - T=1 for asynchronous half-duplex block transmission
- Supports FIFOs for receive and transmit buffers (up to 128 characters) with threshold
- Supports configurable timing functions:
  - Smart card activation time
  - Smart card reset time
  - Guard time
  - Timeout timers
- Supports synchronous and any other non-ISO 7816 and non-EMV cards

7.15.2 Block Diagram

The top diagram of Smart Card Reader is as follows.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 7-112 SCR Block Diagram



7.15.3 Functional Descriptions

7.15.3.1 External Signals

The following table describes the external signals of SCR.

Table 7-48 SCR External Signals

Signal	Description	Type
SIM-PWREN	Smart Card Power Enable	O
SIM-CLK	Smart Card Clock	O
SIM-DATA	Smart Card Data	I/O
SIM-RST	Smart Card Reset	O
SIM-DET	Smart Card Detect	I
SIM-VPPEN	Smart Card Program Voltage Enable	O
SIM-VPPPP	Smart Card Vpp Pause and Program Control	O

7.15.3.2 Clock Sources

The following table describes the clock sources of SCR.

Table 7-49 SCR Clock Sources

Clock Sources	Description
APB1 Bus	SCR clock source. Refer to CCU for details on APB1.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.15.3.3 Timing Diagram

Please refer to ISO/IEC 7816 and EMV2000 Specification.

7.15.3.4 Clock Generator

The clock generator generates the clock signal and the baud clock impulse signal of the Smart Card, which are used in the timing of the SCR.

The clock signal of the SCR is used as the main clock for the smart card. Its frequency can be adjusted by using the Smart Card Clock Divisor (SCCDIV). This value is used to divide the system clock. The SCCLK frequency is given by the following equation:

$$f_{scclk} = \frac{f_{sysclk}}{2*(SCCDIV+1)}$$

f_{scclk} -- Smart Card Clock Frequency

f_{sysclk} -- System Clock (PCLK) Frequency

The baud clock impulse signal is used to transmit and receive serial between the SCR and the Smart Card. The baud rate can be modified by using the Baud Clock Divisor (BAUDDIV). The value is used to divide the system clock. The BUAD rate is given by the following equation:

$$BAUD = \frac{f_{sysclk}}{2*(BAUDDIV+1)}$$

$BAUD$ -- Baud rate of the data stream between Smart Card and Reader.

The duration of one bit, Elementary Time Unit (ETU), is defined in the ISO/IEC 7816-3 specification. During the first answer to reset response after the cold reset, the initial ETU must be equal to 372 Smart Card Clock Cycles.

$$\frac{1}{BAUD} = ETU = \frac{372}{f_{scclk}}$$

In this case, the BAUDDIV should be

$$BAUDDIV = \frac{372*f_{sysclk}}{2*f_{scclk}} - 1 = 372*(SCCDIV+1) - 1.$$

After the ATR is completed, the ETU can be changed according to Smart Card abilities.

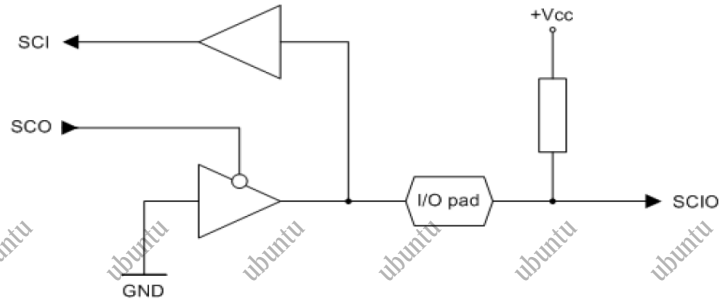
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

$$\frac{1}{BAUD} = ETU = \frac{F}{D} * \frac{1}{f_{scclk}}$$

F is the clock rate conversion integer. **D** is the baud rate adjustment integer.

7.15.3.5 SCR IO Pad Configuration

Figure 7-113 SCR IO Pad Configuration Diagram



7.15.4 Register List

Module Name	Base Address
SCR	0x02505000

Register Name	Offset	Description
SCR_CSR	0x0000	Smart Card Reader Control and Status Register
SCR_INTEN	0x0004	Smart Card Reader Interrupt Enable Register 1
SCR_INTST	0x0008	Smart Card Reader Interrupt Status Register 1
SCR_FCSR	0x000C	Smart Card Reader FIFO Control and Status Register
SCR_FCNT	0x0010	Smart Card Reader RX and TX FIFO Counter Register
SCR_RPT	0x0014	Smart Card Reader RX and TX Repeat Register
SCR_DIV	0x0018	Smart Card Reader Clock and Baud Divisor Register
SCR_LTIM	0x001C	Smart Card Reader Line Time Register
SCR_CTIM	0x0020	Smart Card Reader Character Time Register
SCR_LCTLR	0x0030	Smart Card Reader Line Control Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
SCR_FSM	0x003C	Smart Card Reader FSM Register
SCR_DT	0x0040	Smart Card Reader Debounce Time Register
SCR_FIFO	0x0100	Smart Card Reader RX and TX FIFO Access Point

7.15.5 Register Description

7.15.5.1 0x0000 Smart Card Reader Control and Status Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: SCR_CSR
Bit	Read/Write	Default/Hex	Description
31	R	0x0	SCDET Smart Card Detected This bit is set to '1' when the scdetect input is active at least for a debounce time.
30:25	/	/	/
24	R/W	0x0	SCDETPOL Smart Card Detect Polarity This bit sets the polarity of the scdetect signal. 0: Low Active 1: High Active
23:22	R/W	0x0	Protocol Selection (PTLSEL) 00: T=0 01: T=1, no character repeating and no guard time is used when T=1 protocol is selected. 10: Reserved 11: Reserved
21	R/W	0x0	ATRSTFLUSH ATR Start Flush FIFO When enabled, both FIFOs are flushed before the ATR is started.
20	R/W	0x0	TSRXE TS Receive Enable When set to '1', the TS character (the first ATR character) will be stored in RXFIFO during the card session.
19	R/W	0x0	CLKSTPPOL Clock Stop Polarity The value of the scclk output during the clock stop state.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: SCR_CSR
Bit	**Read/Write**	**Default/Hex**	**Description**
18	R/W	0x0	PECRXE Parity Error Character Receive Enable Enables storage of the characters received with wrong parity in RX FIFO.
17	R/W	0x0	MSBF MSB First When the bit is high, the inverse bit ordering convention (MSB to LSB) is used.
16	R/W	0x0	DATAPOL Data Polarity When the bit is high, the inverse level convention is used (A='1', Z='0').
15:12	/	/	/
11	R/W	0x0	DEACT Deactivation Setting this bit initializes the deactivation sequence. When the deactivation is finished, the DEACT bit is automatically cleared.
10	R/W	0x0	ACT Activation Setting this bit initializes the activation sequence. When the activation is finished, the ACT bit is automatically cleared.
9	R/W	0x0	WARMRST Warm Reset Command. Writing '1' to this bit initializes Warm Reset of the Smart Card. This bit is always read as '0'.
8	R/W	0x0	CLKSTOP Clock Stop When this bit is asserted and the smart card I/O line is in 'Z' state, the SCR core stops the driving of the smart card clock signal after the CLKSTOPDELAY time expires. The smart card clock is re-started immediately after the CLKSTOP signal is de-asserted. New character transmission can be started after CLKSTARTDELAY time. The expiration of both times is signaled by the CLKSTOPRUN bit in the interrupt registers.
7:3	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0000			Register Name: SCR_CSR
Bit	**Read/Write**	**Default/Hex**	**Description**
2	R/W	0x0	GINTEN Global Interrupt Enable When the bit is high, the IRQ output assertion is enabled.
1	R/W	0x0	RXEN Receiving Enable When enabled, the characters sent by the Smart Card are received by the UART and stored in RX FIFO. The reception is internally disabled while a transmission is in progress.
0	R/W	0x0	TXEN Transmission Enable. When enabled, the characters are read from TX FIFO and transmitted through UART to the Smart Card.

7.15.5.2 0x0004 Smart Card Reader Interrupt Enable Register (Default Value: 0x0000_0000)

Offset: 0x0004			Register Name: SCR_INTEN
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	/	/	/
23	R/W	0x0	SCDEA Smart Card Deactivation Interrupt Enable
22	R/W	0x0	SCACT Smart Card Activation Interrupt Enable
21	R/W	0x0	SCINS Smart Card Inserted Interrupt Enable
20	R/W	0x0	SCREM Smart Card Removed Interrupt Enable
19	R/W	0x0	ATRDONE ATR Done Interrupt Enable
18	R/W	0x0	ATRFAIL ATR Fail Interrupt Enable
17	R/W	0x0	C2CFULL Two Consecutive Characters Limit Interrupt Enable
16	R/W	0x0	CLKSTOPRUN Smart Card Clock Stop/Run Interrupt Enable

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0004			Register Name: SCR_INTEN
Bit	**Read/Write**	**Default/Hex**	**Description**
15:13	/	/	/
12	R/W	0x0	RXPERR RX Parity Error Interrupt Enable
11	R/W	0x0	RXDONE RX Done Interrupt Enable
10	R/W	0x0	RXFIFOTHD RX FIFO Threshold Interrupt Enable
9	R/W	0x0	RXFIFOFULL RX FIFO Full Interrupt Enable
8:5	/	/	/
4	R/W	0x0	TXPERR TX Parity Error Interrupt Enable
3	R/W	0x0	TXDONE TX Done Interrupt Enable
2	R/W	0x0	TXFIFOTHD TX FIFO Threshold Interrupt Enable
1	R/W	0x0	TXFIFOEMPTY TX FIFO Empty Interrupt Enable
0	R/W	0x0	TXFIFODONE TX FIFO Done Interrupt Enable

7.15.5.3 0x0008 Smart Card Reader Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0008			Register Name: SCR_INTST
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	/	/	/
23	R/W1C	0x0	SCDEA Smart Card Deactivation Interrupt When enabled, this interrupt is asserted after the Smart Card deactivation sequence is completed. Write '1' to clear.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0008			Register Name: SCR_INTST
Bit	**Read/Write**	**Default/Hex**	**Description**
22	R/W1C	0x0	SCACT Smart Card Activation Interrupt When enabled, this interrupt is asserted after the Smart Card activation sequence is completed. Write '1' to clear.
21	R/W1C	0x0	SCINS Smart Card Inserted Interrupt When enabled, this interrupt is asserted after the smart card insertion. Write '1' to clear.
20	R/W1C	0x0	SCREM Smart Card Removed Interrupt When enabled, this interrupt is asserted after the smart card removal. Write '1' to clear.
19	R/W1C	0x0	ATRDONE ATR Done Interrupt When enabled, this interrupt is asserted after the ATR sequence is completed. Write '1' to clear.
18	R/W1C	0x0	ATRFAIL ATR Fail Interrupt When enabled, this interrupt is asserted if the ATR sequence fails. Write '1' to clear.
17	R/W1C	0x0	C2CFULL Two Consecutive Characters Limit Interrupt When enabled, this interrupt is asserted if the time between two consecutive characters, and transmitted between the Smart Card and the Reader in both directions, is equal to the Two Characters Delay Limit. The C2CFULL interrupt is internally enabled from the ATR start to the deactivation or ATR restart initialization. It is recommended to use this counter to detect unresponsive Smart Cards. Write '1' to clear.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0008			Register Name: SCR_INTST
Bit	**Read/Write**	**Default/Hex**	**Description**
16	R/W1C	0x0	CLKSTOPRUN Smart Card Clock Stop/Run Interrupt When enabled, this interrupt is asserted in two cases: 1. When the smart card clock is stopped. 2. When the new character can be started after the clock restart. To distinguish between the two interrupt cases, we recommend reading the CLKSTOP bit in the SCR_CTRL1 register. Write '1' to clear.
15:13	/	/	/
12	R/W1C	0x0	RXPERR RX Parity Error Interrupt When enabled, this interrupt is asserted after the character with wrong parity was received when the number of the repeated receptions exceeds RXREPEAT value or T=1 protocol is used. Write '1' to clear.
11	R/W1C	0x0	RXDONE RX Done Interrupt When enabled, this interrupt is asserted after a character was received from the Smart Card. Write '1' to clear.
10	R/W1C	0x0	RXFIFOTHD RX FIFO Threshold Interrupt When enabled, this interrupt is asserted if the number of bytes in RX FIFO is equal or exceeds the RX FIFO threshold. Write '1' to clear.
9	R/W1C	0x0	RXFIFOFULL RX FIFO Full Interrupt When enabled, this interrupt is asserted if the RX FIFO is filled up. Write '1' to clear.
8:5	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0008			Register Name: SCR_INTST
Bit	**Read/Write**	**Default/Hex**	**Description**
4	R/W1C	0x0	TXPERR TX Parity Error Interrupt When enabled, this interrupt is asserted if the Smart Card signals wrong character parity during the guard time after the character transmission was repeated TXREPEAT times or T=1 protocol is used. Write '1' to clear.
3	R/W1C	0x0	TXDONE TX Done Interrupt When enabled, this interrupt is asserted after one character was transmitted to the smart card. Write '1' to clear.
2	R/W1C	0x0	TXFIFOTHD TX FIFO Threshold Interrupt When enabled, this interrupt is asserted if the number of bytes in TX FIFO is equal or less than the TX FIFO threshold. Write '1' to clear.
1	R/W1C	0x0	TXFIFOEMPTY TX FIFO Empty Interrupt When enabled, this interrupt is asserted if the TX FIFO is empty. Write '1' to clear.
0	R/W1C	0x0	TXFIFODONE TX FIFO Done Interrupt When enabled, this interrupt is asserted after all bytes from TX FIFO are transferred to the Smart Card. Write '1' to clear.

7.15.5.4 0x000C Smart Card Reader FIFO Control and Status Register (Default Value: 0x0000_0101)

Offset: 0x000C			Register Name: SCR_FCSR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10	R/W	0x0	RXFIFOFLUSH Flush RX FIFO RX FIFO is flushed when writing '1' to this bit.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x000C			Register Name: SCR_FCSR
Bit	**Read/Write**	**Default/Hex**	**Description**
9	R	0x0	RXFIFOFULL RX FIFO Full
8	R	0x1	RXFIFOEMPTY RX FIFO Empty
7:3	/	/	/
2	R/W	0x0	TXFIFOFLUSH Flush TX FIFO TX FIFO is flushed when writing '1' to this bit.
1	R	0x0	TXFIFOFULL TX FIFO Full
0	R	0x1	TXFIFOEMPTY TX FIFO Empty

7.15.5.5 0x0010 Smart Card Reader FIFO Count Register (Default Value: 0x0000_0000)

Offset: 0x0010			Register Name: SCR_FIFOCNT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	R/W	0x0	RXFTH RX FIFO Threshold These bits set the interrupt threshold of RX FIFO. The interrupt is asserted when the number of the received bytes is equal to or more than the threshold.
23:16	R/W	0x0	TXFTH TX FIFO Threshold These bits set the interrupt threshold of TX FIFO. The interrupt is asserted when the number of bytes in TX FIFO is equal to or less than the threshold.
15:8	R	0x0	RXFCNT RX FIFO Counter These bits provide the number of bytes stored in the RXFIFO.
7:0	R	0x0	TXFCNT TX FIFO Counter These bits provide the number of bytes stored in the TXFIFO.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.15.5.6 0x0014 Smart Card Reader Repeat Control Register (Default Value: 0x0000_0000)

Offset: 0x0014			Register Name: SCR_REPEAT
Bit	**Read/Write**	**Default/Hex**	**Description**
15:8	/	/	/
7:4	R/W	0x0	RXRPT RX Repeat This is a 4-bit register that specifies the number of the re-transmission characters after the wrong parity was detected. The re-transmission of the character is requested by the error signal during the guard time.
3:0	R/W	0x0	TXRPT TX Repeat This is a 4-bit register that specifies the number of the re-transmission characters after the Smart Card signals the wrong parity during the guard time.

7.15.5.7 0x0018 Smart Card Reader Clock Divisor Register (Default Value: 0x0000_0000)

Offset: 0x0018			Register Name: SCR_CLKDIV
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	R/W	0x0	BAUDDIV Baud Clock Divisor This 16-bit register defines the divisor value used to generate the Baud Clock from the system clock. $BAUD = \frac{f_{sysclk}}{2*(BAUDDIV+1)}$
15:0	R/W	0x0	SCCDIV Smart Card Clock Divisor This 16-bit register defines the divisor value used to generate the Smart Card Clock from the system clock. $f_{scclk} = \frac{f_{sysclk}}{2*(SCCDIV+1)}$ f_{scclk} is the frequency of Smart Card Clock Signal. f_{sysclk} is the frequency of APB Clock.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

7.15.5.8 0x001C Smart Card Reader Line Time Register (Default Value: 0x0000_0000)

Offset: 0x001C			Register Name: SCR_LTIM
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	/	/	/
23:16	R/W	0x0	ATR ATR Start Limit This 16-bit register defines the maximum time between the rising edge of the *scrstn* signal and the start of the ATR response. ATR Start Limit = 256* ATR* T_{scclk} .
15:8	R/W	0x0	RST Reset Duration This 16-bit register sets the duration of the Smart Card reset sequence. This value is the same for the cold and warm reset. Cold/Warm Reset Duration = 256* RST* T_{scclk} .
7:0	R/W	0x0	ACT Activation/Deactivation Time This 16-bit register sets the duration of each part of the activation and deactivation sequence. Activation/Deactivation Duration = 256* ACT * T_{scclk} . $T_{scclk} = \frac{1}{f_{scclk}}$ is Smart Card Clock Cycle.

7.15.5.9 0x0020 Smart Card Reader Character Time Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: SCR_CTIM
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	R/W	0x0	CHARLIMIT Character Limit This 16-bit register sets the maximum time between the leading edges of two consecutive characters. The value is ETUs.
15:8	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0020			Register Name: SCR_CTIM
Bit	Read/Write	Default/Hex	Description
7:0	R/W	0x0	GUARDTIME Character Guard time This 8-bit register sets a delay at the end of each character transmitted from the Smart Card Reader to the Smart Card. The value is in ETUs. The parity error is included during the guard time.

7.15.5.10 0x0030 Smart Card Reader Line Control Register (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: SCR_PAD
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	DSCVPPPP Direct Smart Card Vpp Pause/Prog It provides the direct access to the SCVPPPP output.
6	R/W	0x0	DSCVPPEN Direct Smart Card Vpp Enable It provides the direct access to the SCVPPEN output.
5	R/W	0x0	AUTOADEAVPP Automatic Vpp Handling When high, it enables automatic handling of the DSVPPEN and DSVPPPP signals during activation and deactivation sequence.
4	R/W	0x0	DSCVCC Direct Smart Card VCC When DIRACCPADS='1', the DSCVCC bit provides the direct access to the SCVCC pad.
3	R/W	0x0	DSCRST Direct Smart Card Clock When DIRACCPADS='1', the DSCRST bit provides the direct access to the SCRST pad.
2	R/W	0x0	DSCCLK Direct Smart Card Clock When DIRACCPADS='1', the DSCCLK bit provides the direct access to the SCCLK pad.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0030			Register Name: SCR_PAD
Bit	**Read/Write**	**Default/Hex**	**Description**
1	R/W	0x0	DSCIO Direct Smart Card Input/Output When DIRACCPADS='1', the DSCIO bit provides the direct access to the SCIO pad.
0	R/W	0x0	DIRACCPADS Direct Access to Smart Card Pads When high, it disables a serial interface functionality and enables direct control of the smart card pads by using following bit[1:4].

7.15.5.11 0x003C Smart Card Reader FSM Register (Default Value: 0x0000_0000)

Offset: 0x003C			Register Name: SCR_FSM
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	R	0x0	ATR_STRUCTURE_FSM
23:16	R	0x0	ATR_FSM
15:8	R	0x0	ACT_FSM
7:0	R	0x0	SCR_FSM

7.15.5.12 0x0040 Smart Card Reader Debounce Time Register (Default Value: 0x0000_03FF)

Offset: 0x0040			Register Name: SCR_DT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x3ff	SCR_DEBOUNCE_TIME Set the debounce time value for card insert detection. The time unit is the cycle of SCCLK.

7.15.5.13 0x0100 Smart Card Reader FIFO Data Register (Default Value: 0x0000_0000)

Offset: 0x0100			Register Name: SCR_FIFO
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0100			Register Name: SCR_FIFO
Bit	**Read/Write**	**Default/Hex**	**Description**
7:0	R/W	0x0	FIFO_DATA This 8-bit register provides access to the RX and TX FIFO buffers. The TX FIFO is accessed during the APB write-transmission. The RX FIFO is accessed during the APB read-transmission.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Contents

8 Security System 1047
8.1 Crypto Engine 1047
8.1.1 Overview 1047
8.1.2 Block Diagram 1048
8.1.3 Functional Descriptions 1049
8.1.4 Programming Guidelines 1068
8.1.5 Register List 1071
8.1.6 Register Description 1072
8.2 Security ID 1076

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figures

Figure 8-1 CE Block Diagram 1048
Figure 8-2 DES Encryption and Decryption 1049
Figure 8-3 3DES Encryption and Decryption of a 3-key Operation and a 2-key Operation 1050
Figure 8-4 ECB Mode Encryption and Decryption 1051
Figure 8-5 CBC Mode Encryption and Decryption 1052
Figure 8-6 CTR Mode Encryption and Decryption 1053
Figure 8-7 CFB Mode Encryption and Decryption 1054
Figure 8-8 OFB Mode Encryption and Decryption 1055
Figure 8-9 CTS Mode Encryption and Decryption 1056
Figure 8-10 Structure of Task Descriptor Chaining 1057
Figure 8-11 Word Address of Message 1064
Figure 8-12 Byte Order 1064
Figure 8-13 Bit Order 1064
Figure 8-14 The Storage Method of 32-bit IV 1065
Figure 8-15 The Storage Method of 64-bit IV 1065
Figure 8-16 Task Request Process 1068

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

8 Security System

8.1 Crypto Engine

8.1.1 Overview

The Crypto Engine (CE) module is one encryption/decryption algorithm accelerator. It supports kinds of symmetric, asymmetric, Hash, and RNG algorithms.

The symmetric algorithm supports data encryption and decryption by following the data encryption standard (DES), 3DES, or advanced encryption standard (AES) algorithms. It can encrypt or decrypt a large amount of data effectively.

The Rivest-Shamir-Adleman (RSA) asymmetric algorithm is used for data encryption/decryption and digital signature verification. It is a public key encryption/decryption algorithm implemented through the modular exponentiation operation.

The Hash algorithm supports data integrity authentication and digital signature. The hash supports MD5, SHA1, SHA224, SHA256, SHA384, SHA512, HMAC-SHA1, and HMAC-SHA256 algorithms.

The RNG algorithm can generate true-random numbers and pseudo-random numbers.

The R329 has two separate CE controllers: one CE is for secure world, the other CE is for non-secure world. The software interface of the controller is simple, only setting interrupt control, task description address, and load tag. The algorithm control information is written in memory by task descriptor, then the CE automatically reads it when executing a request. It supports parallel execution of 4 channels and has an internal DMA controller to transfer data between CE and memory.

The CE has the following features:

- Symmetrical algorithm: AES, DES, 3DES
- Hash algorithm: MD5, SHA1, SHA224, SHA256, SHA384, SHA512, HMAC-SHA1, HMAC-SHA256
- Asymmetrical algorithm: RSA512/1024/2048-bit
- 160-bit hardware PRNG with 175-bit seed
- 256-bit hardware TRNG
- Electronic codebook (ECB), cipher block chaining (CBC), counter (CTR), cipher text stealing (CTS), 128-output feedback (OFB), 1-/8-/64-/128-cipher feedback (CFB) modes for AES algorithm
- ECB, CBC, CTR modes for DES/3DES algorithm
- 128-, 192-, 256-bit key size for AES algorithms
- 16-, 32-, 64-, 128-bit wide size for AES CTR mode

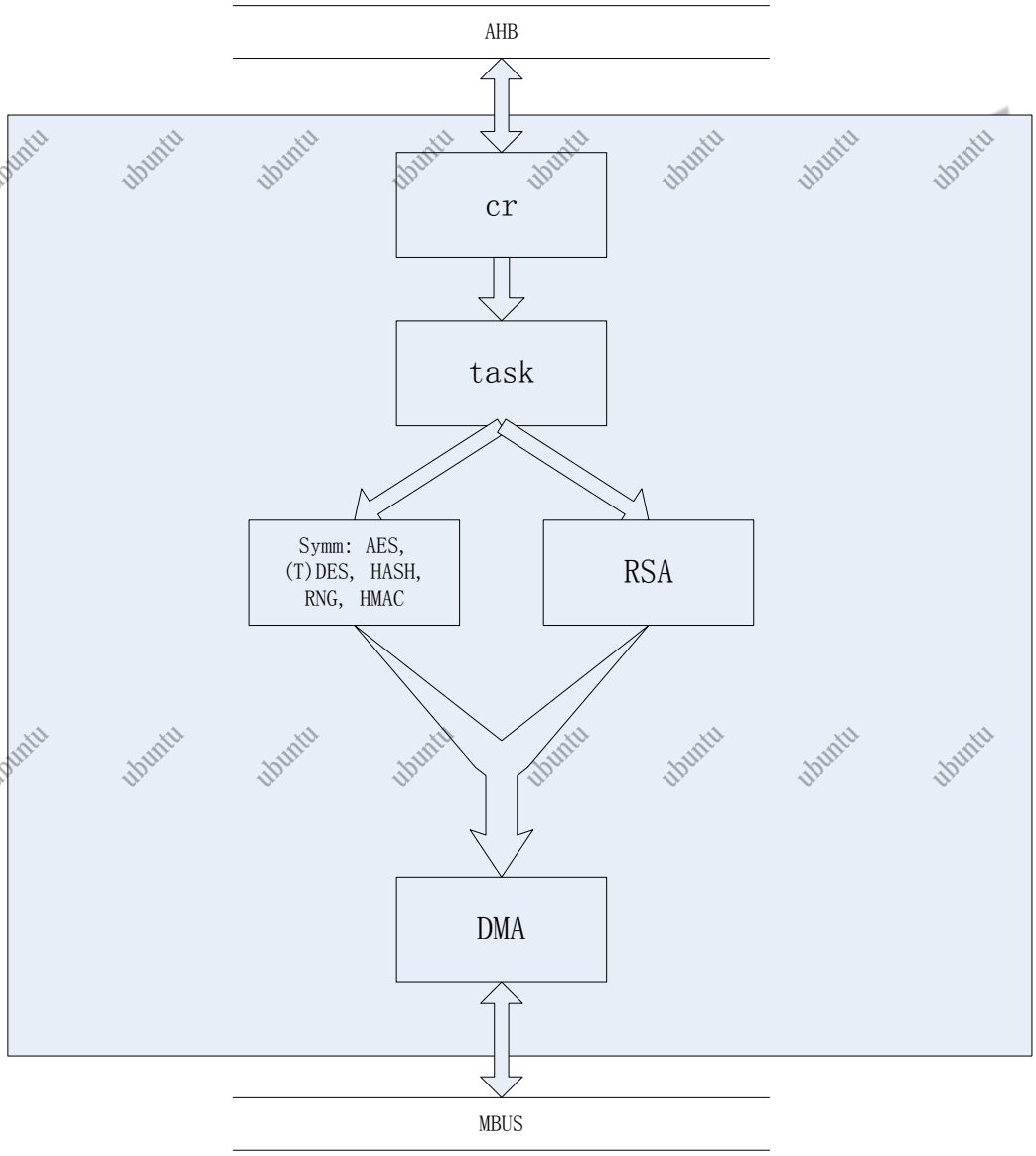
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

- 16-, 32-, 64-bit wide size for DES/3DES CTR mode
- One or more blocks mode for MD5/SHA1/SHA224/SHA256/SHA384/SHA512
- Internal DMA controller for data transfer with memory
- Supports secure and non-secure interfaces respectively

8.1.2 Block Diagram

The following figure shows the block diagram of Crypto Engine.

Figure 8-1 CE Block Diagram



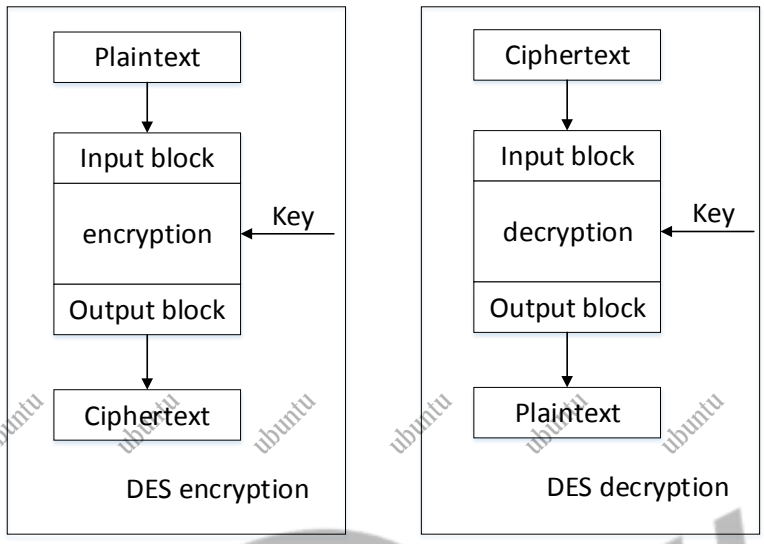
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

8.1.3 Functional Descriptions

8.1.3.1 DES Algorithm

Figure 8-2 shows the DES encryption and decryption operation.

Figure 8-2 DES Encryption and Decryption

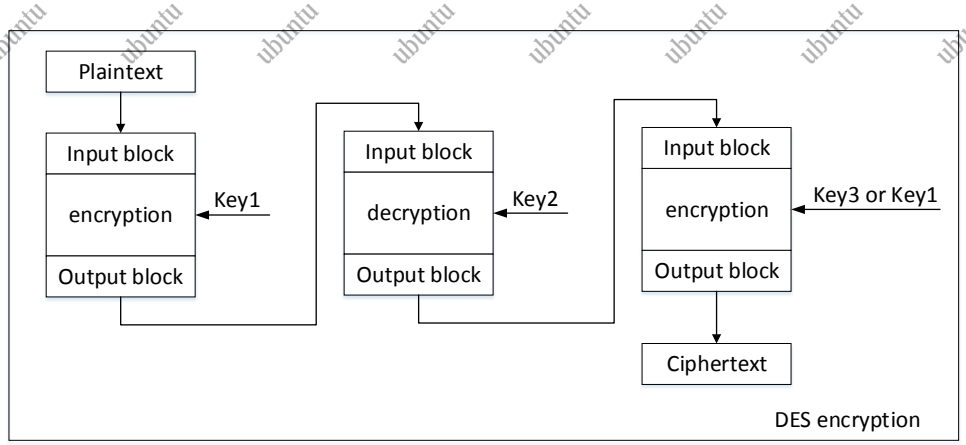


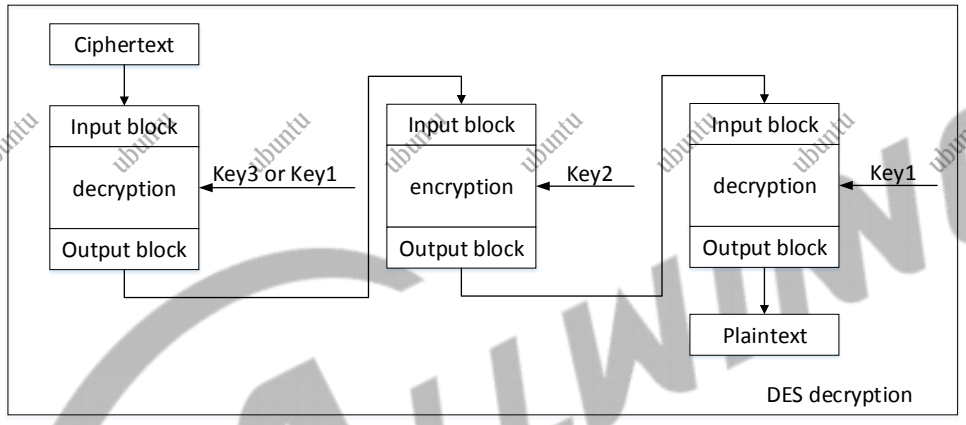
8.1.3.2 3DES Algorithm

The 3DES algorithm supports both 3-key and 2-key operations. A 2-key operation can be regarded as a simplified 3-key operation. To be specific, key 3 is represented by key 1 in a 2-key operation. Figure 8-3 shows the 3DES encryption and decryption operation of a 3-key operation and a 2-key operation.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 8-3 3DES Encryption and Decryption of a 3-key Operation and a 2-key Operation





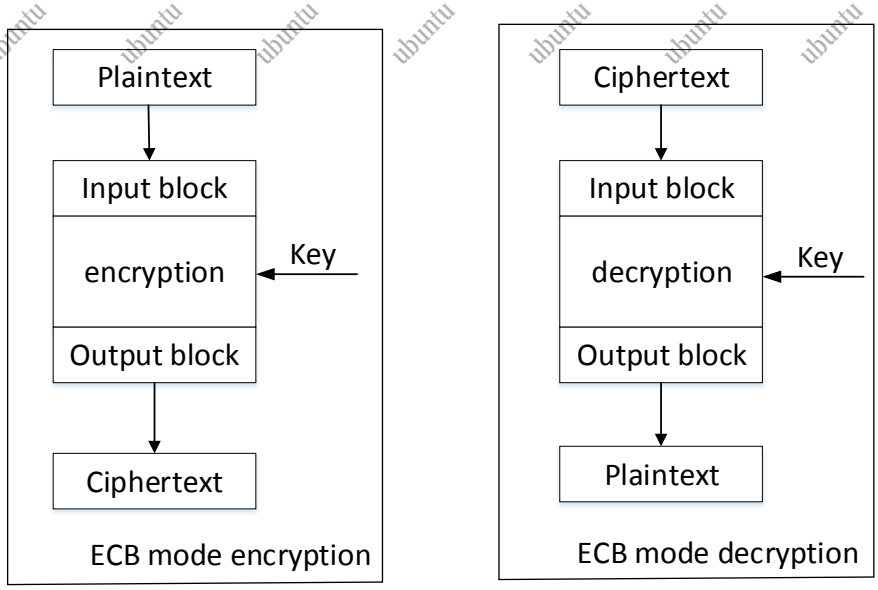
8.1.3.3 ECB Mode

The ECB mode is a confidentiality mode that features, for a given key, the assignment of a fixed ciphertext block to each plaintext block, analogous to the assignment of code words in a codebook.

In ECB mode, encryption and decryption algorithms are directly applied to the block data. The operation of each block is independent, so the plaintext encryption and ciphertext decryption can be performed concurrently.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 8-4 ECB Mode Encryption and Decryption

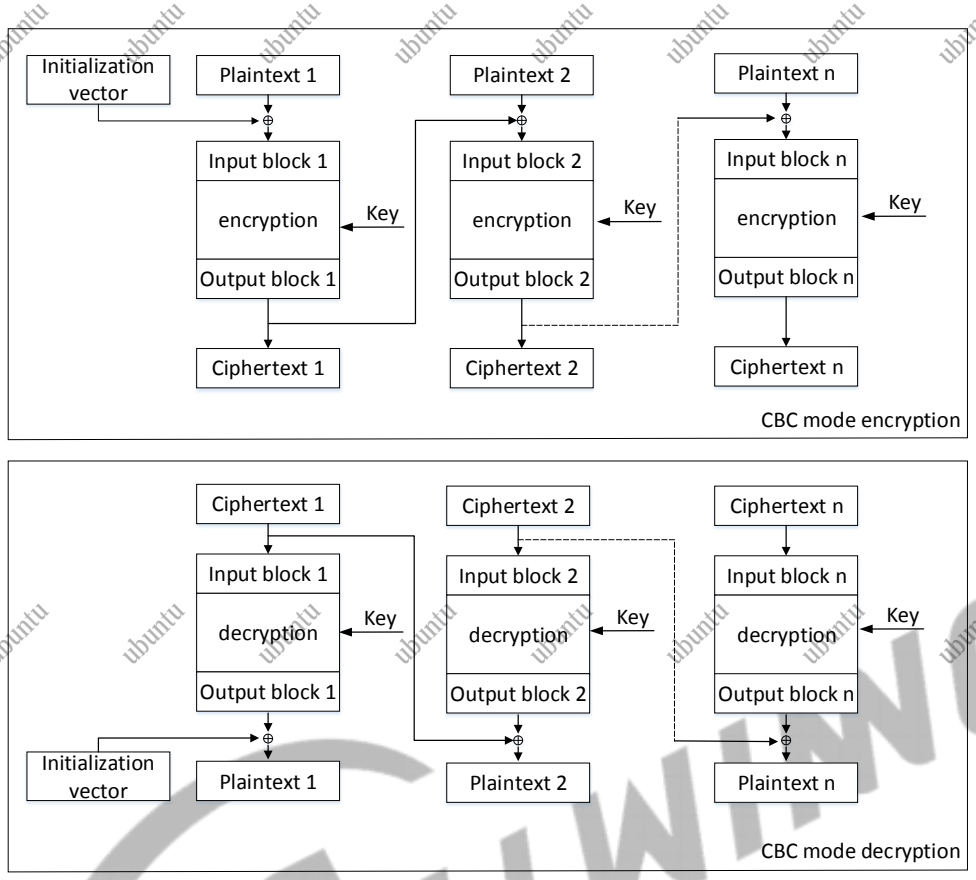


8.1.3.4 CBC Mode

The CBC mode is a confidentiality mode whose encryption process features the combining of the plaintext blocks with the previous ciphertext blocks. The CBC mode requires an initialization vector (IV) to combine with the first plaintext block. The encryption process of each plaintext block is related to the block processing result of the previous ciphertext blocks, so encryption operations cannot be concurrently performed in CBC mode. The decryption operation is independent of output plain text of the previous block, so decryption operations can be performed concurrently.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 8-5 CBC Mode Encryption and Decryption

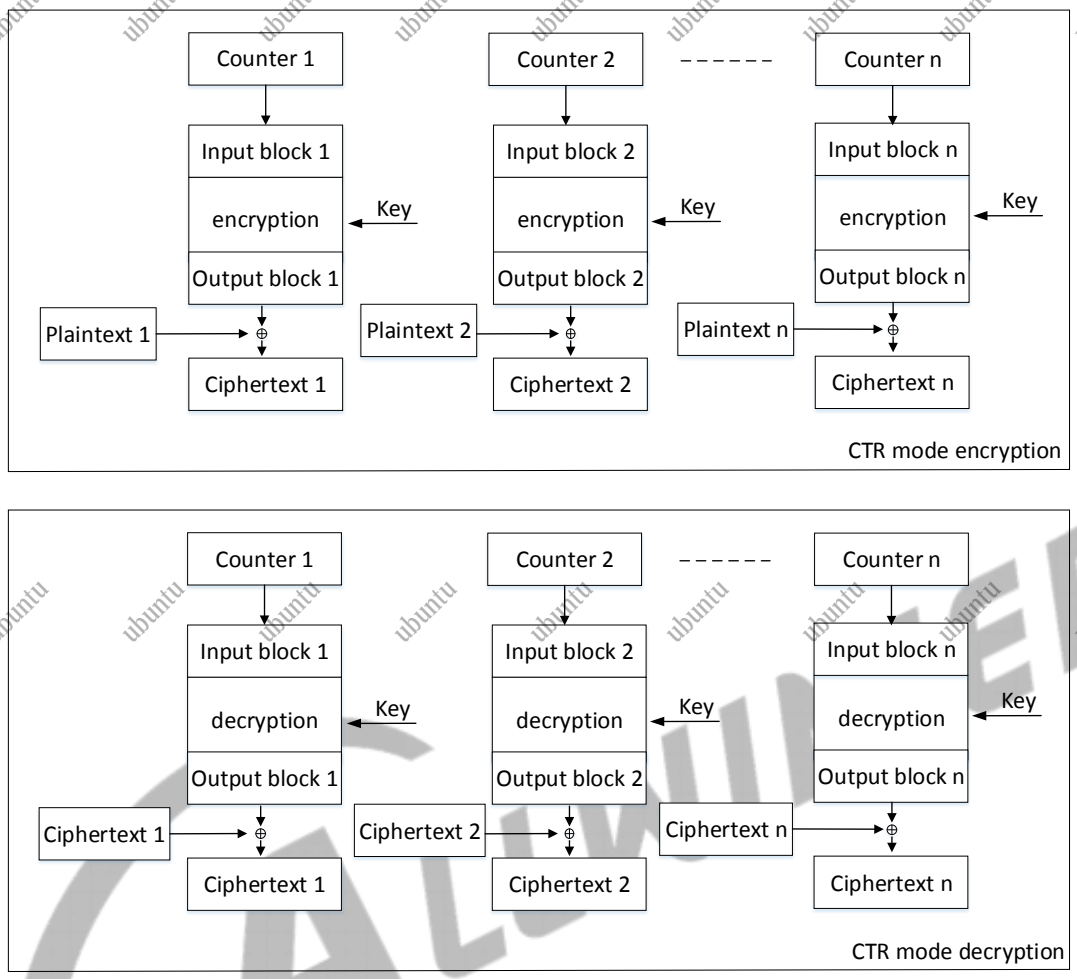


8.1.3.5 CTR Mode

The CTR mode is a confidentiality mode that features the application of the forward cipher to a set of input blocks, called counters, to produce a sequence of output blocks that are exclusive-ORed with the plaintext to produce the ciphertext, and vice versa. All of the counters must be distinct.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 8-6 CTR Mode Encryption and Decryption



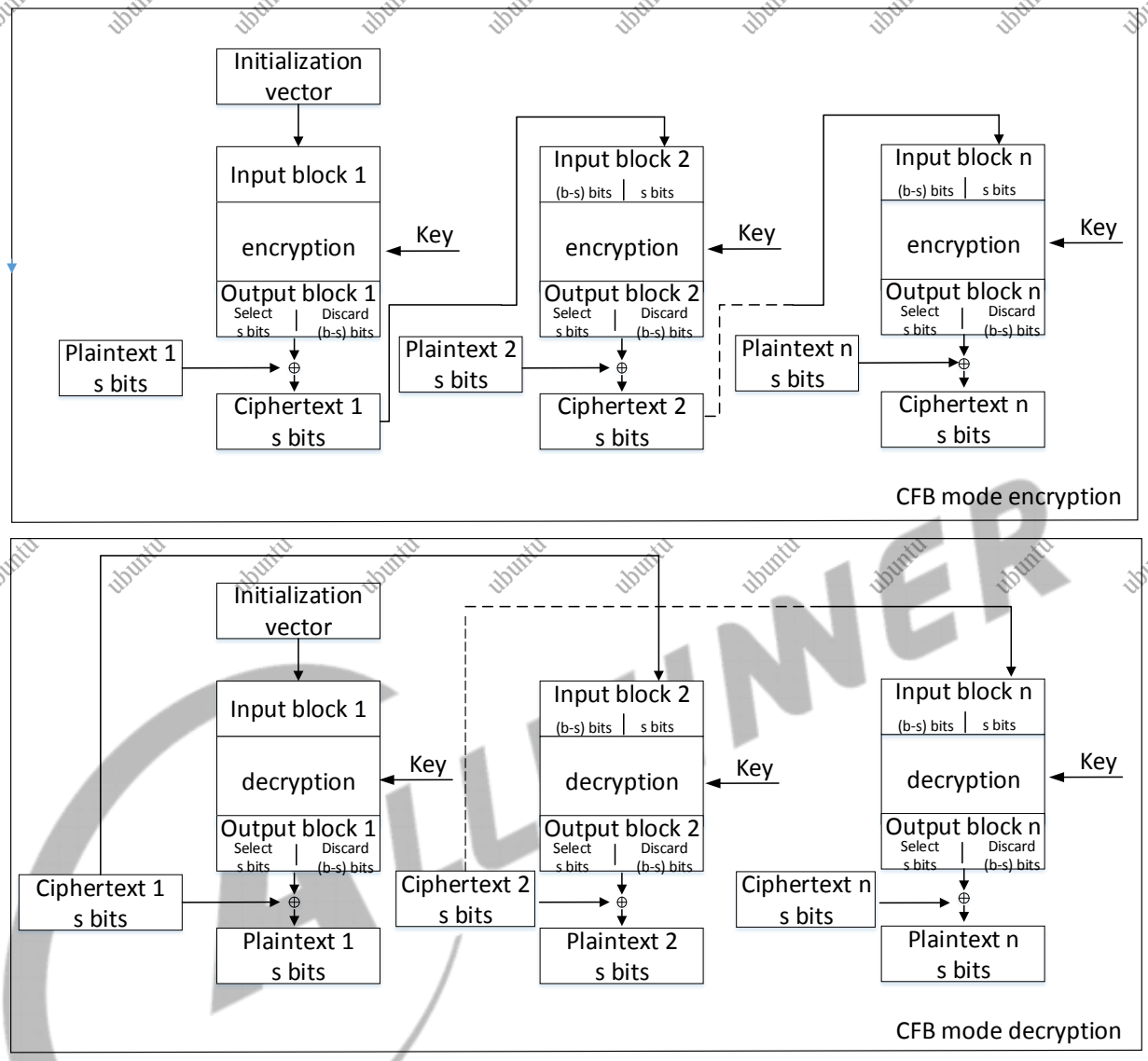
8.1.3.6 CFB Mode

The CFB mode is a confidentiality mode that features the feedback of successive ciphertext segments into the input blocks of the forward cipher to generate output blocks that are exclusive-ORed with the plaintext to produce the ciphertext, and vice versa. The CFB mode requires an IV as the initial input block, and the forward cipher operation is applied to the IV to produce the first output block. The first ciphertext segment is produced by exclusive-ORing the first plaintext segment with the *s* most significant bits of the first output block. The value of s is 1 bit, 8 bits, 64 bits, or 128 bits.

The following figure shows the s-bit CFB mode of the AES algorithms.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 8-7 CFB Mode Encryption and Decryption

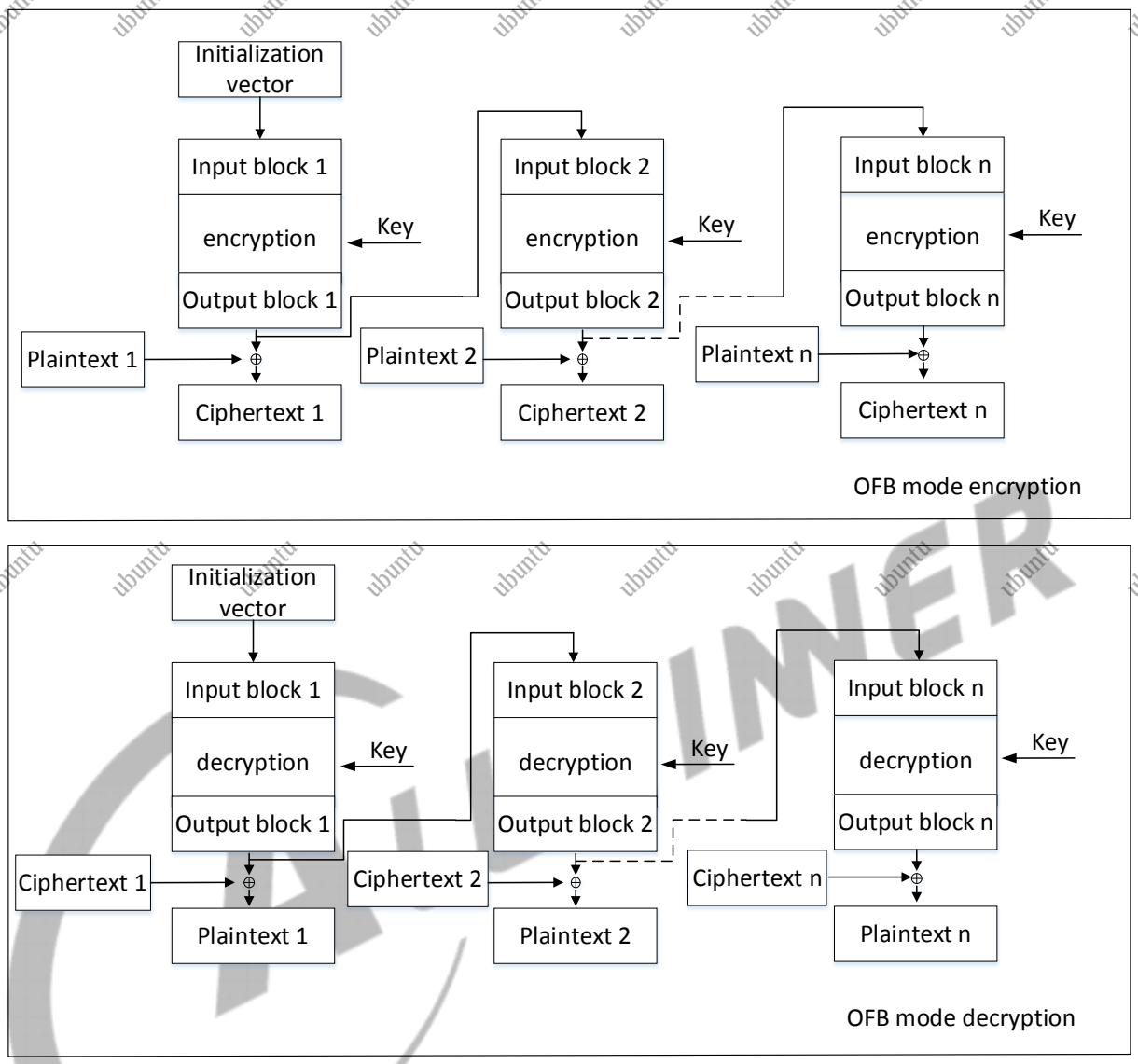


8.1.3.7 OFB Mode

The OFB mode is a confidentiality mode that features the iteration of the forward cipher on an IV to generate a sequence of output blocks that are exclusive-ORed with the plaintext to produce the ciphertext, and vice versa. If a same key is used, different IVs must be used to ensure operation security.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 8-8 OFB Mode Encryption and Decryption

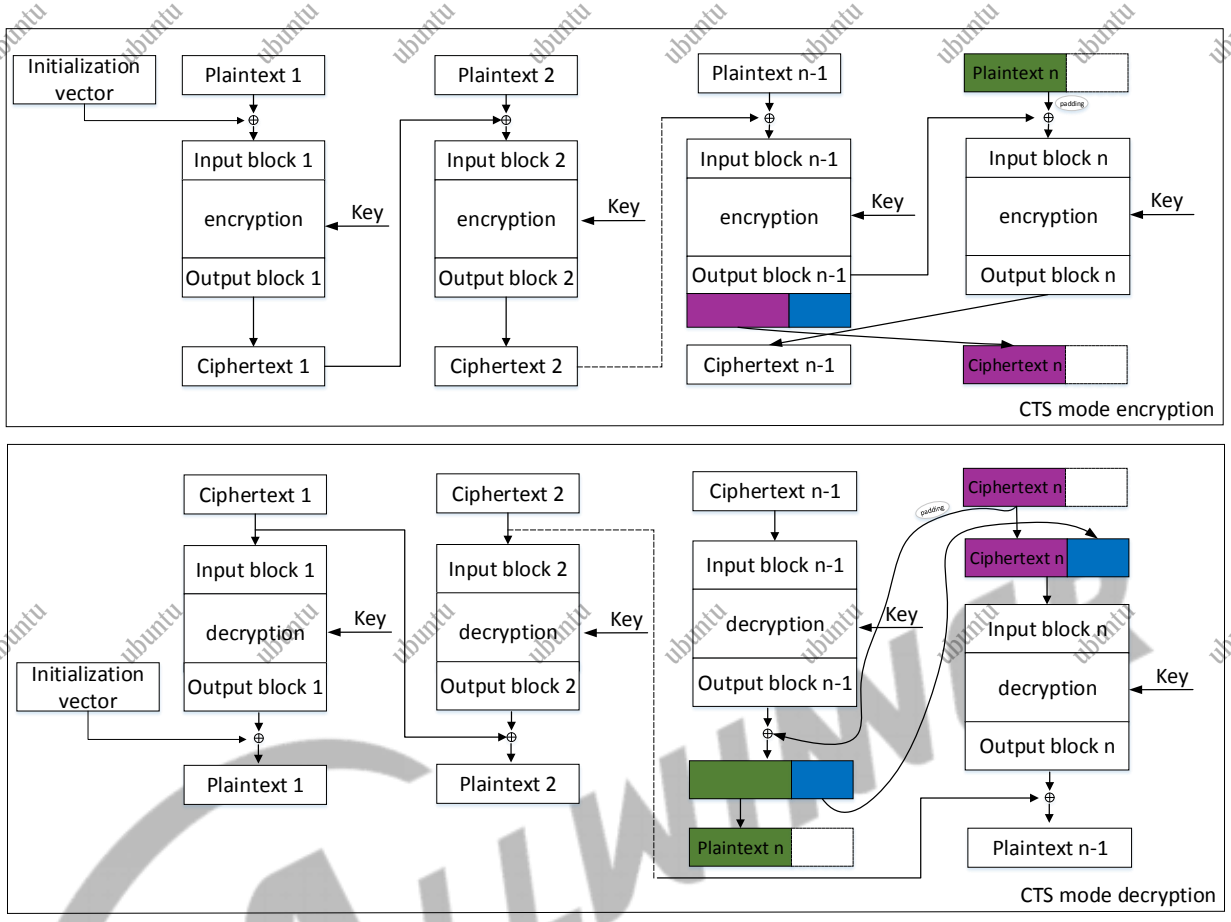


8.1.3.8 CTS Mode

The CTS mode is a confidentiality mode that accepts any plaintext input whose bit length is greater than or equal to the block size but not necessarily a multiple of the block size. Below are the diagrams for CTS encryption and decryption.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 8-9 CTS Mode Encryption and Decryption



8.1.3.9 HASH Algorithm

The hash algorithms support MD5, SHA1, SHA224, SHA256, SHA384, SHA512, HMAC-SHA1, and HMAC-SHA256. All algorithms are iterative, one-way hash functions that can process a message to produce a condensed representation called a *message digest*. When a message is received, the *message digest* can be used to verify whether the data has changed, that is, to verify its integrity.

The hash algorithm of R329 supports block-aligned total length of the input data (padded by software), that is, a multiple of 64 bytes. The message length after padding by software is used as the configured data length for the hash algorithm.

8.1.3.10 RSA Algorithm

The RSA is a public key encryption/decryption algorithm implemented through the modular exponentiation operation.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

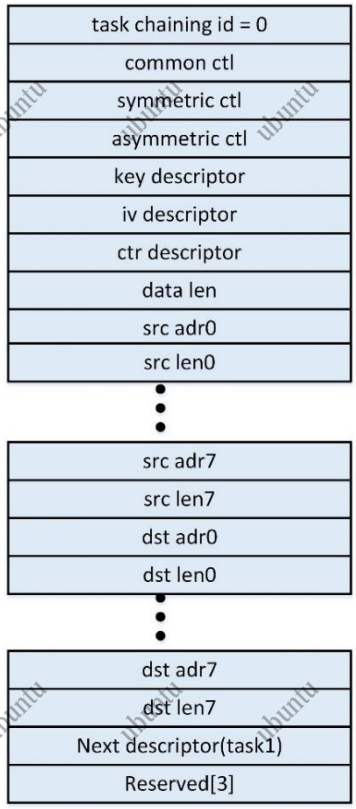
The ciphertext is obtained as follows: $C = M^E \bmod N$. The plaintext is obtained as follows: $M = C^D \bmod N$.

M indicates the plaintext, C indicates the ciphertext, (N, E) indicates the public key, and (N, D) indicates the private key.

8.1.3.11 Task Descriptor

The software makes request through task descriptor, including algorithm type, algorithm mode, key address, source/destination address and size, and so on. The structure of the task descriptor is as follows.

Figure 8-10 Structure of Task Descriptor Chaining



The bit definitions of the task descriptor are as follows.

Task ID

Bit	Read/Write	Default/Hex	Description
31:4	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Bit	Read/Write	Default/Hex	Description
3:0	R/W	0x0	CHN Task channel ID Indicates which channel the task is running on. It supports 0 to 3.

Common Control

Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	Interrupt enable (IE) for the current task 0: disable interrupt 1: enable interrupt Represents whether an interrupt signal is generated when the task chain ends at the end of this task. When the last task in a task chain ends, the operation of the task chain will end normally; if a task fails in the middle, the task chain will be aborted abnormally. And it is determined whether to generate an interrupt signal according to the IE configuration of the current task when the current task ends or aborts. Therefore, if you want to use interrupts, it is recommended that not only the IE of the last task of each task chain is configured to 1 to generate the end interrupt of the task chain, but also the IEs of other tasks in this task chain are also configured to 1. The purpose is to generate an interrupt signal once an abnormal error occurs in these tasks and the interrupt is aborted.
30:17	/	/	/
16	R/W	0x0	IV mode IV mode for SHA1/SHA224/SHA256/SHA384/SHA512/MD5 or constants 0: use initial constants defined in FIPS-180 1: use input iv
15	R/W	0x0	Last HMAC plaintext 0: not the last HMAC plaintext package 1: the last HMAC plaintext package
14:9	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Bit	Read/Write	Default/Hex	Description
8	R/W	0x0	OP DIR Algorithm Operation Direction 0: Encryption 1: Decryption Configure according to the requirements of encryption or decryption.
7	/	/	/
6:0	R/W	0x0	Algorithm Type 0x0: AES 0x1: DES 0x2: Triple DES (3DES) 0x10: MD5 0x11: SHA-1 0x12: SHA-224 0x13: SHA-256 0x14: SHA-384 0x15: SHA-512 0x16: HMAC-SHA1 0x17: HMAC-SHA256 0x20: RSA 0x30: TRNG 0x31: PRNG Others: reserved

Symmetric Control

Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:20	R/W	0x0	KEY_SELECT key select for AES 0000: Select input CE_KEYx (Normal Mode) 0001: Select {SSK} 0010: Select {HUK} 0011: Select {RSSK} 0100-0111: Reserved 1000-1111: Select internal Key n (n from 0 to 7)

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Bit	Read/Write	Default/Hex	Description
19:18	R/W	0x0	CFB_WIDTH AES-CFB width 00: CFB1 01: CFB8 10: CFB64 11: CFB128
17	R/W	0x0	PRNG_LD Load new 15 bits key into linear feedback shift register (LFSR) for PRNG. When the algorithm type is PRNG, it is necessary to post-process the random number output by PRNG through the linear shift operation to generate the operand. When the PRNG_LD is configured to 1, use iv_addr[14:0] as the input number for linear shift operation, and do XOR operation between the data and the random number output by PRNG to generate the post-processing result of further operation.
16	R/W	0x0	CTS_LPKG AES CTS last package flag When set to '1', it means this is the last package for AES-CTS mode (the size of the last package is larger than 128 bits).
15:12	/	/	/
11:8	R/W	0x0	ALGORITHM_MODE CE algorithm mode 0000: Electronic Code Book (ECB) mode 0001: Cipher Block Chaining (CBC) mode 0010: Counter (CTR) mode 0011: Cipher Text Stealing (CTS) mode 0100: Output Feedback (OFB) mode 0101: Cipher Feedback (CFB) mode Other: reserved
7:4	/	/	/
3:2	R/W	0x0	CTR WIDTH Counter width for CTR mode 00: 16-bit Counter 01: 32-bit Counter 10: 64-bit Counter 11: 128-bit Counter

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Bit	Read/Write	Default/Hex	Description
1:0	R/W	0x0	AES KEY SIZE 00: 128-bit 01: 192-bit 10: 256-bit 11: Reserved

Asymmetric Control

Bit	Read/Write	Default/Hex	Description
31	/	/	/
30:28	R/W	0x0	RSA Pubic Modulus Width 000: 512-bit 001: 1024-bit 010: 2048-bit Other: reserved
27:0	/	/	/

Key Descriptor

Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	Key Address The address of KEY that needs to be stored.

IV Descriptor

Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	IV Address The address of IV that needs to be stored.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Counter Descriptor

Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	CTR Data Output Address The address of CTR data output that needs to be stored.

Data Length

Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	Data Length Configure the data length of the corresponding segment. The data length size needs to be consistent with dst_data_length (destination data length 0 +... + destination data length 7). The data length field in the task descriptor has different meanings for different algorithms. For AES-CTS, the data length field indicates byte numbers of source data, for others indicate word numbers of source data. For PRNG, the data length should be 5 words aligned. For TRNG, it should be 8 words aligned.

Source Address 0~7

Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	Source Data Address The address of the source data that needs to be stored.

Source Data Length 0~7

Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	Source Data Length The length of the source data.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Destination Address 0~7

Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	Destination Data Address The address of the destination data that needs to be stored.

Destination Data Length 0~7

Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	Destination Data Length The length of the destination data.

Next Descriptor Address

Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	Next Task Address The address where the descriptor of the next task in a task-chain is saved. If there is the only task or the last task of a task-chain, the next task address must be 32'h0.

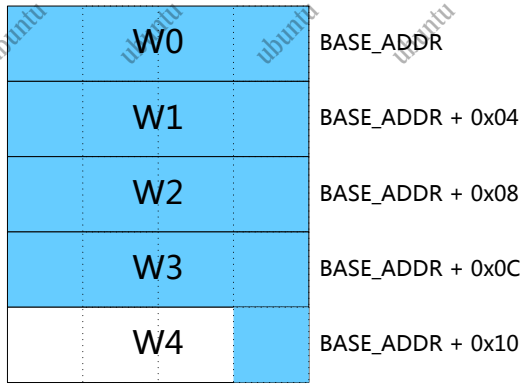
8.1.3.12 Storing Message

In the application, a message may not be stored contiguously in the memory, but divided into multiple segments. Or a piece of continuously stored messages can be artificially split into multiple pieces as needs. Then each segment corresponds to a set of the source address and source length in the descriptor. Multiple segments correspond to groups 0-7 source address/source length in sequence.

Each task supports up to 8 message segments, and the data volume of each message segment supports up to 4 GWord (AES-CTS is 1 GByte). The total amount of all segments in a task (that is a package) supports up to 4 GWord (AES-CTS is 1 GByte). If a message is divided into multiple packages, all others are required to be whole words; when the last package of AES-CTS is less than one word, 0 needs to be padded, and those less than one word are counted as one word. The following figure shows the address order structure.

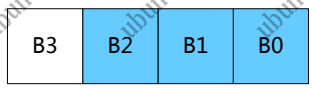
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Figure 8-11 Word Address of Message



Byte order: low byte first, high byte last. When the data is less than one word, the low byte is filled first. The following figure shows the byte order structure (blue means it is filled by the message).

Figure 8-12 Byte Order



Bit order: high bit first, low bit last. When the data is less than one Byte, the high bit is filled first. The following figure shows the bit order structure.

Figure 8-13 Bit Order

8.1.3.13 Storing Key

The length of KEY must be an integer multiple of word, refer to section 8.1.3.15 "Algorithm Length Properties".

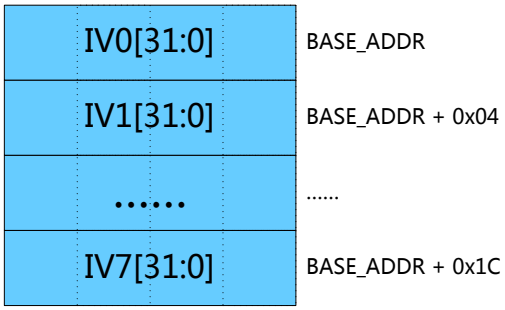
8.1.3.14 Storing IV

For different algorithms, the length of IV is different. But they are integer multiples of word. To keep the byte order of IV and HASH digest output consistent, the byte order of IV is different from that of the message. For the multi-packet operation, the first address of the digest output result of the previous HASH can be directly configured to the first address of the next IV, and the software does not need to do any processing on the digest.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

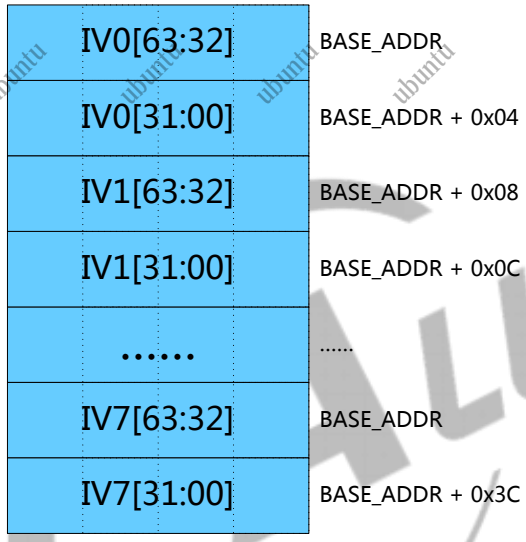
The following figure shows the storage method of 32-bit IV value.

Figure 8-14 The Storage Method of 32-bit IV



The following figure shows the storage method of 64-bit IV value.

Figure 8-15 The Storage Method of 64-bit IV



8.1.3.15 Algorithm Length Properties

The algorithm length has different requirements for different algorithms.

Table 8-1 Symmetric Algorithm Configuration Properties

Algorithm	Length Setting				Alignment	Software Padding
	Source Size	Destination Size	KEY	IV		
AES (except CTS)	< 4 GWord	< 4 GWord	AES-128: 4 Word AES-192: 6 Word AES 256: 8 Word	4 Word	Word-aligned	need

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Algorithm	Length Setting				Alignment	Software Padding
	Source Size	Destination Size	KEY	IV		
AES-CTS	< 1 GByte	< 1 GByte	AES-128: 4 word AES-192: 6 word AES 256: 8 word	4 Word	Word-aligned	need
DES	< 4 GWord	< 4 GWord	2 Word	2 Word	Word-aligned	need
TDES	< 4 GWord	< 4 GWord	6 Word	2 Word	Word-aligned	need

Table 8-2 Hash Algorithm Configuration Properties

Algorithm	Length Setting				Alignment	Software Padding
	Source Size	Destination Size	KEY	IV		
MD5	< 4 GWord	4 Word	Fixed to 0	4 Word	Word-aligned	need
SHA-1	< 4 GWord	5 Word	Fixed to 0	5 Word	Word-aligned	need
SHA-224	< 4 GWord	8 Word	Fixed to 0	8 Word	Word-aligned	need
SHA-256	< 4 GWord	8 Word	Fixed to 0	8 Word	Word-aligned	need
SHA-384	< 4 GWord	16 Word	Fixed to 0	16 Word	Word-aligned	need
SHA-512	< 4 GWord	16 Word	Fixed to 0	16 Word	Word-aligned	need
HMAC-SHA1	< 4 GWord	5 Word	16 Word	5 Word	Word-aligned	need
HMAC-SHA256	< 4 GWord	8 Word	16 Word	8 Word	Word-aligned	need

Table 8-3 RNG Configuration Properties

Algorithm	Length Setting				Alignment	Software Padding
	Source Size	Destination Size	KEY	IV		
TRNG	< 4 GWord	< 4 GWord	Fixed to 0	4 Word	Word-aligned	need
PRNG	< 4 GWord	< 4 GWord	6 Word	4 Word	Word-aligned	need

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Table 8-4 Asymmetric Algorithm Configuration Properties

Algorithm	Length Setting				Alignment	Software Padding
	Source Size	Destination Size	KEY	IV		
RSA512	16 Word	16 Word	16 Word	Not use IV	Word-aligned	need
RSA1024	32 Word	32 Word	32 Word	Not use IV	Word-aligned	need
RSA2048	64 Word	64 Word	64 Word	Not use IV	Word-aligned	need

8.1.3.16 Security Operation

When the CPU issues request to the CE module, the CE module will save the secure mode of CPU. When executing this request, this state bit works as a access flag for the inner and system resources. For access to SID module through the AHB bus, only the secure mode can succeed, or else these keys will be read to 0 or cannot write. When issuing MBUS read and write requests, the CE will use send this secure mode bit to BUS, so secure requests can access secure and non-secure space, but non-secure requests only can access non-secure space.

8.1.3.17 Error Detection

The CE module includes error detection for task configuration, data computing error, and authentication invalid. When the algorithm type in task descriptor is read into the CE module, the CE will check whether this type is supported through checking algorithm type field in common control. If the type value is out of scope, the CE will issue interrupt signal and set error state. Each type has certain input and output data size. After getting a task descriptor, the input size and output size configuration will be checked to avoid size error. If the size configuration is wrong, the CE will issue interrupt signal and set error state.

8.1.3.18 Clock Requirement

Clock Name	Description	Requirement
hclk	AHB bus clock	24 MHz ~ 200 MHz
mclk	MBUS clock	24 MHz ~ 400 MHz
ce_clk	CE work clock	24 MHz ~ 400 MHz

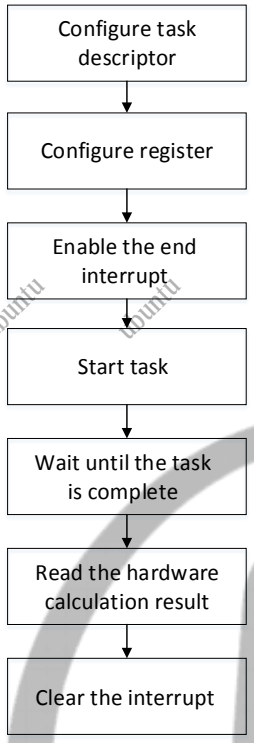
Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

8.1.4 Programming Guidelines

8.1.4.1 Symmetrical/Asymmetrical/Hash/RNG Algorithm Operation Process

The following figure shows the process of an algorithm operation.

Figure 8-16 Task Request Process



Step 1 The software should configure a task descriptor in memory, including the related fields in the descriptor. The channel id corresponds to one channel in CE. According to algorithm type, the software should set the fields in common control, symmetric control, asymmetric control, then provide key/iv/ctr address and the data length of this task. The source and destination address and size are set based on the upper application. If there is another task concatenating after this task, then set its descriptor address at the next descriptor field. For more details for task descriptor, see section 8.1.4.2, section 8.1.4.3 and section 8.1.4.4.

Step 2 The software should set registers. Configure the first address of the task descriptor structure to CE Task Descriptor Address Register. Configure the source/destination address to CE Current Source Address Register/CE Current Destination Address Register.

Step 3 Enable the end interrupt of the corresponding task channel by setting CE Interrupt Control Register.

Step 4 The software reads CE Task Load Register to ensure that the bit0 is 0. If the bit0 is not read out to be 0, wait until it is 0, then configure the bit0 to be 1 to start task.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Step 5 Wait for interrupt status by reading CE Interrupt Status Register.

Step 6 Read the result from the destination address.

Step 7 Clear the interrupt.

8.1.4.2 Configuring Task Descriptor of AES

- **Common control**: Configure Common Control[6:0] to 0x0 to select AES algorithm type.
- **Symmetric control**: According to the corresponding algorithm requirements, configure Symmetric Control to select the key size, CTR width, CTS last package flag, CFB width, and AES algorithm mode, and so on.
- **Asymmetric control**: The symmetric algorithm does not need to be configured for this field.
- **Key descriptor**: Because the storage of the key requires word alignment, ensure that this descriptor is the first address of the KEY (word address).
- **IV descriptor**: In the task that requires the IV value, configure the first address of the storage space where the IV is stored here. Because the storage of the IV requires word alignment, ensure that this descriptor is the first address of the IV (word address).
- **Data length**: Configure the data length of the corresponding segment. The data length size needs to be consistent with dst_data_length (destination data length 0 +... + destination data length 7). When the algorithm is CTS mode, the higher 30-bit of the data length is the word numbers of data volume; when the data_length[1:0] is 0, the data length is the higher 30-bit, otherwise it is increased by 1. For AES CTS, the data length indicates the byte numbers of the source data; for other algorithms, it indicates the word numbers.
- **Source address**: The first address of source data segments. Because the storage of the source data requires word alignment, ensure that this descriptor is the first address (word address).
- **Source data length**: The data volume of source data segments. The unit is word, and those less than one word are counted as one word. Note that only the last word of the entire message is allowed to be non-integer words, and the others must be integer words.
- **Destination address**: The first address of destination data segments. Because the storage of the destination data requires word alignment, ensure that this descriptor is the first address (word address).
- **Destination data length**: The data volume of destination data segments. The unit is word, and those less than one word are counted as one word. Note that only the last word of the entire message is allowed to be non-integer words, and the others must be integer words.
- **Next descriptor**: The first address of the next task descriptor. Because the storage of the descriptor requires word alignment, ensure that this descriptor is the first address (word address).
- **Reserved**: Configure to 0x0.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

8.1.4.3 Configuring Task Descriptor of HASH

- **Common control**
  - **Algorithm type**: Configure Common Control[6:0] to select SHA or HMAC algorithm type.
  - **Last HMAC plaintext**: If the algorithm type is HMAC, and the task is the last package of the message or if the message has only one package, then Common Control[15] needs to set to 1.
  - **IV mode**: The Common Control[16] (IV MODE) bit is only set to 1 in the following two scenarios, except that the bit must be configured to 0. (1). When the message is split into multiple packages, the Common Control[16] bit of other packages needs to be set to 1, except that the bit of the first package needs to be cleared to 0. (2). In special applications, if you need to customize the IV value to form the initial value of a certain HASH algorithm, you need to set the Common Control[16] bit of the first (or only one) package to 1, and the first address of the storage space where the customized IV value is stored in IV address.
- **Key descriptor**: Because the storage of the key requires word alignment, ensure that this descriptor is the first address of the KEY (word address).
- **IV descriptor**: In the task that requires the IV value, configure the first address of the storage space where the IV is stored here. Because the storage of the IV requires word alignment, ensure that this descriptor is the first address of the IV (word address).
- **Data length**: Configure the data length of the corresponding segment. The data length size needs to be consistent with dst_data_length (destination data length 0 +... + destination data length 7).
- **Source address**: The first address of source data segments. Because the storage of the source data requires word alignment, ensure that this descriptor is the first address (word address).
- **Source data length**: The data volume of source data segments. The unit is word, and those less than one word are counted as one word. Note that only the last word of the entire message is allowed to be non-integer words, and the others must be integer words.
- **Destination address**: The first address of destination data segments. Because the storage of the destination data requires word alignment, ensure that this descriptor is the first address (word address).
- **Destination data length**: The data volume of destination data segments. The unit is word, and those less than one word are counted as one word. Note that only the last word of the entire message is allowed to be non-integer words, and the others must be integer words.
- **Next descriptor**: The first address of the next task descriptor. Because the storage of the descriptor requires word alignment, ensure that this descriptor is the first address (word address).
- **Reserved**: Configure to 0x0.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

8.1.4.4 Configuring Task Descriptor of RSA

- **Common control**: Configure Common Control[6:0] to 0x20 to select RSA algorithm type.
- **Asymmetric control**: Configure Asymmetric Control[30:28] to select the RSA width.
- **Key descriptor**: Because the storage of the key requires word alignment, ensure that this descriptor is the first address of the KEY (word address).
- **Data length**: Configure the data length of the corresponding segment. The data length size needs to be consistent with dst_data_length (destination data length 0 +... + destination data length 7).
- **Source address**: The first address of source data segments. Because the storage of the source data requires word alignment, ensure that this descriptor is the first address (word address).
- **Source data length**: The data volume of source data segments. The unit is word, and those less than one word are counted as one word. Note that only the last word of the entire message is allowed to be non-integer words, and the others must be integer words.
- **Destination address**: The first address of destination data segments. Because the storage of the destination data requires word alignment, ensure that this descriptor is the first address (word address).
- **Destination data length**: The data volume of destination data segments. The unit is word, and those less than one word are counted as one word. Note that only the last word of the entire message is allowed to be non-integer words, and the others must be integer words.
- **Next descriptor**: The first address of the next task descriptor. Because the storage of the descriptor requires word alignment, ensure that this descriptor is the first address (word address).
- **Reserved**: Configure to 0x0.

8.1.5 Register List

Module Name	Base Address
CE_NS	0x03040000
CE_S	0x03040800

Register Name	Offset	Description
CE_TDA	0x0000	Task Descriptor Address
CE_CTL	0x0004	Control Register
CE_ICR	0x0008	Interrupt Control Register

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Register Name	Offset	Description
CE_ISR	0x000C	Interrupt Status Register
CE_TLR	0x0010	Task Load Register
CE_TSR	0x0014	Task Status Register
CE_ESR	0x0018	Error Status Register
CE_CSA	0x0024	DMA Current Source Address
CE_CDA	0x0028	DMA Current Destination Address
CE_TPR	0x002C	Throughput Register

8.1.6 Register Description

8.1.6.1 0x0000 CE Task Descriptor Address Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: CE_TDA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	Task Descriptor Address Configure as the first address of the descriptor structure.

8.1.6.2 0x0004 CE Control Register (Default Value: 0x0000_0000)

Offset: 0x0004			Register Name: CE_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:4	/	/	/
3	R/W	0x0	RSA CLK Gating Enable (only for CE_S) 0: RSA clock gating enabled 1: RSA clock gating disabled
2:0	/	/	/

8.1.6.3 0x0008 CE Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0008			Register Name: CE_ICR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:4	/	/	/

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0008			Register Name: CE_ICR
Bit	**Read/Write**	**Default/Hex**	**Description**
3:0	R/W	0x0	Task Channel3~0 Interrupt Enable 0: Disable 1: Enable

8.1.6.4 0x000C CE Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x000C			Register Name: CE_ISR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:4	/	/	/
3:0	R/W1C	0x0	Task Channel3~0 End Pending 0: Not finished 1: Finished It indicates whether task is completed. Write the corresponding channel bit of the register to clear the end flag. When the last task in the task chain ends, the operation of the task chain will end normally. If the task fails in the middle, the task chain will be aborted. The CE_ISR register will be automatically updated when it ends normally or aborts abnormally. And it is determined whether to generate an interrupt signal according to the IE configuration (bit31) of Common Control when the current task ends or aborts. If using interrupt, after receiving the interrupt, read the corresponding channel bit of CE_ISR to judge whether it ends successfully or stops failure. If not using interrupt, the CE_ISR status register can be continuously queried for the channel bit until the successful end flag is set or the failure stop flag is set. Write the corresponding channel bit of the register to clear the end flag. If it fails to stop, you can read the error code on the channel corresponding to the **CE_ESR** register.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

8.1.6.5 0x0010 CE Task Load Register (Default Value: 0x0000_0000)

Offset: 0x0010			Register Name: CE_TLR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	Task Load When set, the CE can load the descriptor of task if the task FIFO is not full.

8.1.6.6 0x0014 CE Task Status Register (Default Value: 0x0000_0000)

Offset: 0x0014			Register Name: CE_TSR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1:0	R	0x0	Running Channel Number 00: Task channel0 01: Task channel1 10: Task channel2 11: Task channel3

8.1.6.7 0x0018 CE Error Status Register (Default Value: 0x0000_0000)

Offset: 0x0018			Register Name: CE_ESR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:12	R/W1C	0x0	Task Channel3 Error Type xxx1: Algorithm not support xx1x: Data length error x1xx: Reserved 1xxx: Reserved
11:8	R/W1C	0x0	Task Channel2 Error Type xxx1: Algorithm not support xx1x: Data length error x1xx: Reserved 1xxx: Reserved

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Offset: 0x0018			Register Name: CE_ESR
Bit	Read/Write	Default/Hex	Description
7:4	R/W1C	0x0	Task Channel1 Error Type xxx1: Algorithm not support xx1x: Data length error x1xx: Reserved 1xxx: Reserved
3:0	R/W1C	0x0	Task Channel0 Error Type xxx1: Algorithm not support xx1x: Data length error x1xx: Reserved 1xxx: Reserved

8.1.6.8 0x0024 CE Current Source Address Register (Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: CE_CSA
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	Current source address

8.1.6.9 0x0028 CE Current Destination Address Register (Default Value: 0x0000_0000)

Offset: 0x0028			Register Name: CE_CDA
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	Current destination address

8.1.6.10 0x002C CE Throughput Register (Default Value: 0x0000_0000)

Offset: 0x002C			Register Name: CE_TPR
Bit	Read/Write	Default/Hex	Description
31:0	R/WC	0x0	It indicates the throughput writing to this register at last time. Writing to this register will clear it to 0.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

8.2 Security ID

The Security ID (SID) is used to program and read keys which include chip ID, thermal sensor, HASH code, and security key, and so on.

The SID module has the following features:

- 1 Kbit electrical fuse (eFuse)
- The module register is non-secure forever, the eFuse has secure zone and non-secure zone
- Backup eFuse information by using SID_SRAM
- A fuse only can program one time
- Burning the key to the SID
- Reading the key use status in the SID
- Loading the key to the CE

CAUTION

Before performing the burning operation, ensure that the power supply of the eFuse power pin is stable. After the burning operation is completed, cancel the power supply of the eFuse power pin.

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Appendix: Glossary

The following table contains acronyms and abbreviations used in this document.

Term	Meaning
A	
ADC	Analog-to-Digital Converter
AE	Automatic Exposure
AEC	Audio Echo Cancellation
AES	Advanced Encryption Standard
AGC	Automatic Gain Control
AHB	AMBA High-Speed Bus
ALC	Automatic Level Control
ANR	Active Noise Reduction
APB	Advanced Peripheral Bus
ARM	Advanced RISC Machine
AVS	Audio Video Synchronization
B	
BROM	Boot ROM
C	
CIR	Consumer Infrared
CMOS	Complementary Metal-Oxide Semiconductor
CP15	Coprocessor 15
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
D	
DDR	Double Data Rate
DES	Data Encryption Standard
DLL	Delay-Locked Loop
DMA	Direct Memory Access
DRC	Dynamic Range Compression
DVFS	Dynamic Voltage and Frequency Scaling
E	
ECC	Error Correction Code
eFuse	Electrical Fuse, A one-time programmable memory
EHCI	Enhanced Host Controller Interface
eMMC	Embedded Multi-Media Card
ESD	Electrostatic Discharge
F	
FBGA	Fine Pitch Ball Grid Array
FEL	Fireware Exchange Launch
FIFO	First In First Out
G	
GIC	Generic Interrupt Controller

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Term	Meaning
GPIO	General Purpose Input Output
I	
I2C	Inter Integrated Circuit
I2S	Inter IC Sound
J	
JEDEC	Joint Electron Device Engineering Council
JPEG	Joint Photographic Experts Group
JTAG	Joint Test Action Group
L	
LCD	Liquid-Crystal Display
LFBGA	Low Profile Fine Pitch Ball Grid Array
LRADC	Low Rate Analog to Digital Converter
LSB	Least Significant Bit
M	
MAC	Media Access Control
MIC	Microphone
MLC	Multi-Level Cell
MMC	Multimedia Card
MSB	Most Significant Bit
N	
N/A	Not Application
NMI	Non Maskable Interrupt
NTSC	National Television Standards Committee
NVM	Non Volatile Storage Medium
O	
OHCI	Open Host Controller Interface
OTP	One Time Programmable
OWA	One Wire Audio
P	
PAL	Phase Alternating Line
PCM	Pulse Code Modulation
PHY	Physical Layer Controller
PID	Packet Identifier
PLL	Phase-Locked Loop
POR	Power-On Reset
PRCM	Power Reset Clock Management
PWM	Pulse Width Modulation
R	
R	Read only/non-Write
RGB	Read Green Blue
RGMII	Reduced Gigabit Media Independent Interface
RMII	Reduced Media Independent Interface
ROM	Read Only Memory

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Term	Meaning
RSA	Rivest-Shamir-Adleman
RTC	Real Time Clock
S	
SAR	Successive Approximation Register
SD	Secure Digital
SDIO	Secure Digital Input Output
SDK	Software Development Kit
SDRAM	Synchronous Dynamic Random Access Memory
SDXC	Secure Digital Extended Capacity
SLC	Single-Level Cell
SoC	System on Chip
SPI	Serial Peripheral Interface
SRAM	Static Random Access Memory
T	
TDES	Triple Data Encryption Standard
TWI	Two Wire Interface
U	
UART	Universal Asynchronous Receiver Transmitter
UDF	Undefined
USB DRD	Universal Serial Bus Dual Role Device
UTMI	USB2.0 Transceiver Macrocell Interface

Copyright©Allwinner Technology Co.,Ltd. All Rights Reserved.

Copyright©2020 Allwinner Technology Co.,Ltd. All Rights Reserved.

This documentation is the original work and copyrighted property of Allwinner Technology Co.,Ltd ("Allwinner"). No part of this document may be reproduced, modify, publish or transmitted in any form or by any means without prior written consent of Allwinner.

Trademarks and Permissions

Allwinner and the Allwinner logo (incomplete enumeration) are trademarks of Allwinner Technology Co.,Ltd. All other trademarks, trade names, product or service names mentioned in this document are the property of their respective owners.

Important Notice and Disclaimer

The purchased products, services and features are stipulated by the contract made between Allwinner Technology Co.,Ltd ("Allwinner") and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Please read the terms and conditions of the contract and relevant instructions carefully before using, and follow the instructions in this documentation strictly. Allwinner assumes no responsibility for the consequences of improper use (including but not limited to overvoltage, overclock, or excessive temperature).

The information in this document is provided just as a reference or typical applications, and is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents. Allwinner is not responsible for any damage (including but not limited to indirect, incidental or special loss) or any infringement of third party rights arising from the use of this document. All statements, information, and recommendations in this document do not constitute a warranty or commitment of any kind, express or implied.

No license is granted by Allwinner herein express or implied or otherwise to any patent or intellectual property of Allwinner. Third party licences may be required to implement the solution/product. Customers shall be solely responsible to obtain all appropriately required third party licences. Allwinner shall not be liable for any licence fee or royalty due in respect of any required third party licence. Allwinner shall have no warranty, indemnity or other obligations with respect to third party licences.