A50 Datasheet

High Performance Quad-Core Tablet Solution

Revision 1.2

Nov.13, 2018

DECLARATION

THIS DOCUMENTATION IS THE ORIGINAL WORK AND COPYRIGHTED PROPERTY OF ALLWINNER TECHNOLOGY (“ALLWINNER”). REPRODUCTION IN WHOLE OR IN PART MUST OBTAIN THE WRITTEN APPROVAL OF ALLWINNER AND GIVE CLEAR ACKNOWLEDGEMENT TO THE COPYRIGHT OWNER.

THE PURCHASED PRODUCTS, SERVICES AND FEATURES ARE STIPULATED BY THE CONTRACT MADE BETWEEN ALLWINNER AND THE CUSTOMER. 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 FURNISHED BY ALLWINNER IS PROVIDED JUST AS A REFERENCE OR TYPICAL APPLICATIONS, ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS DOCUMENT DO NOT CONSTITUTE A WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. ALLWINNER RESERVES THE RIGHT TO MAKE CHANGES IN CIRCUIT DESIGN AND/OR SPECIFICATIONS AT ANY TIME WITHOUT NOTICE.

NOR FOR ANY INFRINGEMENTS OF PATENTS OR OTHER RIGHTS OF THE THIRD PARTIES WHICH MAY RESULT FROM ITS USE. NO LICENSE IS GRANTED BY IMPLICATION OR OTHERWISE UNDER ANY PATENT OR PATENT RIGHTS 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 MATTERS COVERED UNDER ANY REQUIRED THIRD PARTY LICENCE.

Revision History

Revision	Date	Description
1.0	Jun. 27, 2018	Initial Release Version
1.1	Aug.01, 2018	**Chapter 2 Hardware** 2.3.5, Update the power on and power down sequences
1.2	Nov.13,2018	**Chapter 2 Hardware** 2.3.10, Update the electrical parameters of Audio-Codec in table 2-15. **Chapter 9 Audio** 9.3.1, Update the content of overview. 9.3.2.8, Modify the Typical Application Diagram of Audio Codec. **Chapter 13 Reflow Profile** Update the parameter of reflow profile in table 13-1.

Contents

Figures

Tables

About This Document

Purpose

This document describes the features,logical structures,functions,operating modes,and related registers of each module about A50. The document also describes the interface timings and related parameters in diagrams.In addition,the document describes the pins,pin usages,performance parameters, and package dimension of A50 in detail.

Intended Audience

The document is intended for:

- Design and maintenance personnel for electronics
- Sales personnel for electronic parts and components

Conventions

Symbol Conventions

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

Symbol	Description
WARNING	A warning means that injury or death is possible if the instructions are not obeyed.
CAUTION	A caution means that damage to equipment is possible.
NOTE	Provides additional information to emphasize or supplement important points of the main text.

Table Content Conventions

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

Symbol	Description
-	The cell is blank.

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

Reset Value Conventions

In the register definition tables:

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.

Numerical System

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

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

The expressions of addresses and data are described as follows.

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, XX1 indicates 001,011,101 or 111.

Acronyms and Abbreviations

The following table contains acronyms and abbreviations used in this document.

ADC	Analog-to-Digital Converter
AE	Automatic Exposure
AEC	Audio Echo Cancellation
AES	Advanced Encryption Standard
AF	Automatic Focus
AGC	Automatic Gain Control
AHB	AMBA High-Speed Bus
ALC	Automatic Level Control
ANR	Active Noise Reduction

APB	Advanced Peripheral Bus
ARM	Advanced RISC Machine
AVS	Audio Video Standard
AWB	Automatic White Balance
BROM	Boot ROM
CIR	Consumer Infrared
CMOS	Complementary Metal-Oxide Semiconductor
CP15	Coprocessor 15
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
CSI	Camera Serial Interface
CVBS	Composite Video Broadcast Signal
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
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
FBGA	Fine Ball Grid Array
FEL	Fireware Exchange Launch
FIFO	First In First Out
GIC	Generic Interrupt Controller
GPIO	General Purpose Input Output
HD	High Definition
HDCP	High-bandwidth Digital Content Protection
HDMI	High-Definition Multimedia Interface
HiSPI	High-Speed Serial Pixel Interface
I2C	Inter Integrated Circuit
I2S	Inter IC Sound
ISP	Image Signal Processor
JEDEC	Joint Electron Device Engineering Council
JPEG	Joint Photographic Experts Group
JTAG	Joint Test Action Group
KEYADC	Analog to Digital Converter for Key
LCD	Liquid-Crystal Display
LSB	Least Significant Bit
LVDS	Low Voltage Differential Signaling
MAC	Media Access Control
MIC	Microphone
MIPI	Mobile Industry Processor Interface
MLC	Multi-Level Cell

MMC	Multimedia Card
MPEG	Motion Pictures Expert Group
MSB	Most Significant Bit
N/A	Not Application
NMI	Non Maskable Interrupt
NTSC	National Television Standards Committee
NVM	Non Volatile Storage Medium
OHCI	Open Host Controller Interface
OSD	On-Screen Display
OTP	One Time Programmable
OWA	One Wire Audio
PAL	Phase Alternating Line
PCM	Pulse Code Modulation
PHY	Physical Layer Controller
PID	Packet Identifier
PLL	Phase-Locked Loop
POR	Power-On Reset
PWM	Pulse Width Modulation
R	Read only/non-Write
RGB	Read Green Blue
RGMII	Reduced Gigabit Media Independent Interface
RMII	Reduced Media Independent Interface
ROM	Read Only Memory
RSA	Rivest-Shamir-Adleman
RTC	Real Time Clock
SAR	Successive Approximation Register
SD	Secure Digital
SDIO	Secure Digital Input Output
SDK	Software Development Kit
SDRAM	Sychronous 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
TDES	Triple Data Encryption Standard
TFBGA	Thin Fine Ball Grid Array
TWI	Two Wire Interface
UART	Universal Asynchronous Receiver Transmitter
UDF	Undefined
USB OTG	Universal Serial Bus On The Go
UTMI	USB2.0 Transceiver Macrocell Interface

1 Product Description

1.1 Description

A50 is a SOC solution for extreme machine and low cost tablet. The A50 integrates DRAM controller which supports DDR3, DDR4, LPDDR3, LPDDR4, So that kinds of DRAM types can be chosed flexibly. Especially, A50 integrates NAND Flash Controller with LDPC error correction function, Which supports the application of the mainstream 2D/3D SLC/MLC/TLC Nand Flash on the market. Through the adoption of advanced technology and low power design architecture, A50 integrates quad-core CPU with operating frequency up to 1.8GHz, and also includes advanced graphics computing processor GPU with operating frequency up to 600MHz,which provides sufficient computing capacity for the A50 product. It also supports video playback for 1080p@60fps and mainstream high-definition video decoding including H.264 by 1080p@60fps, Built in ISP that supports 13M MIPI-CSI camera. The A50 integrates various display output interfaces including MIPI-DSI, RGB, LVDS. And also integrates memory interface including eMMC, Nand, SD Card. To deliver better architecture scalability, the A50 comes with extensive connectivity and interfaces, such as USB OTG/Host, SPI, UART, TWI,PWM, LRADC. A50 has a flexible memory collocation scheme, sufficient computing resources, mainstream decoding and encoding ability, and abundance of peripheral interface. Greatly reduce the cost of the overall plan and support the development requirements of a variety of differentiated products.

1.2 Application Scenarios

1.2.1 Tablet Application Solution

The typical application scenario of the A50 is shown in Figure 1-1.

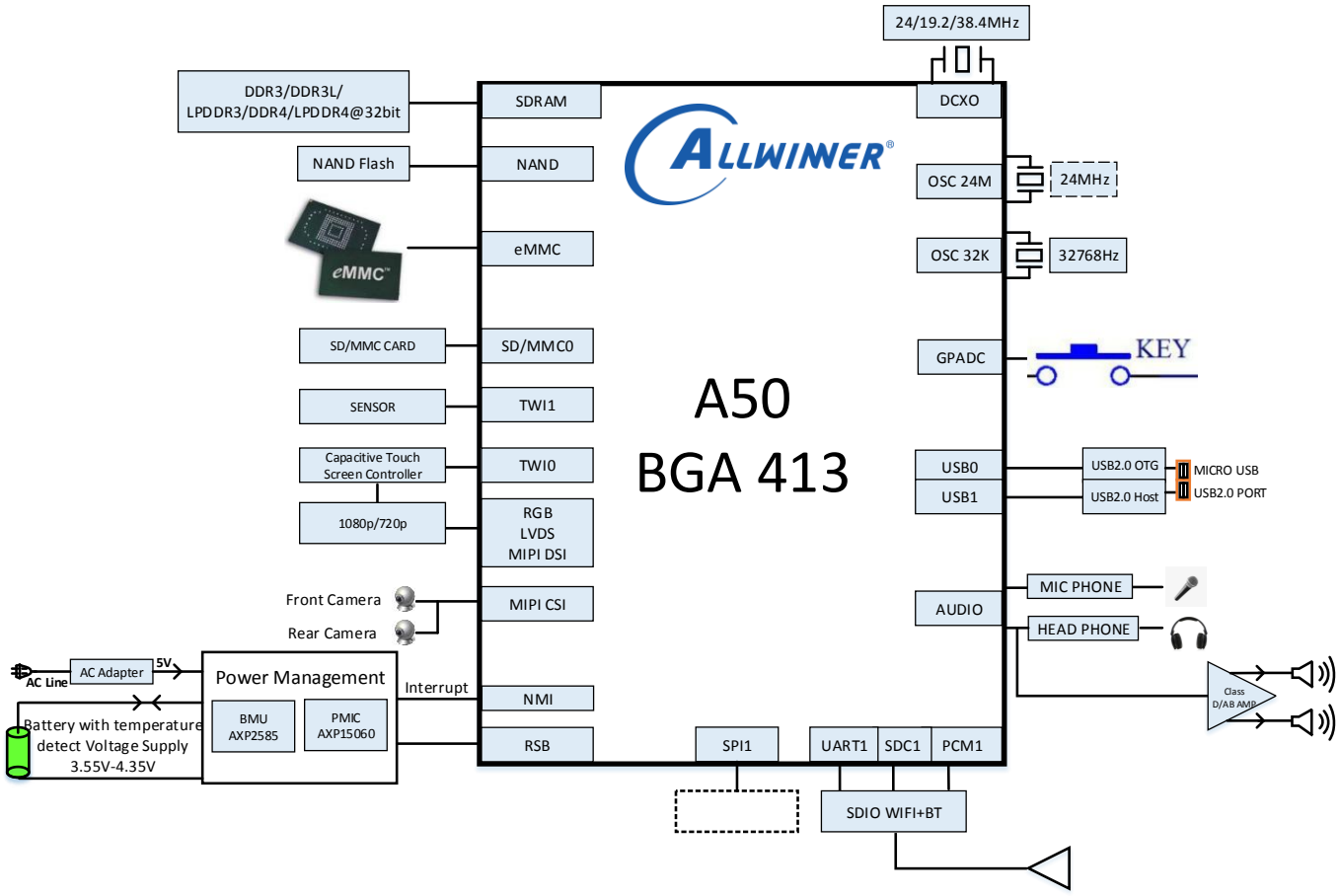


Figure1- 1. A50 Application Diagram

1.3 Architecture

1.3.1 Overview

The logic block diagram of the A50 is shown in Figure 1-2.

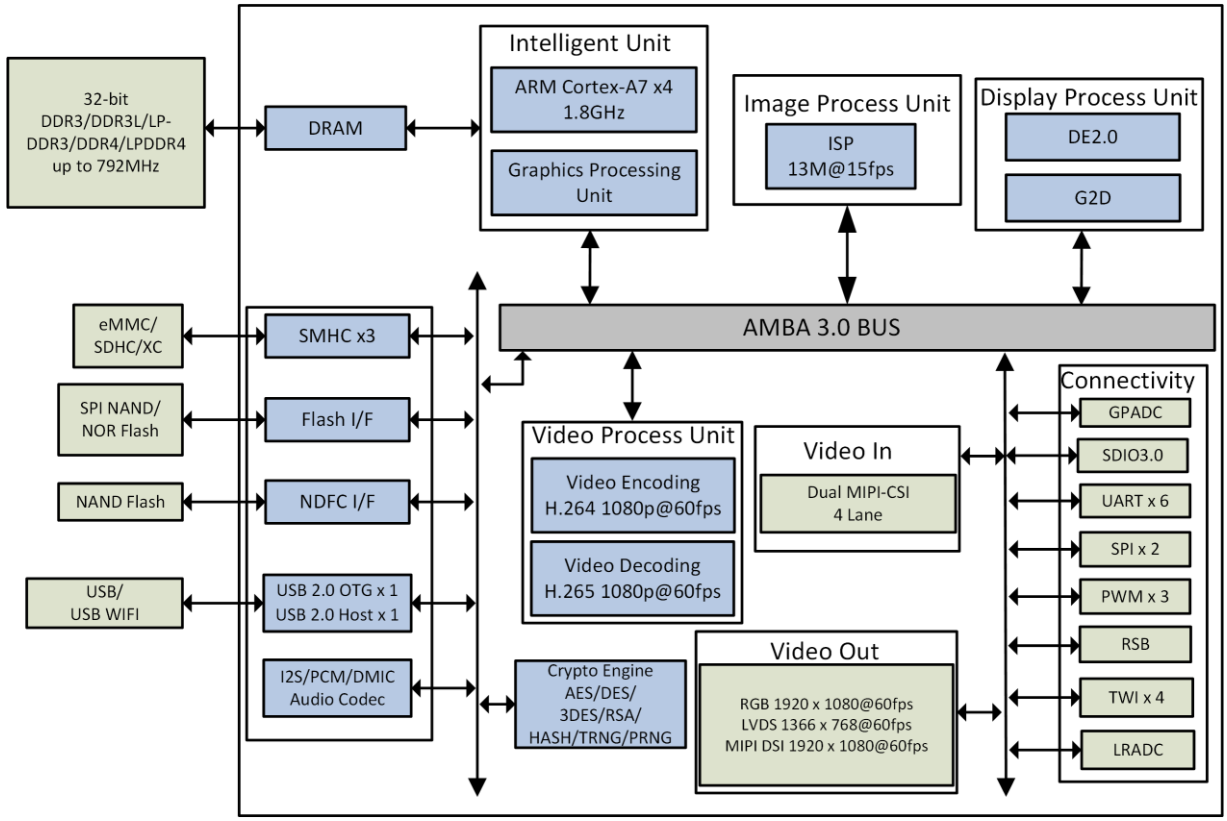


Figure1- 2. A50 Logic Block Diagram

1.3.2 CPU Architecture

- Quad-core ARM Cortex™-A7@1.8GHz
- Power-efficient ARM v7 architecture
- 32-bit execution states for scalable high performance
- Trustzone technology supported
- Supports NEON Advanced SIMD(Single Instruction Multiple Data)instruction for acceleration of media and signal processing functions
- Supports Large Physical Address Extensions(LPAE)
- VFPv4 Floating Point Unit
- 32KB L1 Instruction cache and 32KB L1 Data cache per core
- 512KB L2 cache shared

1.3.3 GPU Architecture

- Mali400 MP2
- Supports OpenGL ES 2.0/1.1
- Supports OpenVG 1.1

1.3.4 Video Encoding and Decoding

1.3.4.1 Video Encoding Specifications

- Supports H.264 video encoder up to 1080p@60fps
- Supports MJPEG video encoder up to 1080p@30fps
- Supports input formats: YU12/YV12/NV12/NV21/YUYV/YVYU/UYVY/VYUY/ARGB/BGRA/RGBA/ABGR/YU16/YV16/TILE32/TILE128

1.3.4.2 Video Decoding Specifications

- Supports multi-format video decoder, include:
  - H.265 Main/L4.1: 1080p@60fps
  - H.264 BP/MP/HP: 1080p@60fps
  - MPEG1 MP/HL : 1080p@60fps
  - MPEG2 MP/HL : 1080p@60fps
  - MPEG4-XVID SP/ASP : 1080p@60fps
  - VC1 SP/MP/AP : 1080p@30fps
  - VP8 : 1080p@60fps
  - VP9 : 720p@30fps
  - AVS/AVS+: 1080p@60fps
  - MJPEG:1080p@60fps
  - WMV7/8 :1080p@60fps
- Supports 1080p blu-ray 3D
- Supports frame compatible 3D format,size:3840x1080,1920x2160
- Supports decoding output format: T32x32, YV12, NV12 NV21
- Supports HEVC decoder 1080p@60fps

1.3.5 Video and Graphics Processing

1.3.5.1 Display Engine 2.0

- Output size up to 2048 x 2048
- Supports four alpha blending channels for main display
- Supports four overlay layers in each channel, and has a independent scaler
- Supports potter-duff compatible blending operation
- Supports input format semi-planar of YUV422/YUV420/YUV411 and planar of YUV422/YUV420/YUV411, ARGB8888/XRGB8888/RGB888/ARGB4444/ARGB1555/RGB565
- Supports Frame Packing/Top-and-Bottom/Side-by-Side Full/Side-by-Side Half 3D format data
- Supports SmartColor 2.0 for excellent display experience
  - Adaptive detail/edge enhancement
  - Adaptive color enhancement
  - Adaptive contrast enhancement and fresh tone rectify

- Content adaptive backlight control

- Supports writeback for miracast

1.3.5.2 Graphics 2D(G2D)

- Supports Layer size up to 2048 x 2048 pixels
- Supports input/output format: YUV422(semi-planar and planar format) /YUV420(semi-planar and planar format)/P010/P210/P410/Y8/ARGB8888/XRGB8888/RGB888/ARGB4444/ ARGB1555/ARGB2101010 and RGB565
- Supports Horizontal and Verticalflip, Clockwise 0/90/180/270 degree rotate

1.3.6 ISP

- Supports one individual image signal processor(ISP), supports 4224 x 3168 resolution
- Adjustable 3A functions, including automatic exposure(AE), automatic white balance(AWB) and automatic focus (AF)
- Gamma correction and Color enhancement
- Supports spatial(2D) de-noise filter
- Supports contrast enhance and sharping
- Supports chrominance noise reduction
- Supports defect pixel correction

1.3.7 Video Interfaces

1.3.7.1 Input

MIPI CSI

- Compliant with MIPI-CSI2 V1.00 and MIPI DPHY V1.00.00
- 1/2/3/4 Data Lanes Configuration and up to 1Gbps per Lane in HS Transmission
- Maximum to 13M@15fps, or 8M@30fps with 4 data lane
- Supports format: YUV422-8bit/10bit, YUV420-8bit, RAW-8, RAW-10, RAW-12, RGB888, RGB565.

VIPP

Supports Dual Video Input Post Processor(VIPP), Supports 4224 x 3168 resolution

Supports image interception

The output scaling of width is 1/8~1x

The output scaling of height is 1/8~1x

1.3.7.2 Output

- Supports one channel MIPI DSI output, MIPI DSI is 4-lane
  - Single channel(4-lane) MIPI DSI with up to 1920x1080@60fps
  - Compliant with MIPI DSI V1.0 and MIPI D-PHY V1.0
  - Supports Video mode(Normal mode and Burst mode)
- Supports LVDS output

- Supports LVDS interface with single link, up to 1366 x 768@60fps
- Supports RGB output
  - Supports RGB888, RGB666 and RGB565 with dither function
  - Supports RGB interface with DE/SYNC mode, up to 1920 x 1080@60fps
  - Supports i8080 interface, up to 800 x 480@60fps
  - Supports Gamma correction with R/G/B channel independence

1.3.8 Audio Interfaces

- Integrated audio codec, supporting 20-bit audio input and output
- Inter-IC sound(I2S)/time division multiplex(TDM) interface for connecting to an external audio codec
- Integrated digital microphone, supports maximum 8 digital microphones

1.3.9 Security Engine

Crypto Engine(CE)

- Supports Symmetrical algorithm: AES,DES,3DES,XTS
  - Supports ECB,CBC,CTS,CTR,CFB,OFB,CBC-MAC mode for AES
  - Supports 128/192/256-bit key for AES
  - Supports ECB,CBC,CTR,CBC-MAC mode for DES/TDES
  - Supports 256/512-bit key for XTS
- Supports Hash algorithm: MD5,SHA,HMAC
  - Supports SHA1,SHA224,SHA256,SHA384,SHA512 for SHA
  - Supports HMAC-SHA1,HMAC-SHA256 for HMAC
  - MD5,SHA,HMAC are padded using hardware
- Supports Pubic Key algorithm: RSA, ECC
  - RSA supports 512/1024/2048/3072/4096-bit width
  - ECC Supports 160/224/256/384/521-bit width
- Supports 160-bit hardware PRNG with 175-bit seed
- Supports 256-bit hardware TRNG
- Internal Embedded DMA to do data transfer
- Supports secure and non-secure interfaces respectively
- Supports task chain mode for each request. Task or task chain are executed at request order
- Symmetric, Asymmetric, HASH ctrl logics are separate, Which can handle task simultaneously.
- 8 scatter group(sg) are supported for both input and output data
- DMA has multiple channels, each corresponding to one suit of algorithms
- Supports 2.5K-bit EFUSE for chip ID and security application

Embedded Crypto Engine(EMCE)

- Connects directly to SMHC or NDFC for disc encryption application
- AES algorithm
- Supports ECB, CBC,XTS, CTS modes
- 128-bit, 192-bit and 256-bit key size for AES
- Supports Key scrambling
- Supports internal scrambling Key security control

- Supports the dynamic configuration for encryption configuration function through task chain

1.3.10 Peripheral Interfaces

- Power-on reset(POR),external reset input
- One internal RTC
- One channel general purpose analog-to-digital converter(GPADC)
- One channel low rate analog-to-digital converter(LRADC)
- Six UART interfaces
- Two SPI interfaces
- Three PWM interfaces
- Three SD3.0/SDIO3.0 interfaces,supporting secure digital extended capacity(SDXC)
- One USB2.0 OTG interface, one USB2.0 Host interface
- Four TWI interfaces, 106 GPIO interfaces

1.3.11 External Memory Interfaces

- DDR3/DDR3L/LPDDR3/DDR4/LPDDR4 interface
  - 32-bit DDR4, up to 792MHz
  - 32-bit DDR3/DDR3L, up to 792MHz
  - 32-bit LPDDR3, up to 672MHz
  - 32-bit LPDDR4, up to 672MHz
  - Maximum capacity of 2048 Mbytes
- Supports SPI Nor Flash interface
  - SPI, Dual SPI, and Quad SPI mode
  - 3 bytes or 4 bytes address mode
- Supports SPI Nand Flash interface
- Supports eMMC 5.0 interface
- Nand Flash interface
  - 8-bit data width
  - 4-,8-,24-,40-,64-,80-bit BCH per 1024 bytes
  - LDPC
  - 8-bit SLC/MLC/TLC flash and EF-NAND memory
- Booting from SPI NOR,SPI NAND,eMMC,SD, NAND Flash, one-key FEL
- Hardware boot pin select

1.3.12 Physical Specifications

- Power consumption
  - TBD
  - Multi-level power-saving mode
- Operating voltages
  - 0.9V core voltage
  - 3.3V IO voltage and 3.6V margin voltage

- 1.2V ,1.5V,1.5V,1.2V, or 1.1V DDR4/DDR3/DDR3L/LPDDR3/LPDDR4 interface voltage

- Package
  - Restrictions on the use of certain hazardous substances(RoHS), thin&fine-pitch ball grid array(TFBGA)
  - Body size of 12.8 mm x 12.3 mm, 0.5 mm ball pitch

1.4 Boot Modes

The A50 can boot from the following devices:

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

During power-on reset, the Boot Media despends on the values of BOOT_SEL[0] , the details are shown in Table 1-1.

Table1- 1. Signal Values and Boot Modes

BOOT_SEL[0]	Boot Mode
0	Booting by polling from SD(SMHC0), NAND FLASH , ,eMMC(SMHC2), SPI NOR
1	Booting by polling from SD(SMHC0), eMMC(SMHC2), NAND FLASH , SPI NAND

Figures

Tables

2 Hardware

2.1 Package and Pinout

2.1.1 Package

The A50 uses the TFBGA package, it has 413 pins, its body size is 12.8 mm x 12.3 mm, and its ball pitch is 0.5mm pitch. Figure 2-1 shows the top view of the 12.8mm x 12.3mm package, Figure 2-2 shows the bottom view of the 12.8mm x 12.3mm package , Figure 2-3 shows the enlarged view of detail “B” in the bottom view, and Figure 2-4 shows the side view of the 12.8mm x 12.3mm package, Figure 2-5 shows the enlarged view of detail “A” in the side view. Figure 2-6 shows the dimensions of the 12.8mm x 12.3mm package.

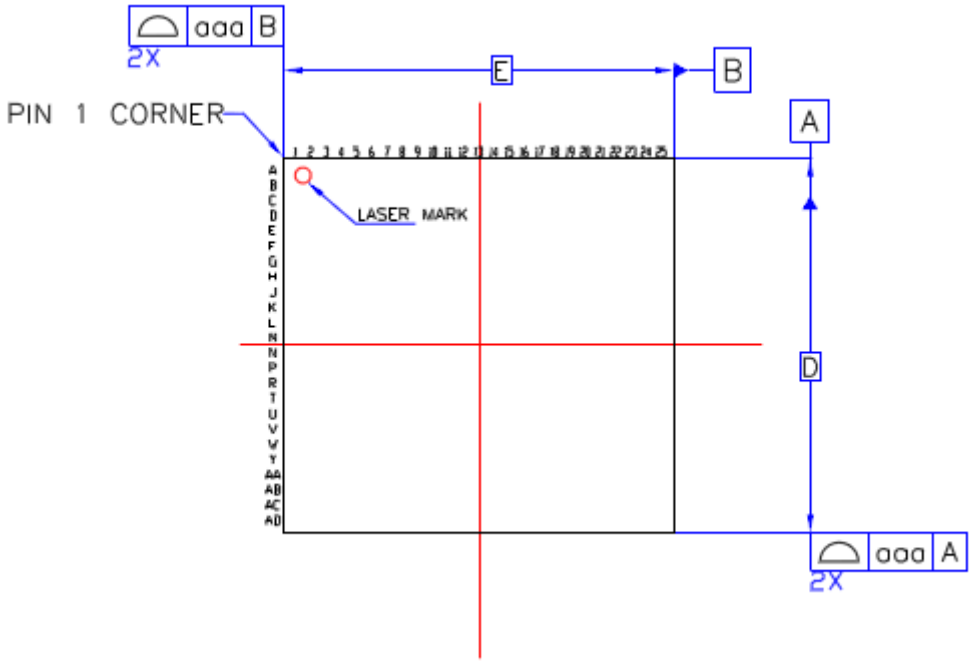


Figure2- 1. 12.8 mm x 12.3 mm Package Top View

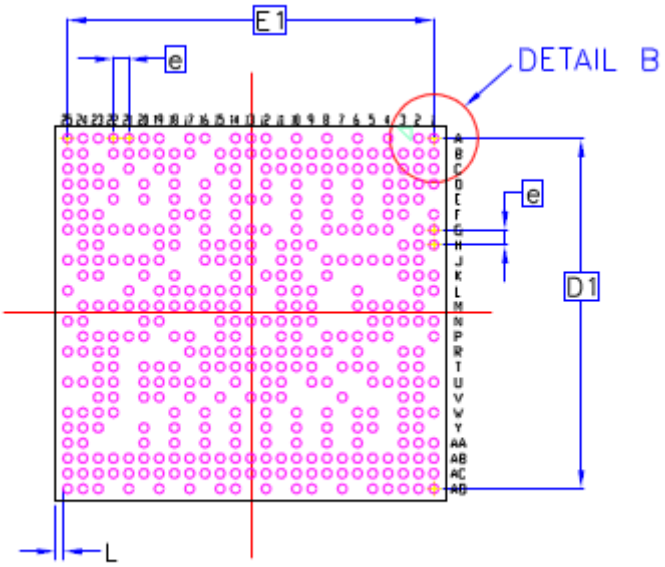


Figure2- 2. 12.8 mm x 12.3 mm Package Bottom View

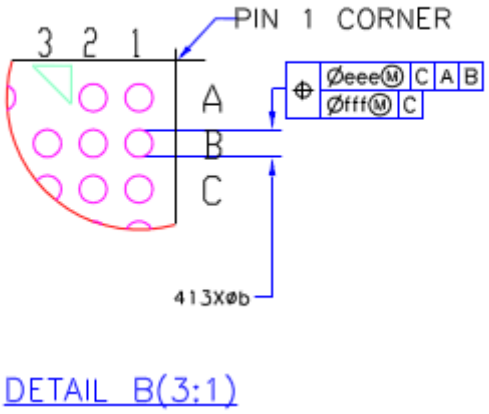


Figure2- 3. Enlarged View of Detail "B"

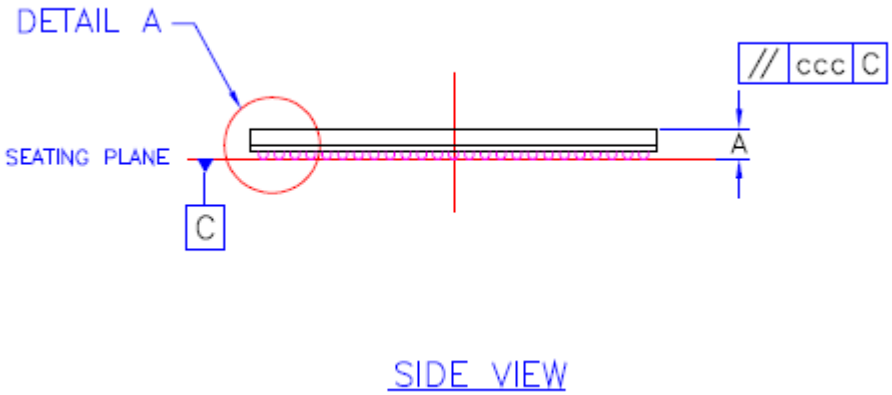


Figure2- 4. 12.8 mm x 12.3 mm Package Side View

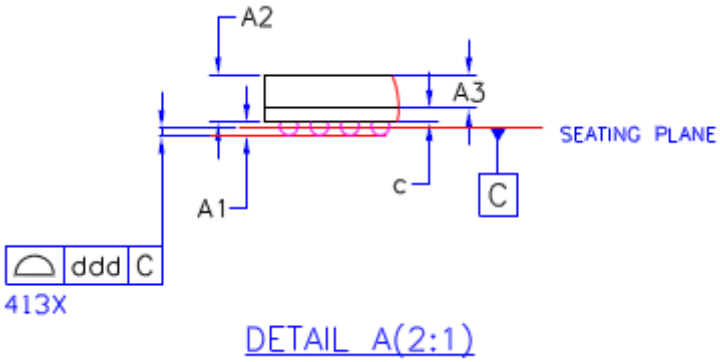


Figure2- 5. Enlarged View of Detail "A"

SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	0.89	0.97	1.05
A1	0.16	0.21	0.26
A2	0.71	0.76	0.81
A3	0.53 BASIC		
c	0.19	0.23	0.27
D	12.20	12.30	12.40
D1	11.50 BASIC		
E	12.70	12.80	12.90
E1	12.00 BASIC		
e	0.50 BASIC		
b	0.25	0.30	0.35
L	0.25 REF		
aaa	0.15		
ccc	0.15		
ddd	0.08		
eee	0.15		
fff	0.05		

Figure2- 6. 12.8 mm x 12.3 mm Package Dimensions

2.1.2 Pinout

For details about pinout of the A50, see the *A50 PINOUT.xls.*

2.2 Pin Description

For details about pin description of the A50, see the *A50 PINOUT.xls.*

2.3 Electrical Characteristics

2.3.1 Power Consumption Parameters

If you have questions about power consumption parameters, contact Allwinner FAE.

2.3.2 Thermal Resistance Parameters

Table 2-1 shows thermal resistance parameters. The following thermal resistance characteristics is based on JEDEC JESD51 standard, because the actual system design could be different with JEDEC JESD51 , the simulating result data is a reference only, please prevail in the actual application condition test.

NOTE

Test condition: four-layer board(2s2p),natural convection, no air flow.

Table2- 1. Thermal Resistance Parameters

Parameter	Symbol	Min	Typ	Max	Unit
Junction-to-Ambient Thermal Resistance	θ_{JA}	-	27.2	-	°C/W
Junction-to-Board Thermal Resistance	θ_{JB}	-	12.5	-	°C/W
Junction-to-Case Thermal Resistance	θ_{JC}	-	6.67	-	°C/W

2.3.3 Absolute Maximum Ratings

Absolute Maximum Ratings are those values beyond which damage to the device may occur. Table 2-2 specifies the absolute maximum ratings over the operating junction temperature range of commercial and extended temperature devices.

CAUTION

Stresses beyond those listed under Table 2-2 may affect reliability or cause permanent damage to the device. These are stress ratings only. Functional operation of the device at these or any other conditions beyond those indicated under Section 2.3.4, *Recommended Operating Conditions*, is not implied. Exposure to absolute maximum rated conditions for extended periods may affect device reliability.

Table2- 2. Absolute Maximum Ratings

Symbol	Parameter	Min	Max	Unit
AVCC	Power Supply for Analog Part	-0.3	2.16	V
CPVIN	Power Supply for CPVDD	-0.3	2.16	V
VCC_PC	Digital GPIO C Power	-0.3	3.96	V
VCC_PD	Digital GPIO D Power	-0.3	3.96	V

VCC_PE	Digital GPIO E Power		-0.3	3.96	V
VCC_PF	Digital GPIO F Power		-0.3	3.96	V
VCC_PG	Digital GPIO G Power		-0.3	3.96	V
VCC_PL	Digital GPIO L Power		-0.3	3.96	V
VCC_IO	Power Supply for 3.3V Digital Part		-0.3	3.96	V
VCC_RTC	Power Supply for RTC		-0.3	2.16	V
VCC_PLL	Power Supply for System PLL		-0.3	2.16	V
VCC_LVDS	Power Supply for LVDS		-0.3	2.16	V
VCC_MDSI	Power Supply for MIPI DSI		-0.3	2.16	V
VCC_MCSI	Power Supply for MIPI CSI		-0.3	2.16	V
VCC_USB	Power Supply for USB		-0.3	3.96	V
VCC_DRAM	Power Supply for DRAM		-0.3	1.8	V
VDD18_DRAM	Power Supply for DRAM Controller		-0.3	2.16	V
VDD_CPU	Power Supply for CPU0~3		-0.3	1.08	V
VDD_CPUS	Power Supply for CPUS		-0.3	1.08	V
VDD_SYS	Power Supply for System		-0.3	1.08	V
T_{STG}	Storage Temperature		-40	125	°C
V_{ESD}	Electrostatic Discharge	Human Body Model(HBM)(1)	-	2000	V
		Charged Device Model(CDM)(2)	-	250	V
$I_{Latch-up}$	Latch-up I-test performance current-pulse injection on each IO pin(3)		pass		
	Latch-up over-voltage performance voltage injection on each IO pin(4)		pass		

(1). Test method: JEDEC JS-001-2014. JEDEC publication JEP155 states that 500V HBM allows safe manufacturing with a standard ESD control process.

(2). Test method: JEDEC JS-002-2014. JEDEC publication JEP157 states that 250V CDM allows safe manufacturing with a standard ESD control process.

(3). Current test performance: Pins stressed per JEDEC JESD78E and passed with I/O pin injection current as defined in JEDEC.

(4). Over voltage performance: Supplies stressed per JEDEC JESD78E and passed voltage injection as defined in JEDEC.

2.3.4 Operating Conditions

All A50 modules are used under the operating conditions contained in Table 2-3.

Table2- 3. Operating Conditions

Symbol	Parameter	Min	Typ	Max	Unit
Ta	Ambient Operating Temperature	-20	-	70	°C
Tj	Junction Temperature Range	-20	-	110	°C
CPVIN	Power Supply for CPVDD	1.7	1.8	1.9	V
AVCC	Power Supply for Analog Part	1.782	1.8	1.818	V
VCC_PC	Digital GPIO C Power 1.8V Voltage 3.3V Voltage	 1.62 2.97	 1.8 3.3	 1.98 3.63	V
VCC_PD	Digital GPIO D Power 1.8V Voltage 3.3V Voltage	 1.62 2.97	 1.8 3.3	 1.98 3.63	V
VCC_PE	Digital GPIO E Power	1.62	1.8	1.98	V

	1.8V Voltage 3.3V Voltage	2.97	3.3	3.63	
VCC_PF	Digital GPIO F Power 1.8V Voltage 3.3V Voltage	1.62 2.97	1.8 3.3	1.98 3.63	V
VCC_PG	Digital GPIO G Power 1.8V Voltage 3.3V Voltage	1.62 2.97	1.8 3.3	1.98 3.63	V
VCC_PL	Digital GPIO L Power 1.8V Voltage 3.3V Voltage	1.62 2.97	1.8 3.3	1.98 3.63	V
VCC_IO	Power Supply for 3.3V Digital Part	2.97	3.3	3.63	V
VCC_RTC	Power Supply for RTC	1.62	1.8	1.98	V
VCC_PLL	Power Supply for System PLL	1.62	1.8	1.98	V
VCC_EFUSE	Power Supply for EFUSE Program Mode	1.8	1.89	1.98	V
VCC_LVDS	Power Supply for LVDS	1.75	1.8	1.98	V
VCC_MDSI	Power Supply for MIPI DSI	1.62	1.8	1.98	V
VCC_MCSI	Power Supply for MIPI CSI	1.62	1.8	1.98	V
VCC_USB	3.3V Power Supply for USB	3.07	3.3	3.6	V
VDD_USB	0.9V Power Supply for USB	0.837	0.9	0.99	V
VCC_DRAM	Power Supply for DDR4	1.14	1.2	1.26	V
	Power Supply for DDR3	1.425	1.5	1.575	V
	Power Supply for DDR3L	1.425	1.5	1.575	V
	Power Supply for LPDDR3	1.14	1.2	1.3	V
	Power Supply for LPDDR4	1.06	1.1	1.17	V
VDD_DRAM	Power Supply for Digital Part(include Controller and PHY)	0.87	0.9	0.93	V
VDD18_DRAM	Power Supply for DRAM Controller	1.7	1.8	1.95	V
VDD_CPU	Power Supply for CPU0~3	0.81	-	1.08	V
VDD_CPUS	Power Supply for CPUS	0.87	0.9	0.93	V
VDD_SYS	Power Supply for System	0.87	0.9	0.93	V

2.3.5 Power-On and Power-Off Sequences

Figure 2-7 shows an example of the power on sequence for A50 device. The description of the power on sequence is as follows.

- The consequent steps in power on sequence should not start before the previous step supplies have been stabilized within 90~110% of their nominal voltage,unless stated otherwise.
- VCC_RTC should remain powered on continuously,to maintain internal real-time clock status.And it has to be powered on together with VDD_CPUS,or preceding VDD_CPUS.
- VDD_CPUS should be powered on together, or any time after VCC_RTC.
- DCXO 24M clock need to start oscillating and be stable after the RESET is stable for 4ms.
- VDD_SYS,VCC_DRAM, VDD_DRAM and VDD_CPUS start to ramp simultaneously.
- Other power domains can ramp after VDD_SYS, VCC_DRAM, VDD_DRAM and VDD_CPUS are stabilized.
- During the entire power on sequence, the RESET pin must be held on low until all power domains are stable.

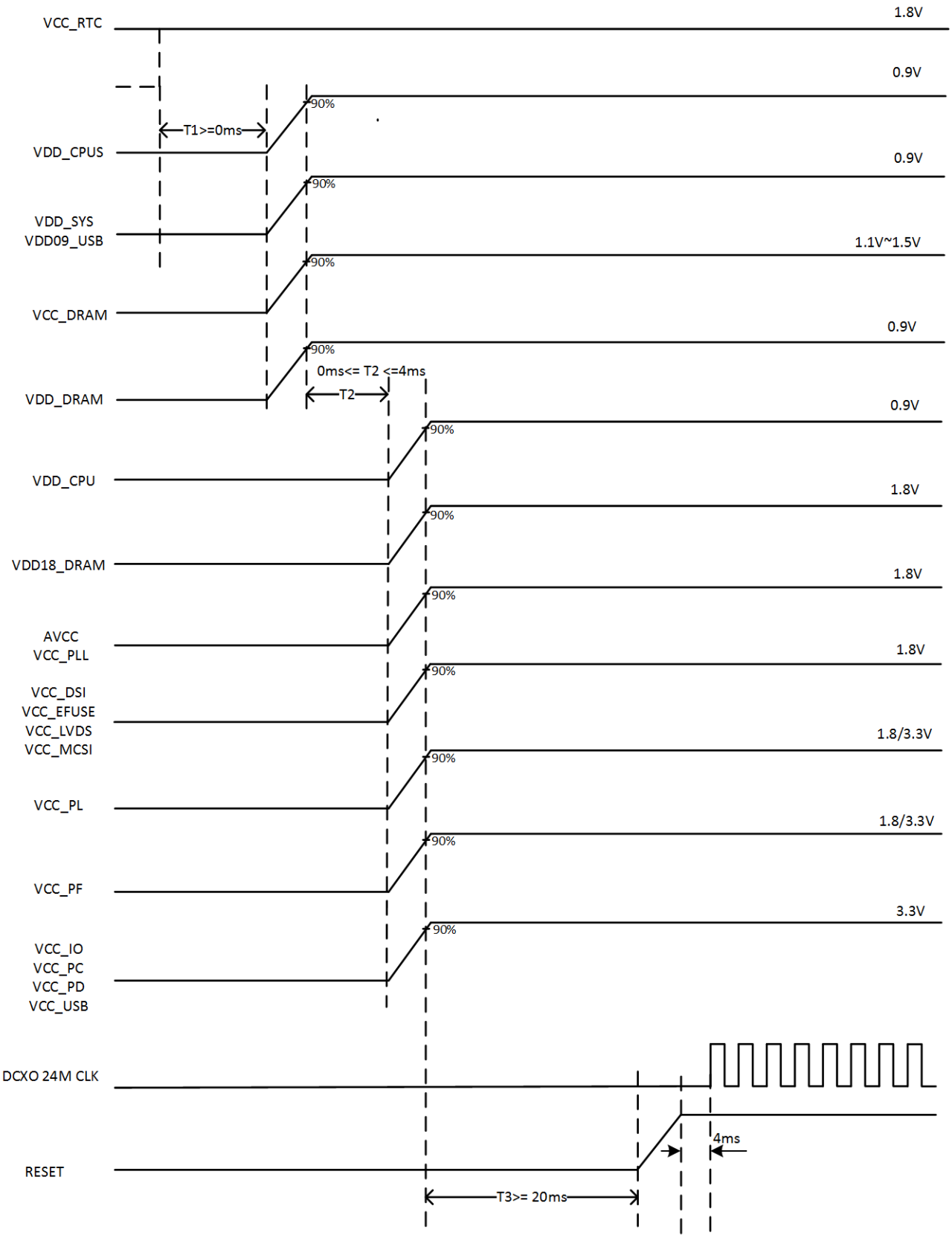


Figure2- 7. A50 Power-On Sequence

Figure 2-8 shows an example of the power off sequence for A50 device.

- Reset A50 device.
- VCC-RTC holds high.
- After PMIC receives the power-down command, pull-down RESET#, and delay T4.

- Other powers start to ramp down simultaneously, and the ramp rate of each power rail is generated by the load on the power.

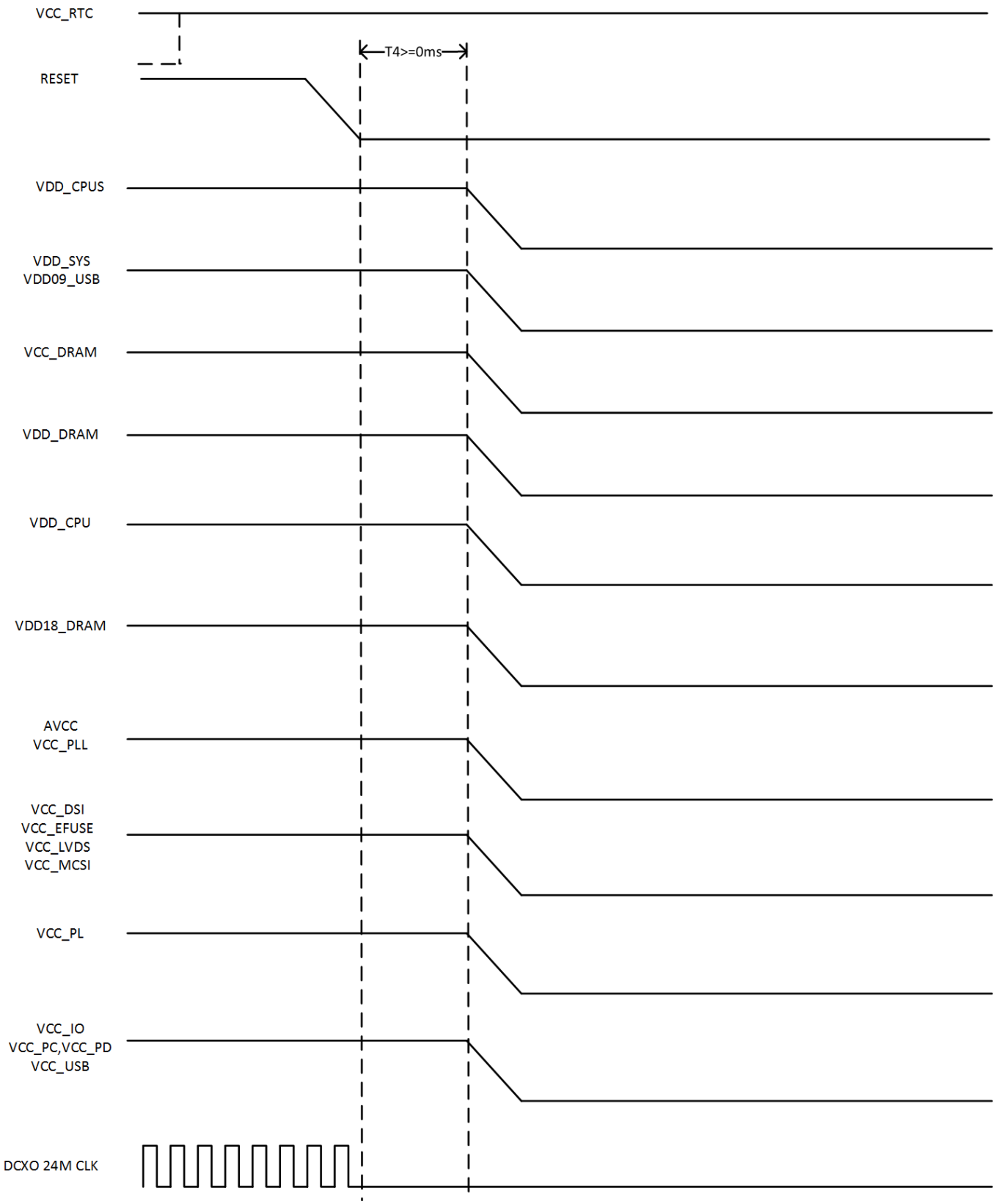


Figure2- 8. A50 Power-Off Sequence

2.3.6 DC Electrical Parameters

Table 2-4 summarizes the DC electrical characteristics of the A50.

Table2- 4. DC Electrical Parameters

Symbol	Parameter	Min	Typ	Max	Unit
V_{IH}	High-Level Input Voltage	0.7 * VCC_IO	-	VCC_IO + 0.3	V
V_{IL}	Low-Level Input Voltage	-0.3	-	0.3 * VCC_IO	V
R_{PU}	Input Pull-up Resistance	80	100	120	kΩ
R_{PD}	Input Pull-down Resistance	80	100	120	kΩ
I_{IH}	High-Level Input Current	-	-	10	uA
I_{IL}	Low-Level Input Current	-	-	10	uA
V_{OH}	High-Level Output Voltage	VCC_IO - 0.3	-	VCC_IO	V
V_{OL}	Low-Level Output Voltage	0	-	0.2	V
I_{OZ}	Tri-State Output Leakage Current	-10	-	10	uA
C_{IN}	Input Capacitance	-	-	5	pF
C_{OUT}	Output Capacitance	-	-	5	pF

2.3.7 SDRAM I/O DC Electrical Parameters

The DDR I/O pads supports DDR3/DDR3L, LPDDR3, DDR4 and LPDDR4 operational modes. The SDRAM Controller(DRAMC) is designed to be compatible with JEDEC-compliant SDRAMs. The DRAMC supports the following memory types:

- DDR3 SDRAM compliant to JESD79-3F DDR3 JEDEC standard release July,2012
- DDR3L SDRAM compliant to JESD79-3-1A DDR3L JEDEC standard release January,2013
- LPDDR3 SDRAM compliant to JESD209-3C LPDDR3 JEDEC standard release August,2015
- DDR4 SDRAM compliant to JESD79-4A DDR4 JEDEC standard release November,2013
- LPDDR4 SDRAM compliant to JESD209-4A LPDDR4 JEDEC standard release November,2015

Table2- 5. DC Input Logic Level

Characteristics	Symbol	Min	Max	Unit
DC input logic high	$V_{IH(DC)}$	V_{ref} +20	-	mV
DC input logic low	$V_{IL(DC)}$	-	V_{ref} –20	mV

Table2- 6. DDR3/DDR3L mode, DC Input Conditions

Parameter	Symbol	Unit	Min	Typ	Max	Notes
Reference Voltage	V_{ref}	V_{DDQ}	30.1%	31.1%	32.1%	Please refer to Note 1 and 2
On-die termination(ODT) programmable resistances	R_{TT}	ohm	-	open, 120, 60	-	Please refer to Note 3

Table2- 7. LPDDR3 and DDR4 mode, DC Input Conditions

Parameter	Symbol	Unit	Min	Typ	Max	Notes
Reference Voltage	V_{ref}	V	-	Variable	-	Please refer to Note 4.

On-die termination(ODT) programmable resistances	R_{TT}	ohm	-	open, 240, 120, 80, 60, 48, 40	-	Please refer to Note 5 and 6

Table2- 8. LPDDR4 mode, DC Input Conditions

Parameter	Symbol	Unit	Min	Typ	Max	Notes
Reference Voltage	V_{ref}	V	-	Variable	-	Please refer to Note 4.
On-die termination(ODT) programmable resistances	R_{TT}	ohm	-	open, 240, 120, 80, 60, 48, 40	-	Please refer to Note 5 and 7.

NOTE

1. If the external V_{ref} to the receivers is enabled, V_{ref} is expected to be set to a nominal value of (VDDQ/2)*RxAtten(RxAttenuation for DDR3/DDR3L is 0.623)through a voltage divider in order to track VDDQ level. It can be adjusted in the system to margin the input DQ signals, although this margin does not necessarily represent the eye height since a change in V_{ref} also changes the input receiver common mode, altering receiver performance.

2. Externally supplied V_{ref} is not recommended. Internal V_{ref} generation through locak V_{ref} generation at each receiver is preferred.

3. For DDR3, ODT is a Thevenin resistance to V_{DDQ}/2.

4. V_{ref} must be set according to the DRAMs loading and termination during reads. Because termination at the DRAMs is configurable, there is no fixed setting. The V_{ref} value is dependent on driver impedance R_{on} and System effective ODT impedence R_{TT}. The V_{ref} value for specific combination of R_{on} and R_{tt} can be calculated from the following equation:

$$V_{ref} = \frac{V_{DDQ}}{2}\left(\frac{2R_{on} + R_{tt}}{R_{on} + R_{tt}}\right)$$

5. ODT is configurable based on DRAM configuration.

6. For DDR4, ODT is a pull-up to V_{DDQ}.

7. For LPDDR4, ODT is a pull-down to V_{SS}.

2.3.8 MIPI RX Electrical Parameters

Table 2-9 shows MIPI-RX differential DC electrical parameters.

Table2- 9. MIPI-RX Differential DC Electrical Parameters

Symbol	Parameter	Min	Typ	Max	Unit
V_{IDTH}	Differential input threshold voltage $(V_P-V_M)/2_{(MIN)}$	-	-	±70	mV
V_{CM}	Common mode voltage range $(V_P+V_M)/2_{(MIN)}$	70	200	330	mV
V_{CM}	Single-ended input voltage range V_P,V_M	-40	-	460	mV
Z_{ID}	Internal termination resister value	80	100	125	Ω

Table 2-10 to Table 2-12 show MIPI electrical parameters.

Table2- 10. MIPI RX High-Speed(HS) DC Parameters

Symbol	Parameter	Min	Typ	Max	Unit
$V_{TERM-EN}$	Single-ended threshold for HS termination enable	-	-	450	mV

Table2- 11. MIPI RX HS AC Parameters

Symbol	Parameter	Min	Typ	Max	Unit
ΔV_{CMRX} (HF)	Common-mode interface beyond 450MHz	-	-	100	mV
ΔV_{CMRX} (LF)	Common-mode interface 50MHz ~450MHz	-50	-	50	mV
C_{CM}	Common-mode termination	-	-	60	pF

Table2- 12. MIPI RX Low-Power(LP) DC Parameters

Symbol	Parameter	Min	Typ	Max	Unit
V_{IHLP}	Logic 1 input voltage	880	-	-	mV
V_{ILLP}	Logic 0 input voltage	-	-	550	mV
V_{HYST}	Input hysteresis	25	-	-	mV

2.3.9 SDIO Electrical Parameters

The SDIO electrical parameters are related to different supply voltage.

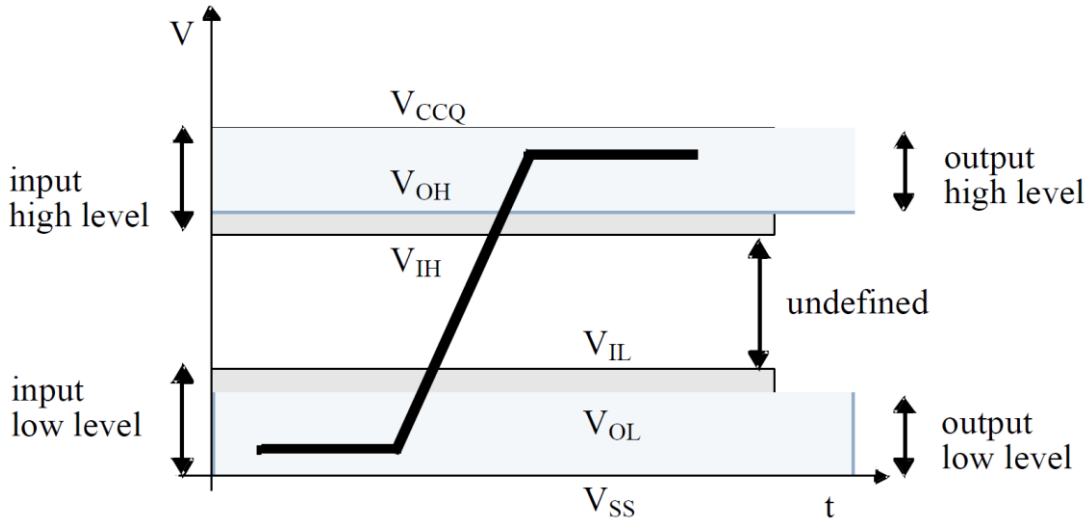


Figure2- 9. SDIO Voltage Waveform

Table 2-13 shows 3.3V SDIO electrical parameters

Table2- 13. 3.3V SDIO Electrical Parameters

Symbol	Parameter	Min	Typ	Max	Unit
VDD	Power voltage	2.7	-	3.6	V
V_{CCQ}	I/O voltage	2.7		3.6	V
V_{OH}	Output high-level voltage	0.75* V_{CCQ}	-	-	V
V_{OL}	Output low-level voltage	-	-	0.125* V_{CCQ}	V
V_{IH}	Input high-level voltage	0.625* V_{CCQ}	-	V_{CCQ} + 0.3	V

V_{IL}	Input low-level voltage	V_{SS}- 0.3	-	0.25* V_{CCQ}	V

Table 2-14 shows 1.8V SDIO electrical parameters.

Table2- 14. 1.8V SDIO Electrical Parameters

Symbol	Parameter	Min	Typ	Max	Unit
VDD	Power voltage	2.7	-	3.6	V
V_{CCQ}	I/O voltage	1.7		1.95	V
V_{OH}	Output HIGH voltage	V_{CCQ} -0.45	-	-	V
V_{OL}	Output LOW voltage	-	-	0.45	V
V_{IH}	Input HIGH voltage	0.625* V_{CCQ} (1)	-	V_{CCQ} + 0.3	V
V_{IL}	Input LOW voltage	V_{SS} - 0.3	-	0.35* V_{CCQ} (2)	V

NOTE

(1). 0.7 * VDD for MMC4.3 or lower.

(2). 0.3 * VDD for MMC4.3 or lower.

2.3.10 Audio Codec Electrical Parameters

Table 2-15 to Table 2-17 show audio codec electrical parameters.

Test Conditions:

VDD-SYS = 0.9V, AVCC=1.8V, TA=25°C, 1kHz sinusoid signal, fs = 48kHz, 16-bit audio data unless otherwise stated.

Table2- 15. Audio Codec Electrical Parameters

Test Path	Parameter	Test Conditions	Min	Typ	Max	Unit
DAC Path	**DAC to Headphone on HPL or HPR(R=10kΩ, CPVDD =1.2V 1POWER)**					
	Full-scale	0dB 1KHz	-	565	-	mVrms
	SNR（A-weighted）	0dB 1KHz	-	95	-	dB
	THD+N	0dB 1KHz	-	-83	-	dB
	DAC to Headphone on HPL or HPR(R=16Ω, CPVDD = 1.2V 1POWER)					
	Full-scale Level	0dB 1kHz	-	420	-	mVrms
	SNR（A-weighted）	0dB 1kHz	-	95	-	dB
	THD+N(P0=11.0mW)	Full-scale Level	-	-40	-	dB
	THD+N(P0=10.0mW)	0dB 1kHz		-55		dB
	THD+N(P0=7.5mW)	0dB 1kHz	-	-75	-	dB
	Crosstalk	R_0dB_L_0data 1kHz/ L_0dB_R_0data 1kHz	-	-64	-	dB
	DAC to Headphone on HPL or HPR(R=32Ω, CPVDD = 1.2V 1POWER)					
	Full-scale Level	0dB 1kHz	-	544	-	mVrms
	SNR（A-weighted）	0dB 1kHz	-	95	-	dB
	THD+N(P0=9.26mW)	Full-scale Level	-	-50	-	dB
	THD+N(P0=7.5mW)	0dB 1kHz	-	-75	-	dB
	Crosstalk	R_0dB_L_0data 1kHz/ L_0dB_R_0data 1kHz	-	-74	-	dB
ADC Path	**MIC2 or MIC3 to ADC via ADC mixer**					
	Full-scale Level	1.6Vpp 1KHz 0dB	-	830	-	mFFS

	SNR(A-weighted)	1.6Vpp 1KHz 0dB	-	93	-	dB
	THD+N	1.6Vpp 1KHz 0dB	-	-74	-	dB
	SNR(A-weighted)	36.09mVpp 1KHz 33dB	-	75	-	dB
	THD+N	36.09mVpp 1KHz 33dB	-	-72	-	dB
	LINEINR to ADC via ADC mixer					
	Full-scale Level	1.5Vpp 1kHz 0dB	-	817	-	mFFS
	SNR（A-weighted）	1.5Vpp 1kHz 0dB	-	96	-	dB
	THD+N	1.5Vpp 1kHz 0dB	-	-86	-	dB
Bypass Path Performance	**LINEINR to HPR via output mixer**					
	Full-scale	1.6Vpp 1kHz 0dB	-	540	-	mVrms
	SNR（A-weighted）	1.6Vpp 1kHz 0dB	-	98	-	dB
	THD+N	1.6Vpp 1kHz 0dB	-	-92	-	dB

Table2- 16. Audio Codec Operation Mode

Operation Mode	Test Conditions	AVCC(1.8V)	CPVIN(1.8V)
0dB Playback to Headphone	16Ω load, Volume: 0x3F, 1Power, CPVDD = 0.9v	4.3mA	55mA
0dB Playback to Headphone	32Ω load, Volume: 0x3F, 1Power, CPVDD = 0.9v	4.3mA	35mA
0data Playback to Headphone	Volume: 0x3F, 1Power, CPVDD = 0.9v	4.3mA	11mA
MIC2 Record	0Vpp input, gain:42dB	4.2mA	NA
MIC3 Record	0Vpp input, gain:42dB	4.2mA	NA
Linein Record	1.5VPP input	3.0mA	NA
Sleep Mode	Headphone Detect Enable	120uA	1uA

Table2- 17. Power Specifications

Parameter	Min	Typ	Max	Unit	Note
AVCC	1.782	1.8	1.818	V	Relative to AGND
CPVIN	1.7	1.8	1.9	V	Analog power for LDO

2.4 PCB Design Recommendations

For details about PCB design recommendations, see the *A50 Hardware Design User Guide*.

2.5 Interface Timings

2.5.1 SDRAM Interface Timing

2.5.1.1 DDR3/DDR3L Parameters

Figure 2-10 shows the DDR3/DDR3L command and address timing diagram. The timing parameters for this diagram shows in Table 2-18.

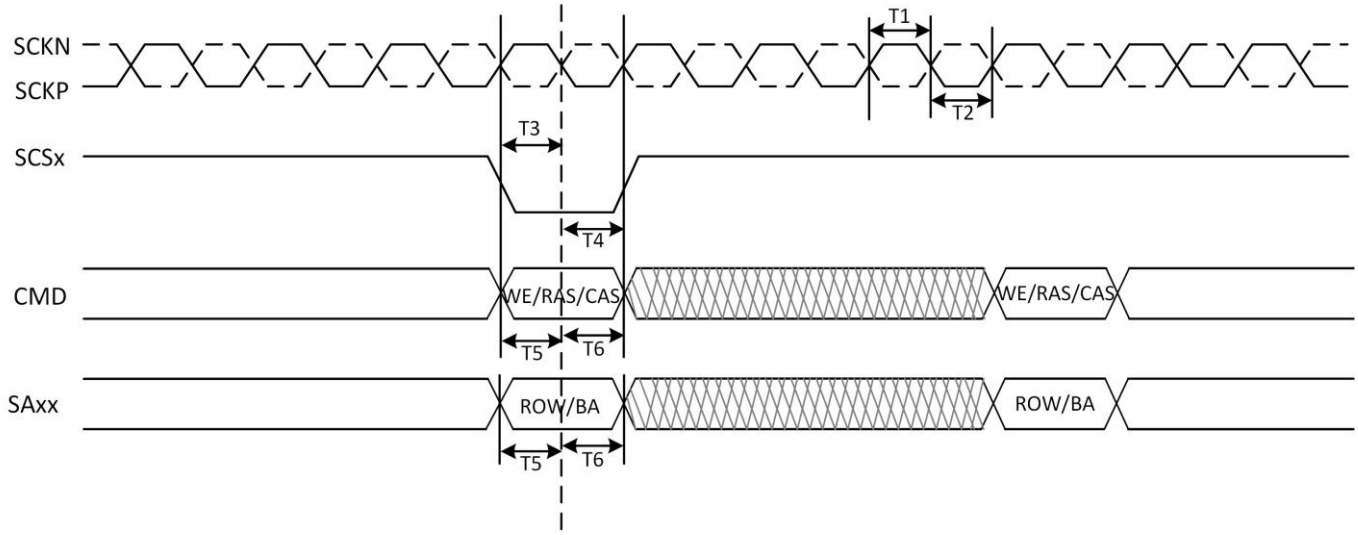


Figure2- 10. DDR3/DDR3L Command and Address Timing

Table2- 18. DDR3/DDR3L Timing Parameters

ID	Parameter	Symbol	Clock = 800 MHz			Unit
			Min	Suggest	Max	
T1	Clock high-level width	t_{CH}	0.47	-	0.53	tck
T2	Clock low-level width	t_{CL}	0.47	-	0.53	tck
T3	CS setup time	t_{IS}	170	295	-	ps
T4	CS hold time	t_{IH}	120	245	-	ps
T5	Command and Address setup time to Clock edge	t_{IS}	170	295	-	ps
T6	Command and Address hold time to Clock edge	t_{IH}	120	245	-	ps

T1 and T2 are in reference to Vref level.

T3,T4,T5, and T6 are in reference to Vih(ac) /Vil(ac) levels. (AC150/DC100).

Figure 2-11 shows the DDR3/DDR3L write timing diagram. The timing parameters for this diagram shows in Table 2-19.

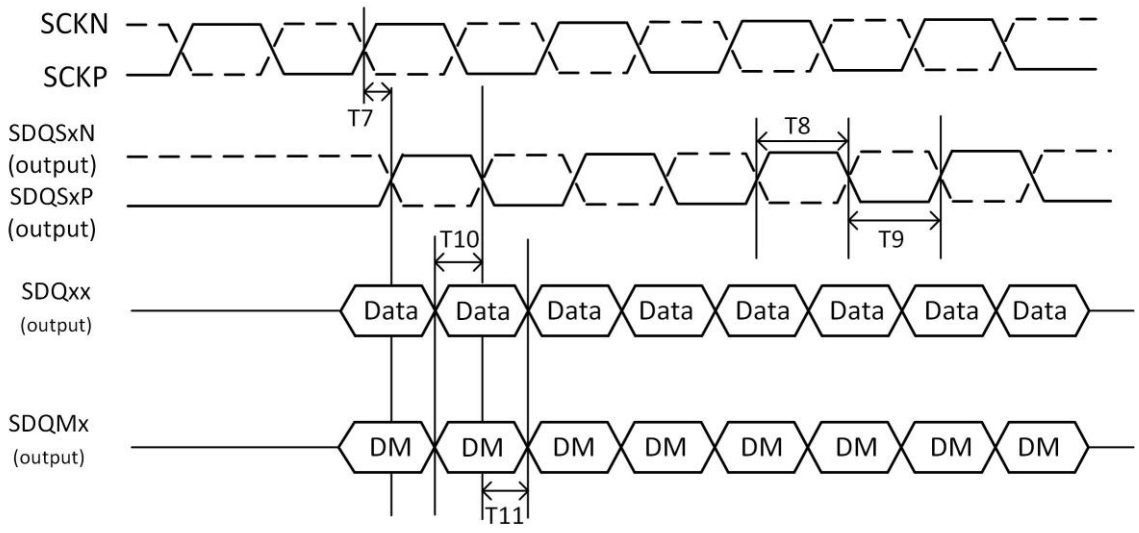


Figure2- 11. DDR3/DDR3L Write Cycle

Table2- 19. DDR3/DDR3L Write Cycle Parameters

ID	Parameter	Symbol	Clock = 800 MHz			Unit
			Min	Suggest	Max	
T7	SDQS rising edge to SCK rising edge	t_{DQSS}	-0.27	-	0.27	t_{CK}
T8	SDQS high level width	t_{DQSH}	0.45	-	0.55	t_{CK}
T9	SDQS low level width	t_{DQSL}	0.45	-	0.55	t_{CK}
T10	Data setup time to SDQS	t_{DS}	10	145	-	ps
T11	Data hold time to SDQS	t_{DH}	45	180	-	ps

To receive the reported setup and hold values, write calibration should be performed in order to locate the SDQSx in the middle of SDQxx window.

T7,T8, and T9 are in reference to Vref level.

T10 and T11 are in reference to Vih(ac) /Vil(ac) levels. (AC150/DC100).

Figure 2-12 shows the DDR3/DDR3L read timing diagram. The timing parameters for this diagram shows in Table 2-20.

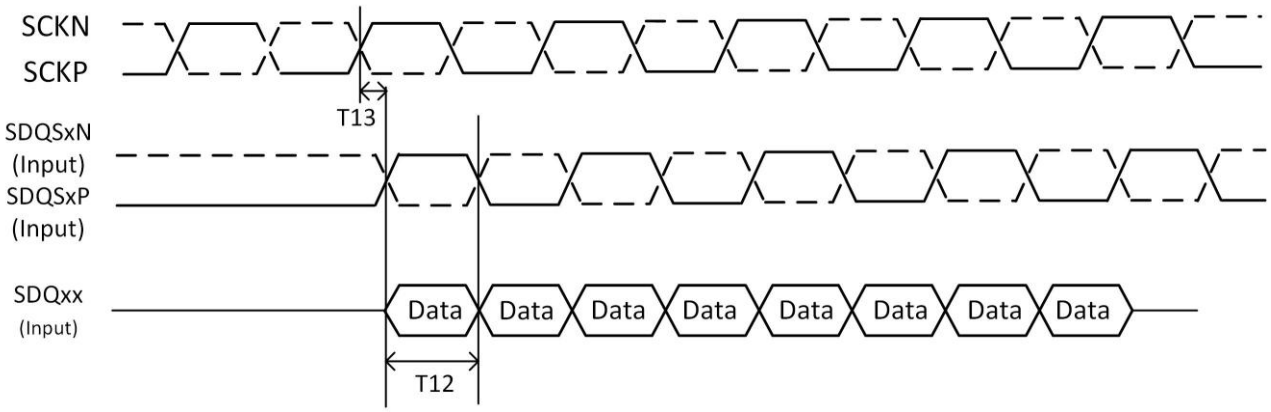


Figure2- 12. DDR3/DDR3L Read Cycle

Table2- 20. DDR3/DDR3L Read Cycle Parameters

ID	Parameter	Symbol	Clock = 800 MHz		Unit
			Min	Max	
T12	Read Data valid width	t_{Data}	200	-	ps

T13	SDQS rising edge to SCK rising edge	t_{DQSCK}	-225	225	ps

T12 and T13 are in reference to Vref level.

2.5.1.2 LPDDR3 Parameters

Figure 2-13 shows the LPDDR3 command and address timing diagram. The timing parameters for this diagram shows in Table 2-21.

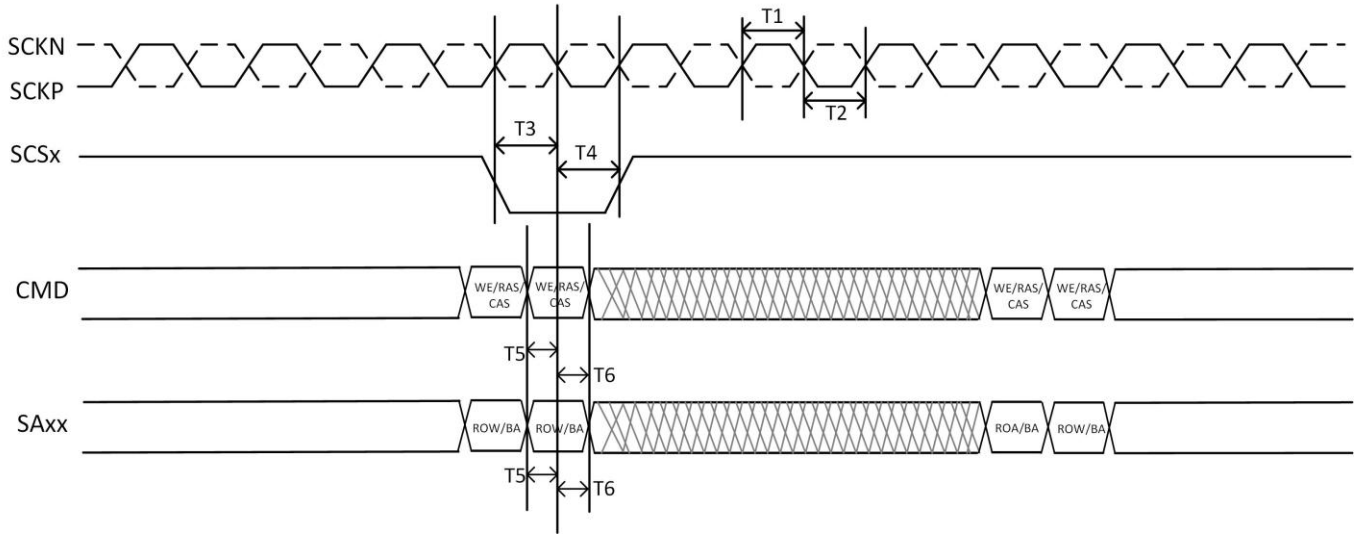


Figure2- 13. LPDDR3 Command and Address Timing Diagram

Table2- 21. LPDDR3 Command and Address Timing Parameters

ID	Parameter	Symbol	Clock = 800 MHz			Unit
			Min	Suggest	Max	
T1	Clock high pulse width	t_{CH}	0.45	-	0.55	t_{CK}
T2	Clock low pulse width	t_{CL}	0.45	-	0.55	t_{CK}
T3	SCSx input setup time	t_{ISCS}	195	347.5	-	ps
T4	SCSx input hold time	t_{IHCS}	220	372.5	-	ps
T5	Address and control input setup time	t_{IAS}	75	152.5	-	ps
T6	Address and control input hold time	t_{IAH}	100	177.5	-	ps

T1 and T2 are in reference to Vref level.

T3,T4,T5, and T6 are in reference to Vih(ac) /Vil(ac) levels. (AC150/DC100).

Figure2-14 shows the LPDDR3 write timing diagram. The timing parameters for this diagram shows in Table2-22.

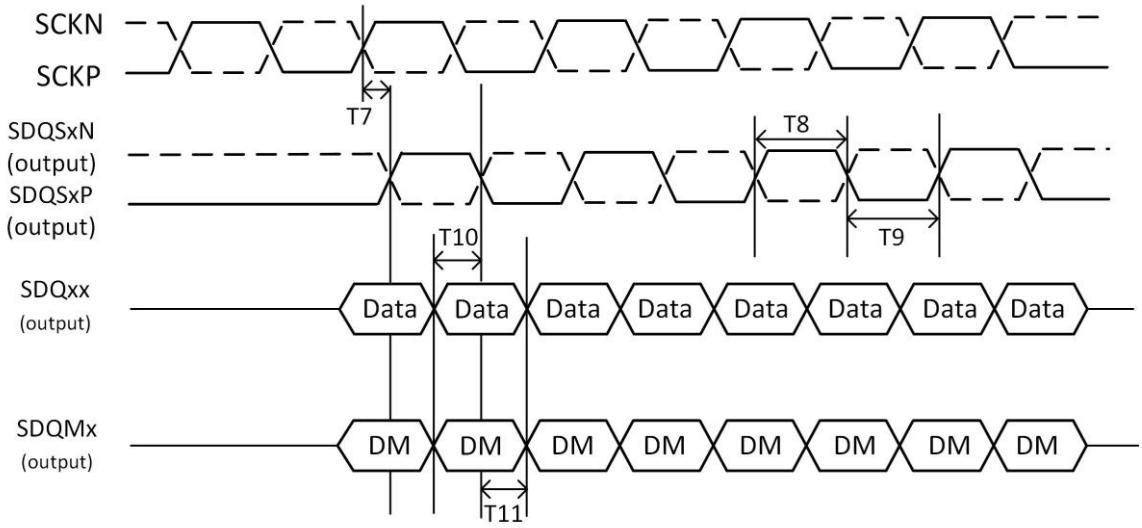


Figure2- 14. LPDDR3 Write Cycle

Table2- 22. LPDDR3 Write Cycle Parameters

ID	Parameter	Symbol	Clock = 800 MHz			Unit
			Min	Suggest	Max	
T7	SDQS rising edge to SCK rising edge	t_{DQSS}	0.75	-	1.25	t_{CK}
T8	SDQS input high-level width	t_{DQSH}	0.4	-	-	t_{CK}
T9	SDQS input low-level width	t_{DQSL}	0.4	-	-	t_{CK}
T10	SDQxx and SDQMx input setup time	t_{DS}	75	152.5	-	ps
T11	SDQxx and SDQMx input hold time	t_{DH}	100	177.5	-	ps

To receive the reported setup and hold values, write calibration should be performed in order to locate the SDQSx in the middle of SDQxx window.

T7,T8, and T9 are in reference to Vref level.

T10 and T11 are in reference to Vih(ac) /Vil(ac) levels. (AC150/DC100).

Figure2-15 shows the LPDDR3 read timing diagram. The timing parameters for this diagram shows in Table2-23.

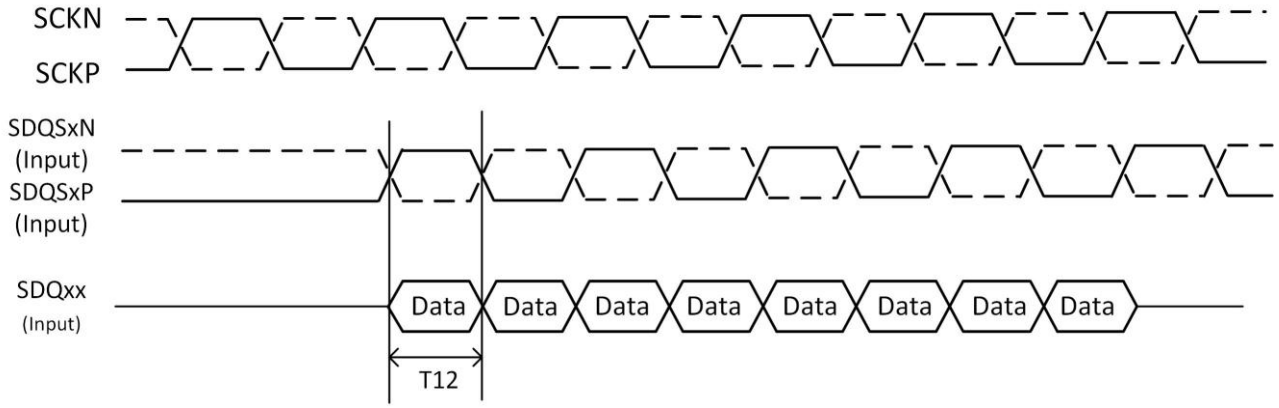


Figure2- 15. LPDDR3 Read Cycle

Table2- 23. LPDDR3 Read Cycle Parameters

ID	Parameter	Symbol	Clock = 800 MHz		Unit
			Min	Max	

T12	Read Data valid width	t_{DATA}	200	-	ps

T12 is in reference to Vref level.

2.5.1.3 DDR4 Parameters

Figure2-16 shows the DDR4 command and address timing diagram. The timing parameters foe this diagram shows in Table2-24.

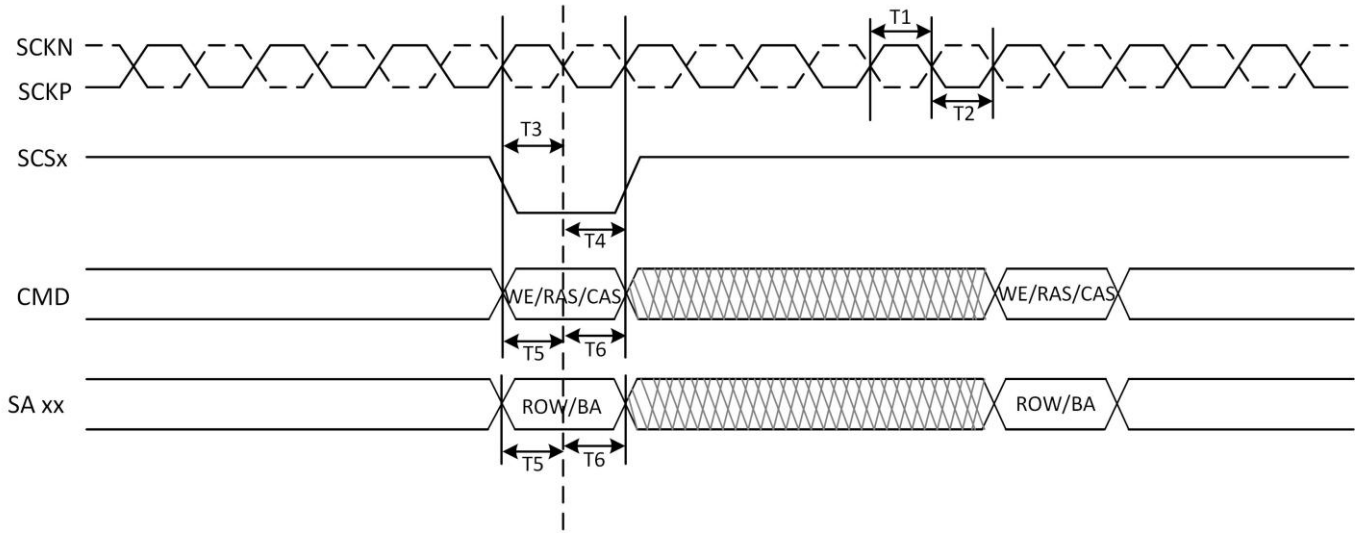


Figure2- 16. DDR4 Command and Address Timing

Table2- 24. DDR4 Timing Parameters

ID	Parameter	Symbol	Clock = 800 MHz			Unit
			Min	Suggest	Max	
T1	Clock high-level width	t_{CH}	0.48	-	0.52	tck
T2	Clock low-level width	t_{CL}	0.48	-	0.52	tck
T3	CS setup time	t_{IS}	115	-	-	ps
T4	CS hold time	t_{IH}	140	-	-	ps
T5	Command and Address setup time to CK	t_{IS}	115	-	-	ps
T6	Command and Address hold time to CK	t_{IH}	140	-	-	ps

T1~T2 are in reference to Vref level.

T3/T5 are in reference to Vih(ac) / Vil(ac) levels.(AC100)

T4/T6 are in reference to Vih(ac) / Vil(ac) levels.(DC75)

Figure2-17 shows the DDR4 write timing diagram. The timing parameter for this diagram shows in Table2-25.

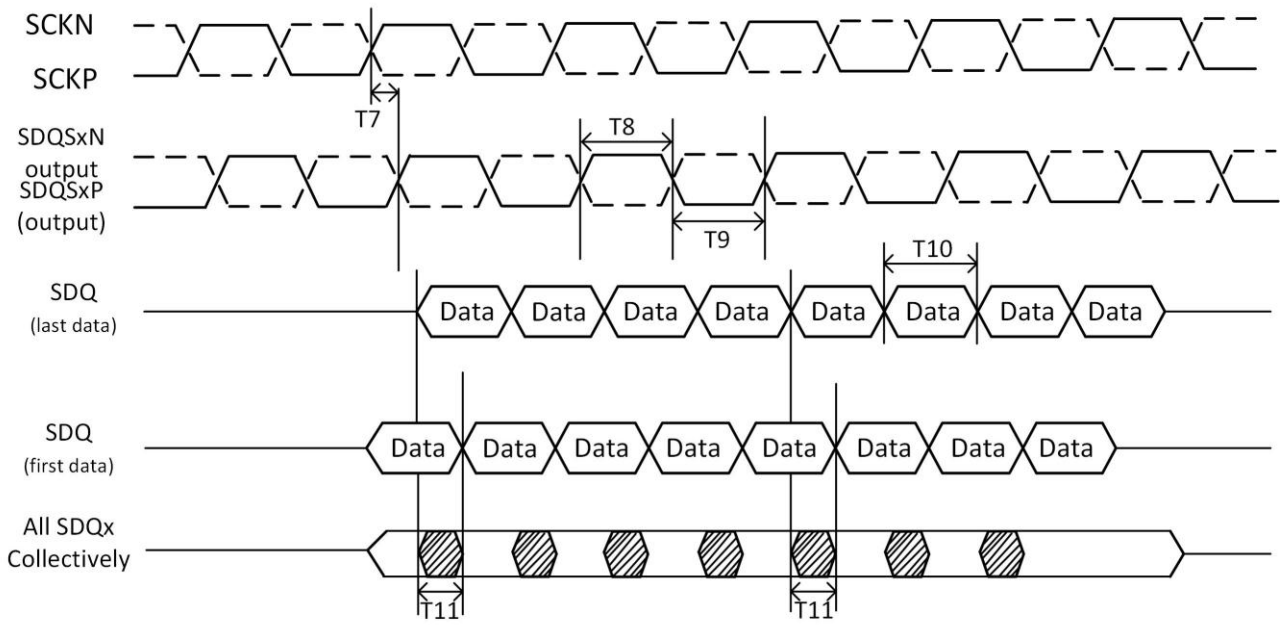


Figure2- 17. DDR4 Write Cycle

Table2- 25. DDR4 Write Cycle Parameters

ID	Parameter	Symbol	Clock = 800 MHz			Unit
			Min	Suggest	Max	
T7	SDQS rising edge to SCK rising edge	t_{DQSS}	-0.27	-	0.27	t_{CK}
T8	SDQS high level width	t_{DQSH}	0.46	-	0.54	t_{CK}
T9	SDQS low level width	t_{DQSL}	0.46	-	0.54	t_{CK}
T10	Data Valid Window per pin per UI	t_{DVWp}	0.66	-	-	t_{ui}
T11	Data Valid Window per device per UI	t_{DVWd}	0.63	-	-	t_{ui}

T7~T9 are in reference to Vref level.

T10 is Data Valid Window per pin per UI and is derived from (t_{QH} - t_{DQSQ}) of each UI on a pin of a given DRAM.

T11 is the Data Valid Window per device per UI and is derived from (t_{QH} - t_{DQSQ}) of each UI on a given DRAM

Figure2-18 shows the DDR4 read timing diagram. The timing parameters for this diagram shows in Table2-26.

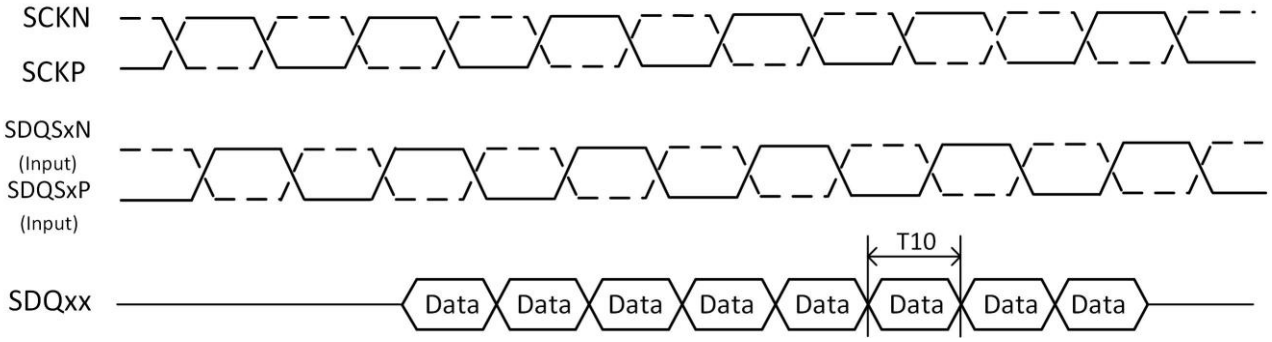


Figure2- 18. DDR4 Read Cycle

Table2- 26. DDR4 Read Cycle Parameters

ID	Parameter	Symbol	Clock = 800 MHz		Unit
			Min	Max	
T10	Data Valid Window per pin per UI	t_{DVWp}	0.66	-	t_{ui}

T10 is in reference to Vref level.

2.5.1.4 LPDDR4 Parameters

Figure2-19 shows the LPDDR4 command and address timing diagram. The timing parameters for this diagram shows in Table2-27.

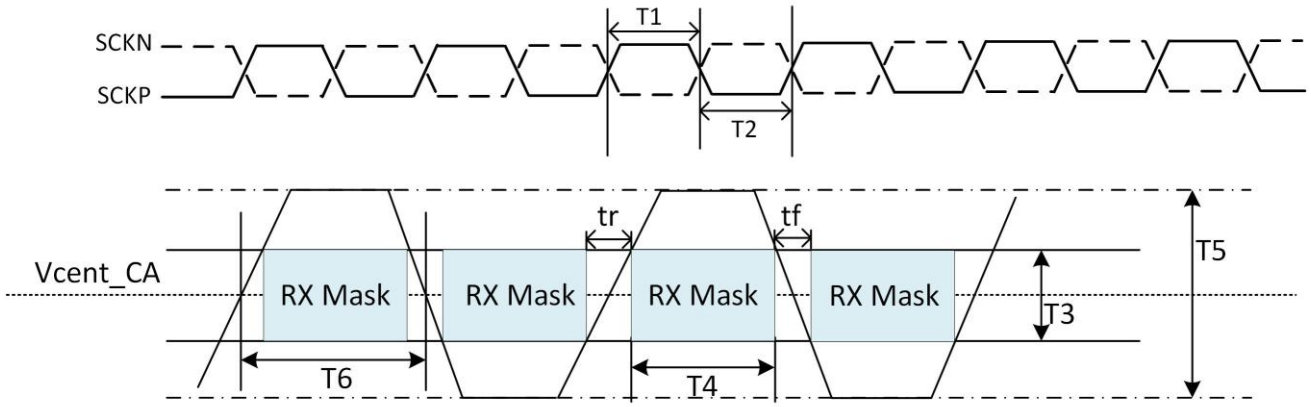


Figure2- 19. LPDDR4 Command and Address Timing Diagram

NOTE

T7 = T3/(tr or tf), signal must be monotonic within tr and tf range.

Table2- 27. LPDDR4 Command and Address Timing Parameters

ID	Parameter	Symbol	Clock = 800 MHz			Unit
			Min	Suggest	Max	
T1	Clock high pulse width	t_{CH}	0.46	-	0.54	t_{CK}
T2	Clock low pulse width	t_{CL}	0.46	-	0.54	t_{CK}
T3	Rx Mask voltage - p-p	V_{cIVW}	-	-	175	mV
T4	Rx timing window	T_{cIVW}	-	-	0.3	UI
T5	CA AC input pulse amplitude pk-pk	V_{IHL_AC}	210	-	-	mV
T6	CA input pulse width	T_{cIPW}	0.55	-	-	UI
T7	Input Slew Rate over VcIVW	SR_{IN_cIVW}	1	TBD	7	V/ns

Figure2-20 shows the LPDDR4 write timing diagram. The timing parameters for this diagram shows in Table2-28.

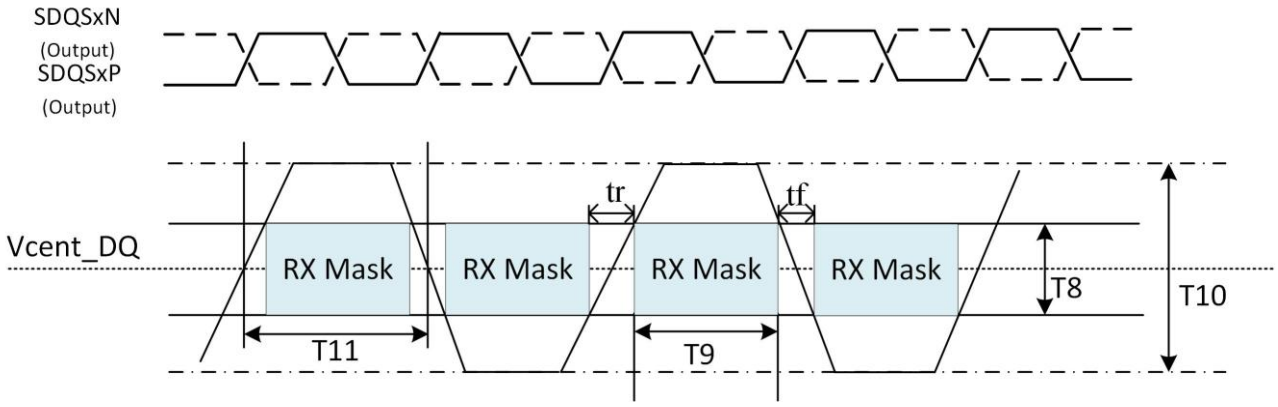


Figure2- 20. LPDDR4 Write Cycle

NOTE

T12 = T8/(tr or tf), signal must be monotonic within tr and tf range.

Table2- 28. LPDDR4 Write Cycle Parameters

ID	Parameter	Symbol	Clock = 800 MHz			Unit
			Min	Suggest	Max	
T8	Rx Mask voltage - p-p total	V_{dIVW_total}		-	140	mV
T9	Rx timing window total (At VdIVW voltage levels)	T_{dIVW_total}		-	0.22	UI
T10	DQ AC input pulse amplitude pk-pk	V_{IHL_AC}	180	-		mV
T11	Input pulse width (At Vcent_DQ)	$T_{dIPW\ DQ}$	0.45	-		UI
T12	Input Slew Rate over VdIVW_total	SR_{IN_dIVW}	1	TBD	7	V/ns

Figure2-21 shows the LPDDR4 read timing diagram. The timing parameters for this diagram shows in Table2-29.

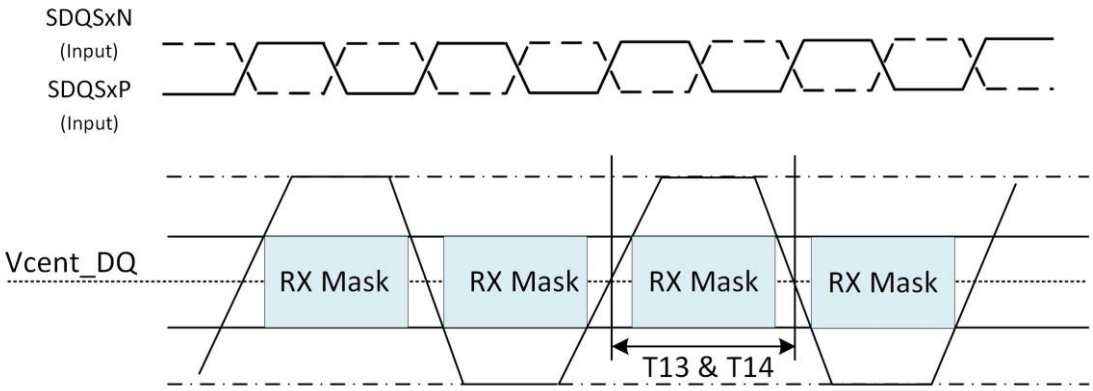


Figure2- 21. LPDDR4 Read Cycle

Table2- 29. LPDDR4 Read Cycle Parameters

ID	Parameter	Symbol	Clock = 800 MHz		Unit
			Min	Max	
T13	DQ output window time total, per pin (DBI-Disabled)	t_{QW_total}	0.75	-	UI
T14	DQ output window time total, per pin (DBI-Enabled)	$t_{QW\_total\_DBI}$	0.75	-	UI

2.5.2 NDFC Interface Timing

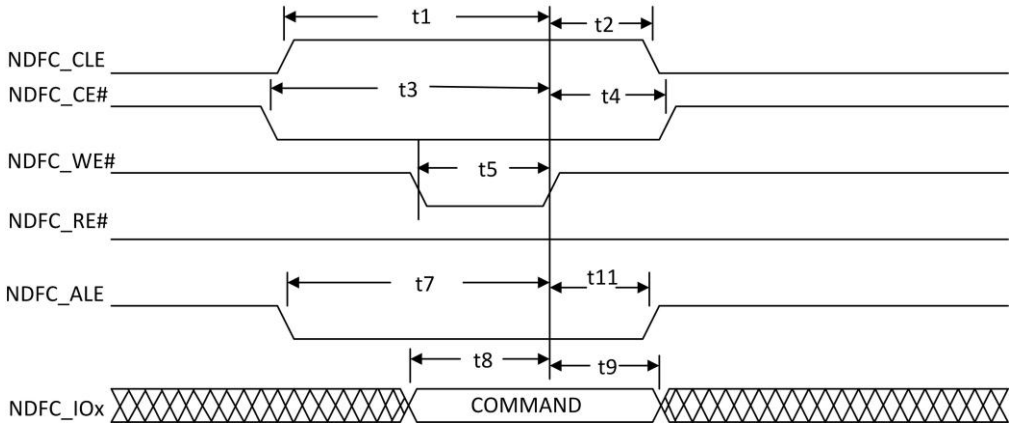


Figure2- 22. Command Cycle Timing

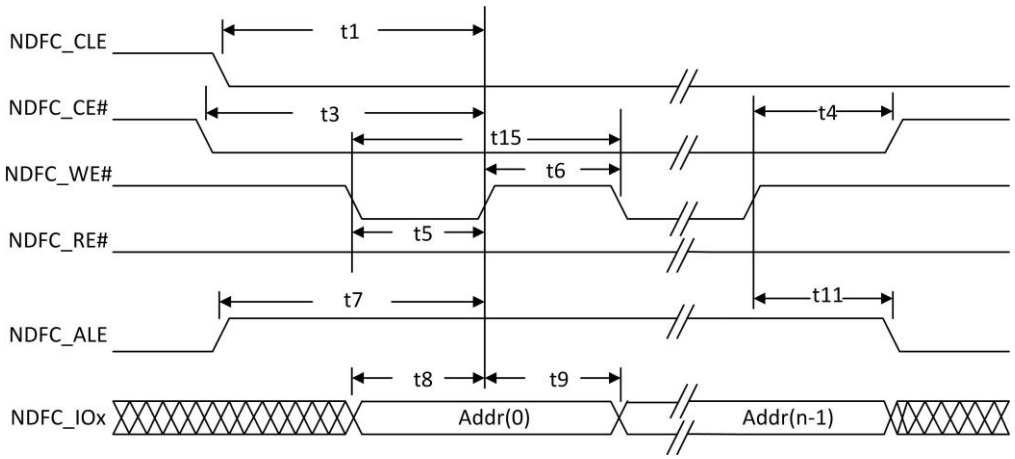


Figure2- 23. Address Cycle Timing

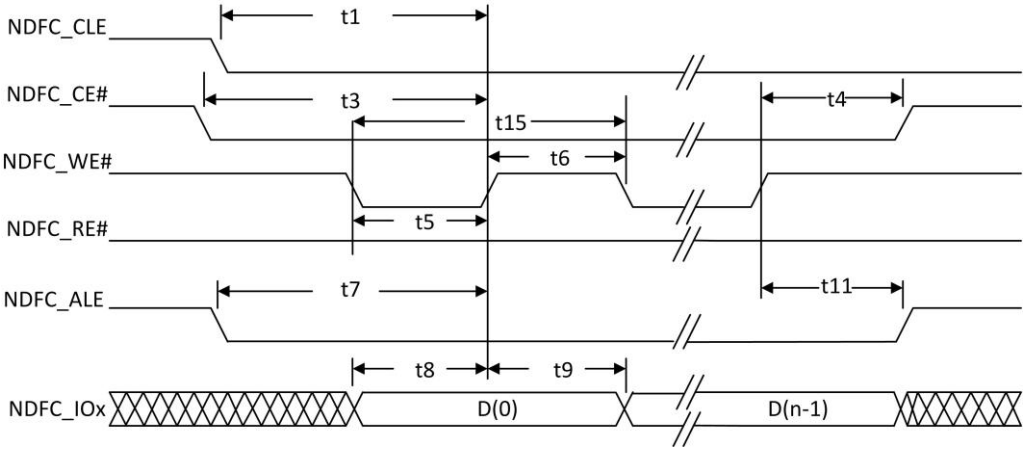


Figure2- 24. Write Data to Flash Cycle Timing

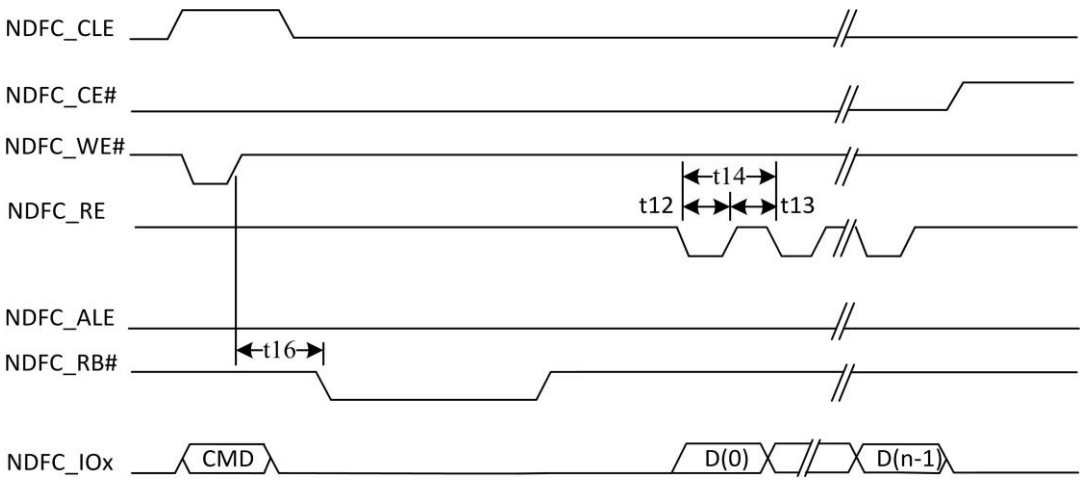


Figure2- 25. Waiting R/B# Ready Timing

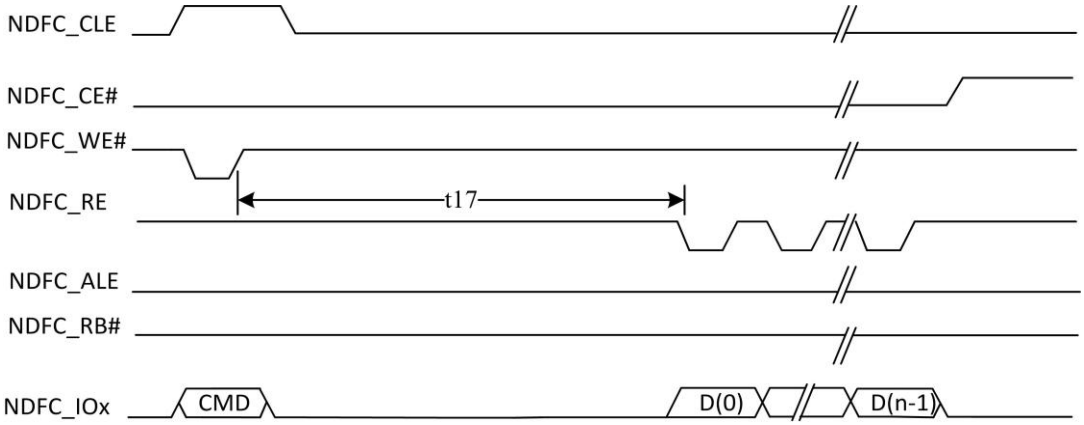


Figure2- 26. WE# High to RE# Low Timing

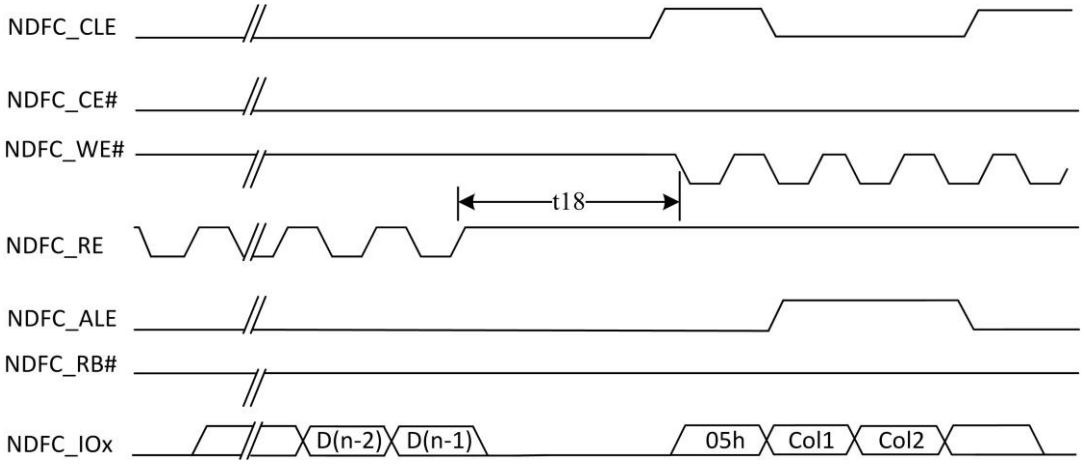


Figure2- 27. RE# High to WE# Low Timing

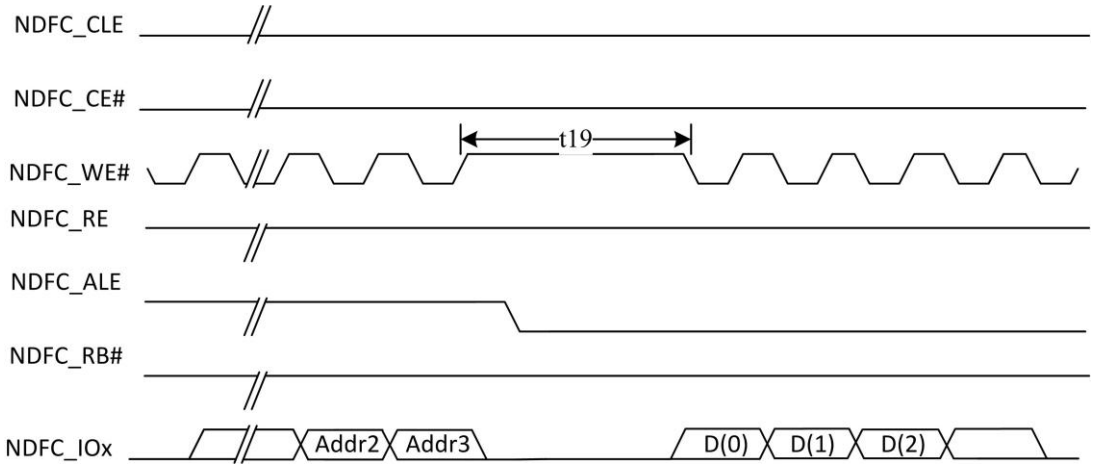


Figure2- 28. Address to Data Loading Timing

Table2- 30. Nand Timing Parameters

Parameter	Symbol	Timing	Unit
NDFC_CLE setup time	t1	2T	ns
NDFC_CLE hold time	t2	2T(1)	ns
NDFC_CE setup time	t3	2T	ns
NDFC_CE hold time	t4	2T	ns
NDFC_WE# pulse width	t5	T	ns
NDFC_WE# hold time	t6	T	ns
NDFC_ALE setup time	t7	2T	ns
Data setup time	t8	T	ns
Data hold time	t9	T	ns
Ready to NDFC_RE# low	t10	3T	ns
NDFC_ALE hold time	t11	2T	ns
NDFC_RE# pulse width	t12	T	ns
NDFC_RE# hold time	t13	T	ns
Read cycle time	t14	2T	ns
Write cycle time	t15	2T	ns
NDFC_WE# high to R/B# busy	t16	T_WB(2)	ns
NDFC_WE# high to NDFC_RE# low	t17	T_WHR(3)	ns
NDFC_RE# high to NDFC_WE# low	t18	T_RHW(4)	ns
Address to Data Loading time	t19	T_ADL(5)	ns
NOTE (1):T is the cycle of clock. **NOTE (2),(3),(4),(5):**This values is configurable in Nand Flash Controller. The value of T_WB could be 28T/44T/60T/76T, the value of T_WHR could be 0T/12T/28T/44T, the value of T_RHW could be 8T/24T/40T/56T, the value of T_ADL could be 0T/12T/28T/44T.			

2.5.3 SMHC Interface Timing

2.5.3.1 SMHC0/1 Interface Timing

2.5.3.1.1 SDR Mode(<100MHz)

The contents of this section can be applied to DS, HS, SDR12, SDR25, SDR50, SDR104（<100MHz） speed mode.

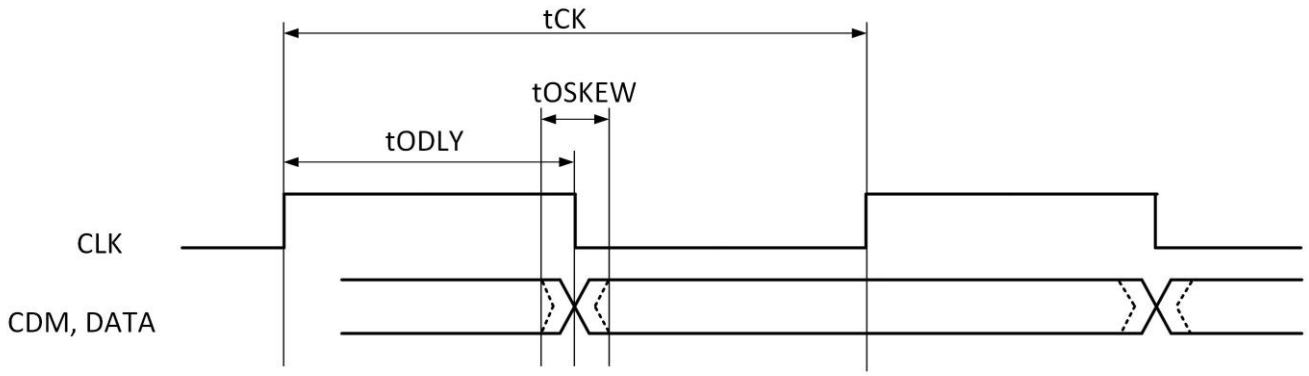


Figure2- 29. SMHC0/1 SDR Mode Output Timing Diagram

Table2- 31. SMHC0/1 SDR Mode Output Timing Parameters

Parameter	Symbol	Min	Typ	Max	Unit
CLK					
Clock frequency	tCK	0	50	50	MHz
Duty cycle	DC	45	50	55	%
Output CMD, DATA(referenced to CLK)					
CMD, Data output delay time	tODLY	-	0.25	0.5	UI
Data output delay skew time	tOSKEW	-	-	0.4	ns

NOTE

Unit Interval(UI) is one bit nominal time. For example, UI=20ns at 50MHz. The GPIO's driver strength level is 2 for test.

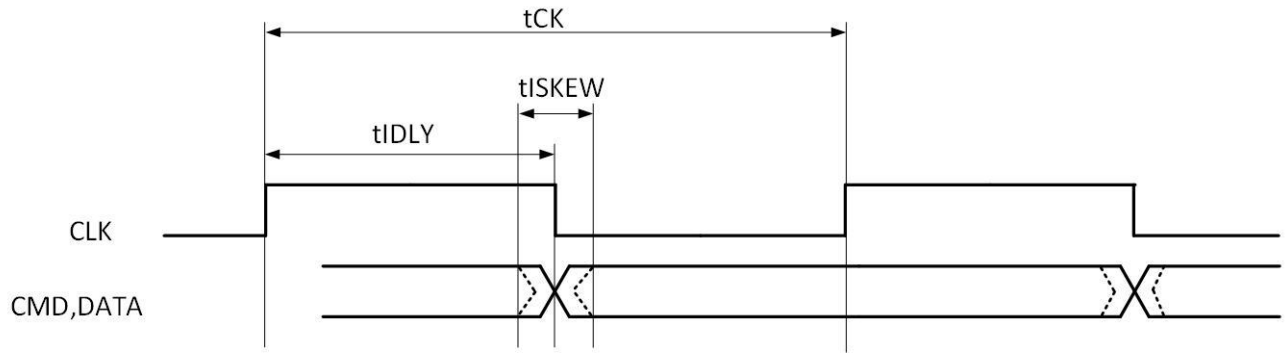


Figure2- 30. SMHC0/1 SDR Mode Input Timing Diagram

Table2- 32. SMHC0/1 SDR Mode Input Timing Parameters

Parameter	Symbol	Min	Typ	Max	Unit
CLK					
Clock frequency	tCK	0	50	50	MHz
Duty cycle	DC	45	50	55	%
Input CMD, DATA(referenced to CLK 50MHz)					
Data input delay in SDR mode. It includes Clock's PCB delay time, Data's PCB delay time and device's data output delay	tIDLY	-	-	20	ns
Data input skew time in SDR mode	tISKEW	-	-	1	ns
NOTE **The GPIO's driver strength level is 2 for test.**					

2.5.3.1.2 DDR50 Mode

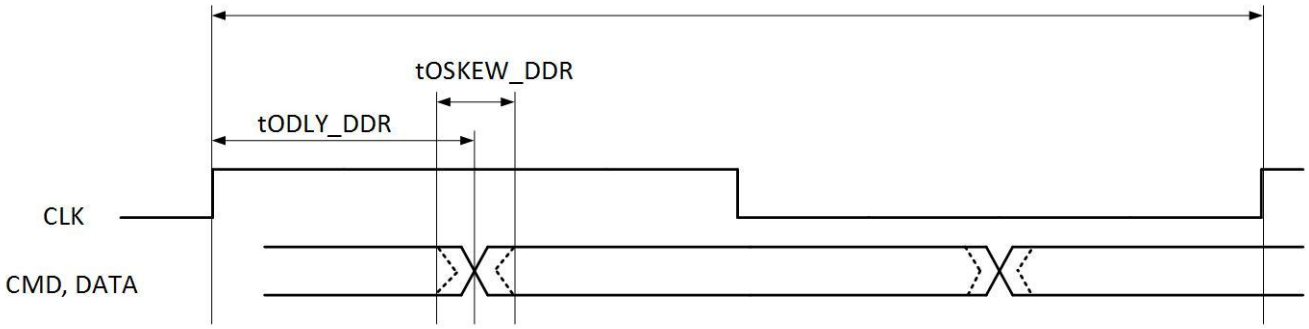


Figure2- 31. SMHC0/1 DDR50 Mode Output Timing Diagram

Table2- 33. SMHC0/1 DDR50 Mode Output Timing Parameters

Parameter	Symbol	Min	Typ	Max	Unit
CLK					
Clock frequency	tCK	0	50	50	MHz
Duty cycle	DC	45	50	55	%
Output CMD, DATA(referenced to CLK)					
CMD, Data output delay time in DDR mode	tODLY	-	0.25	0.25	UI
Data output delay skew time	tOSKEW	-	-	0.4	ns
NOTE **Unit Interval(UI) is one bit nominal time. For example, UI=20ns at 50MHz. The GPIO's driver strength level is 2 for test.**					

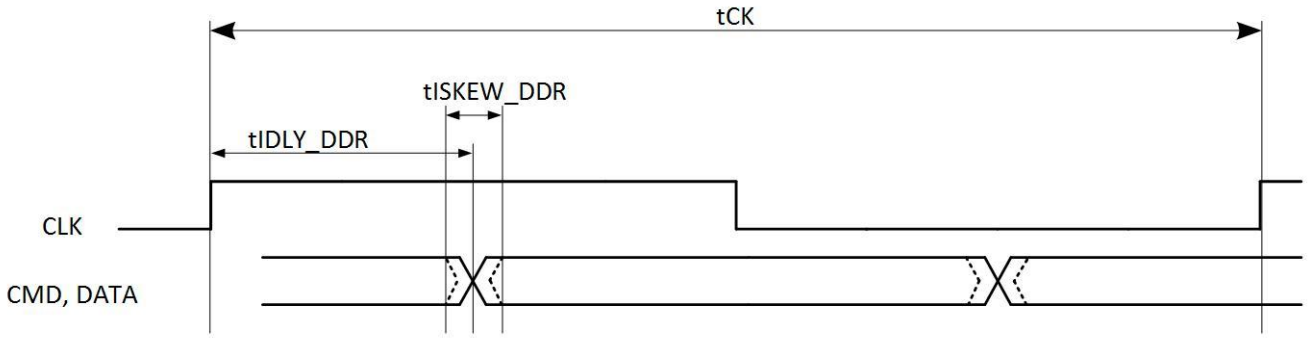


Figure2- 32. SMHC0/1 DDR50 Mode Input Timing Diagram

Table2- 34. SMHC0/1 DDR50 Mode Input Timing Parameters

Parameter	Symbol	Min	Typ	Max	Unit
CLK					
Clock frequency	tCK	0	50	50	MHz
Duty cycle	DC	45	50	55	%
Input CMD, DATA(referenced to CLK 50MHz)					
Data input delay in DDR mode. It includes Clock's PCB delay time, Data's PCB delay time and device's data output delay	tIDLY_DDR	-	-	-	ns
Data input skew time in DDR mode	tISKEW_DDR	-	-	-	ns
NOTE **The GPIO's driver strength level is 2 for test.**					

2.5.3.1.3 SDR104 Mode(>100MHz)

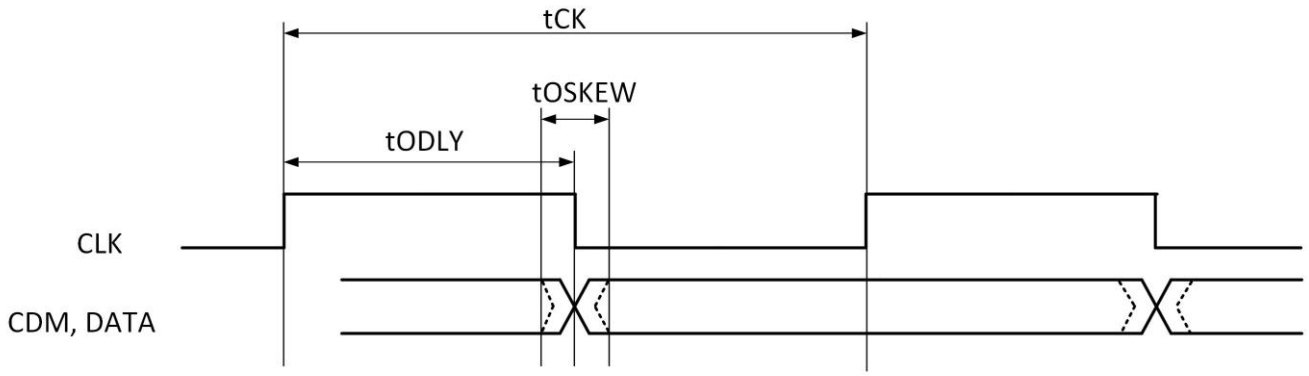


Figure2- 33. SMHC0/1 SDR104 Mode Output Timing Diagram

Table2- 35. SMHC0/1 SDR104 Mode Output Timing Parameters

Parameter	Symbol	Min	Typ	Max	Unit
CLK					
Clock frequency	tCK	0	-	150	MHz
Duty cycle	DC	45	50	55	%

Output CMD, DATA(referenced to CLK)					
CMD, Data output delay time	tODLY	-	0.25	0.5	UI
Data output delay skew time	tOSKEW	-	-	0.4	ns

NOTE

Unit Interval(UI) is one bit nominal time. For example, UI=20ns at 50MHz. The GPIO's driver strength level is 2 for test.

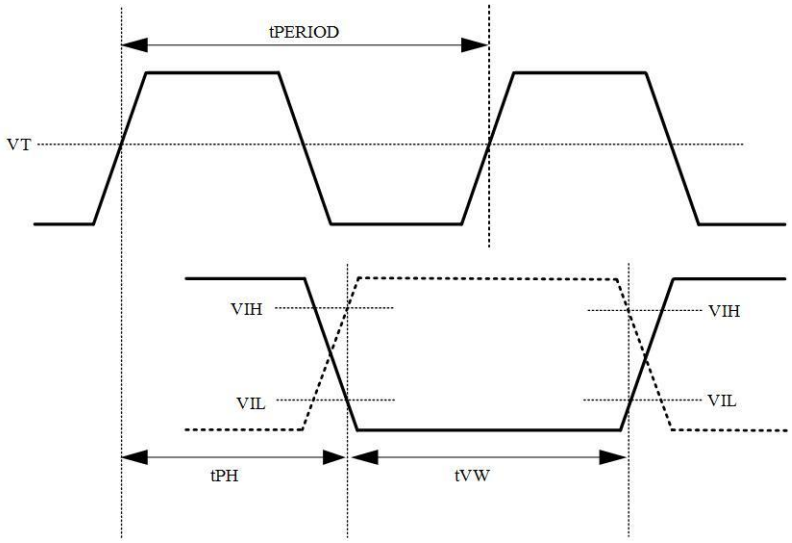


Figure2- 34. SMHC0/1 SDR-104 Mode Input Timing Diagram

Table2- 36. SMHC0/1 SDR-104 Mode Input Timing Parameters

Parameter	Symbol	Min	Typ	Max	Unit	Remark
CLK						
Clock period	tPERIOD	6.66	-	-	ns	Max:150MHz
Duty cycle	DC	45	50	55	%	
Rise time, fall time	tTLH, tTHL	-	-	0.2	UI	
Input CMD, DATA(referenced to CLK)						
Output delay	tPH	0	-	2	UI	
Output delay variation due to temperature change after tuning	dPH	-350[3]	-	1550[4]	ps	
CMD,Data valid window	tVW	0.575	-	-	UI	

NOTE(1): Unit Interval(UI)is one bit nominal time. For example, UI=10ns at 100MHz.
NOTE(2): The GPIO's driver strength level is 3 for test.
NOTE(3): Temperature variation: -20°C
NOTE(4): Temperature variation: 90°C

2.5.3.2 SMHC2 Interface Timing

2.5.3.2.1 HS-SDR/HS-DDR Mode

NOTE

IO volatage is 1.8V or 3.3V.

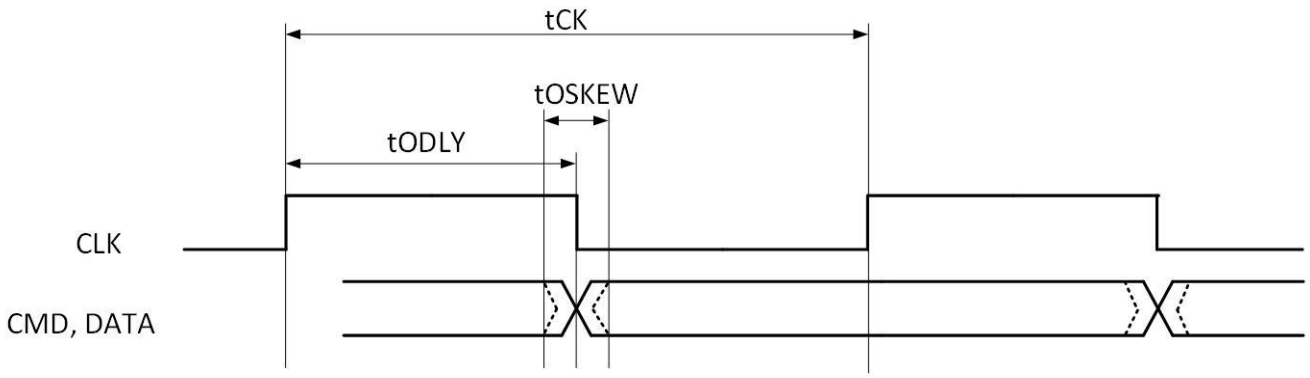


Figure2- 35. SMHC2 HS-SDR Mode Output Timing Diagram

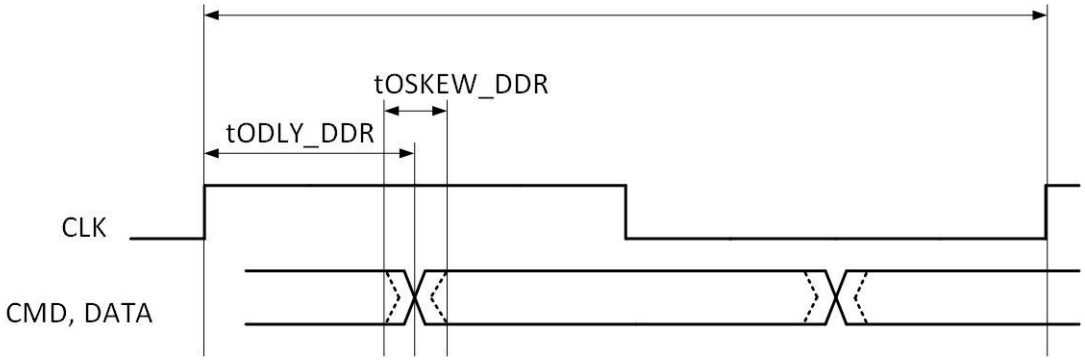


Figure2- 36. SMHC2 HS-DDR Mode Output Timing Diagram

Table2- 37. SMHC2 HS-SDR/HS-DDR Mode Output Timing Parameters

Parameter	Symbol	Min	Typ	Max	Unit	Remark
CLK						
Clock frequency	tCK	0	50	50	MHz	
Duty cycle	DC	45	50	55	%	
Output CMD, DATA(referenced to CLK)						
CMD, Data output delay time	tODLY	-	0.25	0.5	UI	
CMD, Data output delay time in DDR mode	tODLY_DDR	-	0.25	0.25	UI	
Data output delay skew time	tOSKEW	-	-		ns	

NOTE

Unit Interval(UI)is one bit nominal time. For example, UI=20ns at 50MHz.

The GPIO's driver strength level is 2 for test.

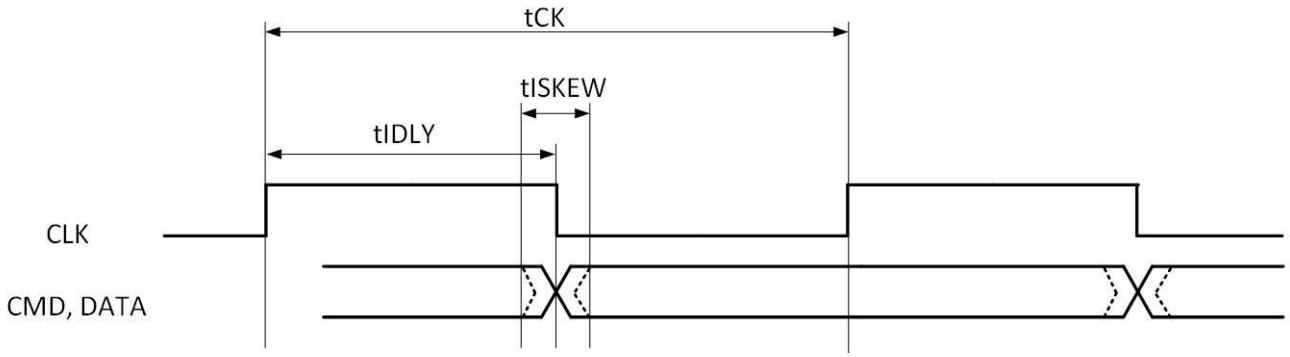


Figure2- 37. SMHC2 HS-SDR Mode Input Timing Diagram

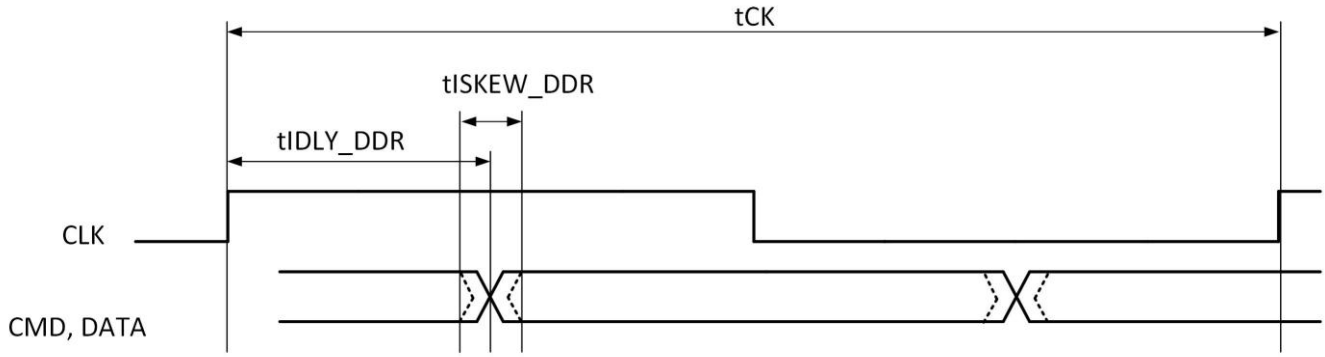


Figure2- 38. SMHC2 HS-DDR Mode Input Timing Diagram

Table2- 38. SMHC2 HS-SDR/HS-DDR Mode Input Timing Parameters

Parameter	Symbol	Min	Typ	Max	Unit	Remark
CLK						
Clock frequency	tCK	0	50	50	MHz	
Duty cycle	DC	45	50	55	%	
Input CMD, DATA(referenced to CLK 50MHz)						
Data input delay in SDR mode. It includes Clock's PCB delay time, Data's PCB delay time and device's data output delay	tIDLY	-	-		ns	
Data input delay in DDR mode.It includes Clock's PCB delay time, Data's PCB delay time and device's data output delay	tIDLY_DDR	-	-		ns	
Data input skew time in SDR mode	tISKEW	-	-		ns	
Data input skew time in DDR mode	tISKEW_DDR	-	-		ns	

NOTE

The GPIO's driver strength level is 2 for test.

2.5.3.2.2 HS200 Mode

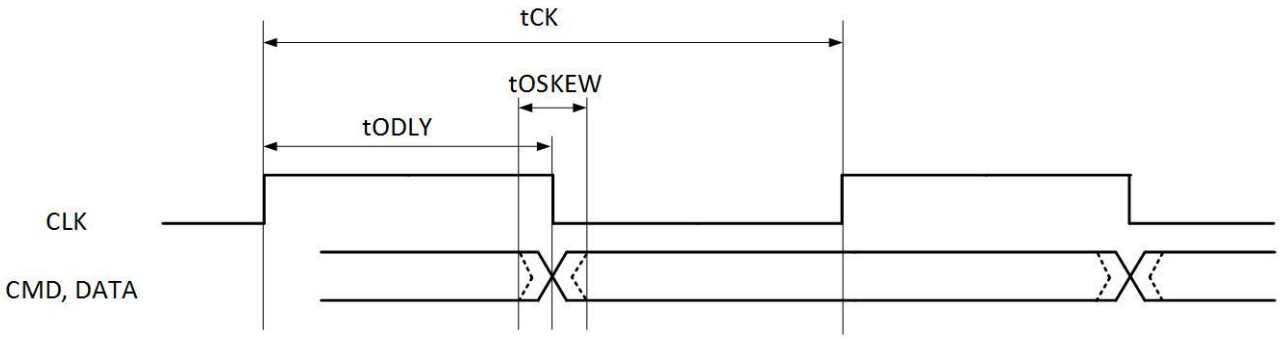


Figure2- 39. SMHC2 HS200 Mode Output Timing Diagram

Table2- 39. SMHC2 HS200 Mode Output Timing Parameters

Parameter	Symbol	Min	Typ	Max	Unit	Remark
CLK						
Clock frequency	tCK	-	-	150	MHz	
Duty cycle	DC	45	50	55	%	
Rise time, fall time	tTLH, tTHL	-	-	0.2	UI	
Output CMD, DATA(referenced to CLK)						
CMD, Data output delay time	tODLY	-	0.25	0.5	UI	
Data output delay skew time	tOSKEW	-	-		ns	

NOTE

Unit Interval(UI)is one bit nominal time. For example, UI=10ns at 100MHz.

The GPIO's driver strength level is 3 for test.

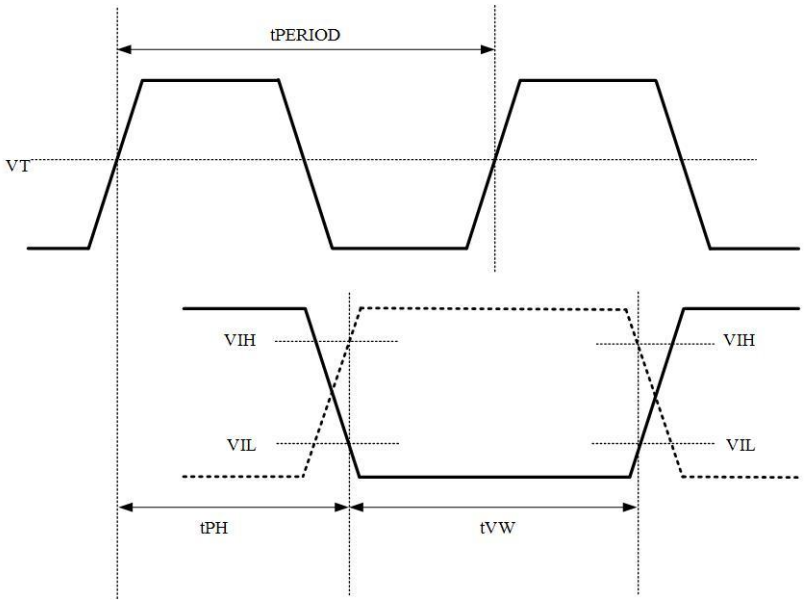


Figure2- 40. SMHC2 HS200 Mode Input Timing Diagram

Table2- 40. SMHC2 HS200 Mode Input Timing Parameters

Parameter	Symbol	Min	Typ	Max	Unit	Remark
CLK						
Clock period	tPERIOD	6.66	-	-	ns	Max:150MHz
Duty cycle	DC	45	50	55	%	
Rise time, fall time	tTLH, tTHL	-	-	0.2	UI	
Input CMD, DATA(referenced to CLK)						
Output delay	tPH	0	-	2	UI	
Output delay variation due to temperature change after tuning	dPH	-350[3]	-	1550[4]	ps	
CMD,Data valid window	tVW	0.575	-	-	UI	
NOTE (1): Unit Interval(UI)is one bit nominal time. For example, UI=10ns at 100MHz. **NOTE (2)**: The GPIO's driver strength level is 3 for test. **NOTE (3)**: Temperature variation: -20°C. **NOTE (4)**: Temperature variation: 90°C.						

2.5.3.2.3 HS400 Mode

The CMD output timing for HS400 mode is the same as CMD output timing for HS200 mode.

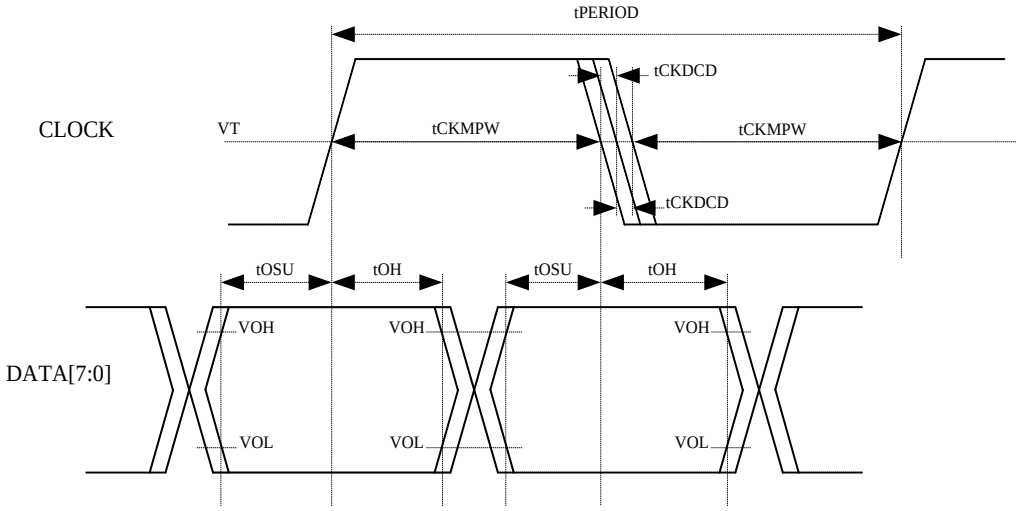


Figure2- 41. SMHC2 HS400 Mode Data Output Timing Diagram

Table2- 41. SMHC2 HS400 Mode Data Output Timing Parameters

Parameter	Symbol	Min	Typ	Max	Unit	Remark
CLK						
Clock period	tPERIOD	10	-	-	ns	Max:100MHz
Clock slew rate	SR	1.125	-	-	V/ns	
Clock duty cycle distortion	tCKDCD	0	-	0.5	ns	
Clock minimum pulse width	tCKMPW	2.2	-	-	ns	
Output DATA(referenced to CLK)						
Data output setup time	tOSU	0.4	-	-	ns	

Data output hold time	tOH	0.4	-	-	ns	
Data output slew rate	SR	0.9	-	-	ns	

NOTE

Unit Interval(UI)is one bit nominal time. For example, UI=10ns at 100MHz.

The GPIO's driver strength level is 3 for test.

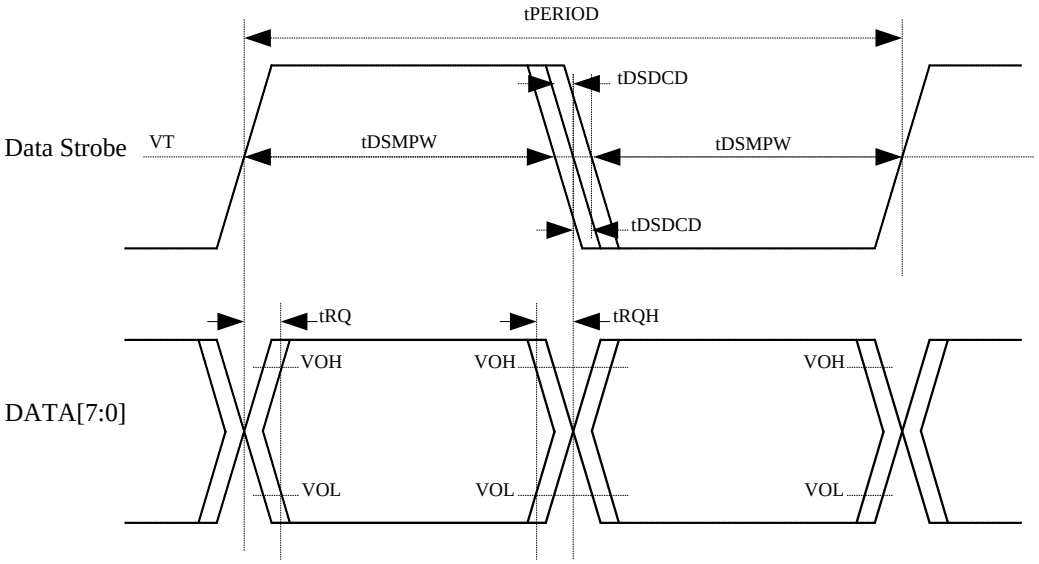


Figure2- 42. SMHC2 HS400 Mode Data Input Timing Diagram

Table2- 42. SMHC2 HS400 Mode Data Input Timing Parameters

Parameter	Symbol	Min	Typ	Max	Unit	Remark
DS(Data Strobe)						
DS period	tPERIOD	10	-	-	ns	Max:100MHz
DS slew rate	SR	1.125	-	-	V/ns	
DS duty cycle distortion	tDSDCD	0.0	-	0.4	ns	
DS minimum pulse width	tDSMPW	2.0	-	-	ns	
Output DATA(referenced to CLK)						
Data input skew	tRQ	-	-	0.4	ns	
Data input hold skew	tRQH	-	-	0.4	ns	
Data input slew rate	SR	0.85	-	-	V/ns	

NOTE

Unit Interval(UI)is one bit nominal time. For example, UI=10ns at 100MHz.

The GPIO's driver strength level is 3 for test.

2.5.4 LCD Interface Timing

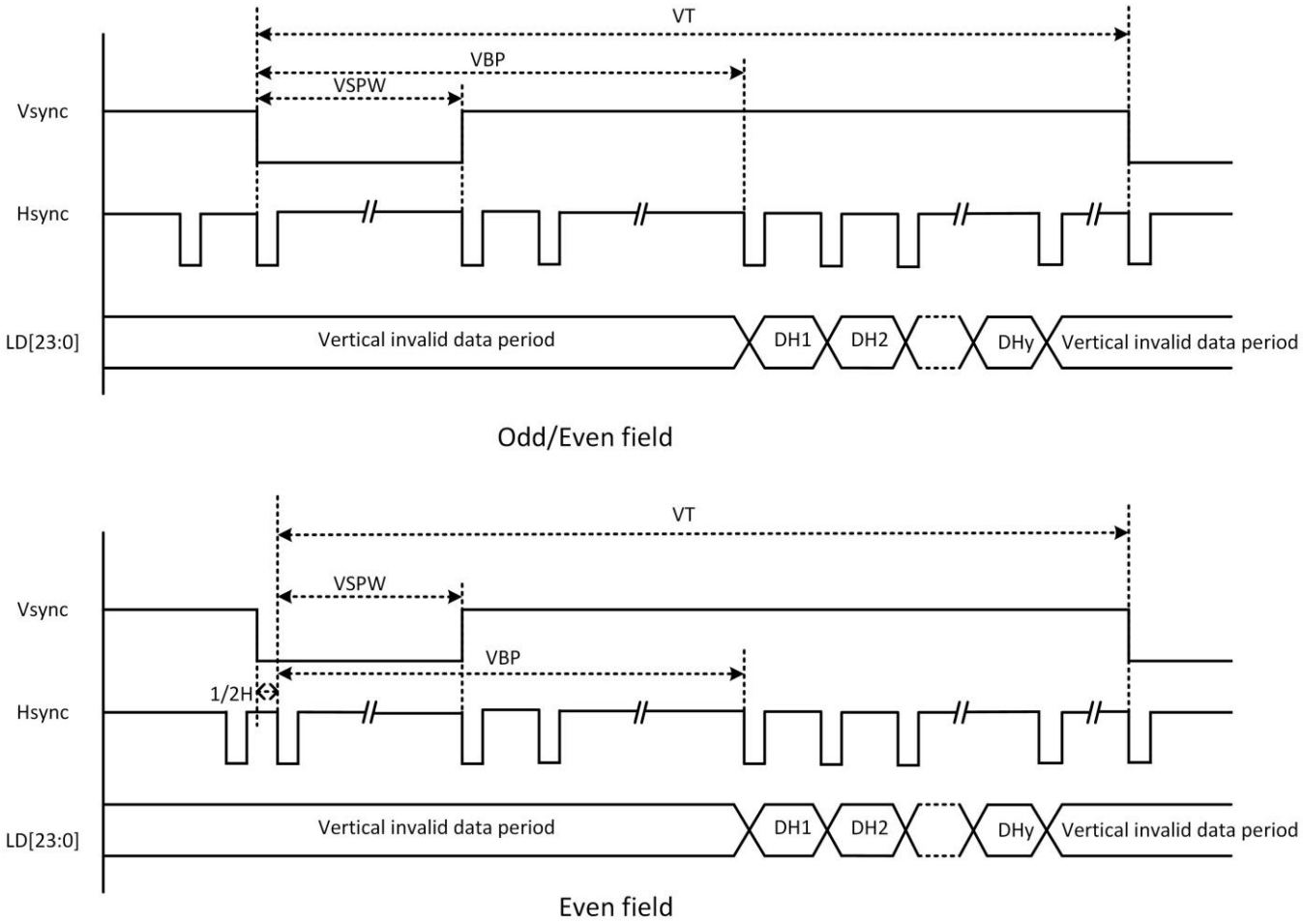


Figure2- 43. HV_IF Interface Vertical Timing

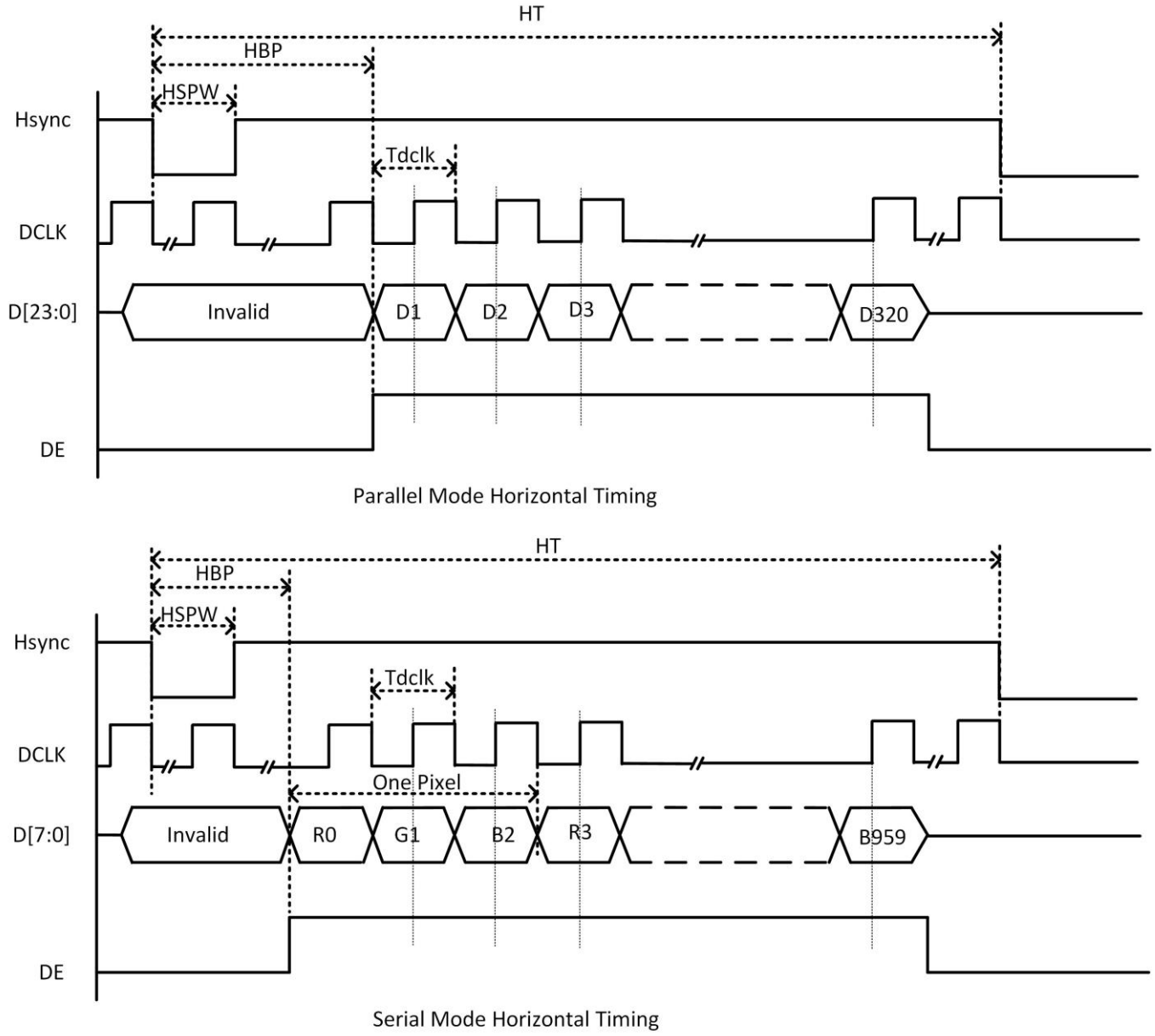


Figure2- 44. HV Interface Horizontal Timing

Table2- 43. HV Interface Timing Constants

Parameter	Symbol	Min	Typ	Max	Unit
DCLK cycle time	tDCLK	5	-	-	ns
HSYNC period time	tHT	-	HT+1	-	tDCLK
HSYNC width	tHSPW	-	HSPW+1	-	tDCLK
HSYNC back porch	tHBP	-	HBP+1	-	tDCLK
VSYNC period time	tVT	-	VT/2	-	tHT
VSYNC width	tVSPW	-	VSPW+1	-	tHT
VSYNC back porch	tVBP	-	VBP+1	-	tHT

NOTE

Vsync: Vertical sync, indicates one new frame

Hsync: Horizontal sync, indicates one new scan line

DCLK: Dot clock, pixel data are sync by this clock

LDE: LCD data enable **LD[23..0]: 24Bit RGB/YUV output from input FIFO for panel**

2.5.5 MIPI-CSI interface timing

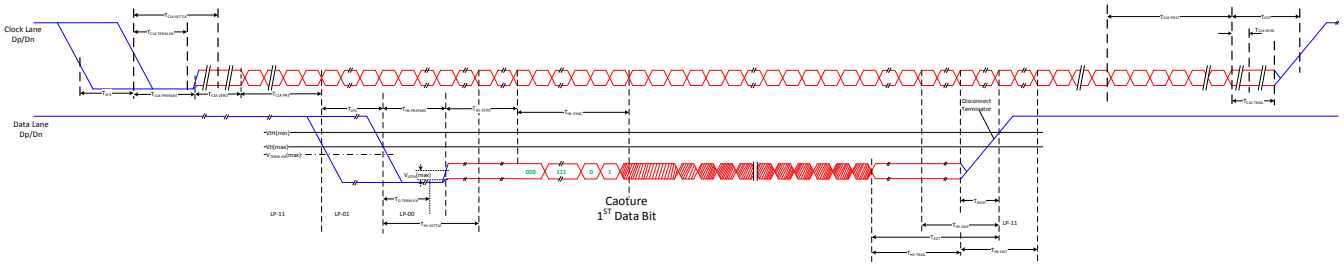


Figure2- 45. MIPI-CSI interface timing

Table2- 44. MIPI-CSI Interface Timing Contants(CLK Lane)

Parameter	Description	Min	Typ	Max	Unit	Notes
$T_{CLK-POST}$	Time that the transmitter continues to send HS clock after the last associated Data Lane has transitioned to LP Mode. Interval is defined as the period from the end of THS-TRAIL to the beginning of $T_{CLK-TRAIL}$.	60ns + 52*UI			ns	5
$T_{CLK-PRE}$	Time that the HS clock shall be driven by the transmitter prior to any associated Data Lane beginning the transition from LP to HS mode.	8			UI	5
$T_{CLK-PREPARE}$	Time that the transmitter drives the Clock Lane LP-00 Line state immediately before the HS-0 Line state starting the HS transmission.	38		95	ns	5
$T_{CLK-SETTLE}$	Time interval during which the HS receiver should ignore any Clock Lane HS transitions, starting from the beginning of TCLK-PREPARE.	95		300	ns	6,7
$T_{CLK-TERM-EN}$	Time for the Clock Lane receiver to enable the HS line termination, starting from the time point when Dn crosses $V_{IL,MAX}$.	Time for Dn to reach $V_{TERM-EN}$		38	ns	6
$T_{CLK-TRAIL}$	Time that the transmitter drives the HS-0 state after the last payload clock bit of a HS transmission burst.	60			ns	5
$T_{CLK-PREPARE}$ + $T_{CLK-ZERO}$	TCLK-PREPARE + time that the transmitter drives the HS-0 state prior to starting the Clock.	300			ns	5
T_{LPX}	Transmitted length of any Low-Power state period	50			ns	4,5

Table2- 45. MIPI CSI Interface Timing Contants(Data lane)

Parameter	Description	Min	Typ	Max	Unit	Notes
$T_{D\text{-}TERM\text{-}EN}$	Time for the Data Lane receiver to enable the HS line termination, starting from the time point when Dn crosses $V_{IL,MAX}$.	Time for Dn to reach $V_{TERM\text{-}EN}$		35ns + 4*UI		6
T_{EOT}	Transmitted time interval from the start of $T_{HS\text{-}TRAIL}$ or $T_{CLK\text{-}TRAIL}$, to the start of the LP-11 state following a HS burst.			105ns + n*12*UI		3.5
$T_{HS\text{-}EXIT}$	Time that the transmitter drives LP-11 following a HS burst.	100			ns	5
$T_{HS\text{-}PREPARE}$	Time that the transmitter drives the Data Lane LP-00 Line state immediately before the HS-0 Line state starting the HS transmission	40ns + 4*UI		85ns + 6*UI	ns	5
$T_{HS\text{-}PREPARE}$ + $T_{HS\text{-}ZERO}$	$T_{HS\text{-}PREPARE}$ + time that the transmitter drives the HS-0 state prior to transmitting the Sync sequence.	145ns + 10*UI			ns	5
$T_{HS\text{-}SETTLE}$	Time interval during which the HS receiver shall ignore any Data Lane HS transitions, starting from the beginning of $T_{HS\text{-}PREPARE}$. The HS receiver shall ignore any Data Lane transitions before the minimum value, and the HS receiver shall respond to any Data Lane transitions after the maximum value.	85ns + 6*UI		145ns + 10*UI	ns	6
$T_{HS\text{-}TRAIL}$	Time that the transmitter drives the flipped differential state after last payload data bit of a HS transmission burst	max(n*8*UI,60ns + n*4*UI)			ns	2,3,5

NOTE

1. The minimum value depends on the bit rate. Implementations should ensure proper operation for all the supported bit rates.

2. If a > b then max(a, b) = a otherwise max(a, b) = b.

3. Where n = 1 for Forward-direction HS mode and n = 4 for Reverse-direction HS mode.

4. T_{LPX} is an internal state machine timing reference. Externally measured values may differ slightly from the specified values due to asymmetrical rise and fall times.

5. Transmitter-specific parameter.

6. Receiver-specific parameter.

7. The stated values are considered informative guidelines rather than normative requirements since this parameter is untestable in typical applications.

2.5.6 I2S/PCM Interface Timing

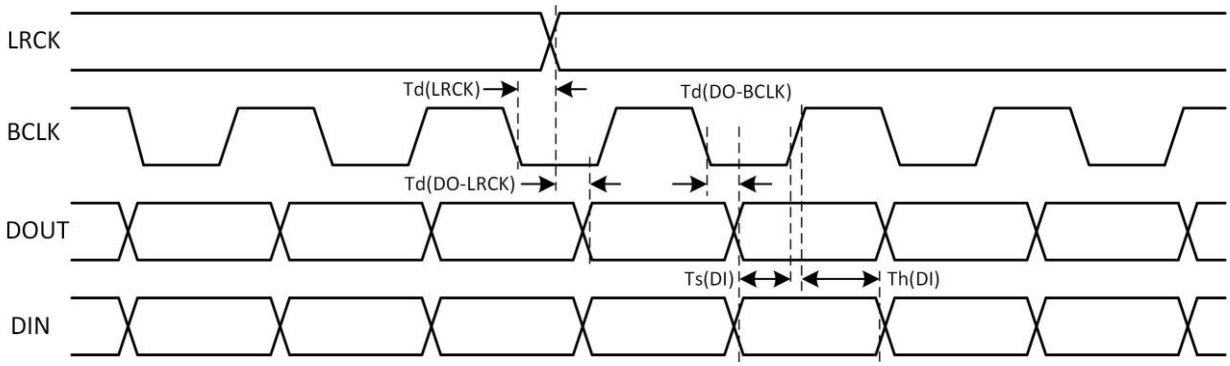


Figure2- 46. I2S/PCM in Master Mode Timing

Table2- 46. I2S/PCM in Master Mode Timing Constants

Parameter	Symbol	Min	Typ	Max	Unit
LRCK Delay	T_d(LRCK)	-	-	10	ns
LRCK to DOUT Delay(For LJF)	T_d(DO-LRCK)	-	-	10	ns
BCLK to DOUT Delay	T_d(DO-BCLK)	-	-	10	ns
DIN Setup	T_s(DI)	4	-	-	ns
DIN Hold	T_h(DI)	4	-	-	ns
BCLK Rise Time	T_r	-	-	8	ns
BCLK Fall Time	T_f	-	-	8	ns

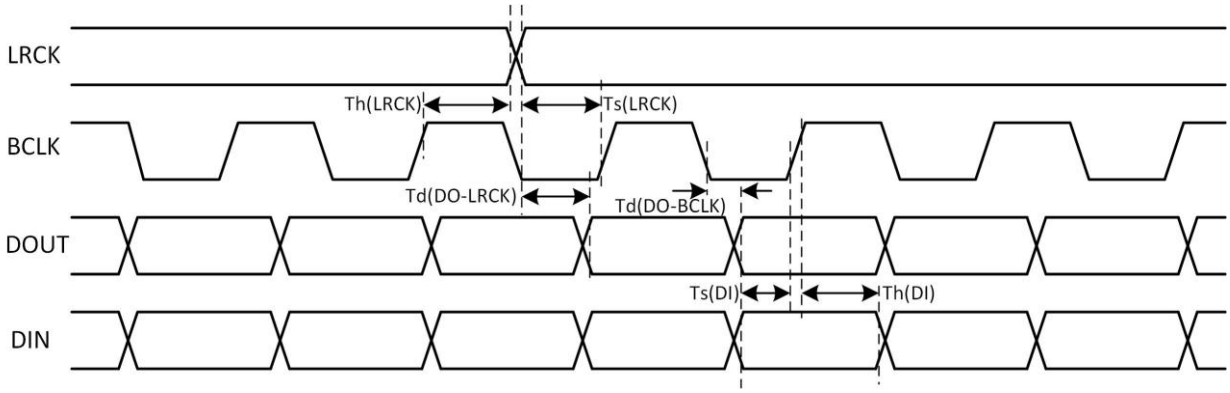


Figure2- 47. I2S/PCM in Slave Mode Timing

Table2- 47. I2S/PCM in Slave Mode Timing Constants

Parameter	Symbol	Min	Typ	Max	Unit
LRCK Setup	T_s(LRCK)	4	-	-	ns
LRCK Hold	T_h(LRCK)	4	-	-	ns
LRCK to DOUT Delay(For LJF)	T_d(DO-LRCK)	-	-	10	ns
BCLK to DOUT Delay	T_d(DO-BCLK)	-	-	10	ns
DIN Setup	T_s(DI)	4	-	-	ns
DIN Hold	T_h(DI)	4	-	-	ns

BCLK Rise Time	T_r	-	-	4	ns
BCLK Fall Time	T_f	-	-	4	ns

2.5.7 DMIC Interface Timing

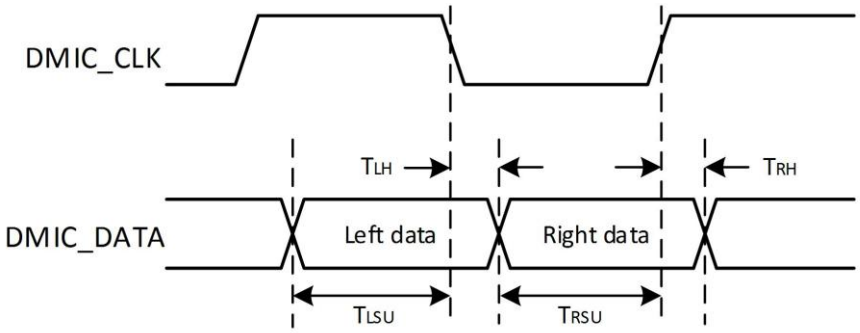


Figure2- 48. DMIC Interface Timing

Table2- 48. DMIC Interface Timing Constants

Parameter	Symbol	Min	Typ	Max	Unit
DMIC_DATA(Left) setup time to falling DMIC_CLK edge	TLSU	15	-	-	ns
DMIC_DATA(Left) hold time from falling DMIC_CLK edge	TLH	0	-	-	ns
DMIC_DATA(Right) setup time to rising DMIC_CLK edge	TRSU	15	-	-	ns
DMIC_DATA(Right) hold time from rising DMIC_CLK edge	TRH	0	-	-	ns

2.5.8 SPI Interface Timing

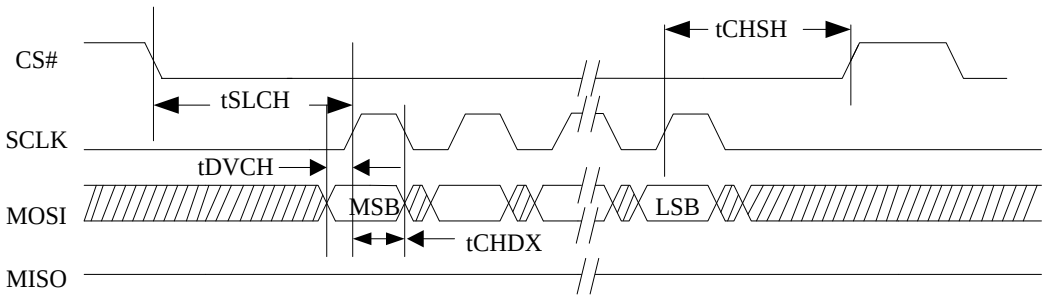


Figure2- 49. SPI MOSI Timing

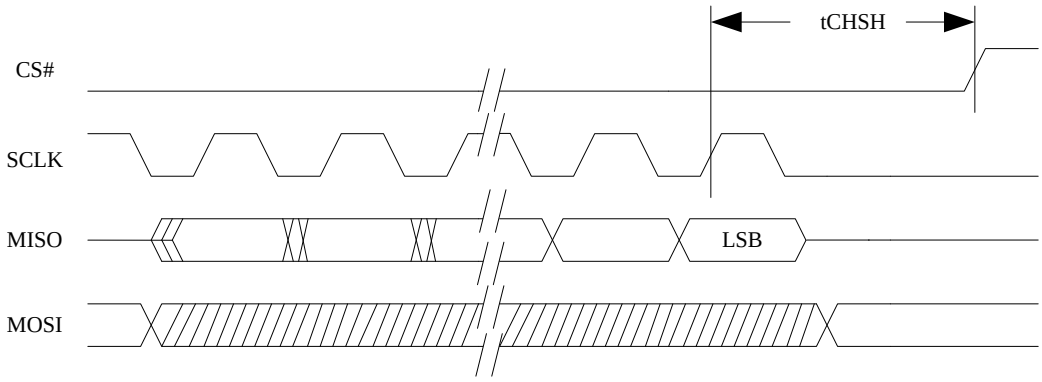


Figure2- 50. SPI MISO Timing

Table2- 49. SPI Timing Constants

Parameter	Symbol	Min	Typ	Max	Unit
CS# active setup time	tSLCH	-	2T	-	ns
CS# active hold time	tCHSH	-	2T	-	ns
Data in setup time	tDVCH	-	T/2-3	-	ns
Data in hold time	tCHDX	-	T/2-3	-	ns

NOTE

T is the cycle of clock.

2.5.9 UART Interface Timing

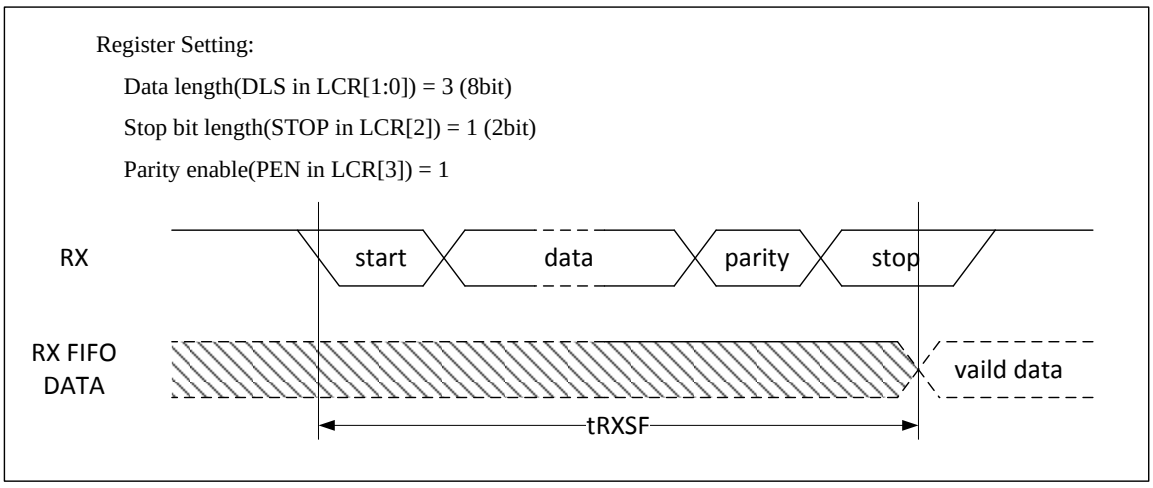


Figure2- 51. UART RX Timing

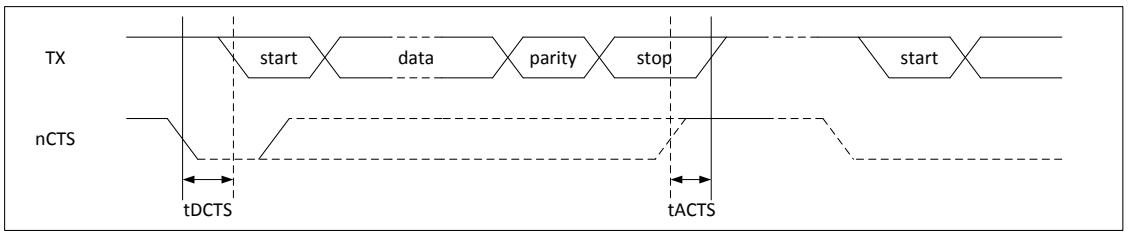


Figure2- 52. UART nCTS Timing

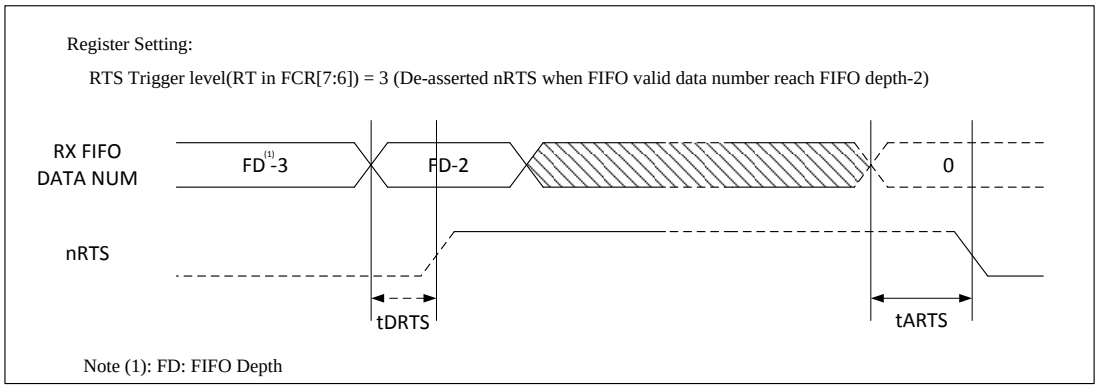


Figure2- 53. UART nRTS Timing

Table2- 50. UART Timing Constants

Parameter	Symbol	Min	Typ	Max	Unit
RX start to RX FIFO	tRXSF	10.5× BRP(1)	-	11× BRP	ns
Delay time of de-asserted nCTS to TX start	tDCTS	-	-	BRP	ns
Step time of asserted nCTS to stop next transmission	tACTS	BRP(1)/4	-	-	ns
Delay time of de-asserted nRTS	tDRTS	-	-	BRP	ns
Delay time of asserted nRTS	tARTS	-	-	BRP	ns

NOTE

BRP:Baud-Rate Period.

2.5.10 TWI Interface Timing

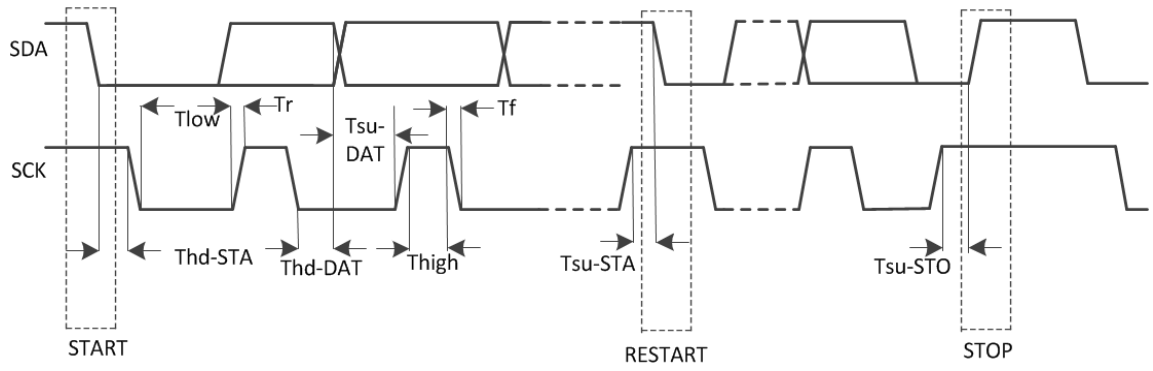


Figure2- 54. TWI Timing

Table2- 51. TWI Timing Constants

Parameter	Symbol	Standard mode		Fast mode		Unit
		Min	Max	Min	Max	
SCK clock frequency	Fsck	0	100	0	400	kHz
Setup time in start	Tsu-STA	4.7	-	0.6	-	us
Hold time in start	Thd-STA	4.0	-	0.6	-	us
Setup time in data	Tsu-DAT	250	-	100	-	ns
Hold time in data	Thd-DAT	5.0	-	-	-	ns
Setup time in stop	Tsu-STO	4.0	-	6.0	-	us
SCK low level time	Tlow	4.7	-	1.3	-	us
SCK high level time	Thigh	4.0	-	0.6	-	ns
SCK/SDA falling time	Tf	-	300	20	300	ns
SCK/SDA rising time	Tr	-	1000	20	300	ns

Figures

Tables

3 System

3.1 Memory Mapping

Module	Address(It is for Cluster CPU)	Size(Bytes)
N-BROM	0x0000 0000—0x0000 BFFF	48K
S-BROM	0x0000 0000—0x0000 FFFF	64K
SRAM A1	0x0002 0000---0x0003 7FFF	96K (Supports operation with Byte, the Clock Source is AHB1)
SRAM C	0x0003 8000---0x0005 8FFF	132K (20K for DE, 112K for VE. The Clock Source is AHB1)
	VE: 0x0003 8000---0x0005 3FFF DE: 0x0005 4000---0x0005 8FFF	
SRAM A2	0x0010 0000---0x0010 3FFFF	16K (Only for CPUS Vector Table)
	0x0010 4000---0x0011 BFFF	96K
Accelerator		
DE	0x0100 0000---0x013F FFFF	4M
G2D	0x0148 0000---0x014B FFFF	256K
GPU	0x0180 0000---0x0183 FFFF	256K
CE_NS	0x0190 4000---0x0190 47FF	2K
CE_S	0x0190 4800---0x0190 4FFF	2K
EMCE	0x0190 5000---0x0190 5FFF	4K
CE_KEY_SRAM	0x0190 8000---0x0190 8FFF	4K
VE SRAM	0x01A0 0000---0x01BF FFFF	2M
VE	0x01C0 E000---0x01C0 FFFF	4K
VIPP_SRAM	0x01D0 0000---0x020FFFFF	4M
VIPP_REG	0x0210 0000---0x0210 2FFF	12K
System Resources		
SYS_CFG	0x0300 0000---0x0300 0FFF	4K
CCMU	0x0300 1000---0x0300 1FFF	4K
DMA	0x0300 2000---0x0300 2FFF	4K
MSGBOX	0x0300 3000---0x0300 3FFF	4K
SPINLOCK	0x0300 4000---0x0300 4FFF	4K
HSTIMER	0x0300 5000---0x0300 5FFF	4K
SID	0x0300 6000---0x0300 6FFF	4K
TIMER	0x0300 9000---0x0300 93FF	1K
PWM	0x0300 A000---0x0300 A3FF	1K

GPIO	0x0300 B000---0x0300 B3FF	1K
PSI	0x0300 C000---0x0300 C3FF	1K
DCU	0x0301 0000---0x0301 FFFF	64K
GIC	0x0302 0000---0x0302 FFFF	64K
IOMMU	0x030F 0000---0x030F FFFF	64K
Memory		
NDFC	0x0401 1000---0x0401 1FFF	4K
SMHC0	0x0402 0000---0x0402 0FFF	4K
SMHC1	0x0402 1000---0x0402 1FFF	4K
SMHC2	0x0402 2000---0x0402 2FFF	4K
MSI_CTRL	0x047FA000---0x047FAFFF	4K
DRAM_CTRL	0x047FB000---0x047FFFFF	20K
PHY_CTRL	0X04800000---0X04FFFFFF	8M
Interfaces		
UART0	0x0500 0000---0x0500 03FF	1K
UART1	0x0500 0400---0x0500 07FF	1K
UART2	0x0500 0800---0x0500 0BFF	1K
UART3	0x0500 0C00---0x0500 0FFF	1K
UART4	0x0500 1000---0x0500 13FF	1K
TWI0	0x0500 2000---0x0500 23FF	1K
TWI1	0x0500 2400---0x0500 27FF	1K
TWI2	0x0500 2800---0x0500 2BFF	1K
SPI0	0x0501 0000---0x0501 0FFF	4K
SPI1	0x0501 1000---0x0501 1FFF	4K
GPADC	0x0507 0000---0x0507 03FF	1K
THS	0x0507 0400---0x0507 07FF	1K
LRADC	0x0507 0800---0x0507 0BFF	1K
I2S/PCM0	0x0509 0000---0x0509 0FFF	4K
I2S/PCM1	0x0509 1000---0x0509 1FFF	4K
DMIC	0x0509 5000---0x0509 53FF	1K
Audio Codec	0x0509 6000---0x0509 67FF	2K
USB2.0_OTG	0x0510 0000---0x051F FFFF	1M
USB2.0_HOST	0x0520 0000---0x052F FFFF	1M
Display		
MIPI_DSI0	0x0650 4000---0x0650 5FFF	8K
DISP_IF_TOP	0x0651 0000---0x0651 0FFF	4K
TCON_LCD0	0x0651 1000---0x0651 1FFF	4K
CSIC	0x0660 0000---0x0661 FFFF	128K
CSI_SRAM	0x0662 0000---0x0669 FFFF	512K
CPUS		
RTC	0x0700 0000---0x0700 03FF	1K
R_CPUS_CFG	0x0700 0400---0x0700 0BFF	2K
R_PRCM	0x0701 0000---0x0701 03FF	1K
R_TIMER	0x0702 0000---0x0702 03FF	1K

R_WDOG	0x0702 0400---0x0702 07FF	1K
R_TWDOG	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
R_UART	0x0708 0000---0x0708 03FF	1K
R_TWI	0x0708 1400---0x0708 17FF	1K
R_RSB	0x0708 3000---0x0708 33FF	1K
CPU Related		
CPU_SYS_CFG	0x0810 0000---0x0810 03FF	1K
TimeStamp_STA	0x0811 0000---0x0811 0FFF	4K
TimeStamp_CTRL	0x0812 0000---0x0812 0FFF	4K
C0_CPUX_CFG	0x0901 0000---0x0901 03FF	1K
C0_CPUX_MBIST	0x0902 0000---0x0902 0FFF	4K
DRAM		
DRAM SPACE	0x4000 0000---0xFFFF FFFF	3G

3.2 CPUX Configuration

3.2.1 Overview

CPUX Configuration(CPUX_CFG) module is used to configure CLUSTER0 control, including power on,reset, cache, debug, and check the status of CPU. It will be used when you want to disable/enable the CPU, cluster switch, CPU status check, and debug, etc.

The CPUX_CFG module includes C0_CPUX_CFG and CPU_SUBSYS_CTRL.

The C0_CPUX_CFG module is used for configuring CLUSTER0, such as reset, control, cache, debug, CPU status.

The CPU_SUBSYS_CTRL module is used for the system resource control of CPU sub-system, such as GIC-400,JTAG.

Features:

- CPU reset system:CORE reset,debug circuit reset and other reset function
- CPU related control: interface control, CP15 control, power on and power down control
- CPU status check: idle status, SMP status, interrupt status and so on
- CPU debug related register for control and status

3.2.2 Block Diagram

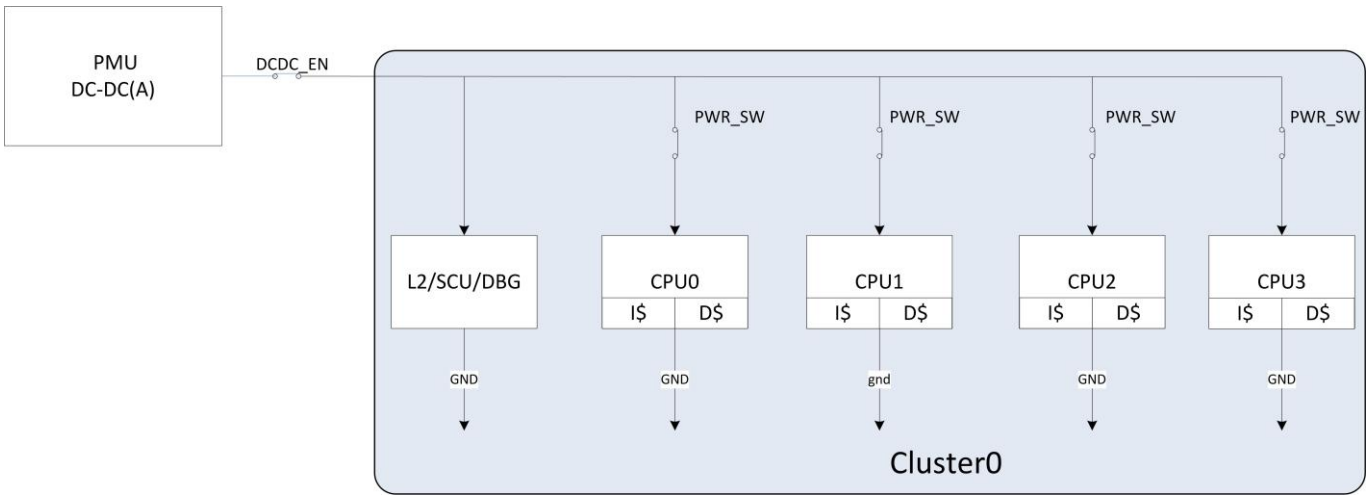


Figure3- 1. CPUX_CFG Block Diagram

The figure above lists the power domain of CLUSTER in default. All power switch of CPU core are default to power on.All CPU pwron_rst is de-asserted, core reset of CPU0 is de-asserted,core reset of CPU [3:1] is asserted.

Since each CPU core and its appended circuits have the same power domain, the processor and related L1 cache, neon and vfp should be taken as a whole core.

3.2.3 Operations and Functional Descriptions

3.2.3.1 Signal Description

For the detail of CPUX signal, please refer to ***ARM Cortex-A7 TRM***.

3.2.3.2 L2 Idle Mode

When the L2 of Cluster needs to enter WFI mode, firstly make sure the CPU0/1/2/3 of Cluster enter WFI mode, which can be checked through the bit[19:16] of **Cluster CPU Status Register**, and then pull high the **ACINACTM** of Cluster by writing 1to the bit0 of **Cluster Control Register1** , and then check whether L2 enters idle status by checking whether the **STANDBYWFIL2** is high. Remember to set the **ACINACTM** to low when exiting the L2 idle mode.

3.2.3.3 CPUX Reset System

The CPUX reset includes **core reset**, **power-on reset** and **H_Reset**. And their scopes rank: **core reset** < **power-on Reset** < **H_Reset**. The description of all reset signal in CPUX Reset System is as follows.

Table3- 1. Reset Signal Description

Reset signal	Description
CORE_RST	This is the primary reset signal which resets the corresponding core logic that includes NEON and VFP, Debug, ETM, breakpoint and watchpoint logic. This maps to a warm reset that covers 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. This maps to a cold reset that covers reset of the processor logic and the integrated debug functionality. This does not reset debug logic in the debug power domain. Including CORE_RST/ETM_RST/DBG_RST.
AXI2MBUS_RST	Reset 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	Reset ETM debug logic circuit.
DBG_RST	Reset only the debug, and breakpoint and watchpoint logic in the processor power domain. It also resets the debug logic for each processor in the debug power domain.
SOC_DBG_RST	Reset all the debug logic including DBG_RST.
MBIST_RST	Resets all resettable registers in the cluster, for entry into, and exit from, MBIST mode.
H_RST	Including PWRON_RST/L2_RST/MBIST_RST/SOC_DBG_RST/C0_CPUX_CFG.
CPU_SUBSYS_RST	Including C0_H_RST/GIC-400/CPU_SUBSYS_CTRL.

3.2.3.4 Operation Principle

The CPU-related operation needs proper configuration of CPUCFG related register, as well as related system control resource including BUS, clock ,reset and power control.

3.2.4 Cluster Configuration Register List

Module Name	Base Address
CPU_CFG	0x09010000

Register Name	Offset	Description
C_RST_CTRL	0x0000	Cluster Reset Control Register
C_CTRL_REG0	0x0010	Cluster Control Register0
C_CTRL_REG1	0x0014	Cluster Control Register1
C_CTRL_REG2	0x0018	Cluster Control Register2
CACHE_CFG_REG	0x0024	Cache Configuration Register
C_CPU_STATUS	0x0080	Cluster CPU Status Register
L2_STATUS_REG	0x0084	Cluster L2 Status Register
DBG_REG0	0x00C0	Cluster Debug Control Register0
DBG_REG1	0x00C4	Cluster Debug Control Register1

3.2.5 Cluster Configuration Register Description

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

Offset: 0x0000			Register Name: C_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
24	R/W	0x1	SOC_DBG_RST Cluster SOC Debug Reset 0: assert 1: de-assert
23:20	R/W	0xF	ETM_RST Cluster ETM Reset Assert 0: assert 1: de-assert
19:16	R/W	0xF	DBG_RST

			Cluster Debug Reset Assert 0: assert 1: de-assert
15:9	/	/	/
8	R/W	0x1	L2_RST Cluster L2 Cache Reset 0: assert 1: de-assert
7:4	/	/	/
3:0	R/W	0x1	CORE_RESET Cluster CPU[3:0] Reset Assert 0: assert 1: de-assert

3.2.5.2 Cluster 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 of barriers onto system bus 0: Barriers are broadcast onto system bus,this requires an AMBA4 interconnect. 1: Barriers are not broadcast onto the system bus.This is compatible with an AXI3 interconnect.
30	R/W	0x0	BROADCAST_INNER. Enable broadcasting of Inner Shareable transactions 0: Inner shareable transactions are not broadcasted externally. 1: Inner shareable transactions are broadcasted externally.
29	R/W	0x0	BROADCAST_OUTER. Enable broadcasting of 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 of 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.
27:12	/	/	/
11:8	R/W	0x0	CP15S_DISABLE. Disable write access to some secure CP15 register.
7:5	/	/	/
4	R/W	0x0	L2_RST_DISABLE.

			Disable automatic L2 cache invalidate at reset. 0: L2 cache is reset by hardware. 1: L2 cache is not reset by hardware.
3:0	R/W	0x0	L1_RST_DISABLE. Disable automatic Cluster CPU[3:0] L1 cache invalidate at reset. 0: L1 cache is reset by hardware. 1: L1 cache is not reset by hardware.

3.2.5.3 Cluster Control Register1(Default Value:0x0000_0000)

Offset: 0x0014			**Register Name: C_CTRL_REG1**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1	R/W	0x0	CRM auto select slow frequency en. 0 : disable auto select 1 : enable auto select
0	R/W	0x0	ACINACTM. Snoop interface is inactive and no longer accepting requests. 0: Snoop interface is active 1: Snoop interface is inactive

3.2.5.4 Cluster Control Register2(Default Value:0x0000_0010)

Offset: 0x0018			**Register Name: C_CTRL_REG2**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:25	/	/	/
24	R/W	0x0	EVENTI Event input for processor wake-up from WFE state. This bit must remain HIGH for at least one clock cycle to be visible by the cores.
23:20	R/W	0x0	EXM_CLR[3:0] Clear the status of interface.
19:0	/	/	/

3.2.5.5 Cache Configuration Register(Default Value: 0x001A_001A)

Offset: 0x24			**Register Name: CACHE_CFG_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:22	/	/	/
21:19	R/W	0x3	EMA_L2D L2 Cache SRAM EMA control port
18:17	R/W	0x1	EMAW_L2D L2 Cache SRAM EMAW control port

16	R/W	0x0	EMAS_L2D L2 Cache SRAM EMAS control port
15:6	/	/	/
5:3	R/W	0x3	EMA Cache SRAM EMA control port
2:1	R/W	0x1	EMAW Cache SRAM EMAW control port
0	R/W	0x0	EMAS Cache SRAM EMAS control port

3.2.5.6 Cluster CPU Status Register(Default Value: 0x000E_0000)

Offset: 0x0080			**Register Name: C_CPU_STATUS**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	/	/	/
27:24	R	0x0	SMP_AMP CPU[3:0] is in Symmetric Multiprocessing mode or Asymmetric Multiprocessing mode. 0: AMP mode 1: SMP mode
23:20	/	/	/
19:16	R	0xE	STANDBYWFI. Indicates if Cluster CPU[3:0] is in WFI standby mode. 0: Processor not in WFI standby mode. 1: Processor in WFI standby mode
15:12	/	/	/
11:8	R	0x0	STANDBYWFE. Indicates if Cluster CPU[3:0] is in the WFE standby mode. 0: Processor not in WFE standby mode 1: Processor in WFE standby mode
7:1	/	/	/
0	R	0x0	STANDBYWFIL2. Indicates if the Cluster L2 memory system is in WFI standby mode. 0: Cluster L2 not in WFI standby mode 1: Cluster L2 in WFI standby mode

3.2.5.7 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 Cluster 0 Debug Control Register0(Default Value:0x0000_000F)

Offset: 0x00C0			Register Name: DBG_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:12	/	/	/
11:8	R/W	0x0	DBGRESTART[3:0] External restart requests.
7:4	/	/	/
3:0	R/W	0xF	C_DBGPWRDUP. Cluster Powered-up 0: Core is powered down 1: Core is powered up

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

Offset: 0x00C4			Register Name: DBG_REG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:12	R/W	0x0	DBGRESTARTED[3:0] Handshake for DBGRESTART.
11:8	/	/	/
7:4	R	0x0	C_DBGNOPWRDWN. No power-down request. Debugger has requested that processor is not powered down. Debug no power down[3:0].
3:0	R	0x0	C_DBGPWRUPREQ. Power up request. Debug power up request[3:0] 0: Do not request that the core is powered up 1: Request that the core is powered up

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	GIC IRQ/FIQ Status Register
GENER_CTRL_REG2	0x0018	General Control Register2

3.2.7 CPU Subsystem Control Register Description

3.2.7.1 General Control Register0(Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: GENER_CTRL_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1	R/W	0x0	Reserved
0	R/W	0x0	GIC_CFGSDISABLE. Disables write access to some secure GIC registers.

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

Offset: 0x000C			Register Name: GIC_JTAG_RST_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:20	/	/	/
19:16	R/W	0x0	EXM_CLR[3:0] Clear the status of interface, for debug
15:12	/	/	/
11	R/W	0x1	CS_RST CoreSight Reset. 0: assert 1: de-assert.
10	/	/	/
9	R/W	0x1	PORTRST Jtag portrst. 0: assert 1: de-assert.
8	R/W	0x1	TRST. Jtag trst. 0: assert 1: de-assert.
7:3	/	/	/
2	R/W	0x1	cluster0_corepll_rst 0: assert 1: de-assert.
1	R/W	0x1	Reserved
0	R/W	0x1	GIC_RESET. Gic_reset_cpu_reg

			0: assert 1: de-assert.

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

Offset: 0x0010			Register Name: C0_INT_EN
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0xFFFF	C0_GIC_EN Interrupt enable control register. Mask irq_out/firq_out to system domain.

3.2.7.4 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	0x0	FIQ_OUT[15:0]
15:0	R/W	0x0	IRQ_OUT[15:0]

3.2.7.5 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
15:1	/	/	/
0	R/W	0x0	C0_TSCLKCHANGE

3.3 CCU

3.3.1 Overview

The CCMU(Clock Control Management Unit) provides the registers to program the PLLs and controls the PLLs configuration and most of the clock generation, division, distribution, synchronization and gating. CCMU input signals include the external clock for the reference frequency (24MHz). The outputs from CCMU are mostly clocks to other blocks in the system.

Features:

- 11 PLLs
- Bus Source and Divisions
- Clock Output Control
- PLL Bias Control
- PLL Tunning Control
- PLL Pattern Control
- Configuring Modules Clock
- Bus Clock Gating
- Bus Software Reset
- PLL Lock Control

3.3.2 Operations and Functional Descriptions

3.3.2.1 System Bus Tree

Figure 3-2 shows the block diagram of the system bus tree.

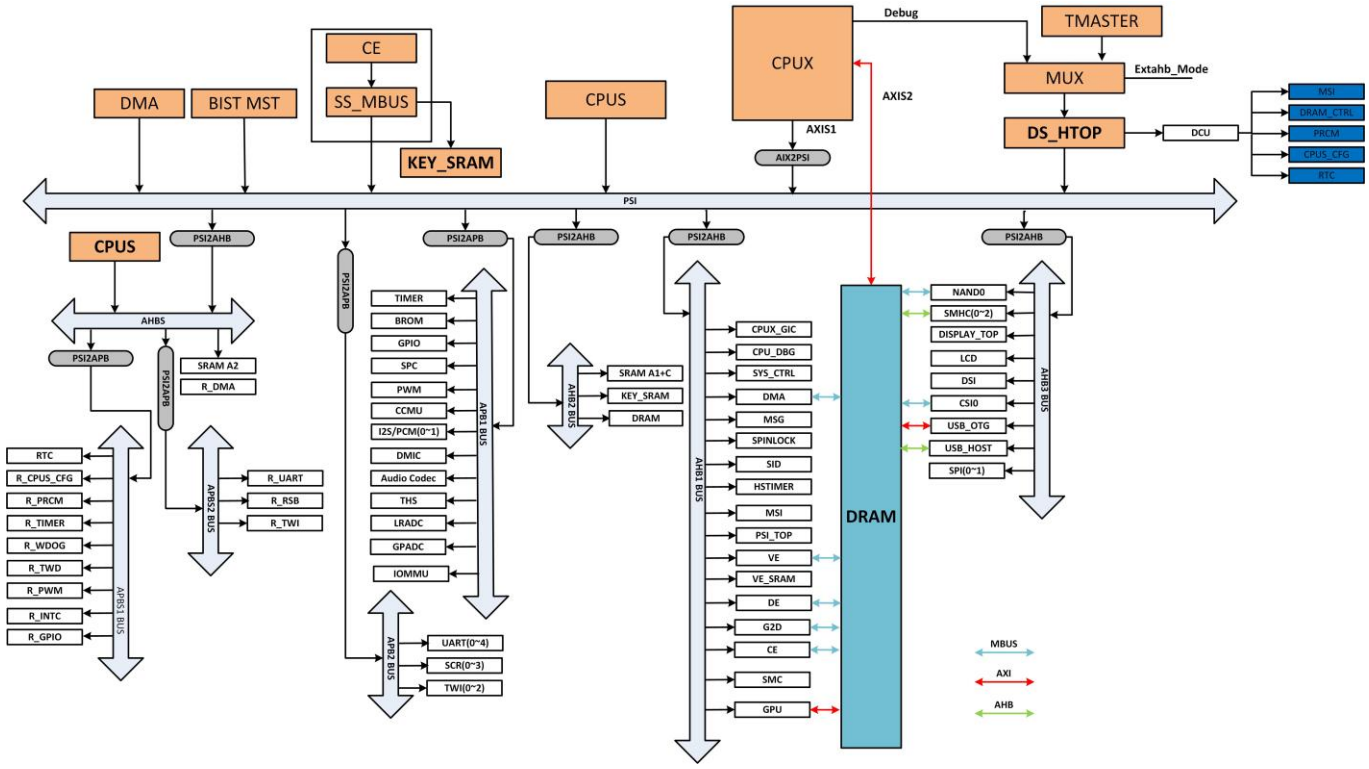


Figure3- 2. System Bus Tree

The above system bus tree is used to introduce bus interface of every module. These modules can divide into two types: bus master, slave. But a part of these modules can be not only used as bus master, but also as slave. DMA, BIST_MST(Built-in Self Test Master), CE, CPUS(M4), CPUX(A7*4) and DS(debug interface) are as bus master that can access corresponding register of every slave through bus. Every slave hangs in corresponding bus.

For example, CPU accesses to RTC module, the process is as follows: CPU instruction firstly passes AXI bus, then goes to PSI bus through AXI2PSI bridge, then goes to AHBS bus through PSI2AHB bridge, and goes to APBS2 through AHB2APB, finally RTC is operated based on relevant bus protocol . The access time from CPU to RTC , is relevant with the CPU clock , AXI bus clock, PSI bus clock, AHBS bus clock and APBS2 bus clock. Any lower bus clock will lead to access time very long.

DRAM supports 3 bus types, DRAM can interact directly with some modules through different bus without using CPU. Red arrow indicates that those modules (CPUX,GPU,USB_OTG) access DRAM through AXI bus. Green arrow indicates that those modules (USB_HOST,SMHC0/1/2) access DRAM through AHB bus. Blue arrow indicates that those modules(DE,CE,VE,G2D,DMA,NDFC,CSI) access DRAM through MBUS .

In above module, the clocks of these modules(such as TWI and UART) to be hung on APB2/APBS2 are from their respective bus clock , however the clocks of most other modules are from related CLK register , such as DE_CLK_REG. Each module clock requirement can refer to their module.

3.3.2.2 Bus Clock Tree

Figure 3-3 shows the block diagram of the bus clock tree.

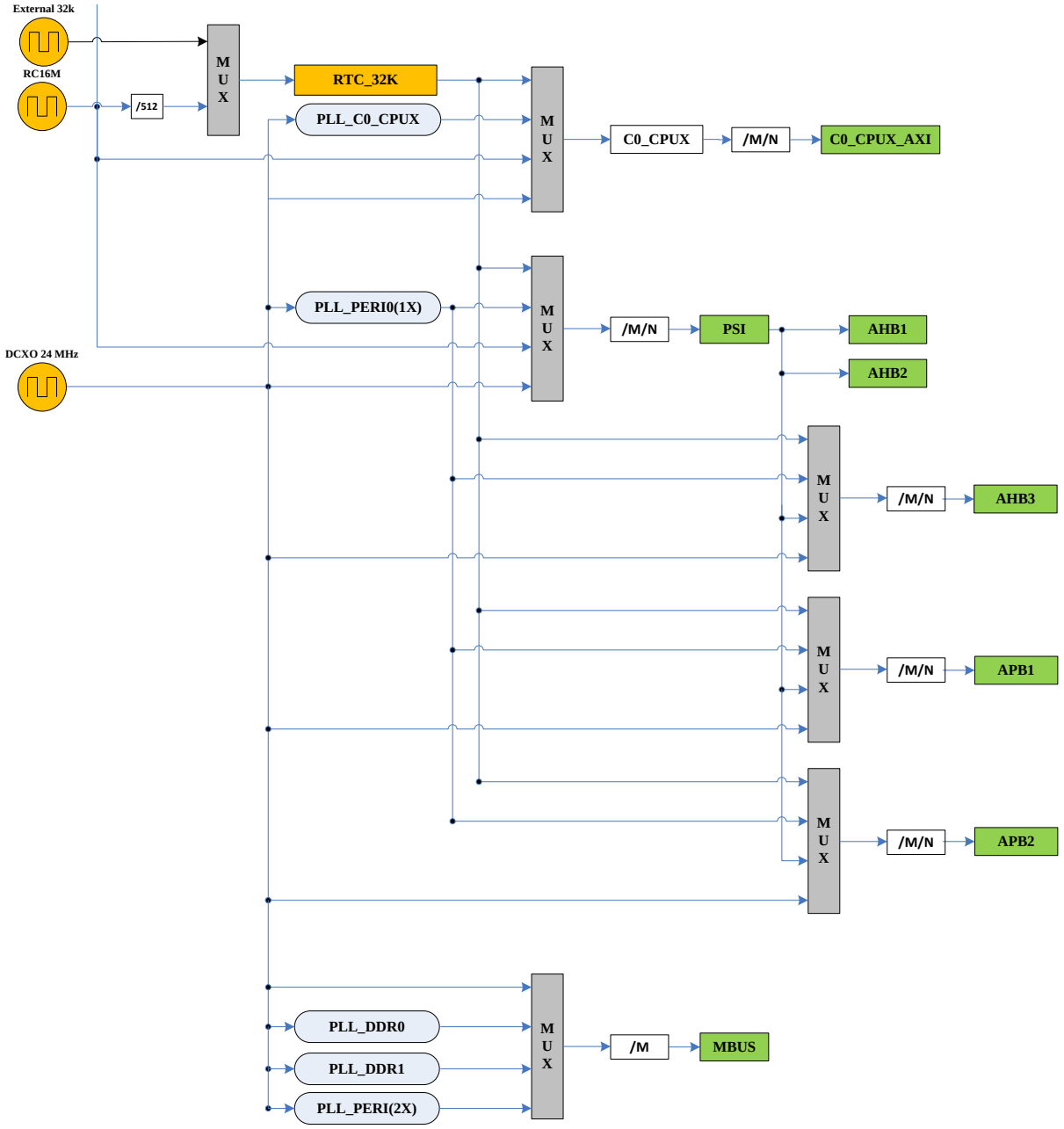


Figure3- 3. Bus Clock Tree

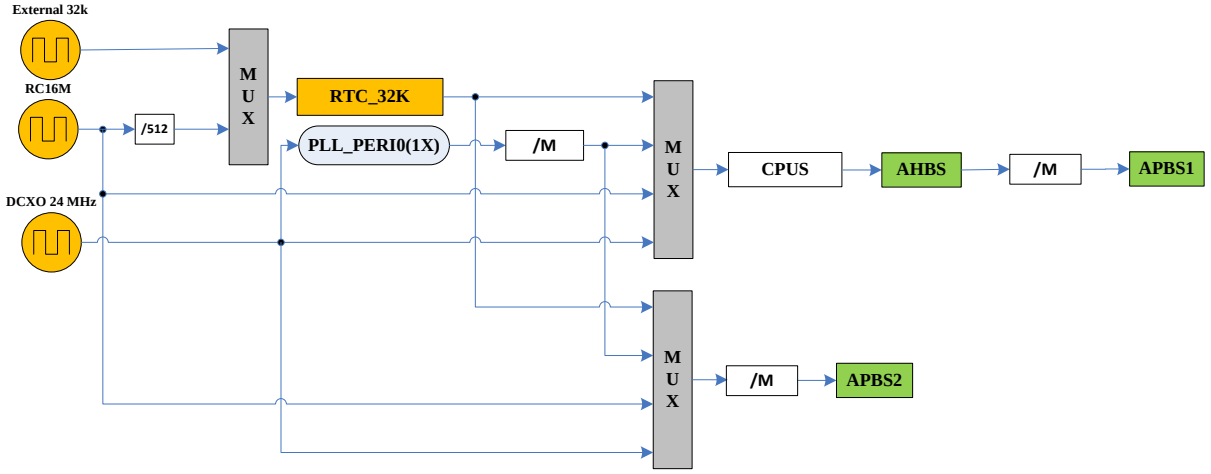


Figure3- 4. Bus Clock Tree

The bus clock tree diagram shows the various bus clock source, relevant divider factor, etc.

For example, the clock source of CPU is from PLL_CPU (48MHz ~ 2GHz), OSC_24M, RC_16M, RTC_32K, etc. The internal AXI bus clock source is from the clock source of CPU , but on the base of CPU clock, two configurable divider factors(M,N) are added to adjust the AXI bus clock.

The clock source of RTC_32K is from external 32 kHz, or internal RC_16M by 512 fractional frequency.

The bus clocks commonly are used as follows.

AXI: 1/3 of CPU frequency.

PSI、AHB1、AHB2、AHB3: 200MHz.

PSI、AHB1、AHB2 is controlled by PSI_AHB1_AHB2_CFG_REG. AHB3 is controlled by AHB3_CFG_REG.

APB1、APB2: maximum of 100MHz, separately controlled by APB1_CFG_REG、APB2_CFG_REG .

AHBS: 200MHz.

APBS1、APBS2: maximum of 100MHz.

3.3.2.3 Module Clock Tree

Figure 3-5 and Figure 3-6 shows the block diagram of the module clock tree.

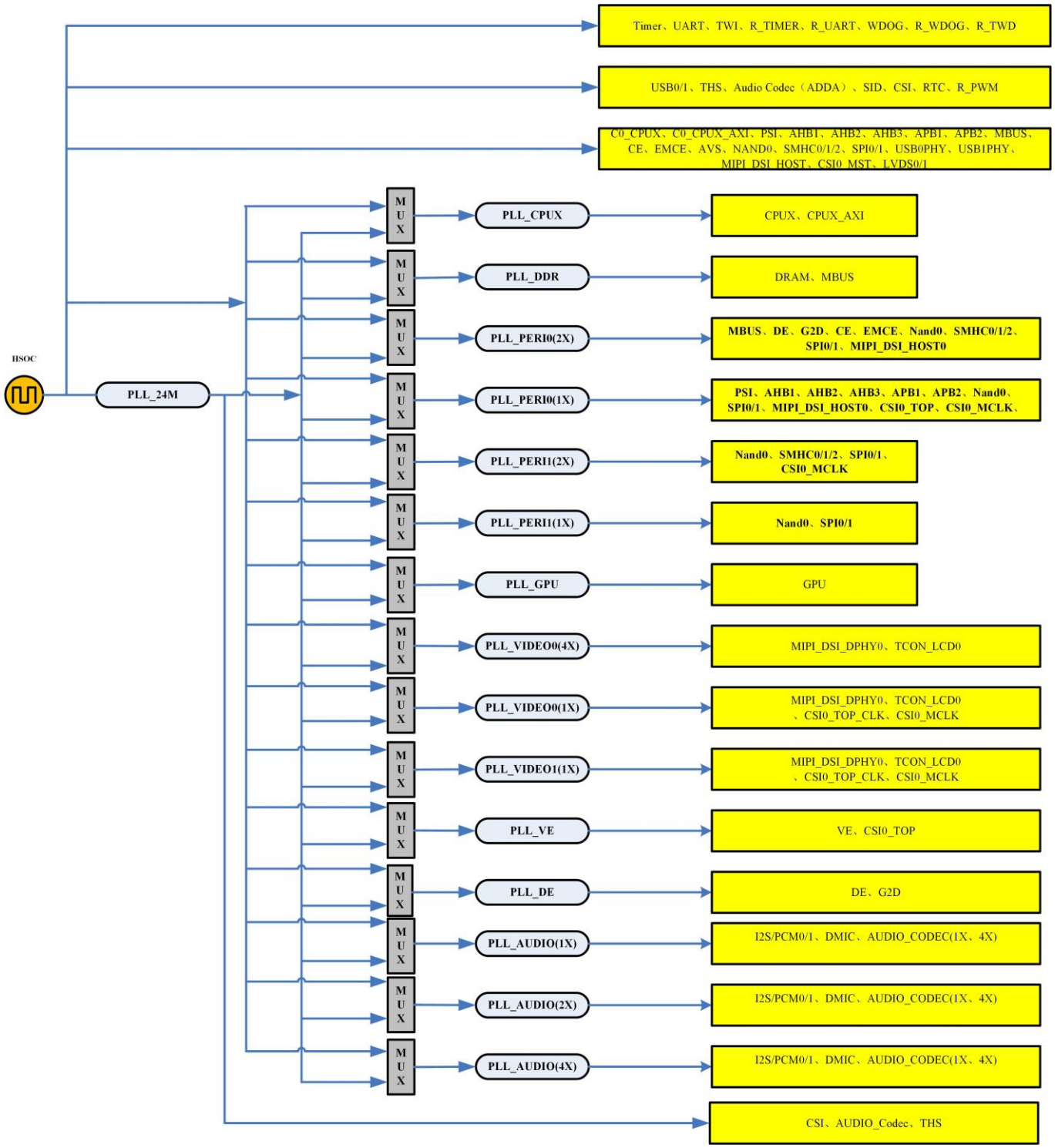


Figure3- 5. Module Clock Tree(high frequence source)

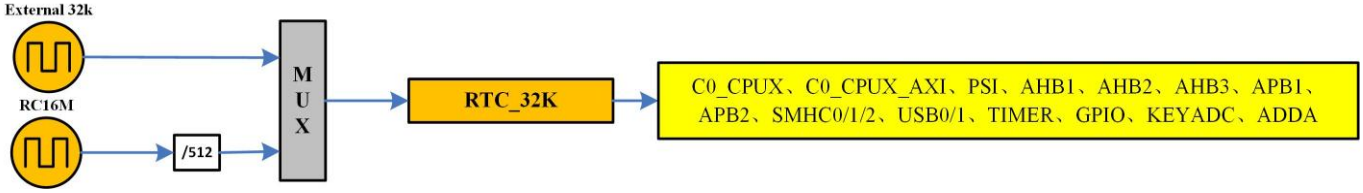


Figure3- 6. Module Clock Tree(low frequence source)

3.3.2.4 Typical Applications

PLL Applications: Use the available clock sources to generate clock roots to the various parts of the chip.

Table3- 2. PLL Typical Applications

PLLs	Typical Application	Description
PLL_CPUX	CPUX, CPUX_AXI	Support DVFS
PLL_DDR	MBUS,DRAM	Support Spread Spectrum Not support linear frequency scaling
PLL_PERI0(2X)	MBUS, DE,G2D,CE,EMCE, NDFC, SMHC0/1/2, SPI0/1, MIPI_DSI_HOST0	Support Spread Spectrum Not support DVFS
PLL_PERI0(1X)	PSI,AHB1,AHB2,AHB3,APB1,APB2, DI,NDFC, SPI0/1 , MIPI_DSI_HOST0, CSI0_MCLK, CSI0_TOP	Not support DVFS
PLL_PERI1(2X)	NDFC, SMHC0/1/2,SPI0/1, CSI0_MCLK	Not support DVFS
PLL_PERI1(1X)	NDFC, SPI0/1	Not support DVFS
PLL_VIDEO0(4X)	TCON_LCD0, MIPI_DSI_DPHY0	Not support DVFS
PLL_VIDEO0(1X)	TCON_LCD0, MIPI_DSI_DPHY0, CSI0_TOP_CLK, CSI0_MCLK	Not support DVFS
PLL_VE	VE, CSI0_TOP	Not support DVFS
PLL_DE	DE, G2D	Not support DVFS
PLL_AUDIO(1X)	I2S/PCM0,I2S/PCM1, DMIC, AUDIO_CODEC(1X, 4X)	Not support DVFS
PLL_AUDIO(2X)	I2S/PCM0,I2S/PCM1, DMIC, AUDIO_CODEC(1X, 4X)	Not support DVFS
PLL_AUDIO(4X)	I2S/PCM0,I2S/PCM1, DMIC, AUDIO_CODEC(1X, 4X)	Not support DVFS

DVFS: Dynamic Voltage and Frequency Scaling.

3.3.2.5 PLL Features

Table3- 3. PLL Features

PLLs	Stable Operating Frequency	Actual Operating Frequency	Spread Spectrum	Linear FM	Rate Control	Pk-Pk	duty	Lock Time
PLL_CPUX	288M~5.0G (24*N)	288M~1800M	No	No	No	< 200ps	Yes	1.5ms
PLL_AUDIO	24.576M、22.5792M (24*N/div1/div2)	24.576M 22.5792M (24.576*4M (22.5792*4M	Yes(Fractional-N)	No	No	< 200ps	Yes(M0 = 2)	500us
PLL_PERI0(2X)	180M~3.0G (24*N/div1/div2)	1.2G	Yes	No	No	< 200ps	Yes	500us
PLL_PERI1(2X)	180M~3.0G (24*N/div1/div2)	1.2G	Yes	No	No	< 200ps	Yes	500us
PLL_Video0(4X)	252M~3G (24*N/Div)	192M~1200M	Yes	No	No	< 200ps	Yes	500us
PLL_Video1(4X)	252M~3G (24*N/Div)	192M~1200M	Yes	No	No	< 200ps	Yes	500us
PLL_VE	180M~3G (24*N/div1/div2)	192M~800M	Yes	No	No	< 200ps	Yes	500us
PLL_DDR	180M~3G (24*N/div1/div2)	192M~2000M	Yes	No	No	200Mhz~800Mhz(< 200ps) 800Mhz~1.3Ghz(< 140ps) 1.3Ghz~2.0Ghz(< 100ps)	Yes	500us
PLL_GPU	180M~3G (24*N/div1/div2)	180M~800M	Yes	No	No	< 200ps	Yes	500us
PLL_DE	180M~3G (24*N/div1/div2)	192M~600M	Yes	No	No	< 200ps	Yes	500us
PLL_24M	180M~3G	24M	No	No	No	< 200ps	Yes	500us

3.3.3 Programming Guidelines

3.3.3.1 PLL

In practical application, other PLLs do not support dynamic frequency scaling except for PLL_CPUX and PLL_GPU.

The user guide of PLL Lock(using PLL_CPUX as an example)

(a).PLL_CPUX from close to open:

- Write 0 to the bit29 of **PLL_CPUX_CTRL_REG**.
- Configure the parameters **(N,P)** of **PLL_CPUX_CTRL_REG**.
- Write 1 to the **Enable** bit of CPU PLL**PLL_CPUX_CTRL_REG**.
- Write 1 to the bit29 of **PLL_CPUX_CTRL_REG**.
- Read the bit28 of **PLL_CPUX_CTRL_REG**, when it is 1, then CPUX PLL is locked.
- Delay 20us.

(b).PLL_CPUX frequency conversion:

- Write 0 to the bit29 of **PLL_CPUX_CTRL_REG**.
- Configure the parameters **(N,P)** of **PLL_CPUX_CTRL_REG**.
- Write 1 to the bit29 of **PLL_CPUX_CTRL_REG**.
- Read the bit28 of **PLL_CPUX_CTRL_REG**, when it is 1, then CPUX PLL is locked.
- Delay 20us.

(c).PLL_CPU from open to close:

- Write 0 to the **Enable** bit of **PLL_CPUX_CTRL_REG**.
- Write 0 to the bit29 of **PLL_CPUX_CTRL_REG**.

NOTE

Parameter definition of PLL formula: P is divider factor, N is multiplier factor. P changes exponentially, other factors change by adding 1.

3.3.3.2 BUS

(1) When setting the BUS clock , you should set the division factor firstly, and after the division factor becomes valid, switch the clock source. The clock source will be switched after at least three clock cycles.

(2) The BUS clock should not be dynamically changed in most applications.

3.3.3.3 Clock Switch

Make sure that the clock source output is valid before the clock source switch, and then set a proper divide ratio; after the division factor becomes valid, switch the clock source.

3.3.3.4 Gating and Reset

Make sure that the reset signal has been released before the release of module clock gating.

3.3.3.5 Spread Spectrum Function

The configuration of spread spectrum follows the following steps.

Step1: Configure PLL_CTRL Register

According to PLL frequency and PLL frequency formula f = [(N + 1)/ (M0 + 1) + X] * 24MHz, compute the value of divisor M0 and divisor M1, calculate factor N and decimal value X, and write M0、M1、N and PLL frequency to the PLL_CTRL register.

Configure the SDM_Enable bit of the PLL_CTRL register to 1 to enable spread spectrum function.

NOTE

Having different PLL calculate formula for different PLL, please refer to each PLL_CTRL register.

Step2: Configure PLL_PAT Register

According to decimal value X and spread spectrum frequency(the bit[18:17] of the PLL_PAT register), calculate WAVE_BOT (= 2^{17} * X1) and WAVE_STEP (= 2^{17} * (X2 –X1) / (24MHz/PREQ)*2).

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

Configure the spread spectrum clock source select bit(SDM_CLK_SEL) to 0 by default. But if the PLL_INPUT_DIV_M1 bit of the PLL_CTRL register is 1, the bit should set to 1.

Write WAVE_BOT、WAVE_STEP、PREQ、SPR_FREQ_MODE and SDM_CLK_SEL to the PLL_PAT register, and configure SIG_DELT_PAT_EN to 1.

Step3: Delay 20us

3.3.4 Register List

Module Name	Base Address
CCMU	0x0300 1000

Register Name	Offset	Description
CCMU Register List		
PLL_CPUX_CTRL_REG	0x0000	PLL_CPUX Control Register
PLL_DDR_CTRL_REG	0x0010	PLL_DDR Control Register
PLL_PERI0_CTRL_REG	0x0020	PLL_PERI0 Control Register
PLL_PERI1_CTRL_REG	0x0028	PLL_PERI1 Control Register
PLL_GPU_CTRL_REG	0x0030	PLL_GPU Control Register
PLL_VIDEO0_CTRL_REG	0x0040	PLL_VIDEO0 Control Register
PLL_VIDEO1_CTRL_REG	0x0048	PLL_VIDEO1 Control Register
PLL_VE_CTRL_REG	0x0058	PLL_VE Control Register
PLL_DE_CTRL_REG	0x0060	PLL_DE Control Register
PLL_AUDIO_CTRL_REG	0x0078	PLL_AUDIO Control Register
PLL_24M_CTRL_REG	0x00B8	PLL_24M Control Register
PLL_DDR_PAT_CTRL_REG	0x0110	PLL_DDR Pattern Control Register
PLL_PERI0_PAT0_CTRL_REG	0x0120	PLL_PERI0 Pattern0 Control Register

PLL_PERI0_PAT1_CTRL_REG	0x0124	PLL_PERI0 Pattern1 Control Register
PLL_PERI1_PAT0_CTRL_REG	0x0128	PLL_PERI1 Pattern0 Control Register
PLL_PERI1_PAT1_CTRL_REG	0x012C	PLL_PERI1 Pattern1 Control Register
PLL_GPU0_PAT0_CTRL_REG	0x0130	PLL_GPU0 Pattern0 Control Register
PLL_GPU0_PAT1_CTRL_REG	0x0134	PLL_GPU0 Pattern1 Control Register
PLL_VIDEO0_PAT0_CTRL_REG	0x0140	PLL_VIDEO0 Pattern0 Control Register
PLL_VIDEO0_PAT1_CTRL_REG	0x0144	PLL_VIDEO0 Pattern1 Control Register
PLL_VIDEO1_PAT0_CTRL_REG	0x0148	PLL_VIDEO1 Pattern0 Control Register
PLL_VIDEO1_PAT1_CTRL_REG	0x014C	PLL_VIDEO1 Pattern1 Control Register
PLL_VE_PAT0_CTRL_REG	0x0158	PLL_VE Pattern0 Control Register
PLL_VE_PAT1_CTRL_REG	0x015C	PLL_VE Pattern1 Control Register
PLL_DE_PAT0_CTRL_REG	0x0160	PLL_DE Pattern0 Control Register
PLL_DE_PAT1_CTRL_REG	0x0164	PLL_DE Pattern1 Control Register
PLL_AUDIO_PAT0_CTRL_REG	0x0178	PLL_AUDIO Pattern0 Control Register
PLL_AUDIO_PAT1_CTRL_REG	0x017C	PLL_AUDIO Pattern1 Control Register
PLL_CPUX_BIAS_REG	0x0300	PLL_CPUX Bias Register
PLL_DDR_BIAS_REG	0x0310	PLL_DDR Bias Register
PLL_PERI0_BIAS_REG	0x0320	PLL_PERI0 Bias Register
PLL_PERI1_BIAS_REG	0x0328	PLL_PERI1 Bias Register
PLL_GPU0_BIAS_REG	0x0330	PLL_GPU0 Bias Register
PLL_VIDEO0_BIAS_REG	0x0340	PLL_VIDEO0 Bias Register
PLL_VIDEO1_BIAS_REG	0x0348	PLL_VIDEO1 Bias Register
PLL_VE_BIAS_REG	0x0358	PLL_VE Bias Register
PLL_DE_BIAS_REG	0x0360	PLL_DE Bias Register
PLL_AUDIO_BIAS_REG	0x0378	PLL_AUDIO Bias Register
PLL_24M_BIAS_REG	0x03B8	PLL_24M Bias Register
PLL_CPUX_TUN_REG	0x0400	PLL_ CPUX Tuning Register
C0_CPUX_AXI_CFG_REG	0x0500	CPUX_AXI Configuration Register
PSI_AHB1_AHB2_CFG_REG	0x0510	PSI_AHB1_AHB2 Configuration Register
AHB3_CFG_REG	0x051C	AHB3 Configuration Register
APB1_CFG_REG	0x0520	APB1 Configuration Register
APB2_CFG_REG	0x0524	APB2 Configuration Register
MBUS_CFG_REG	0x0540	MBUS Configuration Register
DE_CLK_REG	0x0600	DE Clock Register
DE_BGR_REG	0x060C	DE Bus Gating Reset Register
G2D_CLK_REG	0x0630	G2D Clock Register
G2D_BGR_REG	0x063C	G2D Bus Gating Reset Register
GPU_CORE_CLK_REG	0x0670	GPU Clock Register
GPU_BGR_REG	0x067C	GPU Bus Gating Reset Register
CE_CLK_REG	0x0680	CE Clock Register
CE_BGR_REG	0x068C	CE Bus Gating Reset Register
VE_CLK_REG	0x0690	VE Clock Register
VE_BGR_REG	0x069C	VE Bus Gating Reset Register
EMCE_CLK_REG	0x06B0	EMCE Clock Register
EMCE_BGR_REG	0x06BC	EMCE Bus Gating Reset Register

DMA_BGR_REG	0x070C	DMA 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
PSI_BGR_REG	0x079C	PSI Bus Gating Reset Register
PWM_BGR_REG	0x07AC	PWM Bus Gating Reset Register
IOMMU_BGR_REG	0x07BC	IOMMU 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
NDFC_CLK0_REG	0x0810	NDFC_Clock0 Register
NDFC_CLK1_REG	0x0814	NDFC_Clock1 Register
NDFC_BGR_REG	0x082C	NDFC Bus Gating Reset Register
SMHC0_CLK_REG	0x0830	SMHC0 Clock Register
SMHC1_CLK_REG	0x0834	SMHC1 Clock Register
SMHC2_CLK_REG	0x0838	SMHC2 Clock Register
SMHC_BGR_REG	0x084C	SMHC Bus Gating Reset Register
UART_BGR_REG	0x090C	UART Bus Gating Reset Register
TWI_BGR_REG	0x091C	TWI Bus Gating Reset Register
SPI0_CLK_REG	0x0940	SPI0 Clock Register
SPI1_CLK_REG	0x0944	SPI1 Clock Register
SPI_BGR_REG	0x096C	SPI Bus Gating Reset Register
GPADC_BGR_REG	0x09EC	GPADC Bus Gating Reset Register
THS_BGR_REG	0x09FC	THS 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
DMIC_CLK_REG	0x0A40	DMIC Clock Register
DMIC_BGR_REG	0x0A4C	DMIC Bus Gating Reset Register
AUDIO_CODEC_1X_CLK_REG	0x0A50	AUDIO CODEC 1X Clock Register
AUDIO_CODEC_4X_CLK_REG	0x0A54	AUDIO CODEC 4X Clock Register
AUDIO_CODEC_BGR_REG	0x0A5C	AUDIO CODEC 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
MIPI_DSI_DPHY0_HS_CLK_REG	0x0B20	MIPI DSI DPHY0 High Speed Clock Register
MIPI_DSI_HOST0_CLK_REG	0x0B24	MIPI DSI Host0 Clock Register
MIPI_BGR_REG	0x0B4C	MIPI DSI BUS GATING RESET Register
DISPLAY_IF_TOP_BGR_REG	0x0B5C	DISPLAY_IF_TOP BUS GATING RESET Register
TCON_LCD_CLK_REG	0x0B60	TCON LCD Clock Register
TCON_LCD_BGR_REG	0x0B7C	TCON LCD BUS GATING RESET Register

LVDS_BGR_REG	0x0BAC	LVDS BUS GATING RESET Register
CSI_MISC_CLK_REG	0x0C00	CSI MISC Clock Register
CSI_TOP_CLK_REG	0x0C04	CSI TOP Clock Register
CSI_MST_CLK_REG	0x0C08	CSI Master Clock Register
CSI_BGR_REG	0x0C2C	CSI Bus Gating Reset Register
CCMU_SEC_SWITCH_REG	0x0F00	CCMU Security Switch Register
PLL_LOCK_DBG_CTRL_REG	0x0F04	PLL Lock Debug Control Register
PLL_CPUX_HW_FM_REG	0x0F20	PLL_CPUX Hardware FM Register
MOD_SPE_CLK_REG	0x0F30	Module Special Clock Register
HOSC_OUTPUT_CTRL_REG	0x0F40	HOSC Output Control Register

3.3.5 CCU Register Description

3.3.5.1 PLL_CPUX Control Register (Default Value: 0x0A00_1000)

Offset: 0x0000			**Register Name: PLL_CPUX_CTRL_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	PLL_ENABLE 0: Disable 1: Enable The PLL_CPUX= InputFreq*N/P. Where, the range of PLL_CPUX output clock is 200MHz~3GHz. When 24MHz oscillator is in used, and bit[23] = 0, Its default value is 408MHz.
30	/	/	/
29	R/W	0x0	LOCK_ENABLE Lock Enable 0: Disable 1: Enable
28	R	0x0	LOCK 0:Unlocked 1: Locked (It indicates that the PLL has been stable.)
27	R/W	0x1	PLL_OUTPUT_ENABLE 0:Disable 1:Enable It is used to control the PLL output enabling.
26:24	R/W	0x2	PLL_LOCK_TIME PLL Lock Time This bit is a stepping range from a frequency to another frequency,it can adjust the speed of clock changing. **NOTE** **Not recommended to modify the value.**
23	R/W	0x0	PLL Input Sel 0:PLL_24M

			1:HOSC
22:18	/	/	/
17:16	R/W	0x0	PLL_OUT_EXT_DIVP PLL Output External Divider P 00: 1 01: 2 10: 4 11: / When output clock is less than 288MHz,clock frequency is output by dividing P.
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, PLL_FACTOR_N should be more than or equal to 11
7:0	/	/	/

3.3.5.2 PLL_DDR Control Register (Default Value: 0x0800_2301)

Offset: 0x0010			**Register Name: PLL_DDR_CTRL_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	PLL_ENABLE 0: Disable 1: Enable The PLL_DDR = InputFreq*N/M0/M1. When 24MHz oscillator is in used, PLL_DDR is 432MHz by default.
30	/	/	/
29	R/W	0x0	LOCK_ENABLE Lock Enable 0: Disable 1: Enable
28	R	0x0	LOCK 0:Unlocked 1: Locked (It indicates that the PLL has been stable.)
27	R/W	0x1	PLL_OUTPUT_ENABLE 0:Disable 1:Enable It is used to control the PLL output enabling.
26:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE. 0: Disable. 1: Enable.
23	R/W	0x0	PLL Input Sel 0:PLL_24M 1:HOSC

22:16	/	/	/
15:8	R/W	0x23	PLL_FACTOR_N PLL Factor N N= PLL_FACTOR_N +1 PLL_FACTOR_N is from 0 to 254. In application, PLL_FACTOR_N should be more than or equal to 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.

3.3.5.3 PLL_PERI0 Control Register (Default Value: 0x0800_3100)

Offset: 0x0020			**Register Name: PLL_PERI0_CTRL_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	PLL_ENABLE 0: Disable 1: Enable The PLL_PERI0(2X) = InputFreq*N/M0/M1 The PLL_PERI0(1X) = InputFreq*N/M0/M1/2 When the 24MHz oscillator is in used and bit[23] = 0, PLL_PERI0(2X) is 1.2GHz by default. PLL_PERI0(2X) output clock should be 1.2GHz. It is not suggested to modify the value.
30	/	/	/
29	R/W	0x0	LOCK_ENABLE Lock Enable 0: Disable 1: Enable
28	R	0x0	LOCK 0:Unlocked 1: Locked (It indicates that the PLL has been stable.)
27	R/W	0x1	PLL_OUTPUT_ENABLE 0:Disable 1:Enable It is used to control the PLL output enabling.
26:24	/	/	/
23	R/W	0x0	PLL Input Sel 0:PLL_24M 1:HOSC

22:16	/	/	/
15:8	R/W	0x31	PLL_FACTOR_N PLL Factor N N= PLL_FACTOR_N +1 PLL_FACTOR_N is from 0 to 254. In application, PLL_FACTOR_N should be more than or equal to 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	0x0	PLL_OUTPUT_DIV_M0 PLL Output Div M0 M0=PLL_OUTPUT_DIV_M0 + 1 PLL_OUTPUT_DIV_M0 is from 0 to 1.

3.3.5.4 PLL_PERI1 Control Register (Default Value: 0x0800_3100)

Offset: 0x0028			**Register Name: PLL_PERI1_CTRL_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	PLL_ENABLE 0: Disable 1: Enable The PLL_PERI0(2X) = InputFreq*N/M0/M1 The PLL_PERI0(1X) = InputFreq*N/M0/M1/2 When the 24MHz oscillator is in used and bit[23] = 0, PLL_PERI0(2X) is 1.2GHz by default. PLL_PERI0(2X) output clock should be 1.2GHz. It is not suggested to modify the value.
30	/	/	/
29	R/W	0x0	LOCK_ENABLE Lock Enable 0: Disable 1: Enable
28	R	0x0	LOCK 0:Unlocked 1: Locked (It indicates that the PLL has been stable.)
27	R/W	0x1	PLL_OUTPUT_ENABLE 0:Disable 1:Enable It is used to control the PLL output enabling.
26:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE 0: Disable 1: Enable

23	R/W	0x0	PLL Input Sel 0:PLL_24M 1:HOSC
22:16	/	/	/
15:8	R/W	0x31	PLL_FACTOR_N PLL Factor N N= PLL_FACTOR_N +1 PLL_FACTOR_N is from 0 to 254. In application, PLL_FACTOR_N should be more than or equal to 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	0x0	PLL_OUTPUT_DIV_M0 PLL Output Div M0 M0=PLL_OUTPUT_DIV_M0 + 1 PLL_OUTPUT_DIV_M0 is from 0 to 1.

3.3.5.5 PLL_GPU Control Register (Default Value: 0x0800_2301)

Offset: 0x0030			**Register Name: PLL_GPU_CTRL_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	PLL_ENABLE 0: Disable 1: Enable The PLL_GPU = 24MHz*N/M0/M1. When HOSC is equal to 24MHz and bit[23] is equal to 0, PLL_GPU is 432MHz by default.
30	/	/	/
29	R/W	0x0	LOCK_ENABLE Lock Enable 0: Disable 1: Enable
28	R	0x0	LOCK 0:Unlocked 1: Locked (It indicates that the PLL has been stable.)
27	R/W	0x1	PLL_OUTPUT_ENABLE 0:Disable 1:Enable It is used to control the PLL output enabling.
26:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE 0: Disable 1: Enable

23	R/W	0x0	PLL Input Sel 0:PLL_24M 1:HOSC
22:16	/	/	/
15:8	R/W	0x23	PLL_FACTOR_N PLL Factor N N= PLL_FACTOR_N +1 PLL_FACTOR_N is from 0 to 254. In application, PLL_FACTOR_N should be more than or equal to 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.

3.3.5.6 PLL_VIDEO0 Control Register (Default Value: 0x0800_6203)

Offset: 0x0040			**Register Name: PLL_VIDEO0_CTRL_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	PLL_ENABLE 0: Disable 1: Enable For application, PLL_VIDEO0(4X)=InputFreq *N/M. PLL_VIDEO0(1X)=24MHz*N/M/4. When HOSC is 24MHz and bit[23] = 0, PLL_VIDEO0(4X) is 1188MHz by default.
30	/	/	/
29	R/W	0x0	LOCK_ENABLE Lock Enable 0: Disable 1: Enable
28	R	0x0	LOCK_STATUS 0:Unlocked 1: Locked (It indicates that the PLL has been stable.)
27	R/W	0x1	PLL_OUTPUT_ENABLE 0:Disable 1:Enable It is used to control the PLL output enabling.
26:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE

			0: Disable 1: Enable
23	R/W	0x0	PLL Input Sel 0:PLL_24M 1:HOSC
22:16	/	/	/
15:8	R/W	0x62	PLL_FACTOR_N PLL Factor N N= PLL_FACTOR_N +1 PLL_FACTOR_N is from 0 to 254. In application, PLL_FACTOR_N should be more than or equal to 11.
7:2	/	/	/
1	R/W	0x1	PLL_INPUT_DIV_M PLL Input Div M M1=PLL_INPUT_DIV_M + 1 PLL_INPUT_DIV_M is from 0 to 1.
0	R/W	0x1	PLL_OUTPUT_DIV_D PLL Output Div D (This factor is only for test.) M0=PLL_OUTPUT_DIV_D + 1 PLL_OUTPUT_DIV_D is from 0 to 1. For test, PLL_VIDEO0(4X) =24MHz*N/M/D

3.3.5.7 PLL_VIDEO1 Control Register (Default Value: 0x0800_6203)

Offset: 0x0048			**Register Name: PLL_VIDEO1_CTRL_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	PLL_ENABLE 0: Disable 1: Enable When PLL_24M Bypass， PLL_VIDEO1(4X)=InputFreq*N/M. PLL_VIDEO1(1X)= InputFreq*N/M/4. When HOSC is 24MHz and bit[23] = 0, PLL_VIDEO0(4X) is 1188MHz by default.
30	/	/	/
29	R/W	0x0	LOCK_ENABLE Lock Enable 0: Disable 1: Enable
28	R	0x0	LOCK_STATUS 0:Unlocked 1: Locked (It indicates that the PLL has been stable.)
27	R/W	0x1	PLL_OUTPUT_ENABLE 0:Disable 1:Enable

			It is used to control the PLL output enabling.
26:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE 0: Disable 1: Enable
23	R/W	0x0	PLL Input Sel 0:PLL_24M 1:HOSC
22:16	/	/	/
15:8	R/W	0x62	PLL_FACTOR_N PLL Factor N N= PLL_FACTOR_N +1 PLL_FACTOR_N is from 0 to 254. In application, PLL_FACTOR_N should be more than or equal to 11.
7:2	/	/	/
1	R/W	0x1	PLL_INPUT_DIV_M PLL Input Div M M1=PLL_INPUT_DIV_M + 1 PLL_INPUT_DIV_M is from 0 to 1.
0	R/W	0x1	PLL_OUTPUT_DIV_D PLL Output Div D(This factor is only for test.) M0=PLL_OUTPUT_DIV_D + 1 PLL_OUTPUT_DIV_M0 is from 0 to 1. For test, PLL_VIDEO1(4X) =24MHz*N/M/D

3.3.5.8 PLL_VE Control Register (Default Value: 0x0800_2301)

Offset: 0x0058			**Register Name: PLL_VE_CTRL_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	PLL_ENABLE 0: Disable 1: Enable The PLL_VE = InputFreq*N/M0/M1. When HOSC is 24MHz and bit[23] = 0, PLL_VE is 432MHz by default.
30	/	/	/
29	R/W	0x0	LOCK_ENABLE Lock Enable 0: Disable 1: Enable
28	R	0x0	LOCK-STATUS 0:Unlocked 1: Locked (It indicates that the PLL has been stable.)
27	R/W	0x1	PLL_OUTPUT_ENABLE 0:Disable 1:Enable

			It is used to control the PLL output enabling.
26:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE 0: Disable 1: Enable
23	R/W	0x0	PLL Input Sel 0:PLL_24M 1:HOSC
22:16	/	/	/
15:8	R/W	0x23	PLL_FACTOR_N PLL Factor N N= PLL_FACTOR_N +1 PLL_FACTOR_N is from 0 to 254. In application, PLL_FACTOR_N should be more than or equal to 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.

3.3.5.9 PLL_DE Control Register (Default Value: 0x0800_2301)

Offset: 0x0060			**Register Name: PLL_DE_CTRL_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	PLL_ENABLE 0: Disable 1: Enable The PLL_DE = InputFreq*N/M0/M1. When HOSC is 24MHz and bit[23] = 0, PLL_DE is 432MHz by default.
30	/	/	/
29	R/W	0x0	LOCK_ENABLE Lock Enable 0: Disable 1: Enable
28	R	0x0	LOCK_STATUS 0:Unlocked 1: Locked (It indicates that the PLL has been stable.)
27	R/W	0x1	PLL_OUTPUT_ENABLE 0:Disable 1:Enable It is used to control the PLL output enabling.

26:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE 0: Disable 1: Enable
23	R/W	0x0	PLL Input Sel 0:PLL_24M 1:HOSC
22:16	/	/	/
15:8	R/W	0x23	PLL_FACTOR_N PLL Factor N N= PLL_FACTOR_N +1 PLL_FACTOR_N is from 0 to 254. In application, PLL_FACTOR_N should be more than or equal to 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.

3.3.5.10 PLL_AUDIO Control Register (Default Value: 0x0814_2A01)

Offset: 0x0078			**Register Name: PLL_AUDIO_CTRL_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	PLL_ENABLE 0: Disable 1: Enable This PLL is for Audio. The PLL_AUDIO = InputFreq*N/(Input_div+1)/(Output_div+1)/(P+1). PLL_AUDIO(2X)= InputFreq *N/(Input_div+1)/4 PLL_AUDIO(4X) = InputFreq *N/(Input_div+1)/2 When HOSC is 24MHz and bit[23] = 0, PLL_AUDIO is 24.5714 MHz by default.
30	/	/	/
29	R/W	0x0	LOCK_ENABLE Lock Enable 0: Disable 1: Enable
28	R	0x0	LOCK_STATUS 0:Unlocked 1: Locked (It indicates that the PLL has been stable.)
27	R/W	0x1	PLL_OUTPUT_ENABLE

			0:Disable 1:Enable It is used to control the PLL output enabling.
26:25	/	/	/
24	R/W	0x0	PLL_SDM_ENABLE 0: Disable 1: Enable
23	R/W	0x0	PLL Input Sel 0:PLL_24M 1:HOSC
22	/	/	/
21:16	R/W	0x14	PLL_POST_DIV_T PLL Post-div T T= PLL_POST_DIV_T +1 PLL_POST_DIV_T is from 0 to 63.
15:8	R/W	0x2A	PLL_FACTOR_N PLL Factor N N= PLL_FACTOR_N +1 PLL_FACTOR_N is from 0 to 254. In application, PLL_FACTOR_N should be more than or equal to 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.

3.3.5.11 PLL_24M Control Register (Default: 0x2893_3101)

Offset: 0x00B8			Register Name: PLL_24M_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	PLL_ENABLE. 0: Disable. 1: Enable. The PLL_24M = HOSC*N/M0/M1/P. This PLL default value is 24M In order to obtain the equal Duty ratio of clock, M0 need to be set to 1.
30	/	/	/
29	R/W	0x1	LOCK_ENABLE Lock Enable 0: Disable 1: Enable

28	R	0x0	LOCK 0:Unlocked 1: Locked (It indicates that the PLL has been stable.)
27	R/W	0x1	PLL_OUTPUT_ENABLE 0:Disable 1:Enable It is used to control the PLL output enabling.
26:24	/	/	/
23	R/W	0x1	PLL_BYP 0:PLL Output 1:Bypass
22	/	/	/
21:16	R/W	0x13	PLL_POST_DIV_P. PLL Post-div P. P= PLL_POST_DIV_P +1 PLL_POST_DIV_P is from 0 to 63.
15:8	R/W	0x31	PLL_FACTOR_N PLL Factor N. N= PLL_FACTOR_N +1 PLL_FACTOR_N is from 0 to 254. In application, PLL_FACTOR_N should be more than or equal to 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.

3.3.5.12 PLL_DDR Pattern Control Register (Default Value: 0x0000_0000)

Offset: 0x0110			Register Name: PLL_DDR_PAT_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	SIG_DELT_PAT_EN Sigma-Delta Pattern Enable
30:29	R/W	0x0	SPR_FREQ_MODE Spread Frequency Mode 00: DC=0 01: DC=1 10: Triangular(1 bit) 11: Triangular(n bit)
28:20	R/W	0x0	WAVE_STEP Wave Step

19	R/W	0x0	SDM_CLK_SEL SDM Clock Select 0: 24MHz 1: 12MHz When PLL_INPUT_DIV_M1 is 1,this register is set to 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

3.3.5.13 PLL_PERI0 Pattern0 Control Register (Default: 0x0000_0000)

Offset: 0x0120			Register Name: PLL_PERI0_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: 24MHz 1: 12MHz When PLL_INPUT_DIV_M1 is '1', this register is set to 1.
18:17	R/W	0x0	FREQ. Frequency. 00: 31.5KHz 01: 32KHz 10: 32.5KHz 11: 33KHz
16:0	R/W	0x0	WAVE_BOT. Wave Bottom.

3.3.5.14 PLL_PERI0 Pattern1 Control Register (Default: 0x0000_0000)

Offset: 0x0124	Register Name: PLL_PERI0_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.

3.3.5.15 PLL_PERI1 Pattern0 Control Register (Default Value: 0x0000_0000)

Offset: 0x0128			Register Name: PLL_PERI1_PAT0_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	SIG_DELT_PAT_EN Sigma-Delta Pattern Enable
30:29	R/W	0x0	SPR_FREQ_MODE Spread Frequency Mode 00: DC=0 01: DC=1 10: Triangular(1 bit) 11: Triangular(n bit)
28:20	R/W	0x0	WAVE_STEP Wave Step
19	R/W	0x0	SDM_CLK_SEL SDM Clock Select 0: 24MHz 1: 12MHz When PLL_INPUT_DIV_M1 is 1,this register is set to 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

3.3.5.16 PLL_PERI1 Pattern1 Control Register (Default Value: 0x0000_0000)

Offset: 0x012C			Register Name: PLL_PERI1_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

3.3.5.17 PLL_GPU0 Pattern0 Control Register (Default Value: 0x0000_0000)

Offset: 0x0130			Register Name: PLL_GPU0_PAT0_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	SIG_DELT_PAT_EN Sigma-Delta Pattern Enable
30:29	R/W	0x0	SPR_FREQ_MODE Spread Frequency Mode 00: DC=0 01: DC=1 10: Triangular(1 bit) 11: Triangular(n bit)
28:20	R/W	0x0	WAVE_STEP Wave Step
19	R/W	0x0	SDM_CLK_SEL SDM Clock Select 0: 24MHz 1: 12MHz When PLL_INPUT_DIV_M1 is 1,this register is set to 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

3.3.5.18 PLL_GPU0 Pattern1 Control Register (Default Value: 0x0000_0000)

Offset: 0x0134			Register Name: PLL_GPU0_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

3.3.5.19 PLL_VIDEO0 Pattern0 Control Register (Default Value: 0x0000_0000)

Offset: 0x0140			Register Name: PLL_VIDEO0_PAT0_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	SIG_DELT_PAT_EN Sigma-Delta Pattern Enable
30:29	R/W	0x0	SPR_FREQ_MODE Spread Frequency Mode 00: DC=0 01: DC=1 10: Triangular(1 bit) 11: Triangular(n bit)
28:20	R/W	0x0	WAVE_STEP Wave Step
19	R/W	0x0	SDM_CLK_SEL SDM Clock Select 0: 24MHz 1: 12MHz When PLL_INPUT_DIV_M1 is 1, this register is set to 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

3.3.5.20 PLL_VIDEO0 Pattern1 Control Register (Default Value: 0x0000_0000)

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

3.3.5.21 PLL_VIDEO1 Pattern0 Control Register (Default Value: 0x0000_0000)

Offset: 0x0148			Register Name: PLL_VIDEO1_PAT0_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	SIG_DELT_PAT_EN

			Sigma-Delta Pattern Enable
30:29	R/W	0x0	SPR_FREQ_MODE Spread Frequency Mode 00: DC=0 01: DC=1 10: Triangular(1 bit) 11: Triangular(n bit)
28:20	R/W	0x0	WAVE_STEP Wave Step
19	R/W	0x0	SDM_CLK_SEL SDM Clock Select 0: 24MHz 1: 12MHz When PLL_INPUT_DIV_M1 is 1, this register is set to 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

3.3.5.22 PLL_VIDEO1 Pattern1 Control Register (Default Value: 0x0000_0000)

Offset: 0x014C			Register Name: PLL_VIDEO1_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

3.3.5.23 PLL_VE Pattern0 Control Register (Default Value: 0x0000_0000)

Offset: 0x0158			Register Name: PLL_VE_PAT0_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	SIG_DELT_PAT_EN Sigma-Delta Pattern Enable
30:29	R/W	0x0	SPR_FREQ_MODE Spread Frequency Mode 00: DC=0 01: DC=1

			10: Triangular(1 bit) 11: Triangular(n bit)
28:20	R/W	0x0	WAVE_STEP Wave Step
19	R/W	0x0	SDM_CLK_SEL SDM Clock Select 0: 24MHz 1: 12MHz When PLL_INPUT_DIV_M1 is 1, this register is set to 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

3.3.5.24 PLL_VE Pattern1 Control Register (Default Value: 0x0000_0000)

Offset: 0x015C			Register Name: PLL_VE_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

3.3.5.25 PLL_DE Pattern0 Control Register (Default Value: 0x0000_0000)

Offset: 0x0160			Register Name: PLL_DE_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 11: Triangular
28:20	R/W	0x0	WAVE_STEP Wave Step
19	R/W	0x0	SDM_CLK_SEL

			SDM Clock Select 0: 24MHz 1: 12MHz When PLL_INPUT_DIV_M1 is 1, this register is set to 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

3.3.5.26 PLL_DE Pattern1 Control Register (Default Value: 0x0000_0000)

Offset: 0x0164			Register Name: PLL_DE_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

3.3.5.27 PLL_AUDIO Pattern0 Control Register (Default Value: 0x0000_0000)

Offset: 0x0178			Register Name: PLL_AUDIO_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 11: Triangular
28:20	R/W	0x0	WAVE_STEP Wave Step
19	R/W	0x0	SDM_CLK_SEL SDM Clock Select 0: 24MHz 1: 12MHz When PLL_INPUT_DIV_M1 is 1, this register is set to 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

3.3.5.28 PLL_AUDIO Pattern1 Control Register (Default Value: 0x0000_0000)

Offset: 0x017C			Register Name: PLL_AUDIO_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

3.3.5.29 PLL_CPUX Bias Register (Default Value: 0x8010_0000)

Offset: 0x0300			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	/	/	/

3.3.5.30 PLL_DDR Bias Register (Default Value: 0x0003_0000)

Offset: 0x0310			Register Name: PLL_DDR_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	/	/	/

3.3.5.31 PLL_PERI0 Bias Register (Default Value: 0x0003_0000)

Offset: 0x0320	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	/	/	/

3.3.5.32 PLL_PERI1 Bias Register (Default Value: 0x0003_0000)

Offset: 0x0328			Register Name: PLL_PERI1_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	/	/	/

3.3.5.33 PLL_GPU0 Bias Register (Default Value: 0x0003_0000)

Offset: 0x0330			Register Name: PLL_GPU0_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	/	/	/

3.3.5.34 PLL_VIDEO0 Bias Register (Default Value: 0x0003_0000)

Offset: 0x0340			Register Name: PLL_VIDEO0_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	/	/	/

3.3.5.35 PLL_VIDEO1 Bias Register (Default Value: 0x0003_0000)

Offset: 0x0348			Register Name: PLL_VIDEO1_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	/	/	/

3.3.5.36 PLL_VE Bias Register (Default Value: 0x0003_0000)

Offset: 0x0358			Register Name: PLL_VE_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	/	/	/

3.3.5.37 PLL_DE Bias Register (Default Value: 0x0003_0000)

Offset: 0x0360			Register Name: PLL_DE_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	/	/	/

3.3.5.38 PLL_AUDIO Bias Register (Default Value: 0x0003_0000)

Offset: 0x0378			Register Name: PLL_AUDIO_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	/	/	/

3.3.5.39 PLL_24M Bias Register (Default: 0x0003_0000)

Offset: 0x03B8			Register Name: PLL_24M _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	/	/	/

3.3.5.40 PLL_CPUX Tuning Register (Default Value: 0x4440_4000)

Offset: 0x0400			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 verify
14:8	R/W	0x40	C_B_IN C-B-IN [6:0] for verify
7	R/W	0x0	C_OD1 C-REG-OD1 for verify
6:0	RO	0x0	C_B_OUT C-B-OUT [6:0] for verify

3.3.5.41 CPUX_AXI Configuration Register (Default Value: 0x0000_0301)

Offset: 0x0500			Register Name: CPUX_AXI_CFG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: OSC24M 01: RTC_32K 10: RC16M 11: PLL_CPUX CPUX Clock = Clock Source CPUX_AXI Clock = Clock Source/M CPUX _APB Clock = Clock Source/N
23:10	/	/	/
9:8	R/W	0x3	CPUX _APB_FACTOR_N 0: /1 1: /2 2&3: /4
7:2	/	/	/
1:0	R/W	0x1	FACTOR_ M Factor M.(M= FACTOR_M +1) The range of M is from 1 to 4.

3.3.5.42 PSI_AHB1_AHB2 Configuration Register (Default Value: 0x0000_0000)

Offset: 0x0510	Register Name: PSI_AHB1_AHB2_CFG_REG

Bit	Read/Write	Default/Hex	Description
31:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: OSC24M 01: RTC_32K 10: RC16M 11: PLL_PERI0(1X) PSI_AHB1_AHB2 CLK = Clock Source/M/N.
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:2	/	/	/
1:0	R/W	0x0	FACTOR_M Factor M.(M= FACTOR_M +1) The range of M is from 1 to 4.

3.3.5.43 AHB3 Configuration Register (Default Value: 0x0000_0000)

Offset: 0x051C			Register Name: AHB3_CFG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: OSC24M 01: RTC_32K 10: PSI 11: PLL_PERI0(1X) AHB3 CLK = Clock Source/M/N.
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:2	/	/	/
1:0	R/W	0x0	FACTOR_M Factor M.(M= FACTOR_M +1) The range of M is from 1 to 4.

3.3.5.44 APB1 Configuration Register (Default Value: 0x0000_0000)

Offset: 0x520			Register Name: APB1_CFG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: OSC24M 01: RTC_32K 10: PSI 11: PLL_PERI0(1X) APB1 CLK = Clock Source/M/N.
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:2	/	/	/
1:0	R/W	0x0	FACTOR_M Factor M.(M= FACTOR_M +1) The range of M is from 1 to 4.

3.3.5.45 APB2 Configuration Register (Default Value: 0x0000_0000)

Offset: 0x524			Register Name: APB2_CFG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: OSC24M 01: RTC_32K 10: PSI 11: PLL_PERI0(1X) APB2 CLK = Clock Source/M/N.
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:2	/	/	/
1:0	R/W	0x0	FACTOR_M

			Factor M.(M= FACTOR_M +1) The range of M is from 1 to 4.

3.3.5.46 MBUS Configuration Register (Default Value: 0xC000_0000)

Offset: 0x540			Register Name: MBUS_CFG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x1	CLK_GATING Gating Special Clock 0: Clock is OFF 1: Clock is ON MBUS CLK = Clock Source/M.
30	R/W	0x1	MBUS_RST MBUS Reset 0: Assert 1: De-assert
29:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: OSC24M 01: PLL_PERI0(2X) 10: PLL_DDR 11: /
23:3	/	/	/
2:0	R/W	0x0	FACTOR_M Factor M.(M= FACTOR_M +1) The range of M is from 1 to 8.

3.3.5.47 DE Clock Register (Default Value: 0x0000_0000)

Offset: 0x0600			Register Name: DE_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.
30:25	/	/	/
24	R/W	0x0	CLK_SRC_SEL Clock Source Select 0: PLL_DE 1: PLL_PERI0(2X)
23:4	/	/	/
3:0	R/W	0x0	FACTOR_M

			Factor M.(M= FACTOR_M +1) The range of M is from 1 to 16.

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

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

3.3.5.49 G2D Clock Register (Default: 0x0000_0000)

Offset: 0x0630			Register Name: G2D_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.
30:25	/	/	/
24	R/W	0x0	CLK_SRC_SEL. Clock Source Select 0: PLL_DE 1: PLL_PERI0(2X)
23:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M.(M= FACTOR_M +1) FACTOR_M is from 0 to 15.

3.3.5.50 G2D Bus Gating Reset Register (Default: 0x0000_0000)

Offset: 0x063C			Register Name: G2D _BGR_REG
Bit	Read/Write	Default/Hex	Description
31:17	/	/	/
16	R/W	0x0	G2D_RST. G2D Reset.

			0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	G2D_GATING. Gating Clock For G2D 0: Mask 1: Pass

3.3.5.51 GPU Clock Register (Default Value: 0x0000_0000)

Offset: 0x0670			Register Name: GPU_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.
30:25	/	/	/
24	R/W	0x0	CLK_SRC_SEL Clock Source Select 0: PLL_GPU 1: /
23:3	/	/	/
2:0	R/W	0x0	FACTOR_M Factor M.(M= FACTOR_M +1) The range of M is from 1 to 8.

3.3.5.52 GPU Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x067C			Register Name: GPU_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	GPU_RST GPU Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	GPU_GATING Gating Clock For GPU 0: Mask 1: Pass

3.3.5.53 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: OSC24M 1: PLL_PERI0(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) The range of M is from 1 to 16.

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

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

3.3.5.55 VE Clock Register (Default Value: 0x0000_0000)

Offset: 0x0690	Register Name: VE_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 M.
30:25	/	/	/
24	R/W	0x0	CLK_SRC_SEL Clock Source Select 0: PLL_VE 1: /
23:3	/	/	/
2:0	R/W	0x0	FACTOR_M Factor M.(M= FACTOR_M +1) The range of M is from 1 to 8.

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

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

3.3.5.57 EMCE Clock Register (Default Value: 0x0000_0000)

Offset: 0x06B0			Register Name: EMCE_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: OSC24M

			1: PLL_PERI0(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) The range of M is from 1 to 16.

3.3.5.58 EMCE Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x06BC			Register Name: EMCE_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	EMCE_RST EMCE Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	EMCE_GATING Gating Clock For EMCE 0: Mask 1: Pass

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

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

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

Offset: 0x071C			Register Name: MSGBOX_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	MSGBOX_RST MSGBOX Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	MSGBOX_GATING Gating Clock For MSGBOX 0: Mask 1: Pass

3.3.5.61 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

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

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

3.3.5.63 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 0: Clock is OFF 1: Clock is ON SCLK = OSC24M.
30:0	/	/	/

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

Offset: 0x078C			Register Name: DBGSYS_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	DBGSYS_RST DBGSYS Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	DBGSYS_GATING Gating Clock For DBGSYS 0: Mask 1: Pass

3.3.5.65 PSI Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x079C			Register Name: PSI_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	PSI_RST PSI Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	PSI_GATING Gating Clock For PSI 0: Mask 1: Pass

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

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

3.3.5.67 IOMMU Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x07BC			Register Name: IOMMU_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	IOMMU_GATING Gating Clock For IOMMU 0: Mask 1: Pass

3.3.5.68 DRAM Clock Register (Default Value: 0x0000_0000)

Offset: 0x0800			Register Name: DRAM_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	/	/	/
30	R/W	0x0	MODULE_RST Module Reset 0: Assert 1: De-assert SCLK = Clock Source/M.
29:28	/	/	/
27	R/WAC	0x0	SDRCLK_UPD SDRCLK Configuration 0 Update. 0:Invalid 1:Valid Setting this bit will validate Configuration 0. It will be auto cleared after the Configuration 0 is valid.
26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL

			Clock Source Select 00: PLL_DDR Others: /
23:2	/	/	/
1:0	R/W	0x0	FACTOR_M Factor M.(M= FACTOR_M +1) The range is from 1 to 4

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

Offset: 0x0804			Register Name: MBUS_MAT_CLK_GATING_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:25	/	/	/
24	R/W	0x0	EMCE_MCLK_GATING. Gating MBUS Clock For EMCE 0: Mask 1: Pass
23:11	/	/	/
10	R/W	0x0	G2D_MCLK_GATING. Gating MBUS Clock For G2D 0: Mask 1: Pass
9	R/W	0x0	/
8	R/W	0x0	CSI_MCLK_GATING Gating MBUS Clock For CSI 0: Mask 1: Pass
7:6	/	/	/
5	R/W	0x0	NDFC_MCLK_GATING Gating MBUS Clock For NDFC 0: Mask 1: Pass
4:3	/	/	/
2	R/W	0x0	CE_MCLK_GATING Gating MBUS Clock For CE 0: Mask 1: Pass
1	R/W	0x0	VE_MCLK_GATING Gating MBUS Clock For VE 0: Mask 1: Pass
0	R/W	0x0	DMA_MCLK_GATING Gating MBUS Clock For DMA 0: Mask 1: Pass

NOTE

DE MCLK puts in DE module.

3.3.5.70 DRAM Bus Gating Reset Register (Default Value: 0x0000_0000)

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	0x0	DRAM_GATING Gating Clock For DRAM 0: Mask 1: Pass

3.3.5.71 NDFC Clock0 Register (Default Value: 0x0000_0000)

Offset: 0x0810			Register Name: NDFC_CLK0_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: OSC24M 001: PLL_PERI0(1X) 010: PLL_PERI1(1X) 011: PLL_PERI0(2X) 100: PLL_PERI1(2X) 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) The range of M is from 1 to 16.

3.3.5.72 NDFC Clock1 Register (Default Value: 0x0000_0000)

Offset: 0x0814			Register Name: NDFC_CLK1_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: OSC24M 001: PLL_PERI0(1X) 010: PLL_PERI1(1X) 011: PLL_PERI0(2X) 100: PLL_PERI1(2X) 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) The range of M is from 1 to 16.

3.3.5.73 NDFC Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x082C			Register Name: NDFC_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	NDFC_RST NDFC Reset 0: Assert 1: De-assert

15:1	/	/	/
0	R/W	0x0	NDFC_GATING Gating Clock For NDFC 0: Mask 1: Pass

3.3.5.74 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 000: OSC24M 001: PLL_PERI0(2X) 010: PLL_PERI1(2X) 011: /
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) The range of M is from 1 to 16.

3.3.5.75 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 000: OSC24M 001: PLL_PERI0(2X) 010: PLL_PERI1(2X) 011: /
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) The range of M is from 1 to 16

3.3.5.76 SMHC2 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0838			**Register Name: SMHC2_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 000: OSC24M 001: PLL_PERI0(2X) 010: PLL_PERI1(2X) 011: /
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) The range of M is from 1 to 16.

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

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

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

Offset: 0x090C			Register Name: UART_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:21	/	/	/
20	R/W	0x0	UART4_RST. UART4 Reset. 0: Assert 1: De-assert
19	R/W	0x0	UART3_RST UART3 Reset 0: Assert

			1: De-assert
18	R/W	0x0	UART2_RST UART2 Reset 0: Assert 1: De-assert
17	R/W	0x0	UART1_RST UART1 Reset 0: Assert 1: De-assert
16	R/W	0x0	UART0_RST UART0 Reset 0: Assert 1: De-assert
15:5	/	/	/
4	R/W	0x0	UART4_GATING. Gating Clock For UART4 0: Mask 1: Pass
3	R/W	0x0	UART3_GATING Gating Clock For UART3 0: Mask 1: Pass
2	R/W	0x0	UART2_GATING Gating Clock For UART2 0: Mask 1: Pass
1	R/W	0x0	UART1_GATING Gating Clock For UART1 0: Mask 1: Pass
0	R/W	0x0	UART0_GATING Gating Clock For UART0 0: Mask 1: Pass

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

Offset: 0x091C			Register Name: TWI_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:19	/	/	/
18	R/W	0x0	TWI2_RST TWI2 Reset 0: Assert 1: De-assert
17	R/W	0x0	TWI1_RST

			TWI1 Reset 0: Assert 1: De-assert
16	R/W	0x0	TWI0_RST TWI0 Reset 0: Assert 1: De-assert
15:3	/	/	/
2	R/W	0x0	TWI2_GATING Gating Clock For TWI2 0: Mask 1: Pass
1	R/W	0x0	TWI1_GATING Gating Clock For TWI1 0: Mask 1: Pass
0	R/W	0x0	TWI0_GATING Gating Clock For TWI0 0: Mask 1: Pass

3.3.5.80 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: OSC24M 001: PLL_PERI0(1X) 010: PLL_PERI1(1X) 011: PLL_PERI0(2X) 100: PLL_PERI1(2X) 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) The range of M is from 1 to 16.

3.3.5.81 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: OSC24M 001: PLL_PERI0(1X) 010: PLL_PERI1(1X) 011: PLL_PERI0(2X) 100: PLL_PERI1(2X) 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) The range of M is from 1 to 16.

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

Offset: 0x096C			Register Name: SPI_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17	R/W	0x0	SPI1_RST SPI1 Reset 0: Assert

			1: De-assert
16	R/W	0x0	SPI0_RST SPI0 Reset 0: Assert 1: De-assert
15:2	/	/	/
1	R/W	0x0	SPI1_GATING Gating Clock For SPI1 0: Mask 1: Pass
0	R/W	0x0	SPI0_GATING Gating Clock For SPI0 0: Mask 1: Pass

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

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

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

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

3.3.5.85 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.
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: PLL_AUDIO 01: PLL_AUDIO(2X) 10: PLL_AUDIO(4X) 11: /
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:0	/	/	/

3.3.5.86 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.
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: PLL_AUDIO 01: PLL_AUDIO(2X) 10: PLL_AUDIO(4X) 11: /
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N

			Factor N 00: 1 01: 2 10: 4 11: 8
7:0	/	/	/

3.3.5.87 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
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

3.3.5.88 DMIC Clock Register (Default Value: 0x0000_0000)

Offset: 0x0A40			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/N.
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: PLL_AUDIO 01: PLL_AUDIO(2X)

			10: PLL_AUDIO(4X) 11: /
23:10	/	/	/
9:8	R/W	0x0	FACTOR_N Factor N 00: 1 01: 2 10: 4 11: 8
7:0	/	/	/

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

Offset: 0x0A4C			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

3.3.5.90 AUDIO CODEC 1X Clock Register (Default Value: 0x0000_0000)

Offset: 0x0A50			Register Name: AUDIO_CODEC_1X_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.
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: PLL_AUDIO 01: PLL_AUDIO(2X) 10: PLL_AUDIO(4X) 11: /
23:4	/	/	/
3:0	R/W	0x0	FACTOR_M

			Factor M.(M= FACTOR_M +1) The factor of M is from 1 to 16.

3.3.5.91 AUDIO CODEC 4X Clock Register (Default Value: 0x0000_0000)

Offset: 0x0A54			Register Name: AUDIO_CODEC_4X_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.
30:26	/	/	/
25:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 00: PLL_AUDIO 01: PLL_AUDIO(2X) 10: PLL_AUDIO(4X) 11: /
23:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M.(M= FACTOR_M +1) The factor of M is from 1 to 16.

3.3.5.92 AUDIO CODEC Bus Gating Reset Register (Default Value: 0x0000_0000)

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

3.3.5.93 USB2.0_OTG Clock Register (Default Value: 0x0003_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
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 USBPHY0 0: Clock is OFF 1: Clock is ON SCLK is from OSC24M
28:26	/	/	/
25:24	R/W	0x0	OHCI0_12M_SRC_SEL OHCI0 12M Source Select 00: 12MHz divided from 48MHz 01: 12MHz divided from 24MHz 10: LOSC 11: RC16M
23:18	/	/	/
17	R/W	0x1	PYH_SCLK_SRC_SEL 0:HOSC 1:PLL_24M
16	R/W	0x1	PYH_SCLK_CLK_DIV_Bypass 0:Div2 1:Bypass
15:0	/	/	/

3.3.5.94 USB2.0_HOST Clock Register (Default Value: 0x0003_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 USBPHY1 0: Clock is OFF 1: Clock is ON

28:26	/	/	/
25:24	R/W	0x0	OHCI1_12M_SRC_SEL OHCI1 12M Source Select 00: 12M divided from 48MHz 01: 12M divided from 24MHz 10: LOSC 11: RC16M
23:18	/	/	/
17	R/W	0x1	PYH_SCLK_SRC_SEL 0:HOSC 1:PLL_24M
16	R/W	0x1	PYH_SCLK_CLK_DIV_Bypass 0:Div2 1:Bypass
15:0	/	/	/

3.3.5.95 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	USBOTG_RST USBOTG Reset 0: Assert 1: De-assert
23:22	/	/	/
21	R/W	0x0	USBEHCI1_RST USBEHCI1 Reset 0: Assert 1: De-assert
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	/	/	/
8	R/W	0x0	USBOTG_GATING

			Gating Clock For USBOTG 0: Mask 1: Pass
7:6	/	/	/
5	R/W	0x0	USBEHCI1_GATING Gating Clock For USBEHCI1 0: Mask 1: Pass
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

3.3.5.96 MIPI DSI DPHY0 High Speed Clock Register (Default Value: 0x0000_0000)

Offset: 0x0B20			**Register Name: MIPI_DSI_DPHY0_HS_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_VIDEO0(1X) 01: PLL_VIDEO0(4X) 10:PLL_VIDEO1(1X) 11:/
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) The range of M is from 1 to 16.

3.3.5.97 MIPI DSI Host0 Clock Register (Default Value: 0x0000_0000)

Offset: 0x0B24			Register Name: MIPI_DSI_HOST0_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_PERI0(1X) 01: PLL_PERI0(2X) 10: OSC24M 11:/
23:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M.(M= FACTOR_M +1) The range of M is from 1 to 16.

3.3.5.98 MIPI DSI Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x0B4C			Register Name: MIPI_DSI_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	MIPI_DSI0_RST MIPI_DSI0 Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	MIPI_DSI0_GATING Gating Clock For MIPI_DSI0 0: Mask 1: Pass

3.3.5.99 DISPLAY_IF_TOP Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x0B5C			Register Name: DISPLAY_IF_TOP_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	DISPLAY_IF_TOP_RST DISPLAY_IF_TOP Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	DISPLAY_IF_TOP_GATING Gating Clock For DISPLAY_IF_TOP 0: Mask 1: Pass

3.3.5.100 TCON LCD Clock Register (Default Value: 0x0000_0000)

Offset: 0x0B60			Register Name: TCON_LCD_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
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: PLL_VIDEO0(1X) 001: PLL_VIDEO0(4X) 010: PLL_VIDEO1(1X) Others:/
23:0	/	/	/

3.3.5.101 TCON LCD Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x0B7C			Register Name: TCON_LCD_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	TCON_LCD_RST TCON_LCD Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	TCON_LCD_GATING

			Gating Clock For TCON_LCD 0: Mask 1: Pass

3.3.5.102 LVDS BUS GATING RESET Register (Default: 0x0000_0000)

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

3.3.5.103 CSI MISC Clock Register (Default Value: 0x0000_0000)

Offset: 0x0C00			**Register Name: CSI_MISC_CLK_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:9	/	/	/
8	R/W	0x0	CSI0_MIPI0_MISC_CLK_GATING Gating CSI0_MIPI0 Special Clock,Clock source is OSC24M 0: Clock is OFF 1: Clock is ON
7:1	/	/	/
0	R/W	0x0	CSI_CCI_CLK_GATING Gating CCI Special Clock, Clock source is OSC24M 0: Clock is OFF 1: Clock is ON

3.3.5.104 CSI TOP Clock Register (Default Value: 0x0000_0000)

Offset: 0x0C04			**Register Name: CSI_TOP_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.
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select

			000:PLL_VIDEO0(1X) 001:/ 010:PLL_VE 011:PLL_PERI0(1X) Others:/
23:4	/	/	/
3:0	R/W	0x0	FACTOR_M Factor M.(M= FACTOR_M +1) FACTOR_M is from 0 to 15

3.3.5.105 CSI Master Clock Register (Default Value: 0x0000_0000)

Offset: 0x0C08			Register Name: CSI_MST_CLK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	MCLK0_GATING Gating CSI Master Clock0,this clock output to external device. 0: Clock is OFF 1: Clock is ON MCLK0 = Clock Source/M
30:27	/	/	/
26:24	R/W	0x0	CLK_SRC_SEL Clock Source Select 000: OSC24M 001: PLL_VIDEO0(1X) 010: PLL_PERI0(1X) 011: PLL_PERI1(1X) Others:/
23:5	/	/	/
4:0	R/W	0x0	FACTOR_M Factor M.(M= FACTOR_M +1) The range of M is from 1 to 32.

3.3.5.106 CSI Bus Gating Reset Register (Default Value: 0x0000_0000)

Offset: 0x0C2C			Register Name: CSI_BGR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	CSI_RST CSI Reset 0: Assert 1: De-assert
15:1	/	/	/
0	R/W	0x0	CSI_GATING Gating Clock For CSI

			0: Mask 1: Pass

3.3.5.107 CCMU Security Switch Register (Default Value: 0x0000_0000)

Offset: 0x0F00			Register Name: CCMU_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

NOTE

If the secure bit of SID is not burned, the register is invalid.

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

Offset: 0x0F04			Register Name: PLL_LOCK_DBG_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	DBG_EN Debug Enable 0: Disable 1: Enable
30:25	/	/	/
24:20	R/W	0x0	DBG_SEL Debug Select 00000: PLL_CPUX 00001: / 00010: PLL_DDR 00011:/ 00100: PLL_PERI0(2X) 00101: PLL_PERI1(2X) 00110:PLL_GPU 00111:/ 01000: PLL_VIDEO0(4X)

			01001: PLL_VIDEO1(4X) 01010: / 01011: PLL_VE 01100: PLL_DE 01101: / 01110: / 01111: PLL_AUDIO 10000: PLL_24M 10001: / 10010: / 10011: / 10100: / 10101: / 10110: / 10111: / 11000: / 11001: / 11010: / Others: /
19	/	/	/
18:17	R/W	0x0	UNLOCK_LEVEL Unlock Level 00: 21-29 Clock Cycles 01: 22-28 Clock Cycles 1X: 20-30 Clock Cycles
16	R/W	0x0	LOCK_LEVEL Lock Level 0: 24-26 Clock Cycles 1: 23-27 Clock Cycles
15:0	/	/	/

3.3.5.109 PLL_CPUX Hardware FM Register (Default Value: 0x0000_0000)

Offset: 0x0F20			Register Name: PLL_CPUX_HW_FM_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	PLL_CPUX_MODE 0: Normal Mode 1: Hardware FM Mode In normal mode,the frequency of PLL_CPUX depends on the configuration of PLL_CPUX_CTRL_REG. In hardware FM mode，PLL_CPUX uses hardware FM.
30	R	0x0	PMU_FLAG 0: Unlock 1: Lock The flag is overcurrent flag signal of PMU send. When overcurrent, the status

			can change to Lock, at this time PLL_CPUX is divided down frequency; when overcurrent is cleared, the status can change to Unlock, at this time the frequency of PLL_CPUX depends on the configuration of PLL_CPUX_CTRL_REG.
29:18	/	/	/
17:16	R/W	0x0	PLL_OUT_EXT_DIVP PLL Output External Divider P 00: 1 01: 2 10: 4 11: / When output clock is less than 288MHz,clock frequency is output by dividing P.
15:8	R/W	0x00	PLL_FACTOR_N PLL Factor N N= PLL_FACTOR_N +1 PLL_FACTOR_N is from 0 to 254. In application, PLL_FACTOR_N should be more than or equal to 11.
7:2	/	/	/
1:0	R/W	0x0	PLL_FACTOR_M PLL Factor M M = PLL_FACTOR_M + 1 PLL_FACTOR_M is from 0 to 3.

3.3.5.110 Module Special Clock Register (Default: 0x0007_0000)

Offset: 0x0F30			Register Name: MOD_SPE_CLK_REG
Bit	Read/Write	Default/Hex	Description
31:19			
18	R/W	0x1	CSI_24M_CLK_GATING 0:Mask 1:Pass
17	R/W	0x1	THS\GPADC_24M_CLK_GATING 0:Mask 1:Pass
16	R/W	0x1	ACodec_24M_CLK_GATING 0:Mask 1:Pass
	/	/	
3:2	R/W	0x0	CSI_24M_SRC_SEL 0:PLL_24M 1:24M from PLL_PERI0(1x)/25 2:HOSC 3:/
1	R/W	0x0	THS_24M_SRC_SEL

			0:PLL_24M 1:HOSC
0	R/W	0x0	ACodec_24M_SRC_SEL 0: PLL_24M 1: HOSC

3.3.5.111 HOSC Output Control Register (Default: 0x0000_0000)

Offset: 0x0F40			Register Name: HOSC_OUTPUT_CTRL_REG
Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R/W	0x0	0: When external oscillator is 19.2MHz, 24MHz, or 38.4MHz, the HOSC output is 19.2MHz, 24MHz, or 19.2MHz. 1: When external oscillator is 19.2MHz, 24MHz, or 38.4MHz, the HOSC output is 19.2MHz, 24MHz, or 38.4MHz. It is configured only in the bypass 38.4MHz oscillator.

3.4 Boot System

3.4.1 Overview

The A50 has several ways to boot. It has an integrated the on-chip Boot ROM (BROM) which could be considered the primary program-loader. On startup, A50 starts to fetch the first instruction from address 0x0, where is the BROM located at.
The Boot system is split up into two parts :FEL and Media Boot. The task of FEL is to write the external data to the local NVM, the task of the Media Boot is to loaded from NVM an effective and legitimate BOOT0 up and running.

The Boot system includes the following features:

- Supports CPU0 Boot Process and NON_CPU0 Boot Process
- Supports super standby wakeup process
- Supports Hotplug Process
- Supports mandatory upgrade process through SMHC0 and USB
- Supports GPIO pin or eFuse to select the kind of boot media to boot
- Supports Normal Boot and Secure Boot
- Supports loads only certified firmware for Secure Boot
- Ensures that the Secure Boot is a trusted environment

3.4.2 Operations and Functional Descriptions

3.4.2.1 BROM Description

If the A50 has implemented the ARM TrustZone technology,when the *Secure Enable Bit* is enabled, the A50 will enable the ARM TrustZone technology.
So the BROM is divided into Normal BROM and Secure BROM. The Secure BROM protects against the potential threat of attackers modifying areas of code or data in programmable memory .
On startup,the A50 will read the Secure Enable Bit,if the bit is 0,then mapping Normal BROM code to address 0x0, or mapping Secure BROM code to address 0x0.

3.4.2.2 BROM Process

3.4.2.2.1 Normal BROM Process

In Normal BROM mode,the system boot will start from CPU0 or NON_CPU0, then BROM will read CPU ID number to distinguish between CPU0 or NON_CPUO, then BROM will read the Hotplug Flag Regsiter and the Supper Standby Flag Regsiter ,according to the flag whether to go through the appropriate process.Finally,BROM will read the state of the FEL Pin,if the FEL Pin signal is detected to pulled to high level, then the system will jump to the Try Media Boot

process,or jump to the mandatory upgrade process. Figure 3-7 shows the Normal BROM Process.

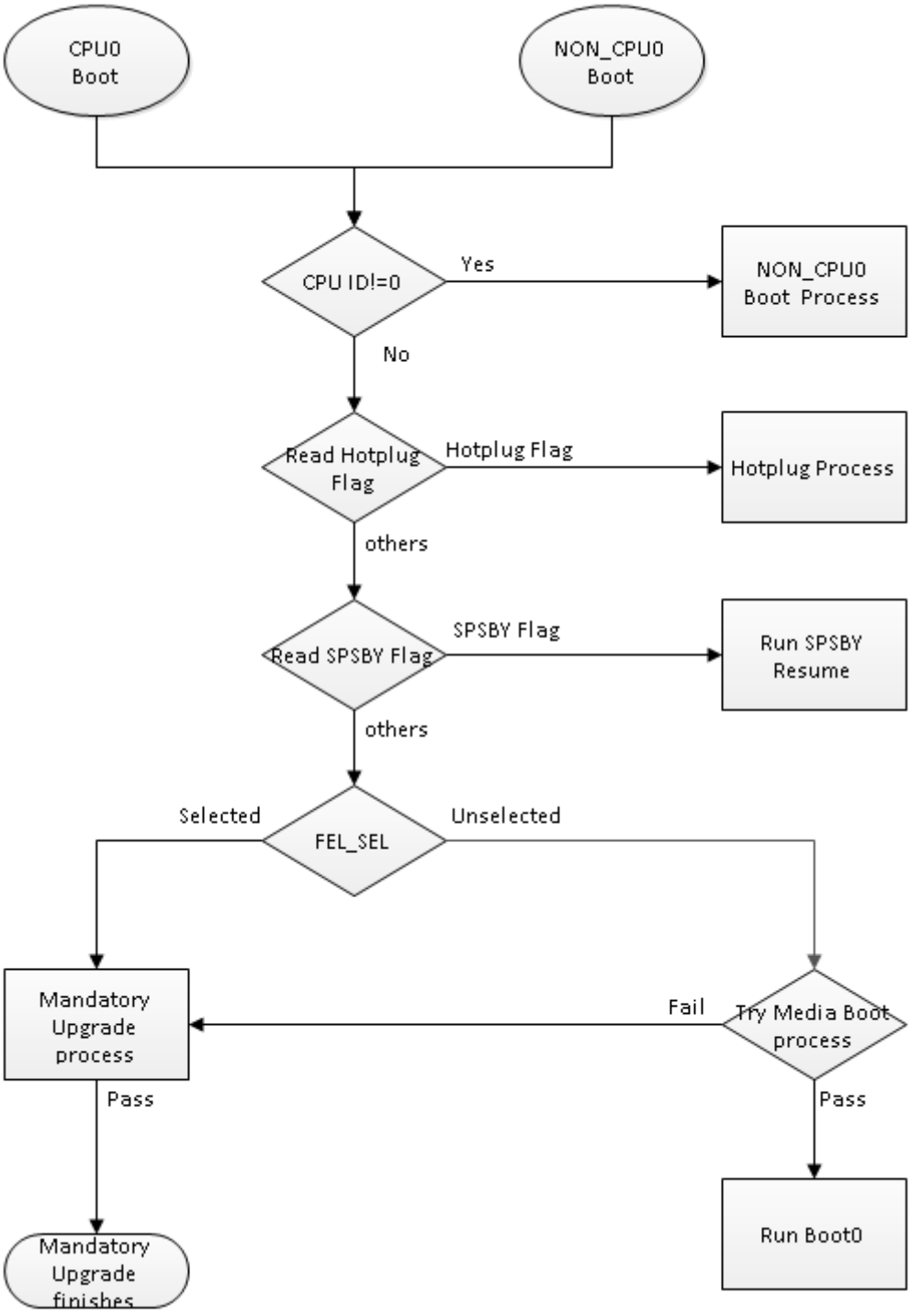


Figure 3-1. Normal BROM Process

3.4.2.2.2 Secure BROM Process

In Security BROM mode, after the try media boot process finishes, the system will go to run Security BROM software. Figure 3-8 shows the Secure BROM Process.

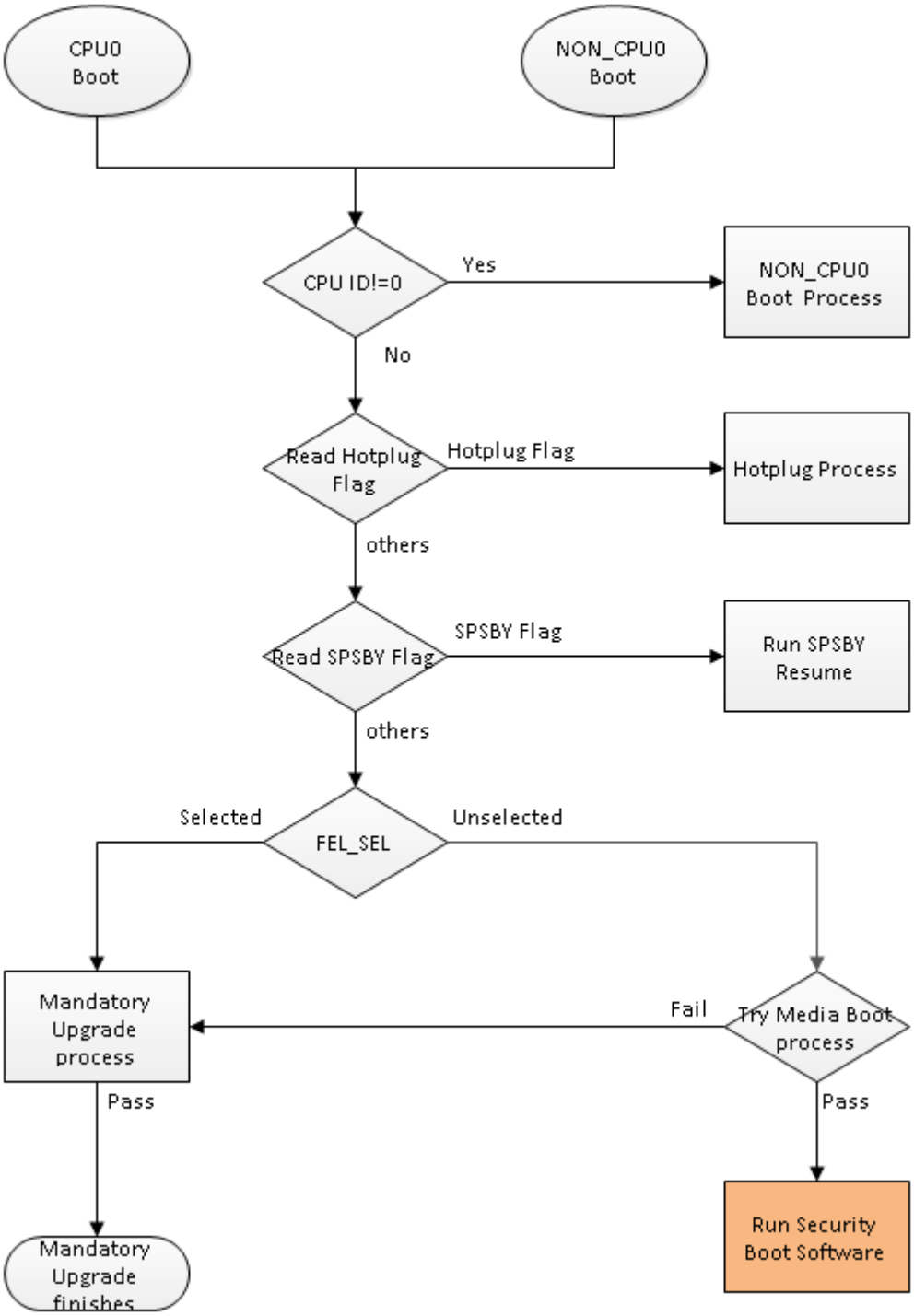


Figure 3-2. Secure BROM Process

The Secure BROM includes the following features:

- Supports X509 certificate
- Supports cryptographic algorithms: AES128,SHA256,RSA2048,AES,DES
- Supports OTP/eFuse

Before runing Security Boot software ,the software must check the software whether it has been modified or replaced,so the system will check and verify the integrity of the certificate,because the certificate has been using the RSA algorithm signature.The system also uses the Crypto Engine (CE) hardware module to accelerate the speed of encryption and decryption. Accordding to using standard cryptography to ensure that the firmware images can be trusted,so the Secure BROM ensures that the system security state is as expected.

3.4.2.3 NON_CPU0 Boot Process

If CPU ID is greater than 0, the system boot from NON_CPU0, BROM will read the Soft Entry Address Register, then jump the Soft Entry Address, and run NON_CPU0 boot code. Figure 3-9 shows the NON_CPU0 Boot Process:

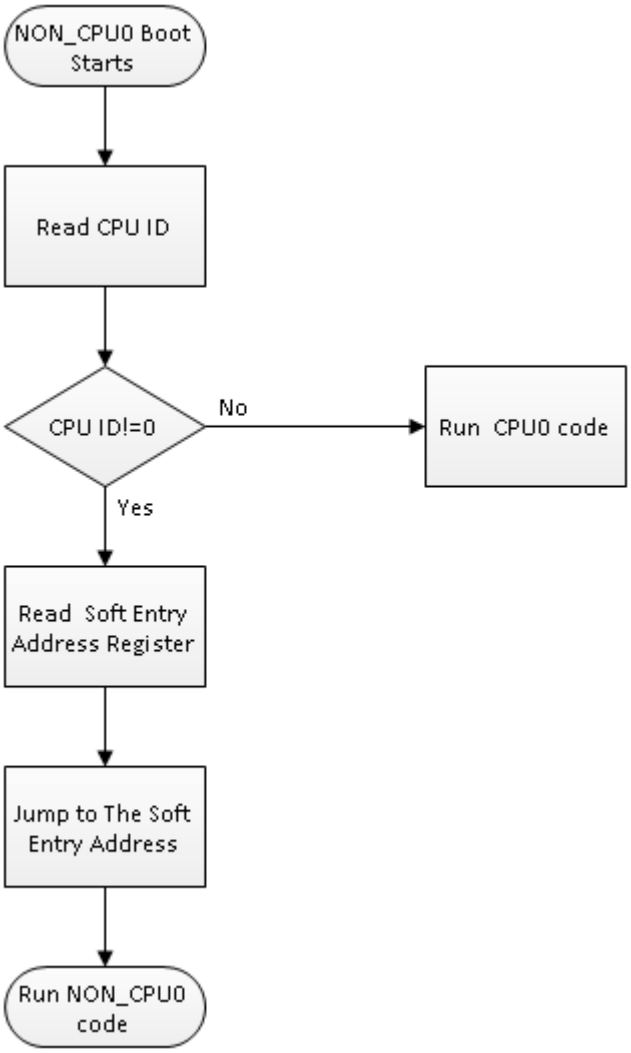


Figure 3-3. NON_CPU0 Boot Process Diagram

NOTE

Each processor has own the Soft Entry Address Register.

3.4.2.4 CPU0 Hotplug Process

The Hotplug Flag determines whether the system will do Hotplug boot, if CPU Hotplug Flag value is equal to 0xFA50392F, read the Soft Entry Register and the system will jump to the Soft Entry Address. Figure3-10 shows the CPU0 Hotplug Process:

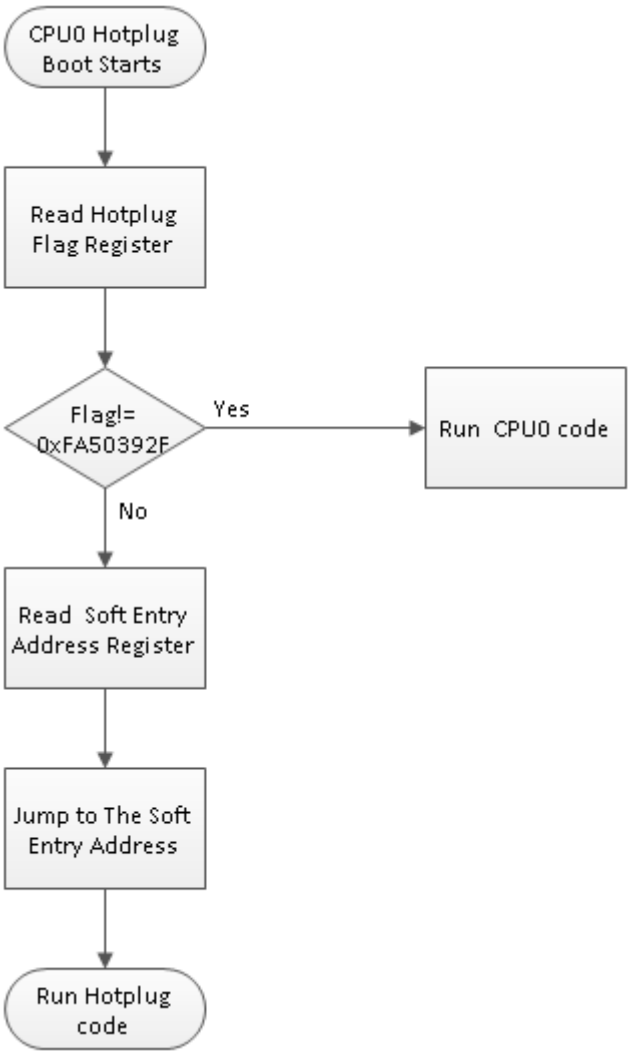


Figure 3-4. CPU0 Hotplug Process Diagram

NOTE

- **The Hotplug Flag Register is 0x070005C0.**
- **The Soft Entry Address Register is 0x070005C4.**

3.4.2.5 Super Standby Wakeup Process

Super Standby(SPSBY) wakeup will be started by CPUS, and will be carried on by CPU0 after the CPU0 released. If the SPSBY register value is checked to be the SPSBY flag, the system will go to SPSBY wakeup process. Figure 3-11 shows the SPSBY Wakeup Process.

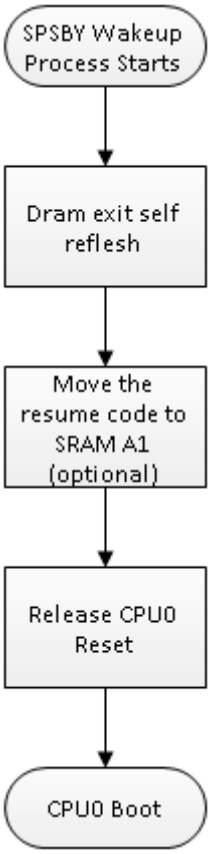


Figure 3-5. SPSBY Wakeup Process

During the SPSBY wakeup, the system will first check the SPSBY resume code pointed by SPSBY resume code pointer.If it is right, the system will run SPSBY wakeup, otherwise the system will jump to the Try Media Boot process. Figure 3-12 shows the SPSBY Resume Code Check Process.

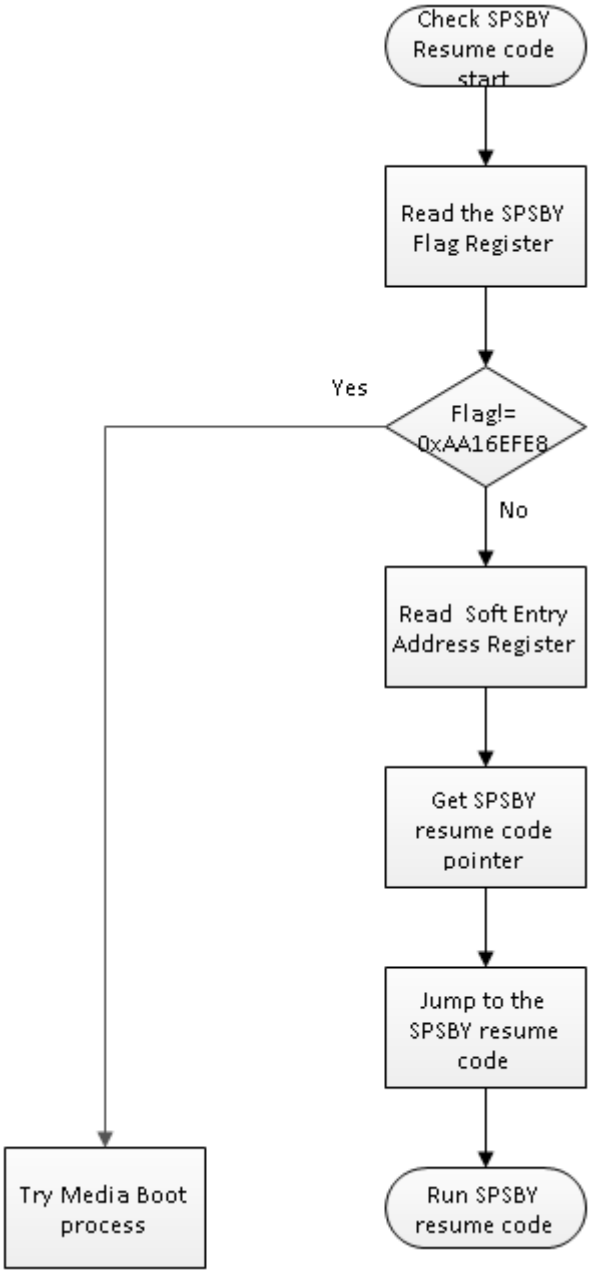


Figure 3-6. SPSBY Resume Code Check Process

3.4.2.6 Mandatory Upgrade Process

If the FEL Pin signal is detected to pull low, then the system will jump to mandatory upgrade process. Figure 3-13 shows the mandatory upgrade process.

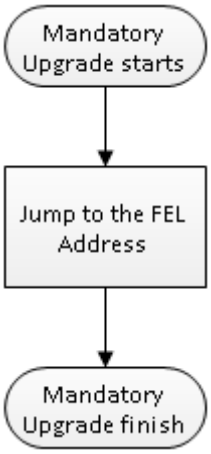


Figure 3-7. Mandatory Upgrade Process

NOTE

The FEL address of the Normal BROM is 0x20. The FEL address of the Secure BROM is 0x64.

3.4.2.7 FEL Process

When the system chooses to enter Mandatory Upgrade Process, and then jump to the FEL Process.Figure 3-14 shows the FEL process.

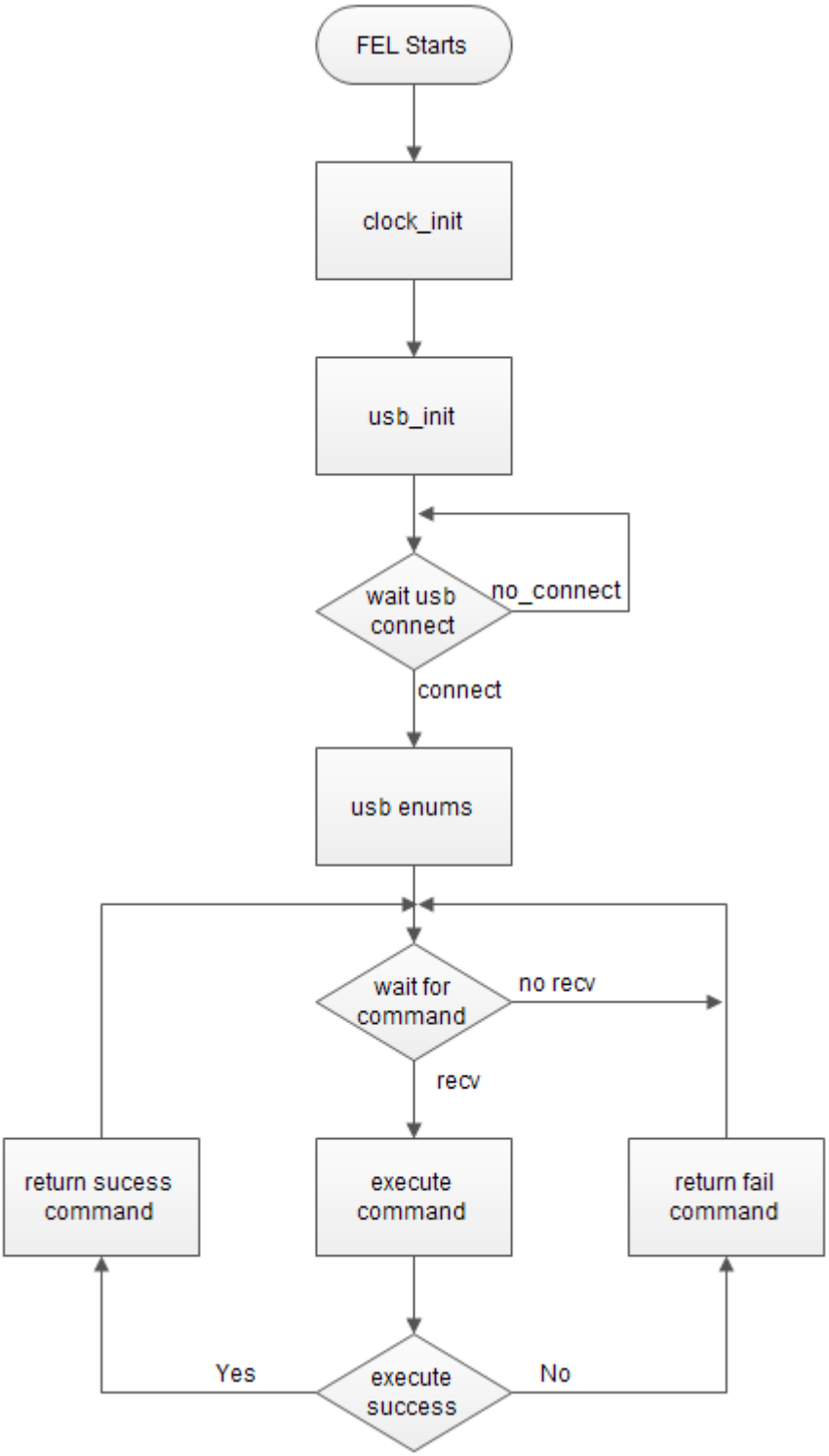


Figure 3-8. USB FEL Process

3.4.2.8 Boot Media Select

The BROM system supports the following boot media:

- SD/MMC
- NAND FLASH
- SPI NAND
- SPI NOR FLASH

There are two ways of Boot Select:GPIO Pin Select and eFuse Select. The BROM will read the state of BOOT_MODE ,

according to the state of BOOT_MODE to decide whether GPIO pin or eFuse to select the kind of boot media to boot. The BOOT_MODE is actually a bit in the SID. Table 3-4 shows BOOT_MODE Setting.

Table3- 4. BOOT_MODE Settin

BOOT_MODE[0]	Boot Select Type
0	GPIO Pin Select
1	eFuse Select

If the state of the BOOT_MODE is 0,that is to choose the GPIO Pin ,which has one pin to select which boot media to boot.Table 3-5 shows GPIO Pin Boot Select Setting.

Table3- 5. GPIO Pin Boot Select Setting

Pin_Boot_Select[0]	Boot media
0	SMHC0->NAND FLASH->SMHC2->SPI_NOR
1	SMHC0->SMHC2->NAND FLASH->SPI_NAND

If the state of the BOOT_MODE is 1, that is to choose the eFuse .eFuse select has 12 bits,so each of the 3 bits is divided into a group of the Boot Select ,so it has four groups of boot_select .Table 3-6 shows eFuse Boot Select Configure.

Table3- 6. eFuse Boot Select Configure

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

Table 3-7 describes each group of the eFuse Boot select settings. The first group to the third group are the same settings,but the fourth group need to be careful.If eFuse_Boot_Select_4 is set to 111,that means the way of the *Try*.The way of *Try* is followed by SMHC0,SMHC2,NAND FLASH,SPI NOR .

Table3- 7. eFuse Boot Select Setting

eFuse_Boot_Select_n	Boot media
000	Try
001	NAND Flash
010	SMHC2
011	SPI NOR
100	SPI NAND
101	Reserved
110	Reserved
111	The next group of the eFuse_Boot_Select,when the n is equal to 4,it will be a way of *Try*.

3.4.2.9 Normal Try Media Boot Process

If the FEL Pin signal is detected to pull to high level, then the system will jump to the Try Media Boot process.
Try Media Boot Process will read the state of BOOT_MODE register,according to the state of BOOT_MODE, GPIO pin or Efuse is descided to select which boot media to boot.Figure 3-15 shows Normal BROM GPIO Pin Boot Select Process.

NOTE

SMHC0 is external SD/TF card. SMHC2 is external eMMC.

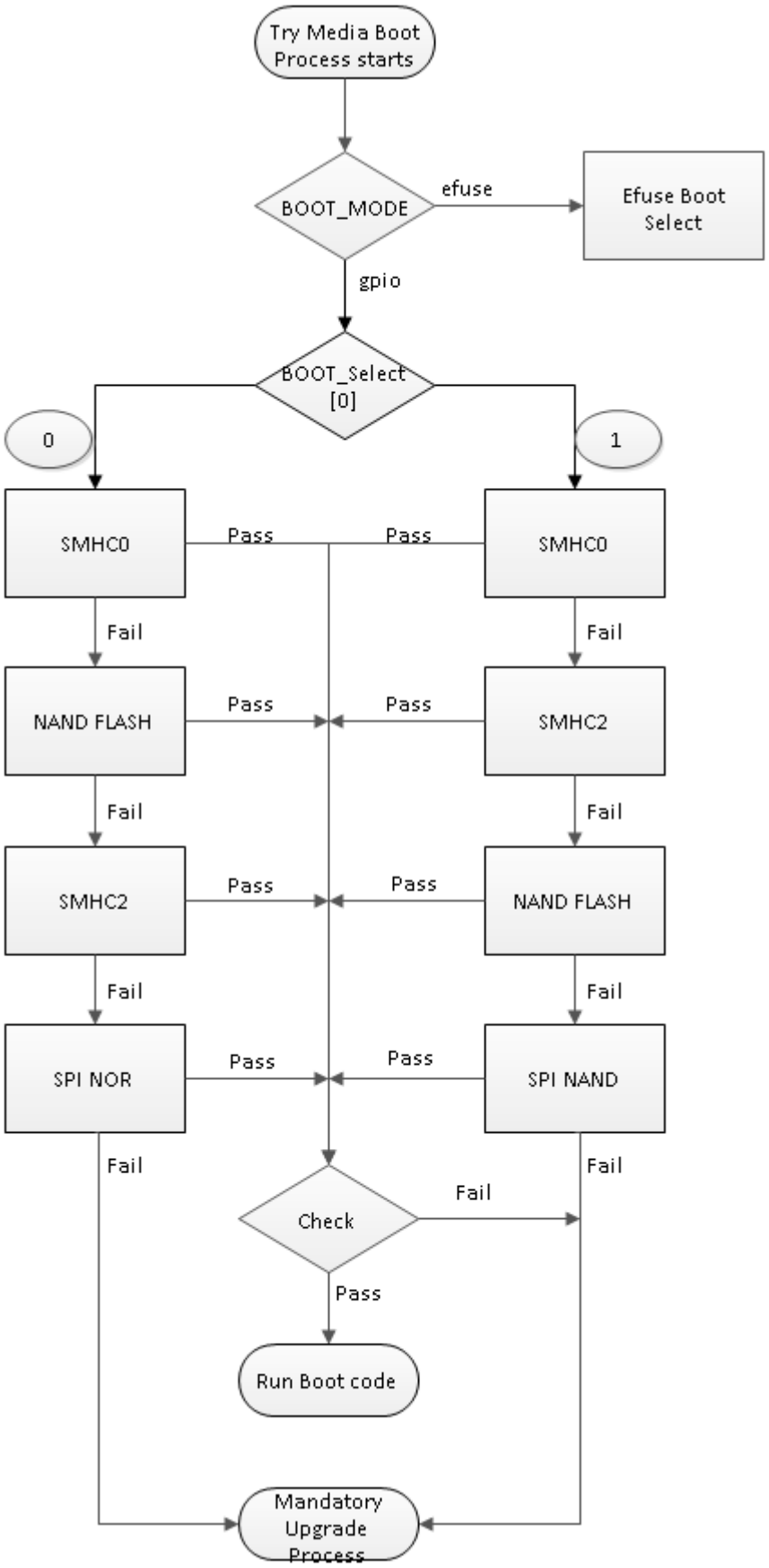


Figure 3-9. GPIO Pin Boot Select Process

Figure 3-16 shows Normal BROM eFuse Boot Select Process.

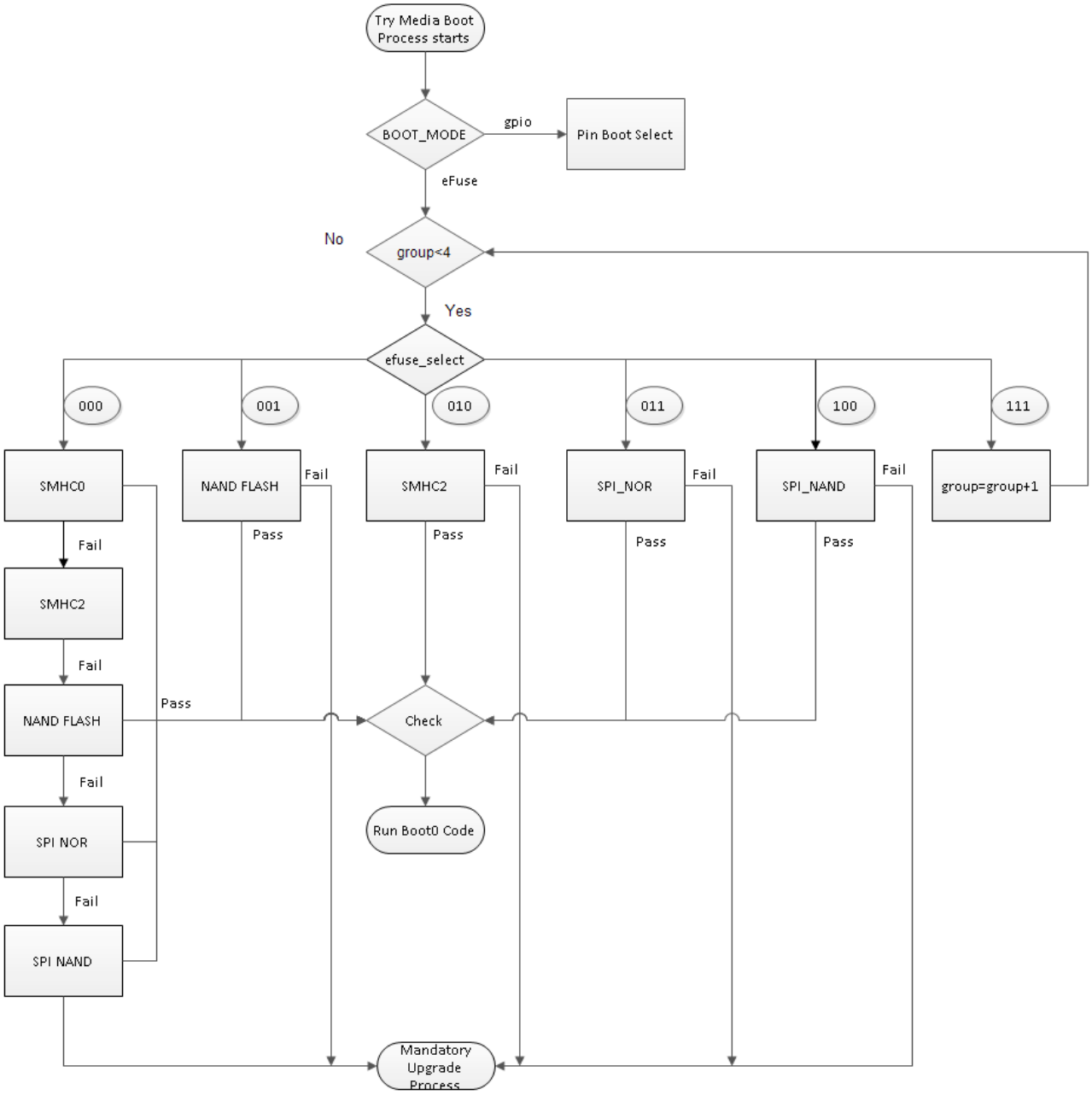


Figure 3-10. eFuse Boot Select Process

3.4.2.10 Secure Try Media Boot Process

Comparison with Normal Try Media Boot Process, the system will verify the integrity of the certificate. If the result is right, it will go to run Security BROM software, or it will go to the Mandatory Upgrade Process. Figure 3-17 shows Secure BROM GPIO Pin Boot Select Process.

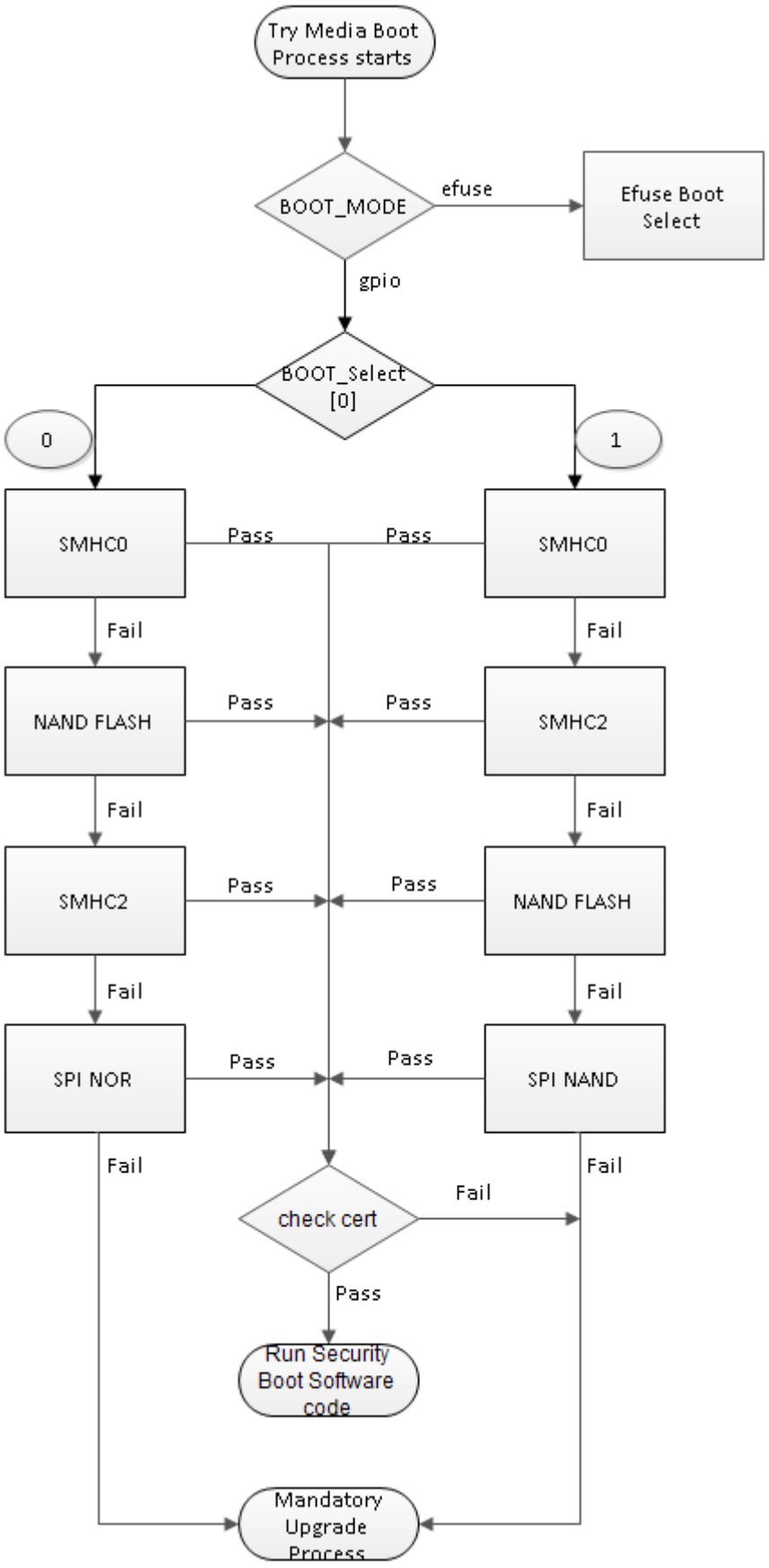


Figure 3-11. Secure BROM GPIO Pin Boot Select Process

Figure 3-18 shows Secure BROM eFuse Boot Select Process.

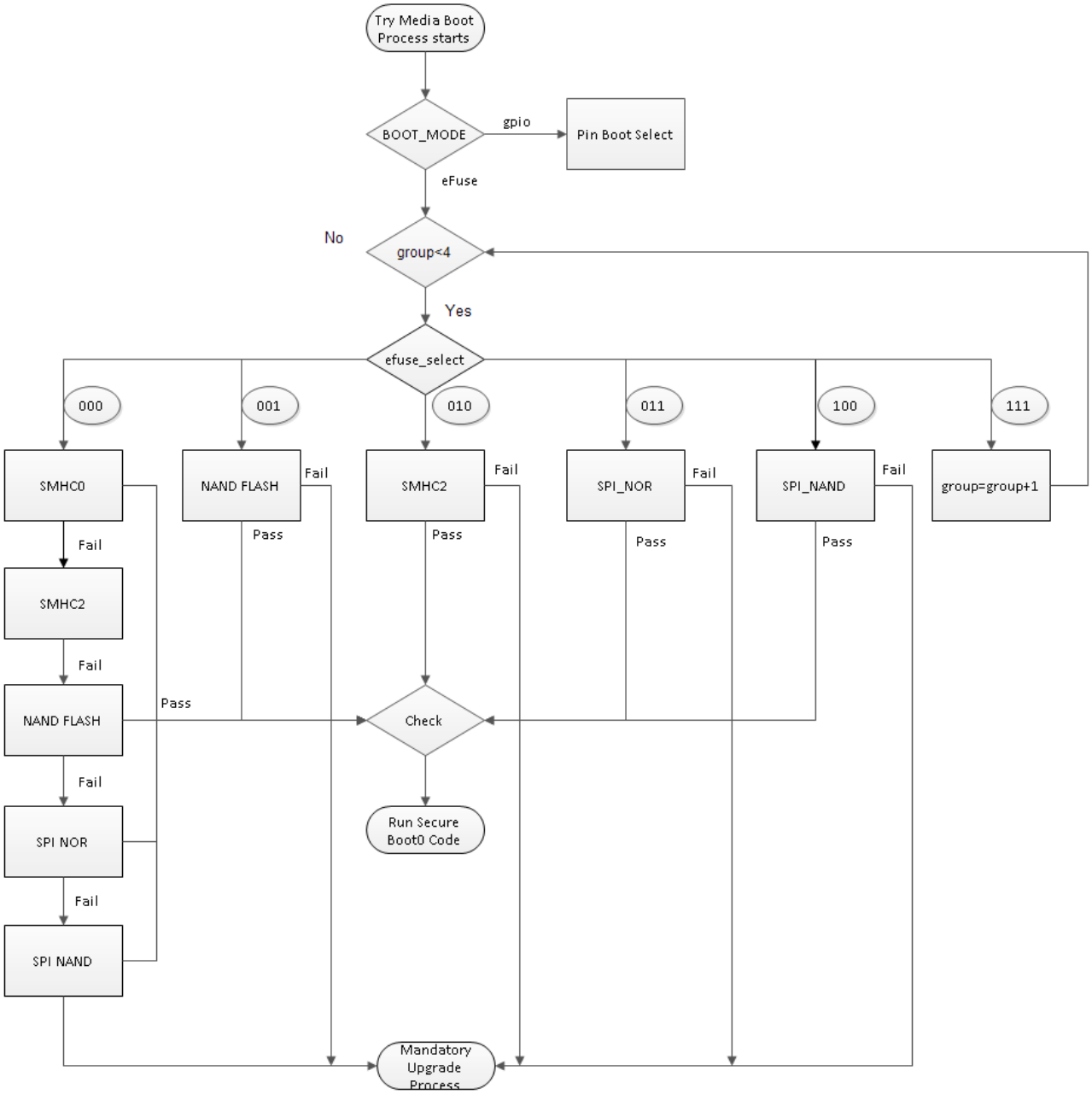


Figure 3-12. Secure BROM eFuse Boot Select Process

3.5 System Configuration

3.5.1 Overview

System configuration module is used to configure parameter for system domain, such as SRAM, CPU, PLL, BROM and so on. This module has the following features:

- SRAM Bist function
- System parameter configuration
- PLL back door configuration
- BROM debug parameter configuration

The address range of SRAM is as follows.

Area	Address	Size(Bytes)
SRAM A1	0x0002 0000---0x0003 7FFF	96K
SRAM C	0x0003 8000---0x0005 8FFF	132K
SRAM A2	0x0010 0000---0x0010 1FFF	8K
	0x0010 4000---0x0011 BFFF	96K

3.5.2 Register List

Module Name	Base Address
SYS_CFG	0x0300 0000

Register Name	Offset	Description
VER_REG	0x0024	Version Register
BROM_OUTPUT_REG	0x00A4	BROM Output Register

3.5.3 Register Description

3.5.3.1 Version Register

Offset:0x0024			Register Name: VER_REG
Bit	Read/Write	Default/Hex	Description
31:10	/	/	/
9	R	UDF	BOOT_SEL_PAD_STA 0:SMHC0->SMHC2->NAND FLASH->SPI_NOR 1:SMHC0->NAND FLASH->SMHC2->SPI_NOR
8	R	UDF	FEL_SEL_PAD_STA Fel_Select_Pin_Status 0: Run_FEL 1:Normal Boot

7:3	/	/	/
2:0	R	0x0	Reserved

3.5.3.2 BROM Output Register (Default Value: 0x0000_0000)

Offset:0x00A4			Register Name: BROM_OUTPUT_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
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_ENALBE. 0: Disable U-Boot pin output 1: Enable U-Boot pin output

3.6 IOMMU

3.6.1 Overview

IOMMU(I/O Memory management unit) is designed for product specific memory requirements. It maps the virtual address(sent by peripheral access memory)to the physical address. IOMMU allows multiple ways to manage the location of physical address, and it can use physical address which has potentially conflict mapping for different processes to allocate memory space, and also allow application of non-continuous address mapping to continuous virtual address space.

Features:

- Supports virtual address to physical address mapping by hardware implementation
- Supports DE、VE、CSI、G2D parallel address mapping
- Supports DE、VE、CSI、G2D bypass function independently
- Supports DE、VE、CSI、G2D prefetch independently
- Supports DE、VE、CSI、G2D interrupt handing mechanism independently
- Supports level1 and level2 TLB for special using, and level2 TLB for sharing
- Supports TLB Fully cleared and Partially disabled
- Supports trigger PTW behavior when TLB miss
- Supports checking the permission

3.6.2 Block Diagram

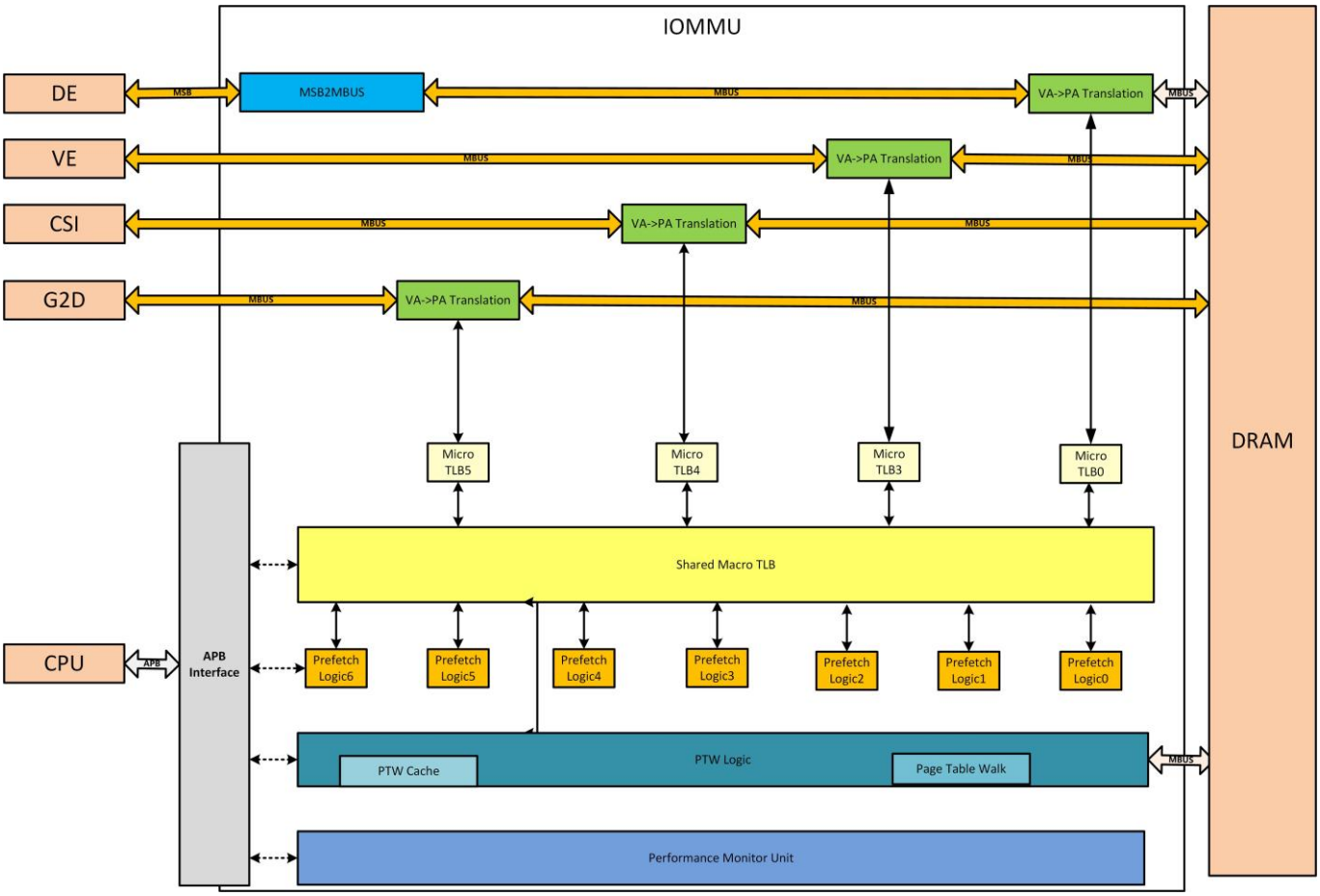


Figure3- 7. IOMMU Block Diagram

IOMMU internal module mainly has the following parts:

Micro TLB: level1 TLB, 64 words. Each peripheral correspondings to a TLB, Caching the level2 page table for the peripheral.

Macro TLB: level2 TLB, 4K words. Each peripheral shares a level2 TLB for caching the level2 page table.

Prefetch Logic: Each Micro TLB corresponds to a Prefetch Logic.

PTW Logic: Page Table Walk. mainly contains PTW Cache and PTW. The PTW Cache is used to store the level1 page table; When the virtual address VA missed in the level1 and level2 TLB, it will trigger the PTW. PTW Cache can store 512 level1 page tables, that is, 512-word

PMU: Performance Monitoring Unit, which is used to count hit efficiency and latency.

APB Interface: IOMMU register instantiation module. CPU reads and writes the IOMMU register by APB bus.

Master relationship is as follows.

Master0: DE

Master3: VE

Master4: CSI

Master5: G2D

3.6.3 Operations and Functional Descriptions

3.6.3.1 Clock Sources

IOMMU contains two clock domains in the module. Address mapping generated by MBUS clock domain, and Register and interrupt processing generated by APB clock domain. The two domains are asynchronous, and they are from different clock sources.

3.6.3.2 Operation Modes

3.6.3.2.1 Initialization

- Release the IOMMU reset signal by writing 1 to the IOMMU Reset Register bit[31];
- Write the base address of the first TLB to the IOMMU Translation Table Base Register;
- Set up the IOMMU Interrupt Enable Register;
- Enable the IOMMU by configuring the IOMMU Enable Register in the final.

3.6.3.2.2 Address Operates

In the process of address mapping, The peripheral virtual address VA[31:12] are retrieved in the Level1 TLB, when TLB hit, the mapping finished, or they are retrieved in the level2 TLB in the same way. If TLB hit, it will write the mapping to the Level1 TLB, and this shows Level1 TLB hit. If Level1 and level2 TLB retrieved miss, it will trigger the PTW. After opening peripheral bypass function, the corresponding register is IOMMU Bypass Register, IOMMU will not map the address for peripheral typed the address, and it will output the virtual address as physical address.The typical application is as follows.

- **Micro TLB hit**

a). The master device sends a transfer command, and also sends the address to the corresponding Micro TLB, and searches virtual address corresponding to the level2 page table;

b). If Micro TLB hit, it will return a corresponding physical addresses and the level2 page table of permission Index;

c). Address transform module converts the virtual address into physical address, and checks the permissions at the same time. If pass, transfer is completed.

- **Micro TLB miss, Macro TLB hit**

a). The master device send a transfer command, and also send the address to the corresponding Micro TLB, and searches virtual address corresponding to the level 2 page table;

b). If Micro TLB misses, then continue to search Macro TLB;

c). If Macro TLB misses, it will return the level2 page table to Micro TLB;

d). Micro receives the page table, and put it to Micro TLB(if this Micro TLB is full, there has replace activities), at the same time send page table entries to address translation module;

e). Address transform module converts the virtual address into physical address, and checks the permissions at the same time. If pass, transfer is completed.

- **Micro TLB miss, Micro TLB miss, PTW Cache hit**

a). The master device sends a transfer command, and also sends the address to the corresponding Micro TLB, and

searches virtual address corresponding to the level 2 page table;

b). If Micro TLB misses, then continue to search Macro TLB;

c). If Macro TLB misses, then it will send the request to the PTW to return the corresponding page table;

d). PTW first accesses PTW Cache, confirms the required level 1 page table to exist in the PTW Cache, sends the page table to PTW logic;

e). PTW logic returns the corresponding level2 page table from memory page table according to Level1 page table, checks the effectiveness, and sends to Macro TLB;

f). Macro TLB stores the level 2 page table (may happen replace activities), and will return the level 2 page table to Micro TLB;

g). Micro TLB receives the page table entries, puts in the Micro TLB (if this Micro TLB is full, there will happen replace activities), and sends page table entries to address translation module;

h). Address transform module converts the virtual address into physical address, and checks the permissions at the same time. If pass, transfer is completed.

- **Micro TLB miss, Micro TLB miss, PTW Cache miss**

a). The master device sends a transfer command, and also sends the address to the corresponding Micro TLB, and searches virtual address corresponding to the level2 page table;

b). If Micro TLB misses, then continue to search Macro TLB;

c). If Macro TLB misses, there will send the request to the PTW to return the corresponding page table;

d). PTW accesses PTW Cache, there is no needing Level1 page table;

e). PTW accesses memory, gets the corresponding Level1 page table and stores in the PTW Cache;

f). PTW logic returns the corresponding level2 page table from memory page table according to Level1 page table, checks the effectiveness, and sends to Macro TLB;

g). Macro TLB stores the level2 page table (may happen replace activities), and will return the level 2 page table to Micro TLB;

h). Micro TLB receives the page table entries, puts in the Micro TLB (if this Micro TLB is full, there will happen replace activities), and sends page table entries to address translation module;

i). Address transform module converts the virtual address into physical address, and checks the permissions at the same time. If pass, transfer is completed.

- **Permission error**

a). Permission check is always performs in the address conversion;

b). Once the permission check makes mistake, new access of the master suspends, before this visit continues;

c). Set the error status register;

d). Trigger interrupt.

- **Invalid Level1 page table**

a). Invalid Level1 page table checks when PTW logic reads the new level page table from memory;

b). The PTW read sequentially two page table entries from the memory (64-bit data, a complete cache line), and stores in the PTW cache;

c). If the current page table is detected invalid, then the error flag is set, and the interrupt is triggered, the cache line need to be invalidated.

NOTE

Invalid page table has two situations: the reading target page table from the memory is invalid; and the page table stored in PTW Cache with target page table is found to be invalid after using;

If a page table is invalid,then total cache line need to be invalidated, that is two page tables.

- **Invalid level2 page table**

a). Invalid level2 page table checks when Macro TLB reads the new level page table from memory;

b). The Macro TLB read sequentially two page table entries from the memory (64-bit data, a complete cache line), and stores in the Macro TLB;

c). If the current page table is detected invalid, then the error flag is set, and the interrupt is triggered, the cache line need to be invalidated.

NOTE

Invalid page table has two situations: the reading target page table from the memory is invalid; and the page table stored in Macro TLB with target page table is found to be invalid after using; if a page table is invalid,then total cache line need to be invalidated, that is two page tables.

Internal address switch process shows in Figure 3-8.

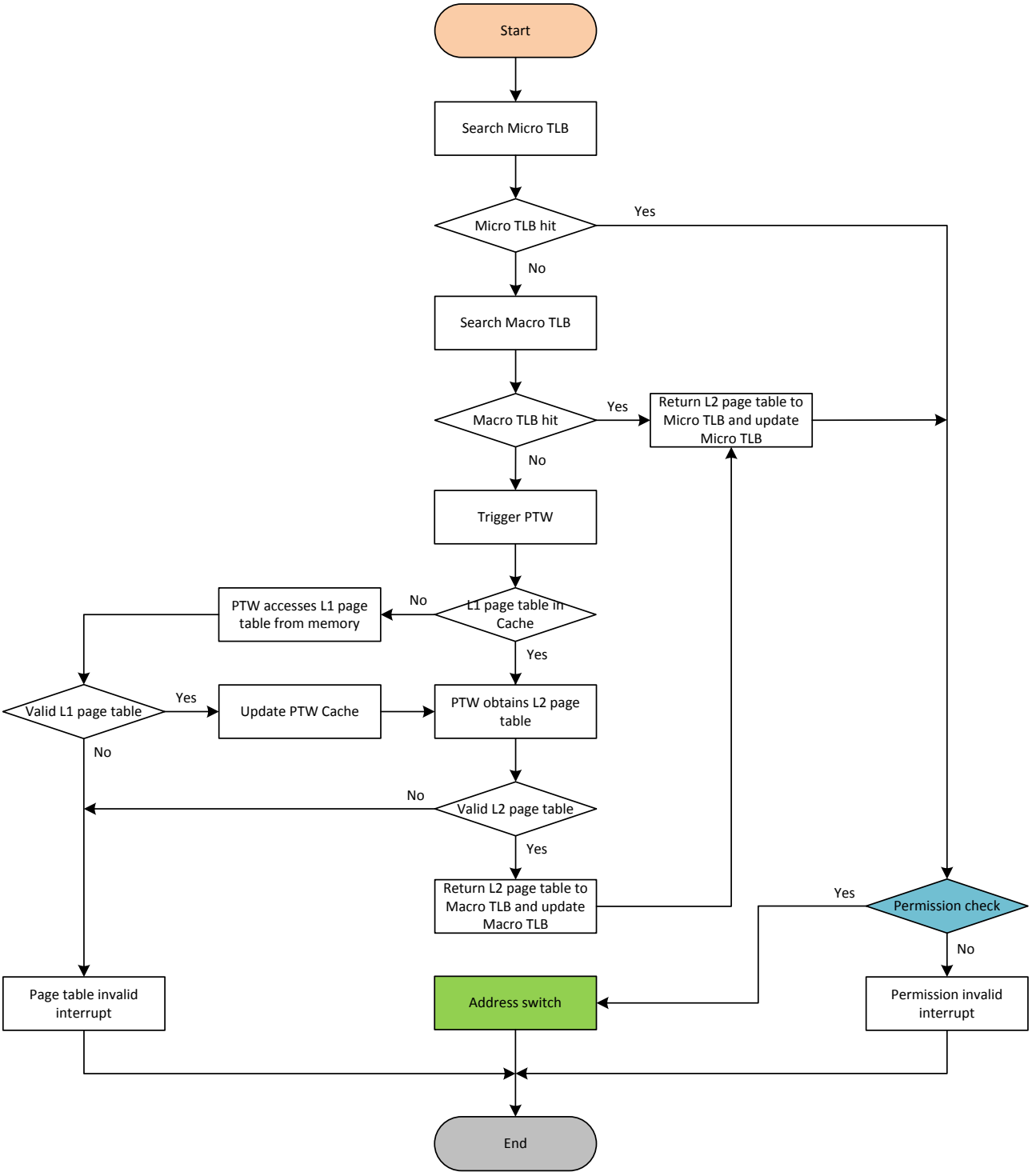


Figure3- 8. Internal Switch Process

3.6.3.2.3 VA-PA Mapping

IOMMU page table is defined as Level2 mapping, the first level is 1M address space mapping, the second level is 4K address space. This version does not support 1K, 16K and other page table size. IOMMU supports a page table only, its meaning is:

All peripherals connected to IOMMU use the same virtual address space;

The virtual address space of the peripherals can overlap;

Different virtual addresses can map to the same physical address space;

Base address of the page table is defined by software, and need 16 KByte address alignment; Page table of the Level2 table item need 1 KByte address alignment. A complete VA-PA address translation process is shown in Figure 3-21.

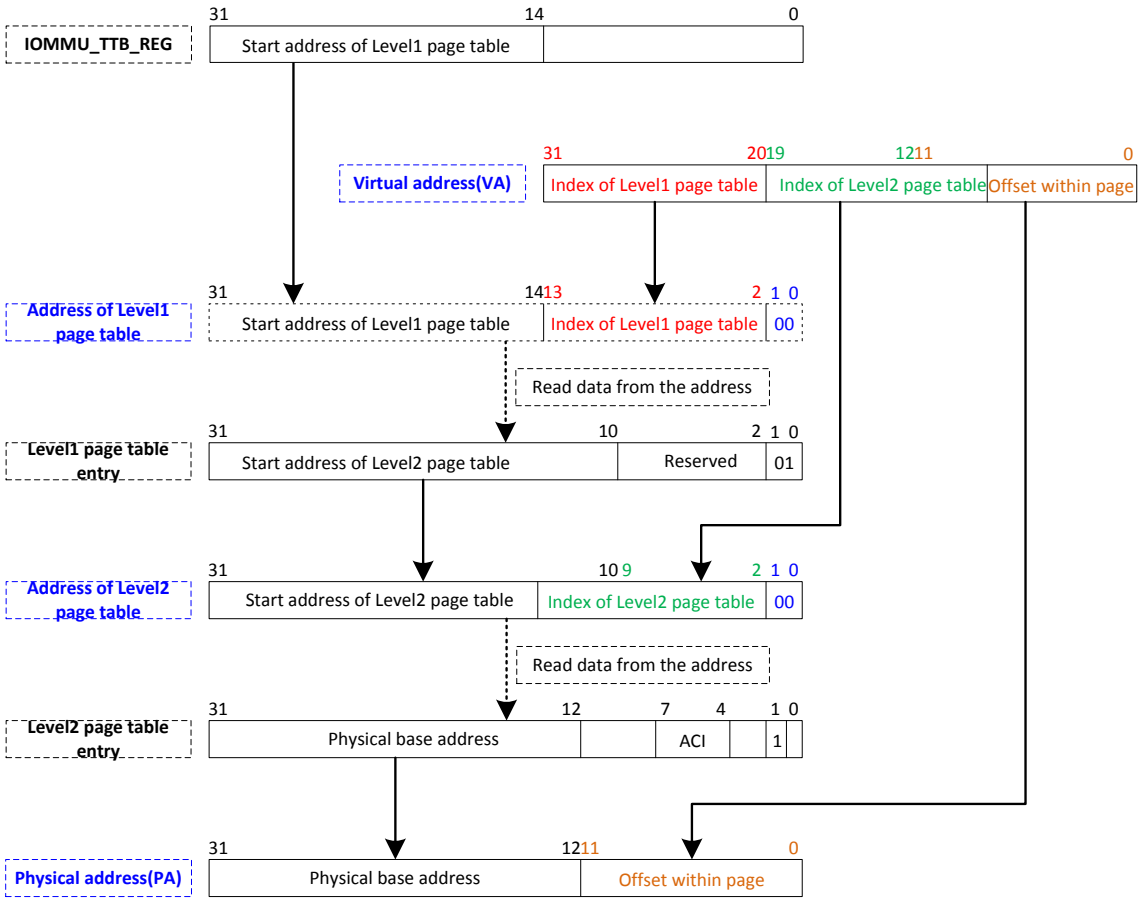


Figure3- 9. VA-PA Switch Process

3.6.3.2.4 Clear and Invalidate TLB

When more page table content refresh or table address changes, all VA-PA mapping which has been cached in TLB will no longer be valid , then you need configure **IOMMU TLB Flush Enable Register** to clear the TLB or PTW Cache. First suspend the TLB or Cache access, then configure the corresponding Flush bit of **IOMMU TLB Flush Enable Register** , after operation takes effect, related peripherals can continue to send new access memory operations.

When some page table is invalid or incorrect mapping, you can set the TLB Invalidation relevant register to invalidate TLB VA-PA mapping pairs. First, write target address to **IOMMU TLB Invalidation Address Register**;then, set configuration values to **IOMMU TLB Invalidation Address Mask Register** , the requirements are as follows:

The value of **IOMMU TLB Invalidation Address Mask Register** cannot be less than the **IOMMU TLB Invalidation Address Register**.

The higher bit of **IOMMU TLB Invalidation Address Mask Register** must be continuous 1, the lower bit must be continuous 0, for example ,0xfffff000, 0xffffe000, 0xffffc000, 0xffff8000, 0xffff0000 belongs to the legal value; and 0xffffd000, 0xffffb000, 0xffffa000, 0xffff9000, 0xffff7000 belongs to illegal values.

Finally, Configure **IOMMU TLB Invalidation Enable Register** to enable invalid operation. Among the method of invalid address is that target address AND mask address gets maximum valid bit and determines destination address range.The figure is as follows.

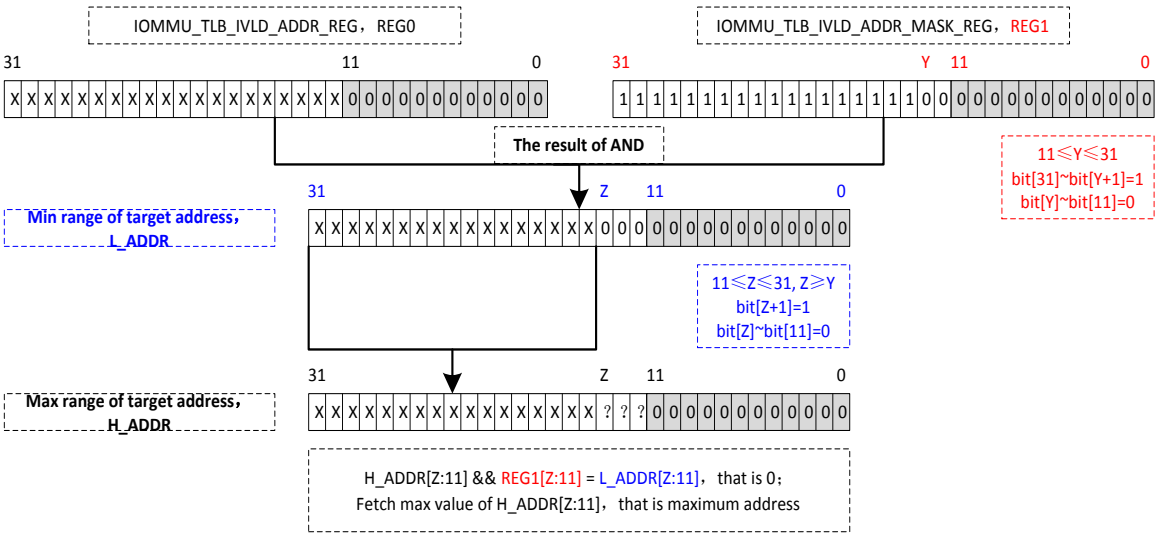


Figure3- 10. Invalid TLB Address Range

For example:

a). When the value of **IOMMU TLB Invalidation Address Mask Register** is 0xFFFFF000 by default,the result of AND is target address,that is ,invalid target address.

b). When the value of **IOMMU TLB Invalidation Address Mask Register** is 0xFFFF0000, the value of **IOMMU TLB Invalidation Address Register** is 0xEEEE1000, then target address range is from 0xEEEE0000 to 0xEEEEF000.

c). When the value of **IOMMU TLB Invalidation Address Mask Register** is 0xFFFFC000, the value of **IOMMU TLB Invalidation Address Register** is 0xEEEE8000, then target address range is from 0xEEEE8000 to 0xEEEEB000.

d). When the value of **IOMMU TLB Invalidation Address Mask Register** is 0xFFFF8000, the value of **IOMMU TLB Invalidation Address Register** is 0xEEEEC000, then target address range is from 0xEEEE8000 to 0xEEEEF000.

e). When the value of **IOMMU TLB Invalidation Address Mask Register** is 0xFFFFC000, the value of **IOMMU TLB Invalidation Address Register** is 0xEEEE0000, then target address range is from 0xEEEE0000 to 0xEEEE3000.

3.6.3.3 Page Table Format

3.6.3.3.1 Level1 Page Table

The format of Level1 page table is as follows.

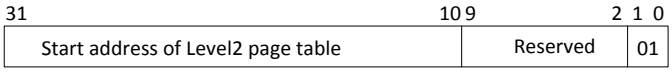


Figure3- 11. Level1 Page Table Format

Bit[31:10]: Base address of Level2 page table;

Bit[9:2]: Reserved;

Bit[1:0]: 01 is valid page table;others are fault;

3.6.3.3.2 Level2 Page Table

The format of Level2 page table is as follows.

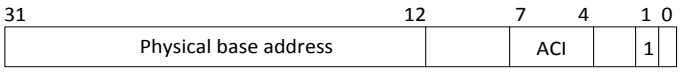


Figure3- 12. Level1 Page Table Format

Bit[31:12]: Physical address of 4K address;

Bit[11:8]: Reserved;

Bit[7:4]: ACI，permission control index; correspond to permission control bit of **IOMMU Domain Authority Control Register**;

Bit[3:2]: Reserved;

Bit[1]: 1 is valid page table; 0 is fault;

Bit[0]: Reserved

3.6.3.3.3 Permission Index

The read/write access control of series register such as **IOMMU Domain Authority Control Register** is as follows.

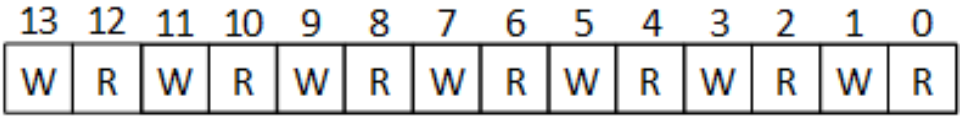


Figure3- 13. Read/Write Permission Control

Bit[1:0]/Bit[17:16]: Master0 read/write permission control;

Bit[3:2]/Bit[19:18]: Master1 read/write permission control;

Bit[5:4]/Bit[21:20]: Master2 read/write permission control;

Bit[7:6]/Bit[23:22]: Master3 read/write permission control;

Bit[9:8]/Bit[25:24]: Master4 read/write permission control;

Bit[11:10]/Bit[27:26]: Master5 read/write permission control;

Bit[13:12]/Bit[29:28]: Master6 read/write permissiom control.

The value of **IOMMU Domain Authority Control Register** is read-only by default. Other registers can configure through system requirement. In address switch process, the corresponding relation between ACI and Domain is as follows.

Table3- 8. Relation between ACI and Domain

ACI	Domain	Register
0	Domain 0	IOMMU Domain Authority Control Register 0
1	Domain 1	IOMMU Domain Authority Control Register 0
2	Domain 2	IOMMU Domain Authority Control Register 1
3	Domain 3	IOMMU Domain Authority Control Register 1
4	Domain 4	IOMMU Domain Authority Control Register 2
5	Domain 5	IOMMU Domain Authority Control Register 2
6	Domain 6	IOMMU Domain Authority Control Register 3
7	Domain 7	IOMMU Domain Authority Control Register 3
8	Domain 8	IOMMU Domain Authority Control Register 4
9	Domain 9	IOMMU Domain Authority Control Register 4
10	Domain 10	IOMMU Domain Authority Control Register 5
11	Domain 11	IOMMU Domain Authority Control Register 5
12	Domain 12	IOMMU Domain Authority Control Register 6

13	Domain 13	IOMMU Domain Authority Control Register 6
14	Domain 14	IOMMU Domain Authority Control Register 7
15	Domain 15	IOMMU Domain Authority Control Register7

After enabled **IOMMU Domain Authority Overwrite Register**, the read/write control permission can override all **IOMMU Domain Authority Control Register**.

3.6.4 Programming Guidelines

3.6.4.1 IOMMU Reset

In order to shield the influence of IOMMU reset , be sure IOMMU is never opened, or no unfinished bus operation, or DRAM and peripherals already open the corresponding switch before the IOMMU module software reset operation.

3.6.4.2 IOMMU Enable

Before opening the IOMMU address mapping function, Translation Table Base register should be correctly configured, or all the masters are in the bypass state, or all the masters do not send the bus command.

3.6.4.3 Configure TTB

Operating the register must close IOMMU address mapping function, namely IOMMU_ENABLE_REG [0] is 0; or Bypass function of all masters is set to 1, or no the state of transfer bus commands.

3.6.4.4 Clear TTB

In the Flush operation, all TLB/Cache access will be suspended; but the operation entered the TLB will continue to complete before the Flush starts.

3.6.4.5 Read/Write VA Data

For virtual address, read and write the corresponding physical address data, be sure IOMMU module address mapping function is normal or not. First, make sure to read or write, and then configure the target virtual address or write data, then start to read or write function, after being finished, check if the results are as expected.

3.6.4.6 PMU Statistics

When PMU function is used for the first time, **Set IOMMU PMU Enable Register** can be enabled; when reading the relevant Register, **Clear IOMMU PMU Enable Register** need be enabled; when PMU function is used next time, **first IOMMU PMU Clear Register** is set, after counter is cleared, **Set IOMMU PMU Enable Register** can be enabled. Given a level2 page table administers continuous 4 KB address, if Micro TLB misses in continuous virtual address, there may

need to return a level2 page table to hit from Macro TLB; but the hit number is not recorded in the Macro TLB hit and Micro TLB hit related register. So true hit rate calculation is as follows:

Hit Rate = N1/M1 + (1-N1/M1)*N2/M2

N1: Micro TLB hit number

M1: Micro TLB access number

N2: Macro TLB hit number

M2: Macro TLB access number

3.6.5 Register List

Module Name	Base Address
IOMMU	0x030F0000

Register Name	Offset	Description
IOMMU_RESET_REG	0x0010	IOMMU Reset Register
IOMMU_ENABLE_REG	0x0020	IOMMU Enable Register
IOMMU_BYPASS_REG	0x0030	IOMMU Bypass Register
IOMMU_AUTO_GATING_REG	0x0040	IOMMU Auto Gating Register
IOMMU_WBUF_CTRL_REG	0x0044	IOMMU Write Buffer Control Register
IOMMU_OOO_CTRL_REG	0x0048	IOMMU Out Of Order Control Register
IOMMU_4KB_BDY_PRT_CTRL_REG	0x004C	IOMMU 4KB Boundary Protect Control Register
IOMMU_TTB_REG	0x0050	IOMMU Translation Table Base Register
IOMMU_TLB_ENABLE_REG	0x0060	IOMMU TLB Enable Register
IOMMU_TLB_PREFETCH_REG	0x0070	IOMMU TLB Prefetch Register
IOMMU_TLB_FLUSH_ENABLE_REG	0x0080	IOMMU TLB Flush Enable Register
IOMMU_TLB_IVLD_ADDR_REG	0x0090	IOMMU TLB Invalidation Address Register
IOMMU_TLB_IVLD_ADDR_MASK_REG	0x0094	IOMMU TLB Invalidation Address Mask Register
IOMMU_TLB_IVLD_ENABLE_REG	0x0098	IOMMU TLB Invalidation Enable Register
IOMMU_PC_IVLD_ADDR_REG	0x00A0	IOMMU PC Invalidation Address Register
IOMMU_PC_IVLD_ENABLE_REG	0x00A8	IOMMU PC Invalidation Enable Register
IOMMU_DM_AUT_CTRL_REG0	0x00B0	IOMMU Domain Authority Control Register 0
IOMMU_DM_AUT_CTRL_REG1	0x00B4	IOMMU Domain Authority Control Register 1
IOMMU_DM_AUT_CTRL_REG2	0x00B8	IOMMU Domain Authority Control Register 2
IOMMU_DM_AUT_CTRL_REG3	0x00BC	IOMMU Domain Authority Control Register 3
IOMMU_DM_AUT_CTRL_REG4	0x00C0	IOMMU Domain Authority Control Register 4
IOMMU_DM_AUT_CTRL_REG5	0x00C4	IOMMU Domain Authority Control Register 5
IOMMU_DM_AUT_CTRL_REG6	0x00C8	IOMMU Domain Authority Control Register 6
IOMMU_DM_AUT_CTRL_REG7	0x00CC	IOMMU Domain Authority Control Register 7
IOMMU_DM_AUT_OVWT_REG	0x00D0	IOMMU Domain Authority Overwrite Register
IOMMU_INT_ENABLE _REG	0x0100	IOMMU Interrupt Enable Register
IOMMU_INT_CLR _REG	0x0104	IOMMU Interrupt Clear Register
IOMMU_INT_STA _REG	0x0108	IOMMU Interrupt Status Register
IOMMU_INT_ERR_ADDR_REG0	0x0110	IOMMU Interrupt Error Address Register 0

IOMMU_INT_ERR_ADDR_REG1	0x0114	IOMMU Interrupt Error Address Register 1
IOMMU_INT_ERR_ADDR_REG2	0x0118	IOMMU Interrupt Error Address Register 2
IOMMU_INT_ERR_ADDR_REG3	0x011C	IOMMU Interrupt Error Address Register 3
IOMMU_INT_ERR_ADDR_REG4	0x0120	IOMMU Interrupt Error Address Register 4
IOMMU_INT_ERR_ADDR_REG5	0x0124	IOMMU Interrupt Error Address Register 5
IOMMU_INT_ERR_ADDR_REG6	0x0128	IOMMU Interrupt Error Address Register 6
IOMMU_INT_ERR_ADDR_REG7	0x0130	IOMMU Interrupt Error Address Register 7
IOMMU_INT_ERR_ADDR_REG8	0x0134	IOMMU Interrupt Error Address Register 8
IOMMU_INT_ERR_DATA_REG0	0x0150	IOMMU Interrupt Error Data Register 0
IOMMU_INT_ERR_DATA_REG1	0x0154	IOMMU Interrupt Error Data Register 1
IOMMU_INT_ERR_DATA_REG2	0x0158	IOMMU Interrupt Error Data Register 2
IOMMU_INT_ERR_DATA_REG3	0x015C	IOMMU Interrupt Error Data Register 3
IOMMU_INT_ERR_DATA_REG4	0x0160	IOMMU Interrupt Error Data Register 4
IOMMU_INT_ERR_DATA_REG5	0x0164	IOMMU Interrupt Error Data Register 5
IOMMU_INT_ERR_DATA_REG6	0x0168	IOMMU Interrupt Error Data Register 6
IOMMU_INT_ERR_DATA_REG7	0x0170	IOMMU Interrupt Error Data Register 7
IOMMU_INT_ERR_DATA_REG8	0x0174	IOMMU Interrupt Error Data Register 8
IOMMU_L1PG_INT _REG	0x0180	IOMMU L1 Page Table Interrupt Register
IOMMU_L2PG_INT _REG	0x0184	IOMMU L2 Page Table Interrupt Register
IOMMU_VA_REG	0x0190	IOMMU Virtual Address Register
IOMMU_VA_DATA_REG	0x0194	IOMMU Virtual Address Data Register
IOMMU_VA_CONFIG_REG	0x0198	IOMMU Virtual Address Configuration Register
IOMMU_PMU_ENABLE _REG	0x0200	IOMMU PMU Enable Register
IOMMU_PMU_CLR _REG	0x0210	IOMMU PMU Clear Register
IOMMU_PMU_ACCESS_LOW_REG0	0x0230	IOMMU PMU Access Low Register 0
IOMMU_PMU_ACCESS_HIGH_REG0	0x0234	IOMMU PMU Access High Register 0
IOMMU_PMU_HIT_LOW_REG0	0x0238	IOMMU PMU Hit Low Register 0
IOMMU_PMU_HIT_HIGH_REG0	0x023C	IOMMU PMU Hit High Register 0
IOMMU_PMU_ACCESS_LOW_REG1	0x0240	IOMMU PMU Access Low Register 1
IOMMU_PMU_ACCESS_HIGH_REG1	0x0244	IOMMU PMU Access High Register 1
IOMMU_PMU_HIT_LOW_REG1	0x0248	IOMMU PMU Hit Low Register 1
IOMMU_PMU_HIT_HIGH_REG1	0x024C	IOMMU PMU Hit High Register 1
IOMMU_PMU_ACCESS_LOW_REG2	0x0250	IOMMU PMU Access Low Register 2
IOMMU_PMU_ACCESS_HIGH_REG2	0x0254	IOMMU PMU Access High Register 2
IOMMU_PMU_HIT_LOW_REG2	0x0258	IOMMU PMU Hit Low Register 2
IOMMU_PMU_HIT_HIGH_REG2	0x025C	IOMMU PMU Hit High Register 2
IOMMU_PMU_ACCESS_LOW_REG3	0x0260	IOMMU PMU Access Low Register 3
IOMMU_PMU_ACCESS_HIGH_REG3	0x0264	IOMMU PMU Access High Register 3
IOMMU_PMU_HIT_LOW_REG3	0x0268	IOMMU PMU Hit Low Register 3
IOMMU_PMU_HIT_HIGH_REG3	0x026C	IOMMU PMU Hit High Register 3
IOMMU_PMU_ACCESS_LOW_REG4	0x0270	IOMMU PMU Access Low Register 4
IOMMU_PMU_ACCESS_HIGH_REG4	0x0274	IOMMU PMU Access High Register 4
IOMMU_PMU_HIT_LOW_REG4	0x0278	IOMMU PMU Hit Low Register 4
IOMMU_PMU_HIT_HIGH_REG4	0x027C	IOMMU PMU Hit High Register 4
IOMMU_PMU_ACCESS_LOW_REG5	0x0280	IOMMU PMU Access Low Register 5

IOMMU_PMU_ACCESS_HIGH_REG5	0x0284	IOMMU PMU Access High Register 5
IOMMU_PMU_HIT_LOW_REG5	0x0288	IOMMU PMU Hit Low Register 5
IOMMU_PMU_HIT_HIGH_REG5	0x028C	IOMMU PMU Hit High Register 5
IOMMU_PMU_ACCESS_LOW_REG6	0x0290	IOMMU PMU Access Low Register 6
IOMMU_PMU_ACCESS_HIGH_REG6	0x0294	IOMMU PMU Access High Register 6
IOMMU_PMU_HIT_LOW_REG6	0x0298	IOMMU PMU Hit Low Register 6
IOMMU_PMU_HIT_HIGH_REG6	0x029C	IOMMU PMU Hit High Register 6
IOMMU_PMU_ACCESS_LOW_REG7	0x02D0	IOMMU PMU Access Low Register 7
IOMMU_PMU_ACCESS_HIGH_REG7	0x02D4	IOMMU PMU Access High Register 7
IOMMU_PMU_HIT_LOW_REG7	0x02D8	IOMMU PMU Hit Low Register 7
IOMMU_PMU_HIT_HIGH_REG7	0x02DC	IOMMU PMU Hit High Register 7
IOMMU_PMU_ACCESS_LOW_REG8	0x02E0	IOMMU PMU Access Low Register 8
IOMMU_PMU_ACCESS_HIGH_REG8	0x02E4	IOMMU PMU Access High Register 8
IOMMU_PMU_HIT_LOW_REG8	0x02E8	IOMMU PMU Hit Low Register 8
IOMMU_PMU_HIT_HIGH_REG8	0x02EC	IOMMU PMU Hit High Register 8
IOMMU_PMU_TL _LOW_REG0	0x0300	IOMMU Total Latency Low Register 0
IOMMU_PMU_TL _HIGH_REG0	0x0304	IOMMU Total Latency High Register 0
IOMMU_PMU_ML _REG0	0x0308	IOMMU Max Latency Register 0
IOMMU_PMU_TL _LOW_REG1	0x0310	IOMMU Total Latency Low Register 1
IOMMU_PMU_TL _HIGH_REG1	0x0314	IOMMU Total Latency High Register 1
IOMMU_PMU_ML _REG1	0x0318	IOMMU Max Latency Register 1
IOMMU_PMU_TL _LOW_REG2	0x0320	IOMMU Total Latency Low Register 2
IOMMU_PMU_TL _HIGH_REG2	0x0324	IOMMU Total Latency High Register 2
IOMMU_PMU_ML _REG2	0x0328	IOMMU Max Latency Register 2
IOMMU_PMU_TL _LOW_REG3	0x0330	IOMMU Total Latency Low Register 3
IOMMU_PMU_TL _HIGH_REG3	0x0334	IOMMU Total Latency High Register 3
IOMMU_PMU_ML _REG3	0x0338	IOMMU Max Latency Register 3
IOMMU_PMU_TL _LOW_REG4	0x0340	IOMMU Total Latency Low Register 4
IOMMU_PMU_TL _HIGH_REG4	0x0344	IOMMU Total Latency High Register 4
IOMMU_PMU_ML _REG4	0x0348	IOMMU Max Latency Register 4
IOMMU_PMU_TL _LOW_REG5	0x0350	IOMMU Total Latency Low Register 5
IOMMU_PMU_TL _HIGH_REG5	0x0354	IOMMU Total Latency High Register 5
IOMMU_PMU_ML _REG5	0x0358	IOMMU Max Latency Register 5
IOMMU_PMU_TL_LOW_REG6	0x0360	IOMMU Total Latency Low Register 6
IOMMU_PMU_TL_HIGH_REG6	0x0364	IOMMU Total Latency High Register 6
IOMMU_PMU_ML _REG6	0x0368	IOMMU Max Latency Register 6

3.6.6 Register Description

3.6.6.1 IOMMU Reset Register (Default Value: 0x8003_007F)

Offset: 0x0010			**Register Name: IOMMU_RESET _REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x1	IOMMU_RESET

			IOMMU Software Reset Switch 0: Set reset signal 1: Release reset signal Before IOMMU software reset operation, ensure IOMMU never be opened; Or no unfinished bus operation; Or DRAM and the peripherals have opened the corresponding switch, for shielding the effects of IOMMU reset .
30:18	/	/	/
17	R/W	0x1	PTW_CACHE_RESET PTW Cache address convert lane software reset switch. 0: Set reset signal 1: Release reset signal When PTW Cache occurs abnormal, the bit is used for resetting PTW Cache individually.
16	R/W	0x1	MACRO_TLB_RESET Macro TLB address convert lane software reset switch. 0: Set reset signal 1: Release reset signal When PTW Cache occurs abnormal, the bit is used for resetting PTW Cache individually.
15:7	/	/	/
6	R/W	0x1	MASTER6_RESET Master6 address convert lane software reset switch. 0: Set reset signal 1: Release reset signal When Master6 occurs abnormal, the bit is used for resetting PTW Cache individually.
5	R/W	0x1	MASTER5_RESET Master5 address convert lane software reset switch. 0: Set reset signal 1: Release reset signal When Master5 occurs abnormal, the bit is used for resetting PTW Cache individually.
4	R/W	0x1	MASTER4_RESET Master4 address convert lane software reset switch. 0: Set reset signal 1: Release reset signal When Master4 occurs abnormal, the bit is used for resetting PTW Cache individually.
3	R/W	0x1	MASTER3_RESET Master3 address convert lane software reset switch. 0: Set reset signal 1: Release reset signal When Master3 occurs abnormal, the bit is used for resetting PTW Cache individually.
2	R/W	0x1	MASTER2_RESET Master2 address convert lane software reset switch.

			0: Set reset signal 1: Release reset signal When Master2 occurs abnormal, the bit is used for resetting PTW Cache individually.
1	R/W	0x1	MASTER1_RESET Master1 address convert lane software reset switch. 0: Set reset signal 1: Release reset signal When Master1 occurs abnormal, the bit is used for resetting PTW Cache individually.
0	R/W	0x1	MASTER0_RESET Master0 address convert lane software reset switch. 0: Set reset signal 1: Release reset signal When Master0 occurs abnormal, the bit is used for resetting PTW Cache individually.

3.6.6.2 IOMMU Enable Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: IOMMU_ENABLE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	ENABLE IOMMU module enable switch 0: Disable IOMMU 1: Enable IOMMU Before IOMMU address mapping function opens, configure the Translation Table Base register; or ensure all masters are in bypass status or no the status of sending bus demand （such as reset）

3.6.6.3 IOMMU Bypass Register (Default Value: 0x0000_007f)

Offset: 0x0030			Register Name: IOMMU_BYPASS_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:7	/	/	/
6	R/W	0x1	MASTER6_BYPASS Master6 bypass switch After open bypass function，IOMMU can not map the address of Master6 sending, and directly output the virtual address to MBUS as physical address. 0: Disable bypass function 1: Enable bypass function
5	R/W	0x1	MASTER5_BYPASS Master5 bypass switch

			After open bypass function，IOMMU can not map the address of Master5 sending, and directly output the virtual address to MBUS as physical address. 0: Disable bypass function 1: Enable bypass function
4	R/W	0x1	MASTER4_BYPASS Master4 bypass switch After open bypass function，IOMMU can not map the address of Master4 sending, and directly output the virtual address to MBUS as physical address. 0: Disable bypass function 1: Enable bypass function
3	R/W	0x1	MASTER3_BYPASS Master3 bypass switch After open bypass function，IOMMU can not map the address of Master3 sending, and directly output the virtual address to MBUS as physical address. 0: Disable bypass function 1: Enable bypass function
2	R/W	0x1	MASTER2_BYPASS Master2 bypass switch After open bypass function，IOMMU can not map the address of Master2 sending, and directly output the virtual address to MBUS as physical address. 0: Disable bypass function 1: Enable bypass function
1	R/W	0x1	MASTER1_BYPASS Master1 bypass switch After open bypass function，IOMMU can not map the address of Master1 sending, and directly output the virtual address to MBUS as physical address. 0: Disable bypass function 1: Enable bypass function
0	R/W	0x1	MASTER0_BYPASS Master0 bypass switch After open bypass function，IOMMU can not map the address of Master0 sending, and directly output the virtual address to MBUS as physical address. 0: Disable bypass function 1: Enable bypass function

NOTE

Operating the register belongs to non-accurate timing sequence control function.That is, before the function is valid, master operation will complete address mapping function, and after the operation will not perform address mapping. It is suggested that master is in reset state or in no any bus operation before operating the register .

3.6.6.4 IOMMU Auto Gating Register (Default Value: 0x0000_0000)

Offset: 0x0040			Register Name: IOMMU_AUTO_GATING_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	IOMMU_AUTO_GATING IOMMU circuit auto gating control. The purpose is decreasing power consumption of the module. 0: Disable auto gating function 1: Enable auto gating function

3.6.6.5 IOMMU Write Buffer Control Register (Default Value: 0x0000_0011)

Offset: 0x0044			Register Name: IOMMU_WBUF_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:5	/	/	/
4	R/W	0x1	MASTER4_WBUF_CTRL Master4 write buffer control bit 0: Disable write buffer 1: Enable write buffer
3:1	/	/	/
0	R/W	0x1	MASTER0_WBUF_CTRL Master0 write buffer control bit 0: Disable write buffer 1: Enable write buffer

3.6.6.6 IOMMU Out Of Order Control Register (Default Value: 0x0000_0011)

Offset: 0x0048			Register Name: IOMMU_OOO_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:5	/	/	/
4	R/W	0x1	MASTER4_OOO_CTRL Master4 out-of-order control bit 0: Disable out-of-order 1: Enable out-of-order
3:1	/	/	/
0	R/W	0x1	MASTER0_OOO_CTRL Master0 out-of-order control bit 0: Disable out-of-order 1: Enable out-of-order

3.6.6.7 IOMMU 4KB Boundary Protect Control Register (Default Value: 0x0000_007F)

Offset: 0x004C			Register Name: IOMMU_4KB_BDY_PRT_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:7	/	/	/
6	R/W	0x1	MASTER6_4KB_BDY_PRT_CTRL Master6 4KB boundary protect control bit 0: Disable 4KB boundary protect 1: Enable 4KB boundary protect
5	R/W	0x1	MASTER5_4KB_BDY_PRT_CTRL Master4 4KB boundary protect control bit 0: Disable 4KB boundary protect 1: Enable 4KB boundary protect
4	R/W	0x1	MASTER4_4KB_BDY_PRT_CTRL Master4 4KB boundary protect control bit 0: Disable 4KB boundary protect 1: Enable 4KB boundary protect
3	R/W	0x1	MASTER3_4KB_BDY_PRT_CTRL Master3 4KB boundary protect control bit 0: Disable 4KB boundary protect 1: Enable 4KB boundary protect
2	R/W	0x1	MASTER2_4KB_BDY_PRT_CTRL Master2 4KB boundary protect control bit 0: Disable 4KB boundary protect 1: Enable 4KB boundary protect
1	R/W	0x1	MASTER1_4KB_BDY_PRT_CTRL Master1 4KB boundary protect control bit 0: Disable 4KB boundary protect 1: Enable 4KB boundary protect
0	R/W	0x1	MASTER0_4KB_BDY_PRT_CTRL Master0 4KB boundary protect control bit 0: Disable 4KB boundary protect 1: Enable 4KB boundary protect

NOTE

When Master sends the virtual address across the 4KB boundary, 4KB protection unit will split it into two serial access.

3.6.6.8 IOMMU Translation Table Base Register (Default Value: 0x0000_0000)

Offset: 0x0050			Register Name: IOMMU_TTB_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:14	R/W	0x0	TTB Level1 page table starting address, aligned to 16 KB. When operating the register , IOMMU address mapping function must be closed, namely IOMMU_ENABLE_REG is 0 ; Or Bypass function of all main

			equipment is set to 1 or not the state of transfer bus commands (such as setting).
13:0	/	/	/

3.6.6.9 IOMMU TLB Enable Register (Default Value: 0x0003_007F)

Offset: 0x0060			Register Name: IOMMU_TLB_ENABLE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17	R/W	0x1	PTW_CACHE_ENABLE PTW Cache enable bit 0: Disable 1: Enable
16	R/W	0x1	MACRO_TLB_ENABLE Macro TLB enable bit 0: Disable 1: Enable
15:7	/	/	/
6	R/W	0x1	MICRO_TLB6_ENABLE Micro TLB6 enable bit 0: Disable 1: Enable
5	R/W	0x1	MICRO_TLB5_ENABLE Micro TLB5 enable bit 0: Disable 1: Enable
4	R/W	0x1	MICRO_TLB4_ENABLE Micro TLB4 enable bit 0: Disable 1: Enable
3	R/W	0x1	MICRO_TLB3_ENABLE Micro TLB3 enable bit 0: Disable 1: Enable
2	R/W	0x1	MICRO_TLB2_ENABLE Micro TLB2 enable bit 0: Disable 1: Enable
1	R/W	0x1	MICRO_TLB1_ENABLE Micro TLB1 enable bit 0: Disable 1: Enable
0	R/W	0x1	MICRO_TLB0_ENABLE Micro TLB0 enable bit 0: Disable

			1: Enable

3.6.6.10 IOMMU TLB Prefetch Register (Default Value: 0x0000_0000)

Offset: 0x0070			Register Name: IOMMU_TLB_PREFETCH_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:7	/	/	/
6	R/W	0x0	MICRO_TLB6_PREFETCH Micro TLB6 prefetch enable 0: Disable 1: Enable
5	R/W	0x0	MICRO_TLB5_PREFETCH Micro TLB5 prefetch enable 0: Disable 1: Enable
4	R/W	0x0	MICRO_TLB4_PREFETCH Micro TLB4 prefetch enable 0: Disable 1: Enable
3	R/W	0x0	MICRO_TLB3_PREFETCH Micro TLB3 prefetch enable 0: Disable 1: Enable
2	R/W	0x0	MICRO_TLB2_PREFETCH Micro TLB2 prefetch enable 0: Disable 1: Enable
1	R/W	0x0	MICRO_TLB1_PREFETCH Micro TLB1 prefetch enable 0: Disable 1: Enable
0	R/W	0x0	MICRO_TLB0_PREFETCH Micro TLB0 prefetch enable 0: Disable 1: Enable

3.6.6.11 IOMMU TLB Flush Enable Register (Default Value: 0x0000_0000)

Offset: 0x0080			Register Name: IOMMU_TLB_FLUSH_ENABLE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17	R/WAC	0x0	PTW_CACHE_FLUSH Clear PTW Cache 0: No clear operation or clear operation completed

			1: Enable is cleared After Flush operation completes, the bit can clear 0 automatically.
16	R/WAC	0x0	MACRO_TLB_FLUSH Clear Macro TLB 0: No clear operation or clear operation completed 1: Enable is cleared After Flush operation completes, the bit can clear 0 automatically.
15:7	/	/	/
6	R/WAC	0x0	MICRO_TLB6_FLUSH Clear Micro TLB6 0: No clear operation or clear operation completed 1: Enable is cleared After Flush operation completes, the bit can clear 0 automatically.
5	R/WAC	0x0	MICRO_TLB5_FLUSH Clear Micro TLB5 0: No clear operation or clear operation completed 1: Enable is cleared After Flush operation completes, the bit can clear 0 automatically.
4	R/WAC	0x0	MICRO_TLB4_FLUSH Clear Micro TLB4 0: No clear operation or clear operation completed 1: Enable is cleared After Flush operation completes, the bit can clear 0 automatically.
3	R/WAC	0x0	MICRO_TLB3_FLUSH Clear Micro TLB3 0: No clear operation or clear operation completed 1: Enable is cleared After Flush operation completes, the bit can clear 0 automatically.
2	R/WAC	0x0	MICRO_TLB2_FLUSH Clear Micro TLB2 0: No clear operation or clear operation completed 1: Enable is cleared After Flush operation completes, the bit can clear 0 automatically.
1	R/WAC	0x0	MICRO_TLB1_FLUSH Clear Micro TLB1 0: No clear operation or clear operation completed 1: Enable is cleared After Flush operation completes, the bit can clear 0 automatically.
0	R/WAC	0x0	MICRO_TLB0_FLUSH Clear Micro TLB1 0: No clear operation or clear operation completed 1: Enable is cleared After Flush operation completes, the bit can clear 0 automatically.

NOTE

When performing flush operation, all TLB/Cache access will be paused.

Before flush starts, the operation that has entered TLB continues to complete.

3.6.6.12 IOMMU TLB Invalidation Address Register (Default Value: 0x0000_0000)

Offset: 0x0090			Register Name: IOMMU_TLB_IVLD_ADDR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:12	R/W	0x0	TLB_IVLD_ADDR TLB invalid address, aligned to 4K
11:0	/	/	/

Operation:

Set the virtual address which needed to be operated in **IOMMU_TLB_IVLD_ADDR_REG**.

Set the mask of virtual address which needed to be operated in **IOMMU_TLB_IVLD_ADDR_MASK_REG**.

Write '1' to **IOMMU_TLB_IVLD_ENABLE_REG[0].**

Read the **IOMMU_TLB_IVLD_ENABLE_REG[0]**, when it is '0', it indicates that invalidation behavior is finished.

NOTE

When performing invalidation operation, TLB/Cache operation has not affected.

After or Before invalidation starts, there is no absolute relationship between same address switch operation and Invalidation operation.

3.6.6.13 IOMMU TLB Invalidation Address Mask Register (Default Value: 0x0000_0000)

Offset: 0x0094			Register Name: IOMMU_TLB_IVLD_ADDR_MASK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:12	R/W	0x0	TLB_IVLD_ADDR_MASK TLB invalid address mask register, aligned to 4K
11:0	/	/	/

3.6.6.14 IOMMU TLB Invalidation Enable Register (Default Value: 0x0000_0000)

Offset: 0x0098			Register Name: IOMMU_TLB_IVLD_ENABLE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/WAC	0x0	TLB_IVLD_ENABLE Enable TLB invalid operation 0: No-operation or operation completed 1: Enable is invalid After invalidation operation completed, the bit can clear 0 automatically. When operating Invalidation, TLB/Cache operation has not affected. After or Before Invalidation starts, there is no absolute relationship between same address switch operation and Invalidation operation.

3.6.6.15 IOMMU PC Invalidation Address Register (Default Value: 0x0000_0000)

Offset: 0x00A0			Register Name: IOMMU_PC_IVLD_ADDR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:20	R/W	0x0	PC_IVLD_ADDR PTW Cache invalid address, aligned to 1M.
19:0	/	/	/

3.6.6.16 IOMMU PC Invalidation Enable Register (Default Value: 0x0000_0000)

Offset: 0x00A8			Register Name: IOMMU_PC_IVLD_ENABLE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/WAC	0x0	PC_IVLD_ENABLE Enable PTW Cache invalid operation 0: No-operation or operation completed 1: Enable is invalid After invalidation operation completed, the bit can clear 0 automatically. After or Before Invalidation starts, there is no absolute relationship between same address switch operation and Invalidation operation.

3.6.6.17 IOMMU Domain Authority Control Register 0 (Default Value: 0x0000_0000)

Offset: 0x00B0			Register Name: IOMMU_DM_AUT_CTRL_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/
29	R/W	0x0	DM1_M6_WT_AUT_CTRL Domain1 write permission control for master6 0: The write-operation is available 1: The write-operation is unavailable
28	R/W	0x0	DM1_M6_RD_AUT_CTRL Domain1 read permission control for master6 0: The read-operation is available 1: The read-operation is unavailable
27	R/W	0x0	DM1_M5_WT_AUT_CTRL Domain1 write permission control for master5 0: The write-operation is available 1: The write-operation is unavailable
26	R/W	0x0	DM1_M5_RD_AUT_CTRL Domain1 read permission control for master5 0: The read-operation is available 1: The read-operation is unavailable
25	R/W	0x0	DM1_M4_WT_AUT_CTRL Domain1 write permission control for master4

			0: The write-operation is available 1: The write-operation is unavailable
24	R/W	0x0	DM1_M4_RD_AUT_CTRL Domain1 read permission control for master4 0: The read-operation is available 1: The read-operation is unavailable
23	R/W	0x0	DM1_M3_WT_AUT_CTRL Domain1 write permission control for master3 0: The write-operation is available 1: The write-operation is unavailable
22	R/W	0x0	DM1_M3_RD_AUT_CTRL Domain1 read permission control for master3 0: The read-operation is available 1: The read-operation is unavailable
21	R/W	0x0	DM1_M2_WT_AUT_CTRL Domain1 write permission control for master2 0: The write-operation is available 1: The write-operation is unavailable
20	R/W	0x0	DM1_M2_RD_AUT_CTRL Domain1 read permission control for master2 0: The read-operation is available 1: The read-operation is unavailable
19	R/W	0x0	DM1_M1_WT_AUT_CTRL Domain1 write permission control for master1 0: The write-operation is available 1: The write-operation is unavailable
18	R/W	0x0	DM1_M1_RD_AUT_CTRL Domain1 read permission control for master1 0: The read-operation is available 1: The read-operation is unavailable
17	R/W	0x0	DM1_M0_WT_AUT_CTRL Domain1 write permission control for master0 0: The write-operation is available 1: The write-operation is unavailable
16	R/W	0x0	DM1_M0_RD_AUT_CTRL Domain1 read permission control for master0 0: The read-operation is available 1: The read-operation is unavailable
15:14	/	/	/
13	R	0x0	DM0_M6_WT_AUT_CTRL Domain0 write permission control for master6 0: The write-operation is available 1: The write-operation is unavailable
12	R	0x0	DM0_M6_RD_AUT_CTRL Domain0 read permission control for master6 0: The read-operation is available

			1: The read-operation is unavailable
11	R	0x0	DM0_M5_WT_AUT_CTRL Domain0 write permission control for master5 0: The write-operation is available 1: The write-operation is unavailable
10	R	0x0	DM0_M5_RD_AUT_CTRL Domain0 read permission control for master5 0: The read-operation is available 1: The read-operation is unavailable
9	R	0x0	DM0_M4_WT_AUT_CTRL Domain0 write permission control for master4 0: The write-operation is available 1: The write-operation is unavailable
8	R	0x0	DM0_M4_RD_AUT_CTRL Domain0 read permission control for master4 0: The read-operation is available 1: The read-operation is unavailable
7	R	0x0	DM0_M3_WT_AUT_CTRL Domain0 write permission control for master3 0: The write-operation is available 1: The write-operation is unavailable
6	R	0x0	DM0_M3_RD_AUT_CTRL Domain0 read permission control for master3 0: The read-operation is available 1: The read-operation is unavailable
5	R	0x0	DM0_M2_WT_AUT_CTRL Domain0 write permission control for master2 0: The write-operation is available 1: The write-operation is unavailable
4	R	0x0	DM0_M2_RD_AUT_CTRL Domain0 read permission control for master2 0: The read-operation is available 1: The read-operation is unavailable
3	R	0x0	DM0_M1_WT_AUT_CTRL Domain0 write permission control for master1 0: The write-operation is available 1: The write-operation is unavailable
2	R	0x0	DM0_M1_RD_AUT_CTRL Domain0 read permission control for master1 0: The read-operation is available 1: The read-operation is unavailable
1	R	0x0	DM0_M0_WT_AUT_CTRL Domain0 write permission control for master0 0: The write-operation is available 1: The write-operation is unavailable
0	R	0x0	DM0_M0_RD_AUT_CTRL

			Domain0 read permission control for master0 0: The read-operation is available 1: The read-operation is unavailable

NOTE

Software can be set up 15 different permission control types , which are set in IOMMU_DM_AUT_CTRL_REG0 ~ 7. As well as a default access control type, domain0. The read/write operation of DOMIAN1 ~ 15 is unlimited by default.
Software need set the corresponding permission control domain index of the page table item in the secondary page table entries[7:4], the default value is 0, use domian0, namely the read/write operation is not controlled.
Setting REG_ARD_OVWT can mask Domain control defined by IOMMU_DM_AUT_CTRL_REG0~7.All Level2 page table type are covered by the type of REG_ARD_OVWT. The read/write operation is permitted by default.

3.6.6.18 IOMMU Domain Authority Control Register 1 (Default Value: 0x0000_0000)

Offset: 0x00B4			Register Name: IOMMU_DM_AUT_CTRL_REG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/
29	R/W	0x0	DM3_M6_WT_AUT_CTRL Domain3 write permission control for master6 0: The write-operation is available 1: The write-operation is unavailable
28	R/W	0x0	DM3_M6_RD_AUT_CTRL Domain3 read permission control for master6 0: The read-operation is available 1: The read-operation is unavailable
27	R/W	0x0	DM3_M5_WT_AUT_CTRL Domain3 write permission control for master5 0: The write-operation is available 1: The write-operation is unavailable
26	R/W	0x0	DM3_M5_RD_AUT_CTRL Domain3 read permission control for master5 0: The read-operation is available 1: The read-operation is unavailable
25	R/W	0x0	DM3_M4_WT_AUT_CTRL Domain3 write permission control for master4 0: The write-operation is available 1: The write-operation is unavailable
24	R/W	0x0	DM3_M4_RD_AUT_CTRL Domain3 read permission control for master4 0: The read-operation is available 1: The read-operation is unavailable
23	R/W	0x0	DM3_M3_WT_AUT_CTRL Domain3 write permission control for master3 0: The write-operation is available 1: The write-operation is unavailable

22	R/W	0x0	DM3_M3_RD_AUT_CTRL Domain3 read permission control for master3 0: The read-operation is available 1: The read-operation is unavailable
21	R/W	0x0	DM3_M2_WT_AUT_CTRL Domain3 write permission control for master2 0: The write-operation is available 1: The write-operation is unavailable
20	R/W	0x0	DM3_M2_RD_AUT_CTRL Domain3 read permission control for master2 0: The read-operation is available 1: The read-operation is unavailable
19	R/W	0x0	DM3_M1_WT_AUT_CTRL Domain3 write permission control for master1 0: The write-operation is available 1: The write-operation is unavailable
18	R/W	0x0	DM3_M1_RD_AUT_CTRL Domain3 read permission control for master1 0: The read-operation is available 1: The read-operation is unavailable
17	R/W	0x0	DM3_M0_WT_AUT_CTRL Domain3 write permission control for master0 0: The write-operation is available 1: The write-operation is unavailable
16	R/W	0x0	DM3_M0_RD_AUT_CTRL Domain3 read permission control for master0 0: The read-operation is available 1: The read-operation is unavailable
15:14	/	/	/
13	R/W	0x0	DM2_M6_WT_AUT_CTRL Domain2 write permission control for master6 0: The write-operation is available 1: The write-operation is unavailable
12	R/W	0x0	DM2_M6_RD_AUT_CTRL Domain2 read permission control for master6 0: The read-operation is available 1: The read-operation is unavailable
11	R/W	0x0	DM2_M5_WT_AUT_CTRL Domain2 write permission control for master5 0: The write-operation is available 1: The write-operation is unavailable
10	R/W	0x0	DM2_M5_RD_AUT_CTRL Domain2 read permission control for master5 0: The read-operation is available 1: The read-operation is unavailable
9	R/W	0x0	DM2_M4_WT_AUT_CTRL

			Domain2 write permission control for master4 0: The write-operation is available 1: The write-operation is unavailable
8	R/W	0x0	DM2_M4_RD_AUT_CTRL Domain2 read permission control for master4 0: The read-operation is available 1: The read-operation is unavailable
7	R/W	0x0	DM2_M3_WT_AUT_CTRL Domain2 write permission control for master3 0: The write-operation is available 1: The write-operation is unavailable
6	R/W	0x0	DM2_M3_RD_AUT_CTRL Domain2 read permission control for master3 0: The read-operation is available 1: The read-operation is unavailable
5	R/W	0x0	DM2_M2_WT_AUT_CTRL Domain2 write permission control for master2 0: The write-operation is available 1: The write-operation is unavailable
4	R/W	0x0	DM2_M2_RD_AUT_CTRL Domain2 read permission control for master2 0: The read-operation is available 1: The read-operation is unavailable
3	R/W	0x0	DM2_M1_WT_AUT_CTRL Domain2 write permission control for master1 0: The write-operation is available 1: The write-operation is unavailable
2	R/W	0x0	DM2_M1_RD_AUT_CTRL Domain2 read permission control for master1 0: The read-operation is available 1: The read-operation is unavailable
1	R/W	0x0	DM2_M0_WT_AUT_CTRL Domain2 write permission control for master0 0: The write-operation is available 1: The write-operation is unavailable
0	R/W	0x0	DM2_M0_RD_AUT_CTRL Domain2 read permission control for master0 0: The read-operation is available 1: The read-operation is unavailable

3.6.6.19 IOMMU Domain Authority Control Register 2 (Default Value: 0x0000_0000)

Offset: 0x00B8			**Register Name: IOMMU_DM_AUT_CTRL_REG2**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/

29	R/W	0x0	DM5_M6_WT_AUT_CTRL Domain5 write permission control for master6 0: The write-operation is available 1: The write-operation is unavailable
28	R/W	0x0	DM5_M6_RD_AUT_CTRL Domain5 read permission control for master6 0: The read-operation is available 1: The read-operation is unavailable
27	R/W	0x0	DM5_M5_WT_AUT_CTRL Domain5 write permission control for master5 0: The write-operation is available 1: The write-operation is unavailable
26	R/W	0x0	DM5_M5_RD_AUT_CTRL Domain5 read permission control for master5 0: The read-operation is available 1: The read-operation is unavailable
25	R/W	0x0	DM5_M4_WT_AUT_CTRL Domain5 write permission control for master4 0: The write-operation is available 1: The write-operation is unavailable
24	R/W	0x0	DM5_M4_RD_AUT_CTRL Domain5 read permission control for master4 0: The read-operation is available 1: The read-operation is unavailable
23	R/W	0x0	DM5_M3_WT_AUT_CTRL Domain5 write permission control for master3 0: The write-operation is available 1: The write-operation is unavailable
22	R/W	0x0	DM5_M3_RD_AUT_CTRL Domain5 read permission control for master3 0: The read-operation is available 1: The read-operation is unavailable
21	R/W	0x0	DM5_M2_WT_AUT_CTRL Domain5 write permission control for master2 0: The write-operation is available 1: The write-operation is unavailable
20	R/W	0x0	DM5_M2_RD_AUT_CTRL Domain5 read permission control for master2 0: The read-operation is available 1: The read-operation is unavailable
19	R/W	0x0	DM5_M1_WT_AUT_CTRL Domain5 write permission control for master1 0: The write-operation is available 1: The write-operation is unavailable
18	R/W	0x0	DM5_M1_RD_AUT_CTRL Domain5 read permission control for master1

			0: The read-operation is available 1: The read-operation is unavailable
17	R/W	0x0	DM5_M0_WT_AUT_CTRL Domain5 write permission control for master0 0: The write-operation is available 1: The write-operation is unavailable
16	R/W	0x0	DM5_M0_RD_AUT_CTRL Domain5 read permission control for master0 0: The read-operation is available 1: The read-operation is unavailable
15:14	/	/	/
13	R/W	0x0	DM4_M6_WT_AUT_CTRL Domain4 write permission control for master6 0: The write-operation is available 1: The write-operation is unavailable
12	R/W	0x0	DM4_M6_RD_AUT_CTRL Domain4 read permission control for master6 0: The read-operation is available 1: The read-operation is unavailable
11	R/W	0x0	DM4_M5_WT_AUT_CTRL Domain4 write permission control for master5 0: The write-operation is available 1: The write-operation is unavailable
10	R/W	0x0	DM4_M5_RD_AUT_CTRL Domain4 read permission control for master5 0: The read-operation is available 1: The read-operation is unavailable
9	R/W	0x0	DM4_M4_WT_AUT_CTRL Domain4 write permission control for master4 0: The write-operation is available 1: The write-operation is unavailable
8	R/W	0x0	DM4_M4_RD_AUT_CTRL Domain4 read permission control for master4 0: The read-operation is available 1: The read-operation is unavailable
7	R/W	0x0	DM4_M3_WT_AUT_CTRL Domain4 write permission control for master3 0: The write-operation is available 1: The write-operation is unavailable
6	R/W	0x0	DM4_M3_RD_AUT_CTRL Domain4 read permission control for master3 0: The read-operation is available 1: The read-operation is unavailable
5	R/W	0x0	DM4_M2_WT_AUT_CTRL Domain4 write permission control for master2 0: The write-operation is available

			1: The write-operation is unavailable
4	R/W	0x0	DM4_M2_RD_AUT_CTRL Domain4 read permission control for master2 0: The read-operation is available 1: The read-operation is unavailable
3	R/W	0x0	DM4_M1_WT_AUT_CTRL Domain4 write permission control for master1 0: The write-operation is available 1: The write-operation is unavailable
2	R/W	0x0	DM4_M1_RD_AUT_CTRL Domain4 read permission control for master1 0: The read-operation is available 1: The read-operation is unavailable
1	R/W	0x0	DM4_M0_WT_AUT_CTRL Domain4 write permission control for master0 0: The write-operation is available 1: The write-operation is unavailable
0	R/W	0x0	DM4_M0_RD_AUT_CTRL Domain4 read permission control for master0 0: The read-operation is available 1: The read-operation is unavailable

3.6.6.20 IOMMU Domain Authority Control Register 3 (Default Value: 0x0000_0000)

Offset: 0x00BC			**Register Name: IOMMU_DM_AUT_CTRL_REG3**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/
29	R/W	0x0	DM7_M6_WT_AUT_CTRL Domain7 write permission control for master6 0: The write-operation is available 1: The write-operation is unavailable
28	R/W	0x0	DM7_M6_RD_AUT_CTRL Domain7 read permission control for master6 0: The read-operation is available 1: The read-operation is unavailable
27	R/W	0x0	DM7_M5_WT_AUT_CTRL Domain7 write permission control for master5 0: The write-operation is available 1: The write-operation is unavailable
26	R/W	0x0	DM7_M5_RD_AUT_CTRL Domain7 read permission control for master5 0: The read-operation is available 1: The read-operation is unavailable
25	R/W	0x0	DM7_M4_WT_AUT_CTRL Domain7 write permission control for master4

			0: The write-operation is available 1: The write-operation is unavailable
24	R/W	0x0	DM7_M4_RD_AUT_CTRL Domain7 read permission control for master4 0: The read-operation is available 1: The read-operation is unavailable
23	R/W	0x0	DM7_M3_WT_AUT_CTRL Domain7 write permission control for master3 0: The write-operation is available 1: The write-operation is unavailable
22	R/W	0x0	DM7_M3_RD_AUT_CTRL Domain7 read permission control for master3 0: The read-operation is available 1: The read-operation is unavailable
21	R/W	0x0	DM7_M2_WT_AUT_CTRL Domain7 write permission control for master2 0: The write-operation is available 1: The write-operation is unavailable
20	R/W	0x0	DM7_M2_RD_AUT_CTRL Domain7 read permission control for master2 0: The read-operation is available 1: The read-operation is unavailable
19	R/W	0x0	DM7_M1_WT_AUT_CTRL Domain7 write permission control for master1 0: The write-operation is available 1: The write-operation is unavailable
18	R/W	0x0	DM7_M1_RD_AUT_CTRL Domain7 read permission control for master1 0: The read-operation is available 1: The read-operation is unavailable
17	R/W	0x0	DM7_M0_WT_AUT_CTRL Domain7 write permission control for master0 0: The write-operation is available 1: The write-operation is unavailable
16	R/W	0x0	DM7_M0_RD_AUT_CTRL Domain7 read permission control for master0 0: The read-operation is available 1: The read-operation is unavailable
15:14	/	/	/
13	R/W	0x0	DM6_M6_WT_AUT_CTRL Domain6 write permission control for master6 0: The write-operation is available 1: The write-operation is unavailable
12	R/W	0x0	DM6_M6_RD_AUT_CTRL Domain6 read permission control for master6 0: The read-operation is available

			1: The read-operation is unavailable
11	R/W	0x0	DM6_M5_WT_AUT_CTRL Domain6 write permission control for master5 0: The write-operation is available 1: The write-operation is unavailable
10	R/W	0x0	DM6_M5_RD_AUT_CTRL Domain6 read permission control for master5 0: The read-operation is available 1: The read-operation is unavailable
9	R/W	0x0	DM6_M4_WT_AUT_CTRL Domain6 write permission control for master4 0: The write-operation is available 1: The write-operation is unavailable
8	R/W	0x0	DM6_M4_RD_AUT_CTRL Domain6 read permission control for master4 0: The read-operation is available 1: The read-operation is unavailable
7	R/W	0x0	DM6_M3_WT_AUT_CTRL Domain6 write permission control for master3 0: The write-operation is available 1: The write-operation is unavailable
6	R/W	0x0	DM6_M3_RD_AUT_CTRL Domain6 read permission control for master3 0: The read-operation is available 1: The read-operation is unavailable
5	R/W	0x0	DM6_M2_WT_AUT_CTRL Domain6 write permission control for master2 0: The write-operation is available 1: The write-operation is unavailable
4	R/W	0x0	DM6_M2_RD_AUT_CTRL Domain6 read permission control for master2 0: The read-operation is available 1: The read-operation is unavailable
3	R/W	0x0	DM6_M1_WT_AUT_CTRL Domain6 write permission control for master1 0: The write-operation is available 1: The write-operation is unavailable
2	R/W	0x0	DM6_M1_RD_AUT_CTRL Domain6 read permission control for master1 0: The read-operation is available 1: The read-operation is unavailable
1	R/W	0x0	DM6_M0_WT_AUT_CTRL Domain6 write permission control for master0 0: The write-operation is available 1: The write-operation is unavailable
0	R/W	0x0	DM6_M0_RD_AUT_CTRL

			Domain6 read permission control for master0 0: The read-operation is available 1: The read-operation is unavailable

3.6.6.21 IOMMU Domain Authority Control Register 4 (Default Value: 0x0000_0000)

Offset: 0x00C0			Register Name: IOMMU_DM_AUT_CTRL_REG4
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/
29	R/W	0x0	DM9_M6_WT_AUT_CTRL Domain9 write permission control for master6 0: The write-operation is available 1: The write-operation is unavailable
28	R/W	0x0	DM9_M6_RD_AUT_CTRL Domain9 read permission control for master6 0: The read-operation is available 1: The read-operation is unavailable
27	R/W	0x0	DM9_M5_WT_AUT_CTRL Domain9 write permission control for master5 0: The write-operation is available 1: The write-operation is unavailable
26	R/W	0x0	DM9_M5_RD_AUT_CTRL Domain9 read permission control for master5 0: The read-operation is available 1: The read-operation is unavailable
25	R/W	0x0	DM9_M4_WT_AUT_CTRL Domain9 write permission control for master4 0: The write-operation is available 1: The write-operation is unavailable
24	R/W	0x0	DM9_M4_RD_AUT_CTRL Domain9 read permission control for master4 0: The read-operation is available 1: The read-operation is unavailable
23	R/W	0x0	DM9_M3_WT_AUT_CTRL Domain9 write permission control for master3 0: The write-operation is available 1: The write-operation is unavailable
22	R/W	0x0	DM9_M3_RD_AUT_CTRL Domain9 read permission control for master3 0: The read-operation is available 1: The read-operation is unavailable
21	R/W	0x0	DM9_M2_WT_AUT_CTRL Domain9 write permission control for master2 0: The write-operation is available 1: The write-operation is unavailable

20	R/W	0x0	DM9_M2_RD_AUT_CTRL Domain9 read permission control for master2 0: The read-operation is available 1: The read-operation is unavailable
19	R/W	0x0	DM9_M1_WT_AUT_CTRL Domain9 write permission control for master1 0: The write-operation is available 1: The write-operation is unavailable
18	R/W	0x0	DM9_M1_RD_AUT_CTRL Domain9 read permission control for master1 0: The read-operation is available 1: The read-operation is unavailable
17	R/W	0x0	DM9_M0_WT_AUT_CTRL Domain9 write permission control for master0 0: The write-operation is available 1: The write-operation is unavailable
16	R/W	0x0	DM9_M0_RD_AUT_CTRL Domain9 read permission control for master0 0: The read-operation is available 1: The read-operation is unavailable
15:14	/	/	/
13	R/W	0x0	DM8_M6_WT_AUT_CTRL Domain8 write permission control for master6 0: The write-operation is available 1: The write-operation is unavailable
12	R/W	0x0	DM8_M6_RD_AUT_CTRL Domain8 read permission control for master6 0: The read-operation is available 1: The read-operation is unavailable
11	R/W	0x0	DM8_M5_WT_AUT_CTRL Domain8 write permission control for master5 0: The write-operation is available 1: The write-operation is unavailable
10	R/W	0x0	DM8_M5_RD_AUT_CTRL Domain8 read permission control for master5 0: The read-operation is available 1: The read-operation is unavailable
9	R/W	0x0	DM8_M4_WT_AUT_CTRL Domain8 write permission control for master4 0: The write-operation is available 1: The write-operation is unavailable
8	R/W	0x0	DM8_M4_RD_AUT_CTRL Domain8 read permission control for master4 0: The read-operation is available 1: The read-operation is unavailable
7	R/W	0x0	DM8_M3_WT_AUT_CTRL

			Domain8 write permission control for master3 0: The write-operation is available 1: The write-operation is unavailable
6	R/W	0x0	DM8_M3_RD_AUT_CTRL Domain8 read permission control for master3 0: The read-operation is available 1: The read-operation is unavailable
5	R/W	0x0	DM8_M2_WT_AUT_CTRL Domain8 write permission control for master2 0: The write-operation is available 1: The write-operation is unavailable
4	R/W	0x0	DM8_M2_RD_AUT_CTRL Domain8 read permission control for master2 0: The read-operation is available 1: The read-operation is unavailable
3	R/W	0x0	DM8_M1_WT_AUT_CTRL Domain8 write permission control for master1 0: The write-operation is available 1: The write-operation is unavailable
2	R/W	0x0	DM8_M1_RD_AUT_CTRL Domain8 read permission control for master1 0: The read-operation is available 1: The read-operation is unavailable
1	R/W	0x0	DM8_M0_WT_AUT_CTRL Domain8 write permission control for master0 0: The write-operation is available 1: The write-operation is unavailable
0	R/W	0x0	DM8_M0_RD_AUT_CTRL Domain8 read permission control for master0 0: The read-operation is available 1: The read-operation is unavailable

3.6.6.22 IOMMU Domain Authority Control Register 5 (Default Value: 0x0000_0000)

Offset: 0x00C4			Register Name: IOMMU_DM_AUT_CTRL_REG5
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/
29	R/W	0x0	DM11_M6_WT_AUT_CTRL Domain11 write permission control for master6 0: The write-operation is available 1: The write-operation is unavailable
28	R/W	0x0	DM11_M6_RD_AUT_CTRL Domain11 read permission control for master6 0: The read-operation is available 1: The read-operation is unavailable

27	R/W	0x0	DM11_M5_WT_AUT_CTRL Domain11 write permission control for master5 0: The write-operation is available 1: The write-operation is unavailable
26	R/W	0x0	DM11_M5_RD_AUT_CTRL Domain11 read permission control for master5 0: The read-operation is available 1: The read-operation is unavailable
25	R/W	0x0	DM11_M4_WT_AUT_CTRL Domain11 write permission control for master4 0: The write-operation is available 1: The write-operation is unavailable
24	R/W	0x0	DM11_M4_RD_AUT_CTRL Domain11 read permission control for master4 0: The read-operation is available 1: The read-operation is unavailable
23	R/W	0x0	DM11_M3_WT_AUT_CTRL Domain11 write permission control for master3 0: The write-operation is available 1: The write-operation is unavailable
22	R/W	0x0	DM11_M3_RD_AUT_CTRL Domain11 read permission control for master3 0: The read-operation is available 1: The read-operation is unavailable
21	R/W	0x0	DM11_M2_WT_AUT_CTRL Domain11 write permission control for master2 0: The write-operation is available 1: The write-operation is unavailable
20	R/W	0x0	DM11_M2_RD_AUT_CTRL Domain11 read permission control for master2 0: The read-operation is available 1: The read-operation is unavailable
19	R/W	0x0	DM11_M1_WT_AUT_CTRL Domain11 write permission control for master1 0: The write-operation is available 1: The write-operation is unavailable
18	R/W	0x0	DM11_M1_RD_AUT_CTRL Domain11 read permission control for master1 0: The read-operation is available 1: The read-operation is unavailable
17	R/W	0x0	DM11_M0_WT_AUT_CTRL Domain11 write permission control for master0 0: The write-operation is available 1: The write-operation is unavailable
16	R/W	0x0	DM11_M0_RD_AUT_CTRL Domain11 read permission control for master0

			0: The read-operation is available 1: The read-operation is unavailable
15:14	/	/	/
13	R/W	0x0	DM10_M6_WT_AUT_CTRL Domain10 write permission control for master6 0: The write-operation is available 1: The write-operation is unavailable
12	R/W	0x0	DM10_M6_RD_AUT_CTRL Domain10 read permission control for master6 0: The read-operation is available 1: The read-operation is unavailable
11	R/W	0x0	DM10_M5_WT_AUT_CTRL Domain10 write permission control for master5 0: The write-operation is available 1: The write-operation is unavailable
10	R/W	0x0	DM10_M5_RD_AUT_CTRL Domain10 read permission control for master5 0: The read-operation is available 1: The read-operation is unavailable
9	R/W	0x0	DM10_M4_WT_AUT_CTRL Domain10 write permission control for master4 0: The write-operation is available 1: The write-operation is unavailable
8	R/W	0x0	DM10_M4_RD_AUT_CTRL Domain10 read permission control for master4 0: The read-operation is available 1: The read-operation is unavailable
7	R/W	0x0	DM10_M3_WT_AUT_CTRL Domain10 write permission control for master3 0: The write-operation is available 1: The write-operation is unavailable
6	R/W	0x0	DM10_M3_RD_AUT_CTRL Domain10 read permission control for master3 0: The read-operation is available 1: The read-operation is unavailable
5	R/W	0x0	DM10_M2_WT_AUT_CTRL Domain10 write permission control for master2 0: The write-operation is available 1: The write-operation is unavailable
4	R/W	0x0	DM10_M2_RD_AUT_CTRL Domain10 read permission control for master2 0: The read-operation is available 1: The read-operation is unavailable
3	R/W	0x0	DM10_M1_WT_AUT_CTRL Domain10 write permission control for master1 0: The write-operation is available

			1: The write-operation is unavailable
2	R/W	0x0	DM10_M1_RD_AUT_CTRL Domain10 read permission control for master1 0: The read-operation is available 1: The read-operation is unavailable
1	R/W	0x0	DM10_M0_WT_AUT_CTRL Domain10 write permission control for master0 0: The write-operation is available 1: The write-operation is unavailable
0	R/W	0x0	DM10_M0_RD_AUT_CTRL Domain10 read permission control for master0 0: The read-operation is available 1: The read-operation is unavailable

3.6.6.23 IOMMU Domain Authority Control Register 6 (Default Value: 0x0000_0000)

Offset: 0x00C8			Register Name: IOMMU_DM_AUT_CTRL_REG6
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/
29	R/W	0x0	DM13_M6_WT_AUT_CTRL Domain13 write permission control for master6 0: The write-operation is available 1: The write-operation is unavailable
28	R/W	0x0	DM13_M6_RD_AUT_CTRL Domain13 read permission control for master6 0: The read-operation is available 1: The read-operation is unavailable
27	R/W	0x0	DM13_M5_WT_AUT_CTRL Domain13 write permission control for master5 0: The write-operation is available 1: The write-operation is unavailable
26	R/W	0x0	DM13_M5_RD_AUT_CTRL Domain13 read permission control for master5 0: The read-operation is available 1: The read-operation is unavailable
25	R/W	0x0	DM13_M4_WT_AUT_CTRL Domain13 write permission control for master4 0: The write-operation is available 1: The write-operation is unavailable
24	R/W	0x0	DM13_M4_RD_AUT_CTRL Domain13 read permission control for master4 0: The read-operation is available 1: The read-operation is unavailable
23	R/W	0x0	DM13_M3_WT_AUT_CTRL Domain13 write permission control for master3

			0: The write-operation is available 1: The write-operation is unavailable
22	R/W	0x0	DM13_M3_RD_AUT_CTRL Domain13 read permission control for master3 0: The read-operation is available 1: The read-operation is unavailable
21	R/W	0x0	DM13_M2_WT_AUT_CTRL Domain13 write permission control for master2 0: The write-operation is available 1: The write-operation is unavailable
20	R/W	0x0	DM13_M2_RD_AUT_CTRL Domain13 read permission control for master2 0: The read-operation is available 1: The read-operation is unavailable
19	R/W	0x0	DM13_M1_WT_AUT_CTRL Domain13 write permission control for master1 0: The write-operation is available 1: The write-operation is unavailable
18	R/W	0x0	DM13_M1_RD_AUT_CTRL Domain13 read permission control for master1 0: The read-operation is available 1: The read-operation is unavailable
17	R/W	0x0	DM13_M0_WT_AUT_CTRL Domain13 write permission control for master0 0: The write-operation is available 1: The write-operation is unavailable
16	R/W	0x0	DM13_M0_RD_AUT_CTRL Domain13 read permission control for master0 0: The read-operation is available 1: The read-operation is unavailable
15:14	/	/	/
13	R/W	0x0	DM12_M6_WT_AUT_CTRL Domain12 write permission control for master6 0: The write-operation is available 1: The write-operation is unavailable
12	R/W	0x0	DM12_M6_RD_AUT_CTRL Domain12 read permission control for master6 0: The read-operation is available 1: The read-operation is unavailable
11	R/W	0x0	DM12_M5_WT_AUT_CTRL Domain12 write permission control for master5 0: The write-operation is available 1: The write-operation is unavailable
10	R/W	0x0	DM12_M5_RD_AUT_CTRL Domain12 read permission control for master5 0: The read-operation is available

			1: The read-operation is unavailable
9	R/W	0x0	DM12_M4_WT_AUT_CTRL Domain12 write permission control for master4 0: The write-operation is available 1: The write-operation is unavailable
8	R/W	0x0	DM12_M4_RD_AUT_CTRL Domain12 read permission control for master4 0: The read-operation is available 1: The read-operation is unavailable
7	R/W	0x0	DM12_M3_WT_AUT_CTRL Domain12 write permission control for master3 0: The write-operation is available 1: The write-operation is unavailable
6	R/W	0x0	DM12_M3_RD_AUT_CTRL Domain12 read permission control for master3 0: The read-operation is available 1: The read-operation is unavailable
5	R/W	0x0	DM12_M2_WT_AUT_CTRL Domain12 write permission control for master2 0: The write-operation is available 1: The write-operation is unavailable
4	R/W	0x0	DM12_M2_RD_AUT_CTRL Domain12 read permission control for master2 0: The read-operation is available 1: The read-operation is unavailable
3	R/W	0x0	DM12_M1_WT_AUT_CTRL Domain12 write permission control for master1 0: The write-operation is available 1: The write-operation is unavailable
2	R/W	0x0	DM12_M1_RD_AUT_CTRL Domain12 read permission control for master1 0: The read-operation is available 1: The read-operation is unavailable
1	R/W	0x0	DM12_M0_WT_AUT_CTRL Domain12 write permission control for master0 0: The write-operation is available 1: The write-operation is unavailable
0	R/W	0x0	DM12_M0_RD_AUT_CTRL Domain12 read permission control for master0 0: The read-operation is available 1: The read-operation is unavailable

3.6.6.24 IOMMU Domain Authority Control Register 7 (Default Value: 0x0000_0000)

Offset: 0x00CC	Register Name: IOMMU_DM_AUT_CTRL_REG7

Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29	R/W	0x0	DM15_M6_WT_AUT_CTRL Domain15 write permission control for master6 0: The write-operation is available 1: The write-operation is unavailable
28	R/W	0x0	DM15_M6_RD_AUT_CTRL Domain15 read permission control for master6 0: The read-operation is available 1: The read-operation is unavailable
27	R/W	0x0	DM15_M5_WT_AUT_CTRL Domain15 write permission control for master5 0: The write-operation is available 1: The write-operation is unavailable
26	R/W	0x0	DM15_M5_RD_AUT_CTRL Domain15 read permission control for master5 0: The read-operation is available 1: The read-operation is unavailable
25	R/W	0x0	DM15_M4_WT_AUT_CTRL Domain15 write permission control for master4 0: The write-operation is available 1: The write-operation is unavailable
24	R/W	0x0	DM15_M4_RD_AUT_CTRL Domain15 read permission control for master4 0: The read-operation is available 1: The read-operation is unavailable
23	R/W	0x0	DM15_M3_WT_AUT_CTRL Domain15 write permission control for master3 0: The write-operation is available 1: The write-operation is unavailable
22	R/W	0x0	DM15_M3_RD_AUT_CTRL Domain15 read permission control for master3 0: The read-operation is available 1: The read-operation is unavailable
21	R/W	0x0	DM15_M2_WT_AUT_CTRL Domain15 write permission control for master2 0: The write-operation is available 1: The write-operation is unavailable
20	R/W	0x0	DM15_M2_RD_AUT_CTRL Domain15 read permission control for master2 0: The read-operation is available 1: The read-operation is unavailable
19	R/W	0x0	DM15_M1_WT_AUT_CTRL Domain15 write permission control for master1 0: The write-operation is available 1: The write-operation is unavailable

18	R/W	0x0	DM15_M1_RD_AUT_CTRL Domain15 read permission control for master1 0: The read-operation is available 1: The read-operation is unavailable
17	R/W	0x0	DM15_M0_WT_AUT_CTRL Domain15 write permission control for master0 0: The write-operation is available 1: The write-operation is unavailable
16	R/W	0x0	DM15_M0_RD_AUT_CTRL Domain15 read permission control for master0 0: The read-operation is available 1: The read-operation is unavailable
15:14	/	/	/
13	R/W	0x0	DM14_M6_WT_AUT_CTRL Domain14 write permission control for master6 0: The write-operation is available 1: The write-operation is unavailable
12	R/W	0x0	DM14_M6_RD_AUT_CTRL Domain14 read permission control for master6 0: The read-operation is available 1: The read-operation is unavailable
11	R/W	0x0	DM14_M5_WT_AUT_CTRL Domain14 write permission control for master5 0: The write-operation is available 1: The write-operation is unavailable
10	R/W	0x0	DM14_M5_RD_AUT_CTRL Domain14 read permission control for master5 0: The read-operation is available 1: The read-operation is unavailable
9	R/W	0x0	DM14_M4_WT_AUT_CTRL Domain14 write permission control for master4 0: The write-operation is available 1: The write-operation is unavailable
8	R/W	0x0	DM14_M4_RD_AUT_CTRL Domain14 read permission control for master4 0: The read-operation is available 1: The read-operation is unavailable
7	R/W	0x0	DM14_M3_WT_AUT_CTRL Domain14 write permission control for master3 0: The write-operation is available 1: The write-operation is unavailable
6	R/W	0x0	DM14_M3_RD_AUT_CTRL Domain14 read permission control for master3 0: The read-operation is available 1: The read-operation is unavailable
5	R/W	0x0	DM14_M2_WT_AUT_CTRL

			Domain14 write permission control for master2 0: The write-operation is available 1: The write-operation is unavailable
4	R/W	0x0	DM14_M2_RD_AUT_CTRL Domain14 read permission control for master2 0: The read-operation is available 1: The read-operation is unavailable
3	R/W	0x0	DM14_M1_WT_AUT_CTRL Domain14 write permission control for master1 0: The write-operation is available 1: The write-operation is unavailable
2	R/W	0x0	DM14_M1_RD_AUT_CTRL Domain14 read permission control for master1 0: The read-operation is available 1: The read-operation is unavailable
1	R/W	0x0	DM14_M0_WT_AUT_CTRL Domain14 write permission control for master0 0: The write-operation is available 1: The write-operation is unavailable
0	R/W	0x0	DM14_M0_RD_AUT_CTRL Domain14 read permission control for master0 0: The read-operation is available 1: The read-operation is unavailable

3.6.6.25 IOMMU Domain Authority Overwrite Register (Default Value: 0x0000_0000)

Offset: 0x00D0			**Register Name: IOMMU_DM_AUT_OVWT_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	DM_AUT_OVWT_ENABLE Domain write/read permission overwrite enable 0: Disable 1: Enable
30:14	/	/	/
13	R/W	0x0	M6_WT_AUT_OVWT_CTRL Master6 write permission overwrite control 0: The write-operation is available 1: The write-operation is unavailable
12	R/W	0x0	M6_RD_AUT_OVWT_CTRL Master6 read permission overwrite control 0: The read-operation is available 1: The read-operation is unavailable
11	R/W	0x0	M5_WT_AUT_OVWT_CTRL Master5 write permission overwrite control 0: The write-operation is available 1: The write-operation is unavailable

10	R/W	0x0	M5_RD_AUT_OVWT_CTRL Master5 read permission overwrite control 0: The read-operation is available 1: The read-operation is unavailable
9	R/W	0x0	M4_WT_AUT_OVWT_CTRL Master5 write permission overwrite control 0: The write-operation is available 1: The write-operation is unavailable
8	R/W	0x0	M4_RD_AUT_OVWT_CTRL Master5 read permission overwrite control 0: The read-operation is available 1: The read-operation is unavailable
7	R/W	0x0	M3_WT_AUT_OVWT_CTRL Master3 write permission overwrite control 0: The write-operation is available 1: The write-operation is unavailable
6	R/W	0x0	M3_RD_AUT_OVWT_CTRL Master3 read permission overwrite control 0: The read-operation is available 1: The read-operation is unavailable
5	R/W	0x0	M2_WT_AUT_OVWT_CTRL Master2 write permission overwrite control 0: The write-operation is available 1: The write-operation is unavailable
4	R/W	0x0	M2_RD_AUT_OVWT_CTRL Master2 read permission overwrite control 0: The read-operation is available 1: The read-operation is unavailable
3	R/W	0x0	M1_WT_AUT_OVWT_CTRL Master1 write permission overwrite control 0: The write-operation is available 1: The write-operation is unavailable
2	R/W	0x0	M1_RD_AUT_OVWT_CTRL Master1 read permission overwrite control 0: The read-operation is available 1: The read-operation is unavailable
1	R/W	0x0	M0_WT_AUT_OVWT_CTRL Master0 write permission overwrite control 0: The write-operation is available 1: The write-operation is unavailable
0	R/W	0x0	M0_RD_AUT_OVWT_CTRL Master0 read permission overwrite control 0: The read-operation is available 1: The read-operation is unavailable

Setting the **REG_ARD_OVWT** can mask the Domain control defined by **IOMMU_DM_AUT_CTRL_REG0~7.** All the property of level 2 are covered by property defined by **REG_ARD_OVWT.** Allow read and write for all by default.

3.6.6.26 IOMMU Interrupt Enable Register (Default Value: 0x0000_0000)

Offset: 0x0100			Register Name: IOMMU_INT_ENABLE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17	R/W	0x0	L2_PAGE_TABLE_INVALID_EN Level2 page table invalid interrupt enable 0: Mask interrupt 1: Enable interrupt
16	R/W	0x0	L1_PAGE_TABLE_INVALID_EN Level1 page table invalid interrupt enable 0: Mask interrupt 1: Enable interrupt
15:7	/	/	/
6	R/W	0x0	MICRO_TLB6_INVALID_EN Micro TLB6 permission invalid interrupt enable 0: Mask interrupt 1: Enable interrupt
5	R/W	0x0	MICRO_TLB5_INVALID_EN Micro TLB5 permission invalid interrupt enable 0: Mask interrupt 1: Enable interrupt
4	R/W	0x0	MICRO_TLB4_INVALID_EN Micro TLB4 permission invalid interrupt enable 0: Mask interrupt 1: Enable interrupt
3	R/W	0x0	MICRO_TLB3_INVALID_EN Micro TLB3 permission invalid interrupt enable 0: Mask interrupt 1: Enable interrupt
2	R/W	0x0	MICRO_TLB2_INVALID_EN Micro TLB2 permission invalid interrupt enable 0: Mask interrupt 1: Enable interrupt
1	R/W	0x0	MICRO_TLB1_INVALID_EN Micro TLB1 permission invalid interrupt enable 0: Mask interrupt 1: Enable interrupt
0	R/W	0x0	MICRO_TLB0_INVALID_EN Micro TLB0 permission invalid interrupt enable 0: Mask interrupt 1: Enable interrupt

NOTE

Invalid page table and permission error can not make one device or multi-devices in system work normally.

Permission error usually happens in MicroTLB. The error generates interrupt, and waits for processing through

software.

Invalid page table usually happens in MacroTLB.The error can not influence the access of other devices.So the error page table need go back the way it comes, but the error should not be written in each level TLB.

3.6.6.27 IOMMU Interrupt Clear Register (Default Value: 0x0000_0000)

Offset: 0x0104			Register Name: IOMMU_INT_CLR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17	W	0x0	L2_PAGE_TABLE_INVALID_CLR Level2 page table invalid interrupt clear bit 0: Invalid operation 1: Clear interrupt
16	W	0x0	L1_PAGE_TABLE_INVALID_CLR Level1 page table invalid interrupt clear bit 0: Invalid operation 1: Clear interrupt
15:7	/	/	/
6	W	0x0	MICRO_TLB6_INVALID_CLR Micro TLB6 permission invalid interrupt clear bit 0: Invalid operation 1: Clear interrupt
5	W	0x0	MICRO_TLB5_INVALID_CLR Micro TLB5 permission invalid interrupt clear bit 0: Invalid operation 1: Clear interrupt
4	W	0x0	MICRO_TLB4_INVALID_CLR Micro TLB4 permission invalid interrupt clear bit 0: Invalid operation 1: Clear interrupt
3	W	0x0	MICRO_TLB3_INVALID_CLR Micro TLB3 permission invalid interrupt clear bit 0: Invalid operation 1: Clear interrupt
2	W	0x0	MICRO_TLB2_INVALID_CLR Micro TLB2 permission invalid interrupt clear bit 0: Invalid operation 1: Clear interrupt
1	W	0x0	MICRO_TLB1_INVALID_CLR Micro TLB1 permission invalid interrupt clear bit 0: Invalid operation 1: Clear interrupt
0	W	0x0	MICRO_TLB0_INVALID_CLR Micro TLB0 permission invalid interrupt clear bit 0: Invalid operation

			1: Clear interrupt

3.6.6.28 IOMMU Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0108			Register Name: IOMMU_INT_STA_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17	R	0x0	L2_PAGE_TABLE_INVALID_STA Level2 page table invalid interrupt status bit 0: Interrupt does not happen or interrupt is cleared 1: Interrupt happens
16	R	0x0	L1_PAGE_TABLE_INVALID_STA Level1 page table invalid interrupt status bit 0: Interrupt does not happen or interrupt is cleared 1: Interrupt happens
15:7	/	/	/
6	R	0x0	MICRO_TLB6_INVALID_STA Micro TLB6 permission invalid interrupt status bit 0: Interrupt does not happen or interrupt is cleared 1: Interrupt happens
5	R	0x0	MICRO_TLB5_INVALID_STA Micro TLB5 permission invalid interrupt status bit 0: Interrupt does not happen or interrupt is cleared 1: Interrupt happens
4	R	0x0	MICRO_TLB4_INVALID_STA Micro TLB4 permission invalid interrupt status bit 0: Interrupt does not happen or interrupt is cleared 1: Interrupt happens
3	R	0x0	MICRO_TLB3_INVALID_STA Micro TLB3 permission invalid interrupt status bit 0: Interrupt does not happen or interrupt is cleared 1: Interrupt happens
2	R	0x0	MICRO_TLB2_INVALID_STA Micro TLB2 permission invalid interrupt status bit 0: Interrupt does not happen or interrupt is cleared 1: Interrupt happens
1	R	0x0	MICRO_TLB1_INVALID_STA Micro TLB1 permission invalid interrupt status bit 0: Interrupt does not happen or interrupt is cleared 1: Interrupt happens
0	R	0x0	MICRO_TLB0_INVALID_STA Micro TLB0 permission invalid interrupt status bit 0: Interrupt does not happen or interrupt is cleared 1: Interrupt happens

3.6.6.29 IOMMU Interrupt Error Address Register 0 (Default Value: 0x0000_0000)

Offset: 0x0110			Register Name: IOMMU_INT_ERR_ADDR_REG0
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	INT_ERR_ADDR0 Virtual address causing Micro TLB0 occurs interrupt

3.6.6.30 IOMMU Interrupt Error Address Register 1 (Default Value: 0x0000_0000)

Offset: 0x0114			Register Name: IOMMU_INT_ERR_ADDR_REG1
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	INT_ERR_ADDR1 Virtual address causing Micro TLB1 occurs interrupt

3.6.6.31 IOMMU Interrupt Error Address Register 2 (Default Value: 0x0000_0000)

Offset: 0x0118			Register Name: IOMMU_INT_ERR_ADDR_REG2
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	INT_ERR_ADDR2 Virtual address causing Micro TLB2 occurs interrupt

3.6.6.32 IOMMU Interrupt Error Address Register 3 (Default Value: 0x0000_0000)

Offset: 0x011C			Register Name: IOMMU_INT_ERR_ADDR_REG3
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	INT_ERR_ADDR3 Virtual address causing Micro TLB3 occurs interrupt

3.6.6.33 IOMMU Interrupt Error Address Register 4 (Default Value: 0x0000_0000)

Offset: 0x0120			Register Name: IOMMU_INT_ERR_ADDR_REG4
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	INT_ERR_ADDR4 Virtual address causing Micro TLB4 occurs interrupt

3.6.6.34 IOMMU Interrupt Error Address Register 5 (Default Value: 0x0000_0000)

Offset: 0x0124			Register Name: IOMMU_INT_ERR_ADDR_REG5
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	INT_ERR_ADDR5 Virtual address causing Micro TLB5 occurs interrupt

3.6.6.35 IOMMU Interrupt Error Address Register 6 (Default Value: 0x0000_0000)

Offset: 0x0128			Register Name: IOMMU_INT_ERR_ADDR_REG6
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	INT_ERR_ADDR6 Virtual address causing Micro TLB6 occurs interrupt

3.6.6.36 IOMMU Interrupt Error Address Register 7 (Default Value: 0x0000_0000)

Offset: 0x0130			Register Name: IOMMU_INT_ERR_ADDR_REG7
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	INT_ERR_ADDR7 Virtual address causing L1 page table occurs interrupt

3.6.6.37 IOMMU Interrupt Error Address Register 8 (Default Value: 0x0000_0000)

Offset: 0x0134			Register Name: IOMMU_INT_ERR_ADDR_REG8
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	INT_ERR_ADDR8 Virtual address causing L2 page table occurs interrupt

3.6.6.38 IOMMU Interrupt Error Data Register 0 (Default Value: 0x0000_0000)

Offset: 0x0150			Register Name: IOMMU_INT_ERR_DATA_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	INT_ERR_DATA0 Corresponding page table of virtual address causing Micro TLB0 occurs interrupt

3.6.6.39 IOMMU Interrupt Error Data Register 1 (Default Value: 0x0000_0000)

Offset: 0x0154			Register Name: IOMMU_INT_ERR_DATA_REG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	INT_ERR_DATA1 Corresponding page table of virtual address causing Micro TLB1 occurs interrupt

3.6.6.40 IOMMU Interrupt Error Data Register 2 (Default Value: 0x0000_0000)

Offset: 0x0158			Register Name: IOMMU_INT_ERR_DATA_REG2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	INT_ERR_DATA2 Corresponding page table of virtual address causing Micro TLB2 occurs interrupt

3.6.6.41 IOMMU Interrupt Error Data Register 3 (Default Value: 0x0000_0000)

Offset: 0x015C			Register Name: IOMMU_INT_ERR_DATA_REG3
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	INT_ERR_DATA3 Corresponding page table of virtual address causing Micro TLB3 occurs interrupt

3.6.6.42 IOMMU Interrupt Error Data Register 4 (Default Value: 0x0000_0000)

Offset: 0x0160			Register Name: IOMMU_INT_ERR_DATA_REG4
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	INT_ERR_DATA4 Corresponding page table of virtual address causing Micro TLB4 occurs interrupt

3.6.6.43 IOMMU Interrupt Error Data Register 5 (Default Value: 0x0000_0000)

Offset: 0x0164			Register Name: IOMMU_INT_ERR_DATA_REG5
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	INT_ERR_DATA5 Corresponding page table of virtual address causing Micro TLB5 occurs interrupt

3.6.6.44 IOMMU Interrupt Error Data Register 6 (Default Value: 0x0000_0000)

Offset: 0x0168			Register Name: IOMMU_INT_ERR_DATA_REG6
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	INT_ERR_DATA6 Corresponding page table of virtual address causing Micro TLB6 occurs interrupt

3.6.6.45 IOMMU Interrupt Error Data Register 7 (Default Value: 0x0000_0000)

Offset: 0x0170			Register Name: IOMMU_INT_ERR_DATA_REG7
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	INT_ERR_DATA7 Corresponding page table of virtual address causing L1 page table occurs interrupt

3.6.6.46 IOMMU Interrupt Error Data Register 8 (Default Value: 0x0000_0000)

Offset: 0x0174			Register Name: IOMMU_INT_ERR_DATA_REG8
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	INT_ERR_DATA8 Corresponding page table of virtual address causing L2 page table occurs interrupt

3.6.6.47 IOMMU L1 Page Table Interrupt Register (Default Value: 0x0000_0000)

Offset: 0x0180			Register Name: IOMMU_L1PG_INT_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R	0x0	DBG_MODE_L1PG_INT Debug Mode address switch causes L1 page table to occur interrupt.
30:7	/	/	/
6	R	0x0	MASTER6_L1PG_INT Master6 address switch causes L1 page table to occur interrupt.
5	R	0x0	MASTER5_L1PG_INT Master5 address switch causes L1 page table to occur interrupt.
4	R	0x0	MASTER4_L1PG_INT Master4 address switch causes L1 page table to occur interrupt.
3	R	0x0	MASTER3_L1PG_INT Master3 address switch causes L1 page table to occur interrupt.
2	R	0x0	MASTER2_L1PG_INT Master2 address switch causes L1 page table to occur interrupt.
1	R	0x0	MASTER1_L1PG_INT Master1 address switch causes L1 page table to occur interrupt.
0	R	0x0	MASTER0_L1PG_INT Master0 address switch causes L1 page table to occur interrupt.

3.6.6.48 IOMMU L2 Page Table Interrupt Register (Default Value: 0x0000_0000)

Offset: 0x0184			Register Name: IOMMU_L2PG_INT_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R	0x0	DBG_MODE_L2PG_INT

			Debug Mode address switch causes L2 page table to occur interrupt.
30:7	/	/	/
6	R	0x0	MASTER6_L2PG_INT Master6 address switch causes L2 page table to occur interrupt.
5	R	0x0	MASTER5_L2PG_INT Master5 address switch causes L2 page table to occur interrupt.
4	R	0x0	MASTER4_L2PG_INT Master4 address switch causes L2 page table to occur interrupt.
3	R	0x0	MASTER3_L2PG_INT Master3 address switch causes L2 page table to occur interrupt.
2	R	0x0	MASTER2_L2PG_INT Master2 address switch causes L2 page table to occur interrupt.
1	R	0x0	MASTER1_L2PG_INT Master1 address switch causes L2 page table to occur interrupt.
0	R	0x0	MASTER0_L2PG_INT Master0 address switch causes L2 page table to occur interrupt.

3.6.6.49 IOMMU Virtual Address Register (Default Value: 0x0000_0000)

Offset: 0x0190			Register Name: IOMMU_VA _REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	VA Virtual address of read/write

3.6.6.50 IOMMU Virtual Address Data Register (Default Value: 0x0000_0000)

Offset: 0x0194			Register Name: IOMMU_VA_DATA _REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	VA_DATA Data of read/write virtual address

3.6.6.51 IOMMU Virtual Address Configuration Register (Default Value: 0x0000_0000)

Offset: 0x0198			Register Name: IOMMU_VA_CONFIG _REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	MASTER7_MODE 0: Prefetch 1: Debug Mode It is used to chose prefetch mode or Debug mode.
31:9	/	/	/
8	R/W	0x0	VA_CONFIG 0: Read operation 1: Write operation

7:1	/	/	/
0	R/WAC	0x0	VA_CONFIG_START 0: No operation or operation completes 1: Start After the operation completes, the bit can clear to 0 automatically.

Read operation process:

a) Write IOMMU_VA_REG[31:0];

b) Write IOMMU_VA_CONFIG _REG[8] to 0;

c) Write IOMMU_VA_CONFIG _REG[0] to 1,start read-process;

d) Query IOMMU_VA_CONFIG _REG[0], until it is 0;

e) Read IOMMU_VA_DATA _REG[31:0]

Write operation process:

a) Write IOMMU_VA _REG[31:0];

b) Write IOMMU_VA_DATA _REG[31:0];

c) Write IOMMU_VA_CONFIG _REG[8] to 1;

d) Write IOMMU_VA_CONFIG _REG[0] to 1， start write-process;

e) QueryIOMMU_VA_CONFIG _REG[0],until it is 0

3.6.6.52 IOMMU PMU Enable Register (Default Value: 0x0000_0000)

Offset: 0x0200			Register Name: IOMMU_PMU_ENABLE _REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	PMU_ENABLE 0: Disable statistical function 1: Enable statistical function

3.6.6.53 IOMMU PMU Clear Register (Default Value: 0x0000_0000)

Offset: 0x0210			Register Name: IOMMU_PMU_CLR _REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/WAC	0x0	PMU_CLR 0: No clear operation or clear operation completes 1: Clear counter data After the operation completes, the bit can clear to 0 automatically.

3.6.6.54 IOMMU PMU Access Low Register 0 (Default Value: 0x0000_0000)

Offset: 0x0230			Register Name: IOMMU_PMU_ACCESS_LOW_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_ACCESS_LOW0

			Record total number of Micro TLB0 access , lower 32-bit register

3.6.6.55 IOMMU PMU Access High Register 0 (Default Value: 0x0000_0000)

Offset: 0x0234			Register Name: IOMMU_PMU_ACCESS_HIGH_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R	0x0	PMU_ACCESS_HIGH0 Record total number of Micro TLB0 access , higher 11-bit register

3.6.6.56 IOMMU PMU Hit Low Register 0 (Default Value: 0x0000_0000)

Offset: 0x0238			Register Name: IOMMU_PMU_HIT_LOW_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_HIT_LOW0 Record total number of Micro TLB0 hit , lower 32-bit register

3.6.6.57 IOMMU PMU Hit High Register 0 (Default Value: 0x0000_0000)

Offset: 0x023C			Register Name: IOMMU_PMU_HIT_HIGH_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R	0x0	PMU_HIT_HIGH0 Record total number of Micro TLB0 hit , higher 11-bit register

3.6.6.58 IOMMU PMU Access Low Register 1 (Default Value: 0x0000_0000)

Offset: 0x0240			Register Name: IOMMU_PMU_ACCESS_LOW_REG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_ACCESS_LOW1 Record total number of Micro TLB1 access , lower 32-bit register

3.6.6.59 IOMMU PMU Access High Register 1 (Default Value: 0x0000_0000)

Offset: 0x0244			Register Name: IOMMU_PMU_ACCESS_HIGH_REG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R	0x0	PMU_ACCESS_HIGH1 Record total number of Micro TLB1 access , higher 11-bit register

3.6.6.60 IOMMU PMU Hit Low Register 1 (Default Value: 0x0000_0000)

Offset: 0x0248			Register Name: IOMMU_PMU_HIT_LOW_REG1
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	PMU_HIT_LOW1 Record total number of Micro TLB1 hit , lower 32-bit register

3.6.6.61 IOMMU PMU Hit High Register 1 (Default Value: 0x0000_0000)

Offset: 0x024C			Register Name: IOMMU_PMU_HIT_HIGH_REG1
Bit	Read/Write	Default/Hex	Description
31:11	/	/	/
10:0	R	0x0	PMU_HIT_HIGH1 Record total number of Micro TLB1 hit , higher 11-bit register

3.6.6.62 IOMMU PMU Access Low Register 2 (Default Value: 0x0000_0000)

Offset: 0x0250			Register Name: IOMMU_PMU_ACCESS_LOW_REG2
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	PMU_ACCESS_LOW2 Record total number of Micro TLB2 access , lower 32-bit register

3.6.6.63 IOMMU PMU Access High Register 2 (Default Value: 0x0000_0000)

Offset: 0x0254			Register Name: IOMMU_PMU_ACCESS_HIGH_REG2
Bit	Read/Write	Default/Hex	Description
31:11	/	/	/
10:0	R	0x0	PMU_ACCESS_HIGH2 Record total number of Micro TLB2 access , higher 11-bit register

3.6.6.64 IOMMU PMU Hit Low Register 2 (Default Value: 0x0000_0000)

Offset: 0x0258			Register Name: IOMMU_PMU_HIT_LOW_REG2
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	PMU_HIT_LOW2 Record total number of Micro TLB2 hit , lower 32-bit register

3.6.6.65 IOMMU PMU Hit High Register 2 (Default Value: 0x0000_0000)

Offset: 0x025C			Register Name: IOMMU_PMU_HIT_HIGH_REG2
Bit	Read/Write	Default/Hex	Description

31:11	/	/	/
10:0	R	0x0	PMU_HIT_HIGH2 Record total number of Micro TLB2 hit , higher 11-bit register

3.6.6.66 IOMMU PMU Access Low Register 3 (Default Value: 0x0000_0000)

Offset: 0x0260			Register Name: IOMMU_PMU_ACCESS_LOW_REG3
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_ACCESS_LOW3 Record total number of Micro TLB3 access, lower 32-bit register

3.6.6.67 IOMMU PMU Access High Register 3 (Default Value: 0x0000_0000)

Offset: 0x0264			Register Name: IOMMU_PMU_ACCESS_HIGH_REG3
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R	0x0	PMU_ACCESS_HIGH3 Record total number of Micro TLB3 access , higher 11-bit register

3.6.6.68 IOMMU PMU Hit Low Register 3 (Default Value: 0x0000_0000)

Offset: 0x0268			Register Name: IOMMU_PMU_HIT_LOW_REG3
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_HIT_LOW3 Record total number of Micro TLB3 hit, lower 32-bit register

3.6.6.69 IOMMU PMU Hit High Register 3 (Default Value: 0x0000_0000)

Offset: 0x026C			Register Name: IOMMU_PMU_HIT_HIGH_REG3
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R	0x0	PMU_HIT_HIGH3 Record total number of Micro TLB3 hit , higher 11-bit register

3.6.6.70 IOMMU PMU Access Low Register 4 (Default Value: 0x0000_0000)

Offset: 0x0270			Register Name: IOMMU_PMU_ACCESS_LOW_REG4
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_ACCESS_LOW4 Record total number of Micro TLB4 access, lower 32-bit register

3.6.6.71 IOMMU PMU Access High Register 4 (Default Value: 0x0000_0000)

Offset: 0x0274			Register Name: IOMMU_PMU_ACCESS_HIGH_REG4
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R	0x0	PMU_ACCESS_HIGH4 Record total number of Micro TLB4 access, higher 11-bit register

3.6.6.72 IOMMU PMU Hit Low Register 4 (Default Value: 0x0000_0000)

Offset: 0x0278			Register Name: IOMMU_PMU_HIT_LOW_REG4
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_HIT_LOW4 Record total number of Micro TLB4 hit, lower 32-bit register

3.6.6.73 IOMMU PMU Hit High Register 4 (Default Value: 0x0000_0000)

Offset: 0x027C			Register Name: IOMMU_PMU_HIT_HIGH_REG4
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R	0x0	PMU_HIT_HIGH4 Record total number of Micro TLB4 hit, higher 11-bit register

3.6.6.74 IOMMU PMU Access Low Register 5 (Default Value: 0x0000_0000)

Offset: 0x0280			Register Name: IOMMU_PMU_ACCESS_LOW_REG5
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_ACCESS_LOW5 Record total number of Micro TLB5 access, lower 32-bit register

3.6.6.75 IOMMU PMU Access High Register 5 (Default Value: 0x0000_0000)

Offset: 0x0284			Register Name: IOMMU_PMU_ACCESS_HIGH_REG5
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R	0x0	PMU_ACCESS_HIGH5 Record total number of Micro TLB5 access, higher 11-bit register

3.6.6.76 IOMMU PMU Hit Low Register 5 (Default Value: 0x0000_0000)

Offset: 0x0288	Register Name: IOMMU_PMU_HIT_LOW_REG5

Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	PMU_HIT_LOW5 Record total number of Micro TLB5 hit, lower 32-bit register

3.6.6.77 IOMMU PMU Hit High Register 5 (Default Value: 0x0000_0000)

Offset: 0x028C			Register Name: IOMMU_PMU_HIT_HIGH_REG5
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R	0x0	PMU_HIT_HIGH5 Record total number of Micro TLB5 hit, higher 11-bit register

3.6.6.78 IOMMU PMU Access Low Register6 (Default Value: 0x0000_0000)

Offset: 0x0290			Register Name: IOMMU_PMU_ACCESS_LOW_REG6
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_ACCESS_LOW6 Record total number of Micro TLB6 access, lower 32-bit register

3.6.6.79 IOMMU PMU Access High Register 6 (Default Value: 0x0000_0000)

Offset: 0x0294			Register Name: IOMMU_PMU_ACCESS_HIGH_REG6
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R	0x0	PMU_ACCESS_HIGH6 Record total number of Micro TLB6 access, higher 11-bit register

3.6.6.80 IOMMU PMU Hit Low Register 6 (Default Value: 0x0000_0000)

Offset: 0x0298			Register Name: IOMMU_PMU_HIT_LOW_REG6
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_HIT_LOW6 Record total number of Micro TLB6 hit, lower 32-bit register

3.6.6.81 IOMMU PMU Hit High Register 6 (Default Value: 0x0000_0000)

Offset: 0x029C			Register Name: IOMMU_PMU_HIT_HIGH_REG6
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R	0x0	PMU_HIT_HIGH6 Record total number of Micro TLB6 hit, higher 11-bit register

3.6.6.82 IOMMU PMU Access Low Register 7 (Default Value: 0x0000_0000)

Offset: 0x02D0			Register Name: IOMMU_PMU_ACCESS_LOW_REG7
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_ACCESS_LOW7 Record total number of Micro TLB7 access, lower 32-bit register

3.6.6.83 IOMMU PMU Access High Register 7 (Default Value: 0x0000_0000)

Offset: 0x02D4			Register Name: IOMMU_PMU_ACCESS_HIGH_REG7
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R	0x0	PMU_ACCESS_HIGH7 Record total number of Micro TLB7 access, higher 11-bit register

3.6.6.84 IOMMU PMU Hit Low Register 7 (Default Value: 0x0000_0000)

Offset: 0x02D8			Register Name: IOMMU_PMU_HIT_LOW_REG7
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_HIT_LOW7 Record total number of Micro TLB7 hit, lower 32-bit register

3.6.6.85 IOMMU PMU Hit High Register 7 (Default Value: 0x0000_0000)

Offset: 0x02DC			Register Name: IOMMU_PMU_HIT_HIGH_REG7
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R	0x0	PMU_HIT_HIGH7 Record total number of Micro TLB7 hit, higher 11-bit register

3.6.6.86 IOMMU PMU Access Low Register 8 (Default Value: 0x0000_0000)

Offset: 0x02E0			Register Name: IOMMU_PMU_ACCESS_LOW_REG8
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_ACCESS_LOW8 Record total number of PTW Cache access, lower 32-bit register

3.6.6.87 IOMMU PMU Access High Register 8 (Default Value: 0x0000_0000)

Offset: 0x02E4			Register Name: IOMMU_PMU_ACCESS_HIGH_REG8
Bit	Read/Write	Default/Hex	Description
31:11	/	/	/
10:0	R	0x0	PMU_ACCESS_HIGH8 Record total number of PTW Cache access, higher 11-bit register

3.6.6.88 IOMMU PMU Hit Low Register 8 (Default Value: 0x0000_0000)

Offset: 0x02E8			Register Name: IOMMU_PMU_HIT_LOW_REG8
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	PMU_HIT_LOW8 Record total number of PTW Cache hit, lower 32-bit register

3.6.6.89 IOMMU PMU Hit High Register 8 (Default Value: 0x0000_0000)

Offset: 0x02EC			Register Name: IOMMU_PMU_HIT_HIGH_REG8
Bit	Read/Write	Default/Hex	Description
31:11	/	/	/
10:0	R	0x0	PMU_HIT_HIGH8 Record total number of PTW Cache hit, higher 11-bit register

3.6.6.90 IOMMU Total Latency Low Register 0 (Default Value: 0x0000_0000)

Offset: 0x0300			Register Name: IOMMU_PMU_TL_LOW_REG0
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	PMU_TL_LOW0 Record total latency of Master0, lower 32-bit register

3.6.6.91 IOMMU Total Latency High Register 0 (Default Value: 0x0000_0000)

Offset: 0x0304			Register Name: IOMMU_PMU_TL_HIGH_REG0
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17:0	R	0x0	PMU_TL_HIGH0 Record total latency of Master0, higher 18-bit register

3.6.6.92 IOMMU Max Latency Register 0 (Default Value: 0x0000_0000)

Offset: 0x0308	Register Name: IOMMU_PMU_ML_REG0

Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	PMU_ ML0 Record the max latency of Master0.

3.6.6.93 IOMMU Total Latency Low Register 1(Default Value: 0x0000_0000)

Offset: 0x0310			Register Name: IOMMU_PMU_TL_LOW_REG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_TL_LOW1 Record total latency of Master1, lower 32-bit register

3.6.6.94 IOMMU Total Latency High Register 1 (Default Value: 0x0000_0000)

Offset: 0x0314			Register Name: IOMMU_PMU_TL_HIGH_REG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17:0	R	0x0	PMU_TL_HIGH1 Record total latency of Master1, higher 18-bit register

3.6.6.95 IOMMU Max Latency Register 1 (Default Value: 0x0000_0000)

Offset: 0x0318			Register Name: IOMMU_PMU_ML_REG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_ML1 Record the max latency of Master1.

3.6.6.96 IOMMU Total Latency Low Register 2 (Default Value: 0x0000_0000)

Offset: 0x0320			Register Name: IOMMU_PMU_TL_LOW_REG2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_TL_LOW2 Record total latency of Master2, lower 32-bit register

3.6.6.97 IOMMU Total Latency High Register 2 (Default Value: 0x0000_0000)

Offset: 0x0324			Register Name: IOMMU_PMU_TL_HIGH_REG2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17:0	R	0x0	PMU_TL_HIGH2 Record total latency of Master2, higher 18-bit register

3.6.6.98 IOMMU Max Latency Register 2 (Default Value: 0x0000_0000)

Offset: 0x0328			Register Name: IOMMU_PMU_ML_REG2
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	PMU_ ML2 Record the max latency of Master2.

3.6.6.99 IOMMU Total Latency Low Register 3 (Default Value: 0x0000_0000)

Offset: 0x0330			Register Name: IOMMU_PMU_TL_LOW_REG3
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	PMU_ TL_LOW3 Record total latency of Master3, lower 32-bit register

3.6.6.100 IOMMU Total Latency High Register 3 (Default Value: 0x0000_0000)

Offset: 0x0334			Register Name: IOMMU_PMU_TL_HIGH_REG3
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17:0	R	0x0	PMU_TL_HIGH3 Record total latency of Master3, higher 18-bit register

3.6.6.101 IOMMU Max Latency Register 3 (Default Value: 0x0000_0000)

Offset: 0x0338			Register Name: IOMMU_PMU_ML_REG3
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	PMU_ ML3 Record the max latency of Master3.

3.6.6.102 IOMMU Total Latency Low Register 4 (Default Value: 0x0000_0000)

Offset: 0x0340			Register Name: IOMMU_PMU_TL_LOW_REG4
Bit	Read/Write	Default/Hex	Description
31:0	R	0x0	PMU_TL_LOW4 Record total latency of Master4, lower 32-bit register

3.6.6.103 IOMMU Total Latency High Register 4 (Default Value: 0x0000_0000)

Offset: 0x0344			Register Name: IOMMU_PMU_TL_HIGH_REG4
Bit	Read/Write	Default/Hex	Description
31:18	/	/	/

17:0	R	0x0	PMU_TL_HIGH4 Record total latency of Master4, higher 18-bit register

3.6.6.104 IOMMU Max Latency Register 4 (Default Value: 0x0000_0000)

Offset: 0x0348			Register Name: IOMMU_PMU_ML_REG4
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_ ML4 Record the max latency of Master4.

3.6.6.105 IOMMU Total Latency Low Register 5 (Default Value: 0x0000_0000)

Offset: 0x0350			Register Name: IOMMU_PMU_TL_LOW_REG5
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_TL_LOW5 Record total latency of Master5, lower 32-bit register

3.6.6.106 IOMMU Total Latency High Register 5 (Default Value: 0x0000_0000)

Offset: 0x0354			Register Name: IOMMU_PMU_TL_HIGH_REG5
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17:0	R	0x0	PMU_TL_HIGH5 Record total latency of Master5, higher 18-bit register

3.6.6.107 IOMMU Max Latency Register 5 (Default Value: 0x0000_0000)

Offset: 0x0358			Register Name: IOMMU_PMU_ML_REG5
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_ ML5 Record the max latency of Master5.

3.6.6.108 IOMMU Total Latency Low Register 6 (Default Value: 0x0000_0000)

Offset: 0x0360			Register Name: IOMMU_PMU_TL_LOW_REG6
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_TL_LOW6 Record total latency of Master6, lower 32-bit register

3.6.6.109 IOMMU Total Latency High Register 6 (Default Value: 0x0000_0000)

Offset: 0x0364			Register Name: IOMMU_PMU_TL_HIGH_REG6
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17:0	R	0x0	PMU_TL_HIGH6 Record total latency of Master6, higher 18-bit register

3.6.6.110 IOMMU Max Latency Register 6 (Default Value: 0x0000_0000)

Offset: 0x0368			Register Name: IOMMU_PMU_ML_REG6
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PMU_ ML6 Record the max latency of Master6.

3.7 Timer

3.7.1 Overview

The timer module implements the timing and counting functions.The timer module includes timer0 and timer1,watchdog, AVS.

The timer0 and timer1 are completely consistent.The timer0/1 has the following features:

- Configurable count clock: LOSC and OSC24M. LOSC is the internal low-frequency clock or the external low-frequency clock by setting LOSC_SRC_SEL. The external low-frequency has much accuracy.
- Configurable 8 prescale factor
- Programmable 32-bit down timer
- Supports two working modes: continue mode and single count mode
- Generates an interrupt when the count is decreased to 0

The watchdog is used to transimit a reset signal to reset the entire system after an exception occurs in the system.The watchdog has the following features:

- Single clock sourece: HOSC_32K(OSC24M/750 , OSC19.2M/600, or OSC38.4M/1200, Oscillator chosing depend on PAD_SEL
- Supports 12 initial values to configure
- Supports the generation of timeout interrupts
- Supports the generation of reset signal
- Supports watchdog restart the timing

The AVS is used to the synchronization of audio and video. The AVS module includes AVS0 and AVS1, the AVS0 and AVS1 are completely consistent.The AVS has the following features:

- Single clock source: OSC24M
- Programmable 33-bit up timer
- Initial value can be updated anytime
- 12-bit frequency divider factor
- Supports Pause/Start function

3.7.2 Block Diagram

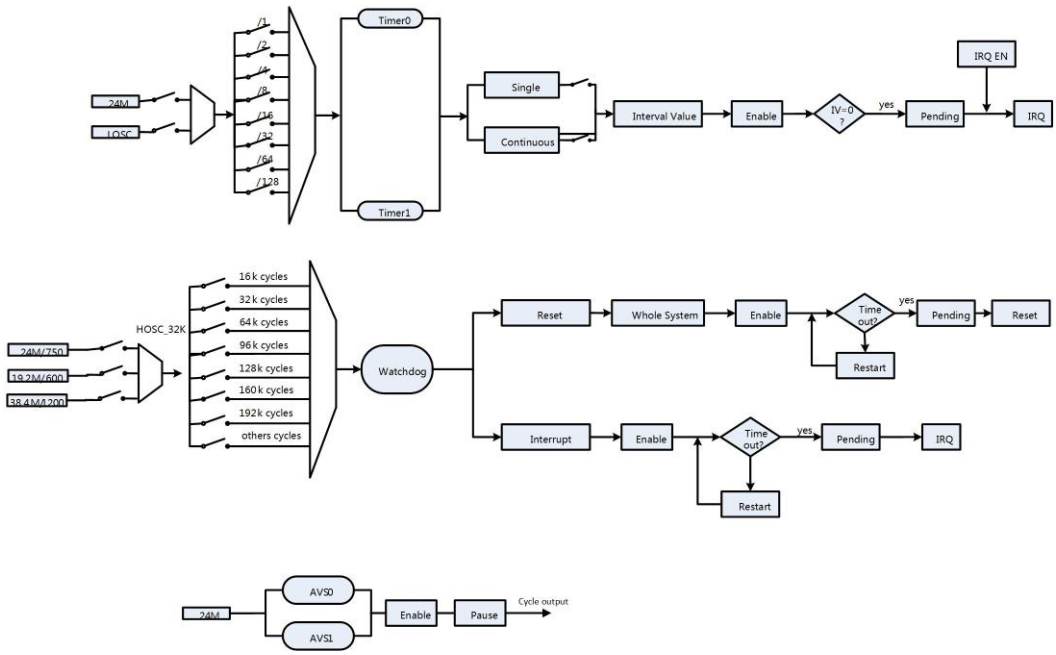


Figure3- 14. Timer Block Diagram

3.7.3 Operations and Functional Descriptions

3.7.3.1 Timer Formula

$$T_{timer0} = \frac{TMR0\_INTV\_VALUE\_REG - TMR0\_CUR\_VALUE\_REG}{TMR0\_CLK\_SRC} \times TMR0\_CLK\_PRES$$

TMR0_INTV_VALUE_REG: timer initial value;

TMR0_CUR_VALUE_REG: timer current counter;

TMR0_CLK_SRC: timer clock source;

TMR0_CLK_PRES: timer clock prescale ratio.

3.7.3.2 Typical Application

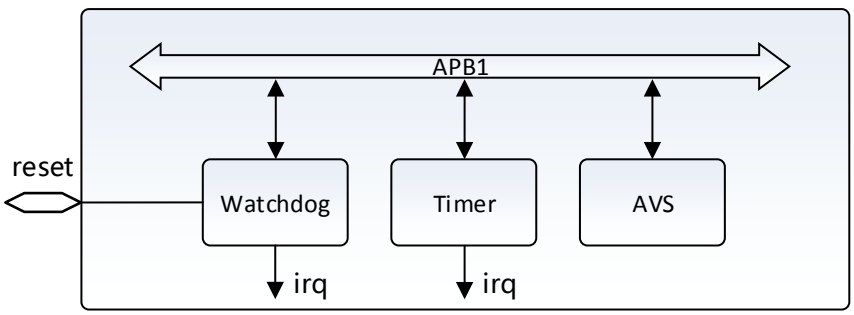


Figure3- 15. Timer Application Diagram

3.7.3.3 Function Implemention

3.7.3.3.1 Timer

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

- **Continuous mode**

The bit7 of TMRn_CTRL_REG is set to the continuous mode, when the count value is decreased to 0, the timer module reloads data from TMRn_INTV_VALUE_REG then continues to count.

- **Single mode**

The bit7 of TMRn_CTRL_REG is set to the single mode, when the count value is decreased to 0, the timer stops counting. The timer starts to count again only when a new initial value is loaded.
Each timer has a prescaler that divides the working clock frequency of each timer by 1,2,4,8,16,32,64,128.

3.7.3.3.2 Watchdog

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

- **Interrupt mode**

WDOG0_CFG_REG is set to 0x10, when the counter value reaches 0 and WDOG0_IRQ_EN_REG is enabled, the watchdog generates an interrupt.

- **Reset mode**

WDOG0_CFG_REG is set to 0x01, when the counter value reaches 0, the watchdog generates a reset signal to reset the entire system.
The clock source of the watchdog is OSC24M/750.There are 12 configurable initial count values.
The watchdog can restart to count by setting **WDOG0_CTRL_REG**: write 0xA57 to bit[12:1],then write 1 to bit[0].

3.7.3.3.3 AVS

The AVS is a 33-bit up counter.The counter value is increased by 1 on each rising edge of the count clock.
The AVS can be operated after its clock gating in CCU module is opened.
The AVS has an OSC24M clock source and a 12-bit division factor N. When the timer increases to N from 0, AVS counter adds 1; when the counter reaches 33-bit upper limit,the AVS will start to count from initial value again.
In counter working process, the division factor and initial counter of the AVS can be changed anytime.And the AVS can stop or start to operate counter anytime.

3.7.3.4 Operating Mode

3.7.3.4.1 Timer Initial

Configure the timer parameters: clock source, prescale factor, working mode. The configuration of these parameters have no sequence,and implement by writing **TMRn_CTRL_REG**.

Write the initial value: write **TMRn_INTV_VALUE_REG** to provide an initial value for the timer; write the bit[1] of **TMRn_CTRL_REG** to load the initial value to the timer, the bit[1] can not be written again before it is cleared automatically.

Enable timer: write the bit[0] of **TMRn_CTRL_REG** to enable timer count; read **TMRn_CUR_VALUE_REG** to get the current count value.

3.7.3.4.2 Timer Interrupt

Enable interrupt: write corresponding interrupt enable bit of **TMR_IRQ_EN_REG**, when timer counter time reaches, the corresponding interrupt generates.

After enter interrupt process, write **TMR_IRQ_STA_REG** to clear the interrupt pending, and execute the process of waiting for the interrupt.

Resume the interrupt and continue to execute the interrupted process.

3.7.3.4.3 Watchdog Initial

Write **WDOG0_CFG_REG** to configure the generation of the interrupts and the output of reset signal.

Write **WDOG0_MODE_REG** to configure the initial count value.

Write **WDOG0_MODE_REG** to enable the watchdog.

3.7.3.4.4 Watchdog Interrupt

Watchdog interrupt is only used for the counter.

Write **WDOG0_IRQ_EN_REG** to enable the interrupt.

After enter the interrupt process, write **WDOG0_IRQ_STA_REG** to clear the interrupt pending, and execute the process of waiting for the interrupt.

Resume the interrupt and continue to execute the interrupted process.

3.7.3.4.5 AVS Start/Pause

Write **AVS_CNT_DIV_REG** to configure the division factor.

Write **AVS_CNT_REG** to configure the initial count value.

Write **AVS_CNT_CTL_REG** to enable AVS counter. Supports Counter pause to count anytime.

3.7.4 Programming Guidelines

3.7.4.1 Timer

Take making a 1ms delay for an example, 24MHz clock source, single mode and 2 pre-scale will be selected in the instance.

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

3.7.4.2 Watchdog Reset

In the following instance making configurations for Watchdog: configurate clock source as 24M/750, configurate Interval Value as 1s and configurate Watchdog Configuration as To whole system. This instance indicates that reset system after 1s.

```
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
```

3.7.4.3 Watchdog Restart

In the following instance making configurations for Watchdog: configurate clock source as 24M/750, configurate *Interval Value* as 1s and configurate *Watchdog Configuration* as *To whole system*. In the following instance, if the time of other codes is larger than 1s, watchdog will reset the whole system. If the sentence of restart watchdog is implemented inside 1s, watchdog will be restarted.

```
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); //Writel 0xA57 at Key Field and Restart Watchdog
```

3.7.5 Register List

Module Name	Base Address
Timer	0x0300 9000

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	Timer 0 Control Register

TMR0_INTV_VALUE_REG	0x0014	Timer 0 Interval Value Register
TMR0_CUR_VALUE_REG	0x0018	Timer 0 Current Value Register
TMR1_CTRL_REG	0x0020	Timer 1 Control Register
TMR1_INTV_VALUE_REG	0x0024	Timer 1 Interval Value Register
TMR1_CUR_VALUE_REG	0x0028	Timer 1 Current Value Register
WDOG_IRQ_EN_REG	0x00A0	Watchdog IRQ Enable Register
WDOG_IRQ_STA_REG	0x00A4	Watchdog Status Register
WDOG_CTRL_REG	0x00B0	Watchdog Control Register
WDOG_CFG_REG	0x00B4	Watchdog Configuration Register
WDOG_MODE_REG	0x00B8	Watchdog Mode 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

3.7.6 Register Description

3.7.6.1 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/W1S	0x0	TMR1_IRQ_EN. Timer 1 Interrupt Enable. 0: No effect 1: Timer 1 Interval Value reached interrupt enable
0	R/W1S	0x0	TMR0_IRQ_EN. Timer 0 Interrupt Enable. 0: No effect 1: Timer 0 Interval Value reached interrupt enable

3.7.6.2 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. Timer 1 IRQ Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending, timer 1 interval value is reached
0	R/W1C	0x0	TMR0_IRQ_PEND. Timer 0 IRQ Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending, timer 0 interval value is reached

3.7.6.3 Timer 0 Control Register(Default Value: 0x0000_0004)

Offset: 0x0010			Register Name: TMR0_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R/W	0x0	TMR0_MODE. Timer 0 mode. 0: Continuous mode. When interval value reached, the timer will not disable automatically. 1: Single mode. When interval value reached, the timer will disable automatically.
6:4	R/W	0x0	TMR0_CLK_PRES. Select the pre-scale of timer 0 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. 00:LOSC 01: OSC24M 10: / 11: /
1	R/W	0x0	TMR0_RELOAD. Timer 0 Reload. 0: No effect 1: Reload timer 0 Interval value After the bit is set, it can not be written again before it is cleared automatically.
0	R/W	0x0	TMR0_EN. Timer 0 Enable. 0: Stop/Pause 1: Start If the timer is started, it will reload the interval value to internal register, and the current counter will count from interval value to 0. If the current counter does not reach the zero, the timer enable bit is set to “0”, the current value counter will pause. At least wait for 2 cycles, the start bit can be set to 1. In timer pause state, the interval value register can be modified. If the timer is started again, and the Software hope the current value register to down-count from the new interval value, the reload bit and the enable bit should be set to 1 at the same time.

3.7.6.4 Timer 0 Interval Value Register(Default Value: 0x0000_0000)

Offset: 0x0014			Register Name: TMR0_INTV_VALUE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	TMR0_INTV_VALUE. Timer 0 Interval Value. The value setting should consider the system clock and the timer clock source.

3.7.6.5 Timer 0 Current Value Register(Default Value: 0x0000_0000)

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

3.7.6.6 Timer 1 Control Register(Default Value: 0x0000_0004)

Offset: 0x0020			Register Name: TMR1_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R/W	0x0	TMR1_MODE. Timer 1 mode. 0: Continuous mode. When interval value reached, the timer will not disable automatically. 1: Single mode. When interval value reached, the timer will disable automatically.
6:4	R/W	0x0	TMR1_CLK_PRES. Select the pre-scale of timer 1 clock source. 000: /1 001: /2 010: /4 011: /8 100: /16 101: /32 110: /64 111: /128
3:2	R/W	0x1	TMR1_CLK_SRC. 00:LOSC 01: OSC24M 10: / 11: /

1	R/W	0x0	TMR1_RELOAD. Timer 1 Reload. 0: No effect 1: Reload timer 1 Interval value After the bit is set, it can not be written again before it's cleared automatically.
0	R/W	0x0	TMR1_EN. Timer 1 Enable. 0: Stop/Pause 1: Start If the timer is started, it will reload the interval value to internal register, and the current counter will count from interval value to 0. If the current counter does not reach the zero, the timer enable bit is set to "0", the current value counter will pause. At least wait for 2 cycles, the start bit can be set to 1. In timer pause state, the interval value register can be modified. If the timer is started again, and the Software hope the current value register to down-count from the new interval value, the reload bit and the enable bit should be set to 1 at the same time.

3.7.6.7 Timer 1 Interval Value Register(Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: TMR1_INTV_VALUE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	TMR1_INTV_VALUE. Timer 1 Interval Value. The value should consider the system clock and the timer clock source.

3.7.6.8 Timer 1 Current Value Register(Default Value: 0x0000_0000)

Offset: 0x0028			Register Name:TMR1_CUR_VALUE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	TMR1_CUR_VALUE. Timer 1 Current Value. Timer 1 current value is a 32-bit down-counter (from interval value to 0).

3.7.6.9 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/W1S	0x0	WDOG_IRQ_EN. Watchdog Interrupt Enable. 0: No effect

			1: Watchdog interrupt enable.

3.7.6.10 Watchdog Status Register (Default Value: 0x0000_0000)

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

3.7.6.11 Watchdog Control Register(Default Value: 0x0000_0000)

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

3.7.6.12 Watchdog Configuration Register (Default Value: 0x0000_0001)

Offset: 0x00B4			Register Name:WDOG_CFG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1:0	R/W	0x1	WDOG_CONFIG. 00:/ 01: to whole system 10: only interrupt 11: /

3.7.6.13 Watchdog Mode Register (Default Value: 0x0000_0000)

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

7:4	R/W	0x0	WDOG_INTV_VALUE. Watchdog Interval Value. Watchdog clock source is OSC24M / 750. If the clock source is turned off, Watchdog 0 will not work. 0000: 16000 cycles (0.5s) 0001: 32000 cycles (1s) 0010: 64000 cycles (2s) 0011: 96000 cycles (3s) 0100: 128000 cycles (4s) 0101: 160000 cycles (5s) 0110: 192000 cycles (6s) 0111: 256000 cycles (8s) 1000: 320000 cycles (10s) 1001: 384000 cycles (12s) 1010: 448000 cycles (14s) 1011: 512000 cycles (16s) 1100: / 1101: / 1110: / 1111: /
3:1	/	/	/
0	R/W1S	0x0	WDOG_EN. Watchdog Enable. 0: No effect 1: Enable the Watchdog

3.7.6.14 AVS Counter Control Register (Default Value: 0x0000_0000)

Offset: 0x00C0			Register Name: AVS_CNT_CTL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:10	/	/	/
9	R/W	0x0	AVS_CNT1_PS. Audio/Video Sync Counter 1 Pause Control 0: Not pause 1: Pause Counter 1
8	R/W	0x0	AVS_CNT0_PS. Audio/Video Sync Counter 0 Pause Control 0: Not pause 1: Pause Counter 0
7:2	/	/	/
1	R/W	0x0	AVS_CNT1_EN. Audio/Video Sync Counter 1 Enable/ Disable. The counter source is OSC24M. 0: Disable 1: Enable
0	R/W	0x0	AVS_CNT0_EN.

			Audio/Video Sync Counter 0 Enable/ Disable. The counter source is OSC24M. 0: Disable 1: Enable

3.7.6.15 AVS Counter 0 Register (Default Value: 0x0000_0000)

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

3.7.6.16 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. Counter 1 for Audio/ Video Sync Application The high 32 bits of the internal 33-bit counter register. The initial value of the internal 33-bit counter register can be set by software. The LSB bit of the 33-bit counter register should be zero when the initial value is updated. It will count from the initial value. The initial value can be updated at any time. It can also be paused by setting AVS_CNT1_PS to '1'. When it is paused, the counter will not increase.

3.7.6.17 AVS Counter Divisor Register (Default Value: 0x05DB_05DB)

Offset: 0x00CC			Register Name: AVS_CNT_DIV_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	/	/	/
27:16	R/W	0x5DB	AVS_CNT1_D. Divisor N for AVS Counter 1 AVS CN1 CLK=24MHz/Divisor_N1. Divisor N1 = Bit [27:16] + 1. The number N is from 1 to 0x7ff. The zero value is reserved. The internal 33-bit counter engine will maintain another 12-bit counter. The 12-bit counter is used for counting the cycle number of one 24MHz clock. When the 12-bit counter reaches (>= N) the divisor value, the internal 33-bit

			counter register will increase 1 and the 12-bit counter will reset to zero and restart again. It can be configured by software at any time.
15:12	/	/	/
11:0	R/W	0x5DB	AVS_CNT0_D. Divisor N for AVS Counter 0 AVS CN0 CLK=24MHz/Divisor_N0. Divisor N0 = Bit [11:0] + 1 The number N is from 1 to 0x7ff. The zero value is reserved. The internal 33-bit counter engine will maintain another 12-bit counter. The 12-bit counter is used for counting the cycle number of one 24MHz clock. When the 12-bit counter reaches (>= N) the divisor value, the internal 33-bit counter register will increase 1 and the 12-bit counter will reset to zero and restart again. It can be configured by software at any time.

3.8 High Speed Timer

3.8.1 Overview

The High Speed Timer(HSTimer) module implements the timing and counting functions.The HSTimer has the following features:

- Timing clock is AHB1 that can provides more accurate timing clock, the normal working frequency is 200MHz
- Configurable 5 prescale factor
- Configurable 56-bit down timer
- Supports 2 working modes: continuous mode and single mode
- Supports test mode
- Generates an interrupt when the count is decreased to 0

3.8.2 Block Diagram

Figure3-28 shows a block diagram of the HSTimer.

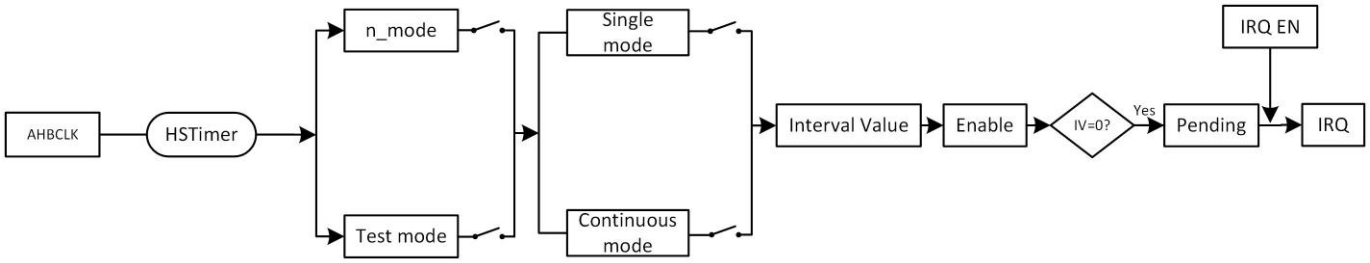


Figure3- 16. HSTimer Block Diagram

3.8.3 Operations and Functional Description

3.8.3.1 HSTimer Formula

$$\frac{(HS\_TMR\_INTV\_HI\_REG<<32 + HS\_TMR\_INTV\_LO\_REG) - (HS\_TMR\_CURNT\_HI\_REG<<32 + HS\_TMR\_CURNT\_LO\_REG)}{AHB1CLK} \times HS\_TMR\_CLK$$

HS_TMR_INTV_HI_REG: Initial of Counter Higher Bit

HS_TMR_INTV_LO_REG: Initial of Counter Lower Bit

HS_TMR_CURNT_HI_REG: Current Value of Counter Higher Bit

HS_TMR_CURNT_LO_REG: Current Vaule of Counter Lower Bit

AHB1CLK: AHB1 Clock Frequency

HS_TMR_CLK: Time Prescale Ratio of Counter

3.8.3.2 Typical Application

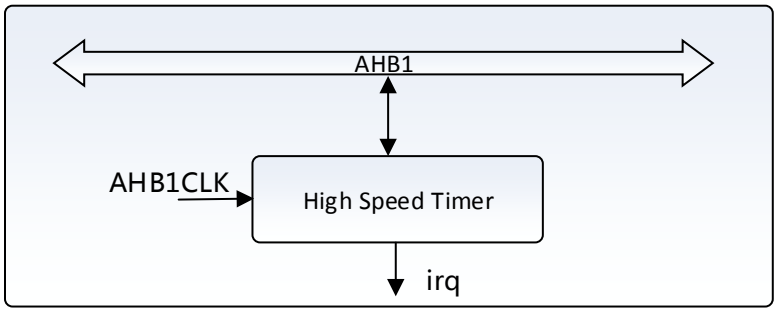


Figure3- 17. HSTimer Application Diagram

The high speed timer is hung on AHB1, and the high speed timer controls registers by AHB1.The high speed timer can generate interrupt.

3.8.3.3 Function Implemention

The timer is a 56-bit down counter, the counter value is decreased by 1 on each rising edge of the count clock.
The high speed timer has two timing modes.

- Continuous mode : The bit7 of **HS_TMR0_CTRL_REG** is set to the continuous mode, when the count value is decreased to 0, the high speed timer module reloads data from **HS_TMR_INTV_LO_REG** and **HS_TMR_INTV_HI_REG** then continues to count.
- Single mode : The bit7 of **HS_TMR0_CTRL_REG** is set to the single mode, when the count value is decreased to 0, the high speed timer stops counting. The high speed timer starts to count again only when a new initial value is loaded.

The high speed timer has two operating modes.

- Normal mode: When the bit31 of **HS_TMR0_CTRL_REG** is set to the normal mode, the high speed timer is used as 56-bit down counter, which can continuous timing and single timing.
- Test mode: When the bit31 of **HS_TMR0_CTRL_REG** is set to the normal mode, then **HS_TMR_INTV_LO_REG** must be set to 0x1, the high speed timer is used as 24-bit down counter, and **HS_TMR_INTV_HI_REG** is the initial value of the high speed timer.

Each high speed timer has a prescaler that divides the working clock frequency of each working timer by 1,2,4,8, 16.

3.8.3.4 Operating Mode

3.8.3.4.1 HSTimer Initialization

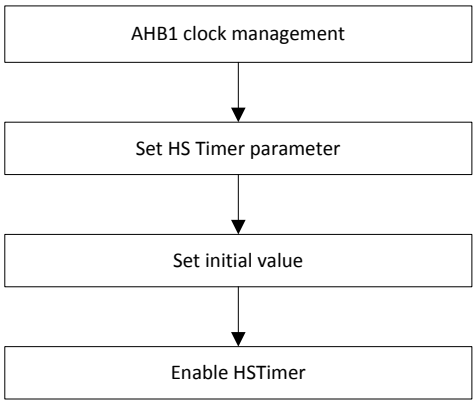


Figure3- 18. HSTimer Initialization Process

AHB1 clock management: Open the clock gating of AHB1 and de-assert the soft reset of AHB1 in CCU.

Configure the corresponding parameters of the high speed timer: clock source, prescaler factor, working mode, counting mode. These parameters that are written to **HS_TMR0_CTRL_REG** have no sequences.

Write the initial value: Firstly write the low-bit register **HS_TMR_INTV_LO_REG**, then write the high-bit register **HS_TMR_INTV_HI_REG**. Write the bit1 of **HS_TMR0_CTRL_REG** to load the initial value. If in timing stop stage of high speed timer, write the bit1 and bit0 of **HS_TMR0_CTRL_REG** to reload the initial value.

Enable high speed timer: Write the bit[0] of **HS_TMR0_CTRL_REG** to enable high speed timer to count. HSTimer starts to counter. Read **HS_TMR_CURNT_LO_REG** and **HS_TMR_CURNT_HI_REG** to get the current counter.

3.8.3.4.2 HSTimer Interrupt

Enable interrupt: Write the corresponding interrupt enable bit of **HS_TMR_IRQ_EN_REG**, when the counting time of high speed timer reaches , the corresponding interrupt generates.

After enter the interrupt process, write **HS_TMR_IRQ_STAS_REG** to clear the interrupt pending.

Resume the interrupt and continue to execute the interrupted process.

3.8.4 Programming Guidelines

Take making a 1us delay using HSTimer0 for an instance as follows, AHB1CLK will be configurated as 100MHz and n_mode,Single mode and 2 pre-scale will be selected in this instance.

```
writel(0x0, HS_TMR0_INTV_HI);                    //Set interval value Hi 0x0
writel(0x32, HS_TMR0_INTV_LO);                   //Set interval value Lo 0x32
writel(0x90, HS_TMR0_CTRL);                      //Select n_mode,2 pre-scale,single mode
writel(readl(HS_TMR0_CTRL)|(1<<1), HS_TMR0_CTRL);  //Set Reload bit
writel(readl(HS_TMR0_CTRL)|(1<<0), HS_TMR0_CTRL);  //Enable HSTimer0
While(!(readl(HS_TMR_IRQ_STAT)&1));              //Wait for HSTimer0 to generate pending
Writel(1,HS_TMR_IRQ_STAT);                       //Clear HSTimer0 pending
```


3.8.5 Register List

Module Name	Base Address
High Speed Timer	0x0300 5000

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 Timer 0 Control Register
HS_TMR0_INTV_LO_REG	0x0024	HS Timer 0 Interval Value Low Register
HS_TMR0_INTV_HI_REG	0x0028	HS Timer 0 Interval Value High Register
HS_TMR0_CURNT_LO_REG	0x002C	HS Timer 0 Current Value Low Register
HS_TMR0_CURNT_HI_REG	0x0030	HS Timer 0 Current Value High Register
HS_TMR1_CTRL_REG	0x0040	HS Timer 1 Control Register
HS_TMR1_INTV_LO_REG	0x0044	HS Timer 1 Interval Value Low Register
HS_TMR1_INTV_HI_REG	0x0048	HS Timer 1 Interval Value High Register
HS_TMR1_CURNT_LO_REG	0x004C	HS Timer 1 Current Value Low Register
HS_TMR1_CURNT_HI_REG	0x0050	HS Timer 1 Current Value High Register

3.8.6 Register Description

3.8.6.1 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/W1S	0x0	HS_TMR1_INT_EN. High Speed Timer 1 Interrupt Enable. 0: disable 1: High Speed Timer 1 Interval Value reached interrupt enable
0	R/W1S	0x0	HS_TMR0_INT_EN. High Speed Timer 0 Interrupt Enable. 0: No effect 1: High Speed Timer 0 Interval Value reached interrupt enable

3.8.6.2 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. High Speed Timer 1 IRQ Pending. Set 1 to the bit will clear it. 0: No effect

			1: Pending, High speed timer 1 interval value is reached
0	R/W1C	0x0	HS_TMR0_IRQ_PEND. High Speed Timer 0 IRQ Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending, High speed timer 0 interval value is reached

3.8.6.3 HS Timer 0 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. High speed timer 0 test mode. In test mode, the low register should be set to 0x1, the high register will down counter. The counter needs to be reloaded. 0: Normal mode 1: Test mode
30:8	/	/	/
7	R/W	0x0	HS_TMR0_MODE. High Speed Timer 0 mode. 0: Continuous mode. When interval value reached, the timer will not disable automatically. 1: Single mode. When interval value reached, the timer will disable automatically.
6:4	R/W	0x0	HS_TMR0_CLK Select the pre-scale of the high speed timer 0 clock sources. 000: /1 001: /2 010: /4 011: /8 100: /16 101: / 110: / 111: /
3:2	/	/	/
1	R/W1S	0x0	HS_TMR0_RELOAD. High Speed Timer 0 Reload. 0: No effect 1: Reload High Speed Timer 0 Interval Value
0	R/W	0x0	HS_TMR0_EN. High Speed Timer 0 Enable. 0: Stop/Pause 1: Start If the timer is started, it will reload the interval value to internal register, and the current counter will count from interval value to 0. If the current counter does not reach the zero, the timer enable bit is set to “0”, the current value counter will pause. At least wait for 2 cycles, the start bit

			can be set to 1. In timer pause state, the interval value register can be modified. If the timer is started again, and the Software hope the current value register to down-count from the new interval value, the reload bit and the enable bit should be set to 1 at the same time.

3.8.6.4 HS Timer 0 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. High Speed Timer 0 Interval Value [31:0].

3.8.6.5 HS Timer 0 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. High Speed Timer 0 Interval Value [55:32].

NOTE

The interval value register is a 56-bit register. When read or write the interval value, the Lo register should be read or written first. And the High register should be written after the Lo register.

3.8.6.6 HS Timer 0 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. High Speed Timer 0 Current Value [31:0].

3.8.6.7 HS Timer 0 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. High Speed Timer 0 Current Value [55:32].

NOTE

HS timer current value is a 56-bit down-counter (from interval value to 0).

The current value register is a 56-bit register. When read or write the current value, the Low register should be read or written first.

3.8.6.8 HS Timer 1 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. High speed timer 1 test mode. In test mode, the low register should be set to 0x1, the high register will down counter. The counter needs to be reloaded. 0: Normal mode 1: Test mode
30:8	/	/	/
7	R/W	0x0	HS_TMR1_MODE. High Speed Timer 0 mode. 0: Continuous mode. When interval value is reaching, the timer will not disable automatically. 1: Single mode. When interval value is reaching, the timer will disable automatically.
6:4	R/W	0x0	HS_TMR1_CLK Select the pre-scale of the high speed timer 1 clock sources. 000: /1 001: /2 010: /4 011: /8 100: /16 101: / 110: / 111: /
3:2	/	/	/
1	R/W1S	0x0	HS_TMR1_RELOAD. High Speed Timer 1 Reload. 0: No effect 1: Reload High Speed Timer 1 Interval Value
0	R/W	0x0	HS_TMR1_EN. High Speed Timer 1 Enable. 0: Stop/Pause 1: Start If the timer is starting, it will reload the interval value to internal register, and the current counter will count from interval value to 0. If the current counter does not reach the zero, the timer enable bit is set to “0”, the current value counter will pause. At least wait for 2 cycles, the start bit can be set to 1. In timer pause state, the interval value register can be modified. If the timer is started again, and the Software hope the current value register to down-count from the new interval value, the reload bit and the enable bit should be set to

			1 at the same time.

3.8.6.9 HS Timer 1 Interval Value Lo Register(Default Value: 0x0000_0000)

Offset: 0x0044			Register Name: HS_TMR0_INTV_LO_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	HS_TMR1_INTV_VALUE_LO. High Speed Timer 1 Interval Value [31:0].

3.8.6.10 HS Timer 1 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. High Speed Timer 1 Interval Value [55:32].

NOTE

the interval value register is a 56-bit register. When read or write the interval value, the Lo register should be read or write first. And the Hi register should be written after the Lo register.

3.8.6.11 HS Timer 1 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. High Speed Timer 1 Current Value [31:0].

3.8.6.12 HS Timer 1 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. High Speed Timer 1 Current Value [55:32].

NOTE

HS timer current value is a 56-bit down-counter (from interval value to 0)

The current value register is a 56-bit register. When read or write the current value, the Lo register should be read or write first.

3.9 GIC

3.9.1 Interrupt Source

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
29	PPI 13	0x0074	PPI 13 interrupt
30	PPI 14	0x0078	PPI 14 interrupt
31	PPI 15	0x007C	PPI 15 interrupt
32	GPADC	0x0080	GPADC interrupt
33	THS	0x0084	THS interrupt
34	LRADC	0x0088	LRADC interrupt
35			

Interrupt Number	Interrupt Source	Interrupt Vector	Description
36			
37			
38			
39	DMIC	0x009C	DMIC interrupt
40	DRAM_PHY	0x00A0	DRAM PHY interrupt
41	DRAM	0x00A4	DRAM interrupt
42	DMA	0x00A8	DMA interrupt
43	MBOX	0x00AC	MBOX interrupt
44	SPINLOCK	0x00B0	SPINLOCK interrupt
45			
46	WDOG	0x00B8	WDOG interrupt
47			
48			
49	BUS_TIMEOUT	0x00C4	Bus Timeout interrupt
50	IOMMU	0x00C8	IOMMU interrupt
51	PSI	0x00CC	PSI interrupt
52			
53	G2D	0x00D4	G2D interrupt
54	CSI0_MIPI_RX0	0x00D8	CSI0_MIPI_RX0 interrupt
55	ISP	0x00DC	ISP interrupt
56			
57	VE	0x00E4	VE interrupt
58	Audio Codec_DET	0x0114	Audio Codec_DET interrupt
59	Audio Codec	0x0118	Audio Codec interrupt
60	NDFC	0x011C	NDFC interrupt
61	DE	0x0134	DE interrupt
62	CE_NS	0x014C	CE_NS interrupt
63	CE_S	0x0150	CE_S interrupt
64	I2S/PCM0	0x0160	I2S/PCM0 interrupt
65	I2S/PCM1	0x0164	I2S/PCM1 interrupt
66	TWI0	0x018C	TWI0 interrupt
67	TWI1	0x0190	TWI1 interrupt
68	TWI2	0x0194	TWI2 interrupt
69	MIPI_DSI	0x01A4	MIPI_DSI interrupt
70	SMHC0	0x01B4	SMHC0 interrupt
71	SMHC1	0x01B8	SMHC1 interrupt
72	SMHC2	0x01BC	SMHC2 interrupt
73	UART0	0x01E4	UART0 interrupt
74	UART1	0x01E8	UART1 interrupt
75	UART2	0x01EC	UART2 interrupt
76	UART3	0x01F0	UART3 interrupt
77	UART4	0x01F4	UART4 interrupt
78	SPI0	0x0204	SPI0 interrupt
79	SPI1	0x0208	SPI1 interrupt

Interrupt Number	Interrupt Source	Interrupt Vector	Description
80	HSTIMER0	0x0224	HSTIMER0 interrupt
81	HSTIMER1	0x0228	HSTIMER1 interrupt
82	TIMER0	0x0244	TIMER0 interrupt
83	TIMER1	0x0248	TIMER1 interrupt
84	TCON_LCD0	0x0264	TCON_LCD0 interrupt
85	USB2.0_DRD_DEVICE	0x0284	USB2.0_DRD_DEVICE
86	USB2.0_DRD_EHCI	0x0288	USB2.0_DRD_EHCI interrupt
87	USB2.0_DRD_OHCI	0x028C	USB2.0_DRD_OHCI interrupt
88	USB2.0_HOST_EHCI	0x0290	USB2.0_HOST_EHCI
89	USB2.0_HOST_OHCI	0x0294	USB2.0_HOST_OHCI interrupt
90	GPIOB	0x02AC	GPIOB interrupt
91	GPIOF	0x02BC	GPIOF interrupt
92	GPIOG	0x02C0	GPIOG interrupt
93	GPIOH	0x02C4	GPIOH interrupt
94	GPU_GP	0x02D0	GPU0 interrupt
95	GPU_GP_MMU	0x02D4	GPU1 interrupt
96	GPU_PP0	0x02D8	GPU2 interrupt
97	GPU_PP0_MMU	0x02DC	GPU3 interrupt
98	GPU_PMU	0x02E0	GPU4 interrupt
99	GPU_PP1	0x02E4	GPU5 interrupt
100	GPU_PP1_MMU	0x02E8	GPU6 interrupt
101	CSI0_DMA0	0x0300	CSI0_DMA0 interrupt
102	CSI0_DMA1	0x0304	CSI0_DMA1 interrupt
103	CSI0_PARSER0	0x0310	CSI0_PARSER0 interrupt
104	CSI0_CCI0	0x0318	CSI0_CCI0 interrupt
105			
106			
107			
108			
109			
110			
111			
112			
113			
114			
115			
116	External NMI	0x0370	External NMI interrupt
117	R_TIMER0	0x0374	R_TIMER0 interrupt
118	R_TIMER1	0x0378	R_TIMER1 interrupt
119	R_TIMER2	0x037C	R_TIMER2 interrupt
120	R_TIMER3	0x0380	R_TIMER3 interrupt
121	R_Alarm0	0x0394	R_Alarm0 interrupt
122	R_Alarm1	0x0398	R_Alarm1 interrupt
123	R_WDOG	0x039C	R_WDOG interrupt

Interrupt Number	Interrupt Source	Interrupt Vector	Description
124	R_TWDOG	0x03A0	R_TWDOG interrupt
125	R_GPIOL	0x03A8	R_GPIOL interrupt
126	R_UART	0x03C4	R_UART0 interrupt
127	R_TWI	0x03D8	R_TWI interrupt
128	R_RSB	0x03EC	R_RSB interrupt
129			
130			
131			
132			
133			
134			
135			
136			
137			
138			
139			
140			
141			
142			
143			
144	C0_CTI0	0x0430	C0_CTI0 interrupt
145	C0_CTI1	0x0434	C0_CTI1 interrupt
146	C0_COMMTX0	0x0440	C0_COMMTX0 interrupt
147	C0_COMMTX1	0x0444	C0_COMMTX1 interrupt
148	C0_COMMRX0	0x0450	C0_COMMRX0 interrupt
149	C0_COMMRX1	0x0454	C0_COMMRX1 interrupt
150	C0_PMU0	0x0460	C0_PMU0 interrupt
151	C0_PMU1	0x0464	C0_PMU1 interrupt
152	C0_AXI_ERROR	0x0470	C0_AXI_ERROR interrupt

3.10 DMA

3.10.1 Overview

The direction memory access (DMA) is used to transfer data between a peripheral and a memory, between peripherals, or between memories. DMA is a high-speed data transfer operation that reduces the CPU resources. DMA has the following features.

- 8 channels DMA
- Provides 32 peripheral DMA requests for data read and 32 peripheral DMA requests for data write
- Supports transfer with linked list
- Supports programmable data width: 8/16/32/64-bit
- Support programmable length of DMA burst
- DRQ response includes wait mode and handshake mode
- Memory Devices supports non-aligned transform
- DMA channel supports pause function

3.10.2 Block Diagram

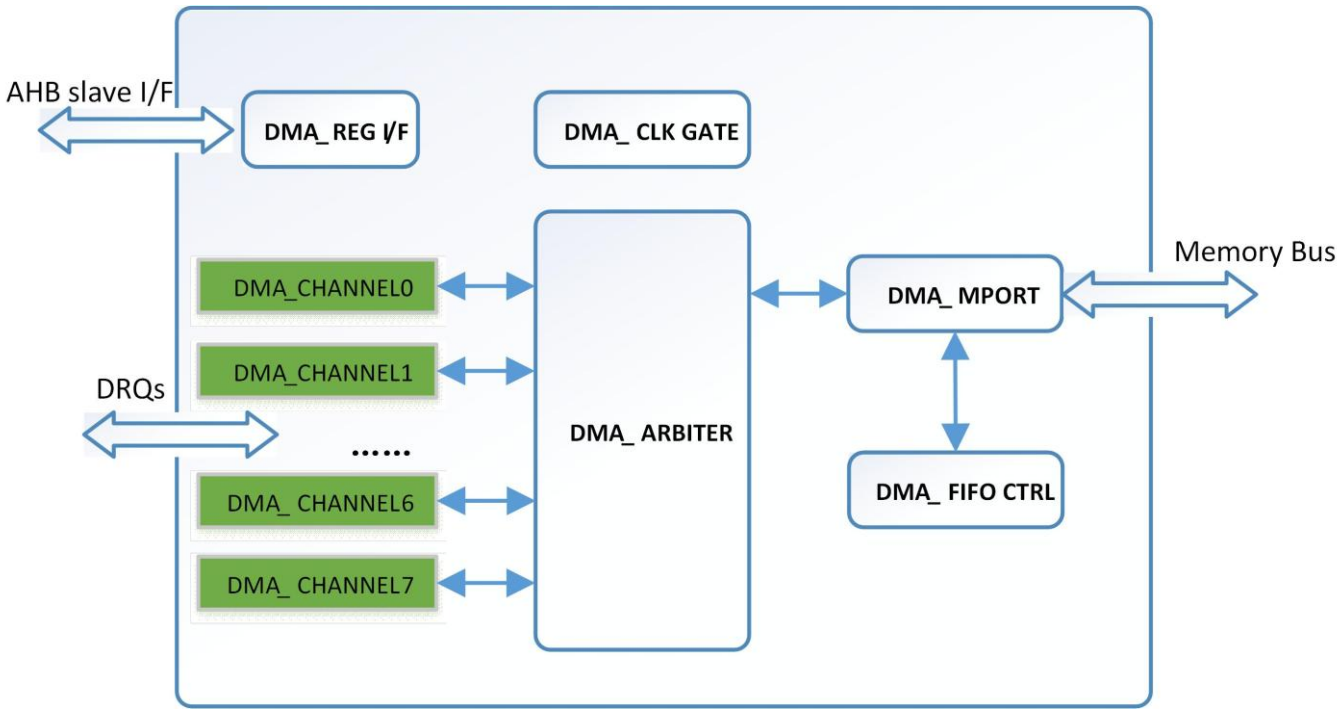


Figure3- 19. DMA Block Diagram

DMA_ARBITER: Arbitrate DMA read/write requirement of each channel, and convert to read/write requirement of each port.

DMA_CHANNEL: DMA transform engine.Each channel is independent. The priorities of DMA channels uses polling mechanism. When the DMA requests from two peripherals are valid simultaneously, if DMA_ARBITER is non-idle ,the next channel of the current channel has the higher priority; if DMA_ARBITER is idle, the channel0 has the highest priority, whereas the channel11 has the lowest priority.

DMA_MPORT: Receive read/write requirement of DMA_ARBITER ,and convert to the corresponding MBUS access.
DMA_FIFOCTL: Internal FIFO cell control module.
DMA_REGIF: Common register module, mainly used to resolve AHB1 demand.
DMA_CLKGATE: Hardware auto clock gating control module.

DMA integrates 8 independent DMA channels. When DMA channel starts, DMA gets DMA descriptor by **DMA_DESC_ADDR_REG** to use for the configuration information of the current DMA package transfer ,and DMA can transfer data between the specified peripherals through the configuration information. When a package transfer finished, DMA judges if the current channel transfer finished through the linked information in descriptor.

3.10.3 Operations and Functional Description

3.10.3.1 Clock and Reset

DMA is on AHB1.The clock of AHB1 influences the transfer efficiency of DMA.

3.10.3.2 Typical Application

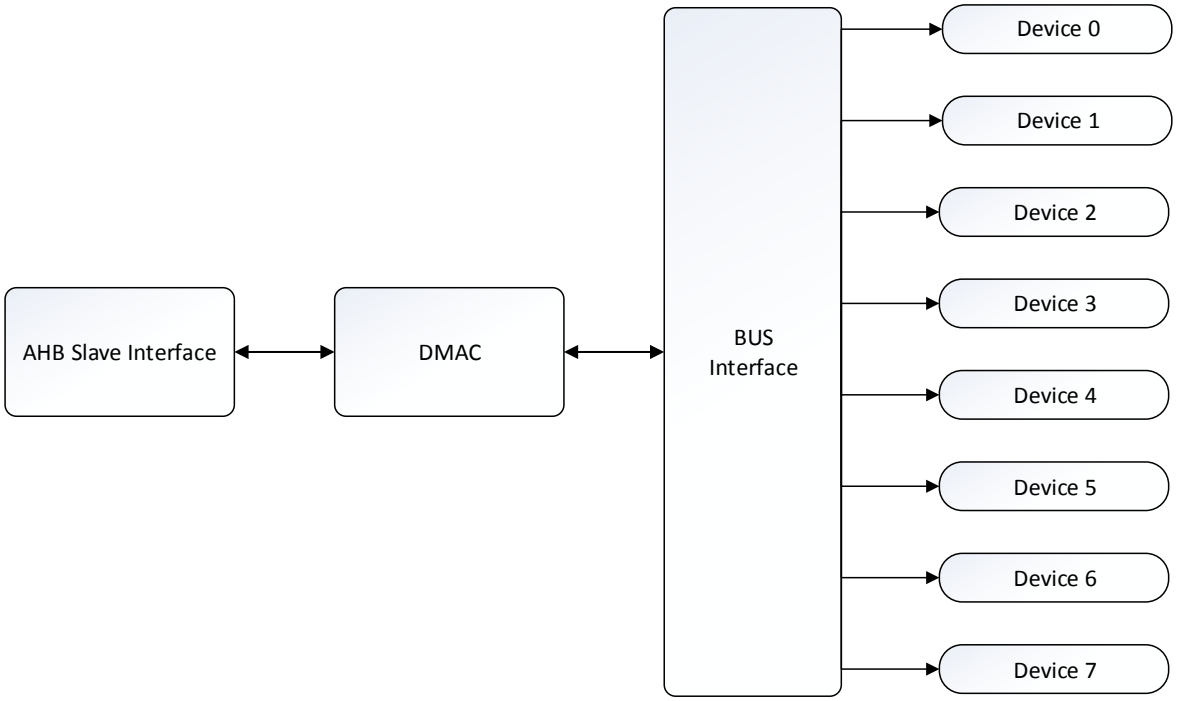


Figure3- 20. DMA Typical Application Diagram

System configure the conresponding information by AHB, enable the channel, and transfer the data between two devices. It improve the efficiency greatly.

3.10.3.3 DRQ Type

Table3- 9. DMA DRQ Table

Source DRQ Type		Destination DRQ Type	
port0	SRAM	port0	SRAM

port1	DRAM	port1	DRAM
port2		port2	
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		port9	
port10	NDFC	port10	NDFC
port11		port11	
port12	GPADC	port12	
port13		port13	
port14	UART0-RX	port14	UART0-TX
port15	UART1-RX	port15	UART1-TX
port16	UART2-RX	port16	UART2-TX
port17	UART3-RX	port17	UART3-TX
port18	UART4-RX	port18	UART4-TX
port19		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	
port29		port29	
Port30	USB2.0_OTG_EP1	Port30	USB2.0_OTG_EP1
Port31	USB2.0_OTG_EP2	Port31	USB2.0_OTG_EP2
Port32	USB2.0_OTG_EP3	Port32	USB2.0_OTG_EP3
Port33	USB2.0_OTG_EP4	Port33	USB2.0_OTG_EP4
Port34	USB2.0_OTG_EP5	Port34	USB2.0_OTG_EP5

3.10.3.4 DMA Descriptor

Figure3- 21. DMA Descriptor

DMA descriptor is the configuration information of DMA transfer that decides the DMA working mode. Each descriptor includes 6 words, in turn, configuration, source address, destination address, byte counter, parameter, link.

Configuration : Configure the following information by DMA_CFG_REG.

- DRQ type of source and destination.
- Transferred address count mode : IO mode indicates the address is fixed during transfer; linear mode indicates the address is increasing during transfer.
- Transferred block length : block length is the amount of DMA transferred data in one-shot valid DRQ. The block length supports 1-bit,4-bit,8-bit or 16-bit mode.
- Transferred data width: data width indicates the data width of every operation, and supports 8-bit,16-bit,32-bit or 64-bit mode.

Source Address: Configure the transferred source address.

Destination Address: Configure the transferred destination address.

DMA reads data from the source address , then writes data to the destination address.

Byte counter: Configure the amount of a package. The maximum package is not more than (2^25-1) bytes. If the amount of the package reaches the maximum value, even if DRQ is valid, DMA should stop the current transfer.

Parameter: Configure the interval between data block. The parameter is valid for non-memory peripherals. When DMA detects that DRQ is high level, DMA transfers block cycle. And during time, the changing of DRQ is ignored.After transferred, DMA waits the setting cycle(WAIT_CYC), then executes the next DRQ detection.

If the value of the link is 0xFFFFF800, the current package is at the end of the linked list. DMA will stop transfer after the package is transferred; if the value of the link is not 0xFFFFF800, the value of the link is considered the descriptor address of the next package.

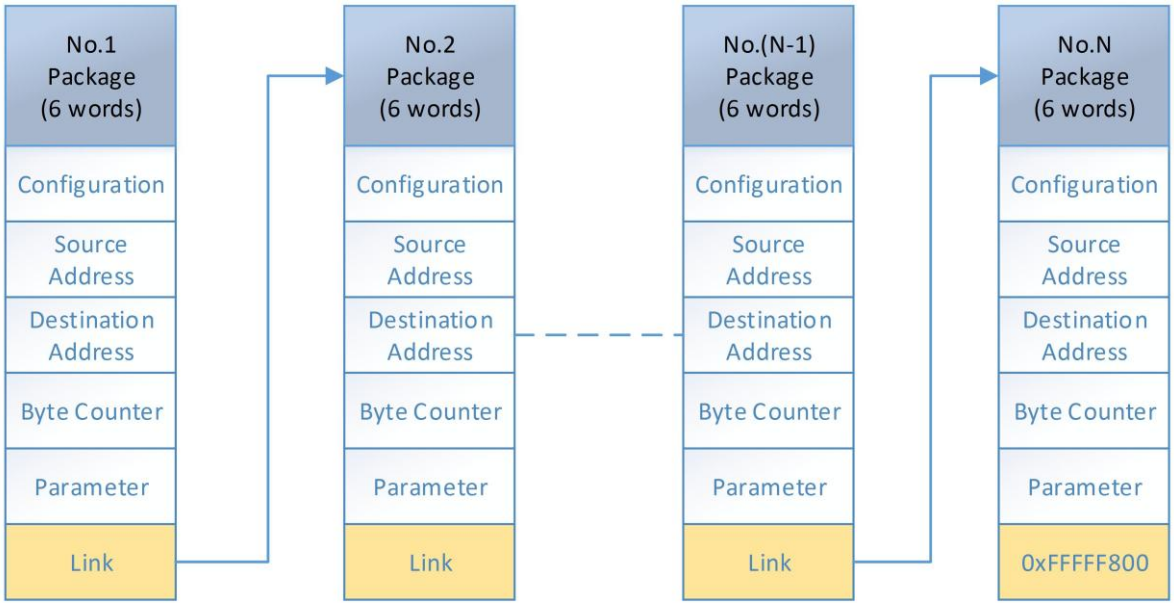


Figure3- 22. DMA Chain Transfer

3.10.3.5 Interrupt

The half package interrupt is enabled, DMA sends half package interrupt after the half package transfer completes. The total package interrupt is enabled, DMA sends package end interrupt after the total package transfer completes. The total queue interrupt is enabled, DMA sends queue end interrupt after the total queue completes. Notice that when CPU does not respond to the interrupts timely, or two DMA interrupts generate very closely, the later interrupt may override the former one. So DMA has only a system interrupt source.

3.10.3.6 Security

DMA supports system Trustzone, and supports DMA channel secure mode. Each DMA channel is secure by default. When system Trustzone is enabled, DMA is secure, only the secure devices can access DMA.

When DMA channel is configured to non-secure, then the channel can only access the non-secure memory area. DMA cannot write data to secure memory area, the read-back data from secure memory area is 0.

3.10.3.7 Clock Gating

DMA CLK GATE module is the clock module of auto-controlled by hardware. DMA CLK GATE module is mainly used to generate the clock of DMA sub-module and the local circuit in module, including clock gating of channel and clock gating of public part.

The clock gating of the channel indicates DMA clock can auto-open when the system accesses the current DMA channel register and DMA channel is enabled. When DMA transfer is completed, DMA channel clock can auto-close after 16 HCLK delay, meanwhile the clock of the corresponding channel control and FIFO control will be closed.

The clock gating of the common part indicates the clock of the common circuit can auto-close when all DMA channels are opened. The common circuit includes the common circuit of FIFO control module, MPORT module and memory bus clock.

DMA clock gating can support all the functions stated above or not by software.

3.10.3.8 Transfer Mode

DMA supports two data transfer modes: wait mode and handshake mode.

Wait Mode

When device request signal enters DMA, the device request signal is transformed into the internal DRQ signal through block and wait counter. The transformed principle is as follows.

- When DMA detects the external request signal valid, DMA starts to operate the device, the internal DRQ always holds high level before the block operating amount reaches.
- When the transfer amount of DMA reaches the block operating amount, the internal DRQ pulls low automatically.
- After the internal DRQ holds low automatically to the DMA cycle of wait counter times, DMA restarts to detect the external request, if the external request signal is valid, then the next transfer starts.

Handshake Mode

- When DMA detects the external request signal valid, DMA starts to operate the device, the internal DRQ always holds high level before the block operating amount reaches.
- When the transfer amount of DMA reaches the block operating amount, the internal DRQ pulls low automatically; meanwhile within the last operation , DMA follows the operating demand to send DMA last signal simultaneously.
- The DMA last signal that is used as a part of DMA demand transmits at BUS, when the device receives the operating demand of DMA last at BUS, the device can judge DMA transfer block length finished, that is before transmit the request again ,DMA operation cannot appear, and a DMA active signal is generated to the DMA controller. Notice that each DRQ signal of device corresponds to an active signal, if the device has many DRQ signals, then DMA returns different active signal through different bus operation.
- When DMA receives the transmitted active signal of devices, DMA ACK signal is returned to devices.
- After the device receives DMA ACK signal, if all operations of devices are completed , FIFO status and DRQ status are refreshed, then active signal is set as invalid.
- When DMA detects the falling edge of active signal, then the corresponding ACK signal is set as invalid, and DMA restarts to detect the external request signal. If the request signal is valid, then the next transfer starts.

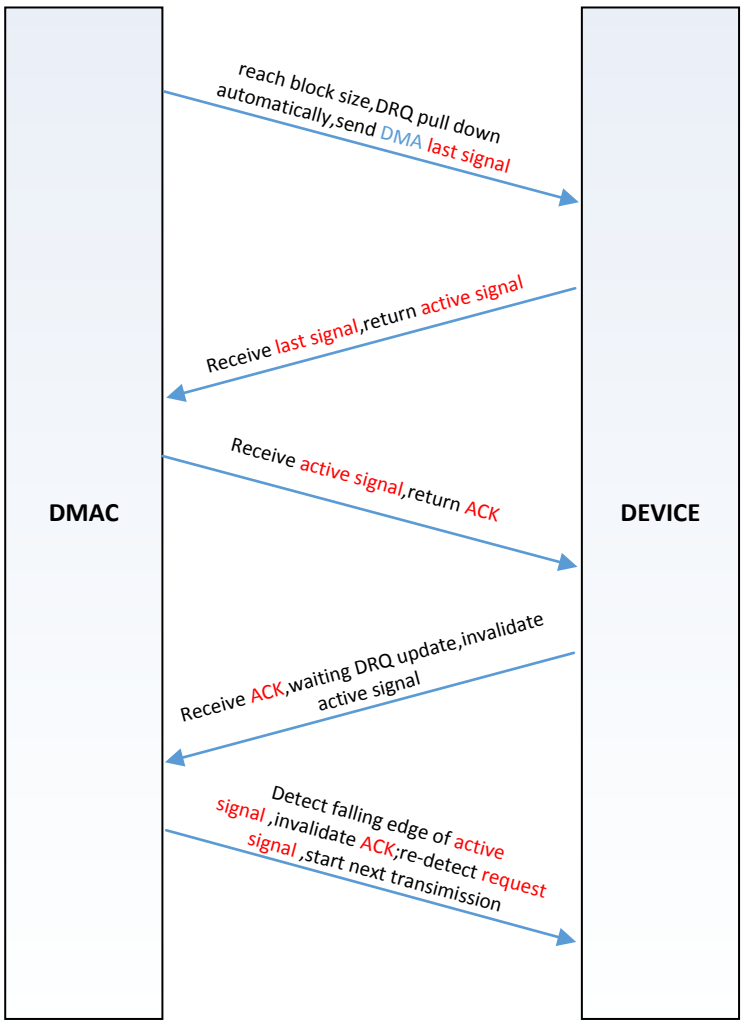


Figure3- 23. DMA HandShake Mode

3.10.3.9 Autoalignment Function

The DMA supports address alignment of non-IO devices, that is when the start address of non-IO devices is non 32-byte aligned, DMA firstly aligns the burst transfer within 32-byte to 32-byte. If the device of a DMA channel is configured to non-IO type, and the start address is 0x86, then DMA firstly aligns 26-byte burst transfer to 0xA0, then DMA transfers by 64-byte burst(maximum transfer amount of MBUS allowed). The address alignment function helps to improve the DRAM access efficiency.

IO devices does not support address alignment, so the bit width of IO devices must match the address offset, or not DMA ignores the non-consistency and indirectly transmits data of the corresponding bit width to the address.

3.10.3.10 Operating Mode

3.10.3.10.1 Clock Control

- The DMA clock is synchronous with AHB1 clock. Make sure that open the DMA gating bit of AHB1 clock before access DMA register.
- The reset input signal of DMA is asynchronous with AHB1, and is low valid by default. Make sure that de-assert the

reset signal of DMA before access DMA register.

- To avoid indefinite state within registers , firstly de-assert the reset signal, secondly open the gating bit of AHB1.
- DMA has the function of clock auto gating ,DMA clock can be disabled in DMA idle state using software to reduce power consumption. DMA enables clock auto gating by default.

3.10.3.10.2 DMA Transfer Process

The DMA transfer process is as follows.

Request DMA channel, and judge the idle state of the channel by the enable or disable of DMA channel.

Write the descriptor with 6-word into memory, the descriptor must be word-aligned. Refer to **3.10.3.4 DMA descriptor** in detail.

Write the start address of storing descriptor to **DMA_DESC_ADDR_REG**.

Enable DMA channel, and write the corresponding channel to **DMA_EN_REG**.

DMA obtains the descriptor information.

Start to transmit a package ,when half package is completed, DMA sends **Half Package Transfer Interrupt**; when total package is completed, DMA sends **Package End Transfer Interrupt**. These interrupt status can be read by **DMA_IRQ_PEND_REG**.

Set **DMA_PAU_REG** to pause or resume the data transfer.

After completed the total package transfer, 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; if the link is other value, the next package starts to transmit. When the transfer ends, DMA sends **Queue End Transfer Interrupt**.

Disable the DMA channel.

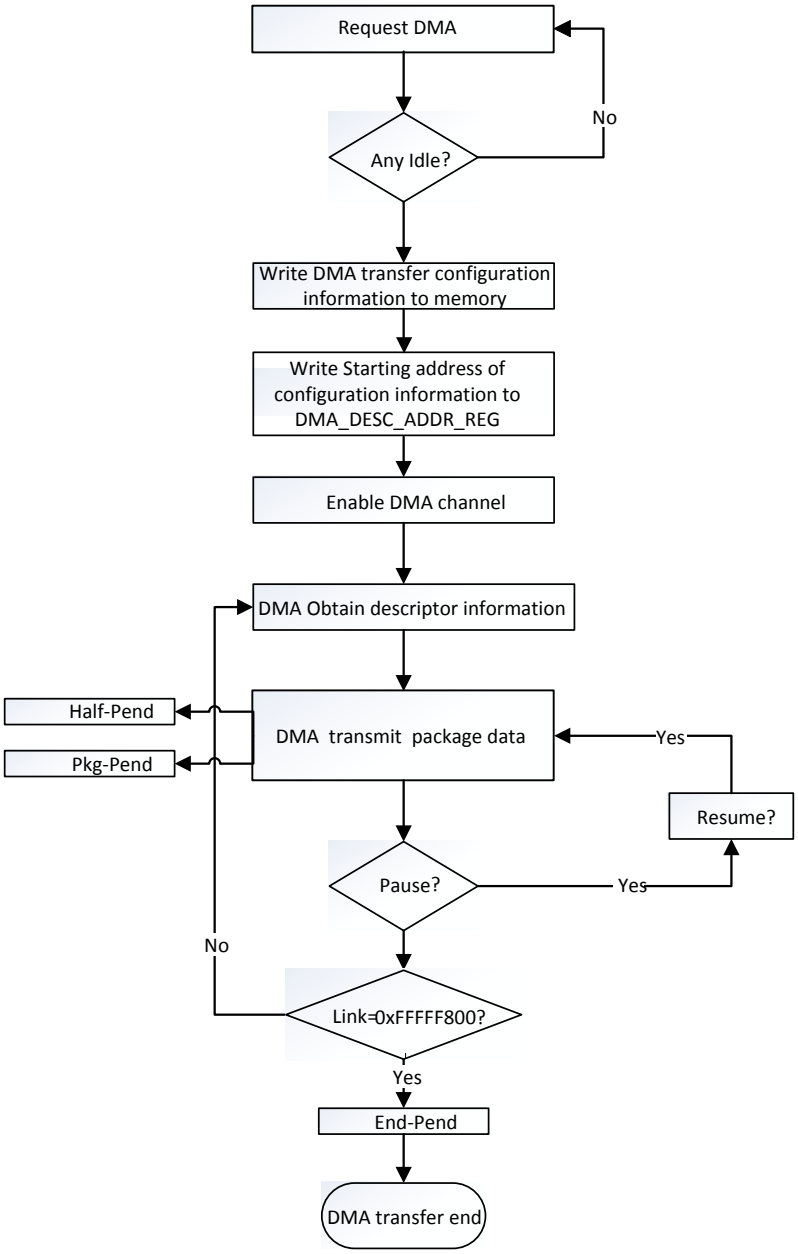


Figure3- 24. DMA Transfer Process

3.10.3.10.3 DMA Interrupt

Enable interrupt: write the corresponding interrupt enable of **DMA_IRQ_EN_REG**, when the corresponding interrupt condition is satisfied,the corresponding interrupt generates.

After enter the interrupt process, write **DMA_IRQ_PEND_REG** to clear the interrupt pending，and execute the process of waiting for the interrupt.

Resume the interrupt and continue to execute the interrupted process.

3.10.4 Programming Guidelines

Transferred data width of the IO type must be same with the offset address of the start address.

Considering that the MBUS protocol does not support non-integer word read operations, the device must be able to ignore the redundant data whose width is not consistent with configuration when reading for non-word width device. That is, for devices that support non-integer word operations, the DMA command must be interpreted based on its own FIFO bit width rather than the read command bit width.

When the DMA transfer is paused, this is equivalent to invalid DRQ. Because DMA transfer command has a certain time delay, DMA will not stop transfer immediately until the current command and the command in Arbiter finished, at most 32-byte data.

DMA application example :

```
writel(0x00000000, mem_address + 0x00);   //Setting configuration, mem_address must be word-aligned
writel(0x00001000, mem_address + 0x04);    // Setting the start address for the source device
writel(0x20000000, mem_address + 0x08);   //Setting the start address for the destination device
writel(0x00000020, mem_address + 0x0C);   // Setting data package size
writel(0x00000000, mem_address + 0x10);   //Setting parameter
writel(0xFFFFF800, mem_address + 0x14);          //Setting the start address for the next descriptor
writel(mem_address, 0x01C02000+ 0x100 + 0x08); //Setting the start address for the DMA channel0 descriptor
do{
If(mem_address == readl(0x01C02000 + 0x100 + 0x08));
break;
}while(1);              //Make sure writing operation valid
writel(0x000000001, 0x01C02000 + 0x100 + 0x00);      // Enable DMA channel0 transfer
```

DMA supports increasing data package in transfer, there are a few points to note here.

- When the value of **DMA Channel Descriptor Address Register** is 0xFFFFF800, it indicates that DMA channel has got back the descriptor of the last package. When DMA channel completed the package data transfer, DMA channel will stop automatically data transfer.
- If needing increase data package,then at first it is essential to judge that whether DMA channel has got back the descriptor of the last package, if DMA channel has got back the descriptor of the last package, then this is impossible for increasing data package, DMA channel need start again. If DMA is not transmitting the last package,then the last descriptor address 0xFFFFF800 can be changed to the start address of the next descriptor.
- To ensure that the data changed valid, we can read again the value of **DMA Channel Descriptor Address Register** after changed the data. If there is not 0xFFFFF800,then it indicates that increasing data package is succeed, and fail otherwise. Because the process of increasing data package need some time, during this time,DMA channel may get back the descriptor of the last package.At the moment we can read again **DMA Channel Current Source Address Register** and **DMA Channel Current Destination Address Register**, if the increasing memory address accords with the information of the increasing data package, then the increasing data package is succeed, and fail otherwise.
- To ensure the higher success rate, it is suggested that increase data package before half package interrupt of penultimate data package.

3.10.5 Register List

Module Name	Base Address
DMA	0x0300 2000

Register Name	Offset	Description
DMA_IRQ_EN_REG0	0x0000	DMA IRQ Enable Register 0

DMA_IRQ_PEND_REG0	0x0010	DMA IRQ Pending Register 0
DMA_SEC_REG	0x0020	DMA Security Register
DMA_AUTO_GATE_REG	0x0028	DMA Auto Gating Register
DMA_STA_REG	0x0030	DMA Status Register
DMA_EN_REG	0x0100+N*0x0040	DMA Channel Enable Register (N=0~7)
DMA_PAU_REG	0x0100+N*0x0040+0x0004	DMA Channel Pause Register(N=0~7)
DMA_DESC_ADDR_REG	0x0100+N*0x0040+0x0008	DMA Channel Descriptor Address Register(N=0~7)
DMA_CFG_REG	0x0100+N*0x0040+0x000C	DMA Channel Configuration Register(N=0~7)
DMA_CUR_SRC_REG	0x0100+N*0x0040+0x0010	DMA Channel Current Source Register(N=0~7)
DMA_CUR_DEST_REG	0x0100+N*0x0040+0x0014	DMA Channel Current Destination Register(N=0~7)
DMA_BCNT_LEFT_REG	0x0100+N*0x0040+0x0018	DMA Channel Byte Counter Left Register(N=0~7)
DMA_PARA_REG	0x0100+N*0x0040+0x001C	DMA Channel Parameter Register(N=0~7)
DMA_MODE_REG	0x0100+N*0x0040+0x0028	DMA Mode Register(N=0~7)
DMA_FDESC_ADDR_REG	0x0100+N*0x0040+0x002C	DMA Former Descriptor Address Register(N=0~7)
DMA_PKG_NUM_REG	0x0100+N*0x0040+0x0030	DMA Package Number Register(N=0~7)

3.10.6 Register Description

3.10.6.1 DMA IRQ Enable Register0 (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: DMA_IRQ_EN_REG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31	/	/	/
30	R/W	0x0	DMA7_QUEUE_IRQ_EN DMA 7 Queue End Transfer Interrupt Enable. 0: Disable 1: Enable
29	R/W	0x0	DMA7_PKG_IRQ_EN DMA 7 Package End Transfer Interrupt Enable. 0: Disable 1: Enable
28	R/W	0x0	DMA7_HLAF_IRQ_EN DMA 7 Half Package Transfer Interrupt Enable. 0: Disable 1: Enable
27	/	/	/
26	R/W	0x0	DMA6_QUEUE_IRQ_EN DMA 6 Queue End Transfer Interrupt Enable. 0: Disable 1: Enable
25	R/W	0x0	DMA6_PKG_IRQ_EN DMA 6 Package End Transfer Interrupt Enable. 0: Disable 1: Enable
24	R/W	0x0	DMA6_HLAF_IRQ_EN

			DMA 6 Half Package Transfer Interrupt Enable. 0: Disable 1: Enable
23	/	/	/
22	R/W	0x0	DMA5_QUEUE_IRQ_EN DMA 5 Queue End Transfer Interrupt Enable. 0: Disable 1: Enable
21	R/W	0x0	DMA5_PKG_IRQ_EN DMA 5 Package End Transfer Interrupt Enable. 0: Disable 1: Enable
20	R/W	0x0	DMA5_HLAF_IRQ_EN DMA 5 Half package Transfer Interrupt Enable. 0: Disable 1: Enable
19	/	/	/
18	R/W	0x0	DMA4_QUEUE_IRQ_EN DMA 4 Queue End Transfer Interrupt Enable. 0: Disable 1: Enable
17	R/W	0x0	DMA4_PKG_IRQ_EN DMA 4 Package End Transfer Interrupt Enable. 0: Disable 1: Enable
16	R/W	0x0	DMA4_HLAF_IRQ_EN DMA 4 Half Package Transfer Interrupt Enable. 0: Disable 1: Enable
15	/	/	/
14	R/W	0x0	DMA3_QUEUE_IRQ_EN DMA 3 Queue End Transfer Interrupt Enable. 0: Disable 1: Enable
13	R/W	0x0	DMA3_PKG_IRQ_EN DMA 3 Package End Transfer Interrupt Enable. 0: Disable 1: Enable
12	R/W	0x0	DMA3_HLAF_IRQ_EN DMA 3 Half Package Transfer Interrupt Enable. 0: Disable 1: Enable
11	/	/	/
10	R/W	0x0	DMA2_QUEUE_IRQ_EN DMA 2 Queue End Transfer Interrupt Enable. 0: Disable

			1: Enable
9	R/W	0x0	DMA2_PKG_IRQ_EN DMA 2 Package End Transfer Interrupt Enable. 0: Disable 1: Enable
8	R/W	0x0	DMA2_HLAF_IRQ_EN DMA 2 Half Package Transfer Interrupt Enable. 0: Disable 1: Enable
7	/	/	/
6	R/W	0x0	DMA1_QUEUE_IRQ_EN DMA 1 Queue End Transfer Interrupt Enable. 0: Disable 1: Enable
5	R/W	0x0	DMA1_PKG_IRQ_EN DMA 1 Package End Transfer Interrupt Enable. 0: Disable 1: Enable
4	R/W	0x0	DMA1_HLAF_IRQ_EN DMA 1 Half Package Transfer Interrupt Enable. 0: Disable 1: Enable
3	/	/	/
2	R/W	0x0	DMA0_QUEUE_IRQ_EN DMA 0 Queue End Transfer Interrupt Enable. 0: Disable 1: Enable
1	R/W	0x0	DMA0_PKG_IRQ_EN DMA 0 Package End Transfer Interrupt Enable. 0: Disable 1: Enable
0	R/W	0x0	DMA0_HLAF_IRQ_EN DMA 0 Half Package Transfer Interrupt Enable. 0: Disable 1: Enable

3.10.6.2 DMA IRQ Pending Status Register 0 (Default Value: 0x0000_0000)

Offset:0x0010			**Register Name: DMA_IRQ_PEND_REG0**
Bit	**Read/Write**	**Default/Hex**	**Description**
31	/	/	/
30	R/W1C	0x0	DMA7_QUEUE_IRQ_PEND. DMA 7 Queue End Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect

			1: Pending
29	R/W1C	0x0	DMA7_PKG_IRQ_PEND DMA 7 Package End Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
28	R/W1C	0x0	DMA7_HLAF_IRQ_PEND. DMA 7 Half Package Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
27	/	/	/
26	R/W1C	0x0	DMA6_QUEUE_IRQ_PEND. DMA 6 Queue End Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
25	R/W1C	0x0	DMA6_PKG_IRQ_PEND. DMA 6 Package End Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
24	R/W1C	0x0	DMA6_HLAF_IRQ_PEND. DMA 6 Half Package Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
23	/	/	/
22	R/W1C	0x0	DMA5_QUEUE_IRQ_PEND. DMA 5 Queue End Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
21	R/W1C	0x0	DMA5_PKG_IRQ_PEND. DMA 5 Package End Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
20	R/W1C	0x0	DMA5_HLAF_IRQ_PEND. DMA 5 Half Package Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
19	/	/	/
18	R/W1C	0x0	DMA4_QUEUE_IRQ_PEND. DMA 4 Queue End Transfer Interrupt Pending. Setting 1 to the bit will

			clear it. 0: No effect 1: Pending
17	R/W1C	0x0	DMA4_PKG_IRQ_PEND. DMA 4 Package End Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
16	R/W1C	0x0	DMA4_HLAF_IRQ_PEND. DMA 4 Half Package Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
15	/	/	/
14	R/W1C	0x0	DMA3_QUEUE_IRQ_PEND. DMA 3 Queue End Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending.
13	R/W1C	0x0	DMA3_PKG_IRQ_PEND. DMA 3 Package End Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
12	R/W1C	0x0	DMA3_HLAF_IRQ_PEND. DMA 3 Half Package Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
11	/	/	/
10	R/W1C	0x0	DMA2_QUEUE_IRQ_PEND. DMA 2 Queue End Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
9	R/W1C	0x0	DMA2_PKG_IRQ_PEND. DMA 2 Package End Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
8	R/W1C	0x0	DMA2_HLAF_IRQ_PEND. DMA 2 Half Package Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
7	/	/	/

6	R/W1C	0x0	DMA1_QUEUE_IRQ_PEND. DMA 1 Queue End Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
5	R/W1C	0x0	DMA1_PKG_IRQ_PEND. DMA 1 Package End Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
4	R/W1C	0x0	DMA1_HLAF_IRQ_PEND. DMA 1 Half Package Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
3	/	/	/
2	R/W1C	0x0	DMA0_QUEUE_IRQ_PEND. DMA 0 Queue End Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
1	R/W1C	0x0	DMA0_PKG_IRQ_PEND. DMA 0 Package End Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending
0	R/W1C	0x0	DMA0_HLAF_IRQ_PEND. DMA 0 Half Package Transfer Interrupt Pending. Setting 1 to the bit will clear it. 0: No effect 1: Pending

3.10.6.3 DMA Security Register (Default Value: 0x0000_0000)

Offset:0x0020			**Register Name: DMA_SEC_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R/W	0x0	DMA7_SEC DMA channel 7 security. 0: Secure 1: Non-secure.
6	R/W	0x0	DMA6_SEC DMA channel 6 security. 0: Secure 1: Non-secure.

5	R/W	0x0	DMA5_SEC DMA channel 5 security. 0: Secure 1: Non-secure.
4	R/W	0x0	DMA4_SEC DMA channel 4 security. 0: Secure 1: Non-secure.
3	R/W	0x0	DMA3_SEC DMA channel 3 security. 0: Secure 1: Non-secure.
2	R/W	0x0	DMA2_SEC DMA channel 2 security. 0: Secure 1: Non-secure.
1	R/W	0x0	DMA1_SEC DMA channel 1 security. 0: Secure 1: Non-secure.
0	R/W	0x0	DMA0_SEC DMA channel 0 security. 0: Secure 1: Non-secure.

3.10.6.4 DMA Auto Gating Register (Default Value: 0x0000_0000)

Offset:0x0028			Register Name: DMA_AUTO_GATE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2	R/W	0x0	DMA_MCLK_CIRCUIT. DMA MCLK interface circuit auto gating bit. 0: Auto gating enable 1: Auto gating disable
1	R/W	0x0	DMA_COMMON_CIRCUIT. DMA common circuit auto gating bit. 0: Auto gating enable 1: Auto gating disable
0	R/W	0x0	DMA_CHAN_CIRCUIT. DMA channel circuit auto gating bit. 0: Auto gating enable 1: Auto gating disable

NOTE

When initializing DMA Controller, the bit2 of DMA_AUTO_GATE_REG should be set up.

3.10.6.5 DMA Status Register (Default Value: 0x0000_0000)

Offset:0x0030			Register Name: DMA_STA_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	/	/	/
30	R	0x0	MBUS FIFO Status 0: Empty 1: Not Empty
29:8	/	/	/
7	R	0x0	DMA7_STATUS DMA Channel 7 Status. 0: Idle 1: Busy
6	R	0x0	DMA6_STATUS DMA Channel 6 Status. 0: Idle 1: Busy
5	R	0x0	DMA5_STATUS DMA Channel 5 Status. 0: Idle 1: Busy
4	R	0x0	DMA4_STATUS DMA Channel 4 Status. 0: Idle 1: Busy
3	R	0x0	DMA3_STATUS DMA Channel 3 Status. 0: Idle 1: Busy
2	R	0x0	DMA2_STATUS DMA Channel 2 Status. 0: Idle 1: Busy
1	R	0x0	DMA1_STATUS DMA Channel 1 Status. 0: Idle 1: Busy
0	R	0x0	DMA0_STATUS DMA Channel 0 Status. 0: Idle 1: Busy

3.10.6.6 DMA Channel Enable Register (Default Value: 0x0000_0000)

Offset:0x0100+N*0x0040+0x0000(N=0~7)			Register Name: DMA_EN_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	DMA_EN. DMA Channel Enable 0: Disable 1: Enable

3.10.6.7 DMA Channel Pause Register (Default Value: 0x0000_0000)

Offset:0x0100+N*0x0040+0x0004(N=0~7)			Register Name: DMA_PAU_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	DMA_PAUSE. Pausing DMA Channel Transfer Data. 0: Resume Transferring 1: Pause Transferring

3.10.6.8 DMA Channel Descriptor Address Register (Default Value: 0x0000_0000)

Offset:0x0100+N*0x0040+0x0008(N=0~7)			Register Name: DMA_DESC_ADDR_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	DMA_DESC_ADDR DMA Channel Descriptor Address. The Descriptor Address must be word-aligned.

3.10.6.9 DMA Channel Configuration Register (Default Value: 0x0000_0000)

Offset:0x0100+N*0x0040+0x000C(N=0~7)			Register Name: DMA_CFG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	/	/	/
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
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.

3.10.6.10 DMA Channel Current Source Address Register (Default Value: 0x0000_0000)

Offset:0x0100+N*0x0040+0x0010(N=0~7)			Register Name: DMA_CUR_SRC_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	DMA_CUR_SRC. DMA Channel Current Source Address, read only.

3.10.6.11 DMA Chacnnel Current Destination Address Register (Default Value: 0x0000_0000)

Offset:0x0100+N*0x0040+0x0014(N=0~7)			Register Name: DMA_CUR_DEST_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	DMA_CUR_DEST. DMA Channel Current Destination Address, read only.

3.10.6.12 DMA Channel Byte Counter Left Register (Default Value: 0x0000_0000)

Offset:0x0100+N*0x0040+0x0018(N=0~7)			Register Name: DMA_BCNT_LEFT_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:25	/	/	/
24:0	R	0x0	DMA_BCNT_LEFT. DMA Channel Byte Counter Left, read only.

3.10.6.13 DMA Channel Parameter Register (Default Value: 0x0000_0000)

Offset:0x0100+N*0x0040+0x001C(N=0~7)			Register Name: DMA_PARA_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R	0x0	WAIT_CYC. Wait Clock Cycles

3.10.6.14 DMA Mode Register (Default Value: 0x0000_0000)

Offset:0x0100+N*0x0040+0x0028(N=0~7)			Register Name: DMA_MODE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:4	/	/	/
3	R/W	0x0	DMA_DST_MODE. 0: Wait mode. 1: Handshake mode.
2	R/W	0x0	DMA_SRC_MODE. 0: Wait mode. 1: Handshake mode.
1:0	/	/	/

3.10.6.15 DMA Former Descriptor Address Register (Default Value: 0x0000_0000)

Offset:0x0100+N*0x0040+0x002C(N=0~7)			Register Name: DMA_FDESC_ADDR_REG
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.

3.10.6.16 DMA Package Number Register (Default Value: 0x0000_0000)

Offset:0x0100+N*0x0040+0x0030(N=0~7)			Register Name: DMA_PKG_NUM_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	DMA_PKG_NUM.

			This register will record the number of packages which has been completed in one transmission.

3.11 RTC

3.11.1 Overview

The RTC(Real Time Clock) is used to display the real time and periodically wakeup .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 7-bit counter for counting year, 4-bit counter for counting month, 5-bit counter for counting day, 3-bit counter for counting week, 5-bit counter for counting hour, 6-bit counter for counting minute, 6-bit counter for counting second
- External connect a 32768Hz low-frequency oscillator for count clock
- Configurable initial value by software anytime
- Periodically alarm to wakeup the external devices
- Stores power-off information in eight 32-bit general purpose register

3.11.2 Block Diagram

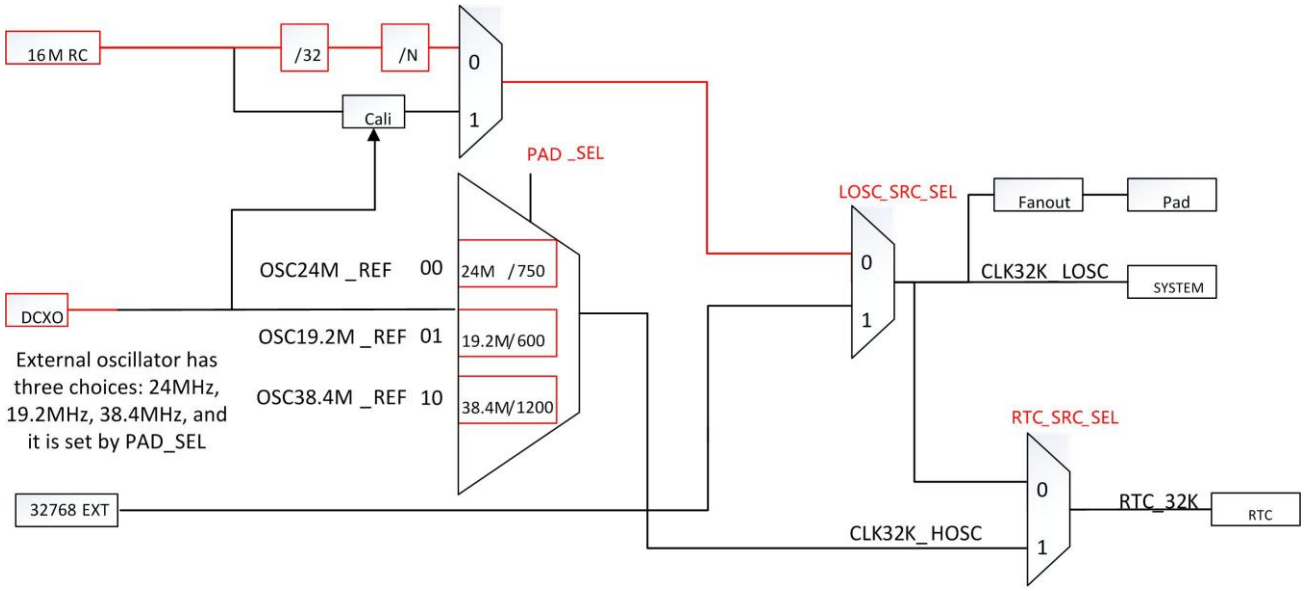


Figure3- 25. RTC Block Diagram

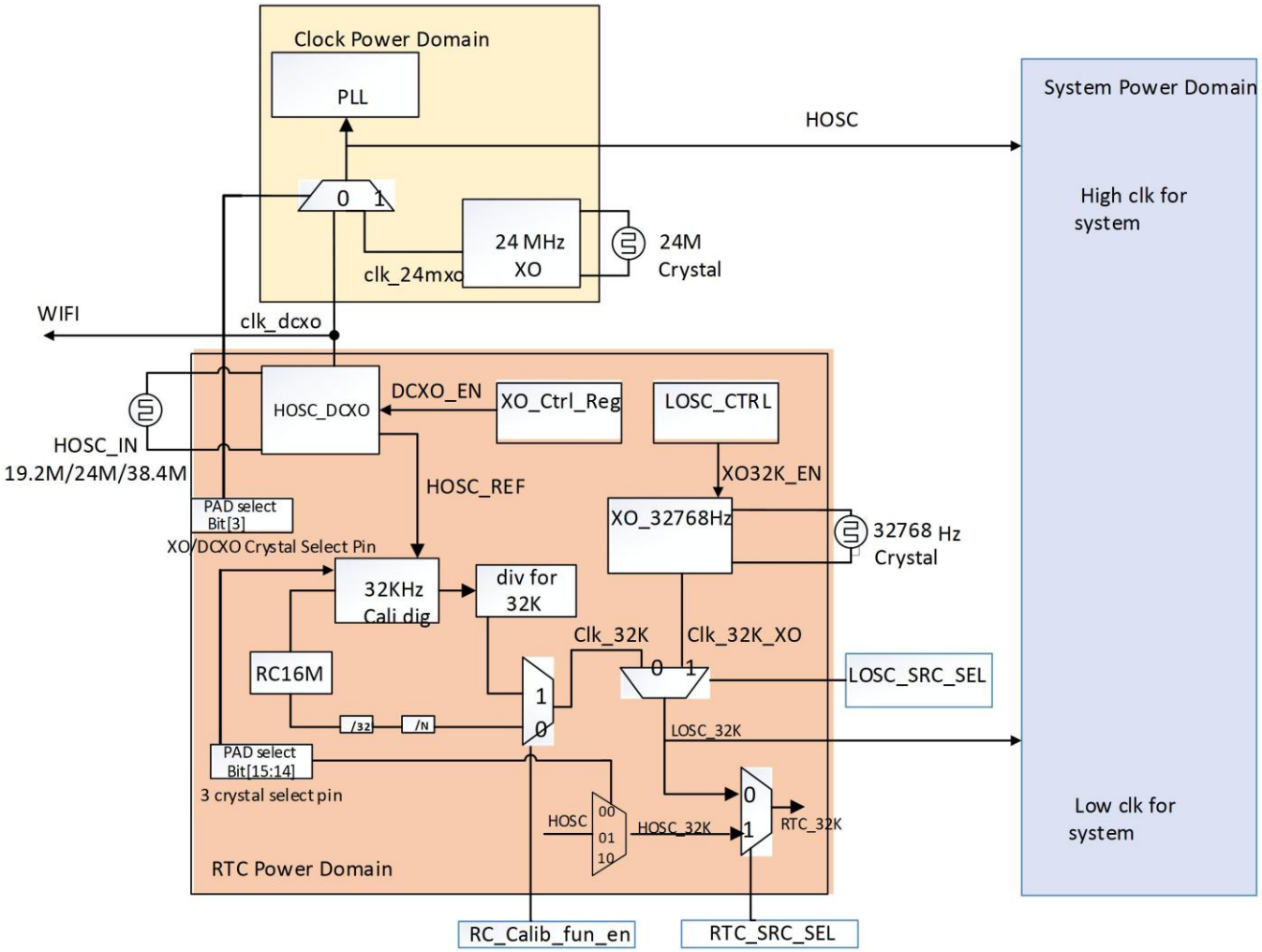


Figure3- 26. RTC Power Domain

Clock sources: HOSC high-frequency crystal oscillator(24M/19.2M/38.4M), 32768Hz low-frequency crystal oscillator, internal 16MHz RC.

Oscillating circuit: 32768Hz XO, HOSC_DCXO.

Controlling circuit: oscillator enable circuit, 32KHz calibration circuit, 32KHz frequency divider circuit, HOSC clock select circuit,32KHz clock select circuit, low-frequency clock source select circuit.

Output clock: HOSC,CLK32K_LOSC,RTC_32K.

3.11.3 Operations and Functional Descriptions

3.11.3.1 External Signals

Table 3-9 describes the external signals of RTC.

Table3- 10. RTC External Signals

Signal	Description
X32KIN	32KHz oscillator input
X32KOUT	32KHz oscillator output
X32KFOUT	32KHz clock fanout, provides low frequency clock for external devices
NMI	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

3.11.3.2 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.
The RTC module accesses its register by APB1.

3.11.3.3 Typical Application

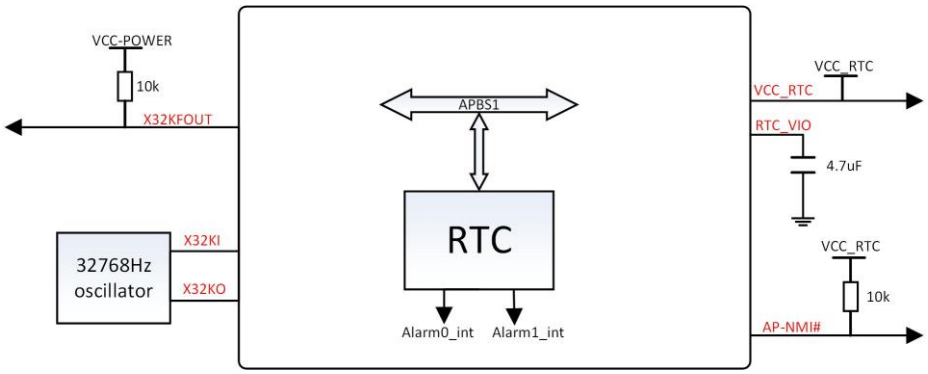


Figure3- 27. RTC Application Diagram

The system accesses RTC register by APB1 to generate the real time.
The external low-frequency oscillator must be 32.768 kHz.
If the external devices need low-frequency oscillator, X32KFOUT can provide.
AP-NMI# and alarm0 in common generate low level signal.

3.11.3.4 Function Implementation

3.11.3.4.1 Clock Sources

The RTC has two clock sources: internal RC , external low frequency oscillator.
The internal RC can change RTC clock by changing division ratio ;the external clock can not change clock.
The RTC selects the internal RC by default, when the system starts, the RTC can select by software the external low frequency oscillator to provide much accuracy clock.
The clock accurate of the RTC is related to the accurate of the external low frequency oscillator. The external oscillator usually selects 32.768 kHz oscillator with ±20ppm frequency tolerance.

3.11.3.4.2 Real Time Clock

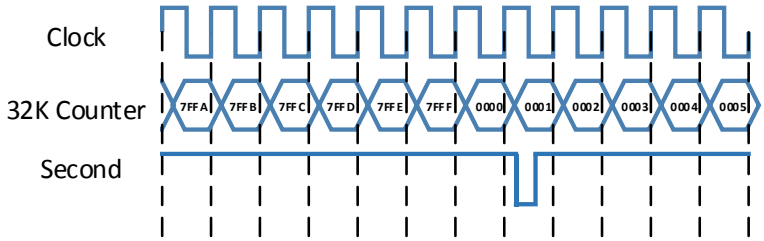


Figure3- 28. RTC Counter

The 32K counter adds 1 on each rising edge of the clock. When the clock number reaches 0x8000,32KHz counter starts to count again from 0, and the second counter adds 1. The 32KHz counter block diagram is as follows.

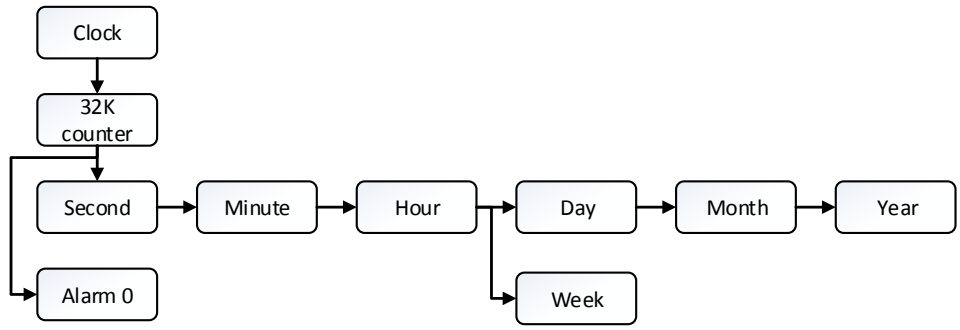


Figure3- 29. RTC 32KHz Counter Block Diagram

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

Table3- 11. RTC Counter Changing Range

Counter	Range
Second	If the counter value is not in the range from 0 to 59, then the counter value can change to 59 automatically.
Minute	If the counter value is not in the range from 0 to 59, then the counter value can change to 59 automatically.
Hour	If the counter value is not in the range from 0 to 23, then the counter value can change to 23 automatically.
Week	If the counter value is not in the range from 0 to 6, then the counter value can change to 6 automatically.
Day	If the counter value is not in the range from 1 to 31, then the counter value can change to the maximum value of that month automatically.
Month	If the counter value is not in the range from 1 to 12, then the counter value can change to 12 automatically.
Year	The software can set a reference year, the leap year can only set by software.

3.11.3.4.3 Alarm 0

The principle of alarm0 is similar to the second counter, the difference is that alarm0 is a 32-bit down counter. When the counter decreases to 0 from the initial value, the RTC generates the interrupt, or outputs low level signal by NMI pin to wakeup power management chip.

3.11.3.4.4 Alarm 1

The alrm1 can set alarm response time and the response cycle. When the system real time satisfies the setting time, the

RTC generates alram1 interrupt to handle alarm interrupt function.

3.11.3.4.5 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.

3.11.3.4.6 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,it is mainly used for internal digital logic.

3.11.3.4.7 RC Calibration

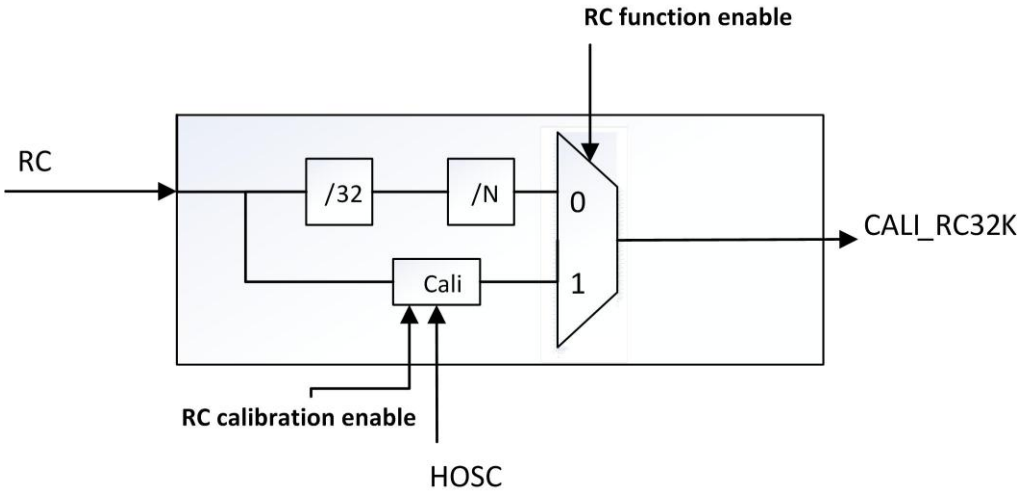


Figure3- 30. RC Calibration Block Diagram

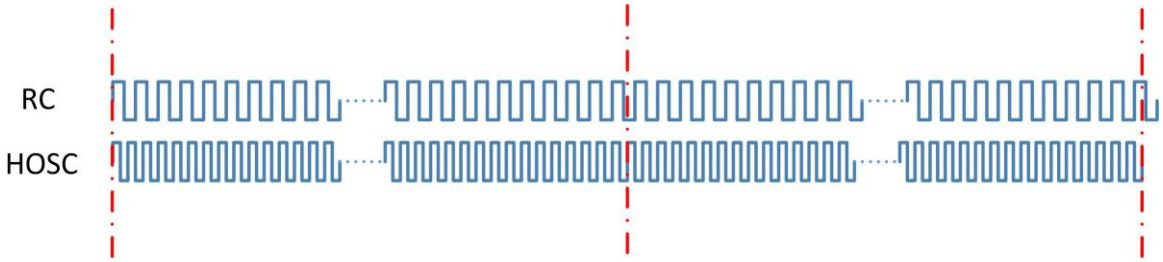


Figure3- 31. RC and 24MHz Waveform

HOSC as a reference clock, calculate the counter number(M) of RC clock within 1ms/16ms.
After calibration, the value of internal RC is get, Lockcount = 1000 * M(MHz).
K has 16-bit,the lower 5-bit is decimal, the higher 11-bit is integer.
The implementation of calibration is to output 32.768kHz.

3.11.3.5 Operating Mode

3.11.3.5.1 RTC Clock Control

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.

Auto switch: After enabled the bit[14] of **LOSC_CTRL_REG**, the RTC automatically switches clock source to the internal oscillator when the external oscillator could not output waveform, the switch status can query by the bit[1] of **LOSC_AUTO_SWT_STA_REG**.

After auto switch is valid, the clock source status bit cannot be changed, because the two functions are independent.

3.11.3.5.2 RTC Calendar

Write time initial value: Write the current time to **RTC_HH_MM_SS_REG** and **RTC_YY_MM_DD_REG**.

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

The leap year function can be set only by the software.

3.11.3.5.3 Alarm0

Enable alram0 interrupt by writing **ALARM0_IRQ_EN**.

Set the counter initial value, write the count-down second number to **ALARM0_COUNTER_REG**.

Enable alarm0 function by writing **ALARM0_ENABLE_REG**, then the software can query alarm count value in real time.

After enter the interrupt process, write **ALARM0_IRQ_STA_REG** to clear the interrupt pending, and execute the process of waiting for the interrupt.

Resume the interrupt and continue to execute the interrupted process.

Power-off wakeup is generated via SoC hardware and PMIC, software only need set pending condition of alarm0, and set 1 to **ALARM0_CONFIG_REG**.

3.11.3.5.4 Alarm 1

Set alarm1 initial value : write the hour,minute,second of alarm1 to **ALARM1_WK_HH_MM_SS**.

Write alarm week number enable bit to **ALARM1_EN_REG**.

When the bit[20:0] of **RTC_HH_MM_SS_REG** is equal to **ALARM1_WK_HH_MM-SS,** and the bit[31:29] of **RTC_HH_MM_SS_REG** is equal to the week number of **ALARM1_EN_REG**, then the condition of alarm 1 is satisfied, pending is set automatically by hardware.

When **ALARM0_IRQ_EN** is set to 1, the RTC enters into the interrupt process, write **ALARM0_IRQ_STA_REG** to clear the interrupt pending, and execute the process of waiting for the interrupt.

Resume the interrupt and continue to execute the interrupted process.

3.11.3.5.5 Fanout

The bit0 of **LOSC_OUT_GATING_REG** is set to 1, and external pull-up resistor and voltage is normal, then 32.768kHz square wave can be output.

3.11.3.5.6 Pad Hold

When the corresponding bit of **GPL_HOLD_OUTPUT_REG** and **GPM_HOLD_OUTPUT_REG** is set to 1, the corresponding pin can hold in stable state(high level,low level or high impedance). The function is used to prevent output pin from changing when corresponding power changes.

3.11.3.5.7 RC Calibration Usage Scenario

Power-on: Select non-accurate 32kHz clock divided by internal RC.

Normal scenario: RTC can select 32kHz clock divided by 24MHz, or use calibration clock.If there has fanout requirement, then calibration clock is needed.

Standby or power-off scenario: Select accurate 32kHz generated by DCXO24M calibrates RC clock.

3.11.4 Programming Guidelines

3.11.4.1 RTC Clock Sources Setting

Configure **LOSC_CTRL_REG** to set RTC clock source.
For example: select external 32kHz clock source as RTC clock.

```
writel(0x16aa4000,LOSC_CTRL);       //writing key field
writel(0x16aa4001,LOSC_CTRL);       //select external 32K clock
```

3.11.4.2 Real Time Clock

```
writel(0x00173b3b,RTC_HMS);
writel(1<<22|1<<8|31<<0,RTC_YMD);
readl(RTC_HMS);
readl(RTC_YMD);
```

3.11.4.3 Alarm 0

```
irq_request(GIC_SRC_R_Alarm0,Alm0_handler);
irq_enable(GIC_SRC_R_Alarm0);
writel(1,ALM0_COUNTER);                    //set 1 seconds corresponding to normal mode.
writel(1,ALM0_EN);
```



```
writel(1,ALM_CONFIG); //NMI output
while(!readl(ALM0_IRQ_STA));
writel(1,ALM0_IRQ_EN);
while(readl(ALM0_IRQ_STA));
```

3.11.4.4 Alarm 1

```
irq_request(GIC_SRC_R_Alarm1,Alm1_handler);
irq_enable(GIC_SRC_R_Alarm1);
writel(0,ALM1_WK_HMS);
writel(0x7f,ALM1_EN);
writel(1,ALM1_IRQ_STA);
writel(0x00173b3b | week<<29,RTC_HMS);          //set 1 seconds corresponding to normal mode
while(readl(RTC_HMS)&0xff);
while(!readl(ALM1_IRQ_STA));
writel(1,ALM1_IRQ_EN);
while(readl(ALM1_IRQ_STA));
```

3.11.5 Register List

Module Name	Base Address
RTC	0x07000000

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 Prescalar Register
INTOSC_CLK_AUTO_CALI_REG	0x000C	Internal OSC Clock Auto Calibration Register
RTC_YY_MM_DD_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
ALARM1_WK_HH_MM_SS	0x0040	Alarm 1 Week HMS Register
ALARM1_ENABLE_REG	0x0044	Alarm 1 Enable Register
ALARM1_IRQ_EN	0x0048	Alarm 1 IRQ Enable Register
ALARM1_IRQ_STA_REG	0x004C	Alarm 1 IRQ Status Register
ALARM_CONFIG_REG	0x0050	Alarm Configuration Register
LOSC_OUT_GATING_REG	0x0060	LOSC Output Gating Register
GP_DATA_REG	0x0100 + N*0x04	General Purpose Register (N=0~7)
DCXO_CTRL_REG	0x0160	DCXO Control Register
CALI_CTRL_REG	0x0164	Calibration Control Register

GPL_HOLD_OUTPUT_REG	0x0180	GPL Hold Output Register
RTC_PWR_MODE_SEL_REG	0x0188	RTC POWER MODE SELECT Register
RTC_VIO_REG	0x0190	RTC_VIO Regulate Register
IC_CHARA_REG	0x1F0	IC Characteristic Register
CRY_CONFIG_REG	0x0210	Crypt Configuration Register
CRY_KEY_REG	0x0214	Crypt Key Register
CRY_EN_REG	0x0218	Crypt Enable Register

3.11.6 Register Description

3.11.6.1 LOSC Control Register (Default Value: 0x0000_4010)

Offset:0x0000			Register Name: LOSC_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	W	0x0	KEY_FIELD Key Field This field should be filled with 0x16AA, and then the bit 0 can be written with the new value.
15	/	/	/
14	R/W	0x1	LOSC_AUTO_SWT_EN LOSC Auto Switch Enable 0: Disable 1: Enable
13:10	/	/	/
9	R/W	0x0	ALM_DDHHMMSS_ACCE ALARM DD-HH-MM-SS access After writing the Alarm Week HH-MM-SS Register, this bit is set and it will be cleared until the real writing operation is finished.
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 seconds.
7	R/W	0x0	RTC_YYMMDD_ACCE RTC YY-MM-DD access After writing the RTC YY-MM-DD Register, this bit is set and it will be cleared until the real writing operation is finished. After writing the RTC YY-MM-DD Register, the RTC YY-MM-DD Register will be refreshed for at most one seconds.
6:5	/	/	/
4	R/W	0x1	EXT_LOSC_EN External 32768Hz Crystal Enable 0: Disable 1: Enable

3:2	R/W	0x0	EXT_LOSC_GSM External 32768Hz Crystal GSM 00: Low 01: / 10: / 11 High
1	R/W	0x0	RTC_SRC_SEL. Low Frequency Clock Source Select. 0: CLK32_LOSC 1: CLK32_HOSC
0	R/W	0x0	LOSC_SRC_SEL LOSC Clock source Select. 'N' is the value of Internal OSC Clock Prescalar Register. 0: Low Frequency Clock from 16M RC 1: External 32.768kHz OSC

NOTE

If the bit[9:7] of LOSC_CTRL_REG is set, the corresponding of Alarm 1 Week HH-MM-SS Register, RTC HH-MM-SS Register, RTC YY-MM-DD Register cannot be written.

3.11.6.2 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 0: External 32.768kHz OSC work normally 1: External 32.768kHz OSC work abnormally
1	R/W1C	0x0	LOSC_AUTO_SWT_PEND LOSC auto switch pending 0: No effect 1: Auto switches pending Setting 1 to this bit will clear it.
0	R	0x0	LOSC_SRC_SEL_STA Checking LOSC Clock Source Status. 'N' is the value of Internal OSC Clock Prescalar Register. 0: Low Frequency Clock from 16M RC 1: External 32.768kHz OSC

3.11.6.3 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_CLK_PRESCAL. Internal OSC Clock Prescalar value N. 00000: 1 00001: 2 00010: 3 ..... 11111: 32

3.11.6.4 Internal OSC Clock Auto Calibration Register (Default Value: 0x7A00_0000)

Offset:0x000C			Register Name: INTOSC_CLK_AUTO_CALI_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:26	/	/	/
25:16	RO	0x1e8	32k Calibration Integer Divide Factor
15:4	RO	0x0	32k Calibration Decimal Divide Factor
3	R/W	0x0	Clk16M_RC_enable 0: Auto gating 1: Soft bypass
2	R/W	0x0	RC Calibration Precise Selection 0: 1ms calibration precise 1: 16ms calibration precise
1	R/W	0x0	RC Calibration Enable 0: Close Calibration circuit 1: Open Calibration circuit
0	R/W	0x0	RC Calibration Function Enable 0: Normal RC 1: Calibrated RC

3.11.6.5 RTC YY-MM-DD Register

Offset:0x0010			Register Name: RTC_YY_MM_DD_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	/	/	/
23	R/W	0x0	LEAP Leap Year 0: Not 1: Leap year This bit can not set by hardware. It should be set or cleared by software.
22:16	R/W	UDF	YEAR Year Range from 0~127.
15:12	/	/	/
11:8	R/W	UDF	MONTH Month

			Range from 1~12.
7:5	/	/	/
4:0	R/W	UDF	DAY Day Range from 1~31.

NOTE

If the written value is not from 1 to 31 in Day Area, it turns into 31 automatically. Month Area and Year Area are similar to Day Area.

The number of days in different month may be different.

3.11.6.6 RTC HH-MM-SS Register

Offset:0x0014			Register Name: RTC_HH_MM_SS_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	R/W	0x0	WK_NO Week number 000: Monday 001: Tuesday 010: Wednesday 011: Thursday 100: Friday 101: Saturday 110: Sunday 111: /
28:21	/	/	/
20:16	R/W	UDF	HOUR Range from 0~23
15:14	/	/	/
13:8	R/W	UDF	MINUTE Range from 0~59
7:6	/	/	/
5:0	R/W	UDF	SECOND Range from 0~59

NOTE

If the written value is not from 0 to 59 in Second Area, it turns into 59 automatically. Minute Area and Hour Area are similar to Second Area.

3.11.6.7 Alarm 0 Counter Register (Default Value: 0x0000_0000)

Offset:0x0020			Register Name: ALARM0_COUNTER_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	ALARM0_COUNTER

			Alarm 0 Counter is based on second. If the second is set to 0, it will be 1 seconds in fact.

3.11.6.8 Alarm 0 Current Value Register

Offset:0x0024			Register Name: ALARM0_CUR_VLU_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	UDF	ALARM0_CUR_VLU Check Alarm 0 Counter Current Values. If the second is set to 0, it will be 1 seconds in fact.

3.11.6.9 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 If this bit is set to "1", the valid bits of Alarm 0 Counter Register will down count to zero, and the alarm pending bit will be set to "1". 0: Disable 1: Enable

3.11.6.10 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

3.11.6.11 Alarm 0 IRQ Status Register (Default Value: 0x0000_0000)

Offset:0x0030			Register Name: ALARM0_IRQ_STA_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W1C	0x0	ALARM0_IRQ_PEND Alarm 0 IRQ Pending bit 0: No effect 1: Pending, alarm 0 counter value is reached

			If alarm 0 irq enable is set to 1, the pending bit will be sent to the interrupt controller.

3.11.6.12 Alarm 1 Week HH-MM-SS Register

Offset:0x0040			Register Name: ALARM1_WK_HH_MM_SS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:21	/	/	/
20:16	R/W	UDF	HOUR Range from 0~23.
15:14	/	/	/
13:8	R/W	UDF	MINUTE Range from 0~59.
7:6	/	/	/
5:0	R/W	UDF	SECOND Range from 0~59.

NOTE

If the written value is not from 0 to 59 in Second Area, it turns into 59 automatically. Minute Area and Hour Area are similar to Second Area.

3.11.6.13 Alarm 1 Enable Register (Default Value: 0x0000_0000)

Offset:0x0044			Register Name: ALARM1_EN_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:7	/	/	/
6	R/W	0x0	WK6_ALM1_EN Week 6 (Sunday) Alarm 1 Enable 0: Disable 1: Enable If this bit is set to “1”, only when the valid bits of Alarm 1 Week HH-MM-SS Register is equal to the bit[20:0] of RTC HH-MM-SS Register and the bit[31:29] of RTC HH-MM-SS Register is 6, the week 6 alarm irq pending bit will be set to “1”.
5	R/W	0x0	WK5_ALM1_EN Week 5 (Saturday) Alarm 1 Enable 0: Disable 1: Enable If this bit is set to “1”, only when the valid bits of Alarm 1 Week HH-MM-SS Register is equal to the bit[20:0] of RTC HH-MM-SS Register and the bit[31:29] of RTC HH-MM-SS Register is 5, the week 5 alarm irq pending bit will be set to “1”.
4	R/W	0x0	WK4_ALM1_EN Week 4 (Friday) Alarm 1 Enable

			0: Disable 1: Enable If this bit is set to “1”, only when the valid bits of Alarm 1 Week HH-MM-SS Register is equal to the bit[20:0] of RTC HH-MM-SS Register and the register bit[31:29] of RTC HH-MM-SS Register is 4, the week 4 alarm irq pending bit will be set to “1”.
3	R/W	0x0	WK3_ALM1_EN Week 3 (Thursday) Alarm 1 Enable 0: Disable 1: Enable If this bit is set to “1”, only when the valid bits of Alarm 1 Week HH-MM-SS Register is equal to the bit[20:0] of RTC HH-MM-SS Register and the bit[31:29] of RTC HH-MM-SS Register is 3, the week 3 alarm irq pending bit will be set to “1”.
2	R/W	0x0	WK2_ALM1_EN Week 2 (Wednesday) Alarm 1 Enable 0: Disable 1: Enable If this bit is set to “1”, only when the valid bits of Alarm 1 Week HH-MM-SS Register is equal to the bit[20:0] of RTC HH-MM-SS Register and the bit[31:29] of RTC HH-MM-SS Register is 2, the week 2 alarm irq pending bit will be set to “1”.
1	R/W	0x0	WK1_ALM1_EN Week 1 (Tuesday) Alarm 1 Enable 0: Disable 1: Enable If this bit is set to “1”, only when the valid bits of Alarm 1 Week HH-MM-SS Register is equal to the bit[20:0] of RTC HH-MM-SS Register and the bit[31:29] of RTC HH-MM-SS Register is 1, the week 1 alarm irq pending bit will be set to “1”.
0	R/W	0x0	WK0_ALM1_EN Week 0 (Monday) Alarm 1 Enable 0: Disable 1: Enable If this bit is set to “1”, only when the valid bits of Alarm 1 Week HH-MM-SS Register is equal to the bit[20:0] of RTC HH-MM-SS Register and the bit[31:29] of RTC HH-MM-SS Register is 0, the week 0 alarm irq pending bit will be set to “1”.

3.11.6.14 Alarm 1 IRQ Enable Register (Default Value: 0x0000_0000)

Offset:0x0048			**Register Name: ALARM1_IRQ_EN**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	ALARM1_IRQ_EN

			Alarm 1 IRQ Enable 0: Disable 1: Enable

3.11.6.15 Alarm 1 IRQ Status Register (Default Value: 0x0000_0000)

Offset:0x004C			Register Name: ALARM1_IRQ_STA_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W1C	0x0	ALARM1_WEEK_IRQ_PEND Alarm 1 Week (0/1/2/3/4/5/6) IRQ Pending. 0: No effect 1: Pending, week counter value is reached If alarm 1 week irq enable is set to 1, the pending bit will be sent to the interrupt controller.

3.11.6.16 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

3.11.6.17 LOSC Output Gating Register (Default Value: 0x0000_0000)

Offset:0x0060			Register Name: LOSC_OUT_GATING_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	LOSC_OUT_GATING Configuration of LOSC output, and without LOSC output by default. 0: Disable LOSC output gating 1: Enable LOSC output gating

3.11.6.18 General Purpose Register (Default Value: 0x0000_0000)

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

NOTE

General purpose register 0~7 value can be stored if the RTC-VIO is larger than 1.0V.

3.11.6.19 DCXO Control Register (Default Value: 0x083F_10F2)

Offset:0x0160			Register Name: DCXO_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	/	/	/
27:24	R/W	0x8	DCXO_ICTRL DCXO current control value
23	/	/	/
22:16	R/W	0x3F	DCXO_TRIM DCXO cap array value Capacity cell is 55fF
15:14	/	/	/
13	R	UDF	PLL_INPUT_CLK_SEL 0: HOSC 1: PLL_24M
12:8	R/W	0x10	DCXO_BG DCXO bandgap output voltage
7	R/W	0x1	DCXO_LDO_INRUSHB DCXO LDO driving capacity signal, active high
6	R/W	0x1	XTAL_MODE Xtal mode enable signal, active high 0: For external clk input mode 1: For normal mode
5:4	R/W	0x3	DCXO_RFCLK_ENHANCE DCXO rfclk enhance Enhance driving capacity of output OUT_RF_REFCLK, 0x0 for 5pF, 0x1 for 10pF, 0x2 for 15pF, 0x3 for 20Pf.
3	UDF	UDF	OSC_CLK_SRC_SEL.(Pad select) High frequency clock source select 0: XO24M 1: DCXO
2	/	/	/
1	R/W	0x1	DCXO_EN DCXO enable 1: Enable 0: Disable
0	/	/	/

3.11.6.20 Calibration Control Register (Default Value: 0x0000_0003)

Offset:0x0164			Register Name: CALI_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	WAKEUP_DCXO_EN Wake up DCXO circuit enable
30:15	/	/	/
16	R/W	0x0	WAKEUP_READY_SLEEP_MODE Calibration wake up ready sleep mode 0: Disable 1: Enable
15:12	R/W	0x0	TIMER FOR READY SLEEP Total timer for ready sleep 0000: 15s 0001: 30s 0010: 45s 0011: 60s 0100: 90s 0101: 120s 0110: 150s Others: /
11:8	R/W	0x0	WAKEUP_CNT FOR READY SLEEP Wake up counter for ready sleep 0000: 250ms 0001: 500ms 0010: 750ms 0011: 1s 0100: 1.25s 0101: 1.5s 0110: 1.75s 0111: 2s 1000: 2.25s 1001: 2.5s 1010: 2.75s 1011: 3s 1100: 3.25s 1101: 3.5s 1110: 3.75s 1111: 4s
7:4	R/W	0x0	WAKEUP_CNT FOR SLEEP Wake up counter for sleep 0000: 250ms 0001: 500ms 0010: 1s 0011: 2s 0100: 3s

			0101: 4s 0110: 5s 0111: 6s 1000: 7s 1001: 8s 1010: 9s 1011: 10s 1100: 11s 1101: 12s 1110: 30s 1111: 60s
3:0	R/W	0x3	WAIT DCXO SEL Select for DCXO active after DCXO enable 0000:1ms 0001:2ms 0010:3ms 0011:4ms ... 1111: 16ms

3.11.6.21 GPL Hold Output Register (Default Value: 0x0000_0000)

Offset:0x0180			Register Name: GPL_HOLD_OUTPUT_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:13	/	/	/
12	R/W	0x0	GPL12_HOLD_OUTPUT Hold the output of GPIOL12 when the power of system changes. The output must be low level (0) or high level (1) or High-Z; any other outputs may not hold on. 0: Hold disable 1: Hold enable
11	R/W	0x0	GPL11_HOLD_OUTPUT Hold the output of GPIOL11 when the power of system changes. The output must be low level (0) or high level (1) or High-Z; any other outputs may not hold on. 0: Hold disable 1: Hold enable
10	R/W	0x0	GPL10_HOLD_OUTPUT Hold the output of GPIOL10 when the power of system changes. The output must be low level (0) or high level (1) or High-Z; any other outputs may not hold on. 0: Hold disable 1: Hold enable
9	R/W	0x0	GPL9_HOLD_OUTPUT Hold the output of GPIOL9 when the power of system changes. The output

			must be low level (0) or high level (1) or High-Z; any other outputs may not hold on. 0: Hold disable 1: Hold enable
8	R/W	0x0	GPL8_HOLD_OUTPUT Hold the output of GPIOL8 when the power of system changes. The output must be low level (0) or high level (1) or High-Z; any other outputs may not hold on. 0: Hold disable 1: Hold enable
7	R/W	0x0	GPL7_HOLD_OUTPUT Hold the output of GPIOL7 when the power of system changes. The output must be low level (0) or high level (1) or High-Z; any other outputs may not hold on. 0: Hold disable 1: Hold enable
6	R/W	0x0	GPL6_HOLD_OUTPUT Hold the output of GPIOL6 when the power of system changes. The output must be low level (0) or high level (1) or High-Z; any other outputs may not hold on. 0: Hold disable 1: Hold enable
5	R/W	0x0	GPL5_HOLD_OUTPUT Hold the output of GPIOL5 when the power of system changes. The output must be low level (0) or high level (1) or High-Z; any other outputs may not hold on. 0: Hold disable 1: Hold enable
4	R/W	0x0	GPL4_HOLD_OUTPUT Hold the output of GPIOL4 when the power of system changes. The output must be low level (0) or high level (1) or High-Z; any other output may not hold on. 0: Hold disable 1: Hold enable
3	R/W	0x0	GPL3_HOLD_OUTPUT Hold the output of GPIOL3 when the power of system changes. The output must be low level (0) or high level (1) or High-Z; any other outputs may not hold on. 0: Hold disable 1: Hold enable
2	R/W	0x0	GPL2_HOLD_OUTPUT Hold the output of GPIOL2 when the power of system changes. The output must be low level (0) or high level (1) or High-Z; any other outputs may not hold on. 0: Hold disable 1: Hold enable

1	R/W	0x0	GPL1_HOLD_OUTPUT Hold the output of GPIOL1 when the power of system changes. The output must be low level (0) or high level (1) or High-Z; any other outputs may not hold on. 0: Hold disable 1: Hold enable
0	R/W	0x0	GPL0_HOLD_OUTPUT Hold the output of GPIOL0 when the power of system changes. The output must be low level (0) or high level (1) or High-Z; any other outputs may not hold on. 0: Hold disable 1: Hold enable

3.11.6.22 RTC Power Mode Select Register (Default Value: 0x0000_0001)

Offset:0x0188			Register Name: RTC_PWR_MODE_SEL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x1	RTC_POW_MOD_SELECT VCC-RTC POWER MODE SELECT 0: 3.3V 1:1.8V

3.11.6.23 RTC_VIO Regulation Register (Default Value: 0x0000_0004)

Offset:0x0190			Register Name:RTC_VIO_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:5	/	/	/
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 0.6V to 1.3V , and the regulation step is 0.1V. 000: 1.0V 001: 0.6V 010: 0.7V 011: 0.8V 100: 0.9V 101: 1.1V 110: 1.2V 111: 1.3V

3.11.6.24 IC Characteristic Register (Default Value: 0x0000_0000)

Offset:0x1F0			Register Name: IC_CHARA_REG
Bit	Read/Write	Default/Hex	Description
31:16	R/W	0x0	IC_CHARA. Key Field. Should be written at value 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.

3.11.6.25 Crypto Configuration Register (Default Value: 0x0000_0000)

Offset:0x0210			**Register Name: CRY_CONFIG_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x0	KEY_FIELD Key Field If you want to read or write Crypt Key Register/Crypt Enable Register , you should write 0x1689 in these bits.

3.11.6.26 Crypto Key Register (Default Value: 0x0000_0000)

Offset:0x0214			**Register Name: CRY_KEY_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	CRY_KEY Crypto Key

3.11.6.27 Crypto Enable Register (Default Value: 0x0000_0000)

Offset:0x0218			**Register Name: CRY_EN_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	CRY_EN Crypto Enable

3.12 Thermal Sensor Controller

3.12.1 Overview

Thermal sensors have became common elements in wide range of modern system on chip (SOC) platform. Thermal sensors are used to constantly monitor the temperature on the chip.

The Thermal Sensor Controller(THS) embeds 2 thermal sensors, sensor0 for GPU,sensor1 for CPU. Thermal sensors can generate interrupt to SW to lower temperature via DVFS, on reaching a certain thermal threshold.

The THS has the following features:

- Temperature Accuracy : ±3℃ from 0℃ to +100℃, ±5℃ from -25℃ to +125℃
- Power supply voltage:1.8V
- Averaging filter for thermal sensor reading
- Supports over-temperature protection interrupt and over-temperature alarm interrupt

3.12.2 Block Diagram

Figure 3-47 shows a block diagram of the Thermal Sensor Controller.

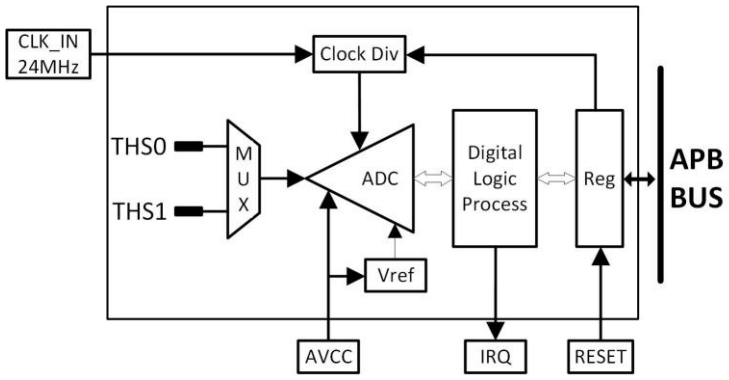


Figure3- 32. Thermal Sensor Controller Block Diagram

3.12.3 Operations and Functional Descriptions

3.12.3.1 Clock Sources

Thermal Sensor Controller gets two clock source. Table 3-11 describes the clock source for Thermal Sensor Controller. Users can see **Clock Controller Unit(CCU)** for clock setting, configuration and gating information.

Table3- 12. Thermal Sensor Controller Clock Sources

Clock Sources	Description
HOSC	HOSC
PLL_24MHz	PLL_24MHz(default)

3.12.3.2 Timing Requirements

CLK_IN = 24MHz
CONV_TIME(Conversion Time) = 1/(24MHz/14Cycles) = 0.583 (us)
TACQ> 1/(24MHz/24Cycles)
THERMAL_PER > ADC Sample Frequency > TACQ+CONV_TIME

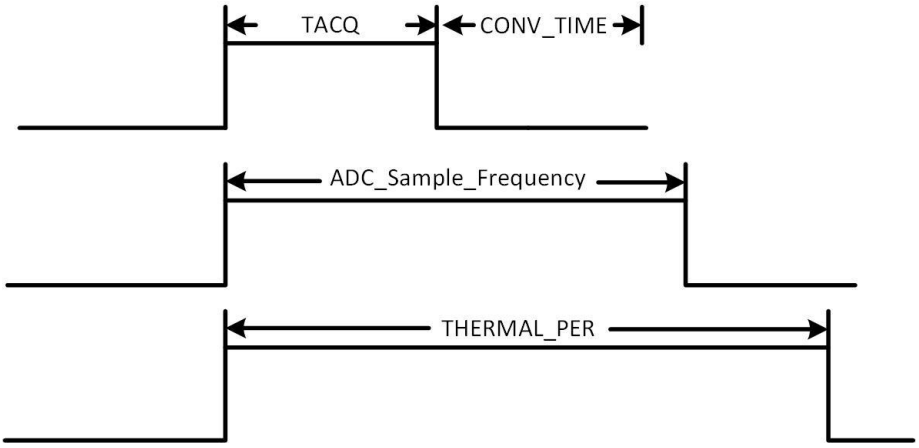


Figure3- 33. Thermal Sensor Time Requirement

3.12.3.3 Interrupt

Thermal Sensor Controller has four interrupt source, such as DATA_IRQ , SHUTDOWN_IRQ, ALARM_IRQ and ALARM_OFF_IRQ. Figure 3-49 shows the thermal sensor interrupt sources.

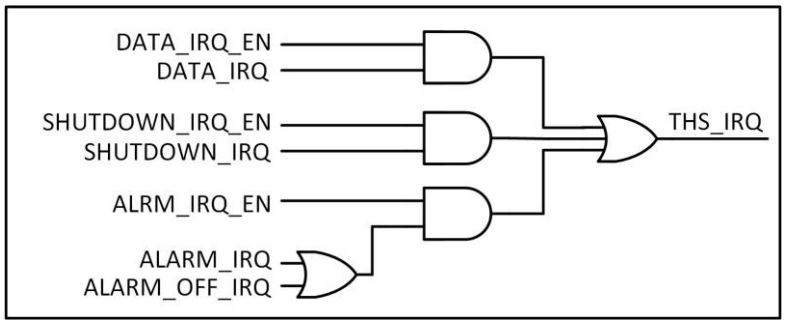


Figure3- 34. Thermal Sensor Controller Interrupt Source

When temperature is higher than Alarm_Threshold, ALARM_IRQ is generated. When temperature is lower than Alarm_Off_Thershold, ALARM_OFF_IRQ is generated. ALARM_OFF_IRQ is fall edge trigger.

3.12.3.4 THS Formula

THS formula: T = (THS_DATA - 2794) / -14.882

3.12.4 Programming Guidelines

The THS is initialized as follows.

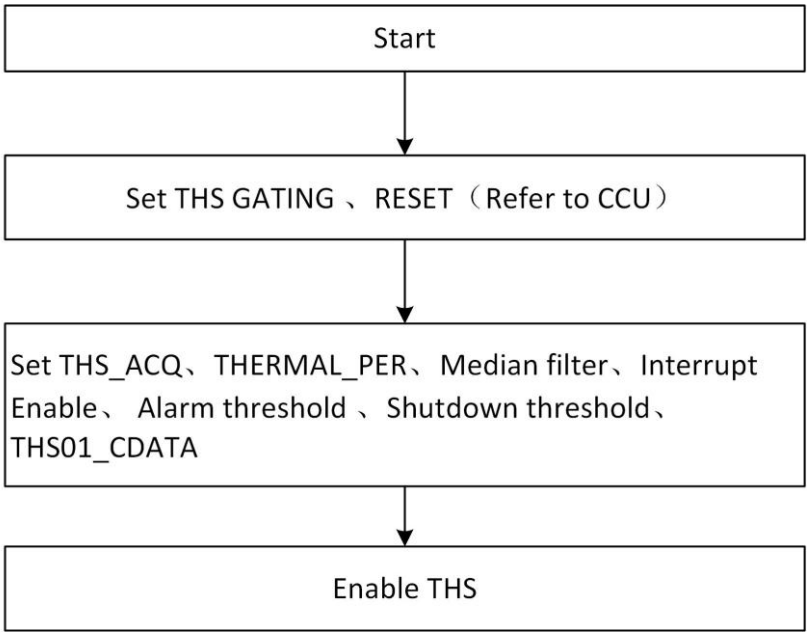


Figure3- 35. THS Initialization Process

The formula of THS is y=-ax+b. In FT stage, THS is calibrated according to ambient temperature, the calibration value is written in EFUSE. Please refer to SID Spec about EFUSE information.

Before enabling THS, read EFUSE value and write the value to **THS01_CDATA**.

(1).Query Mode

Step1: Write 1 to the bit16 of **THS_BGR_REG** to dessert reset.

Step2: Write 1 to the bit0 of **THS_BGR_REG** to open THS clock.

Step3: Write 0x2F to the bit[15:0] of **THS_CTRL** to set ADC acquire time.

Step4: Write 0x1DF to the bit[31:16] of **THS_CTRL** to set ADC sample frequency divider.

Step5: Write 0x3A to the bit[31:12] of **THS_PER** to set THS work period.

Step6: Write 1 to the bit2 of **THS_FILTER** to enable temperature convert filter.

Step7: Write 1 to the bit[1:0] of **THS_FILTER** to select filter type.

Step8: Read THS efuse value from SID, then write the efuse value to **THS01_CDATA** to calibrate THS0/THS1.

Step9: Write 0x3 to the bit[1:0] 0f **THS_EN** to enable THS0/THS1.

Step10: Read the bit[1:0] of **THS_DATA_INTS**, if the value is 0x3, temperature conversion is complete.

Step11: Read the bit[11:0] of **THS0_DATA/THS1_DATA**, calculate THS0/THS1 temperature based on THS formula.

(2). Interrupt Mode

Step1: Write 1 to the bit16 of **THS_BGR_REG** to dessert reset.

Step2: Write 1 to the bit0 of **THS_BGR_REG** to open THS clock.

Step3: Write 0x2F to the bit[15:0] of **THS_CTRL** to set ADC acquire time.

Step4: Write 0x1DF to the bit[31:16] of **THS_CTRL** to set ADC sample frequency divider.

Step5: Write 0x3A to the bit[31:12] of **THS_PER** to set THS work period.

Step6: Write 1 to the bit2 of **THS_FILTER** to enable temperature convert filter.

Step7: Write 1 to the bit[1:0] of **THS_FILTER** to select filter type.

Step8: Read THS efuse value from SID, then write the efuse value to **THS01_CDATA** to calibrate THS0/THS1.

Step9: Write 0x3 to the bit[1:0] of **THS_DATA_INTC** to enable the interrupt of THS0/THS1.

Step10: Set GIC interface based on IRQ 33, write the bit[1] of the 0x03021104 register to 1.

Step11: Put interrupt handler address into interrupt vector table.

Step12: Write 0x3 to the bit[1:0] 0f **THS_EN** to enable THS0/THS1.

Step13: Read the bit[1:0] of **THS_DATA_INTS**, if the value is 0x3, temperature conversion is complete.
Step14: Read the bit[11:0] of **THS0_DATA/THS1_DATA**, calculate THS0/THS1 temperature based on THS formula.

3.12.5 Register List

Module Name	Base Address
Thermal Sensor	0x05070400

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_ALARMO_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
THS0_ALARM_CTRL	0x0040	THS0 Alarm threshold Control Register
THS1_ALARM_CTRL	0x0044	THS1 Alarm threshold Control Register
THS01_SHUTDOWN_CTRL	0x0080	THS0&1 Alarm threshold Control Register
THS01_CDATA	0x00A0	THS0&1 Calibration Data
THS0_DATA	0x00C0	THS0 Data Register
THS1_DATA	0x00C4	THS1 Data Register

3.12.6 Register Description

3.12.6.1 THS Control Register(Default Value : 0x01DF_002F)

Offset: 0x0000			Register Name: THS_CTRL
Bit	**Rear/Write**	**Default/Hex**	**Description**
31:16	R/W	0x1DF (50kHz)	FS_DIV ADC Sample Frequency Divider CLK_IN/(N+1) (N > 0x17)
15:0	R/W	0x2F(2us)	TACQ ADC Acquire Time CLK_IN/(n+1)

3.12.6.2 THS Enable Register(Default Value : 0x0000_0000)

Offset: 0x0004	Register Name: THS_EN

Bit	Rear/Write	Default/Hex	Description
31:2	/	/	/
1	R/W	0x0	THS1_EN Enable temperature measurement sensor1 0:Disable 1:Enable
0	R/W	0x0	THS0_EN Enable temperature measurement sensor0 0:Disable 1:Enable

3.12.6.3 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(10ms)	THERMAL_PER 4096*(n+1)/CLK_IN
11:0	/	/	/

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

Offset: 0x0010			Register Name: THS_DATA_INTC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1	R/W	0x0	THS1_DATA_IRQ_EN Selects temperature measurement data of sensor1 0:Disable 1:Enable
0	R/W	0x0	THS0_DATA_IRQ_EN Selects temperature measurement data of sensor0 0:Disable 1:Enable

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

Offset: 0x0014			Register Name: THS_SHUT_INTC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1	R/W	0x0	SHUT_INT1_EN Selects shutdown interrupt for sensor1 0:Disable 1:Enable
0	R/W	0x0	SHUT_INT0_EN

			Selects shutdown interrupt for sensor0 0:Disable 1:Enable

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

Offset: 0x0018			Register Name: THS_ALARM_INTC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1	R/W	0x0	ALARM_INT1_EN Selects alarm interrupt for sensor1 0:Disable 1:Enable
0	R/W	0x0	ALARM_INT0_EN Selects alarm interrupt for sensor0 0:Disable 1:Enable

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

Offset: 0x0020			Register Name: THS_DATA_INTS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1	R/W1C	0x0	THS1_DATA_IRQ_STS Data interrupt status for sensor1 Write ‘1’ to clear this interrupt.
0	R/W1C	0x0	THS0_DATA_IRQ_STS Data interrupt status for sensor0 Write ‘1’ to clear this interrupt.

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

Offset: 0x0024			Register Name: THS_SHUT_INTS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1	R/W1C	0x0	SHUT_INT1_STS Shutdown interrupt status for sensor1 Write ‘1’ to clear this interrupt.
0	R/W1C	0x0	SHUT_INT0_STS Shutdown interrupt status for sensor0 Write ‘1’ to clear this interrupt .

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

Offset: 0x0028			Register Name: THS_ALARMO_INTS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1	R/W1C	0x0	ALARM_OFF1_STS Alarm interrupt off pending for sensor1 Write '1' to clear this interrupt.
0	R/W1C	0x0	ALARM_OFF0_STS Alarm interrupt off pending for sensor0 Write '1' to clear this interrupt.

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

Offset: 0x002C			Register Name: THS_ALARM_INTS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1	R/W1C	0x0	ALARM_INT1_STS Alarm interrupt pending for sensor1 Write '1' to clear this interrupt.
0	R/W1C	0x0	ALARM_INT0_STS Alarm interrupt pending for sensor0 Write '1' to clear this interrupt.

3.12.6.11 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: Disable 1: Enable
1:0	R/W	0x1	FILTER_TYPE Average Filter Type 00: 2 01: 4 10: 8 11: 16

3.12.6.12 THS0 Alarm Threshold Control Register(Default Value: 0x05A0_0684)

Offset: 0x0040	Register Name: THS0_ALARM_CTRL

Bit	Read/Write	Default/Hex	Description
31:28	/	/	/
27:16	R/W	0x5A0	ALARM0_T_HOT Thermal Sensor0 Alarm threshold for hot temperature
15:12	/	/	/
11:0	R/W	0x684	ALARM0_T_HYST Thermal Sensor0 Alarm threshold for hysteresis temperature

3.12.6.13 THS1 Alarm Threshold Control Register (Default Value: 0x05A0_0684)

Offset: 0x0044			Register Name: THS1_ALARM_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	/	/	/
27:16	R/W	0x5A0	ALARM1_T_HOT Thermal sensor1 Alarm threshold for hot temperature
15:12	/	/	/
11:0	R/W	0x684	ALARM1_T_HYST Thermal sensor1 Alarm threshold for hysteresis temperature

3.12.6.14 THS0&1 Shutdown Threshold Control Register (Default Value: 0x04E9_04E9)

Offset: 0x0080			Register Name: THS01_SHUTDOWN_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	/	/	/
27:16	R/W	0x4E9	SHUT1_T_HOT Thermal Sensor1 Shutdown threshold for hot temperature
15:12	/	/	/
11:0	R/W	0x4E9	SHUT0_T_HOT Thermal Sensor0 Shutdown threshold for hot temperature

3.12.6.15 THS0&1 Calibration Data Register (Default Value: 0x0800_0800)

Offset: 0x00A0			Register Name: THS01_CDATA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	/	/	/
27:16	R/W	0x800	THS1_CDATA Thermal Sensor1 calibration data
15:12	/	/	/
11:0	R/W	0x800	THS0_CDATA Thermal Sensor0 calibration data

3.12.6.16 THS0 Data Register(Default Value: 0x0000_0000)

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

3.12.6.17 THS1 Data Register(Default Value: 0x0000_0000)

Offset: 0x00C4			Register Name: THS1_DATA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:12	/	/	/
11:0	R	0x0	THS1_DATA Temperature measurement data of sensor1

3.13 PSI

3.13.1 Overview

PSI（Peripheral System Interconnect）is a peripheral bus interconnect device based on AHB and APB protocol, which supports 16 AHB master and 16 slave bus. The type of slave bus can be AHB bus or APB bus. Each bus supports 64 slave devices.

3.13.2 Block Diagram

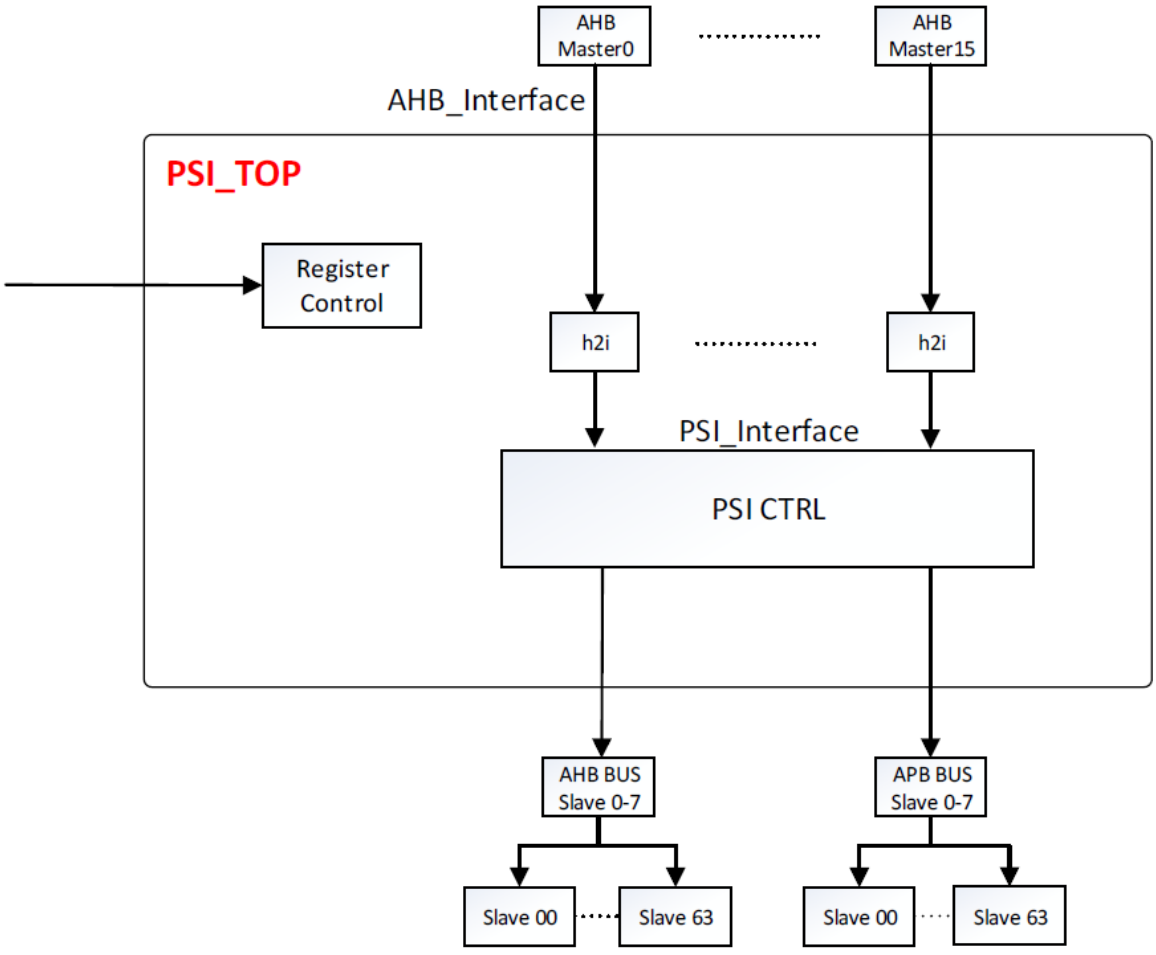


Figure3- 36. PSI Block Diagram

3.14 Message Box

3.14.1 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 channel
- FIFO depth is 4 x 32 bits
- The communication parties are CPUS and CPUX
- Interrupt alarm function

3.14.2 Block Diagram

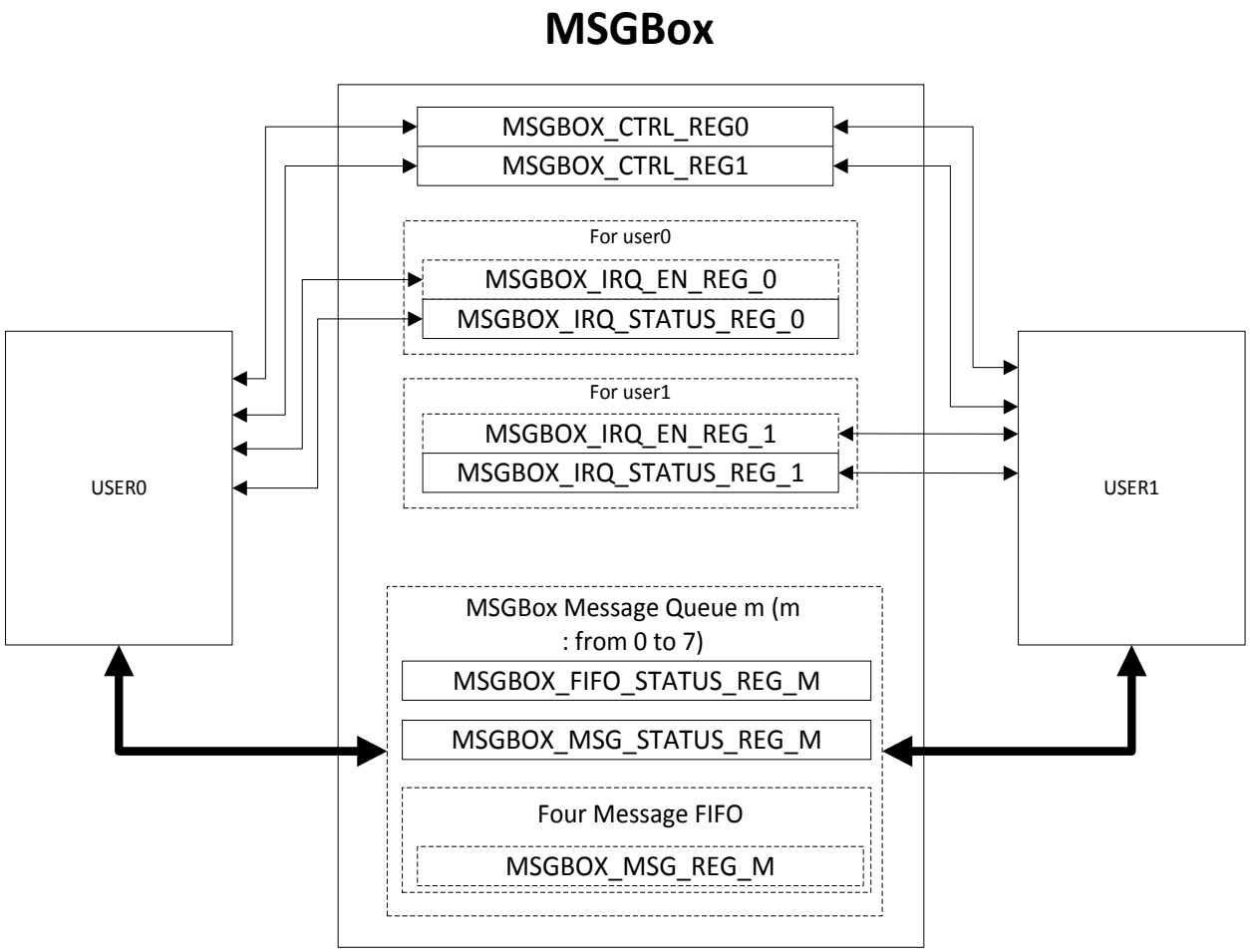


Figure3- 37. Message Box Block Diagram

3.14.3 Operations and Functional Descriptions

3.14.3.1 Clock and Reset

MSGBOX is on AHB1 bus. To access MSGBOX, perform the following steps about AHB1 bus:
Step1: De-assert MSGBOX reset signal.
Step2: Open MSGBOX gating signal.

3.14.3.2 Typical Application

Two different CPU can build communication by configuring MSGBOX. The communication parties have 8 bidirectional channels. If a party is receiver, then another is transmitter. During communication process, the current status can be judged through interrupt or FIFO status.

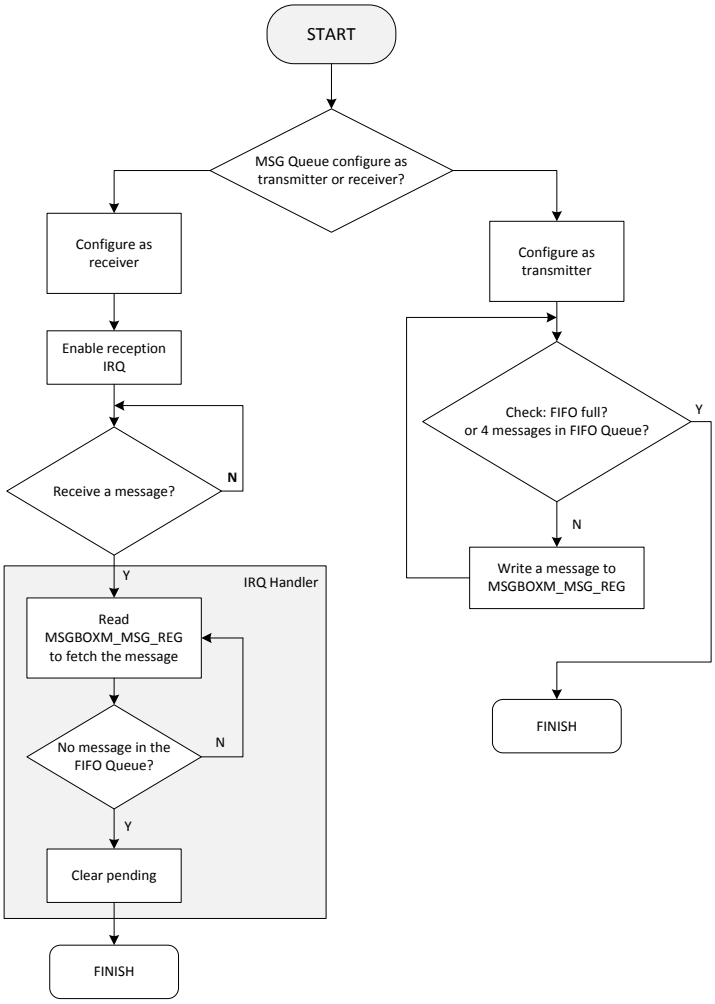


Figure3- 38. Message Box Typical Application Chart

3.14.3.3 Function Implemention

3.14.3.3.1 Transmitter and Receiver Mode

User0 and User1 can be configured as transmitter or receiver, but User0 and User1 cannot configure as same mode in the same channel, that is, User0 is transmitter, User1 must be receiver; User0 is receiver, User1 must be transmitter.

3.14.3.3.2 Interrupt

Interrupt has two types:

- As receiver, when received new information, the interrupt signal can generate.
- As transmitter, when channel FIFO is not full, the interrupt signal can generate.

Eight channels can configure the interrupt enable bit individually, but interrupt controller only has a MSGBOX interrupt number.

3.14.3.3.3 FIFO Status

- When channel FIFO is not full, the value of **FIFO_FULL_FLAG** is 0, at this time ,FIFO can execute write operation.
- When channel FIFO is full, the value of **FIFO_FULL_FLAG** is 1, at this time, if writing data again to FIFO, the first data in FIFO can be overrided.
- FIFO status can be read by **MSGBOXM_MSG_STATUS_REG**.

3.14.3.3.4 Debug Mode

- In debug mode, User0 can transmit data to User0, User1 can transmit data to User1.
- In debug mode, FIFO function will close.

3.14.3.4 Operating Mode

3.14.3.4.1 Transfer Mode Configuration

- Queue n (n=0~3)transmitter mode: Write 1 to the bit[8*n+4] of **MSGBOX_CTRL_REG0**.
- Queue m (m=4~7)transmitter mode : Write 1 to the bit[8*(m-4)+4] of **MSGBOX_CTRL_REG1**.
- Queue n (n=0~3) receiver mode: Write 1 to the bit[8*n] of **MSGBOX_CTRL_REG0**.
- Queue m (m=4~7) receiver mode : Write 1 to the bit[8*(m-4)] of **MSGBOX_CTRL_REG1**.

3.14.3.4.2 Interrupt Check Transfer Status

Configure transmitter and receiver mode through **chapter 3.13.3.4.1. Transfer Mode Configuration**.

Interrupt enable bit: Configure the interrupt enable bit of transmitter/receiver through **MSGBOX_IRQ_EN_REG**.

When FIFO is not full, an interrupt pending generates to remind the transmitter to transmit data, at this time, to write data to FIFO in interrupt handler ,and clear the pending bit and the enable bit of *Transmitter IRQ*.

When FIFO has new data, an interrupt pending generates to remind the receiver to receive data, at this time, to read data from FIFO in interrupt handler, and clear the pending bit and the enable bit of *Receiver IRQ*.

3.14.3.4.3 FIFO Check Transfer Status

Configure transmitter and receiver mode through **chapter 3.14.3.4.1. Transfer Mode Configuration**.

When FIFO is not full, the transmitter fills FIFO to 4*32 bits.

When the receiver considers FIFO is full, then the receiver reads FIFO data, and reads **MSGBOXM_MSG_STATUS_REG** to require the current FIFO number.

3.14.3.4.4 Debug

To use MSGBOX in debug mode, performs the following steps:

Write 1 to the bit0 of **MSGBOX_DEBUG_REG**.

The control bit of the corresponding channel is set to 1.

3.14.4 Programming Guidelines

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 user1 and user0 working process is as follows.

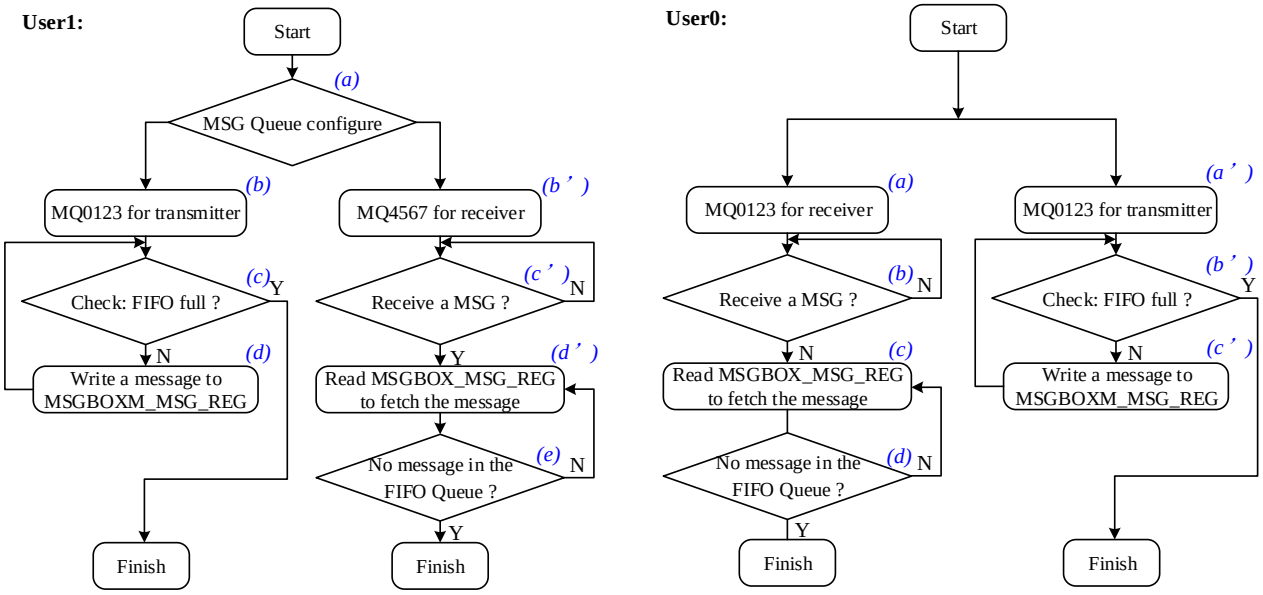


Figure3- 39. User1 and User0 Working Process

User1:

(a) The user1 is as the transmitter of MQ0123 and as the receiver of MQ4567.
(b) Queue n (n=0~3) transmitter mode: write 1 to MSGBOX_CTRL_REG0 bit[8*n+4].
(b') Queue m (m=4~7) receiver mode: write 1 to MSGBOX_CTRL_REG1 bit[8*(m-4)].
(c) Check whether the FIFO is full by the status of the MSGBOXM_FIFO_STATUS_REG. If the FIFO is full, the program finish, otherwise, go to step (d).
(c') Check whether to receive a message by the status of the MSGBOXM_MSG_STATUS_REG. If you donot receive, then continue to wait, otherwise, go to step (d').
(d) Write a message to MSGBOXM_MSG_REG.
(d') Read MSGBOX_MSG_REG to fetch the message.
(e) If there is no message in FIFO Queue, step up to (d'), otherwise, the program finish.

User0:
(a) The user0 is as the receiver of MQ0123.
(a') The user0 is as the transmitter of MQ4567.
(b) Check whether to receive a message by the status of the MSGBOXM_MSG_STATUS_REG. If you donot receive, then continue to wait, otherwise, go to step (c).
(b') Check whether the FIFO is full by the status of MSGBOXM_FIFO_STATUS_REG. If the FIFO is full, the program finish, otherwise, step up to (c').
(c) Read MSGB OX_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 finish.

----------***Example code***----------

User1:

```
//user1 as transmitter of MQ0123 and as receiver of MQ4567
rdata = get_wvalue(MSGBOX_CTRL0) | (1<<4) | (1<<12) | (1<<20) | (1<<28);
put_wvalue(MSGBOX_CTRL0, rdata);
rdata = get_wvalue(MSGBOX_CTRL0) & ~(1<<0) & ~(1<<8) & ~(1<<16) & ~(1<<24);
put_wvalue(MSGBOX_CTRL0, rdata);
rdata = get_wvalue(MSGBOX_CTRL1) & ~(1<<4) & ~(1<<12) & ~(1<<20) & ~(1<<28);
put_wvalue(MSGBOX_CTRL1, rdata);
rdata = get_wvalue(MSGBOX_CTRL1) | (1<<0) | (1<<8) | (1<<16) | (1<<24);
put_wvalue(MSGBOX_CTRL1, rdata);
put_wvalue(PIO_C_DATA,0x55); //to notify user0 that msgbox attribute has changed
for (i=0; i<4; i++)
{
    int j=0;
    for (j=0; j<4; j++)
    {
        if(get_wvalue(MSGBOX_FIFO_STATUS_(i))==0) //fifo i is full?
        {
            put_wvalue(MSGBOX_MSG_(i), 0x12345678-j-i*0x10);
        }
    }
    //aw_delay(5);
    }
//wait for corresponding of user0 receiving msg from MQ4567
```



```
printk("cpu0 step 0!\n");
while(get_wvalue(MSGBOX_MSG_STATUS_(7)) != 4);
printk("cpu0 step 0!\n");
for(i=4;i<8;i++) //test whether the transmit data is right
{
    int j=0;
    for(j=0;j<4;j++)
    {
        if(get_wvalue(MSGBOX_MSG_STATUS_(i)) == (4-j))
        {
            rdata = get_wvalue(MSGBOX_MSG_(i));
            //put_wvalue(0x28040+0x10*(i-4)+4*j,rdata);
            if(rdata != (0x12345678-j-(i-4)*0x10))
                return -1;
        }
    }
}
```

User0:

```
//user0 as receiver of MQ0123 and as transmitter of MQ4567
while(get_wvalue(MSGBOX_MSG_STATUS_(3))!=4); //wait for receiving the msg from user1
//cpu0 receive done, and then, test receive data whether right!
for(i=0;i<4;i++)
{
    for(j=0;j<4;j++)
    {
        if(get_wvalue(MSGBOX_MSG_STATUS_(i))==(4-j))
        {
            data[i][j] = get_wvalue(MSGBOX_MSG_(i));
            data1[i][j] = data[i][j];//store the data from mq0123
        }
    }
}
//cpu0 transmitter of MQ4567
for(i=4;i<8;i++)
{
    for(j=0;j<4;j++)
    {
        if(get_wvalue(MSGBOX_FIFO_STATUS_(i))==0)
        {
            put_wvalue(MSGBOX_MSG_(i),data1[i-4][j]);//Send back what has received
        }
    }
}
```


3.14.5 Register List

Module Name	Base Address
MSGBOX	0x03003000

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*0x20	IRQ Enable for User n (n=0,1)
MSGBOXU_IRQ_STATUS_REG	0x0050+n*0x20	IRQ Status for User n (n=0,1)
MSGBOXM_FIFO_STATUS_REG	0x0100+N*0x04	FIFO Status for Message Queue N(N = 0~7)
MSGBOXM_MSG_STATUS_REG	0x0140+N*0x04	Message Status for Message Queue N(N=0~7)
MSGBOXM_MSG_REG	0x0180+N*0x04	Message Register for Message Queue N(N=0~7)
MSGBOX_DEBUG_REG	0x01C0	MSGBOX Debug Register

3.14.6 Register Description

3.14.6.1 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 Message Queue 3 is a transmitter of user u 0: user0 1: user1
27:25	/	/	/
24	R/W	0x0	RECEPTION_MQ3 Message Queue 3 is a receiver of user u 0: user0 1: user1
23:21	/	/	/
20	R/W	0x1	TRANSMIT_MQ2 Message Queue 2 is a transmitter of user u 0: user0 1: user1
19:17	/	/	/
16	R/W	0x0	RECEPTION_MQ2 Message Queue 2 is a receiver of user u 0: user0 1: user1
15:13	/	/	/

12	R/W	0x1	TRANSMIT_MQ1 Message Queue 1 is a transmitter of user u 0: user0 1: user1
11:9	/	/	/
8	R/W	0x0	RECEPTION_MQ1 Message Queue 1 is a receiver of user u 0: user0 1: user1
7:5	/	/	/
4	R/W	0x1	TRANSMIT_MQ0 Message Queue 0 is a transmitter of user u 0: user0 1: user1
3:1	/	/	/
0	R/W	0x0	RECEPTION_MQ0 Message Queue 0 is a receiver of user u 0: user0 1: user1

3.14.6.2 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 Message Queue 7 is a transmitter of user u 0: user0 1: user1
27:25	/	/	/
24	R/W	0x0	RECEPTION_MQ7 Message Queue 7 is a receiver of user u 0: user0 1: user1
23:21	/	/	/
20	R/W	0x1	TRANSMIT_MQ6 Message Queue 6 is a transmitter of user u 0: user0 1: user1
19:17	/	/	/
16	R/W	0x0	RECEPTION_MQ6 Message Queue 6 is a receiver of user u 0: user0 1: user1
15:13	/	/	/

12	R/W	0x1	TRANSMIT_MQ5 Message Queue 5 is a transmitter of user u 0: user0 1: user1
11:9	/	/	/
8	R/W	0x0	RECEPTION_MQ5 Message Queue 5 is a receiver of user u 0: user0 1: user1
7:5	/	/	/
4	R/W	0x1	TRANSMIT_MQ4 Message Queue 4 is a transmitter of user u 0: user0 1: user1
3:1	/	/	/
0	R/W	0x0	RECEPTION_MQ4 Message Queue 4 is a receiver of user u 0: user0 1: user1

3.14.6.3 MSGBox IRQ Enable Register u(u=0,1)(Default Value: 0x0000_0000)

Offset:0x0040+N*0x20(N=0,1)			**Register Name: MSGBOX_IRQ_EN_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R/W	0x0	TRANSMIT_MQ7_IRQ_EN 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 7 is not full.)
14	R/W	0x0	RECEPTION_MQ7_IRQ_EN 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 7 has received a new message.)
13	R/W	0x0	TRANSMIT_MQ6_IRQ_EN 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 6 is not full.)
12	R/W	0x0	RECEPTION_MQ6_IRQ_EN 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 6 has received a new message.)
11	R/W	0x0	TRANSMIT_MQ5_IRQ_EN 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 5 is not full.)

10	R/W	0x0	RECEPTION_MQ5_IRQ_EN 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 5 has received a new message.)
9	R/W	0x0	TRANSMIT_MQ4_IRQ_EN 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 4 is not full.)
8	R/W	0x0	RECEPTION_MQ4_IRQ_EN 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 4 has received a new message.)
7	R/W	0x0	TRANSMIT_MQ3_IRQ_EN 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 3 is not full.)
6	R/W	0x0	RECEPTION_MQ3_IRQ_EN 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 3 has received a new message.)
5	R/W	0x0	TRANSMIT_MQ2_IRQ_EN 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 2 is not full.)
4	R/W	0x0	RECEPTION_MQ2_IRQ_EN 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 2 has received a new message.)
3	R/W	0x0	TRANSMIT_MQ1_IRQ_EN 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 1 is not full.)
2	R/W	0x0	RECEPTION_MQ1_IRQ_EN 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 1 has received a new message.)
1	R/W	0x0	TRANSMIT_MQ0_IRQ_EN 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 0 is not full.)
0	R/W	0x0	RECEPTION_MQ0_IRQ_EN 0: Disable 1: Enable (It will notify user u by interrupt when Message Queue 0 has received a new message.)

3.14.6.4 MSGBox IRQ Status Register u(Default Value: 0x0000_AAAA)

Offset:0x0050+N*0x20(N=0,1)			Register Name: MSGBOXU_IRQ_STATUS_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R/W	0x1	TRANSMIT_MQ7_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for user u when Message Queue 7 is not full. Setting one to this bit will clear it.
14	R/W	0x0	RECEPTION_MQ7_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for user u when Message Queue 7 has received a new message. Setting one to this bit will clear it.
13	R/W	0x1	TRANSMIT_MQ6_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for user u when Message Queue 6 is not full. Setting one to this bit will clear it.
12	R/W	0x0	RECEPTION_MQ6_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for user u when Message Queue 6 has received a new message. Setting one to this bit will clear it.
11	R/W	0x1	TRANSMIT_MQ5_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for user u when Message Queue 5 is not full. Setting one to this bit will clear it.
10	R/W	0x0	RECEPTION_MQ5_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for user u when Message Queue 5 has received a new message. Setting one to this bit will clear it.
9	R/W	0x1	TRANSMIT_MQ4_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for user u when Message Queue 4 is not full. Set one to this bit will clear it.
8	R/W	0x0	RECEPTION_MQ4_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for user u when Message Queue 4 has received a new message. Setting one to this bit will clear it.
7	R/W	0x1	TRANSMIT_MQ3_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for user u when Message Queue 3 is not full. Setting one to this bit will clear it.
6	R/W	0x0	RECEPTION_MQ3_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for user u when Message Queue 3 has received a new message. Setting one to this bit will clear it.
5	R/W	0x1	TRANSMIT_MQ2_IRQ_PEND

			0: No effect 1: Pending. This bit will be pending for user u when Message Queue 2 is not full. Setting one to this bit will clear it.
4	R/W	0x0	RECEPTION_MQ2_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for user u when Message Queue 2 has received a new message. Setting one to this bit will clear it.
3	R/W	0x1	TRANSMIT_MQ1_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for user u when Message Queue 1 is not full. Setting one to this bit will clear it.
2	R/W	0x0	RECEPTION_MQ1_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for user u when Message Queue 1 has received a new message. Setting one to this bit will clear it.
1	R/W	0x1	TRANSMIT_MQ0_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for user u when Message Queue 0 is not full. Setting one to this bit will clear it.
0	R/W	0x0	RECEPTION_MQ0_IRQ_PEND 0: No effect 1: Pending. This bit will be pending for user u when Message Queue 0 has received a new message. Setting one to this bit will clear it.

3.14.6.5 MSGBox FIFO Status Register m(Default Value: 0x0000_0000)

Offset:0x0100+N*0x04 (N=0~7)			**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 (space is available) 1: The Message FIFO queue is full. This FIFO status register has the status related to the message queue.

3.14.6.6 MSGBox Message Status Register m(Default Value: 0x0000_0000)

Offset:0x0140+N*0x04 (N=0~7)			**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. Here, limited to four messages per message queue. 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. 101~111:/

3.14.6.7 MSGBox Message Queue Register m(Default Value : 0x0000_0000)

Offset:0x0180+N*0x04 (N=0~7)			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 of the message FIFO queue. Reads remove the message from the FIFO queue.

3.14.6.8 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 disable and only one register space 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.

3.15 Spinlock

3.15.1 Overview

In multi-core system,the Spinlock offers hardware synchronization mechanism, lock operation can prevent multi processors from handling data-sharing at the same time, and ensure coherence of data.

The Spinlock has the following features:

- Spinlock module includes 32 lock units
- Two kinds of lock status: locked and unlocked
- Lock time of the processor is predictable(less than 200 cycles)

3.15.2 Block Diagram

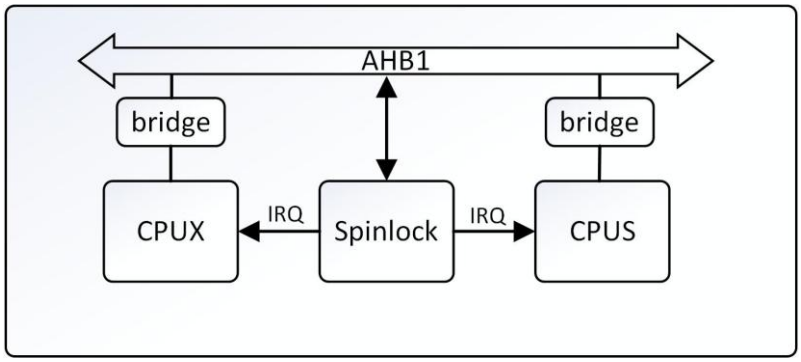


Figure3- 40. Spinlock Block Diagram

3.15.3 Operations and Functional Descriptions

3.15.3.1 Clock and Reset

The Spinlock is hung on AHB1. Before accessing Spinlock register, open the corresponding gating bit on AHB1 and de-assert reset signal. The correct operation order is to de-assert reset signal at first, and then open the corresponding gating signal.

3.15.3.2 Typical Application

A processor lock spinlock0, when the status is locked, the processor executes specific code, and then unlocks code. Other processors is released to start reading/writing operation.

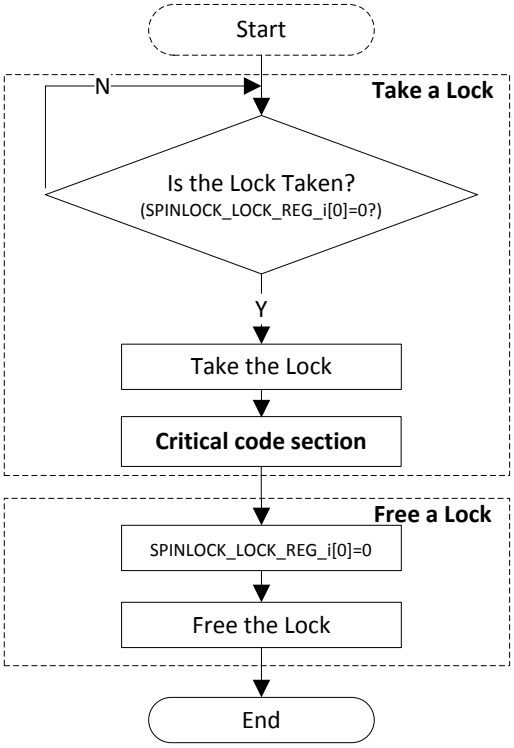


Figure3- 41. Spinlock Typical Application Diagram

3.15.3.3 Function Implemention

3.15.3.3.1 Spinlock State Machine

When a processor uses spinlock, it needs to acquire spinlock's status through **SPINLOCK_STATUS_REG**.
Reading Operation: when return value is 0, spinlock comes into locked status; when read this status bit again, return value is 1, spinlock comes into locked status.
Writing Operating: when the Spinlock is in locked status, the Spinlock can convert to unlocked status through writing 0. After reset, the Spinlock is in unlocked status by default.

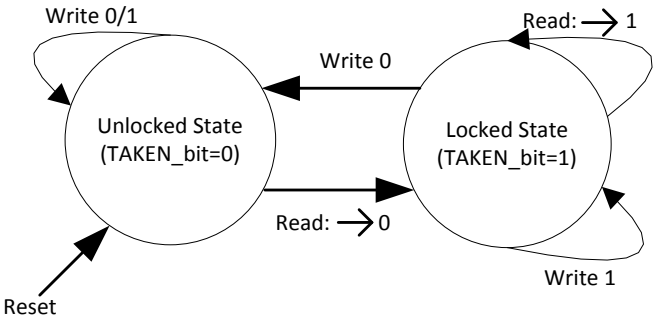


Figure3- 42. Spinlock State Machine

3.15.3.3.2 Interrupt

When **Free Lock** is released(lock status is changed from locked to unlocked), interrupt is generated.

3.15.3.4 Operating Mode

3.15.3.4.1 Switch Status

When the read value from **SPINLOCKN_LOCK_REG** is 0,the Spinlock come into locked status.

Execute application code, the status of **SPINLOCKN_STATUS_REG** is 1.

Write 0 to **SPINLOCKN_LOCK_REG**, the Spinlock comes into unlocked status, the corresponding spinlock is released.

3.15.3.4.2 Interrupt Application

Enable the conresponding spinlock interrupt enable bit in **SPINLOCK_IRQ_EN_REG.**

When the spinlock status changes from Locked to Unlocked, it will generate interrupr, and the conrespongding bit of **SPINLOCK_IRQ_STA_REG** is set to 1.

Execute the interrupt handing function, and clear the interrupt pending bit.

3.15.4 Programming Guidelines

Take CPU0's synchronization with CPUS through Spinlock0 for an example, CPU0 takes the spinlock0 firstly in the instance.

To taking/freeing spinlock0, CPU0 and CPUS perform the following steps:

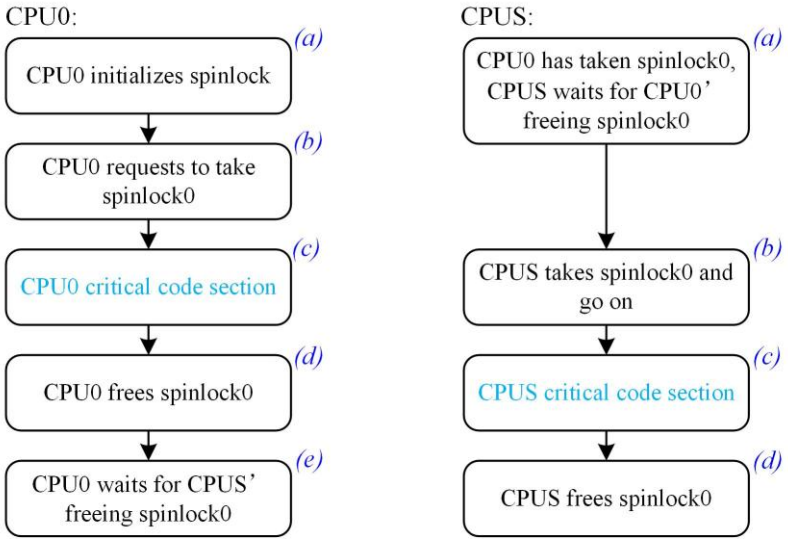


Figure3- 43. CPU0 and CPUS Taking/Freeing Spinlock0 Process

CPU0:

The CPU0 initializes Spinlock.

Firstly, check lock register0 (SPINLOCK_STATUS_REG0) status, if it is taken, check till CPU0 frees spinlock0. Then request to take spinlock0, if fail, retry till lock register0 is taken.

Execute CPU0 critical code.

After executing CPU0 critical code, the CPU0 frees spinlock0.

The CPU0 waits for CPUS' freeing spinlock0.

CPUS:

If the CPU0 has taken spinlock0, the CPUS waits for CPU0' freeing spinlock0.
The CPUS requests to take spinlock0, if fail, retry till lock register0 is taken.
Execute CPU0 critical code.
After executing CPU0 critical code, the CPUS frees spinlock0.

The following codes are for reference.

----------CPU0 of Cluster0----------

```
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 it is taken, check till
    if(rdata != 0)    writel(0, SPINLOCK_LOCK_REG0);    //CPU0 frees spinlock0
    rdata=readl(SPINLOCK_LOCK_REG0);                //request to take spinlock0, if fail, retry till
    if(rdata != 0)    rdata=readl(SPINLOCK_LOCK_REG0);    // 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 fail, retry till
    if(rdata != 0)    rdata=readl(SPINLOCK_LOCK_REG0);    // lock register0 is taken
---------- CPUS critical code section ----------
Step 3: CPUS frees spinlock0
writel (0, SPINLOCK_LOCK_REG0);                     //CPUS frees spinlock0
```

3.15.5 Register List

Module Name	Base Address
Spinlock	0x03004000

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_LOCK_REGN	0x0100+N*0x04	Spinlock Register N (N=0~31)

3.15.6 Register Description

3.15.6.1 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 256 lock registers. 01: This instance has 32 lock registers. 10: This instance has 64 lock registers. 11: This instance has 128 lock registers.
27:9	/	/	/
8	R	0x0	IU0 In-Use flag0, covering lock register0-31 0: All lock register 0-31 are in the NotTaken state. 1: At least one of the lock register 0-31 is in the Taken state.
7:0	/	/	/

3.15.6.2 Spinlock Register Status(Default Value: 0x0000_0000)

Offset: 0x0010			Register Name: SPINLOCK_STATUS_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
[i] (i=0~31)	R	0x0	LOCK_REG_STATUS SpinLock[i] status (i=0~31) 0: The Spinlock is free. 1: The Spinlock is taken.

3.15.6.3 Spinlock Interrupt Enable Register(Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: SPINLOCK_IRQ_EN_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
[i] (i=0~31)	R/W	0x0	LOCK_IRQ_EN SpinLock[i] interrupt enable 0:Disable 1:Enable

3.15.6.4 Spinlock Interrupt Status Register(Default Value: 0x0000_0000)

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

[i] (i=0~31)	R/W1C	0x0	LOCK_IRQ _STATUS. SpinLock[i] interrupt status. 0:No effect 1:Pending Writing 1 will clear this bit.

3.15.6.5 Spinlock Register N (N=0 to 31)(Default Value: 0x0000_0000)

Offset:0x0100+N*0x04 (N=0~31)			Register Name: SPINLOCKN_LOCK_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
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.

Figures

4 Video and Graphics

4.1 DE2.0

The Display Engine(DE) is a hardware composer to transfer image layers from a local bus or a video buffer to the LCD interface. The DE supports four overlay windows to blend, and supports image post-processing in the video channel. The block diagram of DE is shown in Figure 4-1.

The DE has the following features:

- Output size up to 2048 x 2048
- Four alpha blending channels for main display
- Four overlay layers in each channel, and has a independent scaler
- Potter-duff compatible blending operation
- Input format semi-planar of YUV422/YUV420/YUV411 and planar of YUV422/YUV420/YUV411, ARGB8888/ XRGB8888/RGB888/ARGB4444/ARGB1555/RGB565
- Frame Packing/Top-and-Bottom/Side-by-Side Full/Side-by-Side Half 3D format data
- Supports SmartColor2.0 for excellent display experience
  - Adaptive detail/edge enhancement
  - Adaptive color enhancement
  - Adaptive contrast enhancement and fresh tone rectify
  - Content adaptive backlight control
- Supports writeback for miracast.

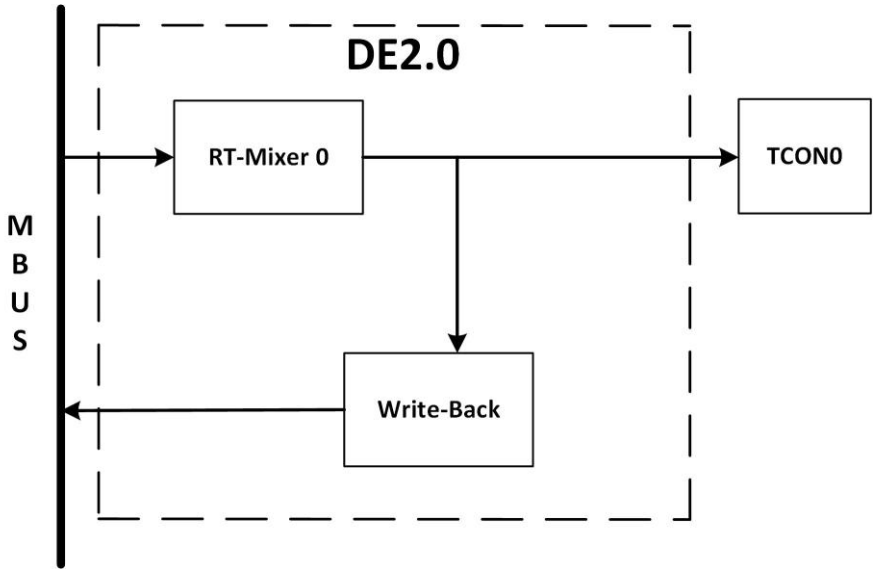


Figure4- 1. Display Syatem Block Diagram

4.2 G2D

The Graphic 2D(G2D) Engine is hardware accelerator for 2D graphic.

The G2D has the following features:

- Layer size up to 2048 x 2048 pixels
- Input/output format: YUV422(semi-planar and planar format) /YUV420(semi-planar and planar format)/P010/P210/P410/Y8/ARGB8888/XRGB8888/RGB888/ARGB4444/ ARGB1555/ARGB2101010 and RGB565
- Horizontal and vertical flip, clockwise 0/90/180/270 degree rotate

4.3 Video Encoding

The Video Encoding consists of the video encoding unit(VE) and JPEG encoder(JPGE). The VE supports H.264 encoding, and JPGE supports JPEG/MJPEG encoding.

4.3.1 VE

4.3.1.1 Overview

The VE is a CODEC that supports H.264 protocol based on ASIC. It is custom-made for the IPC usage and features high compressing rate, low CPU usage, short delay and low power consumption.
The VE has the following features:

- Supports ITU-T H.264 high profile/main profile/baseline profile@level 5.2 encoding
  - Encoding of multiple slice
  - Motion compensation with 1/2 and 1/4 pixel precision
  - Four prediction unit (PU) types of 16x8, 8x16, 16x16 and 8x8 for inter-prediction
  - Two prediction unit types of Intra16x16 and Intra4x4 for intra-prediction
  - Trans4x4
  - CABAC and CAVLC entropy encoding
  - De-blocking filtering
- Supports Classify, MB-RateControl, Fore-3D-Filter, Cyclic-Intra-Refresh, Dynamic-ME and Intra-4x4-Disable Functions in general
- Supports the input picture format of semi-planar YCbCr4:2:0
- Supports H.264 encoding with the performance of 8-megapixel 1080P@60fps
- Supports configurable picture resolutions
  - Minimum picture resolution: 192x96
  - Maximum picture resolution: 4096x4096
  - Step of the picture width or height: 8
- Supports region of interest (ROI) encoding
  - Maximum of 4 ROIs
  - Independent enable/disable control for the encoding function of each ROI
- Supports on-screen display (OSD) encoding protection that can be enabled or disabled
- Supports OSD front-end overlaying
  - OSD overlaying before encoding for a maximum of 16 regions
  - OSD overlaying with any size and at any position (within the size and position range of the picture)
  - 16-level alpha blending
  - OSD overlaying control (enabled or disabled)
- Supports three bit rate control modes: constant bit rate (CBR), variable bit rate (VBR) and FIXQP
- Supports the output bit rate ranging from 2 Kbit/s to 60 Mbit/s
- Supports Frame Buffer Compression

4.3.1.2 Block Diagram

The functional block diagram of the VE is as follows.

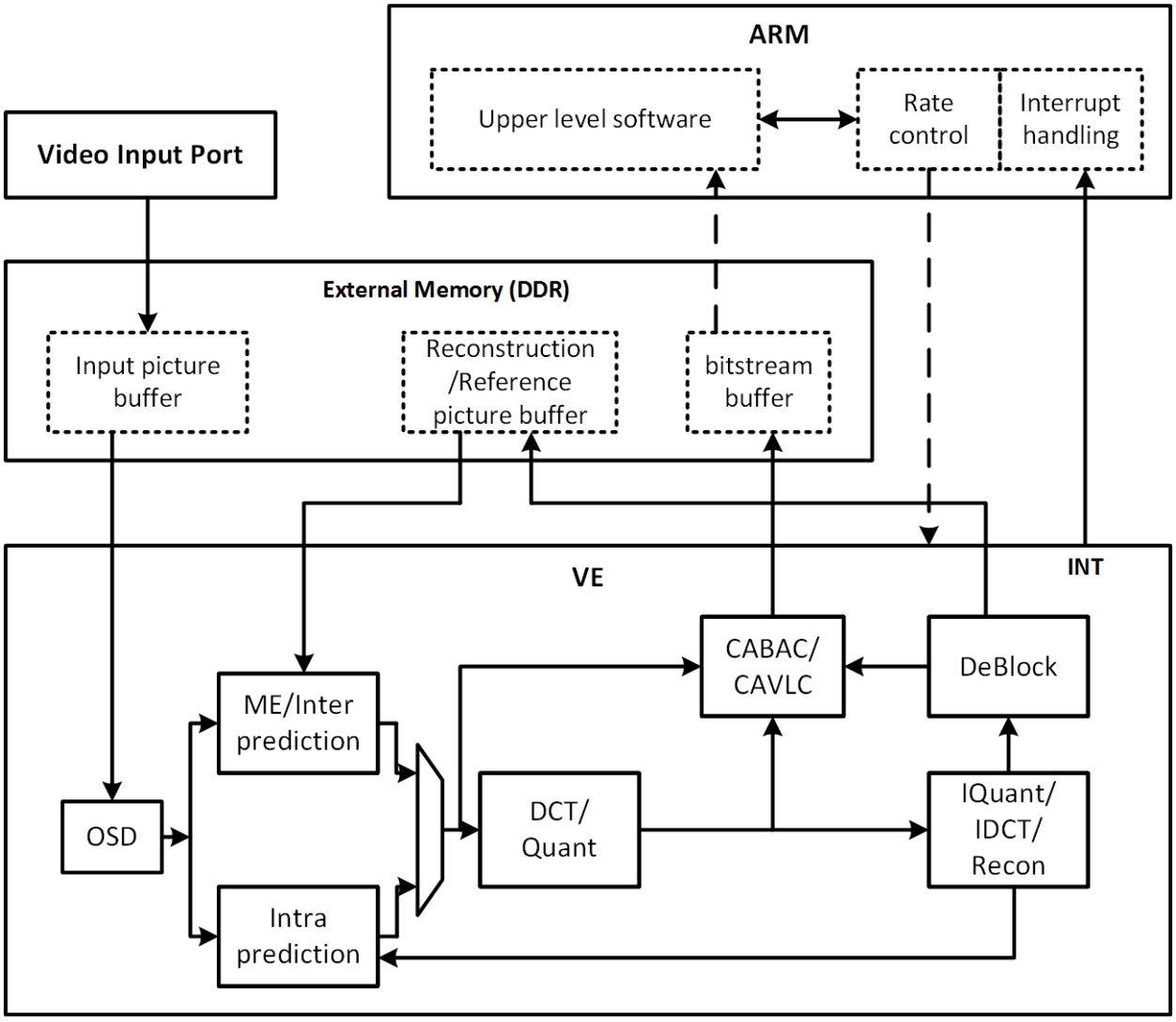


Figure4- 2. VE Block Diagram

Based on related protocols and algorithms, the VE supports motion estimation/inter-prediction, intra-prediction, transform/quantization, inverse transform/inverse quantization, CABAC/CAVLC encoding/stream generation and DeBlock. The ARM software controls the bitrate and handles interrupt.

Before the VE is enabled for video encoding, software allocates three types of buffers mainly in the external DDR SDRAM:

- **Input picture buffer**

  The VE reads the source pictures to be encoded form this buffer during encoding. This buffer is typically written by the Video Input Port module.

- **Reconstruction/Reference picture buffer**

  The VE writes reconstruction pictures to this buffer during encoding. These reconstruction pictures are used as the reference pictures of subsequent pictures. During the encoding of P frames, reference pictures are read from this buffer.

- **Stream buffer**

  This buffer stores encoded streams. The VE writes streams to this buffer during encoding. This buffer is read by software.

4.3.2 JPGE

4.3.2.1 Overview

The JPGE is a high-performance encoder based on ASIC. It supports 64-megapixel snapshot or HD MJPEG encoding.
The JPGE has the following features:

- Supports ISO/IEC 10918-1 (CCITT T.81) baseline process (DCT sequential) encoding
- Encodes the pictures in the chrominance sampling format of YCbCr4:2:0 and YCbCr4:2:2
- Supports multiple input picture formats:
  - Semi-planar YCbCr4:2:0
  - Semi-planar YCbCr4:2:2
- Supports JPEG encoding with the performance of 1080P@60fps
- Supports configurable picture resolutions
  - Minimum picture resolution: 192x96
  - Maximum picture resolution: 4096x4096
- Supports the picture width or height step of 8
- Supports configurable quantization tables for the Y component, Cb component and Cr component respectively
- Supports OSD front-end overlapping
  - OSD overlaying before encoding for a maximum of 16 regions
  - OSD overlaying with any size and at any position (within the size and position range of the picture)
  - 16-level alpha blending
  - OSD overlaying control (enabled or disabled)
- Supports the color-to-gray function
- Supports the MJPEG output bit rate ranging from 2 Kbit/s to 60 Mbit/s

4.3.2.2 Block Diagram

The functional block diagram of the JPGE is as follows.

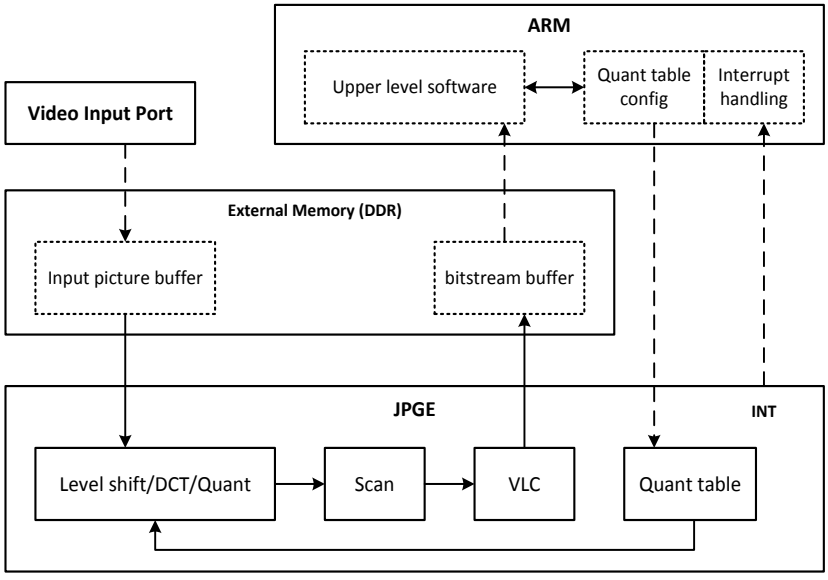


Figure4- 3. JPGE Block Diagram

Based on the protocols that require a large number of operands, the JPGE supports OSD, level shift, DCT, quantization, scanning, VLC encoding and stream generation. The ARM software configures quantization tables and handles interrupt.

Before the JPGE is enabled for video encoding, the software allocates two types of buffers mainly in the external DDR SDRAM:

- **Input picture buffer**
  The JPGE reads the source pictures to be encoded from this buffer during encoding. This buffer is generally written by the Video Input Port module.
- **Stream buffer**
  This buffer stores encoded streams. The JPGE writes streams to this buffer during encoding. This buffer is read by software.

4.4 Video Decoding

4.4.1 Overview

The Video Decoding consists of Video Control Firmware(VCF) running on ARM processor and embedded hardware Video Engine(VE). VCF gets the bitstream from topper software, parses bitstream, invokes the Video Engine, and generates the decoding image sequence. The decoding image sequence is transmitted by the video output controller to the display device under the control of the topper software.

The Video decoding has the following features:

- Supports ITU-T H.265 Main Profile@Level 4.1(or below)
  - Maximum video resolution: 4096 x 2048
  - Maximum decoding rate: 60Mbps, 1080p@60fps
- Supports ITU-T H.264 Base/Main/High Profile
  - Maximum video resolution: 4096 x 2048
  - Maximum decoding rate: 60Mbps, 1080p@60fps
- Supports ITU-T H.263 Base Profile
  - Maximum video resolution: 1080p
  - Maximum decoding rate: 100Mbps, 1080p@60fps
- Supports VP9
  - Maximum video resolution: 720p
  - Maximum decoding rate: 4Mbps, 720p@30fps
- Supports VP8
  - Maximum video resolution: 4096 X 2048
  - Maximum decoding rate: 40Mbps, 1080p@60fps
- Supports VP6 6.0/6.1/6.2
  - Maximum video resolution: 720p
  - Maximum decoding rate: 10Mbps, 720p@30fps
- Supports VC-1 SP/MP/AP
  - Maximum video resolution: 1080p
  - Maximum decoding rate: 100Mbps, 1080p@30fps
- Supports WMV7/8
  - Maximum video resolution: 3840 x 2160
  - Maximum decoding rate: 10Mbps, 1080p@60fps
- Supports AVS+
  - Maximum video resolution: 1080p

- Maximum decoding rate: 30Mbps, 1080p@60fps

- Supports AVS
  - Maximum video resolution: 1080p
  - Maximum decoding rate: 30Mbps, 1080p@60fps
- Supports DIVX3.11
  - Maximum video resolution: 720p
  - Maximum decoding rate: 10Mbps, 720p@30fps
- Supports DIVX4/5 HD
  - Maximum video resolution: 1080p
  - Maximum decoding rate: 10Mbps, 1080p@60fps
- Supports MJPEG
  - Maximum video resolution: 1080p
  - Maximum decoding rate: 300Mbps, 1080p@60fps
- Supports MPEG4-XVID SP/ASP
  - Maximum video resolution: 3840 x 2160
  - Maximum decoding rate: 100Mbps, 1080p@60fps
- Supports MPEG2
  - Maximum video resolution: 1080p
  - Maximum decoding rate: 100Mbps, 1080p@60fps
- Supports MPEG1
  - Maximum video resolution: 1080p
  - Maximum decoding rate: 100Mbps, 1080p@60fps

4.4.2 Block Diagram

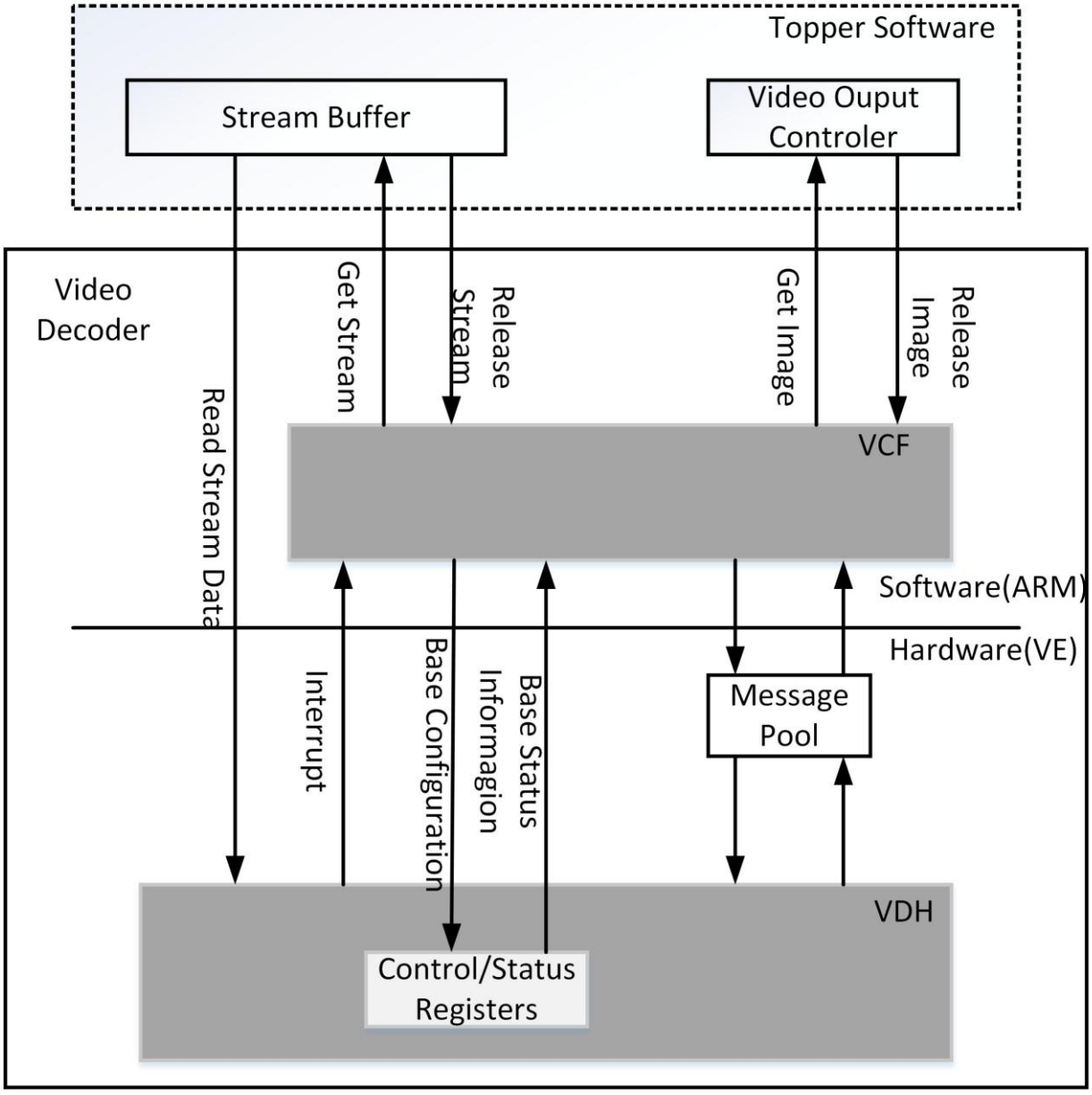


Figure4- 4. Video Decoding Block Diagram

Figures

Tables

5 Memory

5.1 DRAM Controller(DRAMC)

5.1.1 Overview

The SDRAM Controller (DRAMC) provides a simple, flexible, burst-optimized interface to the industy-standard DDR4/DDR3/DDR3L SDRAM and Low Power DDR3/4 SDRAM. It supports up to a 24G bits memory address space.
The DRAMC automatically handles memory management, initialization, and refresh operations. It gives the host CPU a simple command interface, hiding details of the required address, page, and burst handling procedures. All memory parameters are runtime-configurable, including timing, memory setting, SDRAM type, and Extended-Mode-Register setting. To simplify chip system integration, DDR controller works in half rate mode.

Features:

- 32-bit bus width
- Supports 2 chip selects
- Supports DDR4/DDR3/DDR3L/LPDDR3/LPDDR4 SDRAM
- Supports different memory device's power voltage of 1.2V ,1.5V,1.35V,1.2V, 1.1V
- Supports clock frequency up to 792MHz(DDR4)
- Supports clock frequency up to 792MHz for DDR3/DDR3L
- Supports clock frequency up to 672MHz for LPDDR3
- Supports clock frequency up to 672MHz for LPDDR4
- Supports memory capacity up to 24G bits (3G bytes)
- Supports 18 address lines and 3 bank address lines
- Automatically generates initialization and refresh sequences
- Runtime-configurable parameters setting for application flexibility
- Priority of transferring through multiple ports is programmable
- Random read or write operation is supported

5.2 NAND Flash Controller(NDFC)

5.2.1 Overview

The NDFC is the NAND Flash Controller which supports all NAND flash memory available in the market. New type flash can be supported by software re-configuration.

The parallel error correction code (ECC) is built-in NDFC for enhancing reliability. The NDFC provides BCH and LDPC error correction engine. BCH can detect and correct 80-bit error per 1024 bytes data. LDPC has the stronger error correction capability, so that it can adapt to the 3D/TLC Nand Flash in the future. The integrated ECC and parity circuit releases CPU for other tasks. This ECC function can be disabled by software.

NDFC supports command description function, Changing form the original CPU handing register to the MBUS DMA handing register. So that the performance of NDFC is improved.

While retaining the original randomization scheme, it adds new randomization scheme. The random seed is placed in one time, and it allocates 16 locations. Each data block corresponds to a randomized seed.

NDFC supports retransmission command function to check the NAND Flash status.

NDFC provides two selectable channel. Channel 1 indicates that NDFC chose BCH error correction algorithm, 1KB Data Block Size, and old randomization scheme. Channel 2 indicates that NDFC chose LDPC error correction algorithm, 2KB Data Block Size, and new randomization scheme.

The data can be transferred by DMA or by CPU memory-mapped IO method. The NDFC provides automatic timing control for reading or writing external Flash. The NDFC maintains the proper relativity for CLE, CE# and ALE control signal lines. There are three different kinds of modes for serial read access, mode0 is for conventional serial access , mode1 is for EDO type and the mode2 is for extension EDO type. NDFC can monitor the status of R/B# signal line.
Block management and wear leveling management are implemented in software.

Features:

- Supports all SLC/MLC/TLC flash and EF-NAND memory available in the market
- Software configure seed for randomize engine
- Software configure method for adaptability to a variety of system and memory types
- Supports 2CE/2RB
- Supports 3.3V/1.8V IO Voltage
- Supports 8-bit data bus width
- Supports 1024, 2048, 4096, 8192, 16384, 32768 bytes size per page
- Supports full disk encryption(FDE) and decryption function
- Parallel BCH error correction code which correcting up to 80 bits/1024 bytes
- Parallel LDPC error correction engine
- Corrected Error bits number information report
- ECC automatic disable function for all 0xff data
- NDFC status information is reported by its registers and interrupt is supported
- One Command FIFO
- Supports internal DMA controller based on chain-structured descriptor list
- Two 256x32-bit RAM for Pipeline Procession
- Supports SDR, ONFI DDR1.0 , Toggle DDR1.0, ONFI DDR2.0, Toggle DDR2.0 and Raw NAND Flash.
- Supports the Maximum IO Rate 50MHz in SDR mode, and 100MHz in both DDR1.0 and DDR2.0 mode
- Supports self-debug for NDFC debug

5.2.2 Block Diagram

The NDFC system block diagram is shown as follows.

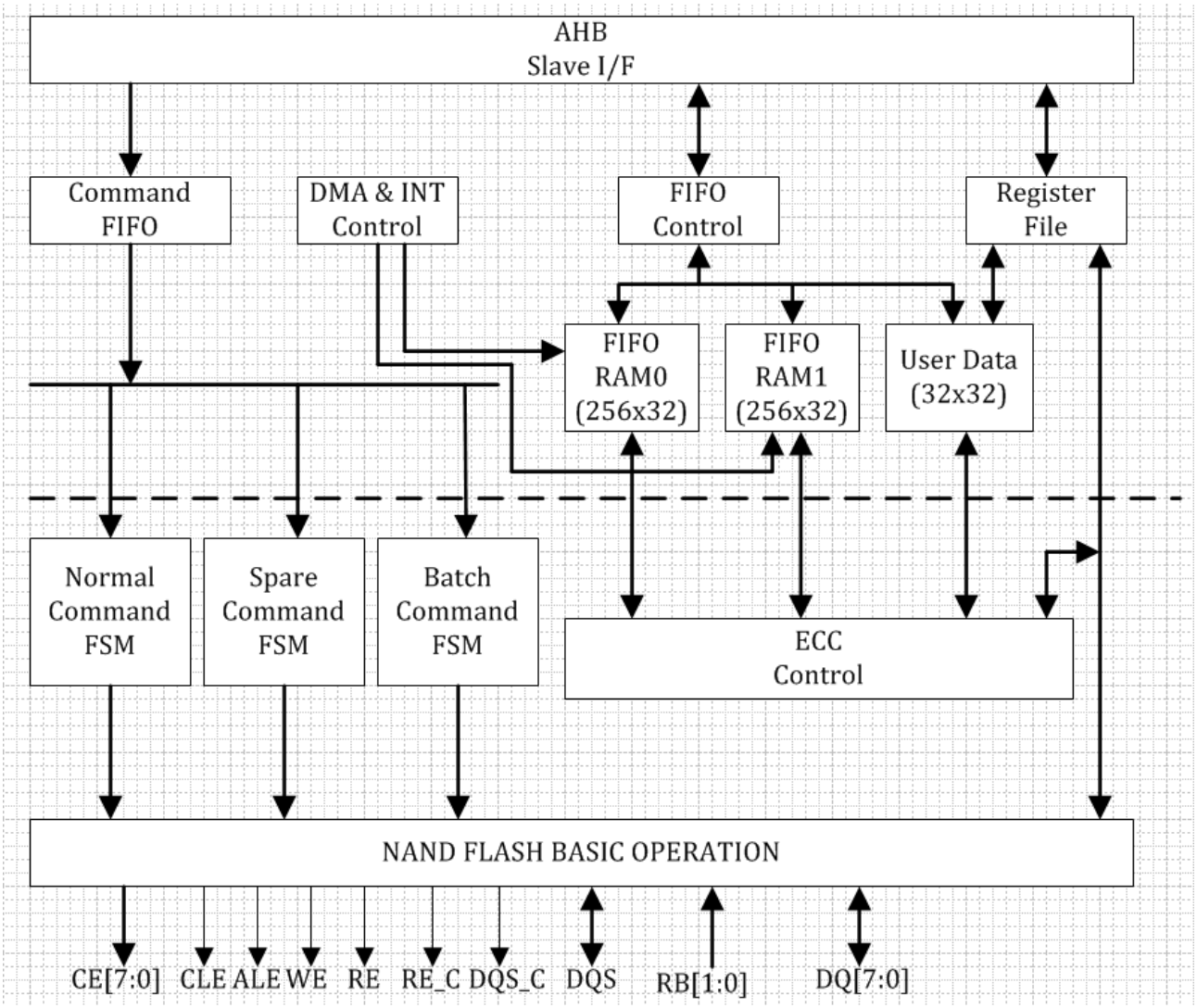


Figure5- 1. NDFC Block Diagram

5.2.3 Operations and Functional Descriptions

5.2.3.1 External Signals

Table 5-1 describes the external signals of NDFC. DQ0~DQ7 and DQS are bidirectional I/O. WE,ALE,CLE,CE,RE are output pin, RB is input pin. The RB pin in the NAND device is an open-drain driver, which must need a pull-up resistor.

Table5- 1. 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_CE1	Chip Enable, Low is Active	O
NAND_RB0	Ready/Busy, Low is Active	I
NAND_RB1	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
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

5.2.3.2 Clock Sources

To ensure ECC efficiency, ECC engine and NDFC internal logic use different clock. The clock of NDFC internal logic is set by NAND0 Clock Register, the clock of ECC engine is set by NAND1 Clock Register .Note that NAND0 Clock Register set the internal logic clock of NDFC, but the frequency of external Nand Flash device is half of NDFC internal logic clock. That is, if external Nand Flash runs at 40MHz, then NDFC need set to 80MHz.

Both ECC engine and NDFC internal logic have five different clock sources. Users can select one of them to make ECC engine or internal logic clock source. Table 5-2 describles the clock sources of NDFC. Users can see CCU in chapter 3.3 for clock setting, configuration and gating information.

Table5- 2. NDFC Clock Sources

Clock Sources	Description
OSC24M	24MHz Crystal
PLL_PERIPH0(1X)	Peripheral Clock, default value is 600MHz
PLL_PERIPH1(1X)	Peripheral Clock, default value is 600MHz
PLL_PERIPH0(2X)	Peripheral Clock, default value is 1200MHz
PLL_PERIPH1(2X)	Peripheral Clock, default value is 1200MHz

5.2.3.3 NDFC Timing Diagram

Typically, there are two kinds of serial access methods. One method is conventional method which fetching data at the rise edge of NDFC_RE# signal line. Another one is EDO type which fetching data at the next fall edge of NDFC_RE# signal line.

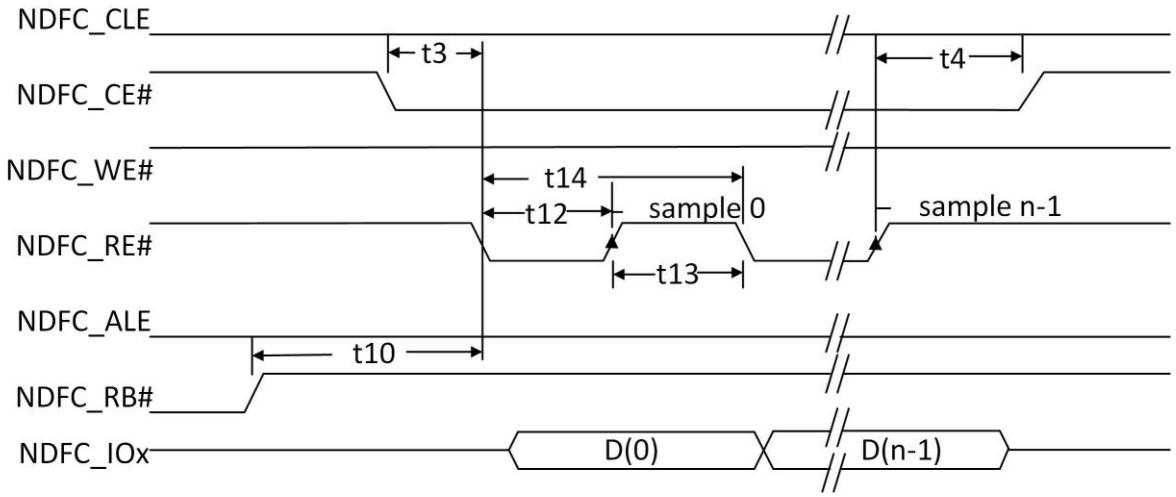


Figure5- 2. Conventional Serial Access Cycle Diagram (SAM0)

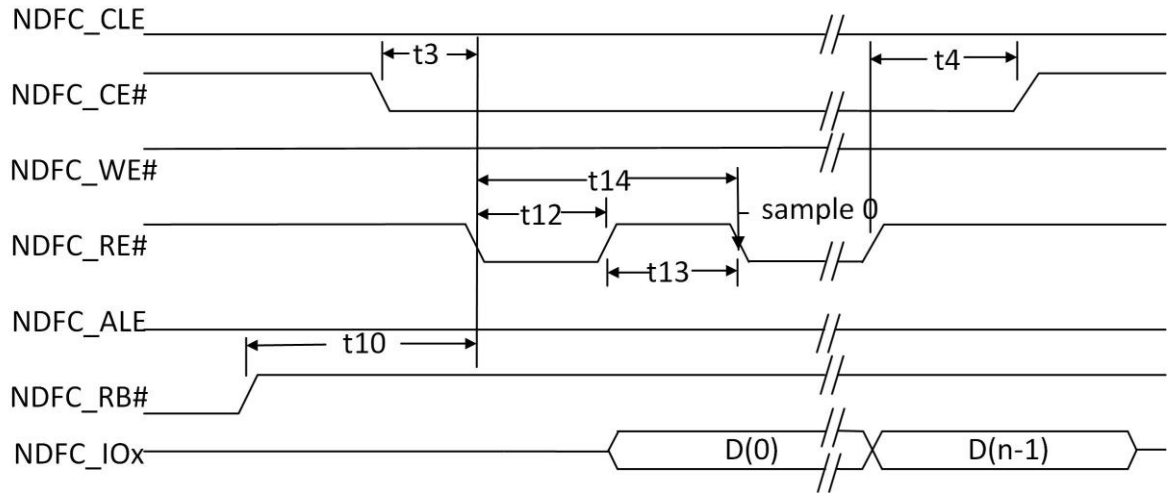


Figure5- 3. EDO Type Serial Access after Read Cycle (SAM1)

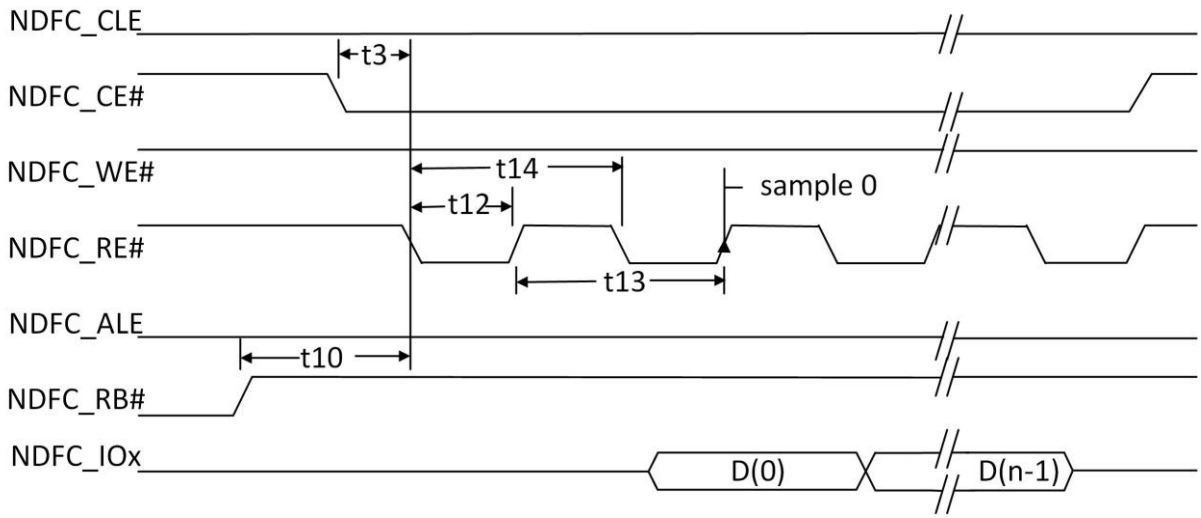


Figure5- 4. Extending EDO Type Serial Access Mode (SAM2)

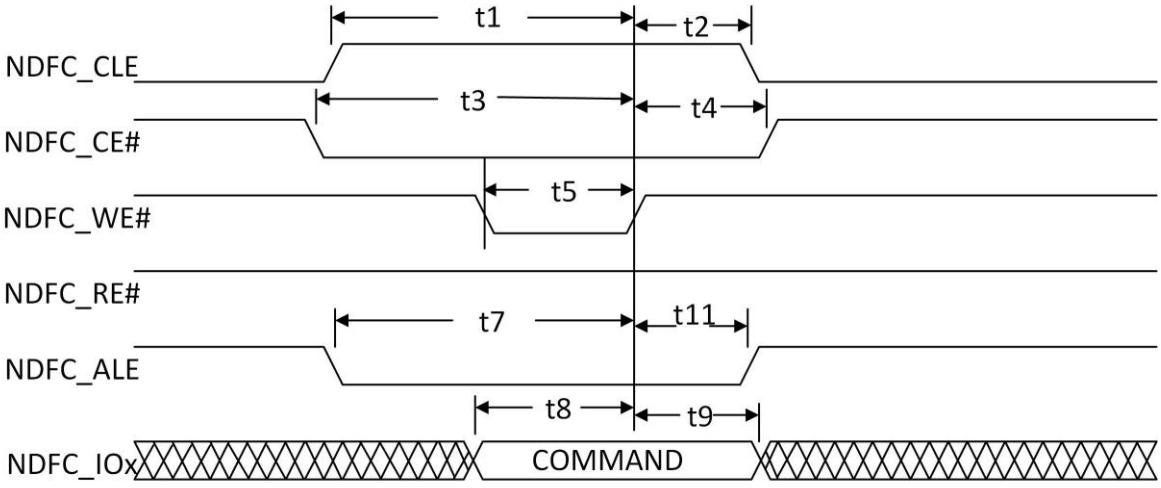


Figure5- 5. Command Latch Cycle

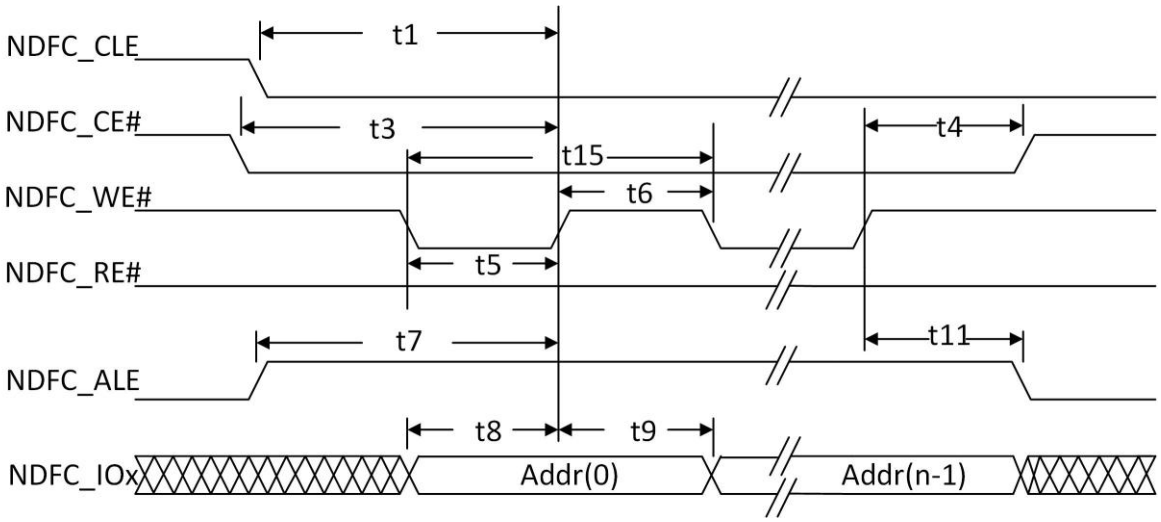


Figure5- 6. Address Latch Cycle

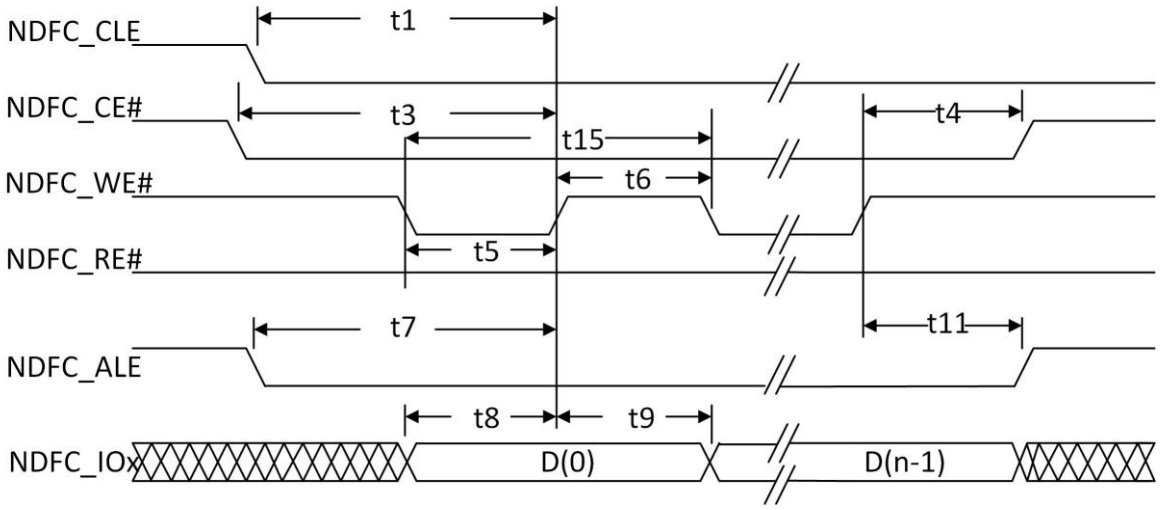


Figure5- 7. Write Data to Flash Cycle

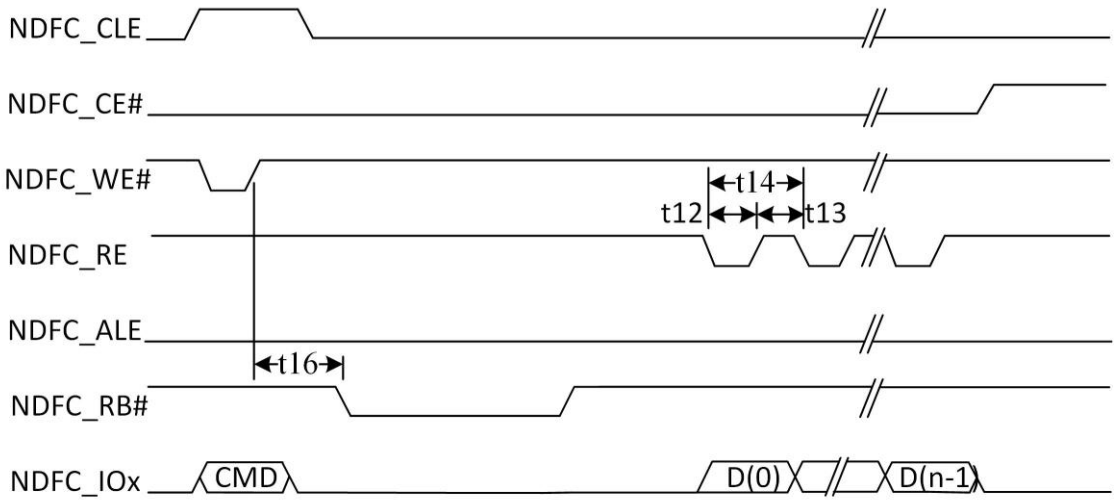


Figure5- 8. Waiting R/B# Ready Diagram

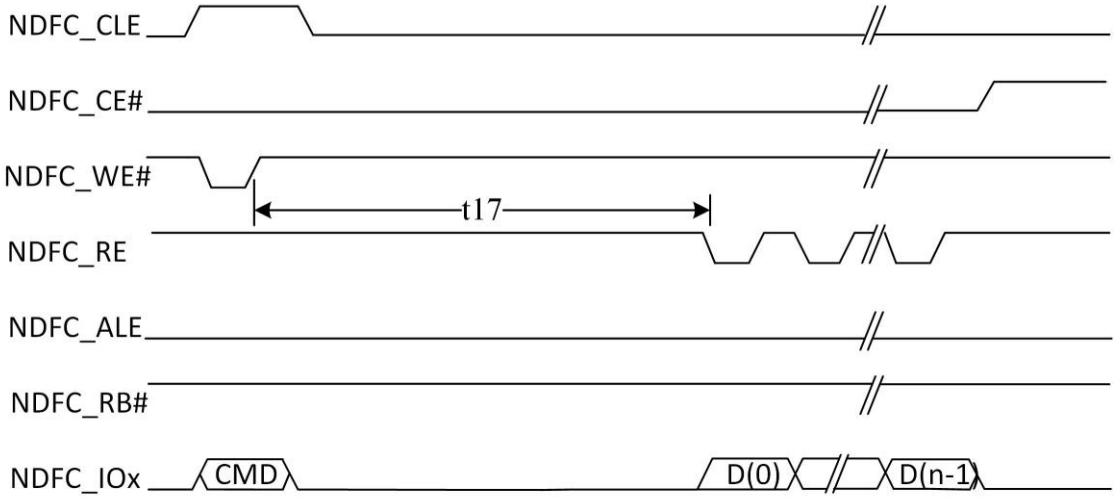


Figure5- 9. WE# High to RE# Low Timing Diagram

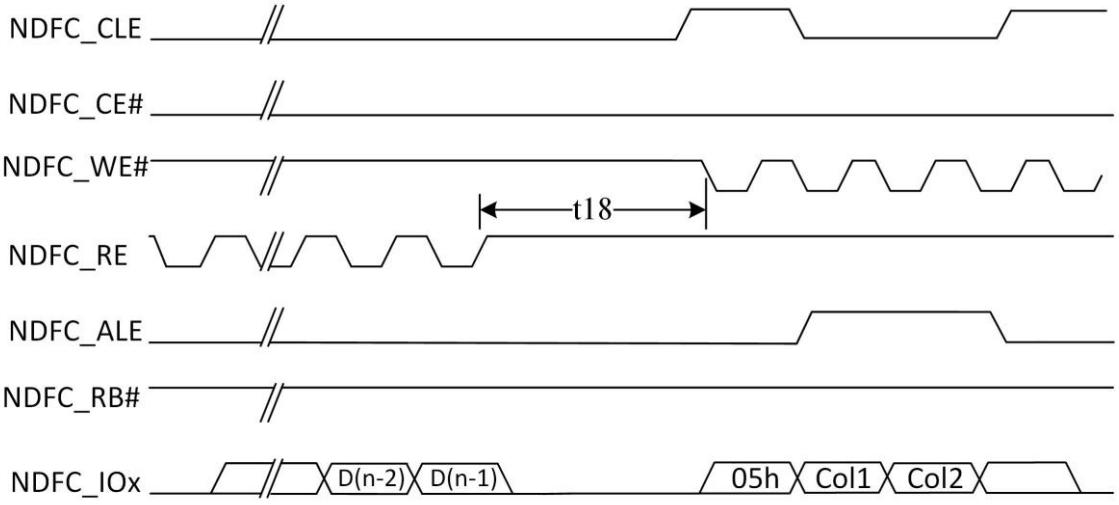


Figure5- 10. RE# High to WE# Low Timing Diagram

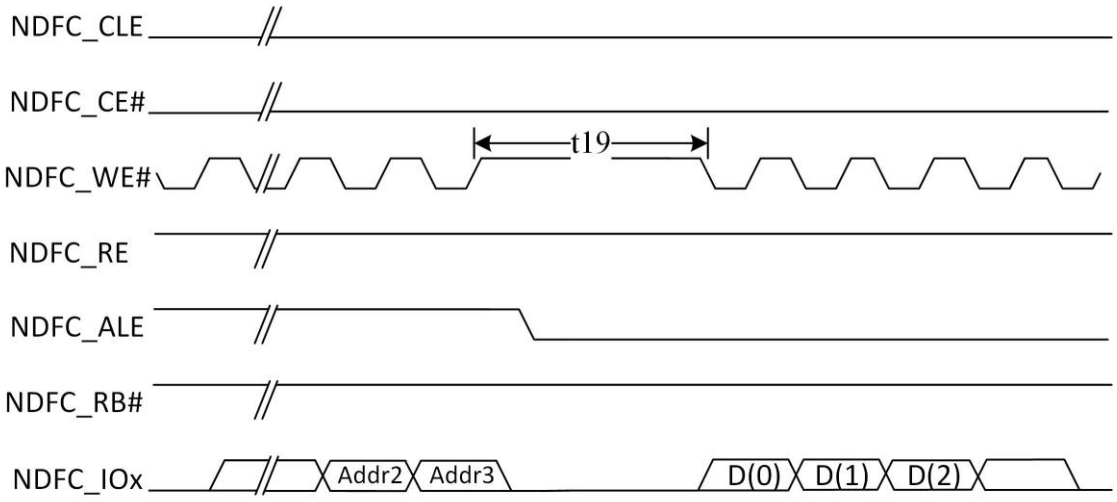


Figure5- 11. Address to Data Loading Timing Diagram

Timing cycle list:

ID	Parameter	Timing	Unit
t1	NDFC_CLE setup time	2T	ns
t2	NDFC_CLE hold time	2T	ns
t3	NDFC_CE setup time	2T	ns
t4	NDFC_CE hold time	2T	ns
t5	NDFC_WE# pulse width	T(1)	ns
t6	NDFC_WE# hold time	T	ns
t7	NDFC_ALE setup time	2T	ns
t8	Data setup time	T	ns
t9	Data hold time	T	ns
t10	Ready to NDFC_RE# low	3T	ns
t11	NDFC_ALE hold time	2T	ns
t12	NDFC_RE# pulse width	T	ns
t13	NDFC_RE# hold time	T	ns
t14	Read cycle time	2T	ns
t15	Write cycle time	2T	ns
t16	NDFC_WE# high to R/B# busy	T_WB(2)	ns
t17	NDFC_WE# high to NDFC_RE# low	T_WHR(3)	ns
t18	NDFC_RE# high to NDFC_WE# low	T_RHW(4)	ns
t19	Address to Data Loading time	T_ADL(5)	ns

Note(1): T is the cycle of the internal clock.

Note(2),(3),(4),(5): These values are configurable in 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, the value of T_ADL could be 0*2T/6*2T/14*2T/22*2T.

5.2.3.4 NDFC Operation Guide

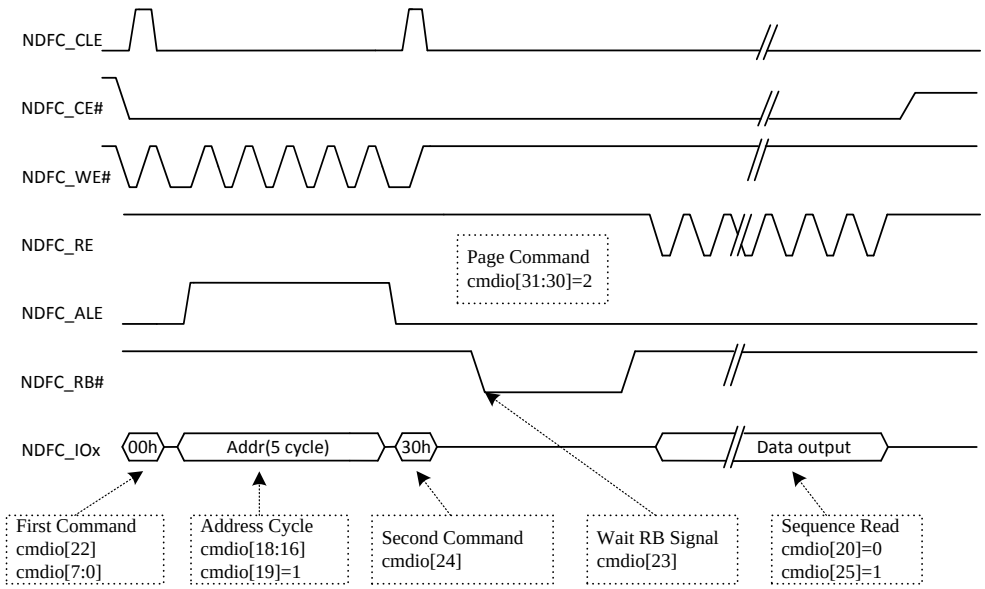


Figure5- 12. Page Read Command Diagram

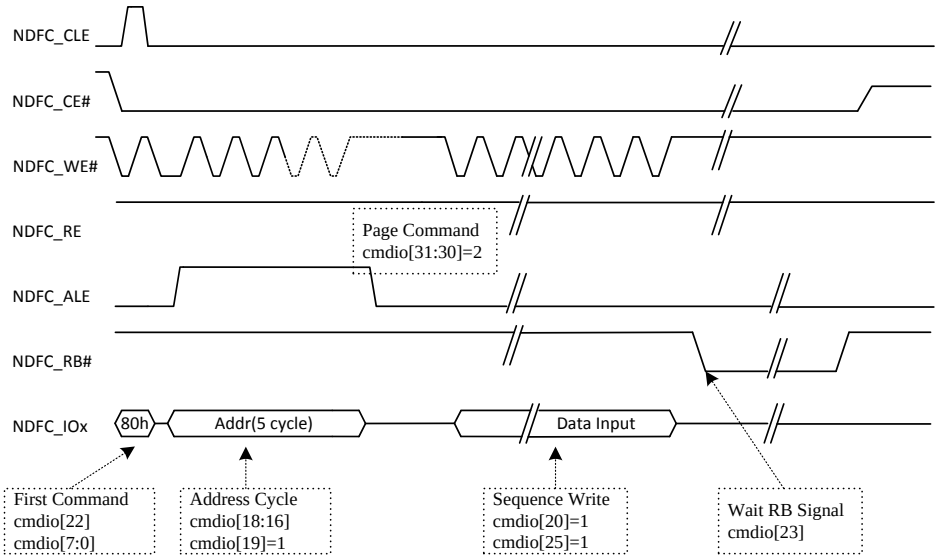


Figure5- 13. Page Program Command Diagram

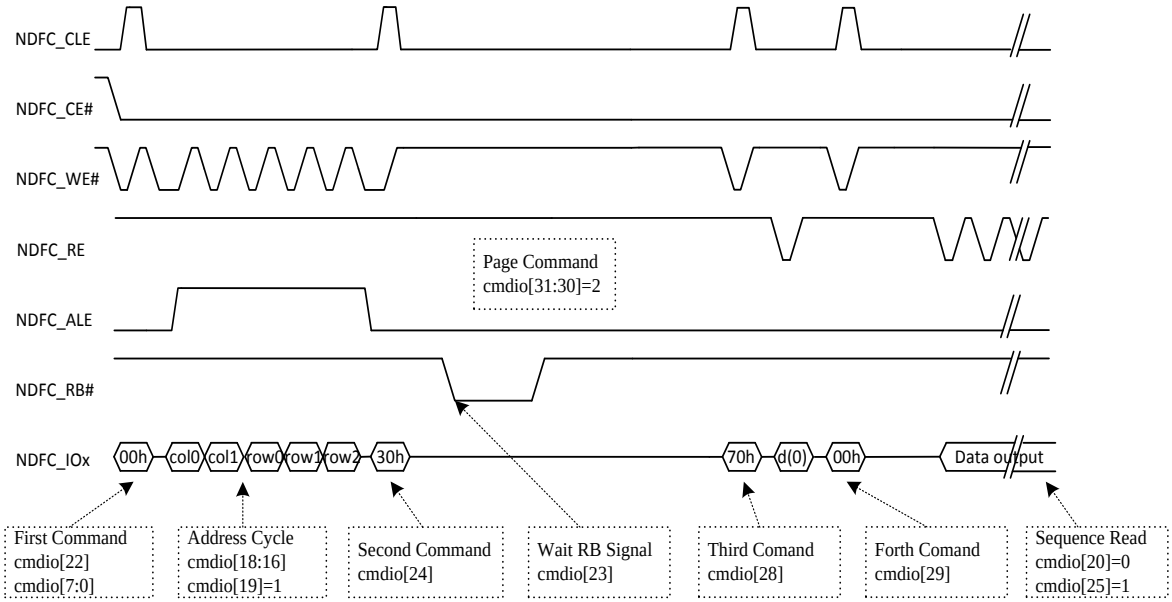


Figure5- 14. EF-NAND Page Read Command Diagram

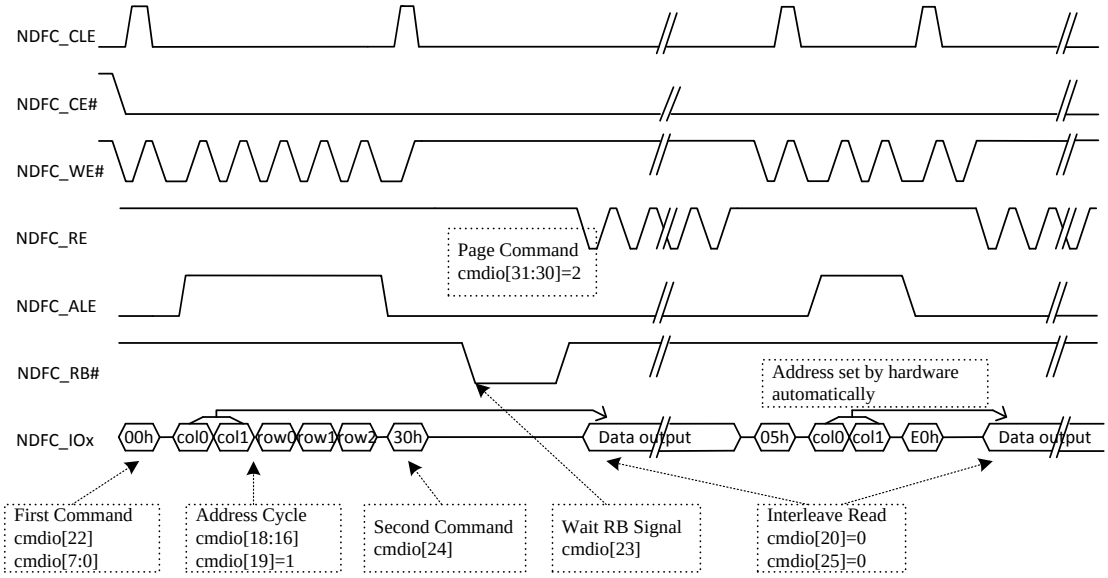


Figure5- 15. Interleave Page Read Command Diagram

5.2.3.5 NDFC Command Descriptors

5.2.3.5.1 NDFC CMD Descriptor Structure

NDFC internal DMA controller can transfer data between DMA FIFO in NDFC and DMA buffer in host memory using CMD descriptors. CMD descriptors reside in the host memory with chain structure shown in Figure 5-16.

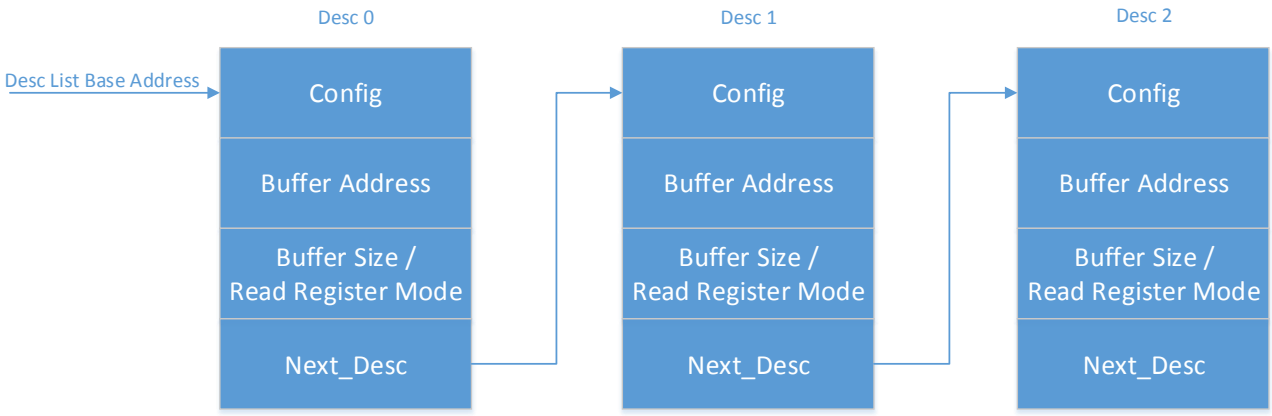


Figure5- 16. Internal CMD Descriptor Chain Structure

The start address of DMA descriptor list must be word (32-bit) aligned, and will be configured to NDFC CMD Descriptor List Base Address Register. Each CMD descriptor is consisted of four words(32-bit).

5.2.3.5.2 NDFC CMD Descriptor Definition

Config	
Bit	**Description**
31:5	/
4	R/W indicator bit When set, this bit indicates that it is writing register, or else it is reading register.
3:2	/
1	LAST_FLAG When set, this bit indicates that the buffers pointed to by this descriptor are the last data buffer
0	FIRST_FLAG When set, this bit indicates that this descriptor points to the first buffer of data.

Buff Addr	
Bit	**Description**
31:16	/
15:0	BUFF_ADDR These bits indicate the physical address of DMA data buffer in host memory. The buffer address must be 8 bytes aligned.

Buffer Size		
Bit	**Description**	
31:0	In the Config field, bit[4] = 1	In the Config field, bit[4] = 0
	BUFF_SIZE These bits indicate the data buffer byte size, which must be 4 bytes aligned. If this filed is 0, the DMA ignores this buffer and points to CMD_DESCR_NEXT which incadites the pointer of the next descriptor physical address.	READ_REGISTER_MODE According to the number of registers that need to be read in the register table, and they must be 8 bytes aligned. The order of reading is “Flash status register”, User_Data, ECC_Status, the number of error correction of each Data Block, starting from

		NDFC_CTL(the offset is 0x00) register in Register List.(Please refer to Figure 5-17.)

NOTE

Bit[4] indicates that whether the register is read register(For example User_Data/ECC_Status). Under the normal read and write NAND Flash operation, All need to configure the write mode.

Figure 5-17 shows the CMD DESCR read register mode, the offset address of DRAM and the corresponding register offset address shown in Figure 5-17.

DRAM偏移值	寄存器偏移值					寄存器对应内容			
0x0000	0x0100	0x0080	0x0084	0x0088		NDFC_FLASH_STA	NDFC_USER_DATA_0	NDFC_USER_DATA_1	NDFC_USER_DATA_2
0x0010	0x008C	0x0090	0x0094	0x0098		NDFC_USER_DATA_3	NDFC_USER_DATA_4	NDFC_USER_DATA_5	NDFC_USER_DATA_6
0x0020	0x009C	0x00A0	0x00A4	0x00A8		NDFC_USER_DATA_7	NDFC_USER_DATA_8	NDFC_USER_DATA_9	NDFC_USER_DATA_10
0x0030	0x00AC	0x00B0	0x00B4	0x00B8		NDFC_USER_DATA_11	NDFC_USER_DATA_12	NDFC_USER_DATA_13	NDFC_USER_DATA_14
0x0040	0x00BC	0x00C0	0x00C4	0x00C8		NDFC_USER_DATA_15	NDFC_USER_DATA_16	NDFC_USER_DATA_17	NDFC_USER_DATA_18
0x0050	0x00CC	0x00D0	0x00D4	0x00D8		NDFC_USER_DATA_19	NDFC_USER_DATA_20	NDFC_USER_DATA_21	NDFC_USER_DATA_22
0x0060	0x00DC	0x00E0	0x00E4	0x00E8		NDFC_USER_DATA_23	NDFC_USER_DATA_24	NDFC_USER_DATA_25	NDFC_USER_DATA_26
0x0070	0x00EC	0x00F0	0x00F4	0x00F8		NDFC_USER_DATA_27	NDFC_USER_DATA_28	NDFC_USER_DATA_29	NDFC_USER_DATA_30
0x0080	0x00FC	0x0038	0x0050	0x0054		NDFC_USER_DATA_31	NDFC_ECC_STA	NDFC_ERR_CNT_0	NDFC_ERR_CNT_1
0x0090	0x0058	0x005C	0x0060	0x0064		NDFC_ERR_CNT_2	NDFC_ERR_CNT_3	NDFC_ERR_CNT_4	NDFC_ERR_CNT_5
0x00A0	0x0068	0x006C	0x0000	0x0004		NDFC_ERR_CNT_6	NDFC_ERR_CNT_7	NDFC_CTL	NDFC_STA
0x00B0	0x0008	0x000C	0x0010	0x0014		NDFC_INTR	NDFC_TIMING_CTL	NDFC_TIMING_CFG	NDFC_ADDR_LOW
0x00C0	0x0018	0x001C	0x0020	0x0024		NDFC_ADDR_HIGH	NDFC_DATA_BLOCK_MASK	NDFC_CNT	NDFC_CMD
0x00D0	0x0028	0x002C	0x0034	0x003C		NDFC_CMD_SET_0	NDFC_CMD_SET_1	NDFC_ECC_CTL	NDFC_DATA_PAT_STA
0x00E0	0x0040	0x0044	0x0048	0x004C		NDFC_EFR	NDFC_RDATA_STA_CTL	NDFC_RDATA_STA_0	NDFC_RDATA_STA_1
0x00F0	0x0070	0x0074	0x0078	0x007C		NDFC_USER_DATA_LEN_0	NDFC_USER_DATA_LEN_1	NDFC_USER_DATA_LEN_2	NDFC_USER_DATA_LEN_3
0x0100	0x0110	0x0114	0x0118	0x011C		NDFC_EFNAND_STA	NDFC_SPARE_AREA	NDFC_PAT_ID	NDFC_DDR2_SPEC_CTL
0x0110	0x0120	0x012C	0x0130	0x0134		NDFC_NDMA_MODE_CTL	NDFC_VALID_DATA_DMA_CNT	NDFC_DATA_DMA_ADDR_0	NDFC_DATA_DMA_ADDR_1
0x0120	0x0138	0x013C	0x0140	0x0144		NDFC_DATA_DMA_ADDR_2	NDFC_DATA_DMA_ADDR_3	NDFC_DATA_DMA_ADDR_4	NDFC_DATA_DMA_ADDR_5
0x0130	0x0148	0x014C	0x0150	0x0154		NDFC_DATA_DMA_ADDR_6	NDFC_DATA_DMA_ADDR_7	NDFC_DATA_DMA_ADDR_8	NDFC_DATA_DMA_ADDR_9
0x0140	0x0158	0x015C	0x0160	0x0164		NDFC_DATA_DMA_ADDR_10	NDFC_DATA_DMA_ADDR_11	NDFC_DATA_DMA_ADDR_12	NDFC_DATA_DMA_ADDR_13
0x0150	0x0168	0x016C	0x0170	0x0174		NDFC_DATA_DMA_ADDR_14	NDFC_DATA_DMA_ADDR_15	NDFC_DATA_DMA_SIZE_2_0	NDFC_DATA_DMA_SIZE_2_1
0x0160	0x0178	0x017C	0x0180	0x0184		NDFC_DATA_DMA_SIZE_2_2	NDFC_DATA_DMA_SIZE_2_3	NDFC_DATA_DMA_SIZE_2_4	NDFC_DATA_DMA_SIZE_2_5
0x0170	0x0188	0x018C	0x0190	0x0194		NDFC_DATA_DMA_SIZE_2_6	NDFC_DATA_DMA_SIZE_2_7	NDFC_RANDOM_SEED_0	NDFC_RANDOM_SEED_1
0x0180	0x0198	0x019C	0x01A0	0x01A4		NDFC_RANDOM_SEED_2	NDFC_RANDOM_SEED_3	NDFC_RANDOM_SEED_4	NDFC_RANDOM_SEED_5
0x0190	0x01A8	0x01AC	0x01B0	0x01B4		NDFC_RANDOM_SEED_6	NDFC_RANDOM_SEED_7	NDFC_RANDOM_SEED_8	NDFC_RANDOM_SEED_9
0x01A0	0x01B8	0x01BC	0x01C0	0x01C4		NDFC_RANDOM_SEED_10	NDFC_RANDOM_SEED_11	NDFC_RANDOM_SEED_12	NDFC_RANDOM_SEED_13
0x01B0	0x01C8	0x01CC	0x0214	0x0218		NDFC_RANDOM_SEED_14	NDFC_RANDOM_SEED_15	NDFC_NDMA_CNT	NDFC_EMCE_CTL
0x01C0	0x021C	0x0220	0x0224	0x0228		NDFC_EMCE_IV_FAC_CMP_VAL	NDFC_EMCE_IV_CAL_FACTOR_0	NDFC_EMCE_IV_CAL_FACTOR_1	NDFC_EMCE_IV_CAL_FACTOR_2
0x01D0	0x022C	0x0230	0x0234	0x0238		NDFC_EMCE_IV_CAL_FACTOR_3	NDFC_EMCE_IV_CAL_FACTOR_4	NDFC_EMCE_IV_CAL_FACTOR_5	NDFC_EMCE_IV_CAL_FACTOR_6
0x01E0	0x023C	0x0240	0x0244	0x0248		NDFC_EMCE_IV_CAL_FACTOR_7	NDFC_EMCE_IV_CAL_FACTOR_8	NDFC_EMCE_IV_CAL_FACTOR_9	NDFC_EMCE_IV_CAL_FACTOR_10
0x01F0	0x024C	0x0250	0x0254	0x0258		NDFC_EMCE_IV_CAL_FACTOR_11	NDFC_EMCE_IV_CAL_FACTOR_12	NDFC_EMCE_IV_CAL_FACTOR_13	NDFC_EMCE_IV_CAL_FACTOR_14
0x0200	0x025C	0x0260	0x0264	0x0268		NDFC_EMCE_IV_CAL_FACTOR_15	NDFC_EMCE_IV_CAL_FACTOR_16	NDFC_EMCE_IV_CAL_FACTOR_17	NDFC_EMCE_IV_CAL_FACTOR_18
0x0210	0x026C	0x0270	0x0274	0x0278		NDFC_EMCE_IV_CAL_FACTOR_19	NDFC_EMCE_IV_CAL_FACTOR_20	NDFC_EMCE_IV_CAL_FACTOR_21	NDFC_EMCE_IV_CAL_FACTOR_22
0x0220	0x027C	0x0280	0x0284	0x0288		NDFC_EMCE_IV_CAL_FACTOR_23	NDFC_EMCE_IV_CAL_FACTOR_24	NDFC_EMCE_IV_CAL_FACTOR_25	NDFC_EMCE_IV_CAL_FACTOR_26
0x0230	0x028C	0x0290	0x0294	0x0298		NDFC_EMCE_IV_CAL_FACTOR_27	NDFC_EMCE_IV_CAL_FACTOR_28	NDFC_EMCE_IV_CAL_FACTOR_29	NDFC_EMCE_IV_CAL_FACTOR_30
0x0240	0x029C	0x02F0	0x02FC	0x0300		NDFC_EMCE_IV_CAL_FACTOR_31	NDFC_VER	NDFC_LDPC_CTL	NDFC_ENC_LDPC_MODE_SET
0x0250	0x0304	0x0308	0x030C	0x0310		NDFC_COR_LDPC_MODE_SET	NDFC_C0_LLR_TBL_0	NDFC_C0_LLR_TBL_1	NDFC_C0_LLR_TBL_2
0x0260	0x0314	0x0318	0x031C	0x0320		NDFC_C0_LLR_TBL_3	NDFC_C0_LLR_TBL_4	NDFC_C0_LLR_TBL_5	NDFC_C0_LLR_TBL_6
0x0270	0x0324	0x0328	0x032C	0x0330		NDFC_C0_LLR_TBL_7	NDFC_C0_LLR_TBL_8	NDFC_C0_LLR_TBL_9	NDFC_C0_LLR_TBL_10
0x0280	0x0334	0x0338	0x033C	0x0340		NDFC_C0_LLR_TBL_11	NDFC_C0_LLR_TBL_12	NDFC_C0_LLR_TBL_13	NDFC_C0_LLR_TBL_14
0x0290	0x0344					NDFC_C0_LLR_TBL_15			

Figure5- 17. NDFC CMD DESCR Read Register

Next Description	
Bit	**Description**
31:0	NEXT_DESC_ADDR These bits indicate the pointer to the physical host memory where the next descriptor is present.

5.2.3.5.3 Buffer Address Description

The content form of the CMD Descriptor's Buffer Address which stored in DRAM is as follows:

Module_0						...	Module_m			...
Offset_0	Cnt_N	Value_0	Value_1	...	Value_N-1	...	Offset_m	Cnt_1	Value_m	...

...	Module_Last			...
...	0x24	Cnt_1	Value_Last	...

As shown in the above table, There are some descriptions

0x24 command register configuration is put into the last position of the Buffer.

Offset_n: the offset address of each register

Cnt_N: The number of consecutive registers starting from Offset_n is N(Cnt_1 indicates that there is 1 register that represent the corresponding offset address)

Value_n: The value of register which offset address is Offset_n($Offset_n$ = ($Offset_{n-1}$) + 4 = … = $Offset_0$ + n*4)

Module_0 contains Offset_0 and N-1 continuous offset address register.

- Configure the offset address for Offset_0 register
- Set the total number of consecutive offset address register to be Cnt_N(That is, 'N').
- Configure the value of register(Value_0/…/Value_N-1), the register's offset address is from Offset_0 to Offset_N-1

When consecutive offset address registers are not included, that is, only including Offset_m register such as Module_m. Set the total number of consecutive offset address register to be Cnt_1, and set the value of Offset_m register to be Value_m.

For example: Writing one Page_Size content to the NAND Flash, and the Batch Command is used.

According to the order of the operation command, it is required to send 80h + 5Address + Data_In + 10h.

Here to send 80h command as an example, the related register of NDFC need to be configured before sending the 80h command. The registers that need to be configured are as follows:

NDFC_CTL(0x00) 、 NDFC_TIMING_CTL(0x0C) 、 NDFC_TIMING_CFG(0x10) 、 NDFC_ADDR_LOW(0x14) 、 NDFC_ADDR_HIGH(0x18) 、 NDFC_DATA_BLOCK_MASK(0x1C) 、 NDFC_CNT(0x20) 、 NDFC_WCMD_SET(0x2C) 、 NDFC_ECC_CTL(0x34)、NDFC_EFR(0x40)、NDFC_USER_DATA_LEN_X(0x70 + 4*x)、NDFC_USER_DATA_X(0x80 + 4*x)、…、NDFC_VALID_DATA_DMA_CNT(0x12C)、NDFC_DATA_DMA_ADDR_0(0x130)、…、NDFC_CMD(0x24).

The content of Buffer Address is configured as follows:

(0x00 + 1 + Value_0x00) + (0x0C + 6 + Value_0x0C + Value_0x10 + Value_0x14 + Value_0x18 + Value_0x1C + Value_0x20) + (0x2C + 1 + Value_0x2C) + (0x34 + 1 + Value_0x34) + (0x40 + 1 + Value_0x40) + (0x70 + 36 + Value_0x70+...+Value_0x80+...) + ... + (0x12C + 41 + Value_0x12C+...+Value_0x130+...+Value_170+...+Value_0x190) + ... + (0x24 + 1 + Value_0x24)

5.2.3.5.4 NDFC CMD Descriptor Operating Instructions

For example:

- Write Page_Size = 16K bytes to NAND Flash. Source data is placed in discrete four addresses of DRAM: 0x42000000、0x42100000、0x42200000、0x42300000, and configure the data is 2K bytes, 3K bytes, 5K bytes and 6K bytes, respectively.
- Read Page_SIZE = 16K bytes to DRAM. Destination data is placed in discrete four addresses of DRAM: 0x43000000、0x43100000、0x43200000、0x43300000, and configure the data is 6K bytes, 2K bytes, 3K bytes and 5K bytes.
- Using a descriptor chain to represent:
  - Using Batch Command to write a Page_Size to NAND Flash
  - Sending 70 command read status
  - Using Batch Command to read a Page_Size to DRAM
  - Reading Flash status + User_Data + ECC_Status

  Suppose the address of 4 descriptor's addresses are 0x50000000、0x50000010、0x50000020、0x50000030. Buffer_Address is 0x60100000、0x60200000、0x60300000、0x60400000.

After determining all of above, you can set the value of the descriptor:

- **First descriptor(Batch Command write a Page_Size into NAND Flash): stored in 0x50000000**

The member value is 0x11(Config), 0x60100000(Buffer Address), Size_CMD_DESCR_1(configured on demand), 0x50000010(Address of the next descriptor).

The related configuration of Data DMA is as follows:

NDFC_VALID_DATA_DMA_CNT = 4， NDFC_DATA_DMA_ADDR_0 = 0x42000000， NDFC_DATA_DMA_ADDR_1 = 0x42100000 ， NDFC_DATA_DMA_ADDR_2 = 0x42200000 ， NDFC_DATA_DMA_ADDR_3 = 0x42300000 ， NDFC_DATA_DMA_BYTE_CNT_0 = 0x0C000800，NDFC_DATA_DMA_BYTE_CNT_1 = 0x18001400.

The content of Buffer_Address is as follows:

... + (0x12C + 5 + 4 + 0x42000000 + 0x42100000 + 0x42200000 + 0x42300000) + (0x170 + 2 + 0x0C000800 + 0x18001400) + ... + (0x24 + 1 + 0x80)

- **Second descriptor(Send 70h command to read Flash status): stored in 0x50000010**

The member value is 0x10(Config), 0x60200000(Buffer Address), Size_cmd_descr_2(configured on demand), 0x50000020(Address of the next descriptor).

The contents of the DRAM Buffer Address are configured as follows:

The register configuration of command retransmission refer to chapter 5.2.3.6 + (0x24 + 1 + 0x70).

- **Third descriptor(Batch Command read a Page_Size into DRAM): stored in 0x50000020**

The member value is 0x10(Config), 0x60300000(Buffer Address), Size_CMD_DESCR_3(Configured on demand), 0x50000030(Address of the next descriptor)

The related configuration of Data DMA is as follows:

NDFC_VALID_DATA_DMA_CNT = 4， NDFC_DATA_DMA_ADDR_0 = 0x43000000， NDFC_DATA_DMA_ADDR_1 = 0x43100000 ， NDFC_DATA_DMA_ADDR_2 = 0x43200000 ， NDFC_DATA_DMA_ADDR_3 = 0x43300000 ， NDFC_DATA_DMA_BYTE_CNT_0 = 0x08001800，NDFC_DATA_DMA_BYTE_CNT_1 = 0x14000C00.

The content of the Buffer_Address is as follows:

... + (0x12C + 5 + 4 + 0x43000000 + 0x43100000 + 0x43200000 + 0x43300000) + (0x170 + 2 + 0x0C000800 + 0x18001400) + ... + (0x24 + 1 + 0x00)

- **Fourth descriptor(Read the Flash status, User_Data and ECC_Status into DRAM): stored in 0x50000030**

The member value is 0x02(Config, Configured to read register mode), 0x60400000(Buffer Address), 1(The original Buffer Size changed to select Read Register Mode), 0x50000040(Address of the next descriptor).

NOTE

- ✧ **The Globel Register can not be configured using CMD Descriptor, and it needs to be set before configuring CMD DESCR.**
- ✧ **When using LDPC Mode, above Source and Destination data needs to be set to 2KB/DATA_DMA_ADDR.**

5.2.3.5.5 CMD Descriptor abnormal instructions

(1). Abnormal Conditions for CMD DESCR Flash Status

- When the CMD_DESCR_FLASH_STA_HANDLE_ENABLE in the NDFC_GLB_CFG register is enabled, and the current descriptor is executed, it is judged whether the condition is satisfied(The result of Flash status AND corresponding bitmap_B is true). If the condition is satisfied, the status value from reading the NAND Flash is regarded as an abnormal condition(Such as erasing/writing operation failed). The CMD_DESCR_FLASH_STA_HANDLE_OCCUR is set to '1'.
- When the CMD_DESCR_FLASH_STA_HANDLE_ENABLE in the NDFC_GLB_CFG register is not enabled, and the current CMD descriptor is executed, it is judged whether the condition is satisfied(The result of Flash status AND corresponding bitmap_B is true). If the condition is satisfied, the CMD_DESCR_FLASH_STA_HANDLE_OCCUR is set to '1'. The subsequent CMD descriptor continue to execute without any other abnormal condition restriction, and the abnormal condition remains to the end of the command descriptor.

(2). Abnormal Conditions for CMD DESCR Time Out

- When the CMD_DESCR_REPT_MODE_TIMEOUT_STA in the NDFC_GLB_CFG is enabled, and the current CMD descriptor is executed, it is judged whether the condition is satisfied(The result of Flash status AND corresponding bitmap_A is not equal to bitmap_A until Time Out). If the condition is satisfied, it will not continue to execute the subsequent CDM descriptor, the CMD_DESCR_REPT_MODE_TIMEOUT_STA is set to '1'.
- When the CMD_DESCR_REPT_MODE_TIMEOUT_STA in the NDFC_GLB_CFG register is not enabled, and the current CMD descriptor is executed, it is judged whether the condition is satisfied(The result of Flash status AND corresponding bitmap_A is not equal to bitmap_A until Time Out). If the condition is satisfied, the corresponding CMD_DESCR_REPT_MODE_TIMEOUT_STA is set to '1'. The subsequent CMD descriptor continue to execute without any other abnormal condition restriction, and the abnormal condition remains to the end of the command descriptor.

(3). Abnormal Conditions for CMD DESCR ECC Error

- When the CMD_DESCR_ECC_ERR_STA in the NDFC_GLB_CFG register is enabled, and the current CMD descriptor is executed, it is judged whether ECC Error is happened. If ECC Error is happened, it will not continue to execute the subsequent CDM descriptor, the CMD_DESCR_ECC_ERR_STA is set to '1'.
- When the CMD_DESCR_ECC_ERR_STA in the NDFC_GLB_CFG register is not enabled, and the current CMD descriptor is executed, it is judged whether ECC Error is happened. If ECC Error is happened, the CMD_DESCR_ECC_ERR_STA is set to '1'. The subsequent CMD descriptor continue to execute without any other abnormal condition restriction, and the abnormal condition remains to the end of the command descriptor.

NOTE

- **Bitmap_A and Bitmap_B are configured in the NDFC_FLASH_STA register.**
- **The command descriptor execution CMD_DESCR_STA is set to '1' as long as CMD Descriptor execution is stopped.(Normal or abnormal stop).**
- **The descriptor interrupt enable is valid during use of the command descriptor chain. The internal interrupt is enabled until the end of the command descriptor chain.**

5.2.3.6 NDFC Command Retransmission Operation

The retransmission command function is used to check the status of NAND Flash. It is mainly used for scenarios such as 0x70 reading status command.

1) Configure the interval of retransmission command(set bit[15:0] in NDFC_CMD_REPT_INVL register)、the maximum number of retransmission commands(set bit[15:0] in NDFC_CMD_REPT_CNT register)、bitmap_A、bitmap_B.

2) Enable the retransmission command (Bit[24] is set to 1 in NDFC_FLASH_STA register). Sending 0x70 read status command will automatically re-send 0x70 command, the synchronize the read data to the bit[7:0] of NDFC_FLASH_STA register (When there is no command to produce data reading , the synchronization value is 0xFF by default.)

3) After updating the data of Flash Status each time, the data will be compared with bitmap_A(bit[15:8]) in NDFC_FLASH_STA register. If the result is equal to bitmap_A or TimeOut, the retransmission command is stopped.

Where,

(a). If the result of Flash Status AND bitmap_A is equal to bitmap_A, it will be further compared with bitmap_B. If the result of Flash status data AND bitmap_B is true (Indicating the operation is failure), the Flash Status flag(bit[1] of NDFC_CMD_DESCR_STA register) will be set up, and then the retransmission command is stopped, or the retransmission command is stopped directly.

(b). If TimeOut happened, the flag of CMD REPT TimeOut (bit[5] of NDFC_STA register and bit[2] of NDFC_CMD_DESCR_STA register) is set up. And then retransmission command is stopped. If the interrupt of TimeOut enabled, it will stop retransmission command when TimeOut is happened, it will enter interrupt program at the same time, and CMD REPT TimeOut flag (bit[5] of NDFC_STA register and bit[2] of NDFC_CMD_DESCR_STA register) is set up.

NOTE

- **Under the command descriptor mode, If the TimeOut interrupt is enable, the retransmission command will be stopped when TimeOut happened. And then the interrupt program will be entered according to the CMD REPT TimeOut flag(Bit[2] in NDFC_CMD_DESCR_STA register).**
- **Under the non-command descriptor mode, If the TimeOut interrupt is enable, the retransmission command will be stopped when TimeOut happened. And then the interrupt program will be entered according to the CMD REPT TimeOut flag(Bit[5] in NDFC_CMD_DESCR_STA register).**
- **It is suggested that when TimeOut happens in command descriptor or non-command descriptor mode, clear all TimeOut flag(Bit[2] in NDFC_CMD_DESCR_STA register and Bit[5] in NDFC_CMD_DESCR_STA register) after using CMA REPT TimeOut flag.**

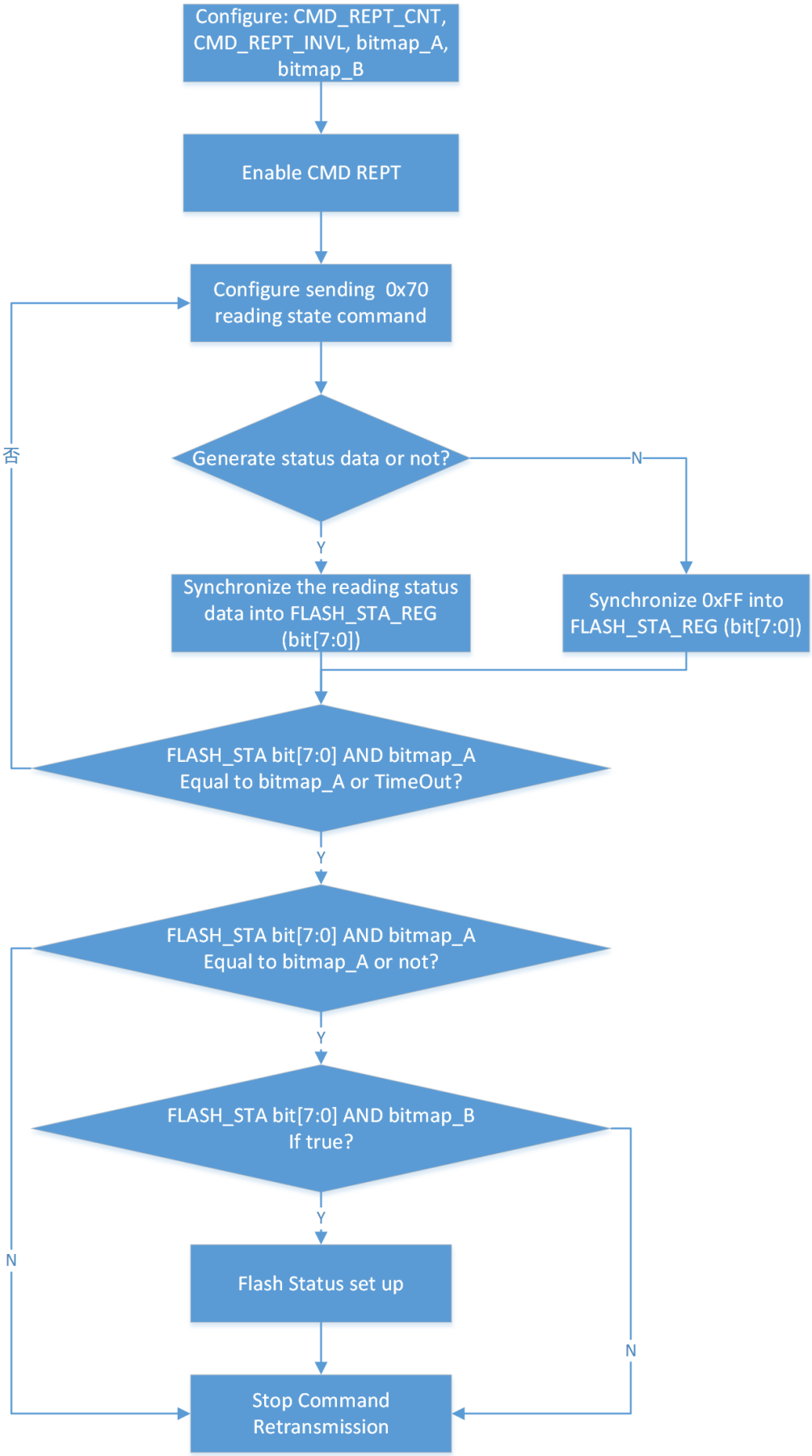


Figure5- 18. Command Repeat Flow Chart

5.2.3.7 Read Soft Bit Information-SRAM

SRAM Name	Main RAM	Spare RAM
SRAM_0	2K bytes	≈2KB/8+User_Data/8=288 bytes
SRAM_1	2K bytes	≈2KB/8+User_Data/8=288 bytes
SRAM_2	2K bytes	≈2KB/8+User_Data/8=288 bytes
SRAM_3	2K bytes	≈2KB/8+User_Data/8=288 bytes
SRAM_4	2K bytes	≈2KB/8+User_Data/8=288 bytes

The Attention Note when using Soft_Decode:

- 2KB: Size of Spare area corresponding to the general Page_Size.
- /8: Taking the Page_Size of 16K byte as an example, the size of Spare area corresponding to each 2K byte(Main_Area) contains the data of User_Data/8, which is approximately equal to 264bytes.
- After adding LDPC, the new revision of SRAM has 5 areas including SRAM0~SRAM4, each SRAM contains Main RAM and Spare RAM. The operation of Set Feature will choose to be placed in SRAM_0.
- When reading Soft-Bit information, it firstly reads Hard-Bit, and then reads the Soft-Bit. The Hard-Bit adopts Batch Command, and the reading data is used to decode and pass to the DRAM through DMA. When the Soft-Bit adopts Batch Command, the data will be transferred out by default.
- Soft-Bit read the data from HB0, SB0, SB1, SB2 using Batch CMD and placed in SRAM1, SRAM2, SRAM3, SRAM4, respectively, Finally read the data from SB3 using Batch CMD and placed in SRAM0. Please note that DMA parameters need to be configured and Soft-Bit Decode need to be enabled before reading data from SB3.
- When the decode mode of Soft-Bit Decode can be set to 1/2/3(Batch CMD). To set the Decode Mode = 2 as an example, Firstly, Batch CMD read data from HB0, SB0 and SB1, and then palce them into SRAM1, SRAM2 and SRAM3. Finally, Batch CDM read from SRAM3 repeatly and place it into SRAM0. Please note that DMA parameters need to be configured and Soft-Bit Decode need to be enabled before reading data for the last time.
- When Soft-Bit decode use Normal CMD for reading Main Area Data and Spare Area Data, To set Decode Mode = 4 as an example, Firstly, it needs to read Main Area Data and Spare Area Data for 4 times(HB0 + SB0 + SB1 + SB2, Read Main Area Data firstly, and Spare Area Data followed) with Normal CMD twice used, and then placed the data into SRAM1, SRAM2, SRAM3, SRAM4, respectively. Finally, it needs to read Main Area Data and Spare Area Data for once(SB3 Data, Read Spare Area Data firstly, and Main Area Data followed) with Normal CMD twice used, and then placed the data into SRAM0. Please note that DMA parameters need to be configured and Soft-Bit Decode need to be enabled before reading Main Area data for the last time.
- When the decode mode of Soft-Bit Decode can be set to 1/2/3(Normal CMD twice used), to set Decode Mode = 2 as an example. Firstly, it needs to read Main Area Data and Spare Area Data for 3 times(HB0 + SB0 + SB1, Read Main Area Data firstly, and Spare Area Data followed) with Normal CMD twice used, and then placed the data into SRAM1, SRAM2, SRAM3, respectively. Finally, read SRAM3 repeatly(Read Spare Area Data firstly, and Main Area Data followed)and place it into SRAM0 with Normal CMD twice used. Please note that DMA parameters need to be configured and Soft-Bit Decode need to be enabled before reading Main Area data for the last time.
- Normal command mode(Non Command descriptor mode): Set LDPC Decode mode =4 as an example, When Normal CMD write NAND Flash, Set Feature operation can be performed between reading Soft-Bit information for 5 times. When Normal CMD read NAND Flash, Get Feature operation is not allowed.
- Command descriptor mode: Set LDPC Decode mode =4 as an example, When Normal CMD write NAND Flash, Set Feature operation can be performed by using AHB(Not by using DMA) between reading Soft-Bit information for 5 times. When Normal CMD read NAND Flash, Get Feature operation is not allowed.

5.2.3.8 NDFC Operation Attention Note

1. Operation Attention Note when using NDFC Data Block Mask Register:

- For Sequence mode, Only continuous Data_Block can be supported. Ther can be no interval in the middle. That is , the value of this register can only be 0x1, 0x3, 0x7 and so on, but Interleave mode does not have this limitation.
- Regardless of Sequence mode or Interleave mode, the corresponding column address is calculated on the basis of the first read Data_Block marked in this register, and written to the 0x14 and 0x18 address registers.

2. Operation Attention Note when using NADFC Enhanced Feature Register:

This register is used to set whether the remaining space is filled with random data when Batch Command function is in used. For example, for a SanDisk sample(SDTNQGAMA-008G), The Page_Size is (16384 + 1280) bytes, the BCH level uses 40-bit/1K bytes. Suppose the User_Data is 32-byte, and the total byte is 1152-byte(14 * 40/8) * 16 + 32 = 1152-byte), and the remaining space is 128bytes(1280 – 1152 = 128 bytes). If you want to write a page, bit[24] of this register can be set to 1, then bit[23:16] = 0x80, the controller can fill 128bytes of random data automatically(Please note that the randomization function must be enabled when you want to send the random data. That is, NDFC_RANDOM_EN is set to 1 in 0x34 register, otherwise the padding data will not be random, but 0.)

3. Operation Attention Note when using NDFC Normal Special Command:

When NDFC Channel 0 is selected(BCH Mode & 1KB DATA_BLOCK_SIZE & RANDOMIZER_OLD), and when using Normal+Special command, the related operating flow is following.

1) Writing process (Interleave Mode)

- Set NDFC_DATA_BLOCK_MASK to 0x01;
- Set the length of User_Data that needs to be written(Corresponding to NDFC_USER_DATA_LEN_0(0x0070) register), and set the value of User_Data(corresponding to NDFC_USER_DATA_0 (0x0080)~NDFC_USER_DATA_X register);
- Using Normal command, 80h + 5Address (3 row_addr + 2 col_addr) + 1KB_Main_Data_In
- Using Special command, 85h + 2Address (2 col_addr) 85h + 2Address (2 col_addr)
- Send 10h command

2) Reading process (Interleave Mode)

- Set NDFC_DATA_BLOCK_MASK to 0x01;
- Set the length of User_Data that needs to be read (Corresponding to NDFC_USER_DATA_LEN_0(0x0070)register)
- Using Normal Command, 00h + 5Address (3 row_addr + 2 col_addr) + 30h + 1KB_Main_Data_Out
- Using Special Command, 05h+2Address (2 col_addr) + E0h
- Reading the content of User_Data, that is, set the NDFC_USER_DATA_LEN_0(0x0070) register according to the length of User_Data. Read the NDFC_USER_DATA_X(0x0080+N*4) register corresponding to the content of User_Data.

NOTE

- **Normal + Special writing operation combined with DMA. Firstly when using Normal CMD, configure the DMA to set up DMA(corresponding to bit[26] of NDFC_CTL register). Note that NDFC_VALID_DATA_DMA_CNT need set to 0 before using Special CMD, and DMA can not be set up when using Special CMD.**
- **Normal + Special reading operation combined with DMA. Firstly when using Normal CMD, configure the DMA, and DMA can not be set up.(corresponding to bit[26] of NDFC_CTL register). And when using Special CMD, configure the DMA to be set up.**

4. Operation Attention Note when using NDFC + EMCE:

- NDFC Channel 0, When Normal CMD used with EMCE, the data requires 512-byte alignment.

- NDFC Channel 1, When Normal CMD used with EMCE, the data requires 2048-byte alignment.
- When NDFC used with EMCE, that need to open the ECC engine.

5. Operation Attention Note when NDFC Channel 1 used with LDPC:

- When configuring LDPC Mode and FW_EXT of Decode, you need to configure the LDPC Mode and FW_EXT at the same time.
- When using LDPC Encode or Decode in NDFC Channel 1, 0x300 and 0x304 register need to be configured once during initialization.
- If Soft-Reset controller is used in NDFC Channel 1, 0x300(Encode register) and 0x304(Decode register) register need to be reconfigured.

6. Operation Attention Note when NDFC Channel 0 and NDFC Channel 1 is switched:

When NDFC Channel 0 and NDFC Channel 1 is switched, it will require Soft-Reset controller.

7. Operation Attention Note when using Soft-Bit Decode:

It requires Soft-Reset controller before configuring information for reading Soft-Bit.

8. NDFC Channel 1 Non-CMD Descriptor With MBUS DMA, a Soft-Reset controller is required before the CMD Descriptor.

9. Using Normal CMD with data return (such as read state operation) after Normal + Special CMD in CPU mode of NDFC Channel 1, the Soft-Reset controller is required before sending Normal CMD with data returned.

10. Operation Attention Note when using CMD REPT:

When using CMD REPT to judge Ready/Busy and Pass/Fail, the operation needs to be performed in two steps. First, RB#Status is judged by using the command retransmission function(Pass/Fail need to be configured as invalid), and then use the command retransmission function to judge Pass/Fail.

11. Operation Attention Note when NDFC Channel 1 DMA used with NDFC_CNT:

- When MBUS DMA configure the DATA_DMA_SIZE register, the DATA_DMA_SIZE_2X(X = 0~7) and DATA_DMA_SIZE_2X_1 in the DATA_DMA_SIZE register need to be configured to 2KB(When the data size is more than or equal to 2KB), or configure the actual value according to the actual demand(the data size is less than 2KB).
- When MBUS DMA configure the DATA_DMA_SIZE register, the DATA_DMA_SIZE_2X(X = 0~7) and DATA_DMA_SIZE_2X_1(X = 0~7) need to be configured as 8-byte aligned and 0 values.
- The transmission data size of MBUS DMA is 8-byte alignment. The difference between the setting value in NDFC_CNT register and the transmission data size of MBUS DMA is less than 4 bytes
- The transmission data size of MBUS DMA is more than 32bytes. The MBUS DMA start address needs to be 4-word(32-byte) aligned.
- The transmission data size of Normal DMA is 4-byte alignment. The Normal DAM starting address needs to be 4-byte aligned. The difference between the setting value in NDFC_CNT register and the transmission data size of Normal DMA is less than 4bytes.

12. Operation Attention Note when using NDFC No Trans function(Main space data does not transferred to DRAM):

Only NDFC Channel 0 supports NDFC No Trans function.

13. Operation Attention Note when using NDFC disorder function:

The disorder function in only valid for MBUS DMA, and this function needs to be disabled under the Normal DMA

mode.

14. Operation Attention Note when using NDFC Channel 1 LDPC Encode and Decode:

- LDPC Hard-Bit Mode: That need configure the encoding and decoding register:
  - Set the NDFC_LDPC_CTL[CH1_HB_LLR_VAL] to the default value
  - NDFC_ENC_LDPC_MODE_SET(0x300) and NDFC_COR_LDPC_MODE_SET(0x304) need to be set, and LDPC decode channel 0 and LDPC Decode channel 1 need to be configured for the same parameters.
- LDPC Soft-Bit Mode(Batch CMD): That need configure the encoding and decoding register:
  - NDFC_C0_LLR_TBL_0 (0x308) ~ NDFC_C1_LLR_TBL_7 (0x344)
  - NDFC_LDPC_CTL (0x2FC)[CH1_SOFT_BIT_IND], NDFC_LDPC_CTL (0x2FC)[CH1_SOFT_BIT_DEC_EN]
  - NDFC_ENC_LDPC_MODE_SET (0x300)
  - NDFC_COR_LDPC_MODE_SET (0x304)

  where, LDPC decode channel 0 and LDPC Decode channel 1 need to be configured for the same parameters.
- LDPC Soft-Bit Mode(Normal CMD with twice used): That need configure the encoding and decoding register:
  - NDFC_C0_LLR_TBL_0 (0x308) ~ NDFC_C1_LLR_TBL_7 (0x344)
  - NDFC_LDPC_CTL (0x2FC)[CH1_SOFT_BIT_IND]
  - NDFC_LDPC_CTL (0x2FC)[CH1_SRAM_MAIN_OR_SPARRER_AREA_IND]
  - NDFC_LDPC_CTL (0x2FC)[CH1_SRAM_IND]
  - NDFC_LDPC_CTL (0x2FC)[CH1_SOFT_BIT_DEC_EN]
  - NDFC_ENC_LDPC_MODE_SET(0x300)
  - NDFC_COR_LDPC_MODE_SET(0x304)

  where, LDPC decode channel 0 and LDPC Decode channel 1 need to be configured for the same parameters.

5.2.4 Programming Guidelines

5.2.4.1 Initializing Nand Flash

The NAND Flash is initialized as follows:

Step1: Read NDFC_STA[NDFC_RB_STA_0] to wait flash idle.

Step2: Write 1 to NDFC_CMD[NDFC_WAIT_RB_EN, NDFC_SEND_FIRST_CMD_EN], send the first command and set wait RB; write 0xFF to NDFC_CMD[NDFC_CMD_LOW_BYTE] to send reset command.

Step3: Read NDFC_ST[NDFC_CMD_INTR_STA] to wait *transfer command end interrupt flag* pending, after pending, write 1 to clear the flag.

5.2.4.2 Erasing Nand Flash

The NAND Flash is erased as follows:

Step1: Read NDFC_ST[NDFC_RB_STAT_0] to wait flash idle.

Step2: Configure NDFC_CMD[NDFC_SEND_FIRST_CMD_EN] to 1 to send the first command, configure NDFC_CMD[NDFC_WAIT_FLAG] to 1 to set wait RB;
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.

Step3: Read NDFC_ST[NDFC_CMD_INT_STA] to wait *transfer command end interrupt flag* pending, after pending, write

1 to clear the flag.

Step4: Read NDFC_ST[NDFC_RB_STAT_0] to wait flash idle.

Step5: Set NDFC_CMD[NDFC_WAIT_FLAG] to 1 to ensure wait RB, set NDFC_CMD[NDFC_SEND_FIRST_CMD_EN] to 1 to send the first command; set NDFC_CMD[NDFC_CMD_LOW_BYTE] to 0xD0 to send erasing command.

Step6: Read NDFC_ST[NDFC_CMD_INT_STA] to wait *transfer command end interrupt flag* pending, after pending, write 1 to clear the flag.

Step7: 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_EN, NDFC_DATA_TRANS_EN] to 0x1 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.

5.2.4.3 Writing Nand Flash

Step1: Erase the address of the block to be operated.

Step2: Read NDFC_ST[NDFC_RB_STAT_0] to wait flash idle.

Step3: Configure RAM0_BASE to write data to Flash.

Step4: Configure NDFC_CNT[NDFC_DATA_CNT] to set transferred data;

Set NDFC_CMD[NDFC_SEND_FIRST_CMD_EN, NDFC_DATA_TRANS_EN, NDFC_ACCESS_DIR] to 1 to send the first command and 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.

Step5: Read NDFC_ST[NDFC_RB_STAT_0] to wait flash idle.

Step6: Configure NDFC_CMD[NDFC_WAIT_RB_EN, NDFC_SEND_FIRST_CMD_EN] to 1 to send the first command and set wait RB; configure NDFC_CMD[NDFC_CMD_LOW_BYTE] to 0x10 to send end command.

Step7: Read NDFC_ST[NDFC_CMD_INTR_STA] to wait *transfer command end interrupt flag* pending, after pending, write 1 to clear the flag.

5.2.4.4 Reading Nand Flash

Step1: Read NDFC_ST[NDFC_RB_STAT_0] to wait flash idle.

Step2: Configure NDFC_CNT[NDFC_DATA_CNT] to set the transferred data;

Configure NDFC_CMD[NDFC_SEND_FIRST_CMD_EN] to 1 to send the first command;

Configure NDFC_CMD[NDFC_ACCESS_DIR] to 0 to set thetransfer direction as reading;

Configure NDFC_CMD[NDFC_SEND_ADDR] to 1 to enable the transfer address;

Cobfigure 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 0x00 to send page read command.

Step3: Read NDFC_STA[NDFC_RB_STAT_0] to wait flash idle.

Step4: Configure NDFC_CMD[NDFC_WAIT_RB_EN, NDFC_SEND_FIRST_CMD_EN] to 1 to send the first command and set wait RB; configure NDFC_CMD[NDFC_CMD_LOW_BYTE] to 0x30 to send end command.

Step5: Read RAM0_BASE to get data from flash.

Step6: Read NDFC_ST[NDFC_CMD_INTR_STA] to wait *transfer command end interrupt flag* pending, after pending, write 1 to clear the flag.

5.2.5 Register List

Module Name	Base Address
NDFC0	0x04011000

Register Name	Offset	Description
NDFC_CTL	0x0000	NDFC Configure and Control Register
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 for data transfer Register
NDFC_CMD	0x0024	Set up NDFC commands Register
NDFC_RCMD_SET_0	0x0028	Read Command Set Register for vendor's NAND memory
NDFC_WCMD_SET_1	0x002C	Write Command Set Register for vendor's NAND memory
NDFC_ECC_CTL	0x0034	ECC Configure and Control Register
NDFC_ECC_ST	0x0038	ECC Status and Operation information Register
NDFC_DATA_PAT_STA	0x003C	NDFC Data Pattern Status Register
NDFC_EFR	0x0040	NDFC Enhanced Feature Register
NDFC_RDATA_STA_CTL	0x0044	Read Data Status Control Register
NDFC_RDATA_STA_0	0x0048	Read Data Status Register 0
NDFC_RDATA_STA_1	0x004C	Read Data Status Register 1
NDFC_ERR_CNT_N	0x0050+0x04*N	NDFC Error Counter Register(N from 0 to 7)
NDFC_USER_DATA_LEN_N	0x0070+0x04*N	NDFC User Data Length Register(N from 0 to 3)
NDFC_USER_DATA_N	0x0080+0x04*N	User Data Field Register N (N from 0 to 31)
NDFC_FLASH_STA	0x100	Flash Status Register
NDFC_CMD_REPT_CNT	0x104	NDFC Command Repeat Counter
NDFC_CMD_REPT_INVL	0x108	NDFC Command Repeat Interval
NDFC_EFNAND_STA	0x0110	EFNAND Status Register
NDFC_SPARE_AREA	0x0114	Spare Area Configure Register
NDFC_PAT_ID	0x0118	Pattern ID Register
NDFC_DDR2_SPEC_CTL	0x011C	NDFC DDR2 Configuration Control Register
NDFC_NDMA_MODE_CTL	0x0120	NDFC Normal DMA Mode Control Register
NDFC_VALID_DATA_DMA_CNT	0x12C	Valid Data DMA Counter Register
NDFC_DATA_DMA_ADDR_N	0x130+0x04*N	Data DMA Address Register (N from 0 to 15)
NDFC_DATA_DMA_SIZE_2N	0x170+0x04*2N	Data DMA Size 2N and Data DMA Size 2N+1 Register (N from 0 to 7)
NDFC_RANDOM_SEED_N	0x190+0x04*N	Random Seed Register (N from 0 to 15)
NDFC_DMA_CNT	0x0214	NDFC DMA Byte Counter Register
NDFC_EMCE_CTL	0x0218	NDFC EMCE Control Register
NDFC_EMCE_IV_FAC_CMP_VAL	0x021C	NDFC EMCE IV_FAC Compare Value Register

NDFC_EMCE_IV_CAL_FACTOR_N	0x0220+0x04*N	NDFC EMCE IV Calculate Factor Register N(N from 0 to 31)
NDFC_IO_DATA	0x2A0	Data Input/ Output Port Address Register
NDFC_VER	0x2F0	NDFC Version Number Register
NDFC_LDPC_CTL	0x2FC	NDFC Soft-Bit Control Register
NDFC_ENC_LDPC_MDOE_SET	0x300	Encode LDPC Mode Setting Register
NDFC_COR_LDPC_MODE_SET	0x304	Correct LDPC Mode Setting Register
NDFC_C0_LLR_TBL_0	0x308	C0 LLR Table 11111-11100 Register
NDFC_C0_LLR_TBL_1	0x30C	C0 LLR Table 11011-11000 Register
NDFC_C0_LLR_TBL_2	0x310	C0 LLR Table 10111-10100 Register
NDFC_C0_LLR_TBL_3	0x314	C0 LLR Table 10011-10000 Register
NDFC_C0_LLR_TBL_4	0x318	C0 LLR Table 01111-01100 Register
NDFC_C0_LLR_TBL_5	0x31C	C0 LLR Table 01011-01000 Register
NDFC_C0_LLR_TBL_6	0x320	C0 LLR Table 00111-00100 Register
NDFC_C0_LLR_TBL_7	0x324	C0 LLR Table 00011-00000 Register
NDFC_C1_LLR_TBL_0	0x328	C1 LLR Table 11111-11100 Register
NDFC_C1_LLR_TBL_1	0x32C	C1 LLR Table 11011-11000 Register
NDFC_C1_LLR_TBL_2	0x330	C1 LLR Table 10111-10100 Register
NDFC_C1_LLR_TBL_3	0x334	C1 LLR Table 10011-10000 Register
NDFC_C1_LLR_TBL_4	0x338	C1 LLR Table 01111-01100 Register
NDFC_C1_LLR_TBL_5	0x33C	C1 LLR Table 01011-01000 Register
NDFC_C1_LLR_TBL_6	0x340	C1 LLR Table 00111-00100 Register
NDFC_C1_LLR_TBL_7	0x344	C1 LLR Table 00011-00000 Register
NDFC_GLB_CFG	0xC00	NDFC Global Configure Register
NDFC_CMD_DESCR_BASE_ADDR	0xC04	First Command Descriptor Memory Address
NDFC_CMD_DESCR_STA	0xC08	NDFC Command Descriptor Status Register
NDFC_CMD_DESCR_INTR	0xC0C	NDFC Command Descriptor Interrupt Control Register

5.2.6 Register Description

5.2.6.1 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 The type of DDR data interface. This bit is valid when NF_TYPE is 0x2 or 0x3. 0: DDR 1: DDR2 This bit is used to set DDR1.0 or DDR2.0 type. And it is valid only when bit[19:18] is set to 0x10 or 0x11.
27:24	R/W	0x0	NDFC_CE_SEL Chip Select for NAND Flash Chips 0000: NDFC Select Chip 0 is selected 0001: NDFC Select Chip 1 is selected

			This bit is used to chose chip select.
23:22	/	/	/
21	R/W	0x0	NDFC_DDR_REPT_MODE DDR Repeat data mode 0: Lower byte 1: Higher byte When DDR mode is in used, and if only one of the edge values needs to be sampled, this bit is used to select the value of rising or falling eadge.
20	R/W	0x0	NDFC_DDR_REPT_EN DDR Repeat Enable 0: Disable 1: Enable When DDR mode is in used, regardless of reading or writing data, it always samples based on the rising or falling eadge of DQS.But for special instructions, for example, Read ID value. The value of DQS rising edge is same as DQS falling edge, so that master only need to select one edge to sample values. Furthermore, Set Feature and Get Feature all operate the internal register of NAND, and it need write(or read) the value only once in DDR mode. In order to avoid mistakes, it need to enable this bit,so that master can sample one edge value in DDR mode. As for choosing to keep rising edge or falling edge to sample values depending on bit[21].
19:18	R/W	0x0	NF_TYPE NAND Flash Type 00: Normal SDR NAND 01: Reserved 10: ONFI DDR NAND 11: Toggle DDR NAND This bit is used to chose the mode of accessing NAND.
17	R/W	0x0	NDFC_CLE_POL NDFC Command Latch Enable (CLE) Signal Polarity Select 0: High active 1: Low active This bit is used to chose active level of CLE(Generally, it is set high active)
16	R/W	0x0	NDFC_ALE_POL NDFC Address Latch Enable (ALE) Signal Polarity Select 0: High active 1: Low active This bit is used to chose active level of ALE(Generally, it is set high active)
15	R/W	0x0	NDFC_DMA_TYPE 0: Dedicated DMA 1: Normal DMA When DMA mode is used to transfer data(bit[14]: NDFC_RAM_METHOD = 1), DMA type is in used. ‘0’ indicates internal DMA and MBUS channel is in used; ‘1’ indicates Normal DMA and AHB channel is in used.  **NOTE**

			When using the command description mode, this bit must set to '1'. Command description mode only supports MBUS DMA for data transmission. **When NDFC Channel 1 use Normal DMA mode, it neet close out-of-sequency.**
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 This bit is used to choose data interactively between controller internal 2KB RAM and DRAM(or sram) controller. '0' indicates AHB bus, '1'indicates DMA. When using the command description mode, this bit must set to '1'. Command description mode only supports MBUS DMA data transmission.
13:12	/	/	/
11:8	R/W	0x0	NDFC_PAGE_SIZE 0000: 1KB 0001: 2KB 0010: 4KB 0011: 8KB 0100: 16KB 0101: 32KB 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 control Chip Select Signals. 1: Chip select signal NDFC_CE# is always active after NDFC is enabled
5	/	/	/
4:3	R/W	0x0	NDFC_RB_SEL NDFC external R/B Signal select The value 0-1 selects the external R/B signal. The same R/B signal can be used for multiple chip select flash. This bit is used to chose the external RB# signal.
2:0	/	/	/

5.2.6.2 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 current read operation is more than threshold value. 1: The number of bit 1 during current read operation is less than or equal to threshold value.

			This field only is valid when NDFC_RDATA_STA_EN is 1. The threshold value is configured in NDFC_RDATA_STA_TH. This bit set to '1' indicates that current read operation found specific data pattern(0x00) under the constraints of the threshold value. **NOTE** **Need to enable the function that whether to calculation the number of '0' and '1' in the current read operation. Enable bit is set in the NAFC_RDATA_STA_CTL register(0x44).** **When the number of '1' in transfer data are less than the threshold value, this bit is set to '1'. Threshold value is set by NDFC_RDATA_STA_TH of NDFC_RDATA_STA_CTL register** **Need to set the number of User_Data to the same value for each Data_Block.**
12	R	0x0	NDFC_RDATA_STA_1 0: The number of bit 0 during current read operation is more than threshold value. 1: The number of bit 0 during 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. This bit set to '1' indicates that current read operation found specific data pattern(0xFF) under the constraints of the threshold value. **NOTE** **Need to enable the function that whether to calculation the number of '0' and '1' in the current read operation. Enable bit is set in the NAFC_RDATA_STA_CTL register(0x44).** **When the number of '0' in transfer data are less than the threshold value, this bit is set to '1'. Threshold value is set by NDFC_RDATA_STA_TH of NDFC_RDATA_STA_CTL register.** **Need to set the number of User_Data to the same value for each Data_Block.**
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 This bit indicates the status of external RB#3 signal: busy or idle 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 This bit indicates the status of external RB#2 signal: busy or idle 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 This bit indicates the status of external RB#1 signal: busy or idle 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 This bit indicates the status of external RB#0 signal: busy or idle state.
7:6	/	/	/
5	R/W1C	0x0	CMD_REPT_TIMEOUT_STA 0: No Time_Out 1: Time_Out happened This bit indicates resend command timeout flag. Write '1' to clear the status bit.
4	R	0x0	NDFC_STA 0: NDFC FSM in IDLE state 1: NDFC FSM in BUSY state When NDFC_STA is 0, NDFC can accept new command and process command. This bit indicates the status of NDFC internal state machine: busy or idle state.
3	R	0x0	NDFC_CMD_FIFO_STA 0: Command FIFO not full and can receive new command 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. This bit indicates the status of NDFC internal command FIFO: busy or idle state.
2	R/W1C	0x0	NDFC_DMA_INTR_STA When it is 1, it means that a pending DMA is completed. It will be clear after writing 1 to this bit or it will be automatically clear before FSM processing an new command. This bit is end of DMA interrupt flag. Write '1' to clear the status bit.
1	R/W1C	0x0	NDFC_CMD_INTR_STA When it is 1, it means that NDFC has finished one Normal Command Mode or one Batch Command Work Mode. It will be clear after writing 1 to this bit or it will be automatically clear before FSM processing an new command. This bit is end of command interrupt flag. Write '1' to clear the status bit.
0	R/W1C	0x0	NDFC_RB_B2R_STA When it is 1, it means that NDFC_R/B# signal is transferred from BUSY state to READY state. It will be clear after writing 1 to this bit. This bit is the flag of RB# signal changing from busy state to idle state. Write '1' to clear the status bit.

5.2.6.3 NDFC Interrupt and DMA Enable Register(Default Value: 0x0000_0000)

Offset: 0x0008			**Register Name: NDFC_INTR**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:6	/	/	/
5	R/W	0x0	NDFC_CMD_REPT_INTR_EN 0: Disable 1: Enable

4:3	/	/	/
2	R/W	0x0	NDFC_DMA_INT_EN Enable or disable interrupt when a pending DMA is completed.
1	R/W	0x0	NDFC_CMD_INT_EN 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: Disable 1: Enable
0	R/W	0x0	NDFC_B2R_INT_ENABLE Enable or disable interrupt when NDFC_RB# signal is transferring from BUSY state to READY state 0: Disable 1: Enable

5.2.6.4 NDFC Timing Control Register(Default Value: 0x0000_0000)

Offset: 0x000C			Register Name: NDFC_TIMING_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:12	/	/	/
11:8	R/W	0x0	NDFC_READ_EDO In SDR mode: 00: Normal 01: EDO 10: E-EDO Others: Reserved In DDR mode: 1~15 is valid.(These bits configure the number of clock when data is valid after RE#'s falling edge)
7:6	/	/	/
5:0	R/W	0x0	NDFC_READ_DELAY_CHAIN NDFC Delay Chain Control. (These bits are only valid in DDR data interface, and configure the relative phase between DQS and DQ[0…7])

5.2.6.5 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
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 RE# high to WE# low cycle number 00: 4*2T 01: 12*2T 10: 20*2T 11: 28*2T
5:4	R/W	0x1	T_WHR WE# high to RE# low cycle number 00: 0*2T 01: 6*2T 10: 14*2T 11: 22*2T

3:2	R/W	0x1	T_ADL Address to Data Loading cycle number 00: 0*2T 01: 6*2T 10: 14*2T 11: 22*2T
1:0	R/W	0x1	T_WB WE# high to busy cycle number 00:14*2T 01: 22*2T 10: 30*2T 11: 38*2T

5.2.6.6 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_DATA_4 NAND Flash 4th Cycle Address Data
23:16	R/W	0x0	ADDR_DATA_3 NAND Flash 3rd Cycle Address Data
15:8	R/W	0x0	ADDR_DATA_2 NAND Flash 2nd Cycle Address Data
7:0	R/W	0x0	ADDR_DATA_1 NAND Flash 1st Cycle Address Data

5.2.6.7 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_DATA_8 NAND Flash 8th Cycle Address Data
23:16	R/W	0x0	ADDR_DATA_7 NAND Flash 7th Cycle Address Data
15:8	R/W	0x0	ADDR_DATA_6 NAND Flash 6th Cycle Address Data
7:0	R/W	0x0	ADDR_DATA_5 NAND Flash 5th Cycle Address Data

5.2.6.8 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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, it is no valid.
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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, it is no valid.
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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, it is no valid.
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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, it is no valid.
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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, it is no valid.
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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, it is no valid.
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: disable

			1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, it is no valid.
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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, it is no valid.
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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, it is no valid.
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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, it is no valid.
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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, it is no valid.
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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, it is no valid.
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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, it is no valid.
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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, it is no valid.
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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, it is no valid.
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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, it is no valid.
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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, 1 Data_Block = 2048 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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, 1 Data_Block = 2048 bytes main field data.
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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, 1 Data_Block = 2048 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: disable 1: enable

			When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, 1 Data_Block = 2048 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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, 1 Data_Block = 2048 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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, 1 Data_Block = 2048 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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, 1 Data_Block = 2048 bytes main field data.
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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, 1 Data_Block = 2048 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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, 1 Data_Block = 2048 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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, 1 Data_Block = 2048 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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, 1 Data_Block = 2048 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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, 1 Data_Block = 2048 bytes main field data.
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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, 1 Data_Block = 2048 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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, 1 Data_Block = 2048 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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, 1 Data_Block = 2048 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: disable 1: enable When BCH is in used, 1 Data block = 1024 bytes main field data. When LDPC is in used, 1 Data_Block = 2048 bytes main field data.

5.2.6.9 NDFC Data Counter Register(Default Value: 0x0000_0000)

Offset: 0x0020			**Register Name: NDFC_CNT**
Bit	**Read/Write**	**Default/Hex**	**Description**

31:11	/	/	/
10:0	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. This bit is used to set data length for one DATA_BLOCK_SIZE transfer in NDFC read or write operation. **NOTE** **In NDFC Channel 0, Normal CMD, NDFC_DATA_CNT≤1024 bytes. Batch CMD, NDFC_DATA_CNT=1024 bytes** **In NDFC Channel 1, Normal CMD，NDFC_DATA_CNT≤2048 bytes. Batch CMD，NDFC_DATA_CNT=2048 bytes**

5.2.6.10 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: Normal Common Command for normal operation 01: Special Command for Flash Spare Field Operation 10: Batch Command for batch process operation 11: Reserved **NOTE** In generally, Normal Command is a single command operations. Normal Command can be used alone. Special Command encodes the data of write operation, or decodes the data of read operation. This procedure needs to open ECC engine, and needs to combine with Normal Command. Batch Command indicates that NDFC use both Normal Command and Special Command. it is for NAND flash Page writing operation and need to combine with DMA. Batch Command can be used alone.
29	R/W	0x0	NDFC_SEND_FOURTH_CMD 0: Don’t send third set command 1: Send it on the external memory’s bus It is used for EF-NAND page read.
28	R/W	0x0	NDFC_SEND_THIRD_CMD 0: Don’t send third set command 1: Send it on the external memory’s bus It is used for EF-NAND page read.
27	R/W	0x0	NDFC_SEND_RANDOM_CMD2_CTL 0: Don’t send random cmd2 (NDFC_RANDOM_CMD2) 1: Send random cmd2

			NOTE **It is only valid in batch cmd operation and writing operation.** **Enable whether to send the second randomize order.** **For Batch Command writing operation, Generally it is not enabled. For most of the NAND Flash writing operation, the first random command is 0x85. But for individual NAND Flash, such as TLC of Toshiba, random wrting operating also acquires the second random command. So it needs to set whether to send the other command, and the value of this command is set by bit[31:24] of 0x28 register.** **For Batch Command reading operation, Generally it is enabled. For most of the NAND Flash reading operation, the first random command is 0x05, and then need to send the second random command 0xE0, the value of this command is set by bit[23:16] of 0x28 register.**
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 setup DRQ to fetching data before output to Flash or NDFC should setup DRQ to sending out 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. This bit is mainly used DMA transfer the data between NDFC two RAM(0x400 and 0x800) and DRAM(The direction is determined by reading or writing operation). When this bit is enabled, it will transfer the main area data from RAM into DRAM for reading operation(For writing operation, it will transfer the main area data from DRAM into RAM).
25	R/W	0x0	NDFC_SEQ User data & BCH check word position. It only is active for Page Command, don't care about this bit for other two commands 0: Interleave Method (on page spare area) 1: Sequence Method (following data block)
24	R/W	0x0	NDFC_SEND_SECOND_CMD_EN 0: Don't send second set command 1: Send it on the external memory's bus It is set whether to send the second command. In generally, reading or writing one page need to send the second cmd. The cmd value is determined by 0x28 and 0x2C register. For writing operation, the second command is set by bit[7:0] of 0x2C register.
23	R/W	0x0	NDFC_WAIT_RB_EN 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 can not when the internal NDFC_RB wire is BUSY.

			This bit is used to set controller whether to need wait RB#. For some operation, it need to wait RB#.
22	R/W	0x0	NDFC_SEND_FIRST_CMD_EN 0: Don't send first set command 1: Send it on the external memory's bus This bit is set whether to send the first command, which is set by bit [7:0]. 1. For Normal CMD or Batch CMD, this bit must enable. 2. For Special CMD, this bit is set according to the requirement. For example, send the Special cmd for 5, or send the Special cmd for 0xE0.
21	R/W	0x0	NDFC_DATA_TRANS_EN 0: No data transfer on external memory bus 1: Data transfer and direction is decided by the field NDFC_ACCESS_DIR This bit is used to set whether to carry on data interchange. The direction of data interchange is determined by bit[20] of 0x24 register.
20	R/W	0x0	NDFC_ACCESS_DIR 0: Read NAND Flash 1: Write NAND Flash This bit is used to set the direction of data interchange.
19	R/W	0x0	NDFC_SEND_ADDR 0: Don't send address 1: Send N cycles address, the number N is specified by NDFC_ADR_NUM field. This bit is used to set whether to send address, the number of address is determined by bit[18:16] of 0x24 register. The contents of address is set by 0x14 and 0x18 register. 0x14 register set the low 32-bit address, and 0x18 register set the high 32-bit address.
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 When setting the hop address, this bit is used to set the number of address which need to send mainly for Interleave mode. In generally, 16KB Page_Size can be represented by 2 bytes. But for some special NAND Flash, such as TLC of Toshiba, when it uses the hop address, it also need to send an additional 3 bytes Row Address. So this bit need to set to '11'.

7:0	R/W	0x0	NDFC_CMD_LOW_BYTE NDFC Command low byte data This command will be sent to external Flash by NDFC.

5.2.6.11 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_WRITE_CMD2 Used for the second random command for Batch write Operation
23:16	R/W	0xE0	NDFC_RANDOM_READ_CMD2 Used for the second random command for Batch Read Operation
15:8	R/W	0x05	NDFC_RANDOM_READ_CMD1 Used for the first random command for Batch Read Operation
7:0	R/W	0x30	NDFC_READ_CMD2 Used for the second command for Batch Read Operation

5.2.6.12 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_EFNAND_READ_CMD1 Used for the first command for EF-NAND Page Read operation
23:16	R/W	0x00	NDFC_EFNAND_READ_CMD2 Used for the second command for EF-NAND Page Read operation.
15:8	R/W	0x85	NDFC_RANDOM_WRITE_CMD1 Used for the first random command for Batch Write Operation
7:0	R/W	0x10	NDFC_PROGRAM_CMD2 Used for the second command for Batch Write Operation

5.2.6.13 NDFC ECC Control Register(Default Value: 0x4a80_0008)

Offset: 0x0034			Register Name: NDFC_ECC_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31	/	/	/
30:16	R/W	0x4a80	NDFC_RANDOM_SEED_CH0 The seed value for randomize engine. It is only active when NDFC_RANDOM_EN is set to '1'. This bit is only valid when NDFC Channel0 is selected.
15:8	R/W	0x0	NDFC_ECC_BCH_MODE_CH0 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_CH0 0: ECC block size 1: Page size This bit is used to set the length of each random operation, and only valid when NDFC Channel0 is selected.
6	R/W	0x0	NDFC_RANDOM_DIRECTION 0: LSB first 1: MSB first This bit is used to set the direction of random operation, and only valid when NDFC Channel0 is selected.
5	R/W	0x0	NDFC_RANDOM_BYPASS_MODE 0: Data Randomize Bypass 1: Data Randomize No Bypass
4	R/W	0x0	NDFC_ECC_EXCEPTION_CH0 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. For this case, no error information is reported. It is only active when ECC is ON. It is only valid when NDFC Channel0 is selected. Under the ECC_EXCEPTION is enabled, when the reading data is '0x00' or '0xFF', the special Data Block is relected in the NDFC_ECC_STA register(0x38). The flag of Specific read data(all 0x00 or all 0xff) will reflect in the NDFC_PAT_ID register(0x118).
3	R/W	0x1	NDFC_ECC_PIPELINE_CH0 Pipeline function enable or disable for batch command 0: Error Correction function no pipeline with next block operation 1: Error Correction pipeline This bit indicates that NDFC internal 2 RAM whether to carry on the ECC check contimuously and alternately. In order to improve the efficiency , this function is always enabled.
2:1	/	/	/
0	R/W	0x0	NDFC_ECC_BYPASS_MODE 0: ECC Bypass

			1: ECC No Bypass

5.2.6.14 NDFC ECC Status Register(Default Value: 0x0000_0000)

Offset: 0x0038			Register Name: NDFC_ECC_STA
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 can't correct them The bit[31] of this register is corresponding the 31th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / INVALID (LDPC Mode).
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 can't correct them The bit[30] of this register is corresponding the 30th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / INVALID (LDPC Mode).
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 can't correct them The bit[29] of this register is corresponding the 29th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / INVALID (LDPC Mode).
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 can't correct them The bit[28] of this register is corresponding the 28th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / INVALID (LDPC Mode).
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 can't correct them The bit[27] of this register is corresponding the 27th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / INVALID (LDPC Mode).
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 can't correct them The bit[26] of this register is corresponding the 26th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / INVALID (LDPC Mode).
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 can't correct them The bit[25] of this register is corresponding the 25th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / INVALID (LDPC Mode).
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 can't correct them The bit[24] of this register is corresponding the 24th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / INVALID (LDPC Mode).
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 can't correct them The bit[23] of this register is corresponding the 23th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / INVALID (LDPC Mode).
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 can't correct them The bit[22] of this register is corresponding the 22th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / INVALID (LDPC Mode).
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 can't correct them The bit[21] of this register is corresponding the 21th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / INVALID (LDPC Mode).
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 can't correct them The bit[20] of this register is corresponding the 20th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / INVALID (LDPC Mode).
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 can't correct them The bit[19] of this register is corresponding the 19th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / INVALID (LDPC Mode).
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 can't correct them The bit[18] of this register is corresponding the 18th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / INVALID (LDPC Mode).
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 can't correct them The bit[17] of this register is corresponding the 17th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / INVALID (LDPC Mode).
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 can't correct them The bit[16] of this register is corresponding the 16th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / INVALID (LDPC Mode).
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 can't correct them The bit[15] of this register is corresponding the 15th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / 2048 bytes (LDPC Mode).
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 can't correct them The bit[14] of this register is corresponding the 14th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / 2048 bytes (LDPC Mode).
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 can't correct them The bit[13] of this register is corresponding the 13th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / 2048 bytes (LDPC Mode).
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 can't correct them The bit[12] of this register is corresponding the 12th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / 2048 bytes (LDPC Mode).
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 can't correct them The bit[11] of this register is corresponding the 11th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / 2048 bytes (LDPC Mode).
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 can't correct them The bit[10] of this register is corresponding the 10th ECC data block. 1 Data_Block

			= 1024 bytes (BCH Mode) / 2048 bytes (LDPC Mode).
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 can't correct them The bit[9] of this register is corresponding the 9th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / 2048 bytes (LDPC Mode).
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 can't correct them The bit[8] of this register is corresponding the 8th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / 2048 bytes (LDPC Mode).
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 can't correct them The bit[7] of this register is corresponding the 7th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / 2048 bytes (LDPC Mode).
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 can't correct them The bit[6] of this register is corresponding the 6th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / 2048 bytes (LDPC Mode).
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 can't correct them The bit[5] of this register is corresponding the 5th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / 2048 bytes (LDPC Mode).
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 can't correct them The bit[4] of this register is corresponding the 4th ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / 2048 bytes (LDPC Mode).
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 can't correct them The bit[3] of this register is corresponding the 3rd ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / 2048 bytes (LDPC Mode).
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 can't correct them The bit[2] of this register is corresponding the 2nd ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / 2048 bytes (LDPC Mode).
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 can't correct them The bit[1] of this register is corresponding the 1st ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / 2048 bytes (LDPC Mode).
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 can't correct them The bit[0] of this register is corresponding the 0 ECC data block. 1 Data_Block = 1024 bytes (BCH Mode) / 2048 bytes (LDPC Mode).

5.2.6.15 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.
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.
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.
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.
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.
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.
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.

5.2.6.16 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. This bit is used to enable wether to fill the redundant data. This function is only valid for Batch Command type(NDFC_CMD_TYPE = 0x10 in 0x24 register).
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, controller would send dummy byte to fill the unused space in one page. **NOTE** **It is only valid in PAGE CMD operation(*NDFC_CMD_TYPE*=0x3).** **This function is disabled when *Dummy_Byte_Count_EN*=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*** = 0x1. Used for setting the amount of redundant data to be filled. This function is valid when NDFC_RANDOM_EN = 0x1, Otherwise, the filled data is 0.
15:9	/	/	/
8	R/W	0x0	NDFC_WP_CTL NAND Flash Write Protect Control Bit 0: Write Protect is active 1: Write Protect is not active When this bit is '0', WP signal line is low level and external NAND flash is on protected state. In generally, Write Protect function is not enable, so that this bit is set to 1.
7	/	/	/
6:0	R/W	0x0	NDFC_ECC_DEBUG For the purpose of debugging ECC engine, special error bits are inserted before writing external Flash Memory. 0: No error is inserted (ECC Normal Operation) n: N bits error are inserted NDFC Channel 0 BCH channel interpolation position starts from the highest bit in data byte. NDFC Channel 1 LDPC channel interpolation starts bit-flipping from bit[7] of each byte.

5.2.6.17 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 current read operation; 1: Enable to count the number of bit 1 and bit 0 during current read operation; The number of bit 1 and bit 0 during current read operation can be used to check whether a page is blank or bad. This bit is used to enable whether to count the number of '0' and '1' in the data of current read operation.
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 current read operation is less than or equal to threshold value, the bit [13] of NDFC_ST register will be set. If the number of bit 0 during current read operation is less than or equal to threshold value, the bit [12] of NDFC_ST register will be set.

5.2.6.18 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 current command. It will be cleared automatically when next command is executed. This function is only valid when NDFC_RDATA_STA_EN = 1.

5.2.6.19 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 input bit 0 during current command. It will be cleared automatically when next command is executed. This function is only valid when NDFC_RDATA_STA_EN = 1.

5.2.6.20 NDFC Error Counter Register N(Default Value: 0x0000_0000)

NDFC Channel 0 uses 8-bit to indicate the number of one Data Block error correction.			
Offset: 0x0050+N*0x04(N=0~7)			Register Name: NDFC_ERR_CNT_N
Bit	**Read/Write**	**Default/Hex**	**Description**

31:24	R	0x00	ECC_COR_NUM_DATA_BLOCK_4X_3 ECC Corrected Bits Number for ECC Data Block[N*0x04+3] 00000000: No corrected bits 00000001: 1 corrected bit 00000010: 2 corrected bits ..... 01010000 : 80 corrected bits Others: Reserved 1 ECC Data Block =1024 bytes When selecting NDFC Channel 0 to correct error: the offset address is from 0x50 to 0x6C, the sum of register is 8, each 8bits represents one Data_Block ECC error correction number.
23:16	R	0x00	ECC_COR_NUM_DATA_BLOCK_4X_2 ECC Corrected Bits Number for ECC Data Block[X*4+2] 00000000: No corrected bit 00000001: 1 corrected bit 00000010: 2 corrected bits ..... 01010000 : 80 corrected bits Others: Reserved 1 ECC Data Block =1024 bytes When selecting NDFC Channel 0 to correct error: the offset address is from 0x50 to 0x6C, the sum of register is 8, each 8bits represents one Data_Block ECC error correction number.
15:8	R	0x00	ECC_COR_NUM_DATA_BLOCK_4X_1 ECC Corrected Bits Number for ECC Data Block[X*4+1] 00000000: No corrected bit 00000001: 1 corrected bit 00000010: 2 corrected bits ..... 01010000 : 80 corrected bits Others: Reserved 1 ECC Data Block =1024 bytes When selecting NDFC Channel 0 to correct error: the offset address is from 0x50 to 0x6C, the sum of register is 8, each 8bits represents one Data_Block ECC error correction number.
7:0	R	0x00	ECC_COR_NUM_DATA_BLOCK_4X_0 ECC Corrected Bits Number for ECC Data Block[X*4+0] 00000000: No corrected bit 00000001: 1 corrected bit 00000010: 2 corrected bits ..... 01010000 : 80 corrected bits Others: Reserved 1 ECC Data Block =1024 bytes When selecting NDFC Channel 0 to correct error: the offset address is from 0x50

			to 0x6C, the sum of register is 8, each 8bits represents one Data_Block ECC error correction number.
NDFC Channel 1 uses 16-bit to indicate the number of one Data Block error correction.			
Offset: 0x0050+ N*0x04 (N=0~7)			**Register Name: NDFC_ERR_CNT_N**
Bit	**Read/Write**	**Bit**	**Description**
31:16	R	0x00	ECC_COR_NUM_DATA_BLOCK_2X_1 ECC Corrected Bits Number for ECC Data Block[X*4+1] 0000000000000000: No corrected bits 0000000000000001: 1 corrected bit 0000000000000010: 2 corrected bits ..... 0000000111111111: 511 corrected bits Others: Reserved 1 ECC Data Block =2048 bytes When selecting NDFC Channel 1 to correct error: the offset address is from 0x50 to 0x6C, the sum of register is 8, each 16bits represents one Data_Block ECC error correction number.
15:0	R	0x00	ECC_COR_NUM_DATA_BLOCK_2X_0 ECC Corrected Bits Number for ECC Data Block[X*4+0] 0000000000000000: No corrected bits 0000000000000001: 1 corrected bit 0000000000000010: 2 corrected bits ..... 0000000111111111: 511 corrected bits Others: Reserved 1 ECC Data Block =2048 bytes When selecting NDFC Channel 1 to correct error: the offset address is from 0x50 to 0x6C, the sum of register is 8, each 16bits represents one Data_Block ECC error correction number.

5.2.6.21 NDFC User Data Length Register N(Default Value: 0x0000_0000)

Offset: 0x0070 + N*0x04(N=0~3)			**Register Name: NDFC_USER_DATA_LEN_N**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x0	USER_DATA_LEN_DATA_BLOCK_8X_7 0000 : no user data 0001 : 04bytes user data 0010 : 08bytes user data 0011 : 12bytes user data 0100 : 16bytes user data 0101 : 20bytes user data 0110 : 24bytes user data 0111 : 28bytes user data 1000 : 32bytes user data Other : reserved

			It's used to indicate user data's length of ECC DATA BLOCK[0x08*N+3].
27:24	R/W	0x0	USER_DATA_LEN_DATA_BLOCK_8X_6 0000 : no user data 0001 : 04bytes user data 0010 : 08bytes user data 0011 : 12bytes user data 0100 : 16bytes user data 0101 : 20bytes user data 0110 : 24bytes user data 0111 : 28bytes user data 1000 : 32bytes user data Other : reserved It's used to indicate user data's length of ECC DATA BLOCK[0x08*N+2].
23:20	R/W	0x0	USER_DATA_LEN_DATA_BLOCK_8X_5 0000 : no user data 0001 : 04bytes user data 0010 : 08bytes user data 0011 : 12bytes user data 0100 : 16bytes user data 0101 : 20bytes user data 0110 : 24bytes user data 0111 : 28bytes user data 1000 : 32bytes user data Other : reserved It's used to indicate user data's length of ECC DATA BLOCK[0x08*N+1].
19:16	R/W	0x0	USER_DATA_LEN_DATA_BLOCK_8X_4 0000 : no user data 0001 : 04bytes user data 0010 : 08bytes user data 0011 : 12bytes user data 0100 : 16bytes user data 0101 : 20bytes user data 0110 : 24bytes user data 0111 : 28bytes user data 1000 : 32bytes user data Other : reserved It's used to indicate user data's length of ECC DATA BLOCK[0x08*N+0].
15:12	R/W	0x0	USER_DATA_LEN_DATA_BLOCK_8X_3 0000 : no user data 0001 : 04bytes user data 0010 : 08bytes user data 0011 : 12bytes user data 0100 : 16bytes user data 0101 : 20bytes user data 0110 : 24bytes user data 0111 : 28bytes user data

			1000 : 32bytes user data Other : reserved It's used to indicate user data's length of ECC DATA BLOCK[0x08*N+0].
11:8	R/W	0x0	USER_DATA_LEN_DATA_BLOCK_8X_2 0000 : no user data 0001 : 04bytes user data 0010 : 08bytes user data 0011 : 12bytes user data 0100 : 16bytes user data 0101 : 20bytes user data 0110 : 24bytes user data 0111 : 28bytes user data 1000 : 32bytes user data Other : reserved It's used to indicate user data's length of ECC DATA BLOCK[0x08*N+0].
7:4	R/W	0x0	USER_DATA_LEN_DATA_BLOCK_8X_1 0000 : no user data 0001 : 04bytes user data 0010 : 08bytes user data 0011 : 12bytes user data 0100 : 16bytes user data 0101 : 20bytes user data 0110 : 24bytes user data 0111 : 28bytes user data 1000 : 32bytes user data Other : reserved It's used to indicate user data's length of ECC DATA BLOCK[0x08*N+0].
3:0	R/W	0x0	USER_DATA_LEN_DATA_BLOCK_8X_0 0000 : no user data 0001 : 04bytes user data 0010 : 08bytes user data 0011 : 12bytes user data 0100 : 16bytes user data 0101 : 20bytes user data 0110 : 24bytes user data 0111 : 28bytes user data 1000 : 32bytes user data Other : reserved It's used to indicate user data's length of ECC DATA BLOCK[0x08*N+0].

5.2.6.22 NDFC User Data Register N(Default Value: 0xFFFF_FFFF)

Offset: 0x0080 + N*0x04(N=0~31)			Register Name: NDFC_USER_DATA_N
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0xffffffff	USER_DATA_VAL_X

			NOTE **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

5.2.6.23 NDFC Flash Status Register (Default Value: 0x0000_0000)

Offset: 0x0100			**Register Name: NDFC_FLASH_STA**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:25	/	/	/
24	R/W	0x0	CMD_REPT_EN 0: Disable 1: Enable
23:16	R/W	0x0	BITMAP_B After Erase or Program is in a pass status, this bit is used to set the value of Erase or Program. And it is used to indicate the flash status: Pass/Fail. 1. Only valid when command repeat function is enabled. 2. In the process of combining the command descriptor operation, If CMD_DESCR_FLASH_STA_HANDLE_EN is enabled, read the value of Flash_STA(bit[7:0]), and AND bitmap_B, if the result is true, the subsequent command descriptor is no longer executed, and the Flash Status of CMD_DESCR_FLASH_STA_HANDLE_STA is enabled. 3. In the process of combining the command descriptor operation, If CMD_DESCR_FLASH_STA_HANDLE_EN is not enabled, read the value of Flash_STA(bit[7:0]), and AND bitmap_B, if the result is true, the Flash Status of CMD_DESCR_FLASH_STA_HANDLE_STA is set to '1' after executing the command descriptor command.
15:8	R/W	0x0	BITMAP_A When command repeat is enabled and reading the Flash status, Judge whether it is ready. After updating the data each time, the data AND bitmap_A with bits. If the result is equal to the value of bitmap_A, the resend command will stop. Only valid when resend command function is enabled.

7:0	R	0x0	SYNC_CMD_REPT_READ_STA Synchronizing resend command read data After resend command is enabled, once the data reading is happened, the first reading data will be synchronizes with the register. When the data reading is not happened, the synchronize value is 0xFF by default. Only valid when resend command function is enabled.

5.2.6.24 NDFC Command Repeat Counter Register (Default Value: 0x0000_0000)

Offset: 0x0104			Register Name: NDFC_CMD_REPT_CNT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x0	CMD_REPT_CNT The max time of command repeat.

5.2.6.25 NDFC Command Repeat Interval Register (Default Value: 0x0000_0000)

Offset: 0x0108			Register Name: NDFC_CMD_REPT_INVL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x0	CMD_REPT_INVL The time interval of command repeat.

5.2.6.26 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

5.2.6.27 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_ADDR This value indicates the spare area first byte address for NDFC interleave page operation. Use this Spare Address for Batch Command Interleave.

5.2.6.28 NDFC Pattern ID Register(Default Value: 0x0000_0000)

Offset: 0x0118			Register Name: NDFC_PAT_ID
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	PAT_ID Special Pattern ID for ECC data block[n] 0: All 0x00 is found 1: All 0xFF is found This bit indicates that special data values for each data block. Valid when ECC_Exception function of NDFC_ECC_CTL register is enabled.

5.2.6.29 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	/	/	/
15:12	R/W	0x0	DLEN_WR The number of latency DQS cycle for write. 0000: No latency 0001: One latency DQS cycle 0010: Two latency DQS cycle 0011: Four latency DQS cycle This bit is used to set the delay cycle of DQS signal in writing operation for DDR2 mode.
11:8	R/W	0x0	DLEN_RD The number of latency DQS cycle for read. 0000: No latency 0001: One latency DQS cycle 0010: Two latency DQS cycle 0011: Four latency DQS cycle This bit is used to set the delay cycle of DQS signal in reading operation for DDR2 mode.
7:3	/	/	/
2	R/W	0x0	EN_RE_C Enable the complementary RE# signal. 0: Disable 1: Enable
1	R/W	0x0	EN_DQS_C Enable the complementary DQS signal. 0: Disable 1: Enable
0	/	/	/

5.2.6.30 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 This bit is used to select the dma_active status
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 delay cycles The counts of hold cycles from DMA last signal high to dma_active high.

5.2.6.31 NDFC Valid Data DMA Counter Register (Default Value: 0x0000_0000)

Offset: 0x012C			Register Name: NDFC_VALID_DATA_DMA_CNT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:5	/	/	/
4:0	R/W	0x0	VALID_DATA_DMA_CNT The number of valid Data DMA: 0~16.

5.2.6.32 NDFC Data DMA Address N Register (Default Value: 0x0000_0000)

Offset: 0x0130 + 0x04*N(N=0~15)			Register Name: NDFC_DATA_DMA_ADDR_N
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	DATA_DMA_ADDR_N Transfer the Data_0~Data_15 DRAM Address by DMA, 32-byte aligned

5.2.6.33 NDFC Data DMA Size 2N and Data DMA Size 2N+1 Register (Default Value: 0x0000_0000)

Offset: 0x0170 + 0x04*N (N=0~7)			Register Name: NDFC_DATA_DMA_SIZE_2N
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	R/W	0x0	DATA_DMA_SIZE_2N_1 Transfer the number byte of Data_2N+1 byDMA, 8-byte aligned.
15:0	R/W	0x0	DATA_DMA_SIZE_2N Transfer the number byte of Data_2N byDMA, 8-byte aligned.

5.2.6.34 NDFC Random Seed N Register (Default Value: 0x0000_0000)

Offset: 0x0190 + 0x04*N (N=0~15)			Register Name: NDFC_RANDOM_SEED_N
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	RANDOM_SEED_N New mode randomize seeds 0~15

5.2.6.35 NDFC Normal DMA Byte Counter Register(Default Value: 0x0000_0000)

Offset: 0x214			Register Name: NDFC_NDMA_CNT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x0	NDMA_CNT DMA data counter for Normal DMA, only valid for Normal DMA

5.2.6.36 NDFC EMCE Control Register(Default Value: 0x0000_0000)

Offset: 0x218			Register Name: NDFC_EMCE_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1:0	R/W	0x0	NDFC_EMCE_ENABLE_TYPE 00 : Disable encryption/decryption and bypass the EMCE module. 01 : Reserved 10: Disable encryption/decryption but not bypass the EMCE module. 11: Enable encryption/decryption. 1. When NDFC combined with EMCE, the amout of transmission data need 512-byte aligned. 2. When NDFC combined with EMCE and Normal Command is in used, MBUS DMA need to be used. 3. When using NDFC Channel 0, Each TASK length is fixed to 1K bytes. 4. When using NDFC Channel 1, Each TASK length is fixed to 2K bytes.

5.2.6.37 NDFC EMCE IV_FAC Compare Value Register(Default Value: 0x0000_0000)

Offset: 0x21C			Register Name: NDFC_EMCE_IV_FAC_CMP_VAL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	NDFC_EMCE_IV_FAC_CMP_VAL This value is set by user. It is EMCE IV factor compare value.

5.2.6.38 NDFC EMCE IV Calculate Factor Register N(Default Value: 0x0000_0000)

Offset: 0x220+0x04*N(N=0 to 31)			Register Name: NDFC_EMCE_IV_CAL_FACTOR_N
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	NDFC_EMCE_IV_CAL_FACTOR_N This factor is set by user. If the factor is equal to NDFC_EMCE_CMP_VAL, the corresponding sector dose not need encryption. If factor is not equal to NDFC_EMCE_CMP_VAL, the corresponding sector need encryption, and its IV factor specified by this register. **NOTE** **NDFC EMCE does not support 32KB page_size.** NDFC_EMCE_IV_CAL_FACTOR_N is set to the value which is different from NDFC_EMCE_V_FAL_CMP_VAL, Bypass signal is controlled by EMCE_Task itself.

5.2.6.39 NDFC IO Data Register (Default Value: 0x0000_0000)

Offset: 0x02A0			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. When using Normal DMA, this bit indicates data buffer address. When writing operation, Normal DMA transfer the data from DRAM to this RAM, and then the master controller write the temporary data from RAM to NAND Flash; When reading operation, the data that master controller reads from NAND Flash is temporary in the RAM, and then Normal DMA transfer the RAM data to DRAM space.

5.2.6.40 NDFC LDPC Control Register (Default Value: 0xA000_0000)

Offset: 0x02FC			Register Name: NDFC_LDPC_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0xA	CH1_HB_LLR_VAL 0x0~0xF Default Value is 0xA
27:6	/	/	/
5	R/W	0x0	CH1_SOFT_BIT_DEC_EN 0: Disable 1: Enable Enable Soft-Bit Decode
4	R/W	0x0	CH1_SRAM_MAIN_OR_SPARE_AREA_IND 0: Main Area RAM 1: Spare Area RAM

			When Soft-Bit, this bit is used to indicate the location of selection in the SRAM, Main Area or Spare Area(If Normal command operation is used two times, this function will be used, invalid when using Batch Command)
3:1	R/W	0x0	CH1_SRAM_IND When Soft-Bit, this bit is used to select the SRAM of the stored location(SRAM0~SRAM4). Put the 2KB HB information into SRAM1, put the 2KB SB0 information into SRAM2, Put the 2KB SB1 information into SRAM3, Put the 2KB SB2 information into SRAM4, Put the 2KB SB3 information into SRAM0. Soft-Bit Decode and other relevant information are configured before reading 2KB SB3 information.
0	R/W	0x0	CH1_SOFT_BIT_IND 0: Hard-Bit 1: Soft-Bit After this bit is set to '1', the data is stored into SRAM, and the data will not be transfer into the LDPC Decode when using Soft-bit.

5.2.6.41 NDFC Encode LDPC Mode Setting Register (Default Value: 0x8800_0000)

Offset: 0x0300			Register Name: NDFC_ENC_LDPC_MODE_SET
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x1	ENC_MODE 0: Decode 1: Encode This bit is used to choose the LDPC Encoder/Decoder mode.
30:28	R/W	0x0	LDPC_MODE 000: mode 0 001: mode 1 010: mode 2 011: mode 3 100: mode 4 101: mode 5 110: mode 6 This field is used to choose the encode LDPC mode.
27	R/W	0x1	FW_EXTEND 0: FW8B for 2KB LDPC 1: FW16B for 2KB LDPC This bit is used to encode FW_EXTEND mode for encoder.
26:0	/	/	/

5.2.6.42 NDFC Correct LDPC Mode Setting Register (Default Value: 0x0000_8000)

Offset: 0x0304			Register Name: NDFC_COR_LDPC_MODE_SET
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	R/W	0x0	C0_LDPC_MODE

			000: LDPC mode 0 001: LDPC mode 1 010: LDPC mode 2 011: LDPC mode 3 100: LDPC mode 4 101: LDPC mode 5 110: LDPC mode 6 This field is used to choose the correct C0 LDPC mode for Channel 0.
28:26	R/W	0x0	C0_DECODE_MODE 000: mode 0 001: mode 1 010: mode 2 011: mode 3 100: mode 4 This field is used to choose the correct C0 DECODE MODE for Channel 0.
25:24	R/W	0x0	C0_SCALE_MODE 00: scale mode 0 for HB 01: scale mode 1 for SB This field is used to choose the LDPC scale parameter for Channel 0.
23:21	R/W	0x0	C1_LDPC_MODE 000: LDPC mode 0 001: LDPC mode 1 010: LDPC mode 2 011: LDPC mode 3 100: LDPC mode 4 101: LDPC mode 5 110: LDPC mode 6 This field is used to choose the correct C1 LDPC mode for Channel 1.
20:18	R/W	0x0	C1_DECODE_MODE 000: mode 0 001: mode 1 010: mode 2 011: mode 3 100: mode 4 This field is used to choose the correct C1 DECODE MODE for Channel 1.
17:16	R/W	0x0	C1_SCALE_MODE 00: scale mode 0 for HB 01: scale mode 1 for SB This field is used to choose the LDPC scale parameter for Channel 1.
15	R/W	0x1	FW_EXTEND 0: FW8B for 2KB LDPC 1: FW16B for 2KB LDPC This bit is used to decode FW_EXTEND mode for decoder.
14:0	/	/	/

5.2.6.43 NDFC C0 LLR Table 11111-11100 Register (Default Value: 0x0000_0000)

Offset: 0x0308			Register Name: NDFC_C0_LLR_TBL_0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	R/W	0x0	C0_LLR_31 LLR of HBSB0SB1SB2SB3=11111 for LDPC channel 0.
26:24	/	/	/
23:19	R/W	0x0	C0_LLR_30 LLR of HBSB0SB1SB2SB3=11110 for LDPC channel 0.
18:16	/	/	/
15:11	R/W	0x0	C0_LLR_29 LLR of HBSB0SB1SB2SB3=11101 for LDPC channel 0.
10:8	/	/	/
7:3	R/W	0x0	C0_LLR_28 LLR of HBSB0SB1SB2SB3=11100 for LDPC channel 0.
2:0	/	/	/

5.2.6.44 NDFC C0 LLR Table 11011-11000 Register (Default Value: 0x0000_0000)

Offset: 0x030C			Register Name: NDFC_C0_LLR_TBL_1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	R/W	0x0	C0_LLR_27 LLR of HBSB0SB1SB2SB3=11011 for LDPC channel 0.
26:24	/	/	/
23:19	R/W	0x0	C0_LLR_26 LLR of HBSB0SB1SB2SB3=11010 for LDPC channel 0.
18:16	/	/	/
15:11	R/W	0x0	C0_LLR_25 LLR of HBSB0SB1SB2SB3=11001 for LDPC channel 0.
10:8	/	/	/
7:3	R/W	0x0	C0_LLR_24 LLR of HBSB0SB1SB2SB3=11000 for LDPC channel 0.
2:0	/	/	/

5.2.6.45 NDFC C0 LLR Table 10111-10100 Register (Default Value: 0x0000_0000)

Offset: 0x0310			Register Name: NDFC_C0_LLR_TBL_2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	R/W	0x0	C0_LLR_23 LLR of HBSB0SB1SB2SB3=10111 for LDPC channel 0.
26:24	/	/	/
23:19	R/W	0x0	C0_LLR_22 LLR of HBSB0SB1SB2SB3=10110 for LDPC channel 0.
18:16	/	/	/

15:11	R/W	0x0	C0_LLR_21 LLR of HBSB0SB1SB2SB3=10101 for LDPC channel 0.
10:8	/	/	/
7:3	R/W	0x0	C0_LLR_20 LLR of HBSB0SB1SB2SB3=10100 for LDPC channel 0.
2:0	/	/	/

5.2.6.46 NDFC C0 LLR Table 10011-10000 Register (Default Value: 0x0000_0000)

Offset: 0x0314			Register Name: NDFC_C0_LLR_TBL_3
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	R/W	0x0	C0_LLR_19 LLR of HBSB0SB1SB2SB3=10011 for LDPC channel 0.
26:24	/	/	/
23:19	R/W	0x0	C0_LLR_18 LLR of HBSB0SB1SB2SB3=10010 for LDPC channel 0.
18:16	/	/	/
15:11	R/W	0x0	C0_LLR_17 LLR of HBSB0SB1SB2SB3=10001 for LDPC channel 0.
10:8	/	/	/
7:3	R/W	0x0	C0_LLR_16 LLR of HBSB0SB1SB2SB3=10000 for LDPC channel 0.
2:0	/	/	/

5.2.6.47 NDFC C0 LLR Table 01111-01100 Register (Default Value: 0x0000_0000)

Offset: 0x0318			Register Name: NDFC_C0_LLR_TBL_4
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	R/W	0x0	C0_LLR_15 LLR of HBSB0SB1SB2SB3=01111 for LDPC channel 0.
26:24	/	/	/
23:19	R/W	0x0	C0_LLR_14 LLR of HBSB0SB1SB2SB3=01110 for LDPC channel 0.
18:16	/	/	/
15:11	R/W	0x0	C0_LLR_13 LLR of HBSB0SB1SB2SB3=01101 for LDPC channel 0.
10:8	/	/	/
7:3	R/W	0x0	C0_LLR_12 LLR of HBSB0SB1SB2SB3=01100 for LDPC channel 0.
2:0	/	/	/

5.2.6.48 NDFC C0 LLR Table 01011-01000 Register (Default Value: 0x0000_0000)

Offset: 0x031C			Register Name: NDFC_C0_LLR_TBL_5
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	R/W	0x0	C0_LLR_11 LLR of HBSB0SB1SB2SB3=01011 for LDPC channel 0.
26:24	/	/	/
23:19	R/W	0x0	C0_LLR_10 LLR of HBSB0SB1SB2SB3=01010 for LDPC channel 0.
18:16	/	/	/
15:11	R/W	0x0	C0_LLR_09 LLR of HBSB0SB1SB2SB3=01001 for LDPC channel 0.
10:8	/	/	/
7:3	R/W	0x0	C0_LLR_08 LLR of HBSB0SB1SB2SB3=01000 for LDPC channel 0.
2:0	/	/	/

5.2.6.49 NDFC C0 LLR Table 00111-00100 Register (Default Value: 0x0000_0000)

Offset: 0x0320			Register Name: NDFC_C0_LLR_TBL_6
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	R/W	0x0	C0_LLR_07 LLR of HBSB0SB1SB2SB3=00111 for LDPC channel 0.
26:24	/	/	/
23:19	R/W	0x0	C0_LLR_06 LLR of HBSB0SB1SB2SB3=00110 for LDPC channel 0.
18:16	/	/	/
15:11	R/W	0x0	C0_LLR_05 LLR of HBSB0SB1SB2SB3=00101 for LDPC channel 0.
10:8	/	/	/
7:3	R/W	0x0	C0_LLR_04 LLR of HBSB0SB1SB2SB3=00100 for LDPC channel 0.
2:0	/	/	/

5.2.6.50 NDFC C0 LLR Table 00011-00000 Register (Default Value: 0x0000_0000)

Offset: 0x0324			Register Name: NDFC_C0_LLR_TBL_7
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	R/W	0x0	C0_LLR_03 LLR of HBSB0SB1SB2SB3=00011 for LDPC channel 0.
26:24	/	/	/
23:19	R/W	0x0	C0_LLR_02 LLR of HBSB0SB1SB2SB3=00010 for LDPC channel 0.
18:16	/	/	/

15:11	R/W	0x0	C0_LLR_01 LLR of HBSB0SB1SB2SB3=00001 for LDPC channel 0.
10:8	/	/	/
7:3	R/W	0x0	C0_LLR_00 LLR of HBSB0SB1SB2SB3=00000 for LDPC channel 0.
2:0	/	/	/

5.2.6.51 NDFC C1 LLR Table 11111-11100 Register (Default Value: 0x0000_0000)

Offset: 0x0328			Register Name: NDFC_C1_LLR_TBL_0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	R/W	0x0	C1_LLR_31 LLR of HBSB0SB1SB2SB3=11111 for LDPC channel 1.
26:24	/	/	/
23:19	R/W	0x0	C1_LLR_30 LLR of HBSB0SB1SB2SB3=11110 for LDPC channel 1.
18:16	/	/	/
15:11	R/W	0x0	C1_LLR_29 LLR of HBSB0SB1SB2SB3=11101 for LDPC channel 1.
10:8	/	/	/
7:3	R/W	0x0	C1_LLR_28 LLR of HBSB0SB1SB2SB3=11100 for LDPC channel 1.
2:0	/	/	/

5.2.6.52 NDFC C1 LLR Table 11011-11000 Register (Default Value: 0x0000_0000)

Offset: 0x032C			Register Name: NDFC_C1_LLR_TBL_1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	R/W	0x0	C1_LLR_27 LLR of HBSB0SB1SB2SB3=11011 for LDPC channel 1.
26:24	/	/	/
23:19	R/W	0x0	C1_LLR_26 LLR of HBSB0SB1SB2SB3=11010 for LDPC channel 1.
18:16	/	/	/
15:11	R/W	0x0	C1_LLR_25 LLR of HBSB0SB1SB2SB3=11001 for LDPC channel 1.
10:8	/	/	/
7:3	R/W	0x0	C1_LLR_24 LLR of HBSB0SB1SB2SB3=11000 for LDPC channel 1.
2:0	/	/	/

5.2.6.53 NDFC C1 LLR Table 10111-10100 Register (Default Value: 0x0000_0000)

Offset: 0x0330			Register Name: NDFC_C1_LLR_TBL_2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	R/W	0x0	C1_LLR_23 LLR of HBSB0SB1SB2SB3=10111 for LDPC channel 1.
26:24	/	/	/
23:19	R/W	0x0	C1_LLR_22 LLR of HBSB0SB1SB2SB3=10110 for LDPC channel 1.
18:16	/	/	/
15:11	R/W	0x0	C1_LLR_21 LLR of HBSB0SB1SB2SB3=10101 for LDPC channel 1.
10:8	/	/	/
7:3	R/W	0x0	C1_LLR_20 LLR of HBSB0SB1SB2SB3=10100 for LDPC channel 1.
2:0	/	/	/

5.2.6.54 NDFC C1 LLR Table 10011-10000 Register (Default Value: 0x0000_0000)

Offset: 0x0334			Register Name: NDFC_C1_LLR_TBL_3
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	R/W	0x0	C1_LLR_19 LLR of HBSB0SB1SB2SB3=10011 for LDPC channel 1.
26:24	/	/	/
23:19	R/W	0x0	C1_LLR_18 LLR of HBSB0SB1SB2SB3=10010 for LDPC channel 1.
18:16	/	/	/
15:11	R/W	0x0	C1_LLR_17 LLR of HBSB0SB1SB2SB3=10001 for LDPC channel 1.
10:8	/	/	/
7:3	R/W	0x0	C1_LLR_16 LLR of HBSB0SB1SB2SB3=10000 for LDPC channel 1.
2:0	/	/	/

5.2.6.55 NDFC C1 LLR Table 01111-01100 Register (Default Value: 0x0000_0000)

Offset: 0x0338			Register Name: NDFC_C1_LLR_TBL_4
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	R/W	0x0	C1_LLR_15 LLR of HBSB0SB1SB2SB3=01111 for LDPC channel 1.
26:24	/	/	/
23:19	R/W	0x0	C1_LLR_14 LLR of HBSB0SB1SB2SB3=01110 for LDPC channel 1.
18:16	/	/	/

15:11	R/W	0x0	C1_LLR_13 LLR of HBSB0SB1SB2SB3=01101 for LDPC channel 1.
10:8	/	/	/
7:3	R/W	0x0	C1_LLR_12 LLR of HBSB0SB1SB2SB3=01100 for LDPC channel 1.
2:0	/	/	/

5.2.6.56 NDFC C1 LLR Table 01011-01000 Register (Default Value: 0x0000_0000)

Offset: 0x033C			Register Name: NDFC_C1_LLR_TBL_5
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	R/W	0x0	C1_LLR_11 LLR of HBSB0SB1SB2SB3=01011 for LDPC channel 1.
26:24	/	/	/
23:19	R/W	0x0	C1_LLR_10 LLR of HBSB0SB1SB2SB3=01010 for LDPC channel 1.
18:16	/	/	/
15:11	R/W	0x0	C1_LLR_09 LLR of HBSB0SB1SB2SB3=01001 for LDPC channel 1.
10:8	/	/	/
7:3	R/W	0x0	C1_LLR_08 LLR of HBSB0SB1SB2SB3=01000 for LDPC channel 1.
2:0	/	/	/

5.2.6.57 NDFC C1 LLR Table 00111-00100 Register (Default Value: 0x0000_0000)

Offset: 0x0340			Register Name: NDFC_C1_LLR_TBL_6
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	R/W	0x0	C1_LLR_07 LLR of HBSB0SB1SB2SB3=00111 for LDPC channel 1.
26:24	/	/	/
23:19	R/W	0x0	C1_LLR_06 LLR of HBSB0SB1SB2SB3=00110 for LDPC channel 1.
18:16	/	/	/
15:11	R/W	0x0	C1_LLR_05 LLR of HBSB0SB1SB2SB3=00101 for LDPC channel 1.
10:8	/	/	/
7:3	R/W	0x0	C1_LLR_04 LLR of HBSB0SB1SB2SB3=00100 for LDPC channel 1.
2:0	/	/	/

5.2.6.58 NDFC C1 LLR Table 00011-00000 Register (Default Value: 0x0000_0000)

Offset: 0x0344			Register Name: NDFC_C1_LLR_TBL_7
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	R/W	0x0	C1_LLR_03 LLR of HBSB0SB1SB2SB3=00011 for LDPC channel 1.
26:24	/	/	/
23:19	R/W	0x0	C1_LLR_02 LLR of HBSB0SB1SB2SB3=00010 for LDPC channel 1.
18:16	/	/	/
15:11	R/W	0x0	C1_LLR_01 LLR of HBSB0SB1SB2SB3=00001 for LDPC channel 1.
10:8	/	/	/
7:3	R/W	0x0	C1_LLR_00 LLR of HBSB0SB1SB2SB3=00000 for LDPC channel 1.
2:0	/	/	/

5.2.6.59 NDFC Global Configure Register (Default Value: 0x0000_0100)

Offset: 0x0C00			Register Name: NDFC_GLB_CFG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	R	0x0	CMD_DESCR_FINISH_NUM indicates the number of command descriptor that have been completed.
15:9	/	/	/
8	R/W	0x1	LDPC_OUTPUT_DISORDER_CTL 0: Open Disorder 1: Close Disorder Shut down the LDPC output disorder function by default. When Normal DMA mode is in used , the disorder function need to be closed.
7	R/W	0x0	CMD_DESCR_ECC_ERROR_HANDLE_EN 0: Diable 1: Enable After this bit is enabled, if ECC error is happened during the execution of the command descriptor, it will stop executing the command descriptor.
6	R/W	0x0	CMD_DESCR_CMD_REPT_TIMEOUT_HANDLE_EN 0: Diable 1: Enable After this bit is enabled, If resend command timeout is happened during the execution of the command descriptor, it will stop executing the command descriptor.
5	R/W	0x0	CMD_DESCR_FLASH_STA_HANDLE_EN 0: Diable 1: Enable After this bit is enabled, Judge whether the condition is satisfied(Flash status register NDFC_FLASH_STA bit[7:0] **AND** bitmap_b), If the result is true, it will

			be no longer executing the follow-up command descriptor.
4	R/W	0x0	REG_ACCESS_METHOD 0: AHB BUS 1: MBUS This bit must be set to '1' before set up the command descriptor. It is used to distinguish the command descriptor and normal CPU operation. When switching to the MBUS to get the register mode, only the Global Register access is valid.
3	R/W	0x0	CMD_DESCR_CTL_BIT 0: Write '0', start executing command descriptor, automatic clear if this operation is finished. 1: Write '1', the controller stops after executing the current command descriptor, automatic clear if this operation is finished. This bit indicates the start-up signal of command descriptor.
2	R/W	0x0	NDFC_CHANNEL_SEL 0: BCH Mode & 1KB DATA_BLOCK_SIZE & RANDOMIZER_OLD 1: LDPC Mode & 2KB DATA_BLOCK_SIZE & RANDOMIZER_NEW When this bit is '0', Data_Block=1K bytes. When this bit is '1', Data_Block=2K bytes
1	R/WAC	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

5.2.6.60 NDFC Command Descriptor Base Address Register (Default Value: 0x0000_0000)

Offset: 0x0C04			Register Name: NDFC_CMD_DESCR_BASE_ADDR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	CMD_DESCR_BASE_ADDR First Command Descriptor Memory Address, 8-byte aligned

5.2.6.61 NDFC Command Descriptor Status Register (Default Value: 0x0000_0000)

Offset: 0x0C08			Register Name: NDFC_CMD_DESCR_STA
Bit	**Read/Write**	**Default/Hex**	**Descriptor**
31:4	/	/	/
3	R/W1C	0x0	CMD_DESCR_ECC_ERR_STA Command Descriptor ECC Error status, write '1' to clear.
2	R/W1C	0x0	CMD_DESCR_REPT_MODE_TIMEOUT_STA Command Descriptor REPT Mode TimeOut flag bit, write '1' to clear.

1	R/W1C	0x0	CMD_DESCR_FLASH_STA_HANDLE_STA Command Descriptor Flash Status Handle flag bit, write '1' to clear. (The result of Flash status register NDFC_FLASH_STA(0x100) bit[7:0] AND bitmap_B)
0	R/W1C	0x0	CMD_DESCR_STA Command descriptor executes halt flag bit, write '1' to clear.

5.2.6.62 NDFC Command Descriptor Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0C0C			**Register Name: NDFC_CMD_DESCR_INTR**
Bit	**Read/Write**	**Default/Hex**	**Descriptor**
31:1	/	/	/
0	R/W	0x0	CMD_DESCR_INTR_EN 0: Disable 1: Enable

5.3 SD/MMC Host Controller(SMHC)

5.3.1 Overview

The SD-MMC Host Controller(SMHC) controls the read/write operations on the secure digital(SD) card and multimedia card(MMC),and supports Secure Digital Memory(SD Memory), UHS-I Card, Secure Digital I/O(SDIO), Multimedia Cards(MMC), eMMC.

Features:

- Supports Secure Digital memory protocol commands(up to SD3.0)
- Supports Secure Digital I/O protocol commands(up to SDIO3.0)
- Supports Multimedia Card protocol commands(up to MMC4.51)
- Supports eMMC boot operation and alternative boot operation
- Supports Command Completion signal and interrupt to host processor and Command Completion Signal disable feature.
- SMHC0 supports SD (Version1.0 to 3.0),4-bit bus width
  - SDR mode 50MHz@3.3V IO pad
  - SDR mode 150MHz@1.8V IO pad
  - DDR mode 50MHz@1.8V IO pad
- SMHC1 supports SDIO(Version1.1 to 3.0),4-bit bus width
  - SDR mode 50MHz@3.3V IO pad
  - SDR mode 150MHz@1.8V IO pad
  - DDR mode 50MHz@1.8V IO pad
- SMHC2 supports MMC(Version3.x to 4.2),eMMC(Version4.3-5.0,compatible with 5.1),8-bit bus width
  - SDR mode 150MHz@1.8V IO pad
  - DDR mode 100MHz@1.8V IO pad
  - DDR mode 50MHz@3.3V IO pad
  - SDR mode 50MHz@3.3V IO pad
- Supports hardware CRC generation and error detection
- Supports programmable baud rate
- Supports host pull-up control
- Supports SDIO interrupts in 1-bit and 4-bit modes
- Supports block size of 1 to 65535 bytes
- Supports descriptor-based internal DMA controller
- Internal 1KB FIFO for data transfer
- Supports inline encryption and decryption based on EMCE(Embedded Crypto Engine)

5.3.2 Block Diagram

Figure 5-19 shows a block diagram of the SMHC0/1.

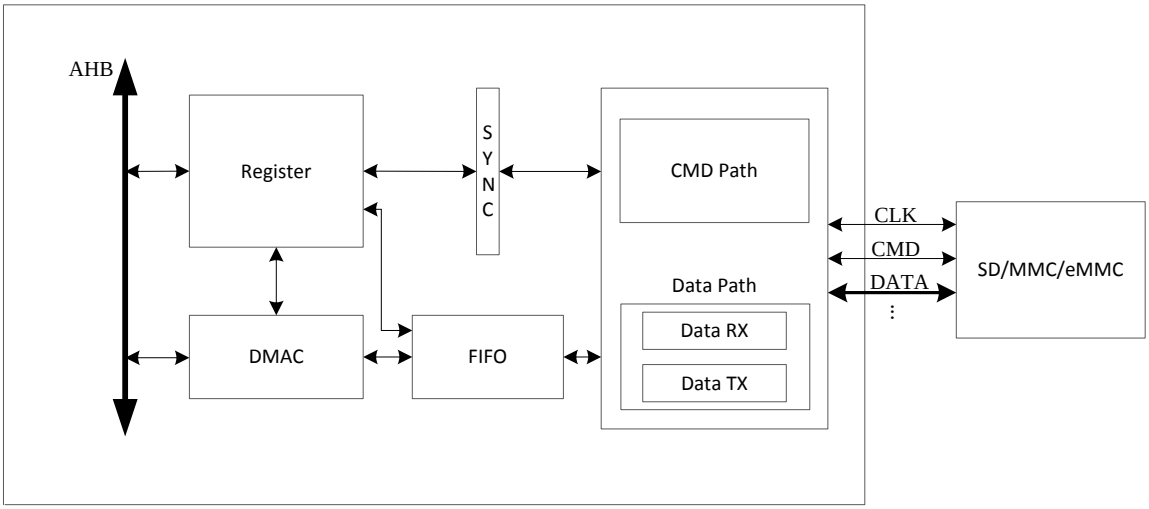


Figure5- 19. SMHC(0/1) Block Diagram

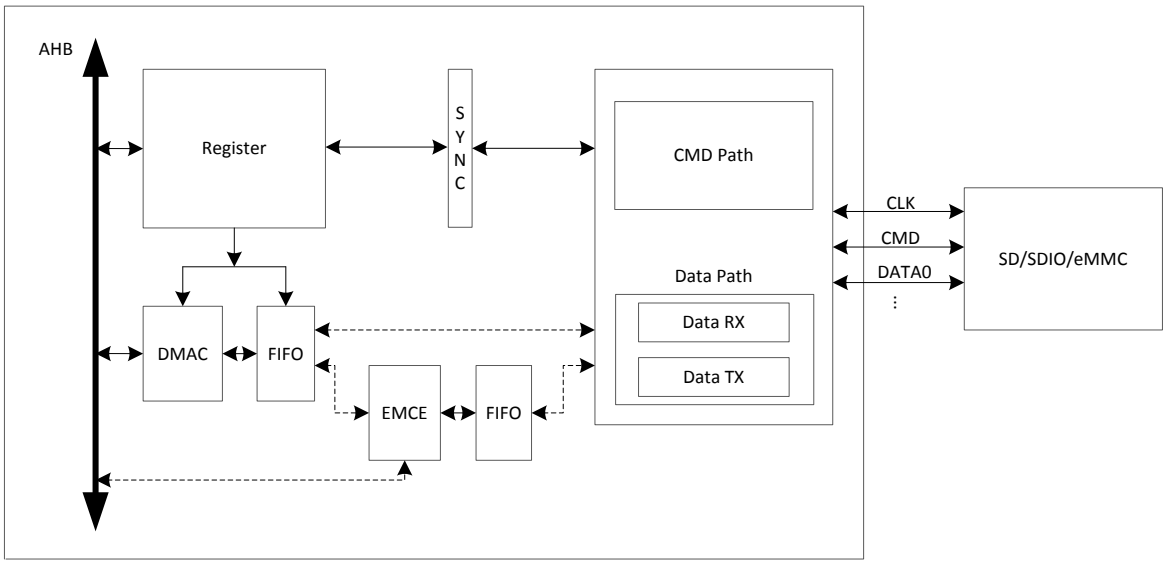


Figure5- 20. SMHC2 Block Diagram

5.3.3 Operations and Functional Descriptions

5.3.3.1 External Signals

Table 5-3 describes the external signals of SMHC.

Table5- 3. SMHC External Signals

Port Name	Width	Direction	Description
SDC0_CLK	1	O	Clock output for SD/TF card
SDC0_CMD	1	I/O	CMD line for SD/TF card
SDC0_D[i] (i=0~3)	4	I/O	Data line for SD/TF card
SDC1_CLK	1	O	Clock output for SDIO Wi-Fi
SDC1_CMD	1	I/O	CMD line for SDIO Wi-Fi
SDC1_D[i] (i=0~3)	4	I/O	Data line for SDIO Wi-Fi

SDC2_CLK	1	O	Clock output for MMC
SDC2_CMD	1	I/O	CMD line for MMC
SDC2_D[i] (i=0~7)	8	I/O	Data line for MMC
SDC2_RST	1	O	Reset signal for MMC
SDC2_DS	1	I	Data strobe for MMC

5.3.3.2 Clock Sources

Each SMHC gets three different clocks. User can select one of them to make SMHC clock source. Table 5-4 describes the clock sources of SMHC. Users can see CCU in chapter 3.3 for clock setting, configuration and gating information.

Table5- 4. SMHC Clock Sources

Clock Sources	Description
OSC24M	24MHz Crystal
PLL_PERIPH0(2X)	Peripheral Clock, the default value is 1.2GHz
PLL_PERIPH1(2X)	Peripheral Clock, the default value is 1.2GHz

5.3.3.3 SMHC Timing Diagram

Please refer to 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)

5.3.3.4 Internal DMA Controller Description

The SMHC has an internal DMA controller (IDMAC) to transfer data between host memory and SMHC port. With a descriptor, IDMAC can efficiently move data from source to destination by automatically loading next DMA transfer arguments, which need less CPU intervention. Before transfer data in IDMAC, host driver should construct a descriptor list, configure arguments of every DMA transfer, then launch the descriptor and start the DMA. IDMAC has an interrupt controller, when enabled, it can interrupt the HOST CPU in situations such as data transmission completed or some errors happened.

5.3.3.4.1 IDMAC Descriptor Structure

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

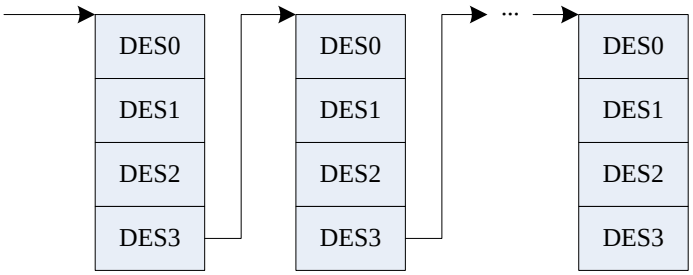


Figure5- 21. IDMAC Descriptor Structure Diagram

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.

5.3.3.4.2 DES0 Definition

Bits	Name	Descriptor
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 transfer is over.
30	ERROR	ERR_FLAG When some error happened in transfer, this bit will be set.
29:5	/	/
4	Chain Flag	CHAIM_MOD When set, this bit indicates that the second address in descriptor is the next descriptor address. Must be set 1.
3	First DES Flag	FIRST_FLAG When set, this bit indicates that this descriptor contains the first buffer of data. Must be set to 1 in first DES.
2	Last DES Flag	LAST_FLAG When set, this bit indicates that the buffers pointed to by this descriptor are the last data buffer
1	Disable Interrupt on completion	CUR_TXRX_OVER_INT_DIS When set, this bit will prevent the setting of the TX/RX interrupt bit of the IDMAC status register for data that ends in the buffer pointed to by this descriptor
0	/	/

5.3.3.4.3 DES1 Definition

SMHC0/SMHC1

Bits	Name	Descriptor
31:16	/	/
15:0	Buffer size	BUFF_SIZE These bits indicate the data buffer byte size, which must be a multiple of 4 bytes. If this filed is 0, the DMA ignores this buffer and proceeds to the next descriptor.

SMHC2

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

5.3.3.4.4 DES2 Definition

Bits	Name	Descriptor
31:0	Buffer address pointer	BUFF_ADDR These bits indicate the physical address of data buffer. The IDMAC ignores DES2[1:0], corresponding to the bus width of 32.

5.3.3.4.5 DES3 Definition

Bits	Name	Descriptor
31:0	Next descriptor address	NEXT_DESP_ADDR These bits indicate the pointer to the physical memory where the next descriptor is present.

5.3.3.5 Calibrate Delay Chain

The sample clock delay chain and Data Strobe delay chain(only in SMHC2) are used to generate delay to make proper timing between sample clock/Data Strobe 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.

The steps to calibrate delay chain are as follows:
Step1: Enable SMHC. In order to calibrate delay chain by operation registers in SMHC, SMHC must be enabled through **SMHC Bus Gating Reset Register** and **SMHC2 Clock Register**.

Step2: Configure a proper clock for SMHC. Calibration delay chain is based on the clock for SMHC from Clock Control Unit(CCU). Calibration delay chain an internal function in SMHC and don't need device. So, it is unnecessary to open clock signal for device. The recommended clock frequency is 200MHz.

Step3: 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.

Step4: Write 0x8000 to **delay control register** to start calibrate delay chain.

Step5: Wait until the flag(Bit14 in **delay control register**) of calibration done is set. The number of delay cells is shown at Bit8~Bit13 in **delay control register**. The delay time generated by these delay cells is equal to the cycle of SMHC's clock nearly. This value is the result of calibration.

Step6: Calculate the delay time of one delay cell according to the cycle of SMHC's clock and the result of calibration.

NOTE

In the above descriptions,delay control register contains SMHC Sample Delay Control Register and SMHC Data Strobe Delay Control Register. Delay Software Enable contains Sample Delay Software Enable and Data Strobe Delay Software Enable. Delay chain contains Sample Delay Software and Data Strobe Delay Software.

5.3.4 Programming Guidelines

5.3.4.1 Initialization

Before data and command are exchanged between a card and the SMHC, the SMHC need to be initialized .The SMHC is initialized as follows.

Step1: Configure GPIO register as SMHC function by Port Controller module; reset clock by writing 1 to SMHC_BGR_REG[SMHCx_RST], open clock gating by writing 1 to SMHC_BGR_REG[SMHCx_GATING]; select clock sources and set division factor by configuring the SMHCx_CLK_REG(x=0,1,2) register.

Step2: Configure SMHC_CTRL to enable total interrupt; configure SMHC_INTMASK to 0xFFCE to enable normal interrupt and error abnormal interrupt, and register interrupt function.

Step3: Configure SMHC_CLKDIV to open clock for device; configure SMHC_CMD as change clock command(for example 0x80202000); send update clock command to deliver clock to device.

Step4: Configure SMHC_CMDARG, configure SMHC_CMD to set response type,etc, then command can send. According to initial process protocol, you can finish SMHC initializing by sending corresponding command one by one.

5.3.4.2 Writing a Single Data Block

To Write a single data block, perform the following steps:

Step1: Write 1 to SMHC_CTRL[DMA_RST] to reset internal DMA controller; write 0x82 to SMHC_DMAC to enable DMAC interrupt, configure AHB master burst transfers; configure SMHC_IDIE to enable transfer interrupt,receive interrupt, and abnormal interrupt.

Step2: Configure SMHC_FIFOTH to determine burst size, TX/RX trigger level. For example, if SMHC_FIFOTH is configured as 0x300F00F0, which indicates that Burst size is 16, TX_TL is 15,RX_TL is 240. Configure SMHC_DLBA to determine the start address of DMA descriptor.

Step3: If writing 1 data block to the first sector, then SMHC_BYCNT[BYTE_CNT] need be set to 0x200, the descriptor is set based on data size; set the data sector address of CMD24(Single Data Block Write) to 0x1, write 0x80002758 to SMHC_CMD, send CMD24 command to write data to device.

Step4: Check whether SMHC_RINTSTS[CC] is 1. If yes, command sends successful; if no, continue to wait until timeout, then exit process.

Step5: Check whether SMHC_IDST_REG[TX_INT] is 1. If yes, writing DMA data transfer is complete, then write 0x337 to SMHC_IDST_REG to clear interrupt flag; if no, continue to wait until timeout, then exit process.

Step6: Check whether SMHC_RINTSTS[DTC] is 1. If yes, data transfer is complete and CMD24 writing operation is

complete. If no, that is, abnormity exists. Read SMHC_RINTSTS,SMHC_STATUS to query existing abnormity.

Step7: Send CMD13 command to query whether device writing operation is complete and whether return to idle state. 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, then check whether the highest bit of SMHC_RESP0(CMD13 response) is 1. If yes, device is in Idle state,then the next command can be sent. If no, device is in busy state, then continue to send CMD13 to wait device idle until timeout exit.

5.3.4.3 Reading a Single Data Block

To read a single data block, perform the following steps:

Step1: Write 1 to SMHC_CTRL[DMA_RST] to reset internal DMA controller; write SMHC_DMAC to 0x82 to enable DMAC interrupt and configure AHB master burst transfers; configure SMHC_IDIE to enable transfer interrupt,receive interrupt, and abnormal interrupt.

Step2: Configure SMHC_FIFOTH to determine burst size, TX/RX trigger level. For example, if SMHC_FIFOTH is configured as 0x300F00F0, which indicates that Burst size is 16, TX_TL is 15,RX_TL is 240. Configure SMHC_DLBA to determine the start address of DMA descriptor.

Step3: If reading 1 data block from the first sector, then SMHC_BYCNT[BYTE_CNT] need be set to 0x200, the descriptor is set based on data size; set the data sector address of CMD17(Single Data Block Read) to 0x1, write 0x80002351 to SMHC_CMD , send CMD17 command to read data from device to DRAM/SRAM.

Step4: Check whether SMHC_RINTSTS[CC] is 1. If yes, command sends successful; if no, continue to wait until timeout, then exit process.

Step5: Check whether SMHC_IDST_REG[RX_INT] is 1. If yes, writing DMA data transfer is complete, then write 0x337 to SMHC_IDST_REG to clear interrupt flag; if no, continue to wait until timeout, then exit process.

Step6: Check whether SMHC_RINTSTS[DTC] is 1. If yes, data transfer is complete and CMD17 reading operation is complete. If no, that is, abnormity exists. Read SMHC_RINTSTS,SMHC_STATUS to query existing abnormity.

5.3.4.4 Writing Open-ended Multiple Data Blocks(CMD25+Auto CMD12)

To write open-ended multiple data blocks, perform the following steps:

Step1: Write 1 to SMHC_CTRL[DMA_RST] to reset internal DMA controller; write SMHC_DMAC to 0x82 to enable DMAC interrupt and configure AHB master burst transfers; configure SMHC_IDIE to enable transfer interrupt,receive interrupt, and abnormal interrupt.

Step2: Configure SMHC_FIFOTH to determine burst size, TX/RX trigger level. For example, if SMHC_FIFOTH is configured as 0x300F00F0, which indicates that Burst size is 16, TX_TL is 15,RX_TL is 240. Configure SMHC_DLBA to determine the start address of DMA descriptor.

Step3: If writing 3 data blocks to the zero sector, then SMHC_BYCNT[BYTE_CNT] need be set to 0x600, the descriptor is set based on data size; set the data sector address of CMD25(Multiple Data Blocks Write) to 0x0, write 0x80003759 to SMHC_CMD, send CMD25 command to write data to device, when data transfer is complete, CMD12 will be sent automatically .

Step4: Check whether SMHC_RINTSTS[CC] is 1. If yes, command sends successful; if no, continue to wait until timeout, then exit process.

Step5: Check whether SMHC_IDST_REG[TX_INT] is 1. If yes, writing DMA data transfer is complete, then write 0x337 to SMHC_IDST_REG to clear interrupt flag; if no, continue to wait until timeout, then exit process.

Step6: Check whether SMHC_RINTSTS[ACD] and SMHC_RINTSTS[DTC] are all 1. If yes, data transfer is complete, CMD12 transfer is complete and CMD25 writing operation is complete. If no, that is, abnormity exists. Read

SMHC_RINTSTS,SMHC_STATUS to query existing abnormity.

Step7: Send CMD13 command to query whether device writing operation is complete and whether return to idle state. 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, then check whether the highest bit of SMHC_RESP0(CMD13 response) is 1. If yes, device is in Idle state,then the next command can be sent. If no, device is in busy state, then continue to send CMD13 to wait device idle until timeout exit.

5.3.4.5 Reading Open-ended Multiple Data Blocks(CMD18+Auto CMD12)

To read open-ended multiple data blocks, perform the following steps:

Step1: Write 1 to SMHC_CTRL[DMA_RST] to reset internal DMA controller; write SMHC_DMAC to 0x82 to enable DMAC interrupt and configure AHB master burst transfers; configure SMHC_IDIE to enable transfer interrupt,receive interrupt, and abnormal interrupt.

Step2: Configure SMHC_FIFOTH to determine burst size, TX/RX trigger level. For example, if SMHC_FIFOTH is configured as 0x300F00F0, which indicates that Burst size is 16, TX_TL is 15,RX_TL is 240. Configure SMHC_DLBA to determine the start address of DMA descriptor.

Step3: If reading 3 data blocks from the zero sector, then SMHC_BYCNT[BYTE_CNT] need be set to 0x600, the descriptor is set based on data size; set the data sector address of CMD18(Multiple Data Blocks Read) to 0x0, write 0x80003352 to SMHC_CMD , send CMD18 command to read data to device, when data transfer is complete, CMD12 will be sent automatically.

Step4: Check whether SMHC_RINTSTS[CC] is 1. If yes, command sends successful; if no, continue to wait until timeout, then exit process.

Step5: Check whether SMHC_IDST_REG[RX_INT] is 1. If yes, writing DMA data transfer is complete, then write 0x337 to SMHC_IDST_REG to clear interrupt flag; if no, continue to wait until timeout, then exit process.

Step6: Check whether SMHC_RINTSTS[ACD] and SMHC_RINTSTS[DTC] are all 1. If yes, data transfer is complete, CMD12 transfer is complete and CMD18 reading operation is complete. If no, that is, abnormity exists. Read SMHC_RINTSTS,SMHC_STATUS to query existing abnormity.

5.3.4.6 Writing Pre-defined Multiple Data Blocks(CMD23+CMD25)

To write pre-defined multiple data blocks, perform the following steps:

Step1: Write 1 to SMHC_CTRL[DMA_RST] to reset internal DMA controller; write SMHC_DMAC to 0x82 to enable DMAC interrupt and configure AHB master burst transfers; configure SMHC_IDIE to enable transfer interrupt,receive interrupt, and abnormal interrupt.

Step2: Configure SMHC_FIFOTH to determine burst size, TX/RX trigger level. For example, if SMHC_FIFOTH is configured as 0x300F00F0, which indicates that Burst size is 16, TX_TL is 15,RX_TL is 240. Configure SMHC_DLBA to determine the start address of DMA descriptor.

Step3: If writing 3 data blocks, then set SMHC_CMDARG to 0x3 to ensure the block number to be operated, send CMD23 command by writing 0x80000157 to SMHC_CMD .Check whether SMHC_RINTSTS[CC] is 1.If yes, command sends successful; if no, continue to wait until timeout, then exit process.

Step4: SMHC_BYCNT[BYTE_CNT] need be set to 0x600, the descriptor is set based on data size; set the data sector address of CMD25(Multiple Data Blocks Write) to 0x0, write 0x80002759 to SMHC_CMD, send CMD25 command to write data to device.

Step5: Check whether SMHC_RINTSTS[CC] is 1. If yes, command sends successful; if no, continue to wait until timeout, then exit process.

Step6: Check whether SMHC_IDST_REG[TX_INT] is 1. If yes, writing DMA data transfer is complete, then write 0x337 to SMHC_IDST_REG to clear interrupt flag; if no, continue to wait until timeout, then exit process.

Step7: Check whether SMHC_RINTSTS[DTC] is 1. If yes, data transfer is complete and CMD25 writing operation is complete. If no, that is, abnormity exists. Read SMHC_RINTSTS,SMHC_STATUS to query existing abnormity.

Step8: Send CMD13 command to query whether device writing operation is complete and whether return to idle state. 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, then check whether the highest bit of SMHC_RESP0(CMD13 response) is 1. If yes, device is in Idle state,then the next command can be sent. If no, device is in busy state, then continue to send CMD13 to wait device idle until timeout exit.

5.3.4.7 Reading Pre-defined Multiple Data Blocks(CMD23+CMD18)

To read pre-defined multiple data blocks, perform the following steps:

Step1: Write 1 to SMHC_CTRL[DMA_RST] to reset internal DMA controller; write SMHC_DMAC to 0x82 to enable DMAC interrupt and configure AHB master burst transfers; configure SMHC_IDIE to enable transfer interrupt,receive interrupt, and abnormal interrupt.

Step2: Configure SMHC_FIFOTH to determine burst size, TX/RX trigger level. For example, if SMHC_FIFOTH is configured as 0x300F00F0, which indicates that Burst size is 16, TX_TL is 15,RX_TL is 240. Configure SMHC_DLBA to determine the start address of DMA descriptor.

Step3: If reading 3 data blocks, then set SMHC_CMDARG to 0x3 to ensure the block number to be operated, send CMD23 command by writing 0x80000157 to SMHC_CMD .Check whether SMHC_RINTSTS[CC] is 1.If yes, command sends successful; if no, continue to wait until timeout, then exit process.

Step4: SMHC_BYCNT[BYTE_CNT] need be set to 0x600, the descriptor is set based on data size; set the data sector address of CMD18(Multiple Data Blocks Read) to 0x0, write 0x80002352 to SMHC_CMD, send CMD18 command to read data from device to DRAM/SRAM.

Step5: Check whether SMHC_RINTSTS[CC] is 1. If yes, command sends successful; if no, continue to wait until timeout, then exit process.

Step6: Check whether SMHC_IDST_REG[TX_INT] is 1. If yes, writing DMA data transfer is complete, then write 0x337 to SMHC_IDST_REG to clear interrupt flag; if no, continue to wait until timeout, then exit process.

Step7: Check whether SMHC_RINTSTS[DTC] is 1. If yes, data transfer is complete and CMD18 writing operation is complete. If no, that is, abnormity exists. Read SMHC_RINTSTS,SMHC_STATUS to query existing abnormity.

5.3.5 Register List

Module Name	Base Address
SMHC0	0x04020000
SMHC1	0x04021000
SMHC2	0x04022000

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_CSDC	0x0054	CRC Status Detect Control Register
SMHC_A12A	0x0058	Auto Command 12 Argument Register
SMHC_NTSR	0x005C	SD New Timing Set Register
SMHC_EMCE	0x0064	Embedded Encrypt and Decrypt Control Register
SMHC_HWRST	0x0078	Hardware Reset Register
SMHC_DMAC	0x0080	DMA Control Register
SMHC_DLBA	0x0084	Descriptor List Base Address Register
SMHC_IDST	0x0088	DMAC Status Register
SMHC_IDIE	0x008C	DMAC 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
SMHC_EDSD	0x010C	eMMC4.5 DDR Start Bit Detection Control Register
SMHC_RES_CRC	0x0110	Response CRC from Device
SMHC_D7_CRC	0x0114	CRC in Data7 from Device
SMHC_D6_CRC	0x0118	CRC in Data6 from Device
SMHC_D5_CRC	0x011C	CRC in Data5 from Device
SMHC_D4_CRC	0x0120	CRC in Data4 from Device
SMHC_D3_CRC	0x0124	CRC in Data3 from Device
SMHC_D2_CRC	0x0128	CRC in Data2 from Device
SMHC_D1_CRC	0x012C	CRC in Data1 from Device
SMHC_D0_CRC	0x0130	CRC in Data0 from Device
SMHC_CRC_STA	0x0134	CRC Status from Device in Write Operation
SMHC_EXT_CMD	0x138	Extended command register
SMHC_EXT_RESP	0x13C	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
/	0x014C-0x01FC	Reserved

SMHC_FIFO	0x0200	Read/ Write FIFO

5.3.6 Register Description

5.3.6.1 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 1: AHB bus 0: DMA bus
30:13	/	/	/
12	R/W	0x0	TIME_UNIT_CMD Time unit for command line Time unit used to calculate command line time out value defined in RTO_LMT. 0: 1 card clock period 1: 256 card clock period
11	R/W	0x0	TIME_UNIT_DAT Time unit for data line Time unit used to calculate 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 eMMC's HS400 speed mode is 8-bit DDR, this filed 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, using 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 change 1: Reset FIFO This bit is auto-cleared after completion of reset operation.
0	R/W	0x0	SOFT_RST Software Reset 0: No change 1: Reset SD/MMC controller This bit is auto-cleared after completion of reset operation.

5.3.6.2 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 always on 1 : Turn off card clock when FSM in IDLE state
16	R/W	0x0	CCLK_ENB Card Clock Enable 0: Card Clock off 1: Card Clock on
15:8	/	/	/
7:0	R/W	0x0	CCLK_DIV Card clock divider n: Source clock is divided by 2*n.(n=0~255) when HS400_MD_EN is set, this field must be cleared.

5.3.6.3 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 time of the Host wait for the data from the Device. Ensure to communicate with the Device,this field must be set to maximum that 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,CMD18). When Host read multiple block(CMD18),the 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 Host write data, this value is no effect.
7:0	R/W	0x40	RTO_LMT Response Timeout Limit

5.3.6.4 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

5.3.6.5 SMHC Block Size Register(Default Value:0x0000_0200)

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

5.3.6.6 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 Number of bytes to be transferred; should be integer multiple of Block Size for block transfers.

5.3.6.7 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 auto cleared when current command is sent. If there is no any response error happened, a command complete interrupt bit (CMD_OVER) will be set in interrupt register. You should not write any other command before this bit is cleared, or a command busy interrupt bit (CMD_BUSY) will be set in interrupt register.
30:29	/	/	/
28	R/W	0x0	VOL_SW Voltage Switch 0: normal command 1: Voltage switch command, set for CMD11 only
27	R/W	0x0	BOOT_ABT Boot Abort Setting this bit will terminate the boot operation.
26	R/W	0x0	EXP_BOOT_ACK Expect Boot Acknowledge. When Software sets this bit along in mandatory boot operation, 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, controller will change clock domain and clock output. No command 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 current data transfer in progress.(CMD12, CMD52 for writing “I/O Abort” in SDIO CCCR)
13	R/W	0x0	WAIT_PRE_OVER Wait Data Transfer Over 0: Send command at once, do not care of data transferring

			1: Wait for data transfer completion before sending current command
12	R/W	0x0	STOP_CMD_FLAG Send Stop CMD Automatically (CMD12) 0: Do not send stop command at end of data transfer 1: Send stop command automatically at end of 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

5.3.6.8 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

5.3.6.9 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

5.3.6.10 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

5.3.6.11 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

5.3.6.12 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

5.3.6.13 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
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	/	/	/

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

Offset: 0x0034			**Register Name: SMHC_MINTSTS**
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	M_CARD_REMOVAL_INT Card Removed
30	R/W	0x0	M_CARD_INSERT Card Inserted
29:17	/	/	/
16	R/W	0x0	M_SDIO_INT SDIO Interrupt
15	R/W	0x0	M_DEE_INT Data End-bit Error When set during receiving data, it means that host controller does not receive valid data end bit.

			When set during transmitting data, it means that host controller does not receive CRC status taken or received CRC status taken is negative.
14	R/W	0x0	M_ACD_INT Auto Command Done When set, it means auto stop command(CMD12) completed.
13	R/W	0x0	M_DSE_BC_INT Data Start Error When set during receiving data, it means that host controller found a error start bit. When set during transmitting data, it means that busy signal is cleared.
12	R/W	0x0	M_CB_IW_INT Command Busy and Illegal Write
11	R/W	0x0	M_FU_FO_INT FIFO Underrun/Overflow
10	R/W	0x0	M_DSTO_VSD_INT Data Starvation Timeout/V1.8 Switch Done
9	R/W	0x0	M_DTO_BDS_INT Data Timeout/Boot Data Start
8	R/W	0x0	M_RTO_BACK_INT Response Timeout/Boot ACK Received
7	R/W	0x0	M_DCE_INT Data CRC Error When set during receiving data, it means that the received data have data CRC error. When set during transmitting data, it means that the received CRC status taken is negative.
6	R/W	0x0	M_RCE_INT Response CRC Error
5	R/W	0x0	M_DRR_INT Data Receive Request When set, it means that there are enough data in FIFO during receiving data.
4	R/W	0x0	M_DTR_INT Data Transmit Request When set, it means that there are enough space in FIFO during transmitting data.
3	R/W	0x0	M_DTC_INT Data Transfer Complete
2	R/W	0x0	M_CC_INT Command Complete
1	R/W	0x0	M_RE_INT Response Error When set, Transmit Bit error or End Bit error or CMD Index error may occurs.
0	/	/	/

5.3.6.15 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 This is write-1-to-clear bits.
30	R/W1C	0x0	CARD_INSERT Card Inserted This is write-1-to-clear bits.
29:17	/	/	/
16	R/W1C	0x0	SDIOI_INT SDIO Interrupt This is write-1-to-clear bits.
15	R/W1C	0x0	DEE Data End-bit Error When set during receiving data, it means that host controller does not receive valid data end bit. When set during transmitting data, it means that host controller does not receive CRC status taken. This is write-1-to-clear bits.
14	R/W1C	0x0	ACD Auto Command Done When set, it means auto stop command(CMD12) completed. This is write-1-to-clear bits.
13	R/W1C	0x0	DSE_BC Data Start Error When set during receiving data, it means that host controller found a error start bit. It is valid at 4-bit or 8-bit bus mode. When it set, host found start bit at data0, but did not find start bit at some or all of the other data lines. When set during transmitting data, it means that busy signal is cleared. This is write-1-to-clear bits.
12	R/W1C	0x0	CB_IW Command Busy and Illegal Write This is write-1-to-clear bits.
11	R/W1C	0x0	FU_FO FIFO Underrun/Overflow This is write-1-to-clear bits.
10	R/W1C	0x0	DSTO_VSD Data Starvation Timeout/V1.8 Switch Done This is write-1-to-clear bits.
9	R/W1C	0x0	DTO_BDS Data Timeout/Boot Data Start When set during receiving data, it means host did not find start bit on data0.

			This is write-1-to-clear bits.
8	R/W1C	0x0	RTO_BACK Response Timeout/Boot ACK Received This is write-1-to-clear bits.
7	R/W1C	0x0	DCE Data CRC Error When set during receiving data, it means that the received data have data CRC error. When set during transmitting data, it means that the received CRC status taken is negative. This is write-1-to-clear bits.
6	R/W1C	0x0	RCE Response CRC Error This is write-1-to-clear bits.
5	R/W1C	0x0	DRR Data Receive Request When set, it means that there are enough data in FIFO during receiving data. This is write-1-to-clear bits.
4	R/W1C	0x0	DTR Data Transmit Request When set, it means that there are enough space in FIFO during transmitting data. This is write-1-to-clear bits.
3	R/W1C	0x0	DTC Data Transfer Complete When set, it means that current command completes even through error occurs. This is write-1-to-clear bits.
2	R/W1C	0x0	CC Command Complete This is write-1-to-clear bits.
1	R/W1C	0x0	RE Response Error When set, Transmit Bit error or End Bit error or CMD Index error may occurs. This is write-1-to-clear bits.
0	/	/	/

5.3.6.16 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 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 not busy 1: card data busy
8	R	0x0	CARD_PRESENT Data[3] status level of DATA[3], checks whether card is present 0: card not present 1: card present
7:4	R	0x0	FSM_STA Command FSM States 0000: Idle 0001: Send init sequence 0010: TX CMD start bit 0011: TX CMD TX bit 0100: TX CMD index + argument 0101: TX CMD CRC7 0110: TX CMD end bit 0111: RX response start bit 1000: RX response IRQ response 1001: RX response TX bit 1010: RX response CMD index 1011: RX response data 1100: RX response CRC7 1101: RX response end bit 1110: CMD path wait NCC 1111: Wait; CMD-to-response turnaround
3	R	0x0	FIFO_FULL FIFO Full 1: FIFO full 0: FIFO not full
2	R	0x1	FIFO_EMPTY FIFO Empty 1: FIFO Empty

			0: FIFO not Empty
1	R	0x1	FIFO_TX_LEVEL FIFO TX Water Level Flag 0: FIFO didn't reach transmit trigger level 1: FIFO reached transmit trigger level
0	R	0x0	FIFO_RX_LEVEL FIFO TX Water Level Flag 0: FIFO didn't reach receive trigger level 1: FIFO reached receive trigger level

5.3.6.17 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 Should be programmed same as DMA controller multiple transaction size. The units for transfers 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) FIFO_DEPTH = 256 FIFO_SIZE = 256 * 32 = 1K Recommended: SMHC2: Burst Size = 16, TX_TL = 240, RX_TL = 15 Other SD card controller: Burst Size = 8, TX_TL =248, RX_TL = 7.
27:24	/	/	/
23:16	R/W	0xF	RX_TL RX Trigger Level 0x0~0xFE: RX Trigger Level is 0~254 0xFF: Reserved FIFO threshold when FIFO request host to receive data from FIFO. When FIFO data level is greater than this value, DMA is request is raised if DMA enabled, or RX interrupt bit is set if interrupt enabled. At the end of packet, if the last transfer is less than this level, the value is ignored and relative request will be raised as usual. Recommended: SMHC2: 15 (means greater than 15) Other SD card controller: 7(means greater 7)

15:8	/	/	/
7:0	R/W	0x0	TX_TL TX Trigger Level 0x1~0xFF: TX Trigger Level is 1~255 0x0: no trigger FIFO threshold when FIFO requests host to transmit data to FIFO. When FIFO data level is less than or equal to this value, DMA TX request is raised if DMA enabled, or TX request interrupt bit is set if interrupt enabled. At the end of packet, if the last transfer is less than this level, the value is ignored and relative request will be raised as usual. Recommended: SMHC2: 240(means less than or equal to 240). Other SD Card Controller: 248(means less than or equal to 248).

5.3.6.18 SMHC Function Select Register(Default Value: 0x0000_0000)

Offset: 0x0044			Register Name: SMHC_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2	R/W	0x0	ABT_RDATA Abort Read Data 0: Ignored 1: After suspend command is issued during read-transfer, software polls card to find when suspend happened. Once suspend occurs, software sets bit to reset data state-machine, which is waiting for next block of data. Used in SDIO card suspends sequence. This bit is auto-cleared once controller reset to idle state.
1	R/W	0x0	READ_WAIT Read Wait 0: Clear SDIO read wait 1: Assert SDIO read wait
0	R/W	0x0	HOST_SEND_MMC_IRQRESQ Host Send MMC IRQ Response 0: Ignored 1: Send auto IRQ response When host is waiting MMC card interrupt response, setting this bit will make controller cancel wait state and return to idle state, at which time, controller will receive IRQ response sent by itself. This bit is auto-cleared after response is sent.

5.3.6.19 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 Number of bytes transferred between card and internal FIFO. The register should be accessed in full to avoid read-coherency problems and read only after data transfer completes.

5.3.6.20 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 Number of bytes transferred between Host/DMA memory and internal FIFO. The register should be accessed in full to avoid read-coherency problems and read only after data transfer completes.

5.3.6.21 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

NOTE

This register is valid only for SMHC2.

5.3.6.22 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. SD_A12A set the argument of command 12 automatically send by controller

5.3.6.23 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_SELEC 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 During update clock operation, clear command line's and data lines' input phase. 0: Disable 1: Enable
23	/	/	/
22	R/W	0x1	DAT_CRC_STATUS_RX_PHASE_CLR Before receive CRC status, clear data lines' input phase. 0: Disable 1: Enable
21	R/W	0x1	DAT_TRANS_RX_PHASE_CLR Before transfer data, clear data lines' input phase. 0: Disable 1: Enable
20	R/W	0x1	DAT_RECV_RX_PHASE_CLR Before receive data, clear data lines' input phase clear 0: Disable 1: Enable
19:17	/	/	/
16	R/W	0x1	CMD_SEND_RX_PHASE_CLR Before send command, command rx phase clear 0: Disable 1: Enable
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: Ignore Default Value: 00
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 Default Value: 0°
3:0	/	/	/

NOTE

This register is reserved for SMHC2, but valid for other SMHCs.

5.3.6.24 SMHC EMCE Control Register (Default Value: 0x0200_0000)

Offset: 0x0064			Register Name: SMHC_EMCE
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	R/W	0x200	SEC_SZ Sector Size of EMCE. Only support 512 bytes.
15:5	/	/	/
4	R/W	0x0	EMCE_ENCR 0:disabled encrypt and decrypt.Encrypt and decrypt will be bypassed in EMCE 1:enable encrypt and decrypt This bit is only available when EMCE_ENB is 1.
3:2	/	/	/
1	R/W	0x0	AC_MD Access Mode 0: Sector mode. The data address to access device is in sector(512 bytes) units. SMHC_CMDARG is a 32bit sector(512 bytes) address. 1: Byte mode. The data address to access device is in bytes units. SMHC_CMDARG is a 32bit byte address.
0	R/W	0x0	EMCE_ENB EMCE Enable 0: EMCE is disabled.EMCE will be bypass by host controller. 1: EMCE is enabled. The data of current command shall go through EMCE.Encrypt or decrypt is decided by EMCE_ENCR. The type and configuration of encryption algorithm are determined by EMCE's initial process, and out of this specification.

NOTE

This register is valid only for SMHC2.

5.3.6.25 SMHC Hardware Reset Register (Default Value: 0x0000_0001)

Offset: 0x78			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 pre-idle state, which requires them to be

			re-initialized.

NOTE

This register is valid only for SMHC2.

5.3.6.26 SMHC DMAC Control Register (Default Value: 0x0000_0000)

Offset: 0x0080			Register Name: SMHC_DMAC
Bit	**Read/Write**	**Default/Hex**	**Description**
31	W	0x0	DES_LOAD_CTRL When DMAC fetches a descriptor, if the valid bit of a descriptor is not set, DMAC FSM will go to the suspend state. Setting this bit will make DMAC re-fetch 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. DE is read/write.
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, INCR8 during 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.

5.3.6.27 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. The LSB bits [1:0] are ignored and taken as all-zero by the IDMAC internally. Hence these LSB bits are read-only.

5.3.6.28 SMHC DMAC 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	DMAC_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 transmission 010: Host Abort received during reception Others: Reserved EB is read-only.
9	R/W1C	0x0	ABN_INT_SUM 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. Writing a 1 clears this bit.
8	R/W1C	0x0	NOR_INT_SUM 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. Writing a 1 clears this bit.
7:6	/	/	/
5	R/W1C	0x0	ERR_FLAG_SUM Card Error Summary. Indicates the status of the transaction to/from the card; also present in RINTSTS. Indicates the logical OR of the following bits: EBE: End Bit Error RTO: Response Timeout/Boot ACK Timeout RCRC: Response CRC SBE: Start Bit Error DRTO: Data Read Timeout/BDS timeout DCRC: Data CRC for Receive RE: Response Error Writing a 1 clears this bit.
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. Writing a 1 clears this bit.
0	R/W1C	0x0	TX_INT Transmit Interrupt. Indicates that data transmission is finished for a descriptor. Writing a '1' clears this bit.

5.3.6.29 SMHC DMAC Interrupt Enable Register (Default Value: 0x0000_0000)

Offset: 0x008C			**Register Name: SMHC_IDIE_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:10	/	/	/
9	R/W	0x0	Reserved
8	R/W	0x0	Reserved
7:6	/	/	/
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, 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, Transmit Interrupt is disabled.

5.3.6.30 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_RD_THLD Card Read Threshold Size
15:3	/	/	/
2	R/W	0x0	CARD_WR_THLD_ENB (only for SMHC2) Card Write Threshold Enable(HS400) 0: Card write threshold disable 1: Card write threshold enabled Host controller initiates write transfer only if card threshold amount of data is available in transmit FIFO
1	R/W	0x0	BCIG (only for SMHC2) 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 Disable 1: Card Read Threshold Enable Host controller initiates Read Transfer only if CARD_RD_THLD amount of space is available in receive FIFO.

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

Offset: 0x0104			Register Name: SMHC_SFC
Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7:4	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 same with BLK_SZ, the device clock may stop at block gap during data receiving. This field is used to control the position of stopping clock. The value can be change between 0x0 and 0xF,but actually the available value and the position of stopping clock must be decided by the actual situation. The value increase one in this field is linked to one cycle(two cycle in DDR mode) that the position of stopping clock moved up.
0	R/W	0x0	BYPASS_EN Bypass enable When set, sample FIFO will be bypassed.

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

Offset: 0x108			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.

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

Offset: 0x010C			**Register Name: SMHC_EDSD**
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	HS400_MD_EN(for SMHC2 only) HS400 Mode Enable 0: Disable 1: Enable 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.

5.3.6.34 SMHC Response CRC Register (Default Value: 0x0000_0000)

Offset: 0x0110			**Register Name: SMHC_RESP_CRC**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:7	/	/	/
6:0	R	0x0	RESP_CRC Response CRC Response CRC from device.

NOTE

This register is reserved for SMHC2, but valid for other SMHCs.

5.3.6.35 SMHC Data7 CRC Register (Default Value: 0x0000_0000)

Offset: 0x0114			Register Name: SMHC_DAT7_CRC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	DAT7_CRC Data[7] CRC CRC in data[7] from device. In 8bit DDR mode, the higher 16 bits indicate the CRC of even data, and the lower 16 bits indicate the CRC of odd data. In 4 bit DDR mode,it is not used. In SDR mode, the higher 16 bits indicate the CRC of all data.

NOTE

This register is reserved for SMHC2, but valid for other SMHCs.

5.3.6.36 SMHC Data6 CRC Register (Default Value: 0x0000_0000)

Offset: 0x0118			Register Name: SMHC_DAT6_CRC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	DAT6_CRC Data[6] CRC CRC in data[6] from device. In 8bit DDR mode, the higher 16 bits indicate the CRC of even data, and the lower 16 bits indicate the CRC of odd data. In 4 bit DDR mode,it is not used. In SDR mode, the higher 16 bits indicate the CRC of all data.

NOTE

This register is reserved for SMHC2, but valid for other SMHCs.

5.3.6.37 SMHC Data5 CRC Register (Default Value: 0x0000_0000)

Offset: 0x011C			Register Name: SMHC_DAT5_CRC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	DAT5_CRC Data[5] CRC CRC in data[5] from device. In 8bit DDR mode, the higher 16 bits indicate the CRC of even data, and the lower 16 bits indicate the CRC of odd data. In 4 bit DDR mode,it is not used. In SDR mode, the higher 16 bits indicate the CRC of all data.

NOTE

This register is reserved for SMHC2, but valid for other SMHCs.

5.3.6.38 SMHC Data4 CRC Register (Default Value: 0x0000_0000)

Offset: 0x0120			Register Name: SMHC_DAT4_CRC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	DAT4_CRC Data[4] CRC CRC in data[4] from device. In 8 bit DDR mode, the higher 16 bits indicate the CRC of even data, and the lower 16 bits indicate the CRC of odd data. In 4 bit DDR mode, the higher 16 bits indicate the CRC of odd data, and the lower 16 bits indicate the CRC of even data. In SDR mode, the higher 16 bits indicate the CRC of all data.

NOTE

This register is reserved for SMHC2, but valid for other SMHCs.

5.3.6.39 SMHC Data3 CRC Register (Default Value: 0x0000_0000)

Offset: 0x0124			Register Name: SMHC_DAT3_CRC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	DAT3_CRC Data[3] CRC CRC in data[3] from device. In 8 bit DDR mode, the higher 16 bits indicate the CRC of even data, and the lower 16 bits indicate the CRC of odd data. In 4 bit DDR mode, the higher 16 bits indicate the CRC of odd data, and the lower 16 bits indicate the CRC of even data. In SDR mode, the higher 16 bits indicate the CRC of all data.

NOTE

This register is reserved for SMHC2, but valid for other SMHCs.

5.3.6.40 SMHC Data2 CRC Register (Default Value: 0x0000_0000)

Offset: 0x0128			Register Name: SMHC_DAT2_CRC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	DAT2_CRC Data[2] CRC CRC in data[2] from device. In 8 bit DDR mode, the higher 16 bits indicate the CRC of even data, and the lower 16 bits indicate the CRC of odd data. In 4 bit DDR mode, the higher 16 bits indicate the CRC of odd data, and the

			lower 16 bits indicate the CRC of even data. In SDR mode, the higher 16 bits indicate the CRC of all data.

NOTE

This register is reserved for SMHC2, but valid for other SMHCs.

5.3.6.41 SMHC Data1 CRC Register (Default Value: 0x0000_0000)

Offset: 0x012C			Register Name: SMHC_DAT1_CRC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	DAT1_CRC Data[1] CRC CRC in data[1] from device. In 8 bit DDR mode, the higher 16 bits indicate the CRC of even data, and the lower 16 bits indicate the CRC of odd data. In 4 bit DDR mode, the higher 16 bits indicate the CRC of odd data, and the lower 16 bits indicate the CRC of even data. In SDR mode, the higher 16 bits indicate the CRC of all data.

NOTE

This register is reserved for SMHC2, but valid for other SMHCs.

5.3.6.42 SMHC Data0 CRC Register (Default Value: 0x0000_0000)

Offset: 0x0130			Register Name: SMHC_DAT0_CRC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	DAT0_CRC Data[0] CRC CRC in data[0] from device. In 8 bit DDR mode, the higher 16 bits indicate the CRC of even data, and the lower 16 bits indicate the CRC of odd data. In 4 bit DDR mode, the higher 16 bits indicate the CRC of odd data, and the lower 16 bits indicate the CRC of even data. In SDR mode, the higher 16 bits indicate the CRC of all data.

NOTE

This register is reserved for SMHC2, but valid for other SMHCs.

5.3.6.43 SMHC CRC Status Register (Default Value: 0x0000_0000)

Offset: 0x0134			Register Name: SMHC_CRC_STA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/

2:0	R	0x0	CRC_STA CRC Status CRC status from device in write operation Positive CRC status token: 3'b010 Negative CRC status token: 3'b101

NOTE

This register is reserved for SMHC2, but valid for other SMHCs.

5.3.6.44 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 set this bit, send CMD23 automatically before send command specified in SMHC_CMD register. When SOFT_RST set, this field will be cleared.

5.3.6.45 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.

5.3.6.46 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 0: Data drive phase offset is 90° at SDR mode, 45° at DDR 8-bit mode, 90^0 at DDR4/HS400 mode. 1: Data drive phase offset is 180° at SDR mode, 90° at DDR 8-bit mode, 0° at DDR4/HS400 mode.
16	R/W	0x1	CMD_DRV_PH_SEL Command Drive Phase Select 0: Command drive phase offset is 90^0 at SDR mode, 45^0 at DDR 8-bit mode, 90^0 at DDR4/HS400 mode.

			1: Command drive phase offset is 180^0 at SDR mode, 90^0 at DDR 8-bit mode, 180^0 at DDR4/HS400 mode.
15:0	/	/	/

5.3.6.47 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, 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 current card clock. The delay time generated by these delay cells is equal to the cycle of card clock nearly. Generally, it is necessary to do drive delay calibration when card clock is changed. This bit is valid only when SAMP_DL_CAL_DONE is set.
7	R/W	0x0	SAMP_DL_SW_EN Sample Delay Software Enable When set, enable sample delay specified at SAMP_DL_SW
6	/	/	/
5:0	R/W	0x0	SAMP_DL_SW Sample Delay Software The relative delay between clock line and command line, data lines. It can be determined according to the value of SAMP_DL, the cycle of card clock and device's input timing requirement.

5.3.6.48 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, 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 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

NOTE

This register is for SMHC2 only.

5.3.6.49 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

Figures

Tables

6 Video Output Interfaces

6.1 TCON_LCD

6.1.1 Overview

The TCON_LCD(Timing Controller) module is used for LCD. The TCON_LCD module includes the following features:

- RGB interface to MIPI DSI, up to 1920x1080@60fps
- Supports i8080 interface, up to 800 x 480@60fps.
- Supports LVDS interface with single link, up to 1366 x 768@60fps
- Supports one channel MIPI DSI output, each MIPI DSI is 4-lane.
  - Single channel(4-lane) MIPI DSI with up to 1920 x 1080@60Hz
  - Compliant with MIPI DSI V1.0 and MIPI D-PHY V1.0.
  - Supports Video mode(Normal mode and Burst mode)
- Supports RGB888, RGB666 and RGB565 with dither function
- Supports Gamma correction with R/G/B channel independence

6.1.2 Block Diagram

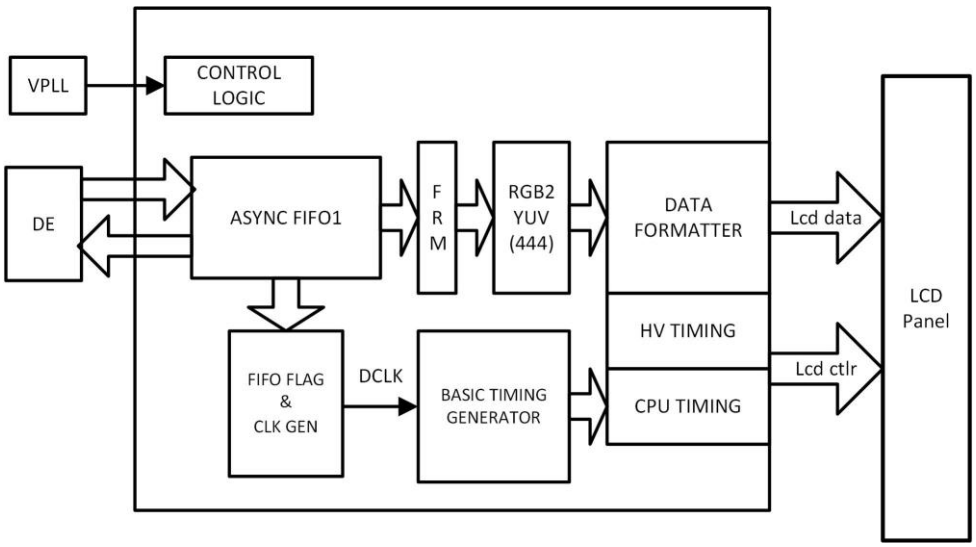


Figure6- 1. TCON_LCD Block Diagram

The TCON_LCD is a module that processes video signals by using a complicated arithmetic and then generates control signals.

6.1.3 Operations and Functional Descriptions

6.1.3.1 Control Signal and Data Port Mapping

	SYNC RGB					CPU cmd	CPU 18bit	CPU 16bit								CPU 8bit							CPU 9bit		LVDS		
I/O	Para RGB	Serial RGB			CCIR 656		256K	256K							65K	256K			65K				256K		Single Link		Dual Link
		1st	2nd	3rd				1st	2nd	3rd	1st	2nd	1st	2nd		1st	2nd	3rd	1st	2nd			1st	2nd	1	2	
IO0	VSYNC					CS																					
IO1	HSYNC					RD																					
IO2	DCLK					WR																				/	/
IO3	DE					RS																				/	/
D23	R7					D23	R5	R5	B5	G5	R5		R5	B5	R4											/	/
D22	R6					D22	R4	R4	B4	G4	R4		R4	B4	R3											/	/
D21	R5					D21	R3	R3	B3	G3	R3		R3	B3	R2											/	/
D20	R4					D20	R2	R2	B2	G2	R2		R2	B2	R1											/	/
D19	R3					D19	R1	R1	B1	G1	R1		R1	B1	R0											/	/
D18	R2					D18	R0	R0	B0	G0	R0		R0	B0	G5											/	/
D15	G7					D15	G5								G4											/	/
D14	G6					D14	G4								G3											/	/
D13	G5					D13	G3																		D3N1		/
D12	G4	D1_7	D2_7	D3_7	D7	D12	G2	G5	R5	B5	G5	B5	G5		G2	R5	G5	B5	R4	G2	R5	G2	D3P1		/		
D11	G3	D1_6	D2_6	D3_6	D6	D11	G1	G4	R4	B4	G4	B4	G4		G1	R4	G4	B4	R3	G1	R4	G1	CKN1		/		
D10	G2	D1_5	D2_5	D3_5	D5	D10	G0	G3	R3	B3	G3	B3	G3		G0	R3	G3	B3	R2	G0	R3	G0	CKP1		/		
D7	B7	D1_4	D2_4	D3_4	D4	D7	B5	G2	R2	B2	G2	B2	G2		B4	R2	G2	B2	R1	B4	R2	B5	D2N1		/		
D6	B6	D1_3	D2_3	D3_3	D3	D6	B4	G1	R1	B1	G1	B1	G1		B3	R1	G1	B1	R0	B3	R1	B4	D2P1		/		
D5	B5	D1_2	D2_2	D3_2	D2	D5	B3	G0	R0	B0	G0	B0	G0		B2	R0	G0	B0	G5	B2	R0	B3	D1N1		/		
D4	B4	D1_1	D2_1	D3_1	D1	D4	B2								B1				G4	B1	G5	B2	D1P1		/		
D3	B3	D1_0	D2_0	D3_0	D0	D3	B1								B0				G3	B0	G4	B1	D0N1		/		
D2	B2					D2	B0														G3	B0	D0P1		/		

6.1.3.2 External Signals

6.1.3.2.1 HV interface

HV I/F is also known as Sync + DE mode, which is widely used in TFT LCD module for PMP/MP4 application.

Table6- 1. HV Panel Signals

Signal	Description	Type
Vsync	Vertical sync, indicates one new frame	O
Hsync	Horizontal sync, indicate one new scan line	O
DCLK	Dot clock, pixel data are sync by this clock	O
LDE	LCD data enable	O
LD[23..0]	24bit RGB/YUV output from input FIFO for panel	O

The timing diagram of HV interface is as follows.

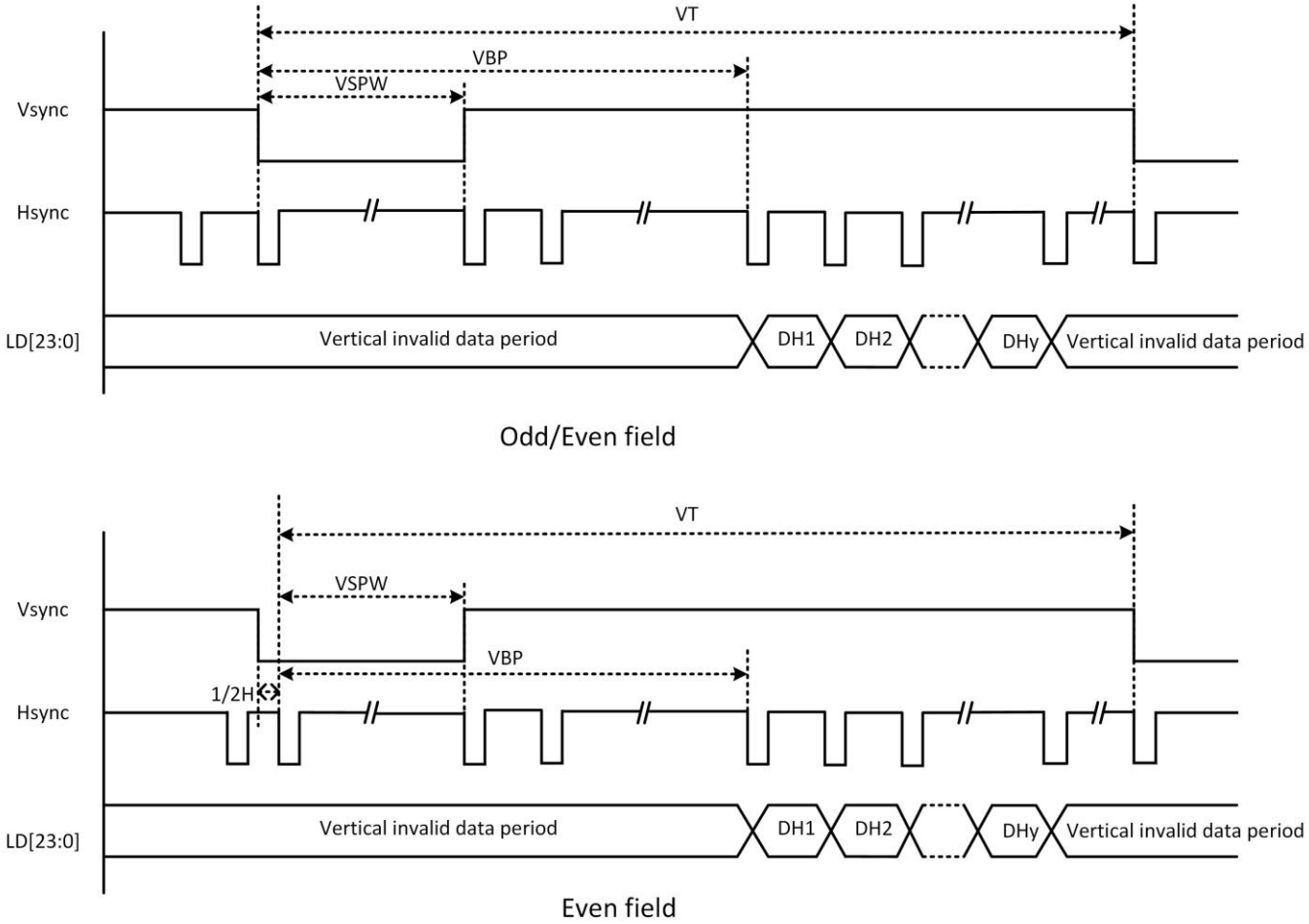


Figure6- 2. HV Interface Vertical Timing

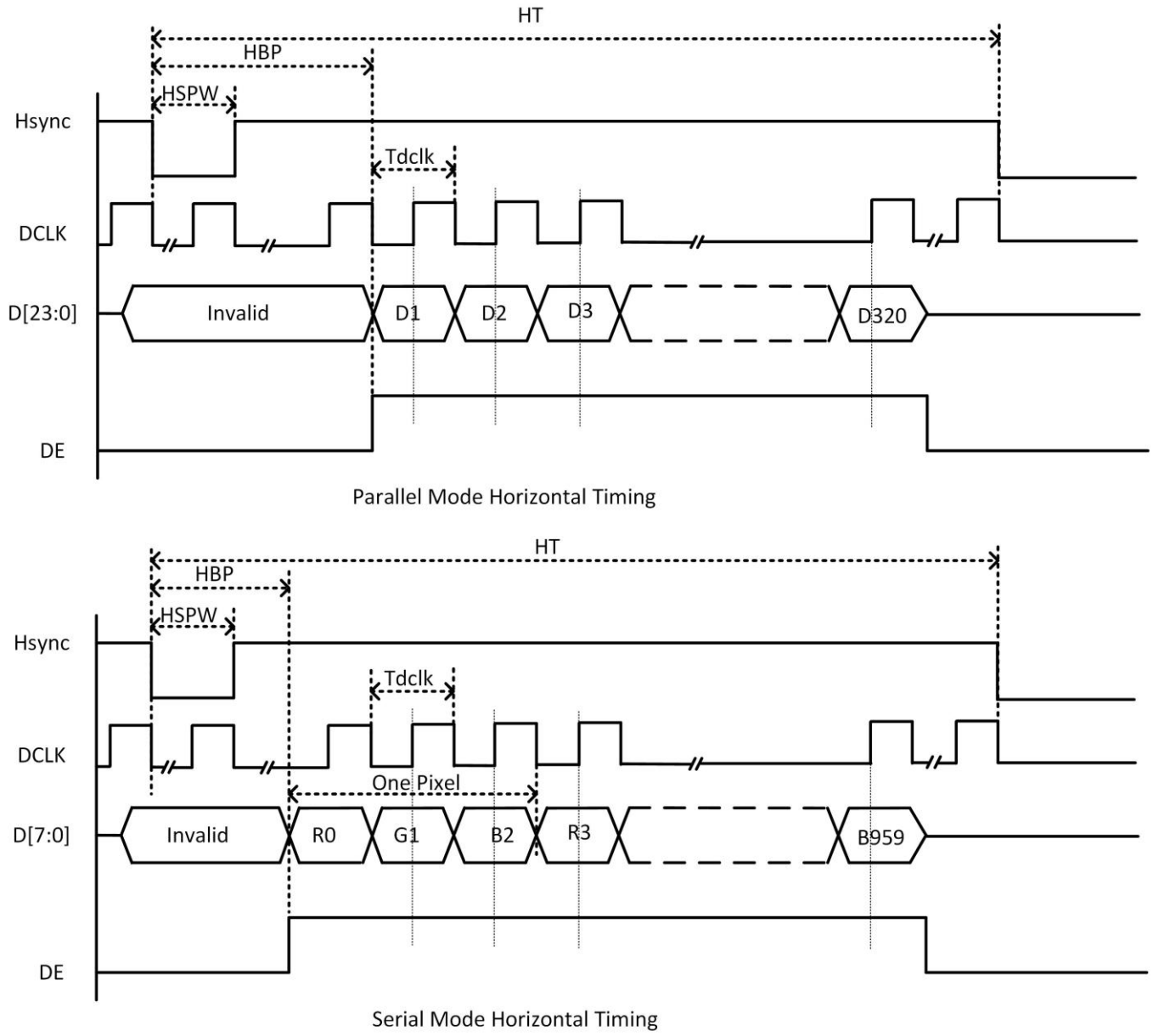


Figure6- 3. HV Interface Horizontal Timing

6.1.3.2.2 BT656 interface

When HV serial YUV output mode is in used, its timing is compatible with BT656. SAV add right before active area every line, EAV add right after active area every line.

Table6- 2. BT656 Panel Signals

Signal	Description	Type
DCLK	Dot clock, pixel data are sync by this clock	O
DATA[7:0]	Data	O

Its logic are:

F = “0” for Field 1 F = “1” for Field 2

V = “1” during vertical blanking

H = “0” at SAV H = “1” at EAV

P3-P0 = protection bits

P3 = V ⊕H

P2 = F ⊕H

P1 = F ⊕V

P0 = F ⊕V ⊕H

Where ⊕represents the exclusive-OR function

The 4 byte SAV/EAV sequences are:

Table6- 3. EAV and SAV Sequence

	8-bit Data								10-bit Data	
	D9 (MSB)	D8	D7	D6	D5	D4	D3	D2	D1	D0
preamble	1	1	1	1	1	1	1	1	1	1
	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0
Status word	1	F	V	H	P3	P2	P1	P0	0	0

6.1.3.2.3 i8080 interface

i8080 I/F LCD panel is most common interface for small size, low resolution LCD panels.
CPU control signals are active low.

Table6- 4. CPU Panel Signals

Main Signal	Description	Type
CS	Chip select, active low	O
WR	Write strobe, active low	O
RD	Read strobe, active low	O
A1	Address bit, controlled by "LCD_CPUI/F" BIT26/25	O
D[23..0]	Digital RGB output signal	I/O

The following figure relationship between basic timing and CPU timing. WR is 180 degree delay of DCLK; CS is active when pixel data are valid; RD is always set to 1; A1 are set by “LCD_CPUI/F”.

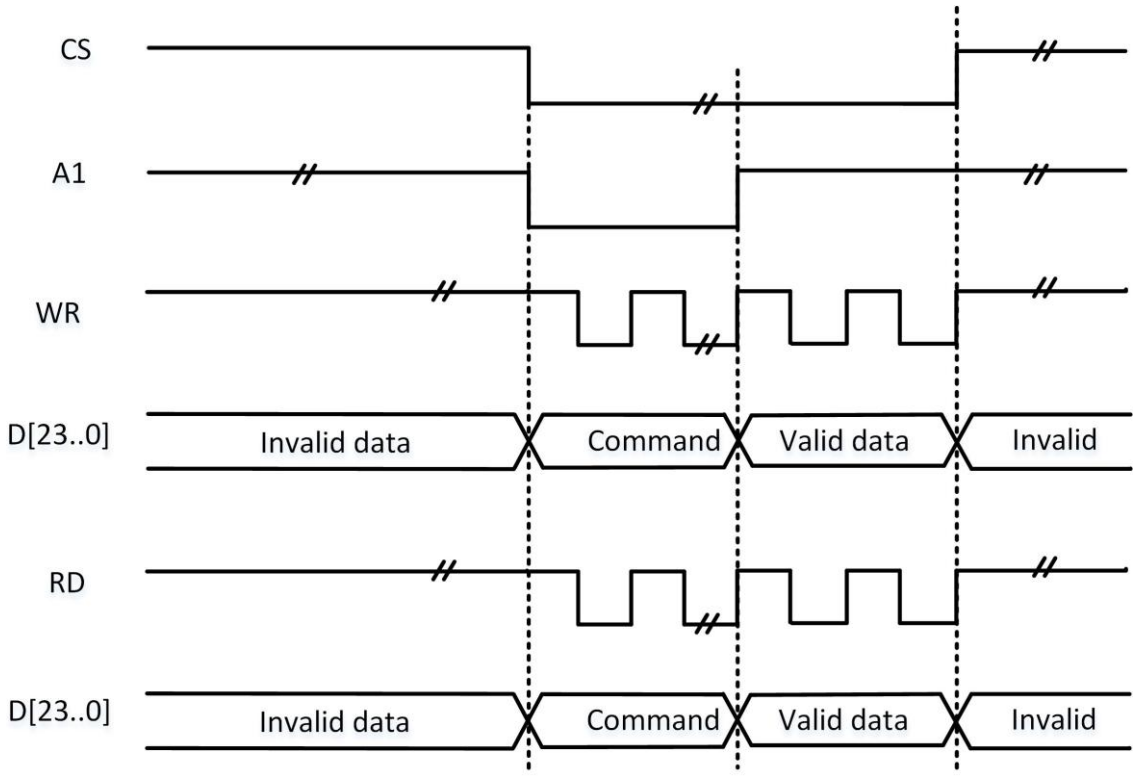


Figure6- 4. CPU interface timing

When CPU I/F is in IDLE state, it can generate WR/RD timing by setting "**Lcd_CPUI/F**". CS strobe is one DCLK width, WR/RD strobe is half DCLK width.

6.1.3.2.4 LVDS interface

Table6- 5. LVDS Panel Signals

Main Signal	Description	Type
CKP	The positive port of clock.	AO
CKN	The negative port of clock.	AO
D0P	The positive port of data channel 0.	AO
D0N	The negative port of data channel 0.	AO
D1P	The positive port of data channel 1.	AO
D1N	The negative port of data channel 1.	AO
D2P	The positive port of data channel 2.	AO
D2N	The negative port of data channel 2.	AO
D3P	The positive port of data channel 3.	AO
D3N	The negative port of data channel 3.	AO

JEDIA mode:

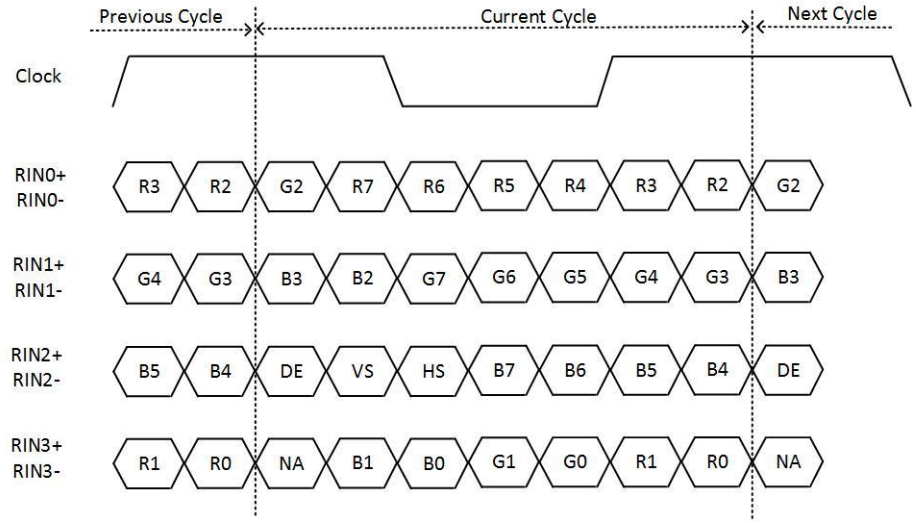


Figure6- 5. LVDS single link JEDIA Mode interface timing

NS mode:

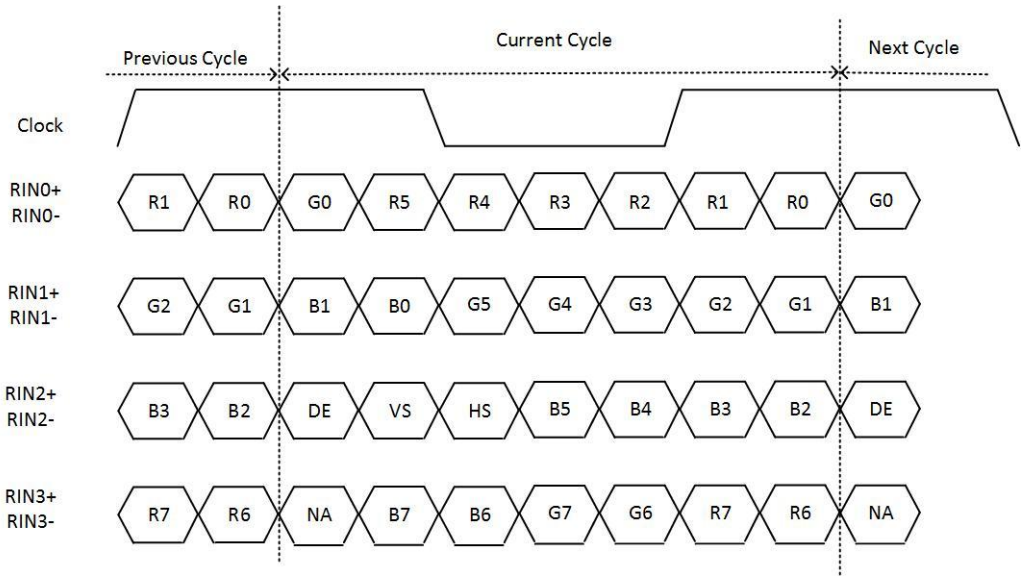


Figure6- 6. LVDS signal link NS Mode interface timing

6.1.3.3 Clock Sources

Table6-6 describes clock sources of TCON LCD.

Table6- 6. TCON LCD Clock Sources

Clock Sources	Description
PLL_VIDEO0(1X)	Video PLL Clock,default value is 297MHz
PLL_VIDEO0(4X)	Video PLL Clock,default value is 1188MHz

6.1.3.4 RGB Gamma Correction

Function: This module correct the RGB input data of DE.

A 256*8*3 Byte register file is used to store the gamma table. The following is the layout.

Table6- 7. RGB Gamma Correction Table

Offset	Value
0x400, 0x401, 0x402	{ B0[7:0], G0[7:0], R0[7:0] }
0x404,	{ B1[7:0], G1[7:0], R1[7:0] }
.....	.....
0x7FC	{ B255[7:0], G255[7:0], R255[7:0] }

6.1.3.5 CEU Module

Function: This module enhance color data from DE.

R' = ((Rr*R + Rg*G + Rb*B + 16)/16 + Rc + 16)/16

G' = ((Gr*R + Gg*G + Gb*B + 16)/16 + Gc + 16)/16

B' = ((Br*R + Bg*G + Bb*B + 16)/16 + Bc + 16)/16

NOTE

Rr, Rg, Rb, ,Gr, Gg, Gb, Br, Bg, Bb s13 (-16,16)

Rc, Gc, Bc s19 (-16384, 16384)

R, G, B u8 [0-255]

R' have the range of [Rmin ,Rmax]

G' have the range of [Rmin ,Rmax]

B' have the range of [Rmin ,Rmax]

6.1.3.6 CMAP Module

Function: This module map color data from DE.

Every 4 input pixels as an unit. an unit is divided into 12 bytes. Output byte can select one of those 12 bytes.Note that even line and odd line can be different, and output can be 12 bytes(4 pixels) or reduce to 6 bytes(2 pixels).

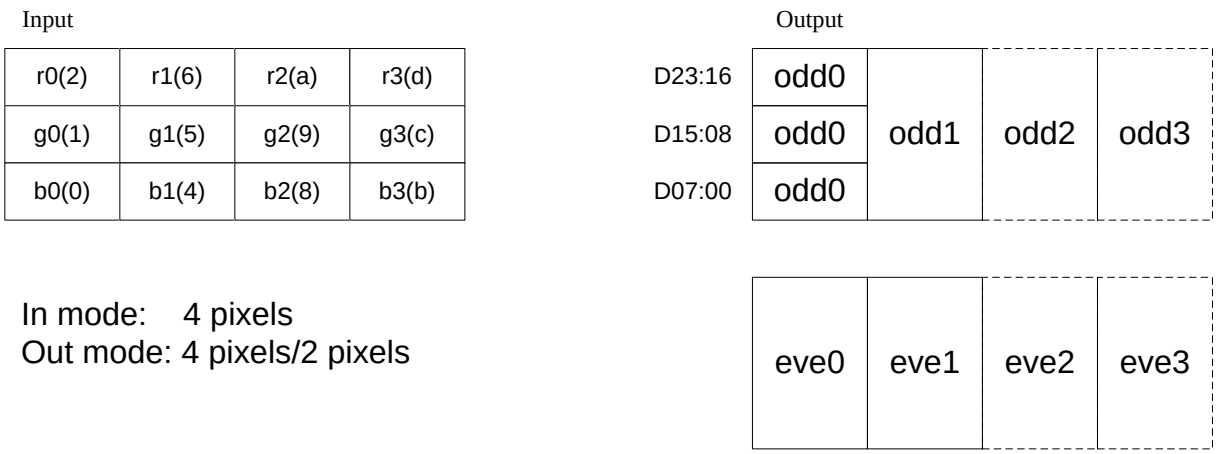


Figure6- 7. CMAP Module

6.1.3.7 FRM module

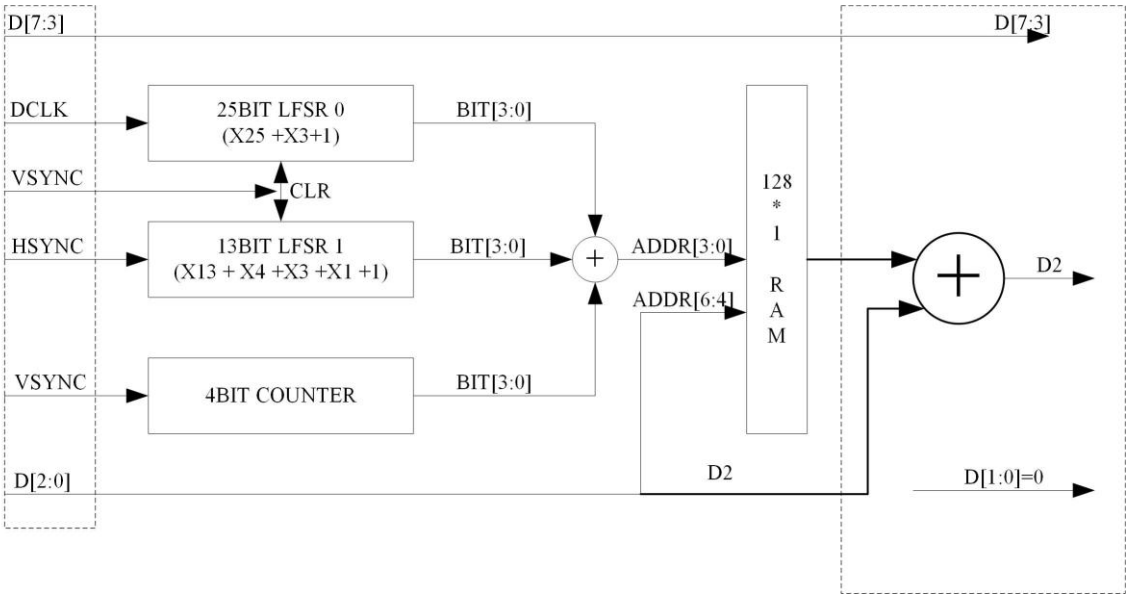


Figure6- 8. FRM module

6.1.4 Programming Guidelines

6.1.4.1 HV Mode Configuration Process

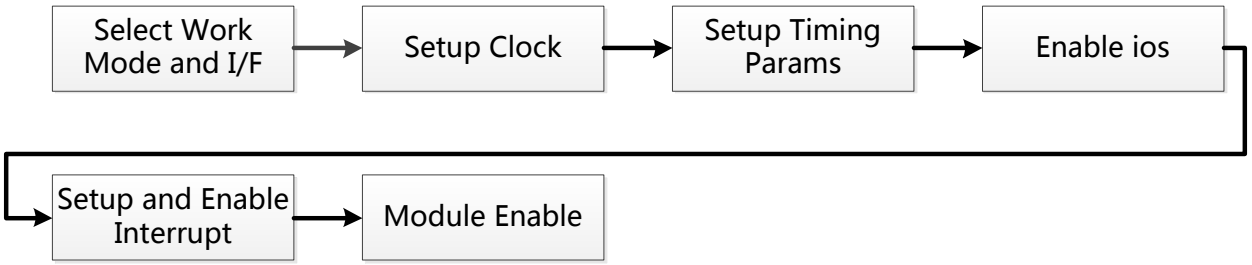


Figure6- 9. HV Mode Initialization

Step1: Select HV interface type: parallel RGB or serial RGB

Step2: Set Clock, If phase changing function need be used, then the bit[31:28] of LCD_DCLK_REG should be set to 0xF.

Step3: Set timing parameter x, ht, hbp, hspw, y, vt, vbp, vspw. Note that hbp includes hspw, vbp includes vspw. And vt need be set to twice as actual value.

Step4: Open io output.

Step5: Set and open interrupt function. Note that when using line interrupt, LCD_LINE_INT_NUM of LCD_GINT1_REG need be set first, then LCD_LINE_INT of LCD_GINT0_REG is set to 1.

Step6: Open mudule enable.

6.1.4.2 LVDS Configuration Process

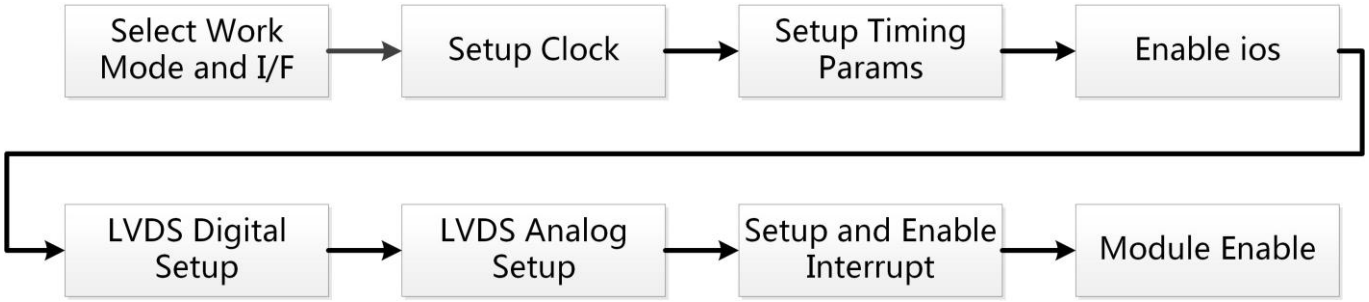


Figure6- 10. LVDS Mode Initial Process

Step closes to HV mode, the Step1 is to select parallel RGB interface type, and to increase LVDS digital part and analog part configurations. Select digital control part before configuring LVDS analog part, then this circuit can determine the routing automatically, such as using TCON_LCD to drive dual link LVDS, should first configure LVDS_EN and dual link choice , then analog setting.

LVDS Digital Setup: It should choose single link or dual link mode, NS mode or JEDIA mode, choose an LVDS clock source, note that when choosing MIPI PLL as clock source of LVDS, the clock source of TCON_LCD also needs to be MIPI PLL.

LVDS Analog Setup: Set analog parameters of LVDS, open LVDS PHY power, set up the common mode voltage and differential voltage amplitude.

6.1.4.3 i8080 Configuration Process

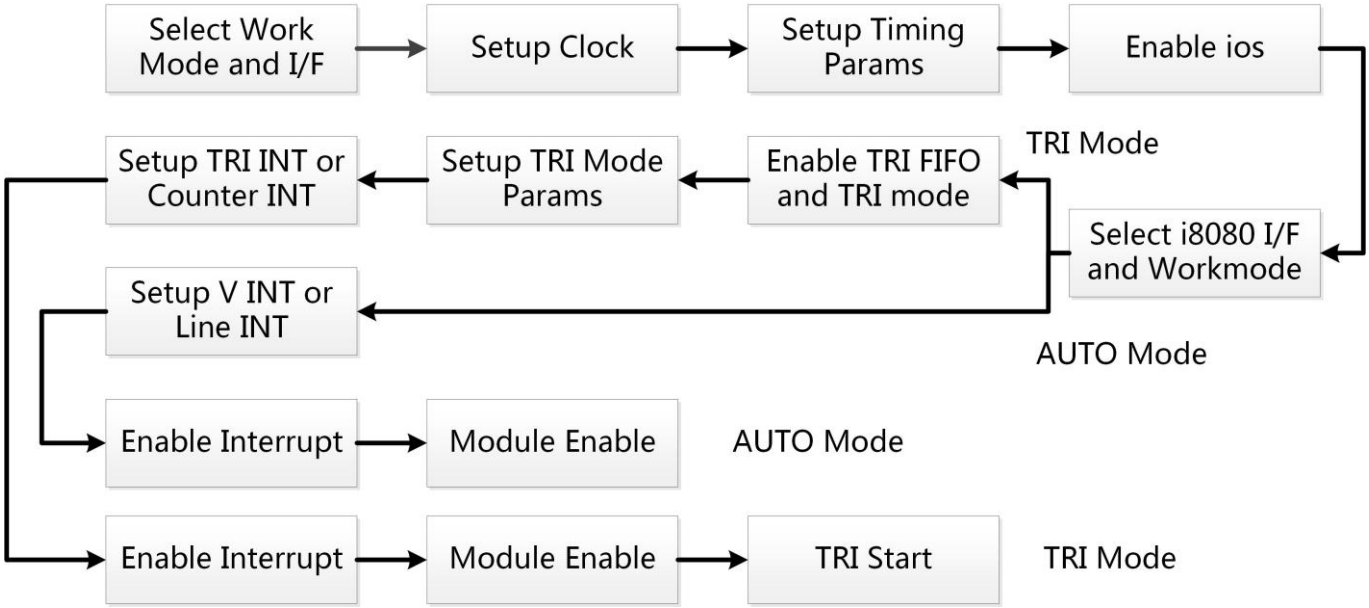


Figure6- 11. i8080 Mode Initial Process

Step1: Select i8080 interface type.

Step2: The step is same as HV mode, but phase changing function is invalid.

Step3: The step is same as HV mode. When using TRI mode, it is best to configure LCD timing parameters in HV mode, or handful of functions such as CMAP will not be able to apply.

Step4: The step is same as HV mode.

Step: Select i8080 interface type and operating mode, the operating mode includes TRI mode and AUTO mode, and the two operating modes are diffsrent.

----------TRI mode----------

Step6: Open TRI FIFO switch, and TRI mode function.
Step7: Set parameters of TRI mode, including block size,block space and block number.

NOTE

When output interface is parallel mode,then the setting value of block space parameter is not less than 20.
When output interface is 2 cycle serial mode,then the setting value of block space parameter is not less than 40.
When output interface is 3 cycle serial mode,then the setting value of block space parameter is not less than 60.
When output interface is 4 cycle DummyRGB mode,then the setting value of block space parameter is not less than 80.
Step8: Set the tri interrupt or counter interrupt. When using the two interrupts, mainly in the interrupt service function the tri start operation need be operated (the bit1 of LCD_CPU_IF_REG is set to"1"). If using TE trigger interrupt, you select the external input pin as a trigger signal, the 24-bit for offset 8c register is set to "1", to open up input of pad.
Step9: Open interrupt total switch.
Step10: Open interrupt total enable.
Step11: Operate tri start operation(the bit1 of LCD_CPU_IF_REG is set to"1")

----------Auto mode----------

Step6: Set and open V interrupt or Line interrupt, the step is the same as HV mode.
Step7： Open module total enable.

6.1.4.4 Notes of TCON_LCD Enable and Disable Sequence

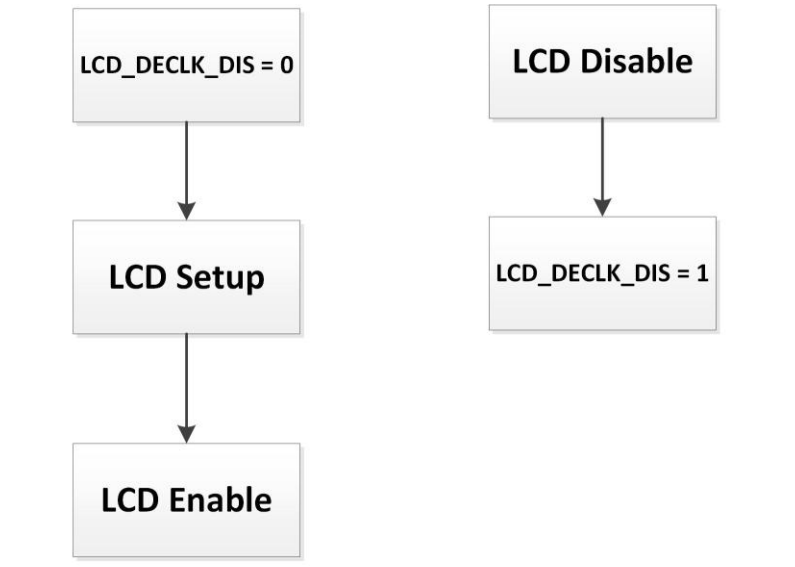


Figure6- 12. TCON_LCD Enable and Disbale Sequence

Bit[0](LCD_DECLK_DIS) of LCD_GCTL_REG is used to control the DE clock switch of TCON_LCD. The default value is ‘1’, which indicates closing the DE clock of TCON_LCD.
In the process of starting or initial TCON_LCD, LCD_DECLK_DIS need be set to ‘0’, Put the DE clock, and then initial the register of TCON_LCD, Start up TCON_LCD at last.
In the process of closing TCON_LCD, close TCON_LCD(all enable of TCON_LCD need be set to ‘0’.). And lastly, LCD_DECLK_DIS is set to ‘1’, close the DE clock.

6.1.5 Register List

Module Name	Base Address
TCON_LCD0	0x06511000

Register Name	Offset	Description
LCD_GCTL_REG	0x000	LCD Global Control Register
LCD_GINT0_REG	0x004	LCD Global Interrupt Register0
LCD_GINT1_REG	0x008	LCD Global Interrupt Register1
LCD_FRM_CTL_REG	0x010	LCD FRM Control Register
LCD_FRM_SEED_REG	0x014+N*0x04	LCD FRM Seed Register(N=0,1,2,3,4,5)
LCD_FRM_TAB_REG	0x02C+N*0x04	LCD FRM Table Register(N=0,1,2,3)
LCD_3D_FIFO_REG	0x03C	LCD 3D FIFO Register
LCD_CTL_REG	0x040	LCD Control Register
LCD_DCLK_REG	0x044	LCD Data Clock Register
LCD_BASIC0_REG	0x048	LCD Basic Timing Register0
LCD_BASIC1_REG	0x04C	LCD Basic Timing Register1
LCD_BASIC2_REG	0x050	LCD Basic Timing Register2
LCD_BASIC3_REG	0x054	LCD Basic Timing Register3
LCD_HV_IF_REG	0x058	LCD HV Panel Interface Register
LCD_CPU_IF_REG	0x060	LCD CPU Panel Interface Register
LCD_CPU_WR_REG	0x064	LCD CPU Panel Write Data Register
LCD_CPU_RD0_REG	0x068	LCD CPU Panel Read Data Register0
LCD_CPU_RD1_REG	0x06C	LCD CPU Panel Read Data Register1
LCD_PORCH_DATA_CTL_REG	0x070	LCD PORCH Data Control register
LCD_PORCH_DATA_REG	0x074	LCD PORCH Data register
LCD_LVDS_IF_REG	0x084	LCD LVDS IF Register
LCD_IO_POL_REG	0x088	LCD IO Polarity Register
LCD_IO_TRI_REG	0x08C	LCD IO Control Register
LCD_DEBUG_REG	0x0FC	LCD Debug Register
LCD_CEU_CTL_REG	0x100	LCD CEU Control Register
LCD_CEU_COEF_MUL_REG	0x110+N*0x04	LCD CEU Coefficient Register0(N=0,1,2,4,5,6,8,9,10)
LCD_CEU_COEF_ADD_REG	0x11C+N*0x10	LCD CEU Coefficient Register1(N=0,1,2)
LCD_CEU_COEF_RANG_REG	0x140+N*0x04	LCD CEU Coefficient Register2(N=0,1,2)
LCD_CPU_TRI0_REG	0x160	LCD CPU Panel Trigger Register0
LCD_CPU_TRI1_REG	0x164	LCD CPU Panel Trigger Register1
LCD_CPU_TRI2_REG	0x168	LCD CPU Panel Trigger Register2
LCD_CPU_TRI3_REG	0x16C	LCD CPU Panel Trigger Register3
LCD_CPU_TRI4_REG	0x170	LCD CPU Panel Trigger Register4
LCD_CPU_TRI5_REG	0x174	LCD CPU Panel Trigger Register5
LCD_CMAP_CTL_REG	0x180	LCD Color Map Control Register
LCD_CMAP_ODD0_REG	0x190	LCD Color Map Odd Line Register0
LCD_CMAP_ODD1_REG	0x194	LCD Color Map Odd Line Register1
LCD_CMAP_EVEN0_REG	0x198	LCD Color Map Even Line Register0

LCD_CMAP_EVEN1_REG	0x19C	LCD Color Map Even Line Register1
LCD_SAFE_PERIOD_REG	0x1F0	LCD Safe Period Register
LCD_LVDS0_ANA_REG	0x0220	LCD LVDS0 Analog Register

6.1.6 Register Description

6.1.6.1 LCD_GCTL_REG(Default Value: 0x0000_0000)

Offset: 0x000			Register Name: LCD_GCTL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	LCD_EN 0: disable 1: enable When it is disabled, the module will be reset to idle state.
30	R/W	0x0	LCD_GAMMA_EN 0: disable 1: enable
29:1	/	/	/
0	R/W	0x1	LCD_DECLK_DIS 0: Pass 1: Mask Pass the DE clock into TCON_LCD.

6.1.6.2 LCD_GINT0_REG(Default Value: 0x0000_0000)

Offset: 0x004			Register Name: LCD_GINT0_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	LCD_VB_INT_EN 0: disable 1: enable Enable the VB interrupt
30	/	/	/
29	R/W	0x0	LCD_LINE_INT_EN 0: disable 1: enable Enable the line interrupt.
28	/	/	/
27	R/W	0x0	LCD_TRI_FINISH_INT_EN 0: Disable 1: Enable Enable the trigger finish interrupt.
26:	R/W	0x0	LCD_TRI_COUNTER_INT_EN 0: Disable 1: Enable

			Enable the trigger counter interrupt.
25:16	/	/	/
15	R/W0C	0x0	LCD_VB_INT_FLAG Asserted during vertical no-display period every frame. Write 0 to clear it.
14	/	/	/
13	R/W0C	0x0	LCD_LINE_INT_FLAG Trigger when SY0 match the current LCD scan line Write 0 to clear it.
12	/	/	/
11	R/W0C	0x0	LCD_TRI_FINISH_INT_FLAG Trigger when cpu trigger mode finish. Write 0 to clear it.
10	R/W0C	0x0	LCD_TRI_COUNTER_INT_FLAG Trigger when tri counter reache this value Write 0 to clear it.
9	R/W0C	0x0	LCD_TRI_UNDERFLOW_FLAG Only used in dsi video mode, tri when sync by dsi but not finish Write 0 to clear it.
8:0	/	/	/

6.1.6.3 LCD_GINT1_REG(Default Value: 0x0000_0000)

Offset: 0x008			**Register Name: LCD_GINT1_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	/	/	/
27:16	R/W	0x0	LCD_LINE_INT_NUM Scan line for LCD line trigger(including inactive lines) Setting it for the specified line for trigger0. **NOTE** **SY0 is writable only when LINE_TRG0 disable.**
15:0	/	/	/

6.1.6.4 LCD_FRM_CTL_REG(Default Value: 0x0000_0000)

Offset: 0x010			**Register Name: LCD_FRM_CTL_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	LCD_FRM_EN 0:disable 1:enable
30:7	/	/	/
6	R/W	0x0	LCD_FRM_MODE_R 0: 6-bit frm output

			1: 5-bit frm output The R component output bits in dither function.
5	R/W	0x0	LCD_FRM_MODE_G 0: 6-bit frm output 1: 5-bit frm output The G component output bits in dither function.
4	R/W	0x0	LCD_FRM_MODE_B 0: 6-bit frm output 1: 5-bit frm output The B component output bits in dither function.
3:2	/	/	/
1:0	R/W	0x0	LCD_FRM_TEST 00: FRM 01: half 5-/6-bit, half FRM 10: half 8-bit, half FRM 11: half 8-bit, half 5-/6-bit Set the test mode of dither function.

6.1.6.5 LCD_FRM_SEED_REG(Default Value: 0x0000_0000)

Offset: 0x014+N*0x04(N=0,1,2,3,4,5)			Register Name: LCD_FRM_SEED_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:25	/	/	/
24:0	R/W	0x0	SEED_VALUE N=0: Pixel_Seed_R N=1: Pixel_Seed_G N=2: Pixel_Seed_B N=3: Line_Seed_R N=4: Line_Seed_G N=5: Line_Seed_B Avoid setting it to 0. Set the seed used in dither function.

6.1.6.6 LCD_FRM_TAB_REG(Default Value: 0x0000_0000)

Offset: 0x02C+N*0x04(N=0,1,2,3)			Register Name: LCD_FRM_TAB_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	FRM_TABLE_VALUE Set the data used in dither function. Usually set as follow: Table0 = 0x01010000 Table1 = 0x15151111 Table2 = 0x57575555 Table3 = 0x7f7f7777

6.1.6.7 LCD_3D_FIFO_REG(Default Value: 0x0000_0000)

Offset: 0x03C			Register Name: LCD_3D_FIFO_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	3D_FIFO_BIST_EN 0: disable 1: enable Enable the 3d fifo bist test function.
30:14	/	/	/
13:4	R/W	0x0	3D_FIFO_HALF_LINE_SIZE The number of data in half line=3D_FIFO_HALF_LINE_SIZE+1. only valid when 3D_FIFO_SETTING set as 2
3:2	/	/	/
1:0	R/W	0x0	3D_FIFO_SETTING 00: by pass 01: used as normal FIFO 10: used as 3D interlace FIFO 11: reserved Set the work mode of 3D FIFO.

6.1.6.8 LCD_CTL_REG(Default Value: 0x0000_0000)

Offset: 0x040			Register Name: LCD_CTL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	LCD_EN 0: disable 1: enable It executes at the beginning of the first blank line of LCD timing.
30:26	/	/	/
25:24	R/W	0x0	LCD_IF 00: HV(Sync+DE) 01: 8080 I/F 1x:reservd Set the interface type of LCD controller.
23	R/W	0x0	LCD_RB_SWAP 0: default 1: swap RED and BLUE data at FIFO1 Enable the function to swap red data and blue data in fifo1.
22	/	/	/
21	R/W	0x0	LCD_FIFO1_RST Write 1 and then 0 at this bit will reset FIFO 1 1 holding time must more than 1 DCLK
20	R/W	0x0	LCD_INTERLACE_EN

			0:disable 1:enable This flag is valid only when LCD_EN == 1
19:9	/	/	/
8:4	R/W	0x0	LCD_START_DELAY STA delay Valid only when LCD_EN == 1
3	/	/	/
2:0	R/W	0x0	LCD_SRC_SEL 000: DE 001: Color Check 010: Grayscale Check 011: Black by White Check 100: Test Data all 0 101: Test Data all 1 110: Reserved 111: Gridding Check

6.1.6.9 LCD_DCLK REG(Default Value: 0x0000_0000)

Offset: 0x044			Register Name: LCD_DCLK REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x0	LCD_DCLK_EN 0000: dclk_en = 0; dclk1_en = 0; dclk2_en = 0; dclkm2_en = 0; 0001: dclk_en = 1; dclk1_en = 0; dclk2_en = 0; dclkm2_en = 0; 0010: dclk_en = 1; dclk1_en = 0; dclk2_en = 0; dclkm2_en = 1; 0011: dclk_en = 1; dclk1_en = 1; dclk2_en = 0; dclkm2_en = 0; 0101: dclk_en = 1; dclk1_en = 0; dclk2_en = 1; dclkm2_en = 0; 1111: dclk_en = 1; dclk1_en = 1; dclk2_en = 1; dclkm2_en = 1; Others:Reversed LCD clock enable.
27:7	/	/	/
6:0	R/W	0x0	LCD_DCLK_DIV Tdclk = Tsclk * DCLKDIV If dclk1&dclk2 used, DCLKDIV >=6 If dclk only, DCLKDIV >=1

6.1.6.10 LCD_BASIC0_REG(Default Value: 0x0000_0000)

Offset: 0x048			Register Name: LCD_BASIC0_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	/	/	/
27:16	R/W	0x0	X Panel width is X+1

15:12	/	/	/
11:0	R/W	0x0	Y Panel height is Y+1

6.1.6.11 LCD_BASIC1_REG(Default Value: 0x0000_0000)

Offset: 0x04C			Register Name: LCD_BASIC1_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	/	/	/
28:16	R/W	0x0	HT Thcycle = (HT+1) * Tdclk Computation 1) parallel:HT = X + BLANK Limitation: 1) parallel :HT >= (HBP +1) + (X+1) +2 2) serial 1: HT >= (HBP +1) + (X+1) *3+2 3) serial 2: HT >= (HBP +1) + (X+1) *3/2+2
15:12	/	/	/
11:0	R/W	0x0	HBP horizontal back porch (in dclk) Thbp = (HBP +1) * Tdclk

6.1.6.12 LCD_BASIC2_REG(Default Value: 0x0000_0000)

Offset: 0x050			Register Name: LCD_BASIC2_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	/	/	/
28:16	R/W	0x0	VT TVT = (VT)/2 * Thsync VT/2 >= (VBP+1) + (Y+1) +2
15:12	/	/	/
11:0	R/W	0x0	VBP Tvbp = (VBP +1) * Thsync

6.1.6.13 LCD_BASIC3_REG(Default Value: 0x0000_0000)

Offset: 0x054			Register Name: LCD_BASIC3_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:26	/	/	/
25:16	R/W	0x0	HSPW Thspw = (HSPW+1) * Tdclk HT> (HSPW+1)
15:10	/	/	/

9:0	R/W	0x0	VSPW Tvspw = (VSPW+1) * Thsync VT/2 > (VSPW+1)

6.1.6.14 LCD_HV_IF_REG(Default Value: 0x0000_0000)

Offset: 0x058			Register Name: LCD_HV_IF_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x0	HV_MODE 0000: 24-bit/1cycle parallel mode 1000: 8bit/3cycle RGB serial mode(RGB888) 1010: 8bit/4cycle Dummy RGB(DRGB) 1011: 8bit/4cycle RGB Dummy(RGBD) 1100: 8bit/2cycle YUV serial mode(CCIR656) Set the HV mode of LCD controller.
27:26	R/W	0x0	RGB888_ODD_ORDER 00: R→G→B 01: B→R→G 10: G→B→R 11: R→G→B Serial RGB888 mode Output sequence at odd lines of the panel (line 1, 3, 5, 7…).
25:24	R/W	0x0	RGB888_EVEN_ORDER 00: R→G→B 01: B→R→G 10: G→B→R 11: R→G→B Serial RGB888 mode Output sequence at even lines of the panel (line 2, 4, 6, 8…).
23:22	R/W	0x0	YUV_SM 00: YUYV 01: YVYU 10: UYVY 11: VYUY Serial YUV mode Output sequence 2-pixel-pair of every scan line.
21:20	R/W	0x0	YUV EAV/SAV F LINE DELAY 00:F toggle right after active video line 01:delay 2 line(CCIR PAL) 10:delay 3 line(CCIR NTSC) 11:reserved Set the delay line mode.
19	R/W	0x0	CCIR_CSC_DIS 0: Enable 1: Disable Only valid when HV mode is “1100”.

			Select '0' LCD convert source from RGB to YUV.
18:0	/	/	/

6.1.6.15 LCD_CPU_IF_REG(Default Value: 0x0000_0000)

Offset: 0x060			Register Name: LCD_CPU_IF_REG
Bit	Read/Write	Default/Hex	Description
31:28	R/W	0x0	CPU_MODE 0000: 18bit/256K mode 0010: 16bit mode0 0100: 16bit mode1 0110: 16bit mode2 1000: 16bit mode3 1010: 9bit mode 1100: 8bit 256K mode 1110: 8bit 65K mode xxx1: 24bit for DSI Set the i8080 interface work mode.
27	/	/	/
26	R/W	0x0	DA Pin A1 value in 8080 mode auto/flash states
25	R/W	0x0	CA Pin A1 value in 8080 mode WR/RD execute
24	/	/	/
23	R	0x0	WR_FLAG 0:Write operation is finishing 1:Write operation is pending The status of write operation.
22	R	0x0	RD_FLAG 0:Read operation is finishing 1:Read operation is pending The status of read operation.
21:18	/	/	/
17	R/W	0x0	AUTO Auto Transfer Mode: If it's 1, all the valid data during this frame are write to panel. Note: This bit is sampled by Vsync
16	R/W	0x0	FLUSH Direct transfer mode: If it's enabled, FIFO1 is regardless of the HV timing, pixels data keep being transferred unless the input FIFO was empty. Data output rate control by DCLK.
15:4	/	/	/
3	R/W	0x0	TRI_FIFO_BIST_EN 0: Disable

			1: Enable Entry addr is 0xFF8.
2	R/W	0x0	TRI_FIFO_EN 0:Disable 1:Enable Enable the trigger FIFO.
1	R/W1S	0x0	TRI_START Write '1' to start a frame flush, write'0' has no effect. This flag indicated frame flush is running. Software must make sure write '1' only when this flag is '0'.
0	R/W	0x0	TRI_EN 0: Trigger mode disable 1: Trigger mode enable Enable trigger mode.

6.1.6.16 LCD_CPU_WR_REG(Default Value: 0x0000_0000)

Offset: 0x064			Register Name: LCD_CPU_WR_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:0	W	0x0	DATA_WR Data write on 8080 bus, launch a write operation on 8080 bus.

6.1.6.17 LCD_CPU_RD0_REG(Default Value: 0x0000_0000)

Offset: 0x068			Register Name: LCD_CPU_RD0_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:0	R	0x0	DATA_RD0 Data read on 8080 bus, launch a new read operation on 8080 bus.

6.1.6.18 LCD_CPU_RD1_REG(Default Value: 0x0000_0000)

Offset: 0x06C			Register Name: LCD_CPU_RD1_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:0	R	0x0	DATA_RD1 Data read on 8080 bus, without a new read operation on 8080 bus.

6.1.6.19 LCD_PORCH_DATA_CTL_REG(Default Value: 0x0000_0000)

Offset: 0x070	Register Name: LCD_PORCH_DATA_CTL_REG

Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:0	R/W	0x0	LCD_PORCH_DATA_CTL 0: Keep the last signal state in active area 1: Set the signal state refer to LCD_PORCH_DATA_REG These control bits are used to control the TCON_LCD data signal states in porch area .One bit for one signal.

6.1.6.20 LCD_PORCH_DATA_REG (Default Value: 0x0000_0000)

Offset: 0x074			Register Name: LCD_PORCH_DATA_REG
Bit	Read/Write	Default/Hex	Description
31:24	/	/	/
23:0	R/W	0x0	LCD_PORCH_DATA 0: Output “0” 1: Output “1” These bits are used to set the TCON_LCD data signal values in porch area .One bit for one signal.These bits are valid when control bits in LCD_PORCH_DATA_CTL set as “1”.

6.1.6.21 LCD_LVDS_IF_REG(Default Value: 0x0000_0000)

Offset: 0x084			Register Name: LCD_LVDS_IF_REG
Bit	Read/Write	Default/Hex	Description
31	R/W	0	LCD_LVDS_EN 0: Disable 1: Enable Enable LVDS interface.
30	R/W	0	LCD_LVDS_LINK 0:Single link 1:Dual link Select work in single link mode or dual link mode.
29	R/W	0	LCD_LVDS_EVEN_ODD_DIR 0: Normal 1: Reverse Set the order of even field and odd field.
28	R/W	0	LCD_LVDS_DIR 1: Normal 2: Reverse Set the LVDS direction.
27	R/W	0	LCD_LVDS_MODE 0: NS mode 1: JEIDA mode Set the LVDS data mode.

26	R/W	0	LCD_LVDS_BITWIDTH 0: 24bit 1: 18bit Set the bit width of data.
25	R/W	0	LCD_LVDS_DEBUG_EN 0: Disable 1: Enable Enable LVDS debug function.
24	R/W	0	LCD_LVDS_DEBUG_MODE 0: Mode0 Random data 1: Mode1 Output CLK period=7/2 LVDS CLK period Set the output signal in debug mode.
23	R/W	0	LCD_LVDS_CORRECT_MODE 0: Mode0 1: Mode1 Set the LVDS correct mode.
22:21	/	/	/
20	R/W	0	LCD_LVDS_CLK_SEL 0: Reserved 1: LCD CLK Select the clock source of LVDS.
19:5	/	/	/
4	R/W	0	LCD_LVDS_CLK_POL 0: Reverse 1: Normal Set the clock polarity of LVDS.
3:0	R/W	0	LCD_LVDS_DATA_POL 0: Reverse 1: Normal Set the data polarity of LVDS.

6.1.6.22 LCD_IO_POL_REG(Default Value: 0x0000_0000)

Offset: 0x088			**Register Name: LCD_IO_POL_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	IO_OUTPUT_SEL 0: normal output 1: register output when set as '1', d[23:0], io0, io1,io3 sync to dclk
30:28	R/W	0x0	DCLK_SEL 000: used DCLK0(normal phase offset) 001: used DCLK1(1/3 phase offset) 010: used DCLK2(2/3 phase offset) 101: DCLK0/2 phase 0 100: DCLK0/2 phase 90

			Others: Reserved
27	R/W	0x0	IO3_INV 0: not invert 1: invert
26	R/W	0x0	IO2_ INV 0: not invert 1: invert
25	R/W	0x0	IO1_INV 0: not invert 1: invert
24	R/W	0x0	IO0_INV 0: not invert 1: invert
23:0	R/W	0x0	DATA_INV LCD output port D[23:0] polarity control, with independent bit control: 0s: normal polarity 1s: invert the specify output

6.1.6.23 LCD_IO_TRI_REG(Default Value: 0x0FFF_FFFF)

Offset: 0x08C			**Register Name: LCD_IO_TRI_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	/	/	/
28	R/W	0x0	RGB_ENDIAN 0: normal 1: bits_invert
27	R/W	0x1	IO3_OUTPUT_TRI_EN 1: disable 0: enable
26	R/W	0x1	IO2_OUTPUT_TRI_EN 1: disable 0: enable
25	R/W	0x1	IO1_OUTPUT_TRI_EN 1: disable 0: enable
24	R/W	0x1	IO0_OUTPUT_TRI_EN 1: disable 0: enable
23:0	R/W	0xFFFFFF	DATA_OUTPUT_TRI_EN LCD output port D[23:0] output enable, with independent bit control: 1: disable 0: enable

6.1.6.24 LCD_DEBUG_REG(Default Value: 0x0000_0000)

Offset: 0x0FC			Register Name: LCD_DEBUG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R	0x0	LCD_FIFO_UNDERFLOW 0: Not underflow 1: Underflow The flag shows whether the fifos in underflow status.
30	/	/	/
29	R	0x1	LCD_FIELD_POL 0: Second field 1: First field The flag indicates the current field polarity.
28	/	/	/
27:16	R	0x0	LCD_CURRENT_LINE The current scan line.
15:0	/	/	/

6.1.6.25 LCD_CEU_CTL_REG(Default Value: 0x0000_0000)

Offset: 0x100			Register Name: LCD_CEU_CTL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	CEU_EN 0: bypass 1: enable
30:0	/	/	/

6.1.6.26 LCD_CEU_COEF_MUL_REG(Default Value: 0x0000_0000)

Offset: 0x110+N*0x04(N=0,1,2,4,5,6,8,9,10)			Register Name: LCD_CEU_COEF_MUL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:13	/	/	/
12:0	R/W	0x0	CEU_COEF_MUL_VALUE signed 13bit value, range of (-16,16) N=0: Rr N=1: Rg N=2: Rb N=4: Gr N=5: Gg N=6: Gb N=8: Br N=9: Bg N=10: Bb

6.1.6.27 LCD_CEU_COEF_ADD_REG(Default Value: 0x0000_0000)

Offset: 0x11C+N*0x10(N=0,1,2)			Register Name: LCD_CEU_COEF_ADD_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:19	/	/	/
18:0	R/W	0x0	CEU_COEF_ADD_VALUE signed 19bit value, range of (-16384, 16384) N=0: Rc N=1: Gc N=2: Bc

6.1.6.28 LCD_CEU_COEF_RANG_REG(Default Value: 0x0000_0000)

Offset: 0x140+N*0x04(N=0,1,2)			Register Name: LCD_CEU_COEF_RANG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	/	/	/
23:16	R/W	0x0	CEU_COEF_RANGE_MIN unsigned 8bit value, range of [0,255]
15:8	/	/	/
7:0	R/W	0x0	CEU_COEF_RANGE_MAX unsigned 8bit value, range of [0,255]

6.1.6.29 LCD_CPU_TRI0_REG(Default Value: 0x0000_0000)

Offset: 0x160			Register Name: LCD_CPU_TRI0_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	/	/	/
27:16	R/W	0x0	BLOCK_SPACE The spaces between data blocks. It should be set >20*pixel.
15:12	/	/	/
11:0	R/W	0x0	BLOCK_SIZE The size of data block.It is usually set as X.

6.1.6.30 LCD_CPU_TRI1_REG(Default Value: 0x0000_0000)

Offset: 0x164			Register Name: LCD_CPU_TRI1_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	R	0x0	BLOCK_CURRENT_NUM Shows the current data block transmitting to panel.
15:0	R/W	0x0	BLOCK_NUM The number of data blocks.It is usually set as Y.

6.1.6.31 LCD_CPU_TRI2_REG(Default Value: 0x0000_0000)

Offset: 0x168			Register Name: LCD_CPU_TRI2_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	R/W	0x20	START_DLY T_{dly} = (Start_Delay +1) * be_clk*8.
15	R/W	0x0	TRANS_START_MODE 0: ECC_FIFO+TRI_FIFO 1: TRI_FIFO Select the FIFOs used in CPU mode.
14:13	R/W	0x0	SYNC_MODE 0x: Auto 10: 0 11: 1 Set the sync mode in CPU interface.
12:0	R/W	0x0	TRANS_START_SET Usual set as the length of a line.

6.1.6.32 LCD_CPU_TRI3_REG(Default Value: 0x0000_0000)

Offset: 0x16C			Register Name: LCD_CPU_TRI3_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/
29:28	R/W	0x0	TRI_INT_MODE 00: Disable 01: Counter mode 10: Te rising mode 11: Te falling mode When set as 01, Tri_Counter_Int occur in cycle of (Count_N+1)×(Count_M+1)×4 dclk. When set as 10 or 11, io0 is map as TE input.
27:24	/	/	/
23:8	R/W	0x0	COUNTER_N The value of counter factor.
7:0	R/W	0x0	COUNTER_M The value of counter factor.

6.1.6.33 LCD_CPU_TRI4_REG(Default Value: 0x0000_0000)

Offset: 0x170			Register Name: LCD_CPU_TRI4_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	/	/	/
28	R/W	0x0	PLUG_MODE_EN 0: disable

			1:enable Enable the plug mode used in dsi command mode.
27:25	/	/	/
24	R/W	0x0	A1 Valid in first Block
23:0	R/W	0x0	D23-D0 Valid in first Block

6.1.6.34 LCD_CPU_TRI5_REG(Default Value: 0x0000_0000)

Offset: 0x174			Register Name: LCD_CPU_TRI5_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:25	/	/	/
24	R/W	0x0	A1 Valid in Block except first
23:0	R/W	0x0	D23-D0 Valid in Block except first

6.1.6.35 LCD_CMAP_CTL_REG(Default Value: 0x0000_0000)

Offset: 0x180			Register Name: LCD_CMAP_CTL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	COLOR_MAP_EN 0: bypass 1: enable Enable the color map function. This module only work when X is divided by 4
30:1	/	/	/
0	R/W	0x0	OUT_FORMAT 0: 4 pixel output mode: Out0 -> Out1 -> Out2 -> Out3 1: 2 pixel output mode: Out0 -> Out1 Set the pixel output format in color map function.

6.1.6.36 LCD_CMAP_ODD0_REG(Default Value: 0x0000_0000)

Offset: 0x190			Register Name: LCD_CMAP_ODD0_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	R/W	0x0	OUT_ODD1 bit15-12: Reserved bit11-08: Out_Odd0[23:16] bit07-04: Out_Odd0[15:8] bit03-00: Out_Odd0[7:0] 0000: in_b0

			0001: in_g0 0010: in_r0 0011: Reserved 0100: in_b1 0101: in_g1 0110: in_r1 0111:Reserved 1000: in_b2 1001: in_g2 1010: in_r2 1011: Reserved 1100: in_b3 1101: in_g3 1110: in_r3 1111: Reserved Indicates the output order of components.
15:0	R/W	0x0	OUT_ODD0 bit15-12: Reservd bit11-08: Out_Odd0[23:16] bit07-04: Out_Odd0[15:8] bit03-00: Out_Odd0[7:0] 0x0: in_b0 0x1: in_g0 0x2: in_r0 0x3: reservd 0x4: in_b1 0x5: in_g1 0x6: in_r1 0x7: reservd 0x8: in_b2 0x9: in_g2 0xa: in_r2 0xb: reservd 0xc: in_b3 0xd: in_g3 0xe: in_r3 0xf: reservd

6.1.6.37 LCD_CMAP_ODD1_REG(Default Value: 0x0000_0000)

Offset: 0x194			Register Name: LCD_CMAP_ODD1_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	R/W	0x0	OUT_ODD3 bit15-12: Reserved bit11-08: Out_Odd0[23:16]

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

6.1.6.38 LCD_CMAP_EVEN0_REG(Default Value: 0x0000_0000)

Offset: 0x198	Register Name: LCD_CMAP_EVEN0_REG

Bit	Read/Write	Default/Hex	Description
31:16	R/W	0x0	OUT_EVEN1 bit15-12: Reserved bit11-08: Out_Odd0[23:16] bit07-04: Out_Odd0[15:8] bit03-00: Out_Odd0[7:0] 0000: in_b0 0001: in_g0 0010: in_r0 0011: Reserved 0100: in_b1 0101: in_g1 0110: in_r1 0111:Reserved 1000: in_b2 1001: in_g2 1010: in_r2 1011: Reserved 1100: in_b3 1101: in_g3 1110: in_r3 1111: Reserved Indicates the output order of components.
15:0	R/W	0x0	OUT_EVEN0 bit15-12: Reserved bit11-08: Out_Odd0[23:16] bit07-04: Out_Odd0[15:8] bit03-00: Out_Odd0[7:0] 0000: in_b0 0001: in_g0 0010: in_r0 0011: Reserved 0100: in_b1 0101: in_g1 0110: in_r1 0111:Reserved 1000: in_b2 1001: in_g2 1010: in_r2 1011: Reserved 1100: in_b3 1101: in_g3 1110: in_r3 1111: Reserved Indicates the output order of components.

6.1.6.39 LCD_CMAP_EVEN1_REG(Default Value: 0x0000_0000)

Offset: 0x19C			Register Name: LCD_CMAP_EVEN1_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	R/W	0x0	OUT_EVEN3 bit15-12: Reserved bit11-08: Out_Odd0[23:16] bit07-04: Out_Odd0[15:8] bit03-00: Out_Odd0[7:0] 0000: in_b0 0001: in_g0 0010: in_r0 0011: Reserved 0100: in_b1 0101: in_g1 0110: in_r1 0111:Reserved 1000: in_b2 1001: in_g2 1010: in_r2 1011: Reserved 1100: in_b3 1101: in_g3 1110: in_r3 1111: Reserved Indicates the output order of components.
15:0	R/W	0x0	OUT_EVEN2 bit15-12: Reserved bit11-08: Out_Odd0[23:16] bit07-04: Out_Odd0[15:8] bit03-00: Out_Odd0[7:0] 0000: in_b0 0001: in_g0 0010: in_r0 0011: Reserved 0100: in_b1 0101: in_g1 0110: in_r1 0111:Reserved 1000: in_b2 1001: in_g2 1010: in_r2 1011: Reserved 1100: in_b3 1101: in_g3 1110: in_r3

			1111: Reserved Indicates the output order of components.

6.1.6.40 LCD_SAFE_PERIOD_REG(Default Value: 0x0000_0000)

Offset: 0x1F0			Register Name: LCD_SAFE_PERIOD_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	/	/	/
28:16	R/W	0x0	SAFE_PERIOD_FIFO_NUM When the data length in line buffer is more than SAFE_PERIOD_FIFO_NUM, LCD controller will allow dram controller to stop working to change frequency.
15:4	R/W	0x0	SAFE_PERIOD_LINE Set a fixed line and during the line time,LCD controller allow dram controller to change frequency.The fixed line should be set in the blanking area.
3	/	/	/
2:0	R/W	0x0	SAFE_PERIOD_MODE 000: unsafe 001: safe 010: safe at ecc_fifo_curr_num > safe_period_fifo_num 011: safe at 2 and safe at sync active 100: safe at line Select the save mode.

6.1.6.41 LCD_LVDS0_ANA_REG(Default Value: 0x0000_0000)

Offset: 0x220			Register Name: LCD_LVDS0_ANA_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0	LVDS_EN_MB Enable the bias circuit of the LVDS_Ana module. 0: Disable 1: Enable
30:25	/	/	/
24	R/W	0	LVDS_HPREN_DRVC Enable clock channel drive. 0: Disable 1: Enable
23:20	R/W	0	LVDS_HPREN_DRV Enable data channel[3:0] drive. 0: Disable 1: Enable
19:17	R/W	0	LVDS_REG_C Adjust current flowing through Rload of Rx to change the differential

			signals amplitude. 0: 216mv 1: 252mv 2: 276mv 3: 312mv 4: 336mv 5: 372mv 6: 395mv 7: 432mv
16	R/W	0	LVDS_REG_DENC Choose data output or PLL test clock output in LVDS_tx.
15:12	R/W	0	LVDS_REG_DEN Choose data output or PLL test clock output in LVDS_tx.
11	/	/	/
10:8	R/W	0	LVDS_REG_R Adjust current flowing through Rload of Rx to change the common signals amplitude. 0: 0.925v 1: 0.950v 2: 0.975v 3: 1.000v 4: 1.025v 5: 1.050v 6: 1.075v 7: 1.100v
7:5	/	/	/
4	R/W	0	LVDS_REG_PLRC LVDS clock channel direction. 0: Normal 1: Reverse
3:0	R/W	0	LVDS_REG_PLR LVDS data channel [3:0] direction. 0: Normal 1: Reverse

6.2 MIPI DSI

6.2.1 Overview

The MIPI Display Serial Interface(MIPI DSI) is a high-speed interface between a host processor and peripheral devices that adhere to MIPI Alliance specifications for mobile device interfaces.This DSI module is composed of a DSI controller which is compliance with MIPI DSI specification V1.01 and a D-PHY module which is compliance with MIPI DPHY specification V1.00.

The MIPI DSI module includes the following features:

- Compliance with MIPI DSI V1.01
- 1/2/3/4 data lanes configuration and up to 1Gbit/s per lane
- Support ECC,CRC generation and EOT package
- Up to 1920 x 1080@60fps with 4 data lanes
- Video mode with sync pulse/sync event,burst mode
- Pixel format: RGB888, RGB666, RGB666 packed, and RGB565
- Supports HS-TX, LP-TX compliance with MIPI D-PHY v1.00
- Low power data transmission

Figures

Tables

7 Video Input Interfaces

7.1 MIPI CSI

7.1.1 Overview

The Camera Serial Interface Specification defines protocols between a host processor and peripheral device that adhere to MIPI Alliance specifications for mobile device interfaces. MIPI-CSI provides the mobile industry a standard, robust, scalable, low-power, high-speed, cost-effective interface that supports a wide range of imaging solutions for mobile devices. The MIPI-CSI is consists of MIPI-DPHY and MIPI-CSI2.

MIPI CSI includes the following features:

- Compliant with MIPI-CSI2 V1.00 and MIPI DPHY V1.00.00
- 1/2/3/4 Data Lanes Configuration and up to 1Gbps per Lane in HS Transmission
- Maximum to 8M@30fps with 4 data lane
- Supports format: YUV422-8bit/10bit,YUV420-8bit/10bit,RAW-8,RAW-10, RAW-12,RGB888,RGB565

CCI includes the following features:

- Compatible with I2C transmission in 7 bit slave ID + 1 bit R/W
- Automatic transmission
- 0/8/16/32 bits register address supported
- 8/16/32 bits data supported
- 64 bytes-FIFO input CCI data supported
- Synchronized with CSI signal and delay trigger supported
- Repeated transmission with sync signal supported

7.1.2 Block Diagram

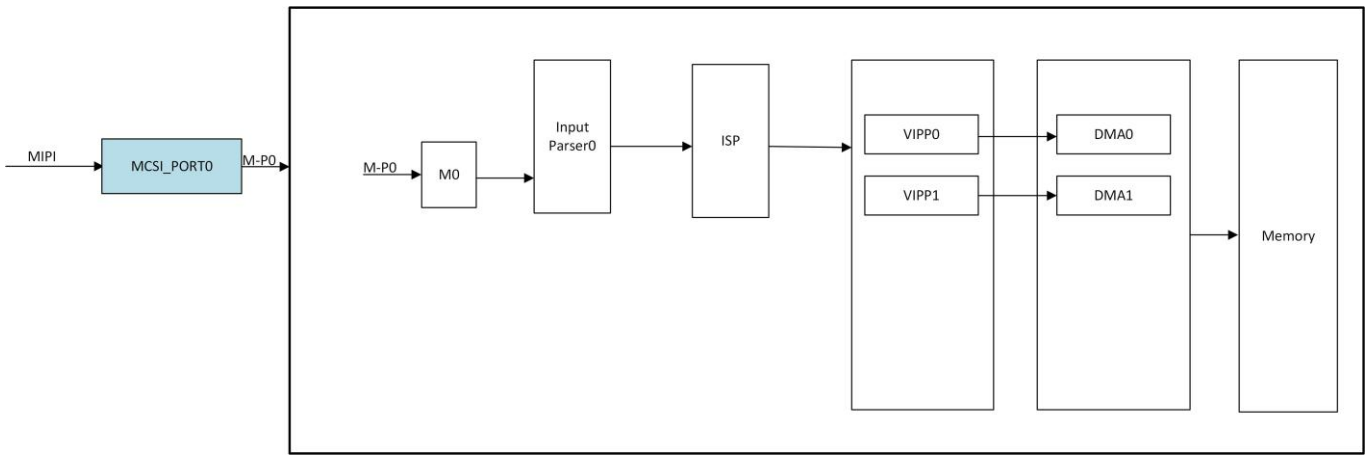


Figure7- 1. MIPI CSI Block Diagram

The CSI is consists of Input Parser, ISP, Video Input Post Process(VIPP) and DMA Control.In addition,the controller has 1 Input Parser, 1 ISP, 2 VIPP and 2 DMA.

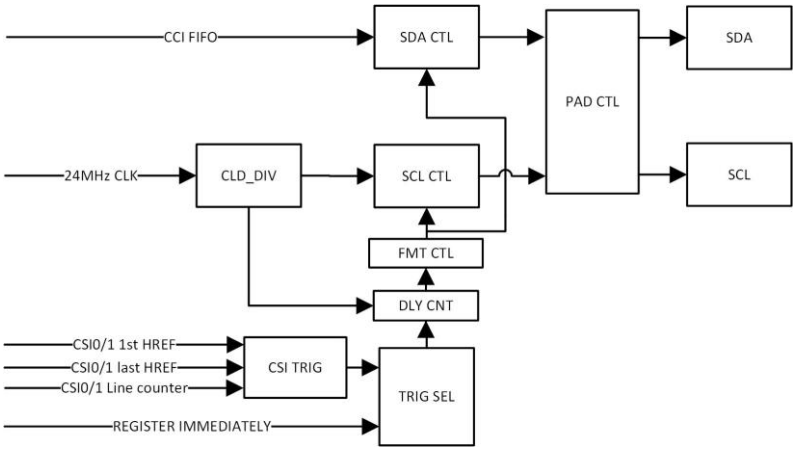


Figure7- 2. CCI Block Diagram

7.1.3 Operations and Functional Descriptions

7.1.3.1 External Signals

Table7- 1. MIPI CSI External Signals

Pin Name	Function Description	Type
MCSIA_CKP	MIPI CSI A positive differential clock line	AI
MCSIA_CKN	MIPI CSI A negative differential clock line	AI
MCSIA_D0P	MIPI CSI A positive differential data line0	AI
MCSIA_D0N	MIPI CSI A negative differential data line0	AI
MCSIA_D1P	MIPI CSI A positive differential data line1	AI
MCSIA_D1N	MIPI CSI A negative differential data line1	AI
MCSIA_D2P	MIPI CSI A positive differential data line2	AI
MCSIA_D2N	MIPI CSI A negative differential data line2	AI
MCSIA_D3P	MIPI CSI A positive differential data line3	AI
MCSIA_D3N	MIPI CSI A negative differential data line3	AI

MCSIB_CKP	MIPI CSI B positive differential clock line	AI
MCSIB_CKN	MIPI CSI B negative differential clock line	AI
MCSIB_D0P	MIPI CSI B positive differential data line0	AI
MCSIB_D0N	MIPI CSI B negative differential data line0	AI
MCSIB_D1P	MIPI CSI B positive differential data line1	AI
MCSIB_D1N	MIPI CSI B negative differential data line1	AI
MCSIB_D2P	MIPI CSI B positive differential data line2	AI
MCSIB_D2N	MIPI CSI B negative differential data line2	AI
MCSIB_D3P	MIPI CSI B positive differential data line3	AI
MCSIB_D3N	MIPI CSI B negative differential data line3	AI

7.1.3.2 CSIC FIFO Distribution

Table7- 2. CSIC FIFO Distribution

Interface	MIPI Interface		
Input format	YUV422		Raw
Output format	Planar	UV combined	Raw/RGB /PRGB
CH0_FIFO0	Y	Y	All pixels data
CH0_FIFO1	Cb (U)	CbCr (UV)	-
CH0_FIFO2	Cr (V)	-	-

7.1.3.3 Pixel Format Arrangement

RAW-10:

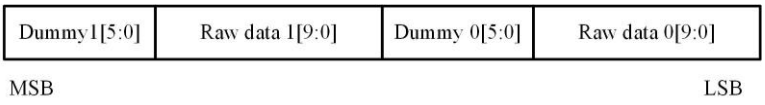


Figure7- 3. RAW-10 Format

RAW-12:

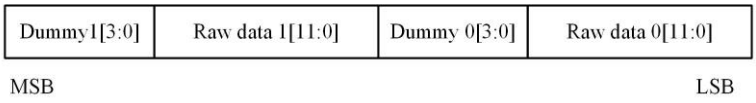


Figure7- 4. RAW-12 Format

YUV-10:

Dummy1[5:0]	Y1[9:0]	Dummy 0[5:0]	Y 0[9:0]

MSB LSB

Figure7- 5. Y of YUV-10 Format

Dummy1[5:0]	V/U 0[9:0]	Dummy 0[5:0]	U/V 0[9:0]

MSB LSB

Figure7- 6. UV Combined of YUV-10 Format

RGB888:

B1[7:0]	R0[7:0]	G0[7:0]	B0[7:0]
G2[7:0]	B2[7:0]	R1[7:0]	G1[7:0]
R3[7:0]	G3[7:0]	B3[7:0]	R2[7:0]

MSB

Figure7- 7. RGB888 Format

PRGB888:

PAD[7:0]	R0[7:0]	G0[7:0]	B0[7:0]

MSB LSB

Figure7- 8. PRGB888 Format

RGB565:

R1[4:0]	G1[5:0]	B1[4:0]	R0[4:0]	G0[5:0]	B0[4:0]

MSB LSB

Figure7- 9. RGB565 Format

7.1.3.4 Offset/Flip Function

7.1.3.4.1 Offset Definition

Offset in horizontal and vertical can be added when receiving image. Unit is pixel.
For YUV422 format, pixel unit is a YU/YV combination.
For YUV420 format, pixel unit is a YU/YV combination in YC line, and only a Y in Y line.
For Bayer and RAW format, pixel unit is a R/G/B single component.
For RGB565, pixel unit is 2 bytes of RGB565 package.
For RGB888, pixel unit is 3 bytes of RGB combination.

7.1.3.4.2 Flip Definition

Both horizontal and vertical flip are supported at the same time. This function is implemented in the process of each FIFO writing data to memory, only flipping the data of separate FIFO, not changing component to FIFO distribution.

If horizontal flip is enabled, one or more pixels will be took as a unit:
For YUV format, a unit of $Y_0U_0Y_1V_1$ will parser and flip the Y component in one channel, and UV will be treated as a whole. In planar output mode, U and V will be flipped separately. In UV combined output mode, UV will be flipped as a whole. So, a sequence of $Y_1U_0Y_0V_1$ will be.

For Bayer_raw format, situation is much like. A GR/BG sequence will be changed to BG/RG. A unit of square has four pixels.

For RGB565/RGB888, one unit of two/three bytes of component will be flipped with original sequence.

7.1.3.5 Camera Communication Interface

The CCI module support master mode i2c-compatible single read and write access to camera and related devices.
It reads a series of packet from FIFO (accessed by registers) and transmit with the format defined in specific register(or packet data).
In compact mode, format register define the slave ID, R/W flag, register address width(0/8/16/32…bit), data width(8/16/32…bit) and access counter.
In complete mode, all data and format will be loaded from memory packet.
The access counter should be set to N(N> 0), and it will read N packets from FIFO. The total bytes should not exceed 64 for FIFO input mode.

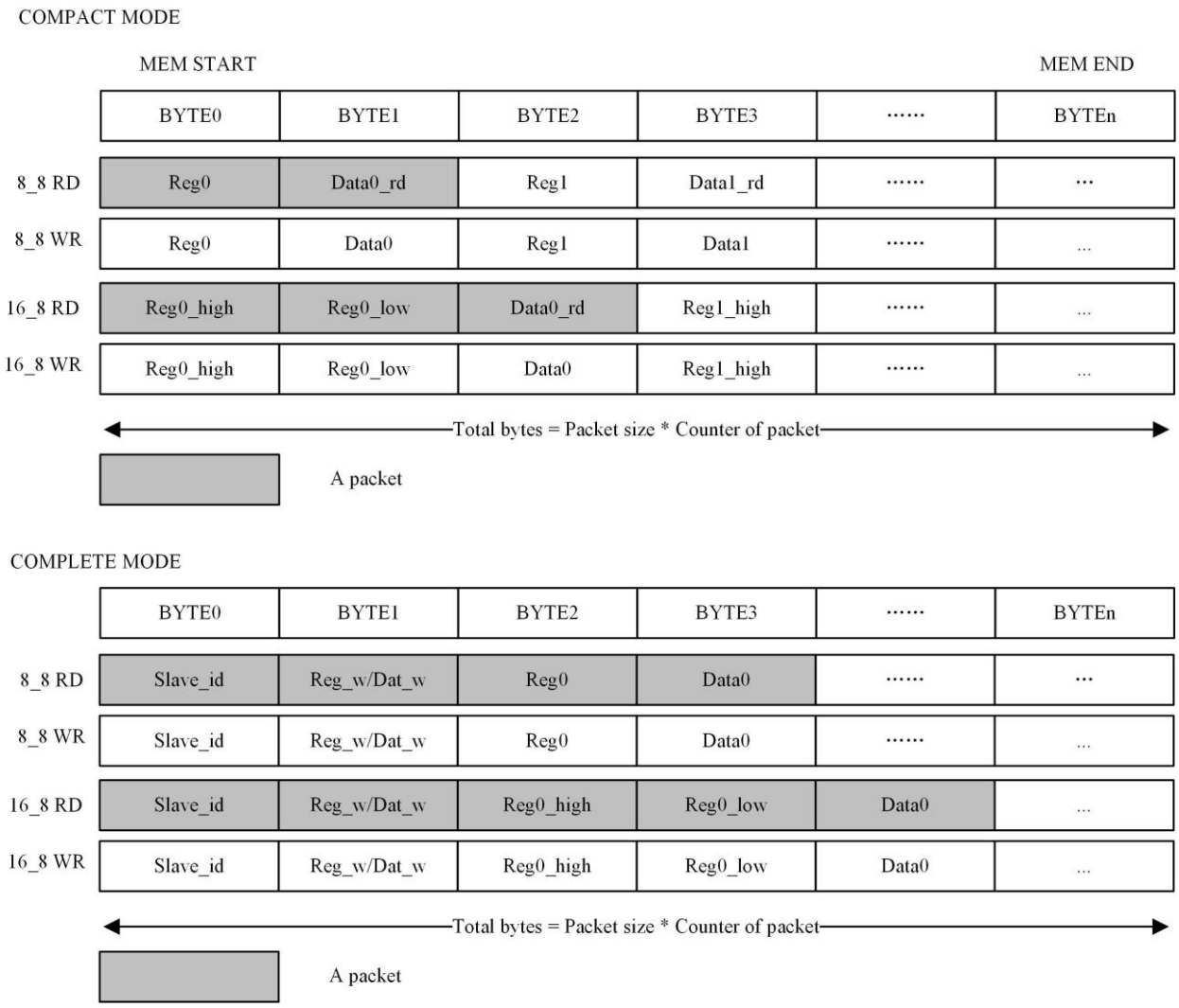


Figure7- 10. R/W Sequence in Compact/Complete Mode

A packet is several bytes filled with register address and data(if in complete mode, slave id and width should be filled too) as the i2c access sequence defined. That is, the low address byte will be transmitted/received first. Bytes will be sent in write access, while some address will be written back with the data received in read access.

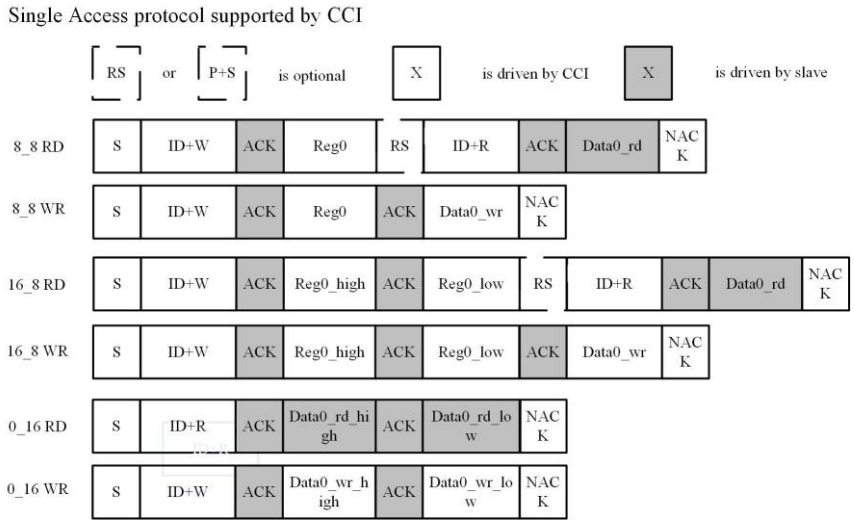


Figure7- 11. Single R/W process of the CCI protocol

After set the execution bit, the module will do the transmission automatically and return the result - success or fail. If any access fail, the whole transmission will be stopped and returns the number when it fail in the access counter.

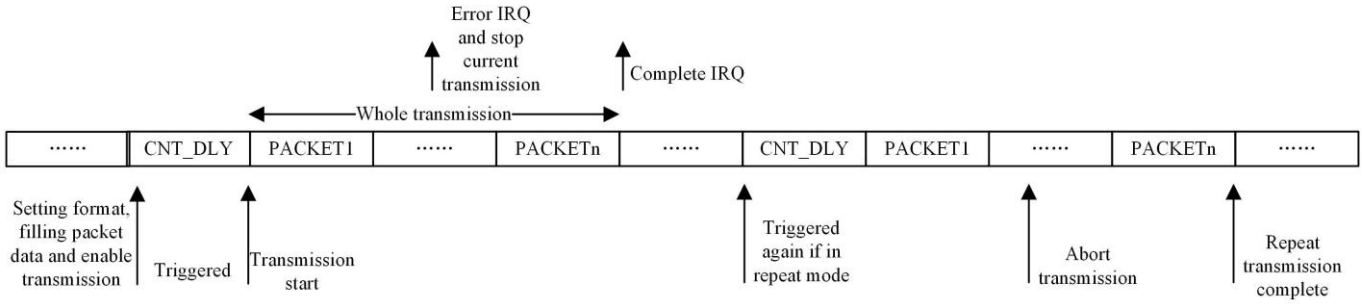


Figure7- 12. CCI transmission control

7.1.4 Register list

Module Name	Base Address
CSIC_BASE	0x06600000
CSIC_TOP	0x06600000
CSIC_PARSER	0x06601000
CSIC_DMA0	0x06609000
CSIC_DMA1	0x06609200
CSIC_VIPP0	0x02101000
CSIC_VIPP1	0x02101400
CSIC_CCI	0x06614000

CSI TOP Register list:

Register Name	Offset	Description
CSIC_TOP_EN_REG	0x0000	CSIC TOP Enable Register
CSIC_PTN_GEN_EN_REG	0x0004	CSIC Pattern Generation Enable Register
CSIC_PTN_CTRL_REG	0x0008	CSIC Pattern Control Register
CSIC_PTN_LEN_REG	0x0020	CSIC Pattern Generation Length Register
CSIC_PTN_ADDR_REG	0x0024	CSIC Pattern Generation Address Register
CSIC_PTN_ISP_SIZE_REG	0x0028	CSIC Pattern ISP Size Register
CSIC_ISP_INPUT0_SEL_REG	0x0030	CSIC ISP Input0 Select Register
CSIC_ISP_INPUT1_SEL_REG	0x0034	CSIC ISP Input1 Select Register
CSIC_ISP_INPUT2_SEL_REG	0x0038	CSIC ISP Input2 Select Register
CSIC_ISP_INPUT3_SEL_REG	0x003C	CSIC ISP Input3 Select Register
CSIC_VIPP0_INPUT_SEL_REG	0x00A0	CSIC VIPP0 Input Select Register
CSIC_VIPP1_INPUT_SEL_REG	0x00A4	CSIC VIPP1 Input Select Register

PARSER Register list:

Register Name	Offset	Description
PRS_EN_REG	0x0000	Parser Enable Register
PRS_MCSIC_IF_CFG_REG	0x0008	Parser MCSIC Interface Configuration Register
PRS_CAP_REG	0x000C	Parser Capture Register
PRS_SIGNAL_STA_REG	0x0010	Parser Signal Status Register
/	0x0014~0X0020	Reserved

PRS_C0_INFMT_REG	0x0024	Parser Channel_0 Input Format Register
PRS_C0_OUTPUT_HSIZE_REG	0x0028	Parser Channel_0 Output Horizontal Size Register
PRS_C0_OUTPUT_VSIZE_REG	0x002C	Parser Channel_0 Output Vertical Size Register
PRS_C0_INPUT_PARA0_REG	0x0030	Parser Channel_0 Input Parameter0 Register
PRS_C0_INPUT_PARA1_REG	0x0034	Parser Channel_0 Input Parameter1 Register
PRS_C0_INPUT_PARA2_REG	0x0038	Parser Channel_0 Input Parameter2 Register
PRS_C0_INPUT_PARA3_REG	0x003C	Parser Channel_0 Input Parameter3 Register
PRS_C0_INT_EN_REG	0x0040	Parser Channel_0 Interrupt Enable Register
PRS_C0_INT_STA_REG	0x0044	Parser Channel_0 Interrupt Status Register
/	0x0048~0X0120	Reserved
PRS_C1_INFMT_REG	0x0124	Parser Channel_1 Input Format Register
PRS_C1_OUTPUT_HSIZE_REG	0x0128	Parser Channel_1 Output Horizontal Size Register
PRS_C1_OUTPUT_VSIZE_REG	0x012C	Parser Channel_1 Output Vertical Size Register
PRS_C1_INPUT_PARA0_REG	0x0130	Parser Channel_1 Input Parameter0 Register
PRS_C1_INPUT_PARA1_REG	0x0134	Parser Channel_1 Input Parameter1 Register
PRS_C1_INPUT_PARA2_REG	0x0138	Parser Channel_1 Input Parameter2 Register
PRS_C1_INPUT_PARA3_REG	0x013C	Parser Channel_1 Input Parameter3 Register
PRS_C1_INT_EN_REG	0x0140	Parser Channel_1 Interrupt Enable Register
PRS_C1_INT_STA_REG	0x0144	Parser Channel_1 Interrupt Status Register

DMA0/1 register list:

Register Name	Offset	Description
CSIC_DMA_EN_REG	0x0000	CSIC DMA Enable Register
CSIC_DMA_CFG_REG	0x0004	CSIC DMA Configuration Register
/	0x0008~0x000C	Reserved
CSIC_DMA_HSIZE_REG	0x0010	CSIC DMA Horizontal Size Register
CSIC_DMA_VSIZE_REG	0x0014	CSIC DMA Vertical Size Register
/	0x0018~0x001C	Reserved
CSIC_DMA_F0_BUFA_REG	0x0020	CSIC DMA FIFO 0 Output Buffer-A Address Register
/	0x0024	Reserved
CSIC_DMA_F1_BUFA_REG	0x0028	CSIC DMA FIFO 1 Output Buffer-A Address Register
/	0x002C	Reserved
CSIC_DMA_F2_BUFA_REG	0x0030	CSIC DMA FIFO 2 Output Buffer-A Address Register
/	0x0034	Reserved
CSIC_DMA_BUF_LEN_REG	0x0038	CSIC DMA Buffer Length Register
CSIC_DMA_FLIP_SIZE_REG	0x003C	CSIC DMA Flip Size Register
/	0x0040~0x0048	Reserved
CSIC_DMA_CAP_STA_REG	0x004C	CSIC DMA Capture Status Register
CSIC_DMA_INT_EN_REG	0x0050	CSIC DMA Interrupt Enable Register
CSIC_DMA_INT_STA_REG	0x0054	CSIC DMA Interrupt Status Register
CSIC_DMA_LINE_CNT_REG	0x0058	CSIC DMA LINE COUNTER Register
CSIC_DMA_FRM_CLK_CNT_REG	0x0060	CSIC DMA Frame Clock Counter Register
CSIC_DMA_ACC_ITNL_CLK_CNT_REG	0x0064	CSIC DMA Accumulated And Internal Clock Counter Register

CSIC_DMA_FIFO_STAT_REG	0x0068	CSIC DMA FIFO Statistic Register
CSIC_DMA_FIFO_THRS_REG	0x006C	CSIC DMA FIFO Threshold Register
CSIC_DMA_PCLK_STAT_REG	0x0070	CSIC DMA PCLK Statistic Register

CCI Register list:

Register Name	Offset	Description
CCI_CTRL	0x0000	CCI Control Register
CCI_CFG	0x0004	CCI Transmission Configuration Register
CCI_FMT	0x0008	CCI Packet Format Register
CCI_BUS_CTRL	0x000C	CCI Bus Control Register
/	0x0010	Reserved
CCI_INT_CTRL	0x0014	CCI Interrupt Control Register
CCI_LC_TRIG	0x0018	CCI Line Counter Trigger Register
/	0x001C~0x00FC	Reserved
CCI_FIFO_ACC	0x0100~0x013C	CCI FIFO Access Register
/	0x0140~0x01FC	Reserved
CCI_RSV_REG	0x0200~0x0220	CCI Reserved Register

7.1.5 Register Description

7.1.5.1 CSIC TOP Enable Register(Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: CSIC_TOP_EN_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	R/W	0x0	Reserved
0	R/W	0x0	CSI_TOP_EN 0: Reset and disable the CSI module 1: Enable the CSI module

7.1.5.2 CSIC Pattern Generation Enable Register(Default Value:0x0000_0000)

Offset: 0x0004			Register Name: CSIC_PTN_GEN_EN_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	/	/	/
23:16	R/W	0x0	PTN_CYCLE Pattern generating cycle counter. The pattern in dram will be generated in cycles of PTN_CYCLE+1.
15:5	/	/	/
4	R/WAC	0x0	PTN_START CSIC Pattern Generating Start 0: Finish other: Start Software write this bit to “1” to start pattern generating from DRAM. When finished, the hardware will clear this bit to “0” automatically. Generating

			cycles depends on PTN_CYCLE.
3:1	/	/	/
0	R/W	0x0	PTN_GEN_EN Pattern Generation Enable

7.1.5.3 CSIC Pattern Control Register(Default Value:0x0000_000F)

Offset: 0x0008			Register Name: CSIC_PTN_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	/	/	/
26:24	R/W	0x0	PTN_PORT_SEL Pattern Generator output port selection 000:MCSIC0 Others:reserved
23:22	/	/	/
21:20	R/W	0x0	PTN_GEN_DATA_WIDTH 00:8bit 01:10bit Others:reserved
19:16	R/W	0x0	PTN_MODE Pattern mode selection 0000:MIPI_csi2 1 data lane 0001:MIPI_csi2 2 data lane 0010:MIPI_csi2 3 data lane 0011:MIPI_csi2 4 data lane Others:reserved
15:10	/	/	/
9:8	R/W	0x0	PTN_GEN_CLK_DIV Packet generator clock divider
7:0	R/W	0xF	PTN_GEN_DLY Clocks delayed before pattern generating start.

7.1.5.4 CSIC Pattern Generation Length Register(Default Value:0x0000_0000)

Offset: 0x0020			Register Name: CSIC_PTN_LEN_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	PTN_LEN The pattern length in byte when generating pattern.

7.1.5.5 CSIC Pattern Generation Address Register(Default Value:0x0000_0000)

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

31:0	R/W	0x0	PTN_ADDR The pattern DRAM address when generating pattern.

7.1.5.6 CSIC Pattern ISP Size Register(Default Value:0x0000_0000)

Offset: 0x0028			Register Name: CSIC_PTN_ISP_SIZE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	/	/	/
28:16	R/W	0x0	Height Vertical size,only valid for ISP mode pattern generation.
15:13	/	/	/
12:0	R/W	0x0	Width Horizontal size,only valid for ISP mode pattern generation.

7.1.5.7 CSIC ISP Input0 Select Register(Default Value:0x0000_0000)

Offset :0x0030			Register Name: CSIC_ISP_INPUT0_SEL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2:0	R/W	0x0	ISP Input0 select 000: input from Parser0 CH0 001~011: reserved 100: input from Parser0 CH1 others: reserved

7.1.5.8 CSIC ISP Input1 Select Register(Default Value:0x0000_0000)

Offset :0x0034			Register Name: CSIC_ISP_INPUT1_SEL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2:0	R/W	0x0	ISP Input1 select 000: input from Parser0 CH1 others: reserved

7.1.5.9 CSIC ISP Input2 Select Register(Default Value:0x0000_0000)

Offset :0x0038			Register Name: CSIC_ISP_INPUT2_SEL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2:0	R/W	0x0	ISP Input2 select 000: input from Parser0 CH2 others: reserved

7.1.5.10 CSIC ISP Input3 Select Register(Default Value:0x0000_0000)

Offset :0x003C			Register Name: CSIC_ISP_INPUT3_SEL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2:0	R/W	0x0	ISP Input3 select 000: input from Parser0 CH3 others: reserved

7.1.5.11 CSIC VIPP0 Input Select Register(Default Value:0x0000_0000)

Offset :0x00A0			Register Name: CSIC_VIPP0_INPUT_SEL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2:0	R/W	0x0	VIPP0 Input select 000: input from ISP CH0 others: reserved

7.1.5.12 CSIC VIPP1 Input Select Register(Default Value:0x0000_0000)

Offset :0x00A4			Register Name: CSIC_VIPP1_INPUT_SEL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2:0	R/W	0x0	VIPP1 Input select 000: input from ISP CH0 001:reserved 010: input from ISP CH1 others: reserved

7.1.5.13 Parser Enable Register(Default Value:0x0000_0000)

Offset: 0x0000			Register Name: PRS_EN_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	MCSIC_EN 0: Reset and disable the MCSIC module 1: Enable the MCSIC module
30:17	/	/	/
16	R/W	0x0	Reserved
15	R/W	0x0	PCLK_EN 0:Gate pclk input 1:Enable pclk input

14:2	/	/	/
1	R/W	0x0	PRS_MODE 0: / 1: MCSI
0	R/W	0x0	PRS_EN 0: Reset and disable the parser module 1: Enable the parser module

7.1.5.14 Parser MCSIC Interface Configuration Register(Default Value:0x0000_0080)

Offset: 0x0008			Register Name: PRS_MCSI_IF_CFG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:6	R/W	0X2	INPUT_SEQ Input data sequence, only valid for YUV422 and YUV420 input format. All data interleaved in one channel: 00: YUYV 01: YVYU 10: UYVY 11: VYUY Y and UV in separated channel: x0: UV x1: VU
5	R/W	0x0	OUTPUT_MODE 0:field mode 1:frame mode
4:0	/	/	/

7.1.5.15 Parser Capture Register(Default Value:0x0000_0000)

Offset: 0x000C			Register Name: PRS_CAP_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	/	/	/
30	R/W	0x0	CH3_HALF_CAP Fps down sample 0: no down sample 1: 1/2 fps
29:26	R/W	0x0	CH3_CAP_MASK Vsync number masked before capture.
25	R/W	0x0	CH3_VCAP_ON Video capture control: Capture the video image data stream on channel 3. 0: Disable video capture

			If video capture is in progress, the CSI stops capturing image data at the end of the current frame, and all of the current frame data is wrote to output FIFO. 1: Enable video capture The CSI starts capturing image data at the start of the next frame.
24	RC/W	0x0	CH3_SCAP_ON Still capture control: Capture a single still image frame on channel 3. 0: Disable still capture. 1: Enable still capture The CSI module starts capturing image data at the start of the next frame. The CSI module captures only one frame of image data. This bit is self clearing and always reads as a 0.
23	/	/	/
22	R/W	0x0	CH2_HALF_CAP Fps down sample 0: no down sample 1: 1/2 fps
21:18	R/W	0x0	CH2_CAP_MASK Vsync number masked before capture.
17	R/W	0x0	CH2_VCAP_ON Video capture control: Capture the video image data stream on channel 2. 0: Disable video capture If video capture is in progress, the CSI stops capturing image data at the end of the current frame, and all of the current frame data is wrote to output FIFO. 1: Enable video capture The CSI starts capturing image data at the start of the next frame.
16	RC/W	0x0	CH2_SCAP_ON Still capture control: Capture a single still image frame on channel 2. 0: Disable still capture. 1: Enable still capture The CSI module starts capturing image data at the start of the next frame. The CSI module captures only one frame of image data. This bit is self clearing and always reads as a 0.
15	/	/	/
14	R/W	0x0	CH1_HALF_CAP Fps down sample 0: no down sample 1: 1/2 fps
13:10	R/W	0x0	CH1_CAP_MASK Vsync number masked before capture.
9	R/W	0x0	CH1_VCAP_ON Video capture control: Capture the video image data stream on channel 1. 0: Disable video capture If video capture is in progress, the CSI stops capturing image data at the end of the current frame, and all of the current frame data is wrote to output

			FIFO. 1: Enable video capture The CSI starts capturing image data at the start of the next frame.
8	RC/W	0x0	CH1_SCAP_ON Still capture control: Capture a single still image frame on channel 1. 0: Disable still capture. 1: Enable still capture The CSI module starts capturing image data at the start of the next frame. The CSI module captures only one frame of image data. This bit is self clearing and always reads as a 0.
7	/	/	/
6	R/W	0x0	CH0_HALF_CAP Fps down sample 0: no down sample 1: 1/2 fps
5:2	R/W	0x0	CH0_CAP_MASK Vsync number masked before capture.
1	R/W	0x0	CH0_VCAP_ON Video capture control: Capture the video image data stream on channel 0. 0: Disable video capture If video capture is in progress, the CSI stops capturing image data at the end of the current frame, and all of the current frame data is wrote to output FIFO. 1: Enable video capture The CSI starts capturing image data at the start of the next frame.
0	RC/W	0x0	CH0_SCAP_ON Still capture control: Capture a single still image frame on channel 0. 0: Disable still capture. 1: Enable still capture The CSI module starts capturing image data at the start of the next frame. The CSI module captures only one frame of image data. This bit is self clearing and always reads as a 0.

7.1.5.16 Parser Signal Status Register(Default Value:0x0000_0000)

Offset: 0x0010			**Register Name: PRS_SIGNAL_STA_REG**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:25	/	/	/
24	R	0x0	PCLK_STA Indicates the pclk status 0:low 1:high
23:0	R	0x0	DATA_STA Indicates the Dn status(n=0~23),MSB for D23,LSB for D0 0:low

			1:high

7.1.5.17 Parser Channel_0 Input Format Register(Default Value:0x0000_0003)

Offset: 0x0024			Register Name: PRS_CH0_INFMT_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:4	/	/	/
3:0	R/W	0x3	INPUT_FMT Input data format 0000: RAW stream 0001: reserved 0010: reserved 0011: YUV422 0100: YUV420 Others: reserved

7.1.5.18 Parser Channel_0 Output Horizontal Size Register(Default Value:0x0500_0000)

Offset: 0x0028			Register Name: PRS_CH0_OUTPUT_HSIZE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	/	/	/
28:16	R/W	0x500	HOR_LEN Horizontal pixel unit length. Valid pixel of a line.
15:13	/	/	/
12:0	R/W	0x0	HOR_START Horizontal pixel unit start. Pixel is valid from this pixel.

7.1.5.19 Parser Channel_0 Output Vertical Size Register(Default Value:0x02D0_0000)

Offset: 0x002C			Register Name: PRS_CH0_OUTPUT_VSIZE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	/	/	/
28:16	R/W	0x2D0	VER_LEN Valid line number of a frame.
15:13	/	/	/
12:0	R/W	0x0	VER_START Vertical line start. data is valid from this line.

7.1.5.20 Parser Channel_0 Input Parameter0 Register(Default Value:0x0000_0000)

Offset: 0x0030	Register Name: PRS_CH0_INPUT_PARA0_REG

Bit	Read/Write	Default/Hex	Description
31:1	/	/	/
0	R	0x0	INPUT_SRC_TYPE 0:Progress 1:Interlace

7.1.5.21 Parser Channel_0 Input Parameter1 Register(Default Value:0x0000_0000)

Offset: 0x0034			Register Name: PRS_CH0_INPUT_PARA1_REG
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:16	R	0x0	INPUT_VT INPUT_VT = INPUT_VB+INPUT_Y
15:14	/	/	/
13:0	R	0x0	INPUT_HT INPUT_HT = INPUT_HB+INPUT_X

7.1.5.22 Parser Channel_0 Input Parameter2 Register(Default Value:0x0000_0000)

Offset: 0x0038			Register Name: PRS_CH0_INPUT_PARA2_REG
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:16	R	0x0	INPUT_VB
15:14	/	/	/
13:0	R	0x0	INPUT_HB

7.1.5.23 Parser Channel_0 Input Parameter3 Register(Default Value:0x0000_0000)

Offset: 0x003C			Register Name: PRS_CH0_INPUT_PARA3_REG
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:16	R	0x0	INPUT_Y
15:14	/	/	/
13:0	R	0x0	INPUT_X

7.1.5.24 Parser Channel_0 Interrupt Enable Register(Default Value:0x0000_0000)

Offset: 0x0040			Register Name: PRS_CH0_INT_EN_REG
Bit	Read/Write	Default/Hex	Description
31:3	/	/	/
2	R/W	0x0	MUL_ERR_INT_EN

			Multi-channel writing error Indicates error has been detected for writing data to a wrong channel.
1	R/W	0x0	INPUT_PARA1_INT_EN 0:disable 1:enable
0	R/W	0x0	INPUT_PARA0_INT_EN 0:disable 1:enable

7.1.5.25 Parser Channel_0 Interrupt Status Register(Default Value:0x0000_0000)

Offset: 0x0044			Register Name: PRS_CH0_INT_STA_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2	R/W1C	0x0	MUL_ ERR_PD Multi-channel writing error
1	R/W1C	0x0	INPUT_SRC_PD1 When the parser input parameter0 register update,this flag set to 1. Write 1 to clear.
0	R/W1C	0x0	INPUT_SRC_PD0 When the parser input parameter1 register,parser input parameter2 register or parser input parameter3 register update,this flag set to 1. Write 1 to clear.

7.1.5.26 Parser Channel_1 Input Format Register(Default Value:0x0000_0003)

Offset: 0x0124			Register Name: PRS_CH1_INFMT_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:4	/	/	/
3:0	R/W	0x3	INPUT_FMT Input data format 0000: RAW stream 0001: reserved 0010: reserved 0011: YUV422 0100: YUV420 Others: reserved

7.1.5.27 Parser Channel_1 Output Horizontal Size Register(Default Value:0x0500_0000)

Offset: 0x0128			Register Name: PRS_CH1_OUTPUT_HSIZE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	/	/	/
28:16	R/W	0x500	HOR_LEN

			Horizontal pixel unit length. Valid pixel of a line.
15:13	/	/	/
12:0	R/W	0x0	HOR_START Horizontal pixel unit start. Pixel is valid from this pixel.

7.1.5.28 Parser Channel_1 Output Vertical Size Register(Default Value:0x02D0_0000)

Offset: 0x012C			Register Name: PRS_CH1_OUTPUT_VSIZE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	/	/	/
28:16	R/W	0x2D0	VER_LEN Valid line number of a frame.
15:13	/	/	/
12:0	R/W	0x0	VER_START Vertical line start. data is valid from this line.

7.1.5.29 Parser Channel_1 Input Parameter0 Register(Default Value:0x0000_0000)

Offset: 0x0130			Register Name: PRS_CH1_INPUT_PARA0_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R	0x0	INPUT_SRC_TYPE 0:Progress 1:Interlace

7.1.5.30 Parser Channel_1 Input Parameter1 Register(Default Value:0x0000_0000)

Offset: 0x0134			Register Name: PRS_CH1_INPUT_PARA1_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/
29:16	R	0x0	INPUT_VT INPUT_VT = INPUT_VB+INPUT_Y
15:14	/	/	/
13:0	R	0x0	INPUT_HT INPUT_HT = INPUT_HB+INPUT_X

7.1.5.31 Parser Channel_1 Input Parameter2 Register(Default Value:0x0000_0000)

Offset: 0x0138			Register Name: PRS_CH1_INPUT_PARA2_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/

29:16	R	0x0	INPUT_VB
15:14	/	/	/
13:0	R	0x0	INPUT_HB

7.1.5.32 Parser Channel_1 Input Parameter3 Register(Default Value:0x0000_0000)

Offset: 0x013C			Register Name: PRS_CH1_INPUT_PARA3_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/
29:16	R	0x0	INPUT_Y
15:14	/	/	/
13:0	R	0x0	INPUT_X

7.1.5.33 Parser Channel_1 Interrupt Enable Register(Default Value:0x0000_0000)

Offset: 0x0140			Register Name: PRS_CH1_INT_EN_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2	R/W	0x0	MUL_ERR_INT_EN Multi-channel writing error Indicates error has been detected for writing data to a wrong channel.
1	R/W	0x0	INPUT_PARA1_INT_EN 0:disable 1:enable
0	R/W	0x0	INPUT_PARA0_INT_EN 0:disable 1:enable

7.1.5.34 Parser Channel_1 Interrupt Status Register(Default Value:0x0000_0000)

Offset: 0x0144			Register Name: PRS_CH1_INT_STA_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2	R/W1C	0x0	MUL_ ERR_PD Multi-channel writing error
1	R/W1C	0x0	INPUT_SRC_PD1 When the parser input parameter0 register update,this flag set to 1. Write 1 to clear.
0	R/W1C	0x0	INPUT_SRC_PD0 When the parser input parameter1 register,parser input parameter2 register or parser input parameter3 register update,this flag set to 1. Write 1 to clear.

7.1.5.35 CSIC DMA Enable Register(Default Value:0x0000_0000)

Offset:0x0000			Register Name: CSIC_DMA_EN_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2	R/W	0x0	CLK_CNT_SPL Sampling time for clk counter per frame 0: Sampling clock counter every frame done 1: Sampling clock counter every vsync
1	R/W	0x0	CLK_CNT_EN clk count per frame enable
0	R/W	0x0	DMA_EN 0: Disable 1: Enable

7.1.5.36 CSIC DMA Configuration Register(Default Value:0x0000_0000)

Offset: 0x0004			Register Name: CSIC_DMA_CFG_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	R/W	0x0	PAD_VAL Padding value when OUTPUT_FMT is prgb888 0x00~0xff
23:20	/	/	/
19:16	R/W	0x0	OUTPUT_FMT Output data format When the input format is set RAW stream 0000: field-raw-8 0001: field-raw-10 0010: field-raw-12 0011: reserved 0100: field-rgb565 0101: field-rgb888 0110: field-prgb888 0111: reserved 1000: frame-raw-8 1001: frame-raw-10 1010: frame-raw-12 1011: reserved 1100: frame-rgb565 1101: frame-rgb888 1110: frame-prgb888 1111: reserved When the input format is set YUV422 0000: field planar YCbCr 422

			0001: field planar YCbCr 420 0010: frame planar YCbCr 420 0011: frame planar YCbCr 422 0100: field planar YCbCr 422 UV combined(UV sequence) 0101: field planar YCbCr 420 UV combined(UV sequence) 0110: frame planar YCbCr 420 UV combined(UV sequence) 0111: frame planar YCbCr 422 UV combined(UV sequence) 1000: filed planar YCbCr 422 UV combined(VU sequence) 1001: field planar YCbCr 420 UV combined(VU sequence) 1010: frame planar YCbCr 420 UV combined(VU sequence) 1011: frame planar YCbCr 422 UV combined(VU sequence) 1100: field planar YCbCr 422 10bit UV combined(UV sequence) 1101: field planar YCbCr 420 10bit UV combined(UV sequence) 1110: field planar YCbCr 422 10bit UV combined(VU sequence) 1111: field planar YCbCr 420 10bit UV combined(VU sequence) When the input format is set YUV420 0000: Reserved 0001: field planar YCbCr 420 0010: frame planar YCbCr 420 0011: reserved 0100: reserved 0101: field planar YCbCr 420 UV combined(UV sequence) 0110: frame planar YCbCr 420 UV combined(UV sequence) 0111~1000: reserved 1001: field planar YCbCr 420 UV combined(VU sequence) 1010: frame planar YCbCr 420 UV combined(VU sequence) 1011~1100: reserved 1101: field planar YCbCr 420 10bit UV combined(UV sequence) 1110: reserved 1111: field planar YCbCr 420 10bit UV combined(VU sequence) Others: reserved
15:14	/	/	/
13	R/W	0x0	VFLIP_EN Vertical flip enable When enabled, the received data will be arranged in vertical flip. 0:Disable 1:Enable
12	R/W	0x0	HFLIP_EN Horizontal flip enable When enabled, the received data will be arranged in horizontal flip. 0:Disable 1:Enable
11:10	R/W	0x0	FIELD_SEL Field selection.

			00: capturing with field 1. 01: capturing with field 2. 10: capturing with either field. 11: reserved
9:8	/	/	/
7:6	R/W	0x0	FPS_DS Fps down sample 0: no down sample 1: 1/2 fps, only receives the first frame every 2 frames 2: 1/3 fps, only receives the first frame every 3 frames 3: 1/4 fps, only receives the first frame every 4 frames
5:2	/	/	/
1:0	R/W	0x0	MIN_SDR_WR_SIZE Minimum size of SDRAM block write 0: 256 bytes (if hflip is enable, always select 256 bytes) 1: 512 bytes 2: 1k bytes 3: 2k bytes

7.1.5.37 CSIC DMA Horizontal Size Register(Default Value:0x0500_0000)

Offset: 0x0010			Register Name: CSIC_DMA_HSIZE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	/	/	/
28:16	R/W	0x500	HOR_LEN When DMA_EN enable,these bits indicate Horizontal pixel unit length. Valid pixel of a line in DMA mode.
15:13	/	/	/
12:0	R/W	0x0	HOR_START Horizontal pixel unit start. Pixel is valid from this pixel.

7.1.5.38 CSIC DMA Vertical Size Register(Default Value:0x02D0_0000)

Offset: 0x0014			Register Name: CSIC_DMA_VSIZE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	/	/	/
28:16	R/W	0x2D0	VER_LEN When DMA_EN enable,these bits indicate Valid line number of a frame in DMA mode.
15:13	/	/	/
12:0	R/W	0x0	VER_START Vertical line start. data is valid from this line.

7.1.5.39 CSIC DMA FIFO 0 Output Buffer-A Address Register(Default Value:0x0000_0000)

Offset: 0x0020			Register Name: CSIC_DMA_F0_BUFA_REG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	F0_BUFA When DMA_EN enable,these bits indicate FIFO 0 output buffer-A address in DMA mode.

7.1.5.40 CSIC DMA FIFO 1 Output Buffer-A Address Register(Default Value:0x0000_0000)

Offset: 0x0028			Register Name: CSIC_DMA_F1_BUFA_REG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	F1_BUFA When DMA_EN enable,these bits indicate FIFO 1 output buffer-A address in DMA mode.

7.1.5.41 CSIC DMA FIFO 2 Output Buffer-A Address Register(Default Value:0x0000_0000)

Offset: 0x0030			Register Name: CSIC_DMA_F2_BUFA_REG
Bit	Read/Write	Default/Hex	Description
31:0	R/W	0x0	F2_BUFA FIFO 2 output buffer-A address

7.1.5.42 CSIC DMA Buffer Length Register(Default Value:0x0280_0500)

Offset: 0x0038			Register Name: CSIC_DMA_BUF_LEN_REG
Bit	Read/Write	Default/Hex	Description
31:30	/	/	/
29:16	R/W	0x280	BUF_LEN_C Buffer length of chroma C in a line. Unit is byte.
15:14	/	/	/
13:0	R/W	0x500	BUF_LEN Buffer length of luminance Y in a line. Unit is byte.

7.1.5.43 CSIC DMA Flip Size Register(Default Value:0x02D0_0500)

Offset: 0x003C			Register Name: CSIC_DMA_FLIP_SIZE_REG
Bit	Read/Write	Default/Hex	Description
31:29	/	/	/
28:16	R/W	0x2D0	VER_LEN

			Vertical line number when in VFLIP mode. Unit is pixel.
15:13	/	/	/
12:0	R/W	0x500	VALID_LEN Valid components of a line when in HFLIP mode. Unit is pixel.

7.1.5.44 CSIC DMA Capture Status Register(Default Value:0x0000_0000)

Offset: 0x004C			Register Name: CSIC_DMA_CAP_STA_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2	R	0x0	FIELD_STA The status of the received field 0: Field 0 1: Field 1
1	R	0x0	VCAP_STA Video capture in progress Indicates the CSI is capturing video image data (multiple frames). The bit is set at the start of the first frame after enabling video capture. When software disables video capture, it clears itself after the last pixel of the current frame is captured.
0	R	0x0	SCAP_STA Still capture in progress Indicates the CSI is capturing still image data (single frame). The bit is set at the start of the first frame after enabling still frame capture. It clears itself after the last pixel of the first frame is captured. For CCIR656 interface, if the output format is frame planar YCbCr 420 mode, the frame end means the field2 end, the other frame end means filed end.

7.1.5.45 CSIC DMA Interrupt Enable Register(Default Value:0x0000_0000)

Offset: 0x0050			Register Name: CSIC_DMA_INT_EN_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R/W	0x0	VS_INT_EN vsync flag The bit is set when vsync come. And at this time load the buffer address for the coming frame. So after this irq come, change the buffer address could only effect next frame
6	R/W	0x0	HB_OF_INT_EN Hblank FIFO overflow The bit is set when 3 FIFOs still overflow after the hblank.
5	R/W	0x0	LC_INT_EN Line counter flag

			The bit is set when the specific line has been written to dram every frame.The line number is set in the line counter register.
4	R/W	0x0	FIFO2_OF_INT_EN FIFO 2 overflow The bit is set when the FIFO 2 become overflow.
3	R/W	0x0	FIFO1_OF_INT_EN FIFO 1 overflow The bit is set when the FIFO 1 become overflow.
2	R/W	0x0	FIFO0_OF_INT_EN FIFO 0 overflow The bit is set when the FIFO 0 become overflow.
1	R/W	0x0	FD_INT_EN Frame done Indicates the CSI has finished capturing an image frame. Applies to video capture mode. The bit is set after each completed frame capturing data is wrote to buffer as long as video capture remains enabled.
0	R/W	0x0	CD_INT_EN Capture done Indicates the CSI has completed capturing the image data. For still capture, the bit is set when one frame data has been wrote to buffer. For video capture, the bit is set when the last frame has been wrote to buffer after video capture has been disabled. For CCIR656 interface, if the output format is frame planar YCbCr 420 mode, the frame end means the field2 end, the other frame end means field end.

7.1.5.46 CSIC DMA Interrupt Status Register(Default Value:0x0000_0000)

Offset: 0x0054			Register Name: CSIC_DMA_INT_STA_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R/W1C	0x0	VS_PD vsync flag
6	R/W1C	0x0	HB_OF_PD Hblank FIFO overflow
5	R/W1C	0x0	LC_PD Line counter flag
4	R/W1C	0x0	FIFO2_OF_PD FIFO 2 overflow
3	R/W1C	0x0	FIFO1_OF_PD FIFO 1 overflow
2	R/W1C	0x0	FIFO0_OF_PD FIFO 0 overflow
1	R/W1C	0x0	FD_PD Frame done

0	R/W1C	0x0	CD_PD Capture done

7.1.5.47 CSIC DMA Line Counter Register(Default Value:0x0000_0000)

Offset: 0x0058			Register Name: CSIC_DMA_LINE_CNT_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:13	/	/	/
12:0	R/W	0x0	LINE_CNT_NUM The LINE_CNT_NUM value is set by user,when internal line counter reach the set value,the LC_PD will set.

7.1.5.48 CSIC DMA Frame Clock Counter Register(Default Value:0x0000_0000)

Offset: 0x0060			Register Name: CSIC_DMA_FRM_CLK_CNT_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	/	/	/
23:0	R	0x0	FRM_CLK_CNT Counter value between every frame. For instant hardware frame rate statics. The internal counter is added by one every 12MHz clock cycle. When frame done or vsync comes, the internal counter value is sampled to FRM_CLK_CNT, and cleared to 0.

7.1.5.49 CSIC DMA Accumulated and Internal Clock Counter Register(Default Value:0x0000_0000)

Offset: 0x0064			Register Name: CSIC_DMA_ACC_ITNL_CLK_CNT_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	R/W1C	0x0	ACC_CLK_CNT The accumulated value of FRM_CLK_CNT for software frame rate statics. Every interrupt of frame done, the software check this accumulated value and clear it to 0. If the ACC_CLK_CNT is larger than 1, the software has lost frame. When frame done or vsync comes, ACC_CLK_CNT = ACC_CLK_CNT + 1, and cleared to 0 when writing 0 to this register.
23:0	R	0x0	ITNL_CLK_CNT The instant value of internal frame clock counter. When frame done interrupt comes, the software can query this counter for judging whether it is the time for updating the double buffer address registers.

7.1.5.50 CSIC DMA FIFO Statistic Register(Default Value:0x0000_0000)

Offset: 0x0068			Register Name: CSIC_DMA_FIFO_STAT_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:12	/	/	/
11:0	R	0x0	FIFO_FRM_MAX Indicates the maximum depth of FIFO being occupied for whole frame. Update at every vsync or framedone.

7.1.5.51 CSIC DMA FIFO Threshold Register(Default Value:0x0000_2400)

Offset: 0x006C			Register Name: CSIC_DMA_FIFO_THRS_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:13	R/W	0x1	FIFO_NEARLY_FULL_TH The threshold of FIFO being nearly full. Indicates that the ISP should stop writing. Only valid when ISP is enabled. 0~7: The smaller the value, the flag of FIFO being nearly full is easier to reach.
12	/	/	/
11:0	R/W	0x400	FIFO_THRS When FIFO occupied memory exceed the threshold, dram frequency can not change.

7.1.5.52 CSIC DMA PCLK Statistic Register(Default Value:0x0000_7FFF)

Offset: 0x0070			Register Name: CSIC_DMA_PCLK_STAT_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31	/	/	/
30:16	R	0x0	PCLK_CNT_LINE_MAX Indicates maximum pixel clock counter value for each line. Update at every vsync or framedone.
15	/	/	/
14:0	R	0x7FFF	PCLK_CNT_LINE_MIN Indicates minimum pixel clock counter value for each line. Update at every vsync or framedone.

7.1.5.53 CSIC DMA Feature List Register(Default Value:0x0000_0001)

Offset: 0x01F0			Register Name: CSIC_FEATURE_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R	0x0	DMA0_EMBEDDED_FBC

			0: only DMA

7.1.5.54 CCI Control Register(Default Value:0x0000_0000)

Offset: 0x0000			Register Name: CCI_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	SINGLE_TRAN 0: Transmission idle 1: Start single transmission Automatically cleared to '0' when finished. Abort current transmission immediately if changing from '1' to '0'. If slave not respond for the expected status over the time defined by TIMEOUT, current transmission will stop. PACKET_CNT will return the sequence number when transmission fail. All format setting and data will be loaded from registers and FIFO when transmission start.
30	R/W	0x0	REPEAT_TRAN 0: transmission idle 1: repeated transmission When this bit is set to 1, transmission repeats when trigger signal (such as VSYNC/ VCAP done) repeats. If changing this bit from '1' to '0' during transmission, the current transmission will be guaranteed then stop.
29	R/W	0x0	RESTART_MODE 0: RESTART 1: STOP+START Define the CCI action after sending register address.
28	R/W	0x0	READ_TRAN_MODE 0: send slave_id+W 1: do not send slave_id+W **Setting this bit to 1 if reading from a slave which register width is equal to 0**.
27:24	R	0x0	TRAN_RESULT 000: OK 001: FAIL Other: Reserved
23:16	R	/	CCI_STA 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 9th SCL clk Other: Reserved
15:2	/	/	/
1	R/W	0x0	SOFT_RESET 0: normal 1: reset
0	R/W	0x0	CCI_EN 0: Module disable 1: Module enable

7.1.5.55 CCI Transmission Configuration Register(Default Value:0x1000_0000)

Offset: 0x0004			Register Name: CCI_CFG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	R/W	0x10	TIMEOUT_N When sending the 9th clock, assert fail signal when slave device did not response after **N*F_{SCL}** cycles. And software must do a reset to CCI module and send a stop condition to slave.
23:16	R/W	0x0	INTERVAL Define the interval between each packet in 40*F_{SCL} cycles. 0~255
15	R/W	0x0	PACKET_MODE Select where to load slave id / data width 0: Compact mode 1: Complete mode In compact mode, slave id/register width / data width will be loaded from CCI_FMT register, only address and data read from memory. In complete mode, they will be loaded from packet memory.
14:7	/	/	/
6:4	R/W	0x0	TRIG_MODE Transmit mode: 000: Immediately, no trigger 001: Reserved 010: CSI0 int trigger 011: CSI1 int trigger
3:0	R/W	0x0	CSI_TRIG CSI Int trig signal select: 0000: First HREF start 0001: Last HREF done 0010: Line counter trigger other: Reserved

7.1.5.56 CCI Packet Format Register(Default Value:0x0011_0001)

Offset: 0x0008			Register Name: CCI_FMT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:25	R/W	0x0	SLV_ID 7bit address
24	R/W	0x0	CMD 0: write 1: read
23:20	R/W	0x1	ADDR_BYTE How many bytes be sent as address 0~15
19:16	R/W	0x1	DATA_BYTE How many bytes be sent/received as data 1~15 Normally use ADDR_DATA with 0_2, 1_1, 1_2, 2_1, 2_2 access mode. If DATA bytes is 0, transmission will not start. In complete mode, the ADDR_BYTE and DATA_BYTE is defined in a byte's high/low 4bit.
15:0	R/W	0x1	PACKET_CNT FIFO data be transmitted as PACKET_CNT packets in current format. Total bytes not exceed 32bytes.

7.1.5.57 CCI Bus Control Register(Default Value:0x0000_2500)

Offset: 0x000C			Register Name: CCI_BUS_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	R/W	0x0	DLY_CYC 0~65535 F_{SCL} cycles between each transmission
15	R/W	0x0	DLY_TRIG 0: disable 1: execute transmission after internal counter delay when triggered
14:12	R/W	0x2	CLK_N CCI bus sampling clock F0=24MHz/2^CLK_N
11:8	R/W	0x5	CLK_M CCI output SCL frequency is F_{SCL}=F1/10=(F0/(CLK_M+1))/10
7	R	/	SCL_STA SCL current status
6	R	/	SDA_STA SDA current status
5	R/W	0x0	SCL_PEN SCL PAD enable
4	R/W	0x0	SDA_PEN SDA PAD enable
3	R/W	0x0	SCL_MOV SCL manual output value

2	R/W	0x0	SDA_MOV SDA manual output value
1	R/W	0x0	SCL_MOE SCL manual output en
0	R/W	0x0	SDA_MOE SDA manual output en

7.1.5.58 CCI Interrupt Control Register(Default Value:0x0000_0000)

Offset: 0x0014			Register Name: CCI_INT_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17	R/W	0x0	S_TRAN_ERR_INT_EN
16	R/W	0x0	S_TRAN_COM_INT_EN
15:2	/	/	/
1	R/W1C	0x0	S_TRAN_ERR_PD
0	R/W1C	0x0	S_TRAN_COM_PD

7.1.5.59 CCI Line Counter Trigger Control Register(Default Value:0x0000_0000)

Offset: 0x0018			Register Name: CCI_LC_TRIG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:13	/	/	/
12:0	R/W	0x0	LN_CNT 0~8191: line counter send trigger when 1st~8192th line is received.

7.1.5.60 CCI FIFO Access Register(Default Value:0x0000_0000)

Offset: 0x0100~0x013f			Register Name: CCI_FIFO_ACC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	DATA_FIFO From 0x100 to 0x13f, CCI data fifo is 64bytes, used in fifo input mode. CCI transmission read/write data from/to fifo in byte.

7.1.5.61 CCI Reserved Register(Default Value:0x0000_0000)

Offset: 0x0200~0x0220			Register Name: CCI_RSV_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	/	From 0x200 to 0x220 address, normal TWI registers are copied here. All transmission will be act like hardware controlling these registers. And don't change them in transmission.

Figures

8 ISP

8.1 Overview

The ISP module supports real time image process for RAW sensor. The main functions are as follows.

- Crop
- Black level correction(BLC)
- Digital gain
- Dynamic defect pixel correction(DPC)
- Crosstalk correction
- 2D denoise
- Contrast enhancement
- Auto balance
- Auto exposure
- Auto focus
- 3A statistic output
- Lens shading correction
- Demosaic
- Picture sharpen
- Color management and enhancement
- Dynamic range compression(DRC)
- Gamma correction
- High-precision scale down
- Defog

The processing capability of the ISP module is as follows.

- Supports 8/10/12 bits RAW data input
- ISP: maximum picture resolution of 4224x3168
- ISP: maximum frame rate of 3264x2448@30fps
- Minimum picture resolution of 192x128
- Minimum horizontal blanking region of 96 pixels
- Minimum vertical blanking region of 32 lines

8.2 Block Diagram

The block diagram of the ISP module is as follows.

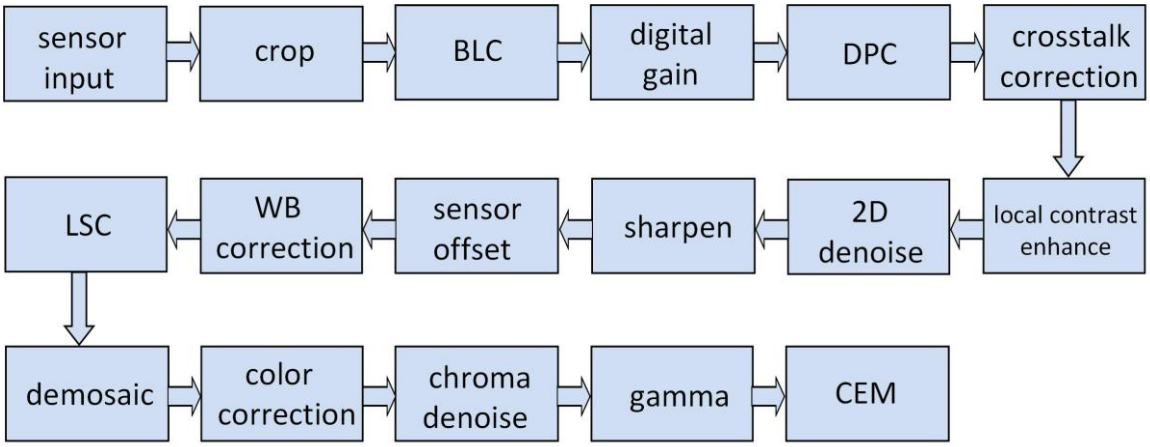


Figure8- 1. ISP Block Diagram

8.3 Module Functions

8.3.1 Crop

The Crop module, which can crop the input picture, is often used to change the aspect ratio of the input picture or extract the region of interest for a picture. For details about the configuration methods, see the <<*image sensor light guide*>>.

8.3.2 BLC

The BLC module adds respectively offset for four Bayer color channels(R,Gr,Gb,and B), to perform optical black correction. The precision of the offset is S13. The module is usually used for wide dynamic sensor.

8.3.3 Digital Gain

The Digital Gain module provides the digital gain, and supports U12Q10-precision .

8.3.4 DPC

The DPC module is used to correct defect pixels in Bayer field. There are two modes to be chosen, normal mode and strong mode. In normal mode, single defect pixel can be corrected. In strong mode, neighbor defect pixels can be corrected.

8.3.5 Crosstalk Correction

The Crosstalk Correction module is used to remove abnormal picture question when Gr and Gb imbalance. Usually, the module is used when sensor CRA unmatches lens CRA.

8.3.6 Contrast Enhance

Before denoising, the Contrast Enhance module implements picture edge sharpening to improve picture edge information, while picture contour is much clearer. But noises may be amplified if the sharpening strength is too high, note that there need to be a balance between the noise and sharpening strength.

8.3.7 2D Denoise

The 2D Denoise module restrains sensor noises in the Bayer field to improve picture quality.

8.3.8 Sharpen

After denoising, the Sharpen module implements picture edge sharpening to improve picture edge information, while picture contour is much clearer.

8.3.9 Sensor Offset

The Sensor Offset module adds respectively offset for four Bayer color channels(R,Gr,Gb,and B) to perform optical black correction. The precision of the offset is S13.

8.3.10 WB Correction

The WB Correction module adds respectively gain for four channels(R,Gr,Gb,and B) to implement white balance correction. The precision of the gain is U12Q8.

8.3.11 LSC

The LSC module implements lens shading correction. According to the radial position of each pixel within picture, the module can lookup R,G,B table to obtain compensation gain and compensate the reduced luminance from picture central to edge. The size of each gain table is 256, and the precision is U12Q10.

8.3.12 Demosaic

The Demosaic module interpolates Bayer field pixel to RGB field while holding clear picture edge and restraining pseudo color.

8.3.13 Color Correction

The Color Correction module applies a 3x3 color gain matrix and a 3x1 offset matrix on the input R/G/B pixels to restore image color. The precision of each value in gain matrix is S12Q8 and the precision of each value in offset matrix is S13.

8.3.14 Chroma Denoise

The Chroma Denoise module restrains color noises of sensor to improve picture quality.

8.3.15 Gamma

The Gamma module applies gamma correction for each color channel(R,G,and B) through looking-up table. Each gamma table has 256 entries and the precision is U12.

8.3.16 CEM

The CEM module adjusts hue and saturation of picture in YUV field, and enhances or restrains specific colors such as blue sky, plant and complexion based on user preference.

Figures

Tables

9 Audio

9.1 I2S/PCM

9.1.1 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:

- Compliant with standard Philips Inter-IC sound (I2S) bus specification
- Compliant with Left-justified, Right-justified, PCM mode, and TDM (Time Division Multiplexing) format
- Full-duplex synchronous work mode
- Master/Slave mode
- Adjustable interface voltage
- Clock up to 24.576MHz
- Adjustable audio sample resolution from 8-bit to 32-bit
- Up to 16 channel(fs = 48 kHz) which has adjustable width from 8-bit to 32-bit
- Sample rate from 8 kHz to 384 kHz(CHAN = 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
- Interrupt and DMA support
- Supports loop back mode for test

9.1.2 Block Diagram

The block diagram of I2S/PCM interface is shown below.

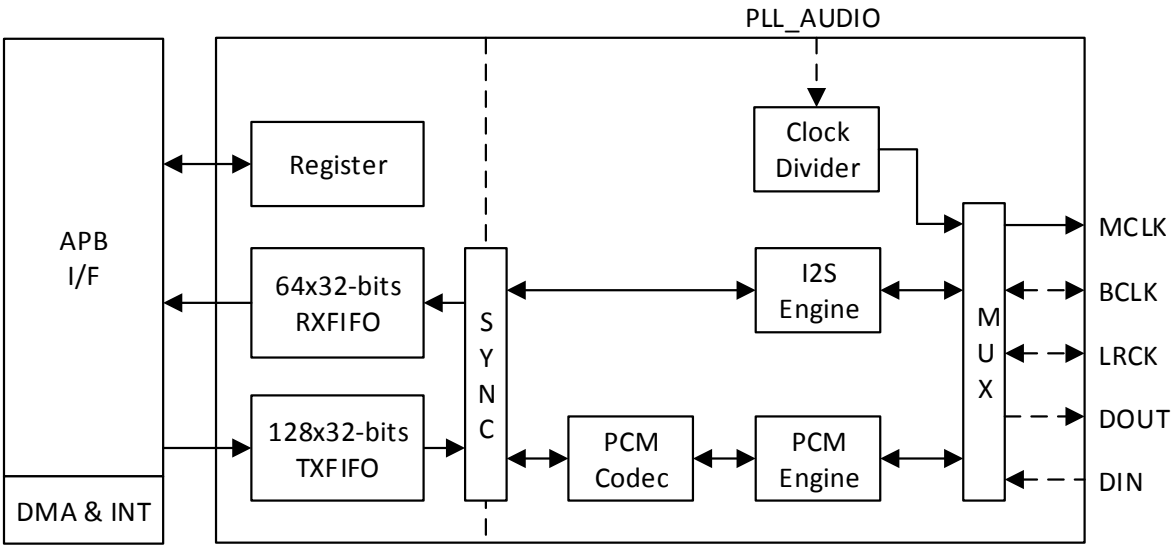


Figure9- 1. I2S/PCM Interface System Block Diagram

9.1.3 Operations and Functional Descriptions

9.1.3.1 External Signals

Table 9-1 describes the external signals of I2S/PCM interface. BCLK and LRCK are bidirectional I/O, when I2S/PCM interface is configured as master device, BCLK and LRCK is output pin; when I2S/PCM interface is configured as slave device, BCLK and LRCK is input pin. MCLK is an output pin for external device. DOUT is always the serial data output pin, and DIN is the serial data input. For information about General Purpose I/O port, see Port Controller.

Table9- 1. I2S/PCM External Signals

Signal Name(x=[1:0])	Description	Type
I2Sx_MCLK	I2S/PCM x Master Clock	O
I2Sx_BCLK	I2S/PCM x Sample Rate Serial Clock	I/O
I2Sx_LRCK	I2S/PCM x Sample Rate Left and Right Channel Select Clock/Sync	I/O
I2Sx_DIN	I2S/PCM x Serial Data Input	I
I2Sx_DOUT	I2S/PCM x Serial Data Output	O

9.1.3.2 Clock Sources

Table 9-2 describes the clock sources for I2S/PCM. Users can see **Chapter 3.3.CCU** for clock setting, configuration and gating information.

Table9- 2. I2S/PCM Clock Sources

Clock Name	Description
PLL_AUDIO	24.576 MHz or 22.5792 MHz generated by PLL_AUDIO to produce 48 kHz or 44.1 kHz serial frequency

9.1.3.3 Typical Application

One I2S/PCM provides one TX and one RX, simultaneously transmits and receives 8-,12-,16-,20-,24-,28- or 32-bit data

from 1,2,4,6,8,or 16 channels. The following figures describe the typical connections over the I2S/PCM interface in master or slave mode.

In master mode, Figure 9-2 and Figure 9-3 show the typical connections over the I2S/PCM interface.

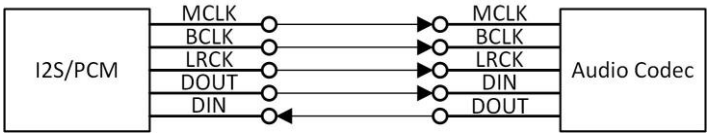


Figure9- 2. I2S/PCM Master Mode1

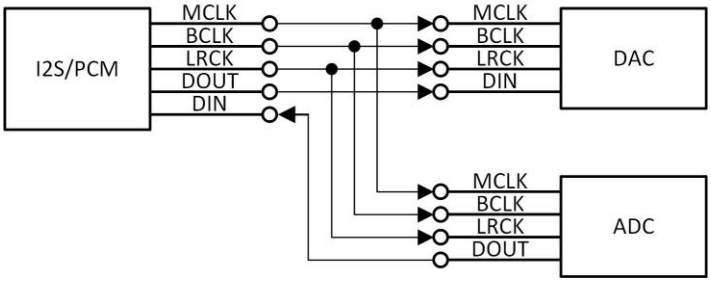


Figure9- 3. I2S/PCM Master Mode2

In slave mode, Figure 9-4 and Figure 9-5 show the typical connections over the I2S/PCM interface.

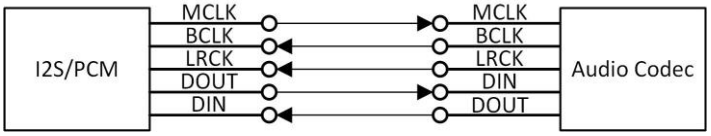


Figure9- 4. I2S/PCM Slave Mode1

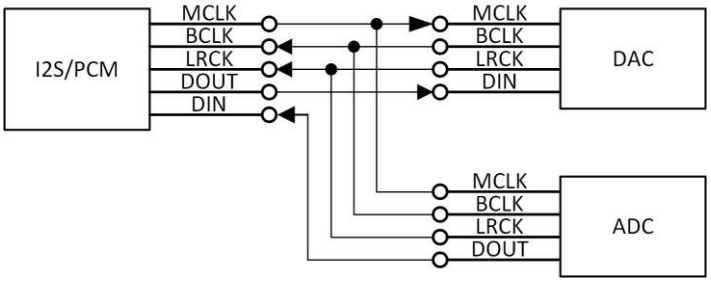


Figure9- 5. I2S/PCM Slave Mode2

9.1.3.4 Timing Diagram

The I2S/PCM supports standard I2S mode,Left-justified I2S mode,Right-justified I2S mode,PCM mode and TDM mode. Software can select any modes by setting the **I2S/PCM Control Register**.Figure 9-6 to Figure 9-10 describe the waveforms for LRCK,BCLK and DOUT,DIN.

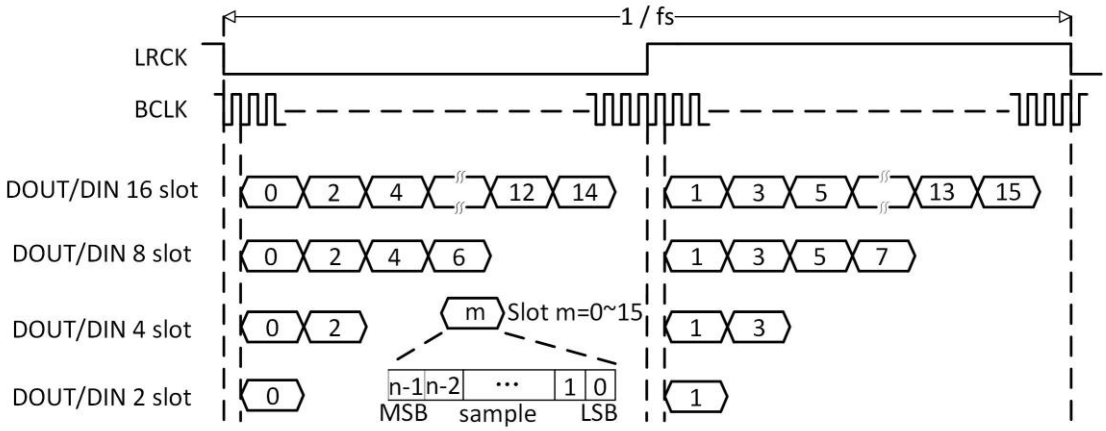


Figure9- 6. I2S Standard Mode Timing

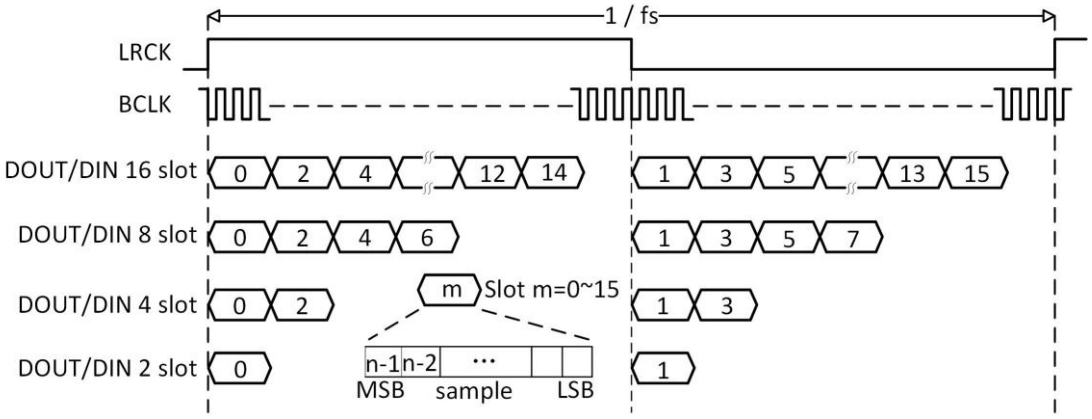


Figure9- 7. Left-Justified Mode Timing

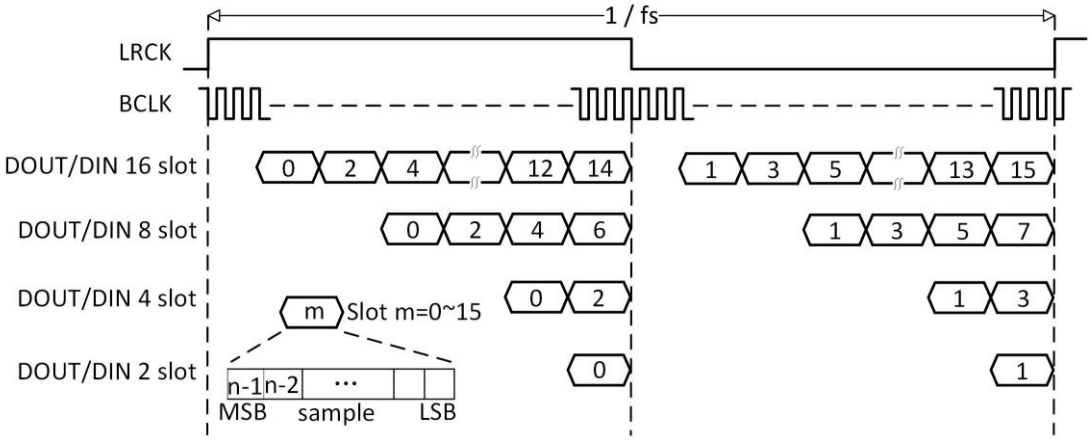


Figure9- 8. Right-Justified Mode Timing

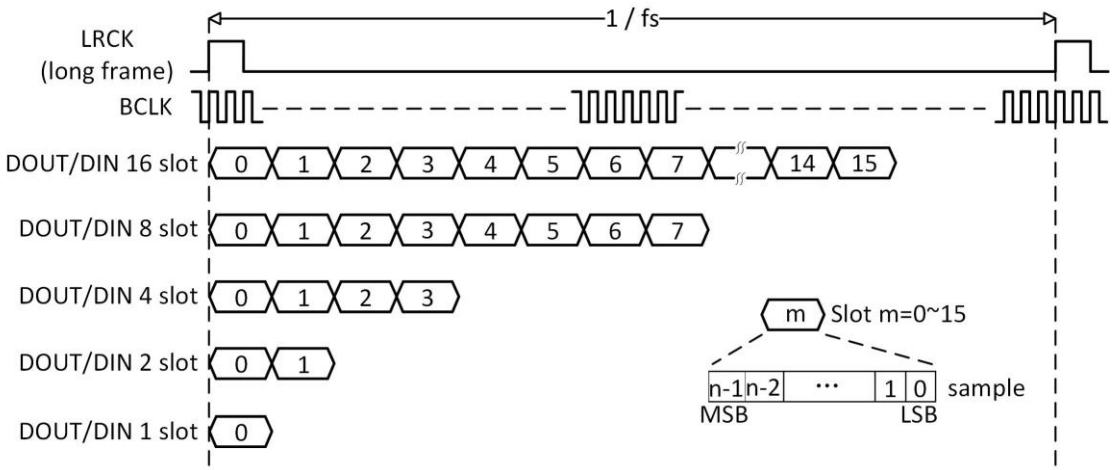


Figure9- 9. PCM Long Frame Mode Timing

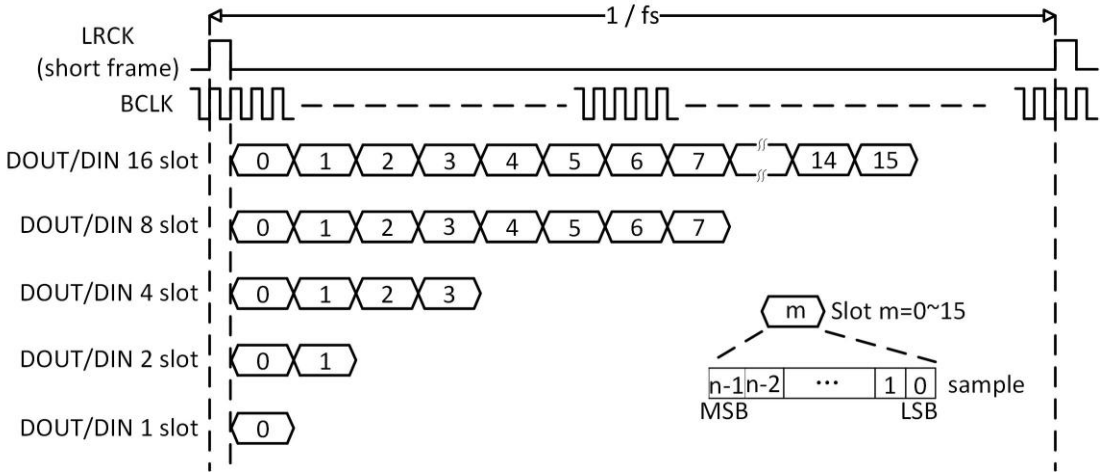


Figure9- 10. PCM Short Frame Mode Timing

9.1.3.5 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. These five steps are described in detail in the following sections.

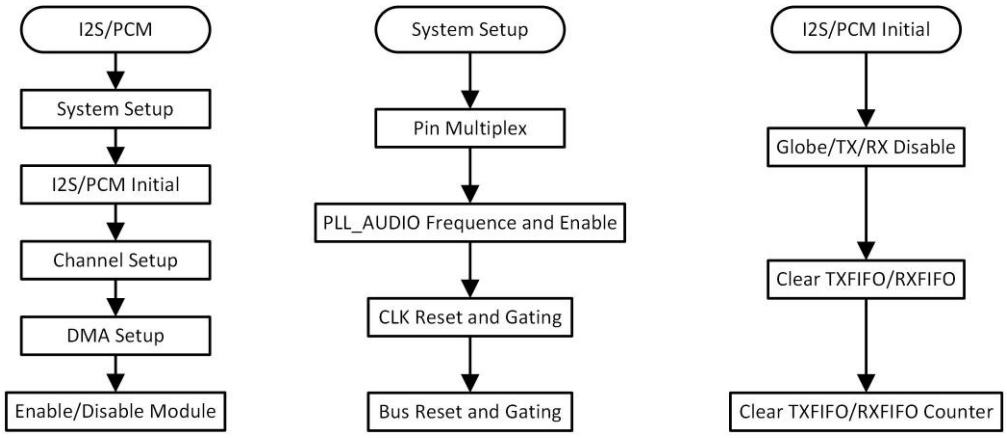


Figure9- 11. I2S/PCM Operation Flow

(1). System setup and I2S/PCM initialization

The clock source for the I2S/PCM should be followed. At first you must disable the PLL_AUDIO through the **PLL_ENABLE** bit of **PLL_AUDIO_CTRL_REG** in the CCU. The second step, you must set up the frequency of the PLL_AUDIO in the **PLL_AUDIO_CTRL_REG**. After that, you must open the I2S/PCM gating through the **I2S/PCM_CLK_REG** when you checkout that the LOCK bit of **PLL_AUDIO_CTRL_REG** becomes to 1. At last, you must reset and open the I2S/PCM bus gating in the **CCU_I2S_BGR_REG**.

After the system setup, the register of I2S/PCM can be setup. At first, you should initialization 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, you must clear the TX/RX FIFO by writing 0 to the bit[25:24] of **I2S/PCM_FCTL**. At last, you can clear the TX FIFO and RX FIFO counter by writing 0 to **I2S/PCM_TXCNT** and **I2S/PCM_RXCNT**.

(2). Channel setup and DMA setup

First, you can setup the I2S/PCM of mater 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 slot, the channel slot number and the trigger level and so on. The setup of register can be found in the specification.

The I2S/PCM supports two methods to transfer the data. The most common way is DMA, the setup of DMA can be found in the **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 **the I2S/PCM_CT**L[2:1]. After that, you must enable I2S/PCM by writing the **Globe Enable** bit to 1 in the I2S/PCM_CTL. Write the **Globe Enable** to 0 to disable I2S/PCM.

9.1.4 Programming Guidelines

The following example assumes that the audio channels are stereo channels in I2S mode, the sampling rate is 48 kHz, the sampling precision is 16 bits. The recording and playback processes are as follows.

----------GPIO configuration----------

Step1: Ensure that I2S/PCM0 GPIO has power supply.

Step2: Configure GPIOB4/GPIOB5/GPIOB6/GPIOB7/GPIOB8 as Function3.

----------Clock configuration----------

Step1: Configure PLL_AUDIO as 24.576MHz, that is, set **PLL_AUDIO Control Register** to 0xA10D0D00, set **PLL_AUDIO Pattern0 Register** to 0xC000AC02 .(If PLL_AUDIO is set as 22.5792MHz,that is, set **PLL_AUDIO Control Register** to 0xA1171500, set **PLL_AUDIO Pattern0 Register** to 0xC001288D).

Step2: Check whether **PLL_AUDIO Control Register**[PLL_AUDIO_LOCK] is 1. If is 1, set **I2S/PCM0 Clock Register** to 0x80000000.

Step3: Write 1 to the bit16 of **I2S/PCM0 Bus Gating Reset Register** to dessert I2S/PCM0 reset.

Step4: Write 1 to the bit0 of **I2S/PCM0 Bus Gating Reset Register** to open I2S/PCM0 gating.

NOTE

Step3 and Step4 is set separately.

----------Initialization I2S/PCM----------

Step1: Set the bit[2:0] of **I2S/PCM Control Register** to 0 to close TXEN,RXEN and GEN.

Step2: Set the bit[25:24] of **I2S/PCM FIFO Control Register** to 0x11 to clear TXFIFO and RXFIFO.

Step3: Set **I2S/PCM TX Counter Register** to 0 to clear TX counter, set **I2S/PCM RX Counter Register** to 0 to clear RX counter.

----------Format configuration----------

Step1: Master/slave configuration. In master mode, the bit[18:17] of **I2S/PCM RX Counter Register** is set to 0x11; in slave mode, the bit[18:17] of **I2S/PCM RX Counter Register** is set to 0.

Step2: Configure the[5:4] of **I2S/PCM Control Register** to 1 to set standard I2S mode, configure the bit[21:20] of **I2S/PCM TX Channel Select Register** to 1, configure the bit[21:20] of **I2S/PCM RX Channel Select Register** to 1.

Step3: Configure the bit[6:4] of **I2S/PCM Format Register0** to 3 to set sample resolution, configure the bit[2:0] of **I2S/PCM Format Register0** to 3 to set channel width.

Step4: Configure the bit[7:4] of **I2S/PCM Channel Configuration Register** to 1 to set RX channel number, configure the bit[3:0] of **I2S/PCM Channel Configuration Register** to 1 to set TX channel number. Configure the bit[19:16] of **I2S/PCM TX Channel Select Register** to 1, configure the bit[1:0] of **I2S/PCM TX Channel Select Register** to 3. Configure the bit[19:16] of **I2S/PCM RX Channel Select Register** to 1.

Step5: Configure the bit[7:0] of **I2S/PCM TX Channel Mapping Register 1** to 0x10000, configure the bit[7:0] of **I2S/PCM RX Channel Mapping Register 1** to 0x10000.

----------Clock divider configuration----------

Step1: Set MCLK divider. Configure the bit[3:0] of **I2S/PCM Clock Divide Register** to 1, that is, MCLK=24.576MHz. Configure the bit8 of **I2S/PCM Clock Divide Register** to 1 to enable MCLK.

Step2: Set BCLK divider. Configure the bit[7:4] of **I2S/PCM Clock Divide Register** to 0x7, that is, BCLK=Sample ratio*Slot_Width*Slot_Num=48K*16*2=1.536MHz.

Step3: Set LRCK divider. Configure the bit[17:8] of **I2S/PCM Format Register** to 0xF, that is, N-1=BCLK/Sample ratio/ Slot_Num =16，N=15.

----------DMA configuration----------

Step1: Set data width of both DMA_SRC and DMA_DEST to 16-bit.

Step2: Set DMA BLOCK SIZE,DMA_SRC BLOCK SIZE and DMA_DEST BLOCK SIZE to 8.

Step3: TX DMA configuration. Set DMA_SRC_DRQ_TYPE to DRAM, set DMA_SRC_ADDR_MODE to Linear Mode, set DMA_DEST_DRQ_TYPE to I2S/PCM0-TX, set DMA_DEST_ADDR_MODE to IO Mode, set DMA_SRC_ADDR to DRAM address of storing data, set DMA_DEST_ADDR to I2S/PCM TXFIFO(0x05090020).

Step4: RX DMA configuration. Set DMA_SRC_DRQ_TYPE to I2S/PCM0-RX, set DMA_SRC_ADDR_MODE to IO Mode, set DMA_DEST_DRQ_TYPE to DRAM, set DMA_DEST_ADDR_MODE to Linear Mode, set DMA_SRC_ADDR to I2S/PCM RXFIFO(0x05090010), set DMA_DEST_ADDR to DRAM address of storing data.

For more details about DMA, please see to DMA in section 3.10.

NOTE

If data is stored in SRAM, then DRAM is modified to SRAM.

----------Recording/playback/pause----------

Step1: Enable globe, set the bit0 of **I2S/PCM Control Register** to 1.

Step2: Recording start: set the bit1 of **I2S/PCM Control Register** to 1, set the bit3 of **I2S/PCM DMA & Interrupt Control Register** to 1.

Step3: Playback start: set the bit2 of **I2S/PCM Control Register** to 1, set the bit7 of **I2S/PCM DMA & Interrupt Control Register** to 1.

Step4: Recording pause: set the bit1 of **I2S/PCM Control Register** to 0, set the bit3 of **I2S/PCM DMA & Interrupt**

Control Register to 0.

Step5: Playback pause: set the bit2 of **I2S/PCM Control Register** to 0, set the bit7 of **I2S/PCM DMA & Interrupt Control Register** to 0.

9.1.5 Register List

Module Name	Base Address
I2S/PCM0	0x05090000
I2S/PCM1	0x05091000

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_TXCHCFG	0x0034	I2S/PCM TX Channel Configuration Register
I2S/PCM_TXCHMAP0	0x0044	I2S/PCM TX Channel Mapping Register0
I2S/PCM_TXCHMAP1	0x0048	I2S/PCM TX 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

9.1.6 Register Description

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

Offset: 0x0000			Register Name: I2S/PCM_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:19	/	/	/
18	R/W	0x1	BCLK_OUT 0: Input 1: Output
17	R/W	0x1	LRCK_OUT 0: Input 1: Output

16:9	/	/	/
8	R/W	0x0	DOUT_EN 0: Disable, Hi-Z State 1: Enable
7	/	/	/
6	R/W	0x0	OUT_MUTE 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 Loop Back Test 0: Normal Mode 1: Loop Back Test When setting to '1' , the bit indicates that the DOUT connects to the DIN.
2	R/W	0x0	TXEN Transmitter Block Enable 0: Disable 1: Enable
1	R/W	0x0	RXEN Receiver Block Enable 0: Disable 1: Enable
0	R/W	0x0	GEN Globe Enable 0: Disable 1: Enable

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

Offset: 0x0004			**Register Name: I2S/PCM_FMT0**
Bit	**Read/Write**	**Default/Hex**	**Description**
31	/	/	/
30	R/W	0x0	LRCK_WIDTH (Only Apply in PCM Mode) LRCK Width 0: LRCK = 1 BCLK Width (Short Frame) 1: LRCK = 2 BCLK Width (Long Frame)
29:20	/	/	/
19	R/W	0x0	LRCK_POLARITY 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 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 individual channel width (Left or Right) . N+1 For example: N = 7: 8 BCLKs width … N = 1023: 1024 BCLKs width
7	R/W	0x0	BCLK_POLARITY 0: Normal mode, DOUT drives data at negative edge 1: Invert mode, DOUT drives data at positive edge
6:4	R/W	0x3	SR Sample Resolution 000: Reserved 001: 8-bit 010: 12-bit 011: 16-bit 100: 20-bit 101: 24-bit 110: 28-bit 111: 32-bit
3	R/W	0x0	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

9.1.6.3 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

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

Offset: 0x000C			Register Name: I2S/PCM_ISTA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:7	/	/	/
6	R/W1C	0x0	TXU_INT

			TXFIFO Underrun Pending Interrupt 0: No pending interrupt 1: TXFIFO underrun pending interrupt Write '1' to clear this interrupt.
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/W1C	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 Write '1' to clear this interrupt or automatic clear if interrupt condition fails.
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/W1C	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 Write '1' to clear this interrupt or automatic clear if interrupt condition fails.

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

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

9.1.6.6 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 0 : Disable 1 : Enable
30:26	/	/	/
25	R/W1C	0x0	FTX Write '1' to flush TXFIFO, self clear to '0'.
24	R/W1C	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
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]}

9.1.6.7 I2S/PCM FIFO Status Register(Default Value: 0x1080_0000)

Offset: 0x0018			Register Name: I2S/PCM_FSTA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	/	/	/
28	R	0x1	TXE TXFIFO Empty 0: No room for new sample in TXFIFO 1: More than one room for new sample in TXFIFO (>= 1 Word)
27:24	/	/	/
23:16	R	0x80	TXE_CNT TXFIFO Empty Space Word Counter
15:9	/	/	/
8	R	0x0	RXA RXFIFO Available 0: No available data in RXFIFO 1: More than one sample in RXFIFO (>= 1 Word)
7	/	/	/
6:0	R	0x0	RXA_CNT RXFIFO available sample word counter

9.1.6.8 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: Disable 1: Enable
6	R/W	0x0	TXUI_EN TXFIFO Underrun Interrupt Enable 0: Disable 1: Enable
5	R/W	0x0	TXOI_EN TXFIFO Overrun Interrupt Enable 0: Disable 1: Enable When setting to ‘1’, an interrupt happens when writing new audio data if TXFIFO is full.
4	R/W	0x0	TXEI_EN TXFIFO Empty Interrupt Enable 0: Disable 1: Enable
3	R/W	0x0	RX_DRQ

			RXFIFO Data Available DRQ Enable 0: Disable 1: Enable When setting to '1', RXFIFO DMA request line is asserted if data is available in RXFIFO.
2	R/W	0x0	RXUI_EN RXFIFO Underrun Interrupt Enable 0: Disable 1: Enable
1	R/W	0x0	RXOI_EN RXFIFO Overrun Interrupt Enable 0: Disable 1: Enable
0	R/W	0x0	RXAI_EN RXFIFO Data Available Interrupt Enable 0: Disable 1: Enable

9.1.6.9 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.

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

Offset: 0x0024			Register Name: I2S/PCM_CLKD
Bit	**Read/Write**	**Default/Hex**	**Description**
31:9	/	/	/
8	R/W	0x0	MCLKO_EN 0: Disable MCLK Output 1: Enable MCLK Output 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
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

9.1.6.11 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 been updated by the initial value, the counter register should count on base of this initial value.

9.1.6.12 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 been updated by the initial value, the counter register should count on base of this initial value.

9.1.6.13 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
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

9.1.6.14 I2S/PCM TX Channel Select Register(Default Value: 0x0000_0000)

Offset: 0x0034			Register Name: I2S/PCM_TXCHSEL
Bit	**Read/Write**	**Default/Hex**	**Description**

31:22	/	/	/
21:20	R/W	0x0	TX_OFFSET TX offset Tune, TX Data offset to LRCK 0: No offset n: Data is offset by n BCLKs to LRCK
19:16	R/W	0x0	TX_CHSEL TX 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	TX_CHEN TX Channel (Slot) Enable, bit[15:0] refer to Slot [15:0]. When one or more Slot(s) is(are) disabled, the affected Slot(s) is(are) set to the disable state. 0: Disable 1: Enable

9.1.6.15 I2S/PCM TX Channel Mapping Register 0(Default Value: 0x0000_0000)

Offset: 0x0044			**Register Name: I2S/PCM_TXCHMAP**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x0	TX_CH15_MAP TX Channel 15 Mapping 0000: 1st Sample ... 0111: 8th Sample 1000: 9th Sample ... 1111: 16th Sample
27:24	R/W	0x0	TX_CH14_MAP TX Channel 14 Mapping 0000: 1st Sample ... 0111: 8th Sample 1000: 9th Sample ... 1111: 16th sample
23:20	R/W	0x0	TX_CH13_MAP TX Channel 13 Mapping 0000: 1st Sample ... 0111: 8th Sample

			1000: 9th Sample ... 1111: 16th Sample
19:16	R/W	0x0	TX_CH12_MAP TX Channel 12 Mapping 0000: 1st Sample ... 0111: 8th Sample 1000: 9th Sample ... 1111: 16th Sample
15:12	R/W	0x0	TX_CH11_MAP TX Channel 11 Mapping 0000: 1st Sample ... 0111: 8th Sample 1000: 9th Sample ... 1111: 16th sample
11:8	R/W	0x0	TX_CH10_MAP TX Channel 10 Mapping 0000: 1st Sample ... 0111: 8th Sample 1000: 9th Sample ... 1111: 16th Sample
7:4	R/W	0x0	TX_CH9_MAP TX Channel 9 Mapping 0000: 1st Sample ... 0111: 8th Sample 1000: 9th Sample ... 1111: 16th Sample
3:0	R/W	0x0	TX_CH8_MAP TX Channel 8 Mapping 0000: 1st Sample ... 0111: 8th Sample 1000: 9th Sample ... 1111: 16th Sample

9.1.6.16 I2S/PCM TX Channel Mapping Register 1(Default Value: 0x0000_0000)

Offset: 0x0048			Register Name: I2S/PCM_TXCHMAP
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x0	TX_CH7_MAP TX Channel 7 Mapping 0000: 1st Sample ... 0111: 8th Sample 1000: 9th Sample ... 1111: 16th Sample
27:24	R/W	0x0	TX_CH6_MAP TX Channel 6 Mapping 0000: 1st Sample ... 0111: 8th Sample 1000: 9th Sample ... 1111: 16th Sample
23:20	R/W	0x0	TX_CH5_MAP TX Channel 5 Mapping 0000: 1st Sample ... 0111: 8th Sample 1000: 9th Sample ... 1111: 16th Sample
19:16	R/W	0x0	TX_CH4_MAP TX Channel 4 Mapping 0000: 1st Sample ... 0111: 8th Sample 1000: 9th Sample ... 1111: 16th Sample
15:12	R/W	0x0	TX_CH3_MAP TX Channel 3 Mapping 0000: 1st Sample ... 0111: 8th Sample 1000: 9th Sample ... 1111: 16th Sample
11:8	R/W	0x0	TX_CH2_MAP TX Channel 2 Mapping

			0000: 1st Sample … 0111: 8th Sample 1000: 9th Sample … 1111: 16th Sample
7:4	R/W	0x0	TX_CH1_MAP TX Channel 1 Mapping 0000: 1st Sample … 0111: 8th Sample 1000: 9th Sample … 1111: 16th Sample
3:0	R/W	0x0	TX_CH0_MAP TX Channel 0 Mapping 0000: 1st Sample … 0111: 8th Sample 1000: 9th Sample … 1111: 16th Sample

9.1.6.17 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	/	/	/

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

Offset: 0x0068			Register Name: I2S/PCM_RXCHMAP0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x0	RX_CH15_MAP RX Channel 15 Mapping 0000: 1st Sample ... 0111: 8th Sample 1000: 9th Sample ... 1111: 16th Sample
27:24	R/W	0x0	RX_CH14_MAP RX Channel 14 Mapping 0000: 1st Sample ... 0111: 8th Sample 1000: 9th Sample ... 1111: 16th Sample
23:20	R/W	0x0	RX_CH13_MAP RX Channel 13 Mapping 0000: 1st Sample ... 0111: 8th Sample 1000: 9th Sample ... 1111: 16th Sample
19:16	R/W	0x0	RX_CH12_MAP RX Channel 12 Mapping 0000: 1st Sample ... 0111: 8th Sample 1000: 9th Sample ... 1111: 16th Sample
15:12	R/W	0x0	RX_CH11_MAP RX Channel 11 Mapping 0000: 1st Sample ... 0111: 8th Sample 1000: 9th Sample ... 1111: 16th Sample
11:8	R/W	0x0	RX_CH10_MAP RX Channel 10 Mapping

			0000: 1st Sample … 0111: 8th Sample 1000: 9th Sample … 1111: 16th Sample
7:4	R/W	0x0	RX_CH9_MAP RX Channel 9 Mapping 0000: 1st Sample … 0111: 8th Sample 1000: 9th Sample … 1111: 16th Sample
3:0	R/W	0x0	RX_CH8_MAP RX Channel 8 Mapping 0000: 1st Sample … 0111: 8th Sample 1000: 9th Sample … 1111: 16th Sample

9.1.6.19 I2S/PCM RX Channel Mapping Register 1(Default Value: 0x0000_0000)

Offset: 0x006C			Register Name: I2S/PCM_RXCHMAP
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x0	RX_CH7_MAP RX Channel 7 Mapping 0000: 1st Sample … 0111: 8th Sample 1000: 9th Sample … 1111: 16th Sample
27:24	R/W	0x0	RX_CH6_MAP RX Channel 6 Mapping 0000: 1st Sample … 0111: 8th Sample 1000: 9th Sample … 1111: 16th Sample
23:20	R/W	0x0	RX_CH5_MAP RX Channel 5 Mapping

			0000: 1st Sample … 0111: 8th Sample 1000: 9th Sample … 1111: 16th Sample
19:16	R/W	0x0	RX_CH4_MAP RX Channel 4 Mapping 0000: 1st Sample … 0111: 8th Sample 1000: 9th Sample … 1111: 16th Sample
15:12	R/W	0x0	RX_CH3_MAP RX Channel 3 Mapping 0000: 1st Sample … 0111: 8th Sample 1000: 9th Sample … 1111: 16th Sample
11:8	R/W	0x0	RX_CH2_MAP RX Channel 2 Mapping 0000: 1st Sample … 0111: 8th Sample 1000: 9th Sample … 1111: 16th Sample
7:4	R/W	0x0	RX_CH1_MAP RX Channel 1 Mapping 0000: 1st Sample … 0111: 8th Sample 1000: 9th Sample … 1111: 16th Sample
3:0	R/W	0x0	RX_CH0_MAP RX Channel 0 Mapping 0000: 1st Sample … 0111: 8th Sample 1000: 9th Sample … 1111: 16th Sample

9.2 DMIC

9.2.1 Overview

The DMIC controller supports a 8-channels digital microphone interface, the DMIC controller can output 128fs or 64fs (fs= ADC sample rate).

The DMIC controller includes the following features:

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

9.2.2 Block Diagram

Figure 9-12 shows a block diagram of the DMIC.

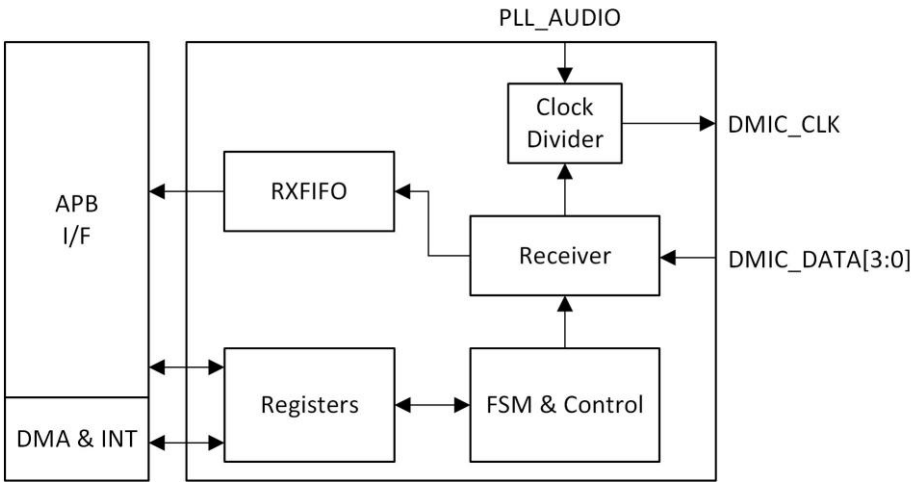


Figure9- 12. DMIC Block Diagram

9.2.3 Operations and Functional Descriptions

9.2.3.1 External Signals

Table 9-3 describes the external signals of DMIC.

Table9- 3. DMIC External Signals

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

9.2.3.2 Clock Sources

Table 9-4 describes the clock source for DMIC. Users can see **Clock Controller Unit(CCU)** for clock setting, configuration and gating information.

Table9- 4. DMIC Clock Sources

Clock Sources	Description
PLL_AUDIO	24.576 MHz or 22.5792 MHz generated by PLL_AUDIO to produce 48 kHz or 44.1 kHz serial frequency.

9.2.3.3 Operation Mode

The software operation of the DMIC is divided into five steps: system setup, DMIC initialization, the channel setup, DMA setup and Enable/Disable module. Five steps are described in detail in the following sections.

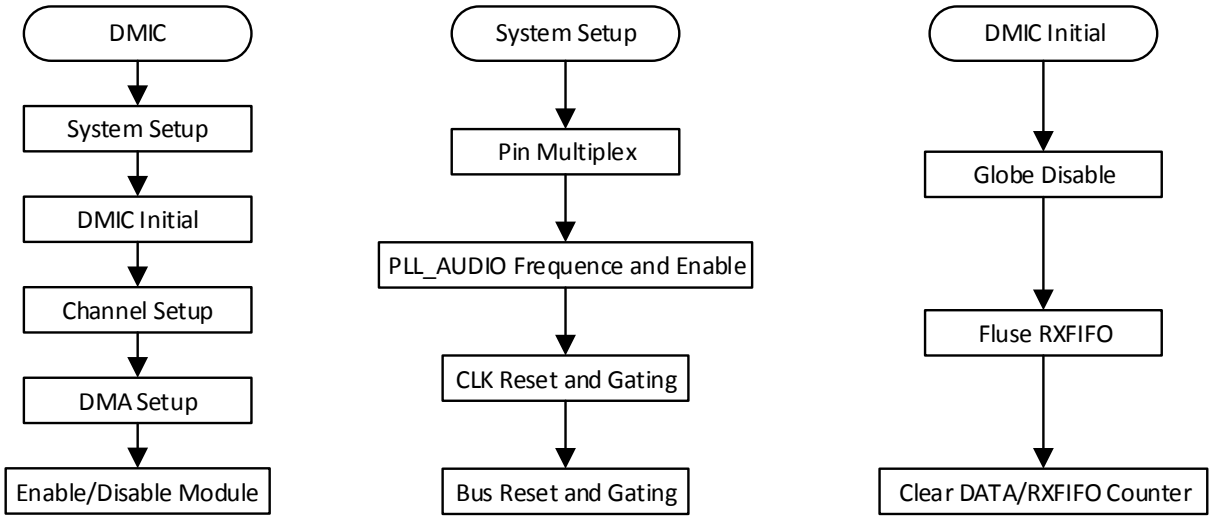


Figure9- 13. DMIC Operation Mode

9.2.3.3.1 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. You can find the function in the pin multiplex specification.

The clock source for the DMIC should be followed. At first you must disable the PLL_AUDIO through the PLL_ENABLE bit of **PLL_AUDIO_CTRL_REG** in the CCU. The second step, you must set up the frequency of the PLL_AUDIO in the **PLL_AUDIO_CTRL_REG**. Then enable PLL_AUDIO. After that, you must open the DMIC gating through the **DMIC_CLK_REG** when you checkout that the LOCK bit of **PLL_AUDIO_CTRL_REG** becomes 1. At last, you must reset and open the DMIC bus gating in the **CCU_DMIC_BGR_REG**.

After the system setup, the register of DMIC can be setup. At first, you should 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, you must flush the RXFIFO by writing 1 to register **DMIC_RXFIFO_CTR**[31]. At last, you can clear the Data/RXFIFO counter by writing 1 to **DMIC_RXFIFO_STA,DMIC_CNT**.

9.2.3.3.2 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 and the RXFIFO trigger level and so on. The setup of 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 the DMA specification. In this module, you just enable the DRQ.

9.2.3.3.3 Enable and Disable DMIC

To enable the function, you can enable **data channel enable bit**(DMIC_EN[7:0]) by writing 1 to it. After that, you must enable DMIC by writing the **Globe Enable bit** to 1 in the **DMIC_EN**[8]. Write the **Globe Enable** to 0 to disable DMIC.

9.2.4 Register List

Module Name	Base Address
DMIC	0x05095000

Register Name	Offset	Description
DMIC_EN	0x0000	DMIC Enable Control Register
DMIC_SR	0x0004	DMIC Sample Rate Register
DMIC_CTR	0x0008	DMIC Control Register
DMIC_ DATA	0x0010	DMIC Data Register
DMIC_INTC	0x0014	DMIC Interrupt Control Register
DMIC_INTS	0x0018	DMIC Interrupt Status Register
DMIC_RXFIFO_CTR	0x001C	DMIC RXFIFO Control Register
DMIC_RXFIFO_STA	0x0020	DMIC RXFIFO Status Register
DMIC_CH_NUM	0x0024	DMIC Channel Numbers Register
DMIC_CH_MAP	0x0028	DMIC Channel Mapping Register
DMIC_CNT	0x002C	DMIC Counter Register
DATA0_DATA1_VOL_CTR	0x0030	Data0 and Data1 Volume Control Register
DATA2_DATA3_VOL_CTR	0x0034	Data2 And Data3 Volume Control Register
HPF_EN_CTR	0x0038	High Pass Filter Enable Control Register
HPF_COEF_REG	0x003C	High Pass Filter Coef Register
HPF_GAIN_REG	0x0040	High Pass Filter Gain Register

9.2.5 Register Description

9.2.5.1 DMIC Enable Control Register (Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: DMIC_EN
Bit	**Read/Write**	**Default/Hex**	**Description**
31:9	/	/	/

8	R/W	0x0	GLOBE_EN DMIC Globe Enable 0: Disable 1: Enable
7	R/W	0x0	DATA3_CHR_EN DATA3 Right Channel Enable 0: Disable 1: Enable
6	R/W	0x0	DATA3_CHL_EN DATA3 Left Channel Enable 0: Disable 1: Enable
5	R/W	0x0	DATA2_CHR_EN DATA2 Right Channel Enable 0: Disable 1: Enable
4	R/W	0x0	DATA2_CHL_EN DATA2 Left Channel Enable 0: Disable 1: Enable
3	R/W	0x0	DATA1_CHR_EN DATA1 Right Channel Enable 0: Disable 1: Enable
2	R/W	0x0	DATA1_CHL_EN DATA1 Left Channel Enable 0: Disable 1: Enable
1	R/W	0x0	DATA0_CHR_EN DATA0 Right Channel Enable 0: Disable 1: Enable
0	R/W	0x0	DATA0_CHL_EN DATA0 Left Channel Enable 0: Disable 1: Enable

9.2.5.2 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.

9.2.5.3 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: 200ms 11: 30ms
8	R/W	0x0	DMICDFEN DMIC RXFIFO Delay Function for Writing Data after GLOBE_EN 0: Disable 1: Enable
7	R/W	0x0	DATA3 Left Data and Right Data Sweep Enable 0: Disable 1: Enable
6	R/W	0x0	DATA2 Left Data and Right Data Sweep Enable 0: Disable 1: Enable
5	R/W	0x0	DATA1 Left Data and Right Data Sweep Enable 0: Disable 1: Enable
4	R/W	0x0	DATA0 Left Data and Right Data Sweep Enable 0: Disable 1: Enable
3:1	/	/	/
0	R/W	0x0	DMIC Oversample Rate 0: 128 (Support 8 kHz ~ 24 kHz) 1: 64 (Support 16 kHz ~ 48 kHz)

9.2.5.4 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

9.2.5.5 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: Disable 1: Enable
1	R/W	0x0	RXFIFO_OVERRUN_IRQ_EN DMIC RXFIFO Overrun IRQ Enable 0: Disable 1: Enable
0	R/W	0x0	DATA_IRQ_EN DMIC RXFIFO Data Available IRQ Enable 0: Disable 1: Enable

9.2.5.6 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 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 interrupt condition fails.

9.2.5.7 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[23:0], 8'h0} Mode 1: Reserved For 16-bit received audio sample: Mode 0: RXDATA[31:0] = {RXFIFO_O[23:8], 16'h0} Mode 1: RXDATA[31:0] = {16{RXFIFO_O[23]}, RXFIFO_O[23:8]}
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

9.2.5.8 DMIC RXFIFO Status Register (Default Value: 0x0000_0000)

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

9.2.5.9 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)

9.2.5.10 DMIC Channel Mapping Register (Default Value: 0x7654_3210)

Offset: 0x0028			Register Name: DMIC_CH_MAP
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	R/W	0x7	DMIC_CH7_MAP DMIC Channel 7 Mapping 0000: DATA0 Left Channel 0001: DATA0 Right Channel 0010: DATA1 Left Channel 0011: DATA1 Right Channel 0100: DATA2 Left Channel 0101: DATA2 Right Channel 0110: DATA3 Left Channel 0111: DATA3 Right Channel
27:24	R/W	0x6	DMIC_CH6_MAP DMIC Channel 6 Mapping 0000: DATA0 Left Channel 0001: DATA0 Right Channel 0010: DATA1 Left Channel 0011: DATA1 Right Channel 0100: DATA2 Left Channel 0101: DATA2 Right Channel 0110: DATA3 Left Channel 0111: DATA3 Right Channel
23:20	R/W	0x5	DMIC_CH5_MAP DMIC Channel 5 Mapping 0000: DATA0 Left Channel 0001: DATA0 Right Channel 0010: DATA1 Left Channel 0011: DATA1 Right Channel 0100: DATA2 Left Channel 0101: DATA2 Right Channel 0110: DATA3 Left Channel 0111: DATA3 Right Channel
19:16	R/W	0x4	DMIC_CH4_MAP DMIC Channel 4 Mapping 0000: DATA0 Left Channel 0001: DATA0 Right Channel 0010: DATA1 Left Channel 0011: DATA1 Right Channel 0100: DATA2 Left Channel 0101: DATA2 Right Channel 0110: DATA3 Left Channel 0111: DATA3 Right Channel
15:12	R/W	0x3	DMIC_CH3_MAP DMIC Channel 3 Mapping

			0000: DATA0 Left Channel 0001: DATA0 Right Channel 0010: DATA1 Left Channel 0011: DATA1 Right Channel 0100: DATA2 Left Channel 0101: DATA2 Right Channel 0110: DATA3 Left Channel 0111: DATA3 Right Channel
11:8	R/W	0x2	DMIC_CH2_MAP DMIC Channel 2 Mapping 0000: DATA0 Left Channel 0001: DATA0 Right Channel 0010: DATA1 Left Channel 0011: DATA1 Right Channel 0100: DATA2 Left Channel 0101: DATA2 Right Channel 0110: DATA3 Left Channel 0111: DATA3 Right Channel
7:4	R/W	0x1	DMIC_CH1_MAP DMIC Channel 1 Mapping 0000: DATA0 Left Channel 0001: DATA0 Right Channel 0010: DATA1 Left Channel 0011: DATA1 Right Channel 0100: DATA2 Left Channel 0101: DATA2 Right Channel 0110: DATA3 Left Channel 0111: DATA3 Right Channel
3:0	R/W	0x0	DMIC_CH0_MAP DMIC Channel0 Mapping 0000: DATA0 Left Channel 0001: DATA0 Right Channel 0010: DATA1 Left Channel 0011: DATA1 Right Channel 0100: DATA2 Left Channel 0101: DATA2 Right Channel 0110: DATA3 Left Channel 0111: DATA3 Right Channel

9.2.5.11 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 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 been updated by the initial value, the counter register should count on base of this initial value It is used for Audio/Video Synchronization

9.2.5.12 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 (-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 (-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 (-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	DATA0R_VOL (-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

9.2.5.13 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 (-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	DATA3R_VOL (-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 (-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 (-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

9.2.5.14 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: Disable 1: Enable
6	R/W	0x0	HPF_DATA3_CHL_EN High Pass Filter DATA3 Left Channel Enable 0: Disable 1: Enable
5	R/W	0x0	HPF_DATA2_CHR_EN High Pass Filter DATA2 Right Channel Enable 0: Disable 1: Enable
4	R/W	0x0	HPF_DATA2_CHL_EN High Pass Filter DATA2 Left Channel Enable 0: Disable 1: Enable
3	R/W	0x0	HPF_DATA1_CHR_EN High Pass Filter DATA1 Right Channel Enable 0: Disable 1: Enable
2	R/W	0x0	HPF_DATA1_CHL_EN High Pass Filter DATA1 Left Channel Enable 0: Disable 1: Enable
1	R/W	0x0	HPF_DATA0_CHR_EN High Pass Filter DATA0 Right Channel Enable 0: Disable 1: Enable
0	R/W	0x0	HPF_DATA0_CHL_EN High Pass Filter DATA0 Left Channel Enable 0: Disable 1: Enable

9.2.5.15 High Pass Filter Coef Register (Default Value: 0x00FF_AA45)

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

9.2.5.16 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	High Pass Filter Gain

9.3 Audio Codec

9.3.1 Overview

The Audio Codec has 2 I2S/PCM interfaces, 2 channels DAC and 1 channels ADC with a high level of mixed-signal integration which ideal for smart phone and other portable devices.The Audio Codec integrates true-ground capless headphone driver to deliver high quality and power-efficient headphone playback without any requirement for DC block capacitors.The integrated hardware DAP engine capable of DRC can be used in record and playback paths.

The Audio Codec has the following features:

- Two audio digital-to-analog(DAC) channels
  - 20-bit sample resolution and 8 kHz to 192 kHz sample rate
  - 95±3dB SNR@A-weight, THD+N -80±3dB, output level 0.56 Vrms
  - DAC power consumption 3.1mA@1.8V
- One analog audio outputs:
  - One stereo headphone output(HPL and HPR)
- Capless stereo headphone driver
  - Headphone driver 10mW(CPVDD = 1.2V, 16Ω load)
  - 95±3dB SNR@A-weight, THD+N -55±3dB, output level 0.4Vrms@16Ω(CPVDD=1.2V)
  - Headphone power consumption 2.4mA@1.8V(AVCC), 100mA@0.9V(CPVDD)
- One audio analog-to-digital(ADC) channels
  - 20-bit sample resolution and 8 kHz to 48 kHz sample rate
  - 90±3dB SNR@A-weight, THD+N -77±3dB
  - ADC power consumption 5.5mA@1.8V
- Three analog audio inputs:
  - Two differential microphone inputs(MICXP and MICXN), with boost pre-amplifiers
  - One line-in input(LINEINR)
- Supports Dynamic Range Controller(DRC) adjusting the ADC recording output
- Supports Dynamic Range Controller(DRC) adjusting the DAC playback output
- Two low-noise analog microphone bias outputs
- Supports analog/digital volume control
- Analog low-power loop from line-in/microphone/ to headphone
- Accessory button press detection
- Two I2S/PCM interfaces
- One 128x24 bits FIFO for data transmit, one 64x24 bits FIFO for data receive
- Programmable FIFO thresholds
- Interrupt and DMA support

9.3.2 Operations and Functional Descriptions

9.3.2.1 Power and Signal Description

9.3.2.1.1 Analog I/O Pins

MICIN2P	I	Positive differential input for MIC2
MICIN2N	I	Negative differential input for MIC2
MICIN3P	I	Positive differential input for MIC3
MICIN3N	I	Negative differential input for MIC3
LINEINR	I	Right single-end input for LINE-IN
HPOUTL	O	Headphone amplifier left channel output
HPOUTR	O	Headphone amplifier right channel output
MIC_DET	I	Headphone MIC detect
HP_DET	I	Headphone Jack detect

9.3.2.1.2 Filter/Reference

MBIAS	O	First bias voltage output for main microphone
HBIAS	O	Second bias voltage output for headset microphone
HPOUTFB	I	Pseudo differential headphone ground feedback
VRA2	O	Internal reference voltage
REXT	O	External reference pin

9.3.2.1.3 Power/Ground

AVCC	P	Analog power 1.8V
AGND	G	Analog ground
CPVIN	P	Analog power for LDO
CPVDD	P	Analog power for headphone charge pump
CPVEE	P	Charge pump negative voltage output
VEE	P	PA negative voltage input

9.3.2.2 Typical Application Clock Requirements

Table9- 5. Audio Codec Typical Application Clock Requirements

Operation Mode	32K	PCLK	24MHz	PLL_AUDIO
Playback	on	on	on	on
Capture	on	on	on	on
Headphone detect	on	on	on	off

9.3.2.3 Typical Application Power Requirements

Table9- 6. Audio Codec Typical Application Power Requirements

Operating Mode	VDD_SYS	AVCC	CPVIN
Playback(Headphone)	on	on	on
Capture	on	on	off
Voice call(Headphone)	on	on	on
Headphone detect	on	on	off

9.3.2.4 Power Domain

Audio Codec System needs four powers, AVCC、VDD_SYS、CPVIN(export to CPVDD) and VDD33(export to MBIAS and HBIAS), show in Figure 9-14. The AVCC is provided to ADC/DAC analog part, headphone and mic detect analog part. VDD_SYS is provided to ADC/DAC digital part, register control, headphone and mic detect digital part. CPVIN is provided to headphone amplifier. VDD33 is provided to MBIAS and HBIAS.(VDD33 is bonding to VCC_IO).

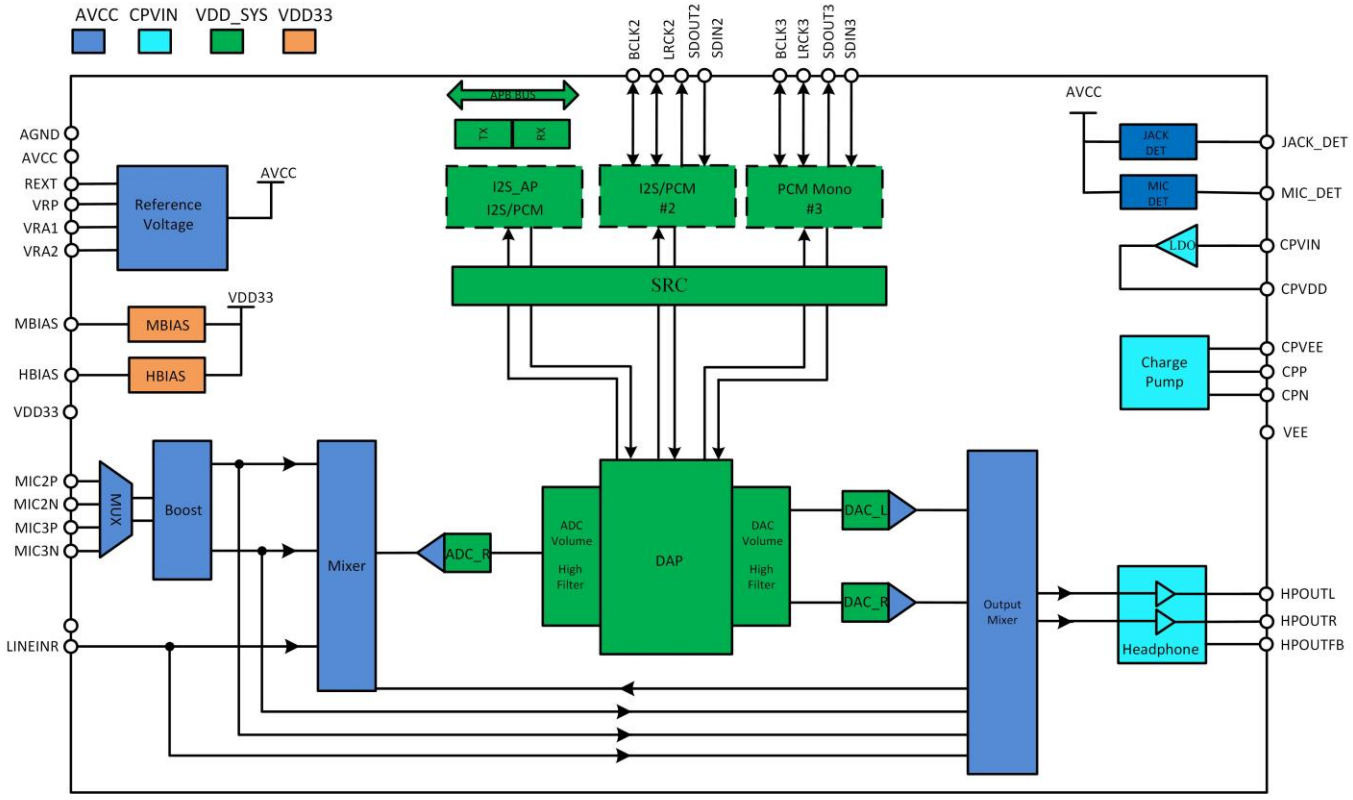


Figure9- 14. Audio Codec Power Domain

9.3.2.5 Clock System

The clock source of audio codec can be selected from CCU Audio PLL. The clock PLL_1x_24M from Audio PLL is always provided 24.576 MHz or 22.5792 MHz(fs=48 kHz or 44.1 kHz) for the I2S_AP MCLK and Mux. System clock of audio can be selected from AIF1CLK or AIF2CLK. AIF1CLK is the reference of the first AIF1 clocking zone. AIF2CLK is the reference

of the second AIF2 clocking zone. And the system clocking must be synchronized with either of the AIFnCLK. The driver should arrange the divider to generate 512*fs(fs=48 kHz or 44.1 kHz) as the SYSCLK.

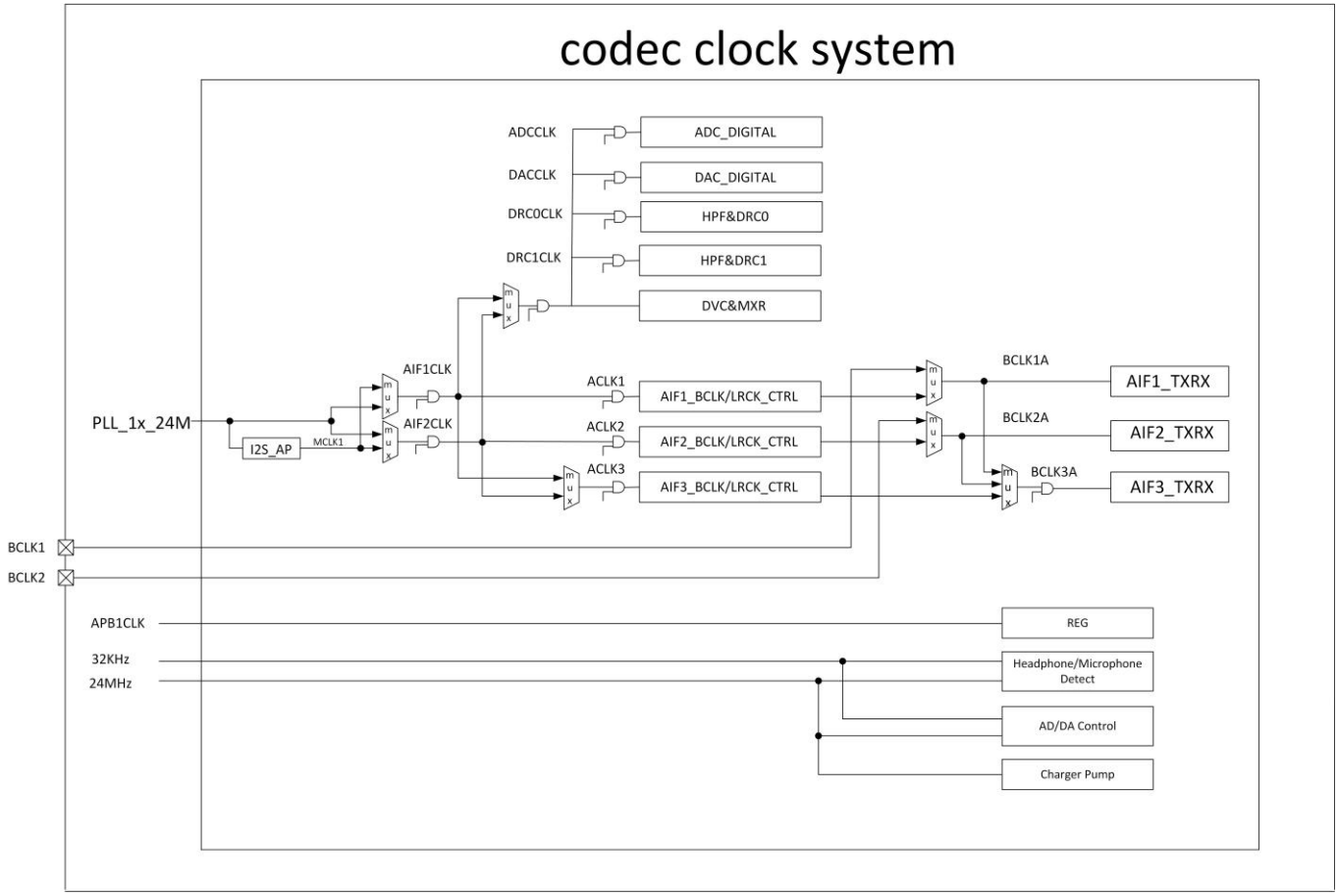


Figure9- 15. System Clock Tree

According to the serial audio data sampling frequency, the device can work in speed ranges of the sampling frequency 8 kHz~192 kHz .The device can work either in master clock mode or slave clock mode.In master mode, BCLK and LRCK are derived internally from AIFnCLK. In slave mode, BCLK and LRCK are supplied externally. BCLK and LRCK must be synchronously derived from the system clock with specific rates.

9.3.2.6 Reset System

9.3.2.6.1 Digital Part Reset System

The SYS_RST will be provided by the VDD_SYS domain, it is provided by the RTC domain, each domain has the de-bounce to confirm the reset system is strong. The codec register part, DVC(Digital Voice Control)&MIX and I2S_AP part will be reset by the SYS_RST during the power on or the system soft writing the reset control logic. The other parts will be reset by the soft configure through writing register.

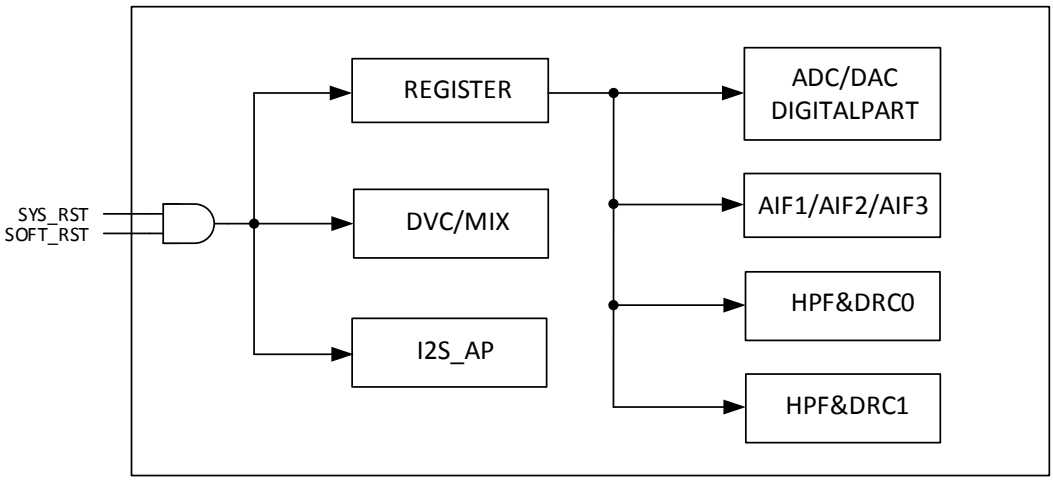


Figure9- 16. Audio Codec Digital Part Reset System

9.3.2.6.2 Analog Part Reset System

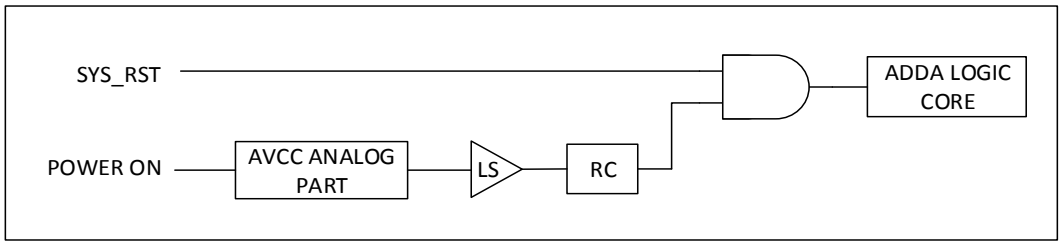


Figure9- 17. Audio Codec Analog Part Reset System

9.3.2.7 Data Path Diagram

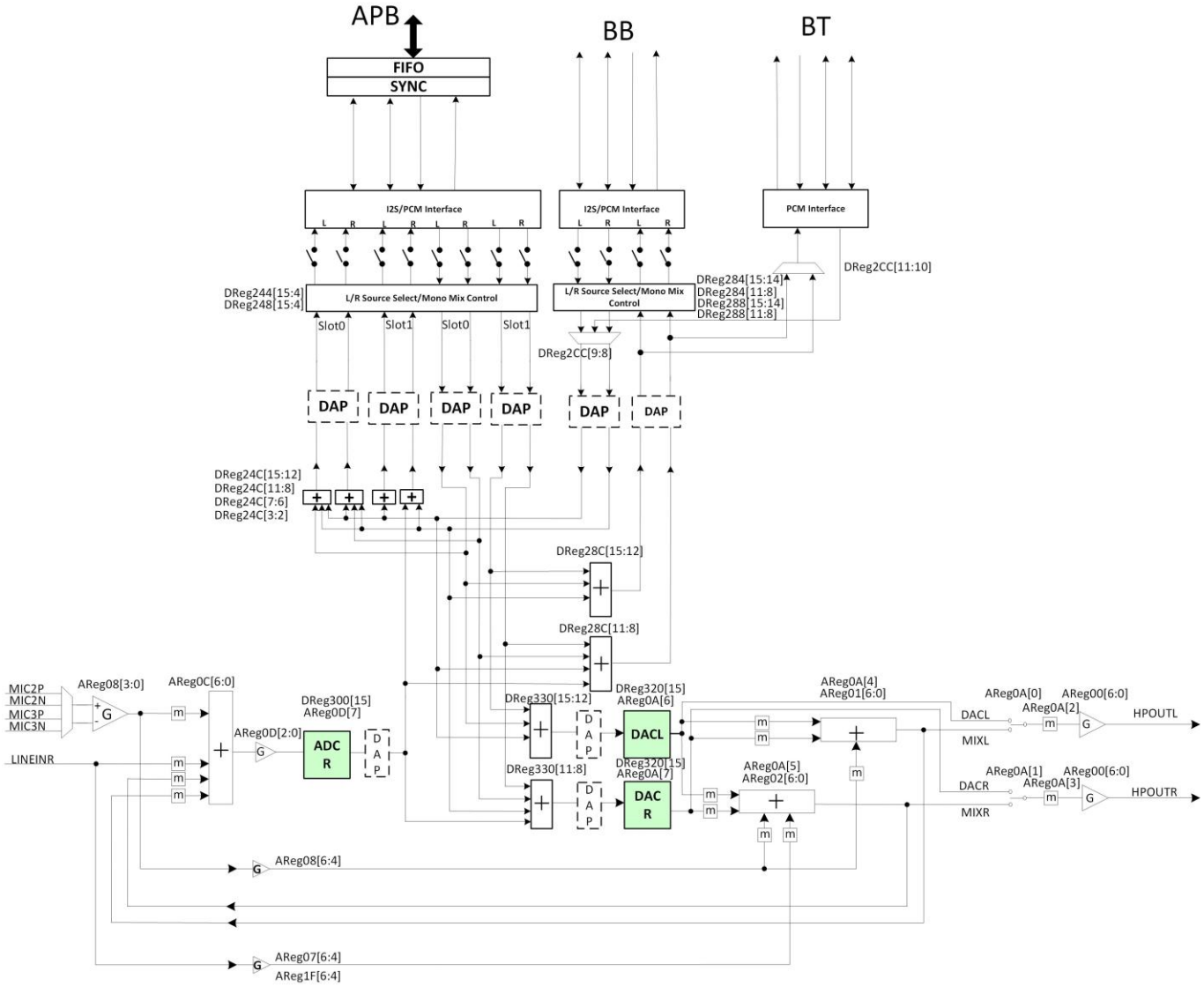


Figure9- 18. Audio Codec Data Path Diagram

9.3.2.8 Typical Application

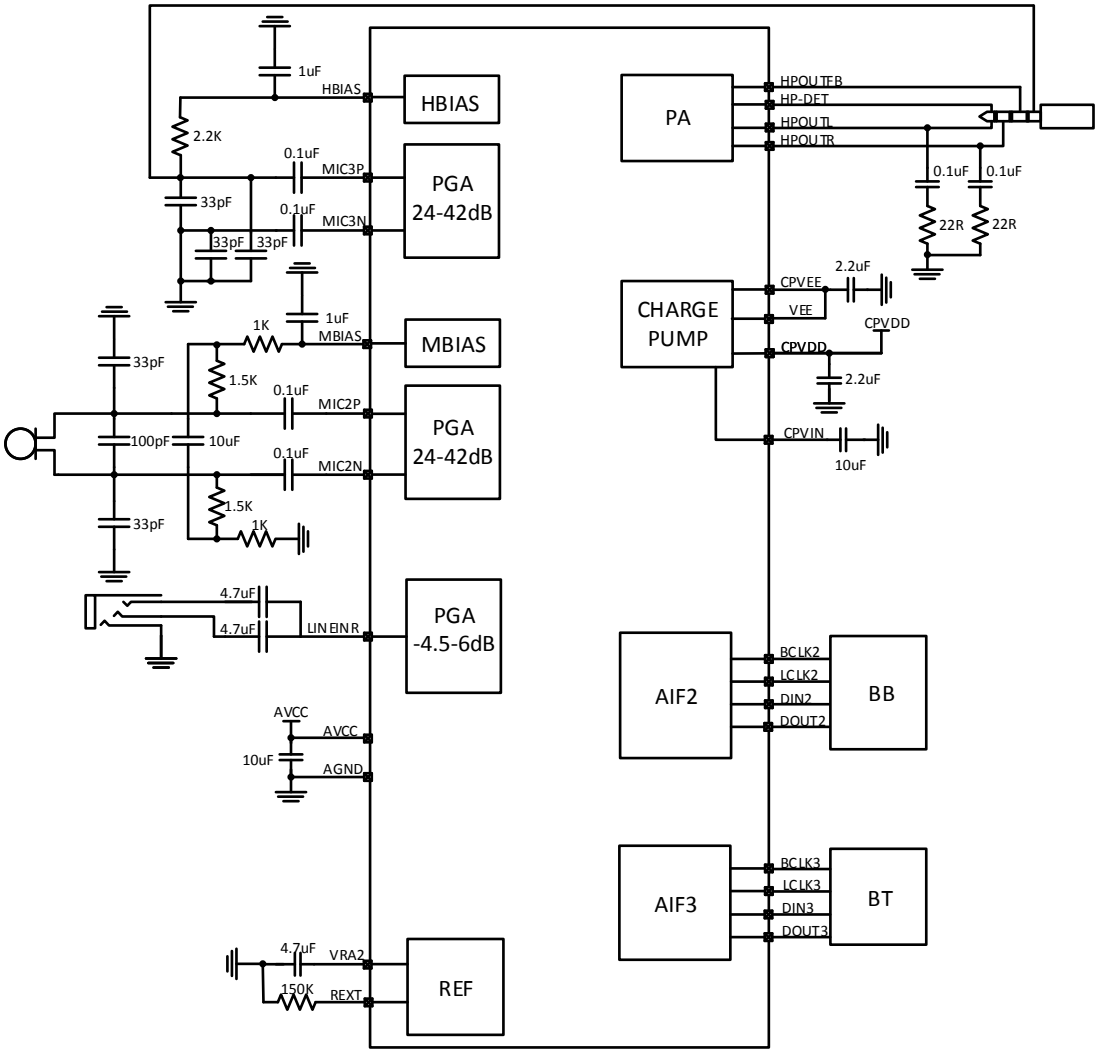


Figure9- 19. Audio Codec Typical Application Diagram

9.3.3 Register List

Module Name	Base Address
Audio Codec	0x05096000

Register Name	Offset	Description
DA_CTL	0x0000	Digital Audio Control Register
DA_FAT0	0x0004	Digital Audio Format Register 0
DA_FAT1	0x0008	Digital Audio Format Register 1
DA_ISTA	0x000C	Digital Audio Interrupt Status Register
DA_RXFIFO	0x0010	Digital Audio RX FIFO Register
DA_FCTL	0x0014	Digital Audio FIFO Control Register
DA_FSTA	0x0018	Digital Audio FIFO Status Register
DA_INT	0x001C	Digital Audio Interrupt Control Register
DA_TXFIFO	0x0020	Digital Audio TX FIFO Register

DA_CLKD	0x0024	Digital Audio Clock Divide Register
DA_TXCNT	0x0028	Digital Audio RX Sample Counter Register
DA_RXCNT	0x002C	Digital Audio TX Sample Counter Register
DA_TXCHSEL	0x0030	Digital Audio TX Channel Select register
DA_TXCHMAP	0x0034	Digital Audio TX Channel Mapping Register
DA_RXCHSEL	0x0038	Digital Audio RX Channel Select register
DA_RXCHMAP	0x003C	Digital Audio RX Channel Mapping Register
SYSCLK_CTL	0x020C	System Clock Control Register
MOD_CLK_ENA	0x0210	Module Clock Control Register
MOD_RST_CTL	0x0214	Module Reset Control Register
SYS_SR_CTRL	0x0218	System Sample Rate Configuration Register
SYS_DVC_MOD	0x0220	System DVC Mode Select Register
AIF1CLK_CTRL	0x0240	AIF1 BCLK/LRCK Control Register
AIF1_ADCDAT_CTRL	0x0244	AIF1 ADCDAT Control Register
AIF1_DACDAT_CTRL	0x0248	AIF1 DACDAT Control Register
AIF1_MIXR_SRC	0x024C	AIF1 Digital Mixer Source Select Register
AIF1_VOL_CTRL1	0x0250	AIF1 Volume Control 1 Register
AIF1_VOL_CTRL2	0x0254	AIF1 Volume Control 2 Register
AIF1_VOL_CTRL3	0x0258	AIF1 Volume Control 3 Register
AIF1_VOL_CTRL4	0x025C	AIF1 Volume Control 4 Register
AIF1_MXR_GAIN	0x0260	AIF1 Digital Mixer Gain Control Register
AIF1_RXD_CTRL	0x0264	AIF1 Receiver Data Discarding Control Register
AIF2_CLK_CTRL	0x0280	AIF2 BCLK/LRCK Control Register
AIF2_ADCDAT_CTRL	0x0284	AIF2 ADCDAT Control Register
AIF2_DACDAT_CTRL	0x0288	AIF2 DACDAT Control Register
AIF2_MXR_SRC	0x028C	AIF2 Digital Mixer Source Select Register
AIF2_VOL_CTRL1	0x0290	AIF2 Volume Control 1 Register
AIF2_VOL_CTRL2	0x0298	AIF2 Volume Control 2 Register
AIF2_MXR_GAIN	0x02A0	AIF2 Digital Mixer Gain Control Register
AIF2_RXD_CTRL	0x02A4	AIF2 Receiver Data Discarding Control Register
AIF3_CLK_CTRL	0x02C0	AIF3 BCLK/LRCK Control Register
AIF3_ADCDAT_CTRL	0x02C4	AIF3 ADCDAT Control Register
AIF3_DACDAT_CTRL	0x02C8	AIF3 DACDAT Control Register
AIF3_SGP_CTRL	0x02CC	AIF3 Signal Path Control Register
AIF3_RXD_CTRL	0x02E4	AIF3 Receiver Data Discarding Control Register
ADC_DIG_CTRL	0x0300	ADC Digital Control Register
ADC_VOL_CTRL	0x0304	ADC Volume Control Register
ADC_DBG_CTRL	0x0308	ADC Debug Control Register
HMIC_CTRL1	0x0310	HMIC Control 1 Register
HMIC_CTRL2	0x0314	HMIC Control 2 Register
HMIC_STS	0x0318	HMIC Status Register
DAC_DIG_CTRL	0x0320	DAC Digital Control Register
DAC_VOL_CTRL	0x0324	DAC Volume Control Register
DAC_DBG_CTRL	0x0328	DAC Debug Control Register
DAC_MXR_SRC	0x0330	DAC Digital Mixer Source Select Register

DAC_MXR_GAIN	0x0334	DAC Digital Mixer Gain Control Register
AC_DAC_DAPCTRL	0x0480	DAC DAP Control Register
AGC_ENA	0x04D0	AGC Enable Register
DRC_ENA	0x04D4	DRC Enable Register
AC_DRC0_HHPFC	0x0600	DRC0 High HPF Coef Register
AC_DRC0_LHPFC	0x0604	DRC0 Low HPF Coef Register
AC_DRC0_CTRL	0x0608	DRC0 Control Register
AC_DRC0_LPFHAT	0x060C	DRC0 Left Peak Filter High Attack Time Coef Register
AC_DRC0_LPFLAT	0x0610	DRC0 Left Peak Filter Low Attack Time Coef Register
AC_DRC0_RPFHAT	0x0614	DRC0 Right Peak Filter High Attack Time Coef Register
AC_DRC0_RPFLAT	0x0618	DRC0 Peak Filter Low Attack Time Coef Register
AC_DRC0_LPFHRT	0x061C	DRC0 Left Peak Filter High Release Time Coef Register
AC_DRC0_LPFLRT	0x0620	DRC0 Left Peak Filter Low Release Time Coef Register
AC_DRC0_RPFHRT	0x0624	DRC0 Right Peak Filter High Release Time Coef Register
AC_DRC0_RPFLRT	0x0628	DRC0 Right Peak Filter Low Release Time Coef Register
AC_DRC0_LRMSHAT	0x062C	DRC0 Left RMS Filter High Coef Register
AC_DRC0_LRMSLAT	0x0630	DRC0 Left RMS Filter Low Coef Register
AC_DRC0_RRMSHAT	0x0634	DRC0 Right RMS Filter High Coef Register
AC_DRC0_RRMSLAT	0x0638	DRC0 Right RMS Filter Low Coef Register
AC_DRC0_HCT	0x063C	DRC0 Compressor Threshold High Setting Register
AC_DRC0_LCT	0x0640	DRC0 Compressor Threshold Low Setting Register
AC_DRC0_HKC	0x0644	DRC0 Compressor Slope High Setting Register
AC_DRC0_LKC	0x0648	DRC0 Compressor Slope Low Setting Register
AC_DRC0_HOPC	0x064C	DRC0 Compressor High Output at Compressor Threshold Register
AC_DRC0_LOPC	0x0650	DRC0 Compressor Low Output at Compressor Threshold Register
AC_DRC0_HLT	0x0654	DRC0 Limiter Threshold High Setting Register
AC_DRC0_LLT	0x0658	DRC0 Limiter Threshold Low Setting Register
AC_DRC0_HKl	0x065C	DRC0 Limiter Slope High Setting Register
AC_DRC0_LKl	0x0660	DRC0 Limiter Slope Low Setting Register
AC_DRC0_HOPL	0x0664	DRC0 Limiter High Output at Limiter Threshold
AC_DRC0_LOPL	0x0668	DRC0 Limiter Low Output at Limiter Threshold
AC_DRC0_HET	0x066C	DRC0 Expander Threshold High Setting Register
AC_DRC0_LET	0x0670	DRC0 Expander Threshold Low Setting Register
AC_DRC0_HKE	0x0674	DRC0 Expander Slope High Setting Register
AC_DRC0_LKE	0x0678	DRC0 Expander Slope Low Setting Register
AC_DRC0_HOPE	0x067C	DRC0 Expander High Output at Expander Threshold
AC_DRC0_LOPE	0x0680	DRC0 Expander Low Output at Expander Threshold
AC_DRC0_HKN	0x0684	DRC0 Linear Slope High Setting Register
AC_DRC0_LKN	0x0688	DRC0 Linear Slope Low Setting Register
AC_DRC0_SFHAT	0x068C	DRC0 Smooth Filter Gain High Attack Time Coef Register
AC_DRC0_SFLAT	0x0690	DRC0 Smooth Filter Gain Low Attack Time Coef Register
AC_DRC0_SFHRT	0x0694	DRC0 Smooth Filter Gain High Release Time Coef Register
AC_DRC0_SFLRT	0x0698	DRC0 Smooth Filter Gain Low Release Time Coef Register

AC_DRC0_MXGHS	0x069C	DRC0 MAX Gain High Setting Register
AC_DRC0_MXGLS	0x06A0	DRC0 MAX Gain Low Setting Register
AC_DRC0_MNGHS	0x06A4	DRC0 MIN Gain High Setting Register
AC_DRC0_MNGLS	0x06A8	DRC0 MIN Gain Low Setting Register
AC_DRC0_EPSHC	0x06AC	DRC0 Expander Smooth Time High Coef Register
AC_DRC0_EPSLC	0x06B0	DRC0 Expander Smooth Time Low Coef Register
AC_DRC0_HPFHGAIN	0x06B8	DRC0 HPF Gain High Coef Register
AC_DRC0_HPFHGAIN	0x06BC	DRC0 HPF Gain Low Coef Register
AC_DRC1_HHPFC	0x0700	DRC1 High HPF Coef Register
AC_DRC1_LHPFC	0x0704	DRC1 Low HPF Coef Register
AC_DRC1_CTRL	0x0708	DRC1 Control Register
AC_DRC1_LPFHAT	0x070C	DRC1 Left Peak Filter High Attack Time Coef Register
AC_DRC1_LPFLAT	0x0710	DRC1 Left Peak Filter Low Attack Time Coef Register
AC_DRC1_RPFHAT	0x0714	DRC1 Right Peak Filter High Attack Time Coef Register
AC_DRC1_RPFLAT	0x0718	DRC1 Peak Filter Low Attack Time Coef Register
AC_DRC1_LPFHRT	0x071C	DRC1 Left Peak Filter High Release Time Coef Register
AC_DRC1_LPFLRT	0x0720	DRC1 Left Peak Filter Low Release Time Coef Register
AC_DRC1_RPFHRT	0x0724	DRC1 Right Peak filter High Release Time Coef Register
AC_DRC1_RPFLRT	0x0728	DRC1 Right Peak filter Low Release Time Coef Register
AC_DRC1_LRMSHAT	0x072C	DRC1 Left RMS Filter High Coef Register
AC_DRC1_LRMSLAT	0x0730	DRC1 Left RMS Filter Low Coef Register
AC_DRC1_RRMSHAT	0x0734	DRC1 Right RMS Filter High Coef Register
AC_DRC1_RRMSLAT	0x0738	DRC1 Right RMS Filter Low Coef Register
AC_DRC1_HCT	0x073C	DRC1 Compressor Threshold High Setting Register
AC_DRC1_LCT	0x0740	DRC1 Compressor Threshold Low Setting Register
AC_DRC1_HKC	0x0744	DRC1 Compressor Slope High Setting Register
AC_DRC1_LKC	0x0748	DRC1 Compressor Slope Low Setting Register
AC_DRC1_HOPC	0x074C	DRC1 Compressor High Output at Compressor Threshold Register
AC_DRC1_LOPC	0x0750	DRC1 Compressor Low Output at Compressor Threshold Register
AC_DRC1_HLT	0x0754	DRC1 Limiter Threshold High Setting Register
AC_DRC1_LLT	0x0758	DRC1 Limiter Threshold Low Setting Register
AC_DRC1_HKl	0x075C	DRC1 Limiter Slope High Setting Register
AC_DRC1_LKl	0x0760	DRC1 Limiter Slope Low Setting Register
AC_DRC1_HOPL	0x0764	DRC1 Limiter High Output at Limiter Threshold
AC_DRC1_LOPL	0x0768	DRC1 Limiter Low Output at Limiter Threshold
AC_DRC1_HET	0x076C	DRC1 Expander Threshold High Setting Register
AC_DRC1_LET	0x0770	DRC1 Expander Threshold Low Setting Register
AC_DRC1_HKE	0x0774	DRC1 Expander Slope High Setting Register
AC_DRC1_LKE	0x0778	DRC1 Expander Slope Low Setting Register
AC_DRC1_HOPE	0x077C	DRC1 Expander High Output at Expander Threshold
AC_DRC1_LOPE	0x0780	DRC1 Expander Low Output at Expander Threshold
AC_DRC1_HKN	0x0784	DRC1 Linear Slope High Setting Register
AC_DRC1_LKN	0x0788	DRC1 Linear Slope Low Setting Register

AC_DRC1_SFHAT	0x078C	DRC1 Smooth Filter Gain High Attack Time Coef Register
AC_DRC1_SFLAT	0x0790	DRC1 Smooth Filter Gain Low Attack Time Coef Register
AC_DRC1_SFHRT	0x0794	DRC1 Smooth Filter Gain High Release Time Coef Register
AC_DRC1_SFLRT	0x0798	DRC1 Smooth Filter Gain Low Release Time Coef Register
AC_DRC1_MXGHS	0x079C	DRC1 MAX Gain High Setting Register
AC_DRC1_MXGLS	0x07A0	DRC1 MAX Gain Low Setting Register
AC_DRC1_MNGHS	0x07A4	DRC1 MIN Gain High Setting Register
AC_DRC1_MNGLS	0x07A8	DRC1 MIN Gain Low Setting Register
AC_DRC1_EPSHC	0x07AC	DRC1 Expander Smooth Time High Coef Register
AC_DRC1_EPSLC	0x07B0	DRC1 Expander Smooth Time Low Coef Register
AC_DRC1_HPFHGAIN	0x07B8	DRC1 HPF Gain High Coef Register
AC_DRC1_HPFHGAIN	0x07BC	DRC1 HPF Gain Low Coef Register
Analog Domain Register		
HP_CTRL	0x00	Headphone Amplifier Control Register
LO_MIX_CTRL	0x01	Left Output Mixer Control Register
RO_MIX_CTRL	0x02	Right Output Mixer Control Register
MIC2_CTRL	0x08	MIC2 Control Register
LINEIN_CTRL	0x09	Linein Control Register
MIX_DAC_CTRL	0x0A	Mixer and DAC Control Register
RO_MIX_CTRL	0x0C	Right Output Mixer Control Register
ADC_CTRL	0x0D	ADC Control Register
MBIAS_CTRL	0x0E	Microphone Bias Control Register
APT_REG	0x0F	Analog Performance Tuning Register
OP_BIAS_CTRL0	0x10	OP BIAS Control Register0
OP_BIAS_CTRL1	0x11	OP BIAS Control Register1
ZC_VOL_CTRL	0x12	ZERO Cross &USB Bias Control Register
BIAS_CAL_DATA	0x13	Bias Calibration Data Register
BIAS_CAL_SET	0x14	Bias Calibration Set Data Register
BD_CAL_CTRL	0x15	Bias & DA16 Calibration Control Register
HP_PA_CTRL	0x16	Headphone PA Control Register
HP_CAL_CTRL	0x17	Headphone Calibration Control
RHP_CAL_DATA	0x18	Right Headphone Calibration DATA
RHP_CAL_SET	0x19	Right Headphone Calibration Setting
LHP_CAL_DATA	0x1A	Left Headphone Calibration DATA
LHP_CAL_SET	0x1B	Left Headphone Calibration Setting
MDET_CTRL	0x1C	MIC Detect Control Register
JACK_MIC_CTRL	0x1D	Jack & Mic Detect Control Register
CP_LDO_CTR	0x21	Charge Pump LDO Output Control Register

9.3.4 Register Description

9.3.4.1 I2S_AP Control Register(Default Value: 0x0000_0000)

Offset: 0x0000	Register Name: DA_CTL

Bit	Read/Write	Default/Hex	Description
31:9	/	/	/
8	R/W	0x0	SDO_EN 0: Disable 1: Enable
7	/	/	/
6	R/W	0x0	ASS Audio sample select when TX FIFO under run 0: Sending zero 1: Sending last audio sample
5	R/W	0x0	MS Master Slave Select 0: Master 1: Slave
4	R/W	0x0	PCM 0: I2S Interface 1: PCM Interface
3	R/W	0x0	LOOP Loop back test 0: Normal mode 1: Loop back test When setting to '1', connecting the SDO with the SDI in Master mode.
2	R/W	0x0	TXEN Transmitter Block Enable 0: Disable 1: Enable
1	R/W	0x0	RXEN Receiver Block Enable 0: Disable 1: Enable
0	R/W	0x0	GEN Globe Enable A disable on this bit overrides any other block or channel enables and flushes all FIFOs. 0: Disable 1: Enable

9.3.4.2 I2S_AP Format Register 0(Default Value: 0x0000_000C)

Offset: 0x0004			Register Name: DA_FAT0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R/W	0x0	LRCP Left/Right Clock Parity 0: Normal

			1: Inverted In DSP/PCM mode 0: MSB is available on 2nd BCLK rising edge after LRC rising edge 1: MSB is available on 1st BCLK rising edge after LRC rising edge
6	R/W	0x0	BCP BCLK Parity 0: Normal 1: Inverted
5:4	R/W	0x0	SR Sample Resolution 00: 16-bit 01: 20-bit 10: 24-bit 11: Reserved
3:2	R/W	0x3	WSS Word Select Size 00: 16 BCLK 01: 20 BCLK 10: 24 BCLK 11: 32 BCLK
1:0	R/W	0x0	FMT Serial Data Format 00: Standard I2S Format 01: Left Justified Format 10: Right Justified Format 11: Reserved

9.3.4.3 I2S_AP Format Register 1(Default Value: 0x0000_4020)

Offset: 0x0008			Register Name: DA_FAT1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:15	/	/	/
14:12	R/W	0x4	PCM_SYNC_PERIOD PCM SYNC Period Clock Number 000: 16 BCLK period 001: 32 BCLK period 010: 64 BCLK period 011: 128 BCLK period 100: 256 BCLK period Others : Reserved
11	R/W	0x0	PCM_SYNC_OUT PCM Sync Out 0: Enable PCM_SYNC output in Master mode 1: Suppress PCM_SYNC whilst keeping PCM_CLK running. Some Codec utilize this to enter a low power state.

10	R/W	0x0	PCM Out Mute Writing 1 forces PCM_OUT to 0
9	R/W	0x0	MLS MSB / LSB First Select 0: MSB First 1: LSB First
8	R/W	0x0	SEXT Sign Extend (only for 16 bit slot) 0: Zeros or audio gain padding at LSB position 1: Sign extension at MSB position When writing the bit is 0, the unused bits are audio gain for 13-bit linear sample and zeros padding for 8-bit companding sample. When writing the bit is 1, the unused bits are both sign extension.
7:6	R/W	0x0	SI Slot Index 00: the 1st slot 01: the 2nd slot 10: the 3rd slot 11: the 4th slot
5	R/W	0x1	SW Slot Width 0: 8 clocks width 1: 16 clocks width **NOTE** **For A-law or u-law PCM sample, if this bit is set to 1, eight zero bits are following with PCM sample.**
4	R/W	0x0	SSYNC Short Sync Select 0: Long Frame Sync 1: Short Frame Sync It should be set '1' for 8 clocks width slot.
3:2	R/W	0x0	RX_PDM PCM Data Mode 00: 16-bit Linear PCM 01: 8-bit Linear PCM 10: 8-bit u-law 11: 8-bit A-law
1:0	R/W	0x0	TX_PDM PCM Data Mode 00: 16-bit Linear PCM 01: 8-bit Linear PCM 10: 8-bit u-law 11: 8-bit A-law

9.3.4.4 I2S_AP Interrupt Status Register(Default Value: 0x0000_0010)

Offset: 0x000C			Register Name: DA_ISTA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:7	/	/	/
6	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
5	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
4	R/W1C	0x1	TXE_INT TX FIFO Empty Pending Interrupt 0: No Pending IRQ 1: FIFO Empty Pending Interrupt Write '1' to clear this interrupt or automatic clear if interrupt condition fails.
3	/	/	/
2	R/W1C	0x0	RXU_INT RX FIFO Underrun Pending Interrupt 0: No Pending Interrupt 1: FIFO Under run Pending Interrupt Write 1 to clear this interrupt
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/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 interrupt condition fails.

9.3.4.5 I2S_AP RX FIFO Register(Default Value: 0x0000_0000)

Offset: 0x0010			Register Name: DA_RXFIFO
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	RX_DATA RX Sample

			Host can get one sample by reading this register. The left channel sample data is first and then the right channel sample.

9.3.4.6 I2S_AP FIFO Control Register(Default Value: 0x0004_00F0)

Offset: 0x0014			Register Name: DA_FCTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31	R/W	0x0	HUB_EN Audio Hub Enable 0: Disable 1: Enable
30:26	/	/	/
25	R/WAC	0x0	FTX Write '1' to flush TX FIFO, automatic clear to '0'.
24	R/WAC	0x0	FRX Write '1' to flush RX FIFO, automatic clear to '0'.
23:19	/	/	/
18:12	R/W	0x40	TXTL TX FIFO Empty Trigger Level Interrupt and DMA request trigger level for TXFIFO normal condition. When the number of data is less than Trigger level, the interrupt or DMA pending will be set. Trigger Level = TXTL
11:10	/	/	/
9:4	R/W	0xF	RXTL RX FIFO Trigger Level Interrupt and DMA request trigger level for RXFIFO normal condition. When the number of data is higher than Trigger level, the interrupt or DMA pending will be set. Trigger Level = RXTL + 1
3	/	/	/
2	R/W	0x0	TXIM TX FIFO Input Mode (Mode 0, 1) 0: Valid data at the MSB of TXFIFO register 1: Valid data at the LSB of TXFIFO register Example for 20-bits transmitted audio sample: Mode 0: FIFO_I[23:0] = { APB_WDATA [31:12], 4'h0} Mode 1: FIFO_I[23:0] = { APB_WDATA [19:0], 4'h0}
1:0	R/W	0x0	RXOM RX FIFO Output Mode (Mode 0, 1, 2, 3) 00: Expanding '0' at LSB of DA_RXFIFO register. 01: Expanding received sample sign bit at MSB of DA_RXFIFO register. 10: Truncating received samples at high half-word of DA_RXFIFO register and low half-word of DA_RXFIFO register is filled by '0'. 11: Truncating received samples at low half-word of DA_RXFIFO register

			and high half-word of DA_RXFIFO register is expanded by its sign bit. Example for 20-bits received audio sample: Mode 0: APB_RDATA[31:0] = {RXFIFO[19:0], 12'h0} Mode 1: APB_RDATA[31:0] = {12{RXFIFO[19]}, RXFIFO[19:0]} Mode 2: APB_RDATA [31:0] = {RXFIFO[19:4], 16'h0} Mode 3: APB_RDATA[31:0] = {16{RXFIFO[19], RXFIFO[19:4]}

9.3.4.7 I2S_AP FIFO Status Register(Default Value: 0x1080_0000)

Offset: 0x0018			Register Name: DA_FSTA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	/	/	/
28	R	0x1	TXE 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)
27:24	/	/	/
23:16	R	0x80	TXE_CNT TX FIFO Empty Space Word Counter
15:9	/	/	/
8	R	0x0	RXA RX FIFO Available 0: No available data in RX FIFO 1: More than one sample in RX FIFO (>= 1 word)
7	/	/	/
6:0	R	0x0	RXA_CNT RX FIFO Available Sample Word Counter

9.3.4.8 I2S_AP DMA & Interrupt Control Register(Default Value: 0x0000_0000)

Offset: 0x001C			Register Name: DA_INT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R/W	0x0	TX_DRQ TX FIFO Empty DRQ Enable 0: Disable 1: Enable
6	R/W	0x0	TXUI_EN TX FIFO Underrun Interrupt Enable 0: Disable 1: Enable
5	R/W	0x0	TXOI_EN TX FIFO Overrun Interrupt Enable 0: Disable

			1: Enable When setting to '1', an interrupt happens when writing new audio data if TX FIFO is full.
4	R/W	0x0	TXEI_EN TX FIFO Empty Interrupt Enable 0: Disable 1: Enable
3	R/W	0x0	RX_DRQ RX FIFO Data Available DRQ Enable 0: Disable 1: Enable When setting to '1', RXFIFO DMA Request line is asserted if Data is available in RX FIFO.
2	R/W	0x0	RXUI_EN RX FIFO Underrun Interrupt Enable 0: Disable 1: Enable
1	R/W	0x0	RXOI_EN RX FIFO Overrun Interrupt Enable 0: Disable 1: Enable
0	R/W	0x0	RXAI_EN RX FIFO Data Available Interrupt Enable 0: Disable 1: Enable

9.3.4.9 I2S_AP TX FIFO Register(Default Value: 0x0000_0000)

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

9.3.4.10 I2S_AP Clock Divide Register(Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: DA_CLKD
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	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.**
6:4	R/W	0x0	BCLKDIV BCLK Divide Ratio from MCLK 000: Divide by 2 (BCLK = MCLK/2) 001: Divide by 4 010: Divide by 6 011: Divide by 8 100: Divide by 12 101: Divide by 16 110: Divide by 32 111: Divide by 64
3:0	R/W	0x0	MCLKDIV MCLK Divide Ratio from Audio PLL Output 0000: Divide by 1 0001: Divide by 2 0010: Divide by 4 0011: Divide by 6 0100: Divide by 8 0101: Divide by 12 0110: Divide by 16 0111: Divide by 24 1000: Divide by 32 1001: Divide by 48 1010: Divide by 64 Others : Reserved

9.3.4.11 I2S_AP TX Counter Register(Default Value: 0x0000_0000)

Offset: 0x0028			**Register Name: DA_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 valve at any time. After been updated by the initial value, the counter register should count on base of this initial value.

9.3.4.12 I2S_AP RX Counter Register(Default Value: 0x0000_0000)

Offset: 0x002C	**Register Name: DA_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 I2S_AP 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 been updated by the initial value, the counter register should count on base of this initial value.

9.3.4.13 I2S_AP TX Channel Select Register(Default Value: 0x0000_0001)

Offset: 0x0030			Register Name: DA_TXCHSEL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2:0	R/W	0x1	TX_CHSEL TX Channel Select 000: 1-ch 001: 2-ch 010: 3-ch 011: 4-ch

9.3.4.14 I2S_AP TX Channel Mapping Register(Default Value: 0x7654_3210)

Offset: 0x0034			Register Name: DA_TXCHMAP
Bit	**Read/Write**	**Default/Hex**	**Description**
31:15	/	/	/
14:12	R/W	0x3	TX_CH3_MAP TX Channel3 Mapping 000: 1st sample 001: 2nd sample 010: 3rd sample 011: 4th sample 1xx: Reserved
11	/	/	/
10:8	R/W	0x2	TX_CH2_MAP TX Channel2 Mapping 000: 1st sample 001: 2nd sample 010: 3rd sample 011: 4th sample 1xx: Reserved
7	/	/	/
6:4	R/W	0x1	TX_CH1_MAP TX Channel1 Mapping

			000: 1st sample 001: 2nd sample 010: 3rd sample 011: 4th sample 1xx: Reserved
3	/	/	/
2:0	R/W	0x0	TX_CH0_MAP TX Channel0 Mapping 000: 1st sample 001: 2nd sample 010: 3rd sample 011: 4th sample 1xx: Reserved

9.3.4.15 I2S_AP RX Channel Select Register(Default Value: 0x0000_0001)

Offset: 0x0038			Register Name: DA_RXCHSEL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:3	/	/	/
2:0	R/W	0x1	RX_CHSEL RX Channel Select 000: 1-ch 001: 2-ch 010: 3-ch 011: 4-ch Others: Reserved

9.3.4.16 I2S_AP RX Channel Mapping Register(Default Value: 0x0000_3210)

Offset: 0x003C			Register Name: DA_RXCHMAP
Bit	**Read/Write**	**Default/Hex**	**Description**
31:15	/	/	/
14:12	R/W	0x3	RX_CH3_MAP RX Channel3 Mapping 000: 1st sample 001: 2nd sample 010: 3rd sample 011: 4th sample Others: Reserved
11	/	/	/
10:8	R/W	0x2	RX_CH2_MAP RX Channel2 Mapping 000: 1st sample 001: 2nd sample

			010: 3rd sample 011: 4th sample Others: Reserved
7	/	/	/
6:4	R/W	0x1	RX_CH1_MAP RX Channel1 Mapping 000: 1st sample 001: 2nd sample 010: 3rd sample 011: 4th sample Others: Reserved
3	/	/	/
2:0	R/W	0x0	RX_CH0_MAP RX Channel0 Mapping 000: 1st sample 001: 2nd sample 010: 3rd sample 011: 4th sample Others: Reserved

9.3.4.17 System Clock Control Register(Default Value: 0x0000_0000)

Offset: 0x020C			Register Name: SYSCLK_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:12	/	/	/
11	R/W	0x0	AIF1CLK_ENA AIF1CLK Enable 0: Disable 1: Enable
10	R/W	0x0	Reserved
9:8	R/W	0x0	AIF1CLK_SRC AIF1CLK Source Select 00: MCLK1 01: Reserved 1X: pll2_1x
7	R/W	0x0	AIF2CLK_ENA AIF2CLK Enable 0: Disable 1: Enable
6	R/W	0x0	Reserved
5:4	R/W	0x0	AIF2CLK_SRC AIF2CLK Source Select 00: MCLK1 01: Reserved 1X: pll2_1x

3	R/W	0x0	SYSCLK_ENA SYSCLK Enable 0: Disable 1: Enable
2:1	R/W	0x0	Reserved
0	R/W	0x0	SYSCLK_SRC System Clock Source Select 0: AIF1CLK 1: AIF2CLK

9.3.4.18 Module Clock Control Register(Default Value: 0x0000_0000)

Offset: 0x0210			Register Name: MOD_CLK_ENA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x0	Module clock enable control 0: Clock disable 1: Clock enable BIT15-AIF1 BIT14-AIF2 BIT13-AIF3 BIT12-Reserved BIT11-Reserved BIT10-Reserved BIT9-Reserved BIT8-Reserved BIT7- Reserved BIT6-HPF & DRC0 BIT5-HPF & DRC1 BIT4-Reserved BIT3-ADC Digital BIT2-DAC Digital BIT1-Reserved BIT0-Reserved

9.3.4.19 Module Reset Control Register(Default Value: 0x0000_0000)

Offset: 0x0214			Register Name: MOD_RST_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x0	Module reset control 0: Reset asserted 1: Reset de-asserted BIT15-AIF1

			BIT14-AIF2 BIT13-AIF3 BIT12-Reserved BIT11-Reserved BIT10-Reserved BIT9-Reserved BIT8-Reserved BIT7- Reserved BIT6-HPF & DRC0 BIT5-HPF & DRC1 BIT4-Reserved BIT3-ADC Digital BIT2-DAC Digital BIT1-Reserved BIT0-Reserved

9.3.4.20 System Sample Rate Configuration Register(Default Value: 0x0000_0000)

Offset: 0x0218			**Register Name: SYS_SR_CTRL**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:12	R/W	0x0	ADDA_FS_AIF1 ADDA Sample Rate synchronized with AIF1 clock zone 0000: 8 kHz 0001: 11.025 kHz 0010: 12 kHz 0011: 16 kHz 0100: 22.05 kHz 0101: 24 kHz 0110: 32 kHz 0111: 44.1 kHz 1000: 48 kHz 1001: 96 kHz 1010: 192 kHz Other: Reserved
11:8	R/W	0x0	ADDA_FS_AIF2 ADDA Sample Rate synchronized with AIF2 clock zone 0000: 8 kHz 0001: 11.025 kHz 0010: 12 kHz 0011: 16 kHz 0100: 22.05 kHz 0101: 24 kHz 0110: 32 kHz 0111: 44.1 kHz

			1000: 48 kHz 1001: 96 kHz 1010: 192 kHz Other: Reserved
7:0	/	/	/

9.3.4.21 System DVC Mode Select Register(Default Value: 0x0000_0000)

Offset: 0x0220			Register Name: SYS_DVC_CLK
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1:0	R/W	0x0	AIFDVC_FS_SEL 0 : DVC output to AIF sync 1 : normal

9.3.4.22 AIF1 BCLK/LRCK Control Register(Default Value: 0x0000_0000)

Offset: 0x0240			Register Name: AIF1CLK_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R/W	0x0	AIF1_MSTR_MOD AIF1 Audio Interface mode select 0: Master mode 1: Slave mode
14	R/W	0x0	AIF1_BCLK_INV AIF1 BCLK Polarity 0: Normal 1: Inverted
13	R/W	0x0	AIF1_LRCK_INV AIF1 LRCK Polarity 0: Normal 1: Inverted
12:9	R/W	0x0	AIF1_BCLK_DIV Select AIF1CLK/BCLK1 Ratio 0000: AIF1CLK/1 0001: AIF1CLK/2 0010: AIF1CLK/4 0011: AIF1CLK/6 0100: AIF1CLK/8 0101: AIF1CLK/12 0110: AIF1CLK/16 0111: AIF1CLK/24 1000: AIF1CLK/32 1001: AIF1CLK/48

			1010: AIF1CLK/64 1011: AIF1CLK/96 1100: AIF1CLK/128 1101: AIF1CLK/192 1110: Reserved 1111: Reserved
8:6	R/W	0x0	AIF1_LRCK_DIV Select BCLK1/LRCK Ratio 000: 16 001: 32 010: 64 011: 128 100: 256 1xx: Reserved
5:4	R/W	0x0	AIF1_WORD_SIZ AIF1 Digital Interface Word Size 00: 8-bit 01: 16-bit 10: 20-bit 11: 24-bit
3:2	R/W	0x0	AIF1_DATA_FMT AIF1 Digital Interface Data Format 00: I2S mode 01: Left mode 10: Right mode 11: DSP mode
1	R/W	0x0	DSP_MONO_PCM DSP Mono Mode Select 0: Stereo mode select 1: Mono mode select
0	R/W	0x0	AIF1_TDMM_ENA AIF1 TDM Mode Enable 0: Disable 1: Enable

9.3.4.23 AIF1 ADCDAT Control Register(Default Value: 0x0000_0000)

Offset: 0x0244			Register Name: AIF1_ADCDAT_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R/W	0x0	AIF1_ADC0L_ENA AIF1 ADC Timeslot 0 left channel enable 0: Disable 1: Enable
14	R/W	0x0	AIF1_ADC0R_ENA

			AIF1 ADC Timeslot 0 right channel enable 0: Disable 1: Enable
13	R/W	0x0	AIF1_ADC1L_ENA AIF1 ADC Timeslot 1 left channel enable 0: Disable 1: Enable
12	R/W	0x0	AIF1_ADC1R_ENA AIF1 ADC Timeslot 1 right channel enable 0: Disable 1: Enable
11:10	R/W	0x0	AIF1_ADC0L_SRC AIF1 ADC Timeslot 0 left channel data source select 00: AIF1 ADC0L 01: AIF1 ADC0R 10: (AIF1 ADC0L+AIF1 ADC0R) 11: (AIF1 ADC0L+AIF1 ADC0R)/2
9:8	R/W	0x0	AIF1_ADC0R_SRC AIF1 ADC Timeslot 0 right channel data source select 00: AIF1 ADC0R 01: AIF1 ADC0L 10: (AIF1 ADC0L+AIF1 ADC0R) 11: (AIF1 ADC0L+AIF1 ADC0R)/2
7:6	R/W	0x0	AIF1_ADC1L_SRC AIF1 ADC Timeslot 1 left channel data source select 00: AIF1 ADC1L 01: AIF1 ADC1R 10: (AIF1 ADC1L+AIF1 ADC1R) 11: (AIF1 ADC1L+AIF1 ADC1R)/2
5:4	R/W	0x0	AIF1_ADC1R_SRC AIF1 ADC Timeslot 1 right channel data source select 00: AIF1 ADC1R 01: AIF1 ADC1L 10: (AIF1 ADC1L+AIF1 ADC1R) 11: (AIF1 ADC1L+AIF1 ADC1R)/2
3	R/W	0x0	AIF1_ADCP_ENA AIF1 ADC Companding enable(8-bit mode only) 0: Disable 1: Enable
2	R/W	0x0	AIF1_ADCUL_ENA AIF1 ADC Companding mode select 0: A-law 1: u-law
1:0	R/W	0x0	AIF1_SLOT_SIZ Select the slot size(only in TDM mode) 00: 8

			01: 16 10: 32 11: Reserved

9.3.4.24 AIF1 DACDAT Control Register(Default Value: 0x0000_0000)

Offset: 0x0248			Register Name: AIF1_DACDAT_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R/W	0x0	AIF1_DAC0L_ENA AIF1 DAC Timeslot 0 left channel enable 0: Disable 1: Enable
14	R/W	0x0	AIF1_DAC0R_ENA AIF1 DAC Timeslot 0 right channel enable 0: Disable 1: Enable
13	R/W	0x0	AIF1_DAC1L_ENA AIF1 DAC Timeslot 1 left channel enable 0: Disable 1: Enable
12	R/W	0x0	AIF1_DAC1R_ENA AIF1 DAC Timeslot 1 right channel enable 0: Disable 1: Enable
11:10	R/W	0x0	AIF1_DAC0L_SRC AIF1 DAC Timeslot 0 left channel data source select 00: AIF1 DAC0L 01: AIF1 DA0R 10: (AIF1 DAC0L+AIF1 DAC0R) 11: (AIF1 DAC0L+AIF1 DAC0R)/2
9:8	R/W	0x0	AIF1_DAC0R_SRC AIF1 DAC Timeslot 0 right channel data source select 00: AIF1 DAC0R 01: AIF1 DAC0L 10: (AIF1 DAC0L+AIF1 DAC0R) 11: (AIF1 DAC0L+AIF1 DAC0R)/2
7:6	R/W	0x0	AIF1_DAC1L_SRC AIF1 DAC Timeslot 1 left channel data source select 00: AIF1 DAC1L 01: AIF1 DAC1R 10: (AIF1 DAC1L+AIF1 DAC1R) 11: (AIF1 DAC1L+AIF1 DAC1R)/2
5:4	R/W	0x0	AIF1_DAC1R_SRC AIF1 DAC Timeslot 1 right channel data source select

			00: AIF1 DAC1R 01: AIF1 DAC1L 10: (AIF1 DAC1L+AIF1 DAC1R) 11: (AIF1 DAC1L+AIF1 DAC1R)/2
3	R/W	0x0	AIF1_DACP_ENA AIF1 DAC Companding enable(8-bit mode only) 00: Disable 01: Enable
2	R/W	0x0	AIF1_DACUL_ENA AIF1 DAC Companding mode select 0: A-law 1: u-law
1	R/W	0x0	Reserved
0	R/W	0x0	AIF1_LOOP_ENA AIF1 loopback enable 0: No loopback 1: Loopback(ADCDAT1 data output to DACDAT1 data input)

9.3.4.25 AIF1 Digital Mixer Source Select Register(Default Value: 0x0000_0000)

Offset: 0x024C			Register Name: AIF1_MXR_SRC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:12	R/W	0x0	AIF1_ADC0L_MXL_SRC AIF1 ADC Timeslot 0 left channel mixer source select 0: Disable 1: Enable Bit15: AIF1 DAC0L data Bit14: AIF2 DACL data Bit13: ADCL data Bit12: AIF2 DACR data
11:8	R/W	0x0	AIF1_ADC0R_MXR_SRC AIF1 ADC Timeslot 0 right channel mixer source select 0: Disable 1: Enable Bit11: AIF1 DAC0R data Bit10: AIF2 DACR data Bit9: ADCR data Bit8: AIF2 DACL data
7:6	R/W	0x0	AIF1_ADC1L_MXR_SRC AIF1 ADC Timeslot 1 left channel mixer source select 0: Disable 1: Enable Bit7: AIF2 DACL data Bit6: ADCL data

5:4	R/W	0x0	Reserved
3:2	R/W	0x0	AIF1_ADC1R_MXR_SRC AIF1 ADC Timeslot 1 right channel mixer source select 0: Disable 1: Enable Bit3: AIF2 DACR data Bit2: ADCR data
1:0	R/W	0x0	Reserved

9.3.4.26 AIF1 Volume Control 1 Register(Default Value: 0x0000_A0A0)

Offset: 0x0250			Register Name: AIF1_VOL_CTRL1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:8	R/W	0xA0	AIF1_ADC0L_VOL AIF1 ADC Timeslot 0 left channel volume (-119.25dB To 71.25dB, 0.75dB/Step) 0x00: Mute 0x01: -119.25dB …… 0x9F : -0.75dB 0xA0 : 0dB 0xA1: 0.75dB …… 0xFF: 71.25dB
7:0	R/W	0xA0	AIF1_ADC0R_VOL AIF1 ADC Timeslot 0 right channel volume (-119.25dB To 71.25dB, 0.75dB/Step) 0x00: Mute 0x01: -119.25dB …… 0x9F: -0.75dB 0xA0: 0dB 0xA1: 0.75dB …… 0xFF: 71.25dB

9.3.4.27 AIF1 Volume Control 2 Register(Default Value: 0x0000_A0A0)

Offset: 0x0254			Register Name: AIF1_VOL_CTRL2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:8	R/W	0xA0	AIF1_ADC1L_VOL AIF1 ADC Timeslot 1 left channel volume

			(-119.25dB To 71.25dB, 0.75dB/Step) 0x00: Mute 0x01: -119.25dB …… 0x9F: -0.75dB 0xA0: 0dB 0xA1: 0.75dB …… 0xFF: 71.25dB
7:0	R/W	0xA0	AIF1_ADC1R_VOL AIF1 ADC Timeslot 1 right channel volume (-119.25dB To 71.25dB, 0.75dB/Step) 0x00: Mute 0x01: -119.25dB …… 0x9F: -0.75dB 0xA0: 0dB 0xA1: 0.75dB …… 0xFF: 71.25dB

9.3.4.28 AIF1 Volume Control 3 Register(Default Value: 0x0000_A0A0)

Offset: 0x0258			Register Name: AIF1_VOL_CTRL3
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:8	R/W	0xA0	AIF1_DAC0L_VOL AIF1 DAC Timeslot 0 left channel volume (-119.25dB To 71.25dB, 0.75dB/Step) 0x00: Mute 0x01: -119.25dB …… 0x9F: -0.75dB 0xA0: 0dB 0xA1: 0.75dB …… 0xFF: 71.25dB
7:0	R/W	0xA0	AIF1_DAC0R_VOL AIF1 DAC Timeslot 0 right channel volume (-119.25dB To 71.25dB, 0.75dB/Step) 0x00: Mute 0x01: -119.25dB …… 0x9F: -0.75dB 0xA0: 0dB

			0xA1: 0.75dB …… 0xFF: 71.25dB

9.3.4.29 AIF1 Volume Control 4 Register(Default Value: 0x0000_A0A0)

Offset: 0x025C			Register Name: AIF1_VOL_CTRL4
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:8	R/W	0xA0	AIF1_DAC1L_VOL AIF1 DAC Timeslot 1 left channel volume (-119.25dB To 71.25dB, 0.75dB/Step) 0x00: Mute 0x01: -119.25dB …… 0x9F: -0.75dB 0xA0: 0dB 0xA1: 0.75dB …… 0xFF: 71.25dB
7:0	R/W	0xA0	AIF1_DAC1R_VOL AIF1 DAC Timeslot 1 right channel volume (-119.25dB To 71.25dB, 0.75dB/Step) 0x00: Mute 0x01: -119.25dB …… 0x9F: -0.75dB 0xA0: 0dB 0xA1: 0.75dB …… 0xFF: 71.25dB

9.3.4.30 AIF1 Digital Mixer Gain Control Register(Default Value: 0x0000_0000)

Offset: 0x0260			Register Name: AIF1_MXR_GAIN
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:12	R/W	0x0	AIF1_ADC0L_MXR_GAIN AIF1 ADC Timeslot 0 left channel mixer gain control 0: 0dB 1: -6dB Bit15: AIF1 DAC0L data Bit14: AIF2 DACL data Bit13: ADCL data

			Bit12: AIF2 DACR data
11:8	R/W	0x0	AIF1_ADC0R_MXR_GAIN AIF1 ADC Timeslot 0 right channel mixer gain control 0: 0dB 1: -6dB Bit11: AIF1 DAC0R data Bit10: AIF2 DACR data Bit9: ADCR data Bit8: AIF2 DACL data
7:6	R/W	0x0	AIF1_ADC1L_MXR_GAIN AIF1 ADC Timeslot 1 left channel mixer gain control 0: 0dB 1: -6dB Bit7: AIF2 DACL data Bit6: ADCL data
5:4	R/W	0x0	Reserved
3:2	R/W	0x0	AIF1_ADC1R_MXR_GAIN AIF1 ADC Timeslot 1 right channel mixer gain control 0: 0dB 1: -6dB Bit3: AIF2 DACR data Bit2: ADCR data
1:0	R/W	0x0	Reserved

9.3.4.31 AIF1 Receiver Data Discarding Control Register(Default Value: 0x0000_0000)

Offset: 0x0264			**Register Name: AIF1_RXD_CTRL**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:8	R/W	0x0	After data receiving progress begins, the first N-data will be discarded. N defined as follows: 0x00: None discarded 0x01: 1-data discarded ... 0xFF: 255-data discarded
7:0	R/W	0x0	Reserved

9.3.4.32 AIF2 BCLK/LRCK Control Register(Default Value: 0x0000_0000)

Offset: 0x0280			**Register Name: AIF2_CLK_CTRL**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R/W	0x0	AIF2_MSTR_MOD AIF2 Audio Interface mode select

			0: Master mode 1: Slave mode
14	R/W	0x0	AIF2_BCLK_INV AIF2 BCLK Polarity 0: Normal 1: Inverted
13	R/W	0x0	AIF2_LRCK_INV AIF2 LRCK Polarity 0: Normal 1: Inverted
12:9	R/W	0x0	AIF2_BCLK_DIV Select the AIF2CLK/BCLK2 ratio 0000: AIF2CLK/1 0001: AIF2CLK/2 0010: AIF2CLK/4 0011: AIF2CLK/6 0100: AIF2CLK/8 0101: AIF2CLK/12 0110: AIF2CLK/16 0111: AIF2CLK/24 1000: AIF2CLK/32 1001: AIF2CLK/48 1010: AIF2CLK/64 1011: AIF2CLK/96 1100: AIF2CLK/128 1101: AIF2CLK/192 1110: Reserved 1111: Reserved
8:6	R/W	0x0	AIF2_LRCK_DIV Select the BCLK2/LRCK2 ratio 000: 16 001: 32 010: 64 011: 128 100: 256 1xx: Reserved
5:4	R/W	0x0	AIF2_WORD_SIZ AIF2 digital interface world length 00: 8-bit 01: 16-bit 10: 20-bit 11: 24-bit
3:2	R/W	0x0	AIF2_DATA_FMT AIF digital interface data format 00: I2S mode 01: Left mode

			10: Right mode 11: DSP mode
1	R/W	0x0	AIF2_MONO_PCM AIF2 Mono PCM mode select 0: Stereo mode select 1: Mono mode select
0	R/W	0x0	Reserved

9.3.4.33 AIF2 ADCDAT Control Register(Default Value: 0x0000_0000)

Offset: 0x0284			Register Name: AIF2_ADCDAT_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R/W	0x0	AIF2_ADCL_EN AIF2 ADC left channel enable 0: Disable 1: Enable
14	R/W	0x0	AIF2_ADCR_EN AIF2 ADC right channel enable 0: Disable 1: Enable
13:12	/	/	/
11:10	R/W	0x0	AIF2_ADCL_SRC AIF2 ADC left channel data source select 00: AIF2 ADCL 01: AIF2 ADCR 10: (AIF2 ADCL+AIF2 ADCR) 11: (AIF2 ADCL+AIF2 ADCR)/2
9:8	R/W	0x0	AIF2_ADCR_SRC AIF2 ADC right channel data source select 00: AIF2 ADCR 01: AIF2 ADCL 10: (AIF2 ADCL+AIF2 ADCR) 11: (AIF2 ADCL+AIF2 ADCR)/2
7:4	/	/	/
3	R/W	0x0	AIF2_ADCP_ENA AIF2 ADC Companding enable(8-bit mode only) 00: Disable 01: Enable
2	R/W	0x0	AIF2_ADCUL_ENA AIF2 ADC Companding mode select 0: A-law 1: u-law
1:0	/	/	/

9.3.4.34 AIF2 DACDAT Control Register(Default Value: 0x0000_0000)

Offset: 0x0288			Register Name: AIF2_DACDAT_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R/W	0x0	AIF2_DACL_ENA AIF2 DAC left channel enable 0: Disable 1: Enable
14	R/W	0x0	AIF2_DACR_ENA AIF2 DAC right channel enable 0: Disable 1: Enable
13:12	R/W	0x0	Reserved
11:10	R/W	0x0	AIF2_DACL_SRC AIF2 DAC left channel data source select 00: AIF2 DACL 01: AIF2 DACR 10: (AIF2 DACL+AIF2 DACR) 11: (AIF2 DACL+AIF2 DACR)/2
9:8	R/W	0x0	AIF2_DACR_SRC AIF2 DAC right channel data source select 00: AIF2 DACR 01: AIF2 DACL 10: (AIF2 DACL+AIF2 DACR) 11: (AIF2 DACL+AIF2 DACR)/2
7:4	R/W	0x0	Reserved
3	R/W	0x0	AIF2_DACP_ENA AIF2 DAC Companding enable(8-bit mode only) 00: Disable 01: Enable
2	R/W	0x0	AIF2_DACUL_ENA AIF2 DAC Companding mode select 0: A-law 1: u-law
1	/	/	/
0	R/W	0x0	AIF2_LOOP_EN AIF2 loopback enable 0: No loopback 1: Loopback(ADCDAT2 data output to DACDAT2 data input)

9.3.4.35 AIF2 Digital Mixer Source Select Register(Default Value: 0x0000_0000)

Offset: 0x028C			Register Name: AIF2_MXR_SRC
Bit	**Read/Write**	**Default/Hex**	**Description**

31:16	/	/	/
15:12	R/W	0x0	AIF2_ADCL_MXR_SRC AIF2 ADC left channel mixer source select 0: Disable 1:Enable Bit15: AIF1 DAC0L data Bit14: AIF1 DAC1L data Bit13: AIF2 DACR data Bit12: ADCL data
11:8	R/W	0x0	AIF2_ADCR_MXR_SRC AIF2 ADC right channel mixer source select 0: Disable 1:Enable Bit11: AIF1 DA0R data Bit10: AIF1 DA1R data Bit9: AIF2 DACL data Bit8: ADCR data
7:0	R/W	0x0	Reserved

9.3.4.36 AIF2 Volume Control 1 Register(Default Value: 0x0000_A0A0)

Offset: 0x0290			Register Name: AIF2_VOL_CTRL1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:8	R/W	0xA0	AIF2_ADCL_VOL AIF2 ADC left channel volume (-119.25dB To 71.25dB, 0.75dB/Step) 0x00: Mute 0x01: -119.25dB …… 0x9F = -0.75dB 0xA0 = 0dB 0xA1 = 0.75dB …… 0xFF = 71.25dB
7:0	R/W	0xA0	AIF2_ADCR_VOL AIF2 ADC right channel volume (-119.25dB To 71.25dB, 0.75dB/Step) 0x00: Mute 0x01: -119.25dB …… 0x9F = -0.75dB 0xA0 = 0dB 0xA1 = 0.75dB ……

			0xFF = 71.25dB

9.3.4.37 AIF2 Volume Control 2 Register(Default Value: 0x0000_A0A0)

Offset: 0x0298			Register Name: AIF2_VOL_CTRL2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:8	R/W	0xA0	AIF2_DACL_VOL AIF2 DAC left channel volume (-119.25dB To 71.25dB, 0.75dB/Step) 0x00: Mute 0x01: -119.25dB …… 0x9F = -0.75dB 0xA0 = 0dB 0xA1 = 0.75dB …… 0xFF = 71.25dB
7:0	R/W	0xA0	AIF2_DACR_VOL AIF2 DAC right channel volume (-119.25dB To 71.25dB, 0.75dB/Step) 0x00: Mute 0x01: -119.25dB …… 0x9F = -0.75dB 0xA0 = 0dB 0xA1 = 0.75dB …… 0xFF = 71.25dB

9.3.4.38 AIF2 Digital Mixer Gain Control Register(Default Value: 0x0000_0000)

Offset: 0x02A0			Register Name: AIF2_MXR_GAIN
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:12	R/W	0x0	AIF2_ADCL_MXR_GAIN AIF2 ADC left channel mixer gain control 0: 0dB 1: -6dB Bit15: AIF1 DAC0L data Bit14: AIF1 DAC1L data Bit13: AIF2 DACR data Bit12: ADCL data
11:8	R/W	0x0	AIF2_ADCR_MXR_GAIN

			AIF2 ADC right channel mixer gain control 0: 0dB 1: -6dB Bit11: AIF1 DAC0R data Bit10: AIF1 DAC1R data Bit9: AIF2 DACL data Bit8: ADCR data
7:0	R/W	0x0	Reserved

9.3.4.39 AIF2 Receiver Data Discarding Control Register(Default Value: 0x0000_0000)

Offset: 0x02A4			Register Name: AIF2_RXD_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:8	R/W	0x0	After data receiving progress begins, the first N-data will be discarded. N defined as follows: 0x00: None discarded 0x01: 1-data discarded ... 0xFF: 255-data discarded
7:0	R/W	0x0	Reserved

9.3.4.40 AIF3 BCLK/LRCK Control Register(Default Value: 0x0000_0000)

Offset: 0x02C0			Register Name: AIF3_CLK_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:15	/	/	/
14	R/W	0x0	AIF3_BCLK_INV AIF3 BCLK Polarity 0: Normal 1: Inverted
13	R/W	0x0	AIF3_LRCK_INV AIF3 LRCK Polarity 0: Normal 1: Inverted
12:6	R/W	0x0	Reserved
5:4	R/W	0x0	AIF3_WORD_SIZ AIF3 digital interface world length 00: 8-bit 01: 16-bit 10: 20-bit 11: 24-bit
3:2	R/W	0x0	Reserved
1:0	R/W	0x0	AIF3_CLOCK_SRC

			AIF3 BCLK/LRCK source control 00: BCLK/LRCK Come from AIF1 01: BCLK/LRCK Come from AIF2 10: BCLK/LRCK is generated by AIF3, and the source clock is AIF1CLK 11: Reserved

9.3.4.41 AIF3 ADCDAT Control Register(Default Value: 0x0000_0000)

Offset: 0x02C4			Register Name: AIF3_ADCDAT_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:4	/	/	/
3	R/W	0x0	AIF3_ADCP_ENA AIF3 ADC Companding enable 00: Disable 01: Enable
2	R/W	0x0	AIF3_ADUL_ENA AIF3 ADC Companding mode select 0: A-law 1: u-law
1:0	R/W	0x0	Reserved

9.3.4.42 AIF3 DACDAT Control Register(Default Value: 0x0000_0000)

Offset: 0x02C8			Register Name: AIF3_DACDAT_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:4	/	/	/
3	R/W	0x0	AIF3_DACP_ENA AIF3 DAC Companding enable(8-bit mode only) 00: Disable 01: Enable
2	R/W	0x0	AIF3_DAUL_ENA AIF3 DAC Companding mode select 00: u-law 01: A-law
1	R/W	0x0	Reserved
0	R/W	0x0	AIF3_LOOP_ENA AIF3 loopback enable 0: No loopback 1: Loopback(ADCDAT3 data output to DACDAT3 data input)

9.3.4.43 AIF3 Signal Path Control Register(Default Value: 0x0000_0000)

Offset: 0x02CC	Register Name: AIF3_SGP_CTRL

Bit	Read/Write	Default/Hex	Description
31:12	/	/	/
11:10	R/W	0x0	AIF3_ADC_SRC AIF3 PCM output source select 00: None 01: AIF2 ADC left channel 10: AIF2 ADC right channel 11: Reserved
9:8	R/W	0x0	AIF2_DAC_SRC AIF2 DAC input source select 00: Left and right inputs from AIF2 01: Left input from AIF3; Right input from AIF2 10: Left input from AIF2; Right input from AIF3 11: Reserved
7	R/W	0x0	AIF3_PINS_TRI AIF3 Pins Tri-state Control 0: AIF3 pins operate normally 1: Tri-state all AIF3 interface pins
6:4	R/W	0x0	AIF3_ADCDAT_SRC AIF3 ADCDAT Source select 0xx: AIF3 Mono PCM output 100: AIF1 ADCDAT1 101: AIF1 DACDAT1 110: AIF2 ADCDAT2 111: AIF2 DACDAT2
3	R/W	0x0	AIF2_ADCDAT_SRC AIF2 ADCDAT2 Source select 0: AIF2 ADCDAT2 1: AIF3 DACDAT3
2	R/W	0x0	AIF2_DACDAT_SRC AIF2 DACDAT2 Source select 0: AIF2 DACDAT2 1: AIF3 DACDAT3
1	R/W	0x0	AIF1_ADCDAT_SRC AIF1 ADCDAT1 Source select 0: AIF1 ADCDAT1 1: AIF3 DACDAT3
0	R/W	0x0	AIF1_DACDAT_SRC AIF1 DACDAT1 Source select 0: AIF1 DACDAT1 1: AIF3 DACDAT3

9.3.4.44 AIF3 Receiver Data Discarding Control Register(Default Value: 0x0000_0000)

Offset: 0x02E4	Register Name: AIF3_RXD_CTRL

Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:8	R/W	0x0	After data receiving progress begins, the first N-data will be discarded. N defined as follows: 0x00: None discarded 0x01: 1-data discarded ... 0xFF: 255-data discarded
7:0	R/W	0x0	Reserved

9.3.4.45 ADC Digital Control Register(Default Value: 0x0000_0000)

Offset: 0x300			Register Name: ADC_DIG_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R/W	0x0	ENAD ADC Digital part enable 0: Disable 1: Enable
14	/	/	/
13	R/W	0x0	ADFIR32 Enable 32-tap FIR filter 0: 64-tap 1: 32-tap
12:5	R/W	0x0	Reserved
4	/	/	/
3:2	R/W	0x0	ADOUT_DTS ADC Delay Time For transmitting data after ENAD 00:5ms 01:10ms 10:20ms 11:30ms
1	R/W	0x0	ADOUT_DLY ADC Delay Function enable for transmitting data after ENAD 0: Disable 1: Enable
0	/	/	/

9.3.4.46 ADC Volume Control Register(Default Value: 0x0000_A0A0)

Offset: 0x0304			Register Name: ADC_VOL_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:8	R/W	0xA0	ADC_VOL_L

			ADC left channel volume (-119.25dB To 71.25dB, 0.75dB/Step) 0x00: Mute 0x01: -119.25dB …… 0x9F: -0.75dB 0xA0: 0dB 0xA1: 0.75dB …… 0xFF:71.25dB
7:0	R/W	0xA0	ADC_VOL_R ADC left channel volume (-119.25dB To 71.25dB, 0.75dB/Step) 0x00: Mute 0x01: -119.25dB …… 0x9F: -0.75dB 0xA0: 0dB 0xA1: 0.75dB …… 0xFF: 71.25dB

9.3.4.47 ADC Debug Control Register(Default Value: 0x0000_0000)

Offset: 0x0308			**Register Name: ADC_DBG_CTRL**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R/W	0x0	ADSW ADC input channel swap enable 0: Disable 1: Enable
14:4	/	/	/
3:0	R/W	0x0	Interface between ADDA digital part and ADDA analog part debug control. 0000: Normal 0001: BCLK2/LRCK2/ADCDAT2/DACDAT2 BCLK3/LRCK3/ADCDAT3/DACDAT3 is multiplexed as follows: BCLK2 – CKAD_A input to ADC analog part LRCK2 – Normal ADCDAT2 – ADIN_L[1] output from ADC analog part DACDAT2 – ADIN_L[0] output from ADC analog part BCLK3 – CKAD_A output from ADC digital part LRCK3 – Normal ADCDAT3 – ADIN_L/R[1] input to ADC digital part DACDAT3 – ADIN_L/R[0] input to ADC digital part 0010: BCLK2/LRCK2/ADADAT2/DACDAT2

BCLK3/LRCK3/ADCDAT3/DACDAT3 is multiplexed as follows:

BCLK2	CKAD_A input to ADC analog part
LRCK2	Normal
ADCDAT2	ADIN_R[1] output from ADC analog part
DACDAT2	ADIN_R[0] output from ADC analog part
BCLK3	CKAD_A output from ADC digital part
LRCK3	Normal
ADCDAT3	ADIN_L/R[1] input to ADC digital part
DACDAT3	ADIN_L/R[0] input to ADC digital part

0011: BCLK2/LRCK2/ADADAT2/DACDAT2

BCLK3/LRCK3/ADCDAT3/DACDAT3 is multiplexed as follows:

BCLK2	CKDA_A output from DAC digital part
LRCK2	DAOUT_L[6] output from DAC digital part
ADCDAT2	DAOUT_L[5] output from DAC digital part
DACDAT2	DAOUT_L[4] output from DAC digital part
BCLK3	DAOUT_L[3] output from DAC digital part
LRCK3	DAOUT_L[2] output from DAC digital part
ADCDAT3	DAOUT_L[1] output from DAC digital part
DACDAT3	DAOUT_L[0] output from DAC digital part

0100: BCLK2/LRCK2/ADADAT2/DACDAT2

BCLK3/LRCK3/ADCDAT3/DACDAT3 is multiplexed as follows:

BCLK2	CKDA_A output from DAC digital part
LRCK2	DAOUT_R[6] output from DAC digital part
ADCDAT2	DAOUT_R[5] output from DAC digital part
DACDAT2	DAOUT_R[4] output from DAC digital part
BCLK3	DAOUT_R[3] output from DAC digital part
LRCK3	DAOUT_R[2] output from DAC digital part
ADCDAT3	DAOUT_R[1] output from DAC digital part
DACDAT3	DAOUT_R[0] output from DAC digital part

0101: BCLK2/LRCK2/ADADAT2/DACDAT2

BCLK3/LRCK3/ADCDAT3/DACDAT3 is multiplexed as follows:

BCLK2	CKDA_A input to DAC analog part
LRCK2	DAOUT_L/R[6] input to DAC analog part
ADCDAT2	DAOUT_L/R[5] input to DAC analog part
DACDAT2	DAOUT_L/R[4] input to DAC analog part
BCLK3	DAOUT_L/R[3] input to DAC analog part
LRCK3	DAOUT_L/R[2] input to DAC analog part
ADCDAT3	DAOUT_L/R[1] input to DAC analog part
DACDAT3	DAOUT_L/R[0] input to DAC analog part

0110: BCLK2/LRCK2/ADADAT2/DACDAT2

BCLK3/LRCK3/ADCDAT3/DACDAT3 is multiplexed as follows:

BCLK2	CKDA_A input to DAC analog part
LRCK2	DAOUT_L/R[6] input to DAC analog part
ADCDAT2	DAOUT_L/R[5] input to DAC analog part
DACDAT2	DAOUT_L/R[4] input to DAC analog part

			BCLK3 DAOUT_L/R[3] input to DAC analog part LRCK3 DAOUT_L/R[2] input to DAC analog part ADCDAT3 DAOUT_L/R[1] input to DAC analog part DACDAT3 DAOUT_L/R[0] input to DAC analog part 0111: Reserved. 1000: BCLK2/LRCK2/ADADAT2/DACDAT2 BCLK3/LRCK3/ADCDAT3/DACDAT3 is multiplexed as follows: BCLK2 HMICADC_EN LRCK2 HMICADC_CLK ADCDAT2 HMICADC[4] DACDAT2 HMICADC[3] BCLK3 HMICADC[2] LRCK3 HMICADC[1] ADCDAT3 HMICADC[0] 1001: BCLK2/LRCK2/ADADAT2/DACDAT2 BCLK3/LRCK3/ADCDAT3/DACDAT3 is multiplexed as follows: BCLK2 JACKDEC_EN LRCK2 JACK_CK ADCDAT2 JACK_DET

9.3.4.48 HMIC Control 1 Register(Default Value: 0x0000_0020)

Offset: 0x0310			Register Name: HMIC_CTRL1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:12	R/W	0x0	HMIC_M Debounce when the MIC Key down or up. 0000:1 samlpe data 0001:2 samlpe data ... 1111 :16 samlpe data
11:8	R/W	0x0	HMIC_N Debounce when earphone plug in or pull out 125ms~2s 0000:125ms 0001:250ms … 1111:2s
7:5	R/W	0x1	MDATA_Threshold_Debounce 000:0 001:1 010:2 011:3

			100:4 101:5 110:6 111:7
4	R/W	0x0	JACK_OUT_IRQ_EN MIC Detect Interrupt Set 0: Disable 1: Enable
3	R/W	0x0	JACK_IN_IRQ_EN MIC Detect Interrupt Set 0: Disable 1: Enable
2:1	/	/	/
0	R/W	0x0	MIC_DET_IRQ_EN MIC Detect Interrupt Set 0: Disable 1: Enable

9.3.4.49 HMIC Control 2 Register(Default Value: 0x0000_0000)

Offset: 0x0314			Register Name: HMIC_CTRL2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:14	R/W	0x0	HMIC_SAMPLE_SELECT Down Sample Setting Select 00 : Down by 1, 128Hz 01 : Down by 2, 64Hz 10 : Down by 4, 32Hz 11 : Down by 8, 16Hz
13	/	/	/
12:8	R/W	0x0	MDATA_Threshold
7:6	R/W	0x0	HMIC_SF HMIC Smooth Filter setting 00: by pass 01: (x1+x2)/2 10: (x1+x2+x3+x4)/4 11: (x1+x2+x3+x4+ x5+x6+x7+x8)/8
5:0	/	/	/

9.3.4.50 HMIC Status Register(Default Value: 0x0000_6000)

Offset: 0x0318			Register Name: HMIC_STS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:15	/	/	/

14:13	R/W	0x3	MDATA_DISCARD After MIC DATA data receiving, the first N-data will be discarded. N defined as follows: 00: None discarded 01: 1-data discarded 10: 2-data discarded 11: 4-data discarded
12:8	R	0x0	HMIC_DATA HMIC Average Data Detect which key is pressed by the value.
7:5	/	/	/
4	R/W1C	0x0	JACK_DET_OIRQ Jack output detect Pending interrupt 0: No Pending IRQ 1: Pending IRQ Writing 1 clear pending
3	R/W1C	0x0	JACK_DET_IIRQ Jack input detect pending interrupt 0: No Pending IRQ 1: Pending IRQ Write 1 to clear pending
2:1	/	/	/
0	R/W1C	0x0	MIC_DET_ST. MIC Detect Pending interrupt 0: No pending IRQ 1: Pending IRQ Write 1 to clear pending

9.3.4.51 DAC Digital Control Register(Default Value: 0x0000_0000)

Offset: 0x0320			**Register Name: DAC_DIG_CTRL**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R/W	0x0	ENDA. DAC Digital Part Enable 0: Disable 1: Enable
14	R/W	0x0	ENHPF HPF Function Enable 0: Enable 1: Disable
13	R/W	0x0	DAFIR32 Enable 32-tap FIR filter 0: 64-tap 1: 32-tap

12	R/W	0x0	Reserved
11:8	R/W	0x0	MODQU Internal DAC Quantization Levels Levels=[7*(21+MODQU[3:0])]/128 Default levels=7*21/128=1.15
7:0	R/W	0x0	Reserved

9.3.4.52 DAC Volume Control Register(Default Value: 0x0000_A0A0)

Offset: 0x0324			Register Name: DAC_VOL_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:8	R/W	0xA0	DAC_VOL_L DAC left channel volume (-119.25dB To 71.25dB, 0.75dB/Step) 0x00: Mute 0x01: -119.25dB …… 0x9F: -0.75dB 0xA0: 0dB 0xA1: 0.75dB …… 0xFF: 71.25dB
7:0	R/W	0xA0	DAC_VOL_R DAC right channel volume (-119.25dB To 71.25dB, 0.75dB/Step) 0x00: Mute 0x01: -119.25dB …… 0x9F: -0.75dB 0xA0: 0dB 0xA1: 0.75dB …… 0xFF: 71.25dB

9.3.4.53 DAC Debug Control Register(Default Value: 0x0000_0000)

Offset: 0x0328			Register Name: DAC_DBG_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R/W	0x0	DASW DAC output channel swap enable 0:Disable 1:Enable

14	R/W	0x0	ENDWA_N DWA Function Disable 0: Enable 1: Disable
13	R/W	0x0	DAC_MOD_DBG DAC Modulator Debug 0: DAC Modulator Normal Mode 1: DAC Modulator Debug Mode
12:8	R/W	0x0	Reserved
7:6	R/W	0x0	DAC_PTN_SEL DAC Pattern Select 00: Normal(Audio sample from DAC mixer) 01: -6 dB sin wave 10: -60 dB sin wave 11: zero data
5:0	R/W	0x0	DVC Digital volume control, ATT=DVC[5:0]*(-1.16dB) 64 steps, -1.16dB/step

9.3.4.54 DAC Digital Mixer Source Select Register(Default Value: 0x0000_0000)

Offset: 0x0330			Register Name: DAC_MXR_SRC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:12	R/W	0x0	DACL_MXR_SRC DAC left channel mixer source select 0: Disable 1:Enable Bit15: AIF1 DAC0L Bit14: AIF1 DAC1L Bit13: AIF2 DACL Bit12: ADCL
11:8	R/W	0x0	DACR_MXR_SRC DAC right channel mixer source select 0: Disable 1:Enable Bit11: AIF1 DAC0R Bit10: AIF1 DAC1R Bit9: AIF2 DACR Bit8: ADCR
7:0	R/W	0x0	Reserved

9.3.4.55 DAC Digital Mixer Gain Control Register(Default Value: 0x0000_0000)

Offset: 0x0334			Register Name: DAC_MXR_GAIN
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:12	R/W	0x0	DACL_MXR_GAIN DAC left channel mixer gain control 0: 0dB 1: -6dB Bit15: AIF1 DAC0L Bit14: AIF1 DAC1L Bit13: AIF2 DACL Bit12: ADCL
11:8	R/W	0x0	DACR_MXR_GAIN DAC right channel mixer gain control 0: 0dB 1: -6dB Bit11: AIF1 DAC0R Bit10: AIF1 DAC1R Bit9: AIF2 DACR Bit8: ADCR
7:0	R/W	0x0	Reserved

9.3.4.56 DAC DAP Control Register(Default Value: 0x0000_0000)

Offset: 0x0480			Register Name: AC_DAC_DAPCTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:7	/	/	/
6	R/W	0x0	DRC1 enable control 0: Disable 1: Enable
5	R/W	0x0	DRC1 left channel HPF enable control 0: Disable 1: Enable
4	R/W	0x0	DRC1 right channel HPF enable control 0: Disable 1: Enable
3	/	/	/
2	R/W	0x0	DRC0 enable control 0: Disable 1: Enable
1	R/W	0x0	DRC0 left channel HPF enable control 0: Disable 1: Enable
0	R/W	0x0	DRC0 right channel HPF enable control

			0: Disable 1: Enable

9.3.4.57 DRC Enable Register(Default Value: 0x0000_0000)

Offset: 0x04D4			Register Name: DRC_ENA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R/W	0x0	AIF1_DAC0_DRC0_ENA AIF1 DAC Timeslot 0 DRC0 enable 0: Disable 1: Enable
14	R/W	0x0	AIF1_DAC1_DRC0_ENA AIF1 DAC Timeslot 1 DRC0 enable 0: Disable 1: Enable
13	R/W	0x0	AIF2_DAC_DRC0_ENA AIF2 DAC DRC0 enable 0: Disable 1: Enable
12	R/W	0x0	DAC_DRC0_ENA DAC DRC0 enable 0: Disable 1: Enable
11:8	/	/	/
7	R/W	0x0	AIF1_ADC0_DRC1_ENA AIF1 ADC Timeslot 0 DRC1 enable 0 : Disable 1 : Enable
6	R/W	0x0	AIF1_ADC1_DRC1_ENA AIF1 ADC Timeslot 1 DRC1 enable 0 : Disable 1 : Enable
5	R/W	0x0	AIF2_ADC_DRC1_ENA AIF2 ADC DRC1 enable 0 : Disable 1 : Enable
4	R/W	0x0	ADC_DRC1_ENA ADC_DRC1 enable 0 : Disable 1 : Enable
3:0	/	/	/

9.3.4.58 DRC0 High HPF Coef Register(Default Value: 0x0000_00FF)

Offset: 0x0600			Register Name: AC_DRC0_HHPFC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R/W	0x0FF	HPF coefficient setting and the data is 3.24 format.

9.3.4.59 DRC0 Low HPF Coef Register(Default Value: 0x0000_FAC1)

Offset: 0x0604			Register Name: AC_DRC0_LHPFC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0xFAC1	HPF coefficient setting and the data is 3.24 format.

9.3.4.60 DRC0 Control Register(Default Value: 0x0000_0080)

Offset: 0x0608			Register Name: AC_DRC0_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R	0x0	DRC delay buffer data output state when DRC delay function is enable and the DRC function disable. After disable DRC function and this bit go to 0, the user should write the DRC delay function bit to 0. 0 : Not complete 1 : Complete
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 uses or not when the DRC disables and the DRC buffer data outputs completely 0 : Not use the buffer 1 : Use the buffer
6	R/W	0x0	DRC gain max limit enable 0 : Disable 1 : Enable
5	R/W	0x0	DRC gain min limit enable. when this function is enabled, it will overwrite the noise detect function.

			0 : Disable 1 : Enable
4	R/W	0x0	Control the DRC to detect noise when ET enable 0 : Disable 1 : Enable
3	R/W	0x0	Signal function select 0 : RMS filter 1 : Peak filter When signal function selects Peak filter, the RMS parameter is unused. (AC_DRC_LRMSHAT / AC_DRC_LRMSLAT / AC_DRC_LRMSHAT / AC_DRC_LRMSLAT) When Signal function selects RMS filter, the Peak filter parameter is unused.(AC_DRC_LPFHAT / AC_DRC_LPFLAT / AC_DRC_RPFHAT / AC_DRC_RPFLAT / AC_DRC_LPFHRT / AC_DRC_LPFLRT / AC_DRC_RPFHRT / AC_DRC_RPFLRT)
2	R/W	0x0	Delay function enable 0 : Disable 1 : Enable When the Delay function enable is disabled, the Signal delay time is unused.
1	R/W	0x0	DRC LT enable 0 : Disable 1 : Enable When the DRC LT is disabled the LT, Kl and OPL parameter is unused.
0	R/W	0x0	DRC ET enable 0 : Disable 1 : Enable When the DRC ET is disabled the ET, Ke and OPE parameter is unused.

9.3.4.61 DRC0 Left Peak Filter High Attack Time Coef Register(Default Value: 0x0000_000B)

Offset: 0x060C			Register Name: AC_DRC0_LPFHAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R/W	0x00B	The left peak filter attack time parameter setting, which determine by the equation that AT = 1-exp(-2.2Ts/ta). The format is 3.24. (1ms)

9.3.4.62 DRC0 Left Peak Filter Low Attack Time Coef Register(Default Value: 0x0000_77BF)

Offset: 0x0610			Register Name: AC_DRC0_LPFLAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x77BF	The left peak filter attack time parameter setting, which determine by the equation that AT = 1-exp(-2.2Ts/ta). The format is 3.24. (1ms)

9.3.4.63 DRC0 Right Peak Filter High Attack Time Coef Register(Default Value: 0x0000_000B)

Offset: 0x0614			Register Name: AC_DRC0_RPFHAT
Bit	Read/Write	Default/Hex	Description
31:11	/	/	/
10:0	R/W	0x00B	The left peak filter attack time parameter setting, which determine by the equation that AT = 1-exp(-2.2Ts/ta). The format is 3.24. (1ms)

9.3.4.64 DRC0 Peak Filter Low Attack Time Coef Register(Default Value: 0x0000_77BF)

Offset: 0x0618			Register Name: AC_DRC0_RPFLAT
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x77BF	The left peak filter attack time parameter setting, which determine by the equation that AT = 1-exp(-2.2Ts/ta). The format is 3.24. (1ms)

9.3.4.65 DRC0 Left Peak Filter High Release Time Coef Register(Default Value: 0x0000_00FF)

Offset: 0x061C			Register Name: AC_DRC0_LPFHRT
Bit	Read/Write	Default/Hex	Description
31:11	/	/	/
10:0	R/W	0x0FF	The left peak filter release time parameter setting, which determine by the equation that RT = exp(-2.2Ts/tr). The format is 3.24. (100ms)

9.3.4.66 DRC0 Left Peak Filter Low Release Time Coef Register(Default Value: 0x0000_E1F8)

Offset: 0x0620			Register Name: AC_DRC0_LPFLRT
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0xE1F8	The left peak filter release time parameter setting, which determine by the equation that RT = exp(-2.2Ts/tr). The format is 3.24. (100ms)

9.3.4.67 DRC0 Right Peak Filter High Release Time Coef Register(Default Value: 0x0000_00FF)

Offset: 0x0624			Register Name: AC_DRC0_RPFHRT
Bit	Read/Write	Default/Hex	Description
31:11	/	/	/
10:0	R/W	0x0FF	The left peak filter attack time parameter setting, which determine by the equation that RT = exp(-2.2Ts/tr). The format is 3.24. (100ms)

9.3.4.68 DRC0 Right Peak Filter Low Release Time Coef Register(Default Value: 0x0000_E1F8)

Offset: 0x0628			Register Name: AC_DRC0_RPFLRT
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0xE1F8	The left peak filter release time parameter setting, which determine by the equation that AT = exp(-2.2Ts/tr). The format is 3.24. (100ms)

9.3.4.69 DRC0 Left RMS Filter High Coef Register(Default Value: 0x0000_0001)

Offset: 0x062C			Register Name: AC_DRC0_LRMSHAT
Bit	Read/Write	Default/Hex	Description
31:11	/	/	/
10:0	R/W	0x001	The left RMS filter average time parameter setting, which determine by the equation that AT = 1-exp(-2.2Ts/tav). The format is 3.24. (10ms)

9.3.4.70 DRC0 Left RMS Filter Low Coef Register(Default Value: 0x0000_2BAF)

Offset: 0x0630			Register Name: AC_DRC0_LRMSLAT
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x2BAF	The left RMS filter average time parameter setting, which determine by the equation that AT = 1-exp(-2.2Ts/tav). The format is 3.24. (10ms)

9.3.4.71 DRC0 Right RMS Filter High Coef Register(Default Value: 0x0000_0001)

Offset: 0x0634			Register Name: AC_DRC0_RRMSHAT
Bit	Read/Write	Default/Hex	Description
31:11	/	/	/
10:0	R/W	0x001	The right RMS filter average time parameter setting, which determine by the equation that AT = 1-exp(-2.2Ts/tav). The format is 3.24. (10ms)

9.3.4.72 DRC0 Right RMS Filter Low Coef Register(Default Value: 0x0000_2BAF)

Offset: 0x0638			Register Name: AC_DRC0_RRMSLAT
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x2BAF	The right RMS filter average time parameter setting, which determine by the equation that AT = 1-exp(-2.2Ts/tav). The format is 3.24. (10ms)

9.3.4.73 DRC0 Compressor Threshold High Setting Register(Default Value: 0x0000_06A4)

Offset: 0x063C			Register Name: AC_DRC0_HCT
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x06A4	The compressor threshold setting, which set by the equation that CTin = -CT/6.0206. The format is 8.24 (-40dB)

9.3.4.74 DRC0 Compressor Threshold Low Setting Register(Default Value: 0x0000_D3C0)

Offset: 0x0640			Register Name: AC_DRC0_LCT
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0xD3C0	The compressor threshold setting, which set by the equation that CTin = -CT/6.0206. The format is 8.24 (-40dB)

9.3.4.75 DRC0 Compressor Slope High Setting Register(Default Value: 0x0000_0080)

Offset: 0x0644			Register Name: AC_DRC0_HKC
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13:0	R/W	0x0080	The slope of the compressor which determine by the equation that Kc = 1/R, there, R is the ratio of the compressor, which always is integer. The format is 6.24. (2 : 1)

9.3.4.76 DRC0 Compressor Slope Low Setting Register(Default Value: 0x0000_0000)

Offset: 0x0648			Register Name: AC_DRC0_LKC
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x0000	The slope of the compressor which determine by the equation that Kc = 1/R, there, R is the ratio of the compressor, which always is integer. The format is 6.24. (2 : 1)

9.3.4.77 DRC0 Compressor High Output at Compressor Threshold Register(Default Value: 0x0000_F95B)

Offset: 0x064C			Register Name: AC_DRC0_HOPC
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0xF95B	The output of the compressor which determine by the equation -OPC/6.0206 The format is 8.24.(-40dB)

9.3.4.78 DRC0 Compressor Low Output at Compressor Threshold Register(Default Value: 0x0000_2C3F)

Offset: 0x0650			Register Name: AC_DRC0_LOPC
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x2C3F	The output of the compressor which determine by the equation OPC/6.0206 The format is 8.24. (-40dB)

9.3.4.79 DRC0 Limiter Threshold High Setting Register(Default Value: 0x0000_01A9)

Offset: 0x0654			Register Name: AC_DRC0_HLT
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x01A9	The limiter threshold setting, which set by the equation that LTin = -LT/6.0206, The format is 8.24. (-10dB)

9.3.4.80 DRC0 Limiter Threshold Low Setting Register(Default Value: 0x0000_34F0)

Offset: 0x0658			Register Name: AC_DRC0_LLT
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x34F0	The limiter threshold setting, which set by the equation that LTin = -LT/6.0206, The format is 8.24. (-10dB)

9.3.4.81 DRC0 Limiter Slope High Setting Register(Default Value: 0x0000_0005)

Offset: 0x065C			Register Name: AC_DRC0_HKI
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13:0	R/W	0x0005	The slope of the limiter which determine by the equation that KI = 1/R, there, R is the ratio of the limiter, which always is integer. The format is 6.24. (50 :1)

9.3.4.82 DRC0 Limiter Slope Low Setting Register(Default Value: 0x0000_1EB8)

Offset: 0x0660			Register Name: AC_DRC0_LKI
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x1EB8	The slope of the limiter which determine by the equation that KI = 1/R,

			there, R is the ratio of the limiter, which always is integer. The format is 6.24. (50 :1)

9.3.4.83 DRC0 Limiter High Output at Limiter Threshold Register(Default Value: 0x0000_FBD8)

Offset: 0x0664			Register Name: AC_DRC0_HOPL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0xFBD8	The output of the limiter which determine by equation OPL/6.0206. The format is 8.24 .(-25dB)

9.3.4.84 DRC0 Limiter Low Output at Limiter Threshold Register(Default Value: 0x0000_FBA7)

Offset: 0x0668			Register Name: AC_DRC0_LOPL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0xFBA7	The output of the limiter which determine by equation OPL/6.0206. The format is 8.24 .(-25dB)

9.3.4.85 DRC0 Expander Threshold High Setting Register(Default Value: 0x0000_0BA0)

Offset: 0x066C			Register Name: AC_DRC0_HET
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x0BA0	The expander threshold setting, which set by the equation that ETin = -ET/6.0206, The format is 8.24. (-70dB)

9.3.4.86 DRC0 Expander Threshold Low Setting Register(Default Value: 0x0000_7291)

Offset: 0x0670			Register Name: AC_DRC0_LET
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x7291	The expander threshold setting, which set by the equation that ETin = -ET/6.0206, The format is 8.24. (-70dB)

9.3.4.87 DRC0 Expander Slope High Setting Register(Default Value: 0x0000_0500)

Offset: 0x0674			Register Name: AC_DRC0_HKE
Bit	**Read/Write**	**Default/Hex**	**Description**
31:14	/	/	/
13:0	R/W	0x0500	The slope of the expander which determine by the equation that Ke = 1/R,

			there, R is the ratio of the expander, which always is integer and the ke must larger than 1/50. The format is 6.24. (1:5)

9.3.4.88 DRC0 Expander Slope Low Setting Register(Default Value: 0x0000_0000)

Offset: 0x0678			Register Name: AC_DRC0_LKE
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x0000	The slope of the expander which determine by the equation that Ke = 1/R, there, R is the ratio of the expander, which always is integer and the ke must larger than 1/50. The format is 6.24. (1:5)

9.3.4.89 DRC0 Expander High Output at Expander Threshold Register(Default Value: 0x0000_F45F)

Offset: 0x067C			Register Name: AC_DRC0_HOPE
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0xF45F	The output of the expander which determine by equation OPE/6.0206. The format is 8.24. (-70dB)

9.3.4.90 DRC0 Expander Low Output at Expander Threshold Register(Default Value: 0x0000_8D6E)

Offset: 0x0680			Register Name: AC_DRC0_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 .(-70dB)

9.3.4.91 DRC0 Linear Slope High Setting Register(Default Value: 0x0000_0100)

Offset: 0x0684			Register Name: AC_DRC0_HKN
Bit	**Read/Write**	**Default/Hex**	**Description**
31:14	/	/	/
13:0	R/W	0x0100	The slope of the linear which determine by the equation that Kn = 1/R, there, R is the ratio of the linear, which always is integer . The format is 6.24. (1:1)

9.3.4.92 DRC0 Linear Slope Low Setting Register(Default Value: 0x0000_0000)

Offset: 0x0688			Register Name: AC_DRC0_LKN
Bit	**Read/Write**	**Default/Hex**	**Description**

31:16	/	/	/
15:0	R/W	0x0000	The slope of the linear which determine by the equation that Kn = 1/R, there, R is the ratio of the linear, which always is integer . The format is 6.24. (1:1)

9.3.4.93 DRC0 Smooth Filter Gain High Attack Time Coef Register(Default Value:0x0000_0002)

Offset: 0x068C			Register Name: AC_DRC0_SFHAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R/W	0x002	The smooth filter attack time parameter setting, which determine by the equation that AT = 1-exp(-2.2Ts/tr). The format is 3.24. (5ms)

9.3.4.94 DRC0 Smooth Filter Gain Low Attack Time Coef Register(Default Value: 0x0000_5600)

Offset: 0x0690			Register Name: AC_DRC0_SFLAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x5600	The smooth filter attack time parameter setting, which determine by the equation that AT = 1-exp(-2.2Ts/tr). The format is 3.24. (5ms)

9.3.4.95 DRC0 Smooth Filter Gain High Release Time Coef Register(Default Value:0x0000_0000)

Offset: 0x0694			Register Name: AC_DRC0_SFHRT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R/W	0x000	The gain smooth filter release time parameter setting, which determine by the equation that RT = 1-exp(-2.2Ts/tr). The format is 3.24. (200ms)

9.3.4.96 DRC0 Smooth Filter Gain Low Release Time Coef Register(Default Value: 0x0000_0F04)

Offset: 0x0698			Register Name: AC_DRC0_SFLRT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x0F04	The gain smooth filter release time parameter setting, which determine by the equation that RT = 1-exp(-2.2Ts/tr). The format is 3.24. (200ms)

9.3.4.97 DRC0 MAX Gain High Setting Register(Default Value: 0x0000_FE56)

Offset: 0x069C			Register Name: AC_DRC0_MXGHS
Bit	**Read/Write**	**Default/Hex**	**Description**

31:16	/	/	/
15:0	R/W	0xFE56	The max gain setting which determine by equation MXG/6.0206. The format is 8.24 and must -20dB <MXG< 30dB .(-10dB)

9.3.4.98 DRC0 MAX Gain Low Setting Register(Default Value: 0x0000_CB0F)

Offset: 0x06A0			Register Name: AC_DRC0_MXGLS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0xCB0F	The max gain setting which determine by equation MXG/6.0206. The format is 8.24 and must -20dB <MXG < 30dB .(-10dB)

9.3.4.99 DRC0 MIN Gain High Setting Register(Default Value: 0x0000_F95B)

Offset: 0x06A4			Register Name: AC_DRC0_MNGHS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0xF95B	The min gain setting which determine by equation MXG/6.0206. The format is 8.24 and must -60dB ≤MNG ≤ -40dB .(-40dB)

9.3.4.100 DRC0 MIN Gain Low Setting Register(Default Value: 0x0000_2C3F)

Offset: 0x06A8			Register Name: AC_DRC0_MNGLS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x2C3F	The min gain setting which determine by equation MNG/6.0206. The format is 8.24 and must -60dB ≤MNG ≤ -40dB .(-40dB)

9.3.4.101 DRC0 Expander Smooth Time High Coef Register(Default Value: 0x0000_0000)

Offset: 0x06AC			Register Name: AC_DRC0_EPSHC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R/W	0x000	The gain smooth filter release and attack time parameter setting in expander region, which determine by the equation that RT = 1-exp(-2.2Ts/tr). The format is 3.24. (30ms)

9.3.4.102 DRC0 Expander Smooth Time Low Coef Register(Default Value: 0x0000_640C)

Offset: 0x06B0			Register Name: AC_DRC0_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 determine by the equation that RT = 1-exp(-2.2Ts/tr). The format is 3.24. (30ms)

9.3.4.103 DRC0 HPF Gain High Coef Register(Default Value: 0x0000_0100)

Offset: 0x06B8			Register Name: AC_DRC0_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)

9.3.4.104 DRC0 HPF Gain Low Coef Register(Default Value: 0x0000_0000)

Offset: 0x06BC			Register Name: AC_DRC0_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)

9.3.4.105 DRC1 High HPF Coef Register(Default Value: 0x0000_00FF)

Offset: 0x0700			Register Name: AC_DRC1_HHPFC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R/W	0x0FF	HPF coefficient setting and the data is 3.24 format.

9.3.4.106 DRC1 Low HPF Coef Register(Default Value: 0x0000_FAC1)

Offset: 0x0704			Register Name: AC_DRC1_LHPFC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0xFAC1	HPF coefficient setting and the data is 3.24 format.

9.3.4.107 DRC1 Control Register(Default Value: 0x0000_0080)

Offset: 0x0708			Register Name: AC_DRC1_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R	0x0	DRC delay buffer data output state when DRC delay function is enable and the DRC function disable. After disable DRC function and this bit go to 0, the user should write the DRC delay function bit to 0;

			0 : not complete 1 : is complete
14:10	/	/	/
13:8	R/W	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 disable, the signal delay time is unused.
7	R/W	0x1	The delay buffer use or not when the DRC disable and the DRC buffer data output completely 0 : don't use the buffer 1 : use the buffer
6	R/W	0x0	DRC gain max limit enable 0 : disable 1 : enable
5	R/W	0x0	DRC gain min limit enable. when this function enable, it will overwrite the noise detect function. 0 : disable 1 : enable
4	R/W	0x0	Control the DRC to detect noise when ET enable 0 : disable 1 : enable
3	R/W	0x0	Signal function Select 0 : RMS filter 1 : Peak filter When Signal function Select Peak filter, the RMS parameter is unused. (AC_DRC_LRMSHAT / AC_DRC_LRMSLAT / AC_DRC_LRMSHAT / AC_DRC_LRMSLAT) When Signal function Select RMS filter, the Peak filter parameter is unused.(AC_DRC_LPFHAT / AC_DRC_LPFLAT / AC_DRC_RPFHAT / AC_DRC_RPFLAT / AC_DRC_LPFHRT / AC_DRC_LPFLRT / AC_DRC_RPFHRT / AC_DRC_RPFLRT)
2	R/W	0x0	Delay function enable 0 : disable 1 : enable When the Delay function enable is disable, the Signal delay time is unused.
1	R/W	0x0	DRC LT enable 0 : disable 1 : enable When the DRC LT is disable the LT, Kl and OPL parameter is unused.
0	R/W	0x0	DRC ET enable

			0 : disable 1 : enable When the DRC ET is disable the ET, Ke and OPE parameter is unused.

9.3.4.108 DRC1 Left Peak Filter High Attack Time Coef Register(Default Value: 0x0000_000B)

Offset: 0x070C			Register Name: AC_DRC1_LPFHAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R/W	0x00B	The left peak filter attack time parameter setting, which determine by the equation that AT = 1-exp(-2.2Ts/ta). The format is 3.24. (1ms)

9.3.4.109 DRC1 Left Peak Filter Low Attack Time Coef Register(Default Value: 0x0000_77BF)

Offset: 0x0710			Register Name: AC_DRC1_LPFLAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x77BF	The left peak filter attack time parameter setting, which determine by the equation that AT = 1-exp(-2.2Ts/ta). The format is 3.24. (1ms)

9.3.4.110 DRC1 Right Peak Filter High Attack Time Coef Register(Default Value: 0x0000_000B)

Offset: 0x0714			Register Name: AC_DRC1_RPFHAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R/W	0x00B	The left peak filter attack time parameter setting, which determine by the equation that AT = 1-exp(-2.2Ts/ta). The format is 3.24. (1ms)

9.3.4.111 DRC1 Peak Filter Low Attack Time Coef Register(Default Value: 0x0000_77BF)

Offset: 0x0718			Register Name: AC_DRC1_RPFLAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x77BF	The left peak filter attack time parameter setting, which determine by the equation that AT = 1-exp(-2.2Ts/ta). The format is 3.24. (1ms)

9.3.4.112 DRC1 Left Peak Filter High Release Time Coef Register(Default Value: 0x0000_00FF)

Offset: 0x071C			Register Name: AC_DRC1_LPFHRT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/

10:0	R/W	0x0FF	The left peak filter release time parameter setting, which determine by the equation that RT = exp(-2.2Ts/tr). The format is 3.24. (100ms)

9.3.4.113 DRC1 Left Peak Filter Low Release Time Coef Register(Default Value: 0x0000_E1F8)

Offset: 0x0720			Register Name: AC_DRC1_LPFLRT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0xE1F8	The left peak filter release time parameter setting, which determine by the equation that RT = exp(-2.2Ts/tr). The format is 3.24. (100ms)

9.3.4.114 DRC1 Right Peak filter High Release Time Coef Register(Default Value: 0x0000_00FF)

Offset: 0x0724			Register Name: AC_DRC1_RPFHRT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R/W	0x0FF	The left peak filter attack time parameter setting, which determine by the equation that RT = exp(-2.2Ts/tr). The format is 3.24. (100ms)

9.3.4.115 DRC1 Right Peak filter Low Release Time Coef Register(Default Value: 0x0000_E1F8)

Offset: 0x0728			Register Name: AC_DRC1_RPFLRT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0xE1F8	The left peak filter release time parameter setting, which determine by the equation that AT = exp(-2.2Ts/tr). The format is 3.24. (100ms)

9.3.4.116 DRC1 Left RMS Filter High Coef Register(Default Value: 0x0000_0001)

Offset: 0x072C			Register Name: AC_DRC1_LRMSHAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R/W	0x001	The left RMS filter average time parameter setting, which determine by the equation that AT = 1-exp(-2.2Ts/tav). The format is 3.24. (10ms)

9.3.4.117 DRC1 Left RMS Filter Low Coef Register(Default Value: 0x0000_2BAF)

Offset: 0x0730			Register Name: AC_DRC1_LRMSLAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x2BAF	The left RMS filter average time parameter setting, which determine by the

			equation that AT = 1-exp(-2.2Ts/tav). The format is 3.24. (10ms)

9.3.4.118 DRC1 Right RMS Filter High Coef Register(Default Value: 0x0000_0001)

Offset: 0x0734			Register Name: AC_DRC1_RRMSHAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10:0	R/W	0x001	The right RMS filter average time parameter setting, which determine by the equation that AT = 1-exp(-2.2Ts/tav). The format is 3.24. (10ms)

9.3.4.119 DRC1 Right RMS Filter Low Coef Register(Default Value: 0x0000_2BAF)

Offset: 0x0738			Register Name: AC_DRC1_RRMSLAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x2BAF	The right RMS filter average time parameter setting, which determine by the equation that AT = 1-exp(-2.2Ts/tav). The format is 3.24. (10ms)

9.3.4.120 DRC1 Compressor Threshold High Setting Register(Default Value: 0x0000_06A4)

Offset: 0x073C			Register Name: AC_DRC1_HCT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x06A4	The compressor threshold setting, which set by the equation that CTin = -CT/6.0206. The format is 8.24 (-40dB)

9.3.4.121 DRC1 Compressor Threshold Low Setting Register(Default Value: 0x0000_D3C0)

Offset: 0x0740			Register Name: AC_DRC1_LCT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0xD3C0	The compressor threshold setting, which set by the equation that CTin = -CT/6.0206. The format is 8.24 (-40dB)

9.3.4.122 DRC1 Compressor Slope High Setting Register(Default Value: 0x0000_0080)

Offset: 0x0744			Register Name: AC_DRC1_HKC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:14	/	/	/
13:0	R/W	0x0080	The slope of the compressor which determine by the equation that Kc = 1/R, there, R is the ratio of the compressor, which always is integer. The

			format is 6.24. (2 : 1)

9.3.4.123 DRC1 Compressor Slope Low Setting Register(Default Value: 0x0000_0000)

Offset: 0x0748			Register Name: AC_DRC1_LKC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x0000	The slope of the compressor which determine by the equation that Kc = 1/R, there, R is the ratio of the compressor, which always is integer. The format is 6.24. (2 : 1)

9.3.4.124 DRC1 Compressor High Output at Compressor Threshold Register(Default Value: 0x0000_F95B)

Offset: 0x074C			Register Name: AC_DRC1_HOPC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0xF95B	The output of the compressor which determine by the equation -OPC/6.0206 The format is 8.24.(-40dB)

9.3.4.125 DRC1 Compressor Low Output at Compressor Threshold Register(Default Value: 0x0000_2C3F)

Offset: 0x0750			Register Name: AC_DRC1_LOPC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x2C3F	The output of the compressor which determine by the equation OPC/6.0206 The format is 8.24. (-40dB)

9.3.4.126 DRC1 Limiter Threshold High Setting Register(Default Value: 0x0000_01A9)

Offset: 0x0754			Register Name: AC_DRC1_HLT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x01A9	The limiter threshold setting, which set by the equation that LTin = -LT/6.0206, The format is 8.24. (-10dB)

9.3.4.127 DRC1 Limiter Threshold Low Setting Register(Default Value: 0x0000_34F0)

Offset: 0x0758			Register Name: AC_DRC1_LLT
Bit	**Read/Write**	**Default/Hex**	**Description**

31:16	/	/	/
15:0	R/W	0x34F0	The limiter threshold setting, which set by the equation that LTin = -LT/6.0206, The format is 8.24. (-10dB)

9.3.4.128 DRC1 Limiter Slope High Setting Register(Default Value: 0x0000_0005)

Offset: 0x075C			Register Name: AC_DRC1_HKI
Bit	**Read/Write**	**Default/Hex**	**Description**
31:14	/	/	/
13:0	R/W	0x0005	The slope of the limiter which determine by the equation that Kl = 1/R, there, R is the ratio of the limiter, which always is integer. The format is 6.24. (50 :1)

9.3.4.129 DRC1 Limiter Slope Low Setting Register(Default Value: 0x0000_1EB8)

Offset: 0x0760			Register Name: AC_DRC1_LKI
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0x1EB8	The slope of the limiter which determine by the equation that Kl = 1/R, there, R is the ratio of the limiter, which always is integer. The format is 6.24. (50 :1)

9.3.4.130 DRC1 Limiter High Output at Limiter Threshold Register(Default Value: 0x0000_FBD8)

Offset: 0x0764			Register Name: AC_DRC1_HOPL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0xFBD8	The output of the limiter which determine by equation OPL/6.0206. The format is 8.24 .(-25dB)

9.3.4.131 DRC1 Limiter Low Output at Limiter Threshold Register(Default Value: 0x0000_FBA7)

Offset: 0x0768			Register Name: AC_DRC1_LOPL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:0	R/W	0xFBA7	The output of the limiter which determine by equation OPL/6.0206. The format is 8.24 .(-25dB)

9.3.4.132 DRC1 Expander Threshold High Setting Register(Default Value: 0x0000_0BA0)

Offset: 0x076C	Register Name: AC_DRC1_HET

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

9.3.4.133 DRC1 Expander Threshold Low Setting Register(Default Value: 0x0000_7291)

Offset: 0x0770			Register Name: AC_DRC1_LET
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x7291	The expander threshold setting, which set by the equation that ETin = -ET/6.0206, The format is 8.24. (-70dB)

9.3.4.134 DRC1 Expander Slope High Setting Register(Default Value: 0x0000_0500)

Offset: 0x0774			Register Name: AC_DRC1_HKE
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13:0	R/W	0x0500	The slope of the expander which determine by the equation that Ke = 1/R, there, R is the ratio of the expander, which always is integer and the ke must larger than 1/50. The format is 6.24. (1:5)

9.3.4.135 DRC1 Expander Slope Low Setting Register(Default Value: 0x0000_0000)

Offset: 0x0778			Register Name: AC_DRC1_LKE
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x0000	The slope of the expander which determine by the equation that Ke = 1/R, there, R is the ratio of the expander, which always is integer and the ke must larger than 1/50. The format is 6.24. (1:5)

9.3.4.136 DRC1 Expander High Output at Expander Threshold Register(Default Value: 0x0000_F45F)

Offset: 0x077C			Register Name: AC_DRC1_HOPE
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0xF45F	The output of the expander which determine by equation OPE/6.0206. The format is 8.24. (-70dB)

9.3.4.137 DRC1 Expander Low Output at Expander Threshold Register(Default Value: 0x0000_8D6E)

Offset: 0x0780			Register Name: AC_DRC1_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 .(-70dB)

9.3.4.138 DRC1 Linear Slope High Setting Register(Default Value: 0x0000_0100)

Offset: 0x0784			Register Name: AC_DRC1_HKN
Bit	Read/Write	Default/Hex	Description
31:14	/	/	/
13:0	R/W	0x0100	The slope of the linear which determine by the equation that Kn = 1/R, there, R is the ratio of the linear, which always is integer . The format is 6.24. (1:1)

9.3.4.139 DRC1 Linear Slope Low Setting Register(Default Value: 0x0000_0000)

Offset: 0x0788			Register Name: AC_DRC1_LKN
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x0000	The slope of the linear which determine by the equation that Kn = 1/R, there, R is the ratio of the linear, which always is integer . The format is 6.24. (1:1)

9.3.4.140 DRC1 Smooth filter Gain High Attack Time Coef Register(Default Value:0x0000_0002)

Offset: 0x078C			Register Name: AC_DRC1_SFHAT
Bit	Read/Write	Default/Hex	Description
31:11	/	/	/
10:0	R/W	0x002	The smooth filter attack time parameter setting, which determine by the equation that AT = 1-exp(-2.2Ts/tr). The format is 3.24. (5ms)

9.3.4.141 DRC1 Smooth filter Gain Low Attack Time Coef Register(Default Value: 0x0000_5600)

Offset: 0x0790			Register Name: AC_DRC1_SFLAT
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x5600	The smooth filter attack time parameter setting, which determine by the equation that AT = 1-exp(-2.2Ts/tr). The format is 3.24. (5ms)

9.3.4.142 DRC1 Smooth filter Gain High Release Time Coef Register(Default Value:0x0000_0000)

Offset: 0x0794			Register Name: AC_DRC1_SFHRT
Bit	Read/Write	Default/Hex	Description
31:11	/	/	/
10:0	R/W	0x000	The gain smooth filter release time parameter setting, which determine by the equation that RT = 1-exp(-2.2Ts/tr). The format is 3.24. (200ms)

9.3.4.143 DRC1 Smooth filter Gain Low Release Time Coef Register(Default Value: 0x0000_0F04)

Offset: 0x0798			Register Name: AC_DRC1_SFLRT
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x0F04	The gain smooth filter release time parameter setting, which determine by the equation that RT = 1-exp(-2.2Ts/tr). The format is 3.24. (200ms)

9.3.4.144 DRC1 MAX Gain High Setting Register(Default Value: 0x0000_FE56)

Offset: 0x079C			Register Name: AC_DRC1_MXGHS
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0xFE56	The max gain setting which determine by equation MXG/6.0206. The format is 8.24 and must -20dB <MXG< 30dB .(-10dB)

9.3.4.145 DRC1 MAX Gain Low Setting Register(Default Value: 0x0000_CB0F)

Offset: 0x07A0			Register Name: AC_DRC1_MXGLS
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0xCB0F	The max gain setting which determine by equation MXG/6.0206. The format is 8.24 and must -20dB <MXG < 30dB .(-10dB)

9.3.4.146 DRC1 MIN Gain High Setting Register(Default Value: 0x0000_F95B)

Offset: 0x07A4			Register Name: AC_DRC1_MNGHS
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0xF95B	The min gain setting which determine by equation MXG/6.0206. The format is 8.24 and must -60dB ≤MNG ≤ -40dB .(-40dB)

9.3.4.147 DRC1 MIN Gain Low Setting Register(Default Value: 0x0000_2C3F)

Offset: 0x07A8			Register Name: AC_DRC1_MNGLS
Bit	Read/Write	Default/Hex	Description
31:16	/	/	/
15:0	R/W	0x2C3F	The min gain setting which determine by equation MNG/6.0206. The format is 8.24 and must -60dB ≤MNG ≤ -40dB .(-40dB)

9.3.4.148 DRC1 Expander Smooth Time High Coef Register(Default Value: 0x0000_0000)

Offset: 0x07AC			Register Name: AC_DRC1_EPSHC
Bit	Read/Write	Default/Hex	Description
31:11	/	/	/
10:0	R/W	0x000	The gain smooth filter release and attack time parameter setting in expander region, which determine by the equation that RT = 1-exp(-2.2Ts/tr). The format is 3.24. (30ms)

9.3.4.149 DRC1 Expander Smooth Time Low Coef Register(Default Value: 0x0000_640C)

Offset: 0x07B0			Register Name: AC_DRC1_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 determine by the equation that RT = 1-exp(-2.2Ts/tr). The format is 3.24. (30ms)

9.3.4.150 DRC1 HPF Gain High Coef Register(Default Value: 0x0000_0100)

Offset: 0x07B8			Register Name: AC_DRC1_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)

9.3.4.151 DRC1 HPF Gain Low Coef Register(Default Value: 0x0000_0000)

Offset: 0x07BC			Register Name: AC_DRC1_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)

9.3.4.152 AC Parameter Configuration Register(Default Value: 0x1000_0000)

Address: 0x050967C0			Register Name: AC_PR_CFG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	/	/	/
28	R/W	0x1	AC_PR_RST AC_PR Reset 0: Assert 1: De-assert
27:25	/	/	/
24	R/W	0x0	AC_PR_RW AC_PR Read Or Write 0: read 1: write
23:22	/	/	/
21:16	R/W	0x0	AC_PR _ADDR AC_PR Address [5:0]
15:8	R/W	0x0	ADDA_PR _WDAT ADDA_PR Write Data [7:0]
7:0	R/W	0x0	ADDA_PR _RDAT ADDA_PR Read Data [7:0]

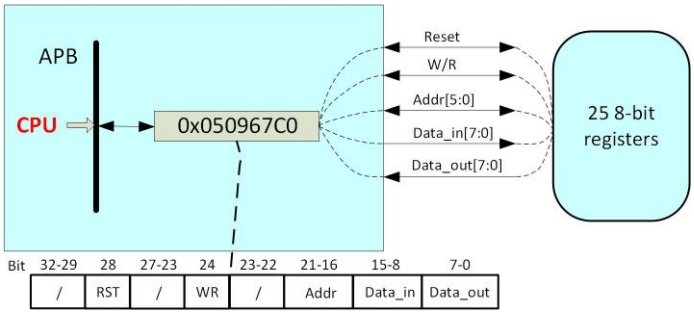
The Analog domain register can be write/Read though the AC_PR Configuration Register(AC_PR_CFG_REG) which is in the PRCM Spec. Thought this register to configure the codec analog domain circuit.

ADDR[5:0] ： AC_PR Address；

W/R：Write/Read Enable；

WDAT[7:0]：Write Data；

RDAT[7:0]： Read Data;



9.3.4.153 Headphone Amplifier Control Register 0（Default Value: 0x00）

Offset: 0x00			Register Name: HP_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
7	R/W	0x0	PA_CLK_GATE PA clock gating control when system VDD is off and Audio analog channel is working, this bit must be set to 1, because the PA clock come from system VDD domain. When this bit is 1, the Zero cross over function will be disabled automatically.

			0: not gating 1: gating
6	R/W	0x0	HPPA_EN Right & Left Headphone PA Enable 0: Disable 1: Enable
5:0	R/W	0x0	HPVOL Headphone Volume Control, (HPVOL): Total 64 level, from 0dB to -62dB, 1dB/step, mute when 000000.

9.3.4.154 Left Output Mixer Control Register（Default Value: 0x00）

Offset: 0x01			Register Name: LO_MIX_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
7	/	/	/
6:0	R/W	0x0	LMIXMUTE Left Output Mixer Mute Control 0: Mute, 1: Not mute Bit 6: / Bit 5: MIC2 Boost stage Bit 4: / Bit 3: / Bit 2: / Bit 1: Left channel DAC Bit 0: Right channel DAC

9.3.4.155 Right Output Mixer Control Register(Default Value: 0x00)

Offset: 0x02			Register Name: RO_MIX_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
7	/	/	/
6:0	R/W	0x0	RMIXMUTE Right Output Mixer Mute Control 0: Mute 1: Not mute Bit 6: / Bit 5: MIC2 Boost stage Bit 4: / Bit 3: / Bit 2: LINEINR Bit 1: Right channel DAC Bit 0: Left channel DAC

9.3.4.156 MIC2 Control Register(Default Value: 0xB4)

Offset: 0x08			Register Name: MIC2_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
7	R/W	0x1	MIC2 Source Select 0: MICIN3 1: MICIN2
6:4	R/W	0x3	MIC2G MIC2 BOOST stage to L or R output mixer Gain Control From -4.5dB to 6dB, 1.5dB/step, default is 0dB
3	R/W	0x0	MIC2AMPEN MIC2 Boost AMP Enable 0: Disable 1: Enable
2:0	R/W	0x4	MIC2BOOST MIC2 Boost AMP Gain Control 0dB when 000, 24dB to 42dB when 001 to 111, 3dB/step, default is 33dB

9.3.4.157 Linein Control Register(Default Value: 0x03)

Offset: 0x09			Register Name: LINEIN_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
7:3	/	/	/
2:0	R/W	0x3	LINEING LINEINR to R output mixer Gain Control From -4.5dB to 6dB, 1.5dB/step, default is 0dB

9.3.4.158 Mixer and DAC Control Register(Default Value: 0x00)

Offset: 0x0A			Register Name: MIX_DAC_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
7	R/W	0x0	DACAREN Internal Analog Right channel DAC Enable 0: Disable 1: Enable
6	R/W	0x0	DACALEN Internal Analog Left channel DAC Enable 0: Disable 1: Enable
5	R/W	0x0	RMIXEN Right Analog Output Mixer Enable 0: Disable 1: Enable
4	R/W	0x0	LMIXEN

			Left Analog Output Mixer Enable 0: Disable 1: Enable
3	R/W	0x0	RHPPAMUTE All input source to Right Headphone PA mute, including Right Output mixer and Internal Right channel DAC: 0: Mute 1: Not mute
2	R/W	0x0	LHPPAMUTE All input source to Left Headphone PA mute, including Left Output mixer and Internal Left channel DAC: 0: Mute 1: Not mute
1	R/W	0x0	RHPIS Right Headphone Power Amplifier (PA) Input Source Select 0: Right channel DAC 1: Right Analog Mixer
0	R/W	0x0	LHPIS Left Headphone Power Amplifier (PA) Input Source Select 0: Left channel DAC 1: Left Analog Mixer

9.3.4.159 Right ADC Mixer Control Register(Default Value: 0x00)

Offset: 0x0C			Register Name: R_ADCMIX_SRC
Bit	**Read/Write**	**Default/Hex**	**Description**
7	/	/	/
6:0	R/W	0x0	RADCMIXMUTE Right ADC Mixer Mute Control: 0: Mute 1: On Bit 6: / Bit 5: MIC2 Boost stage Bit 4: / Bit 3: / Bit 2: LINEINR Bit 1: Right output mixer Bit 0: Left output mixer

9.3.4.160 ADC Control Register (Default Value: 0x03)

Offset: 0x0D			Register Name: ADC_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
7	R/W	0x0	ADCREN

			ADC Right Channel Enable 0: Disable 1: Enable
6	R/W	0x0	ADCLEN ADC Left Channel Enable 0: Disable 1: Enable
5	R/W	0x0	Dither Select 0: New dither off 1: New dither on
4	R/W	0x0	Det Mode 0: Jack in pull low 1: Jack in pull high
3	/	/	/
2:0	R/W	0x3	ADCG ADC Input Gain Control From -4.5dB to 6dB, 1.5dB/step default is 0dB

9.3.4.161 Microphone Bias Control Register(Default Value: 0x21)

Offset: 0x0E			Register Name: MBIAS_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
7	R/W	0x0	MMICBIASEN Master Microphone Bias enable 0: Disable 1: Enable
6:5	R/W	0x1	MBIASSEL MMICBIAS voltage level select 00: 1.88V 01: 2.09V 10: 2.33V 11: 2.50V
4:2	/	/	/
1:0	R/W	0x1	HBIASSEL HMICBIAS voltage level select 00: 1.88V 01: 2.09V 10: 2.33V 11: 2.50V

9.3.4.162 Analog Performance Tuning Register(Default Value: 0xD6)

Offset: 0x0F			Register Name: APT_REG
Bit	**Read/Write**	**Default/Hex**	**Description**

7	R/W	0x1	MMIC BIAS chopper enable 0: Disable 1: Enable
6:5	R/W	0x2	MMIC BIAS chopper clock select 00: 250 kHz 01: 500 kHz 10: 1 MHz 11: 2 MHz
4	R/W	0x1	DITHER ADC dither on/off control 0: dither off 1: dither on
3:2	R/W	0x1	DITHER_CLK_SELECT ADC dither clock select 00: ADC FS * (8/9), about 43 kHz when FS=48 kHz 01: ADC FS * (16/15), about 51 kHz when FS=48 kHz 10: ADC FS * (4/3), about 64 kHz when FS=48 kHz 11: ADC FS * (16/9), about 85 kHz when FS=48 kHz
1:0	R/W	0x2	BIHE_CTRL, BIHE control 00: no BIHE 01: BIHE=7.5 HOSC 10: BIHE=11.5 HOSC 11: BIHE=15.5 HOSC

9.3.4.163 OP BIAS Control Register0 (Default Value: 0x55)

Offset: 0x10			**Register Name: OP_BIAS_CTRL0**
Bit	**Read/Write**	**Default/Hex**	**Description**
7:6	R/W	0x1	OPDRV_OPEAR_CUR. OPDRV/OPEAR output stage current setting
5:4	R/W	0x1	OPADC1_BIAS_CUR. OPADC1 Bias Current Select
3:2	R/W	0x1	OPADC2_BIAS_CUR. OPADC2 Bias Current Select
1:0	R/W	0x1	OPAAF_BIAS_CUR. OPAAF in ADC Bias Current Select

9.3.4.164 OP BIAS Control Register1 (Default Value: 0x55)

Offset: 0x11			**Register Name: OP_BIAS_CTRL1**
Bit	**Read/Write**	**Default/Hex**	**Description**
7:6	R/W	0x1	OPMIC_BIAS_CUR OPMIC Bias Current Control
5:4	R/W	0x1	OPVR_BIAS_CUR.

			OPVR Bias Current Control
3:2	R/W	0x1	OPDAC_BIAS_CUR. OPDAC Bias Current Control
1:0	R/W	0x1	OPMIX_BIAS_CUR. OPMIX/OPLPF/OPDRV/OPEAR Bias Current Control

9.3.4.165 ZERO Cross &USB Bias Control Register(Default Value: 0x02)

Offset: 0x12			Register Name: ZC_VOL_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
7	R/W	0x0	Function enable for master volume change at zero cross over 0: Disable 1: Enable
6	R/W	0x0	Timeout control for master volume change at zero cross over 0: 32ms 1: 64ms
5:3	/	/	/
2:0	R/W	0x2	USB_BIAS_CUR. USB bias current tuning From 23uA to 30uA, Default is 25uA

9.3.4.166 Bias Calibration Data Register

Offset: 0x13			Register Name: BIAS_CAL_DATA
Bit	**Read/Write**	**Default/Hex**	**Description**
7:6	/	/	/
5:0	R	UDF	BIASCALI Bias Calibration Data, 6bit

9.3.4.167 Bias Calibration Set Data Register (Default Value: 0x20)

Offset: 0x14			Register Name: BIAS_CAL_SET
Bit	**Read/Write**	**Default/Hex**	**Description**
7:6	R/W	0x0	SELDETADCDY Select the delay time to pull low the micdet when jack removal 00: 0.5ms 01: 1ms 10: 1.5ms 11: 2ms
5:0	R/W	0x20	BIASVERIFY Bias Register Setting Data

9.3.4.168 Bias Calibration Control Register(Default Value: 0x00)

Offset: 0x15			Register Name: BIAS_CAL_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
7	R/W	0x0	PA_SPEED_SELECT PA setup speed control (for testing) 0: slow 1: fast
6	R/W	0x0	CURRENT_TEST_SELECT Internal current sink test enable (from LINEIN pin) 0:Normal 1: for Debug
5:3	R/W	0x0	PA_POWER_CTRL PA Power Stage Driving Ability Control 000: 1/4 power 001: 1/2 power 011: 3/4 power 111: 1 power **NOTE** **When the field is 1/4 power, the power consumption of PA can decrease,but THD+N lack some dB.**
2	R/W	0x0	BIAS calibration mode select 0: average 1: single
1	R/W1C	0x0	BIAS and HP calibration control Write 1 to this bit, the calibration will be doing again. Then this bit will be reset to zero automatically
0	R/W	0x0	BIASCALIVERIFY Bias Calibration Verify 0: Calibration 1: Register setting

9.3.4.169 Headphone PA Control Register(Default Value: 0xF1)

Offset: 0x16			Register Name: HP_PA_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
7	R/W	0x1	BUFFERENABLE When this bit is written to 0, the buffer in headphone disabled
6	R/W	0x1	HPMUTENABLE When this bit is written to 0, all input to headphone is mute
5	R/W	0x1	HPINPUTENABLE When this bit is written to 0, the input stage of headphone disabled
4	R/W	0x1	HPOUTPUTENABLE When this bit is written to 0, the output stage of headphone disabled

3:2	R/W	0x0	HPPA_DEL Headphone delay time when start up 00: 4ms 01: 8ms 10: 16ms 11: 32ms
1:0	R/W	0x1	CP_CLKS Charge Pump Clock select 00: 250k 01: 330k 10: 400k 11: 500k

9.3.4.170 Headphone Calibration Control(Default Value: 0x04)

Offset: 0x17			Register Name: HP_CAL_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
7	R/W	0x0	HPCALIFIRST When this bit is written to 1 , HEADPHONE calibrates once before enabled.
6	/	/	/
5	R/W	0x0	HPCALIMODE HEADPHONE calibration equilibration MODE select 0: equilibration mode 1: no equilibration
4	R/W	0x0	HPCALIVERIFY HEADPHONE calibration in verify mode enable 0: Disable 1: Enable
3	R/W	0x0	/
2:0	R/W	0x4	HPCALICKS HEADPHONE Calibration clock frequency select 000: 4 … 100: 64 … 111: 512

9.3.4.171 Right Headphone Calibration DAT Register

Offset: 0x18			Register Name: RHP_CAL_DAT
Bit	**Read/Write**	**Default/Hex**	**Description**
7:0	R	UDF	HPRCALI Right Headphone calibration Data

9.3.4.172 Right Headphone Calibration Setting Register(Default Value: 0x80)

Offset: 0x19			Register Name: RHP_CAL_SET
Bit	**Read/Write**	**Default/Hex**	**Description**
7:0	R/W	0x80	HPRCALIVERIFY Right Headphone calibration Setting Data

9.3.4.173 Left Headphone Calibration Data Register

Offset: 0x1A			Register Name: LHP_CAL_DAT
Bit	**Read/Write**	**Default/Hex**	**Description**
7:0	R	UDF	HPLCALI Left Headphone Calibration Data

9.3.4.174 Left Headphone Calibration Setting Register(Default Value: 0x80)

Offset: 0x1B			Register Name: LHP_CAL_SET
Bit	**Read/Write**	**Default/Hex**	**Description**
7:0	R/W	0x80	HPLCALIVERIFY Left Headphone Calibration Setting Data

9.3.4.175 Mic Detect Control Register(Default Value: 0x40)

Offset: 0x1C			Register Name: MDET_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
7	/	/	/
6:4	R/W	0x4	SELDETADCFS Select sample interval of the ADC sample 000: 2ms … 100: 32ms … 111: 256ms
3:2	R/W	0x0	SELDETADCDB Select debounce time when jack removal 00: 128ms 01: 256ms 10: 512ms 11: 1024ms
1:0	R/W	0x0	SELDETADCBF Select the time to enable HBIAS before micadc work 00: 2ms 01: 4ms

			10: 8ms 11: 16ms

9.3.4.176 Jack & Mic Detect Control Register(Default Value: 0x00)

Offset: 0x1D			Register Name: JACK_MIC_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
7	R/W	0x0	JACKDETEN Jack detect enable 0: Disable 1: Enable
6	/	/	/
5	R/W	0x0	HMICBIASEN Handset Microphone Bias enable 0: Disable 1: Enable
4	R/W	0x0	MICADCEN Microphone detect ADC enable 0: Disable 1: Enable
3	R/W	0x0	POPFREE When this bit is 0, HBIAS MICADC is controlled by registor
2	/	/	/
1	R/W	0x0	AUTOPLEN Enable the function to auto pull low micdet when jack removal 0: Disable, 1: Enable
0	R/W	0x0	MICDETPL When this bit is 1and AUTOPLEN is 0, the micdet is pull to gnd

9.3.4.177 Charge Pump LDO Output Control Register(Default Value: 0x00)

Offset: 0x21			Register Name: CP_LDO_CTR
Bit	**Read/Write**	**Default/Hex**	**Description**
7	R/W	0x0	CPLDO Enable 0: Enabled by charge pump 1: Disable
6:5	R/W	0x0	Charge Pump LDO Output Voltage Control 00: 0.9V 01: 1.0V 10: 1.1V 11: 1.2V
4:3	R/W	0x0	CPLDO Bias Current Control 00: Min 11: Max

2:0	/	/	/

Figures

Tables

10 Interfaces

10.1 TWI

10.1.1 Overview

This TWI Controller is designed as an interface between CPU host and the serial TWI bus. It can support all the standard TWI transfer, including slave and master. The communication of the 2-wire bus is carried out by a byte-wise mode based on interrupt or polled handshaking. This TWI controller can be operated in standard mode (100 kbit/s) or fast-mode, supporting data rate up to 400 kbit/s. The 10-bit addressing mode is supported for this specified application. General call addressing is also supported in slave mode

Features:

- Software-programmable for slave or master
- Supports repeated START signal
- Allows 10-bit addressing with TWI bus
- Performs arbitration and clock synchronization
- Own address and general call address detection
- Interrupt on address detection
- Supports speed up to 400 kbit/s (‘fast mode’)
- Allows operation from a wide range of input clock frequency

10.1.2 Block Diagram

Figure 10-1 shows the block diagram of TWI.

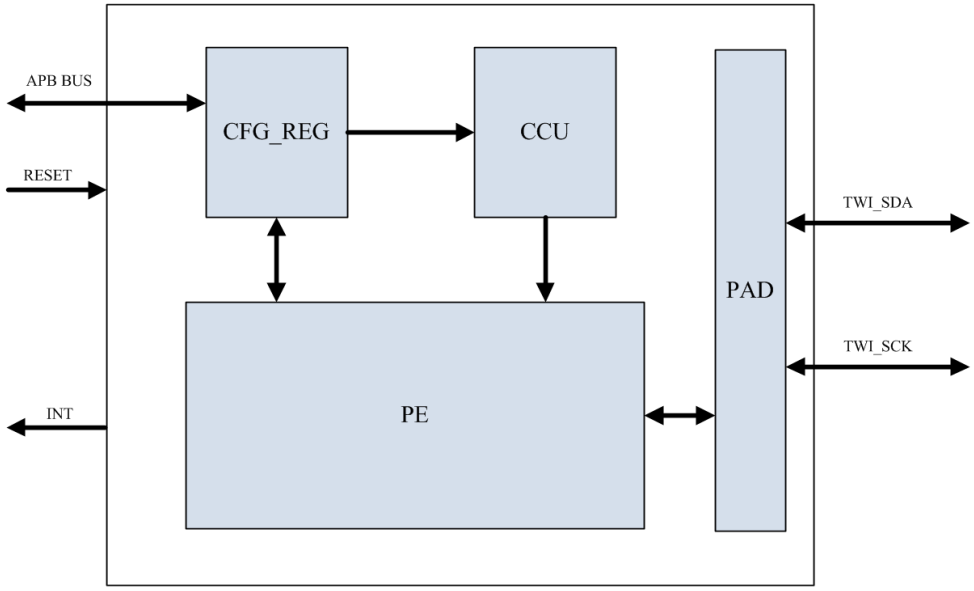


Figure10- 1. TWI Block Diagram

RESET: Module reset signal

INT: Module output interrupt signal

CFG_REG: Module configuration register in TWI

PE: Packet encoding/decoding

CCU: Module clock controller unit

10.1.3 Operations and Functional Descriptions

10.1.3.1 External Signals

The TWI controller has 4 TWIs.Table 10-1 describes the external signals of TWI. TWI_SCK and TWI_SDA are bidirectional I/O,When TWI is configured as Master device,TWI_SCK is output pin;when TWI is configurable as Slave device,TWI_SCK is input pin.other TWI ports are used as General Purpose I/O ports.For information about General Purpose I/O ports,see **Port Controller** in chapter10.

Table10- 1. TWI External Signals

Signal (x=[2:0])	Description	Type
TWIx_SCK	TWI Clock Signal for CPUX	I/O
TWIx_SDA	TWI Serial Data for CPUX	I/O
R_TWI_SCK	TWI Serial Clock Signal for CPUS	I/O
R_TWI_SDA	TWI Serial Data Signal for CPUS	I/O

10.1.3.2 Clock Sources

Each TWI controller has a fixed clock source.APB2 is the clock source of TWI in CPUX and APBS is the clock source of R-TWI in CPUS.The APB Bus get some clock sources.Users can select one of them to be used as APB clock.Table 10-2 describes the clock sources for TWI.Users can see **Clock Controller Unit(CCU)** in chapter3 for clock setting,configuration and gating information.

Table10- 2. TWI Clock Sources

Clock Sources	Description
APBS Bus	TWI in CPUS,for details on APBS refer to CCU
APB2 Bus	TWI in CPUX,for details on APB2 refer to CCU

After select a proper clock, for using the TWI in CPUX,user must open the gating of TWI and release the reset bit. For using the TWI in CPUS,user also need to open the gating of R-TWI and release the reset bit . For more details on the gating/reset operations ,please refer to the **CCU.**

10.1.3.3 Timing Diagram

Data transferred are always in a unit of 8-bit (1 byte), followed by an acknowledge bit. The number of bytes that can be transmitted is unrestricted. Data is transferred in serial with the MSB first. Between each byte of data transfer, a receiver device will hold the clock line SCK low to force the transmitter into a wait state while waiting the response from microprocessor.

The clock line is driven by the master all the time, including the acknowledge-related clock cycle, except for the SCK holding between each byte. After sending each byte, the transmitter releases the SDA line to allow the receiver to pull down the SDA line and send an acknowledge signal (or pull it high to send a "not acknowledge") to the transmitter.

When a slave receiver does not acknowledge the slave address (unable to receive because of no resource available), the data line must be pulled high by the slave so that the master can generate a STOP condition to stop the transfer. When the acknowledge signal is pulled high, slave receiver no longer sends more data. And the master should generate the STOP condition to stop the transfer.

Figure 10-2 provides an illustration the relation of SDA signal line and SCK signal line on the TWI serial bus.

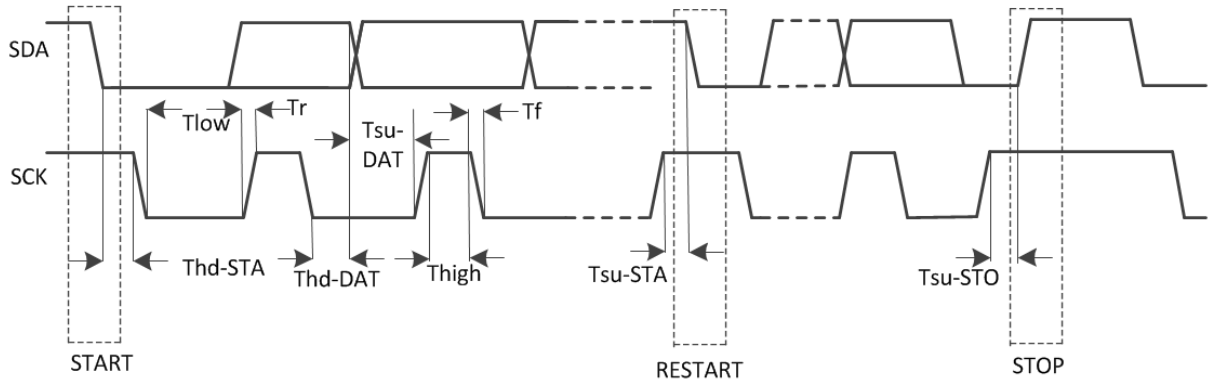


Figure10- 2. TWI Timing Diagram

The timing parameters of TWI timing shows in Table 10-3.

Table10- 3. TWI Timing Constants

Parameter	Symbol	Standard mode		Fast mode		Unit
		Min	Max	Min	Max	
SCK clock frequency	Fsck	0	100	0	400	kHz
Setup time in Start	Tsu-STA	4.7	-	0.6	-	us
Hold time in Start	Thd-STA	4.0	-	0.6	-	us
Setup time in Data	Tsu-DAT	250	-	100	-	ns
Hold time in Data	Thd-DAT	5.0	-	-	-	ns
Setup time in Stop	Tsu-STO	4.0	-	0.6	-	us
SCK low level time	Tlow	4.7	-	1.3	-	us

SCK high level time	Thigh	4.0	-	0.6	-	us
SCK/SDA falling time	Tf	-	300	20	300	ns
SCK/SDA rising time	Tr	-	1000	20	300	ns

10.1.3.4 TWI Controller Operation

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

When the CPU host wants to start a bus transfer, it initiates a bus START to enter the master mode by setting IM_STA bit of the 2WIRE_CNTR register to high (before it must be low). The TWI will assert INT line and INT_FLAG to indicate a completion for the START condition and each consequent byte transfer. At each interrupt, the micro-processor needs to check the 2WIRE_STAT register for current status. A transfer has to be concluded with STOP condition by setting M_STP bit to high.

In Slave Mode, the TWI also constantly samples the bus and look for its own slave address during addressing cycles. Once a match is found, it is addressed and interrupt the CPU host with the corresponding status. Upon request, the CPU host should read the status, read/write 2WIRE_DATA data register, and set the 2WIRE_CNTR control register. After each byte transfer, a slave device always halt the operation of remote master by holding the next low pulse on SCL line until the microprocessor responds to the status of previous byte transfer or START condition.

10.1.4 Programming Guidelines

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

Figure 10-3 shows a software operation flow of TWI Initialization.

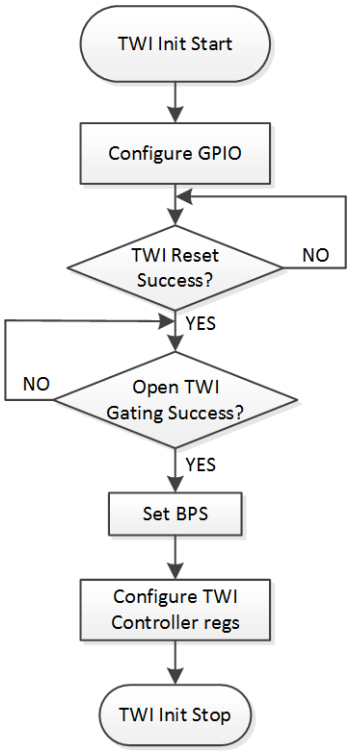


Figure10- 3. TWI Initial Flow

Figure 10-4 shows a software operation flow of TWI write to device.

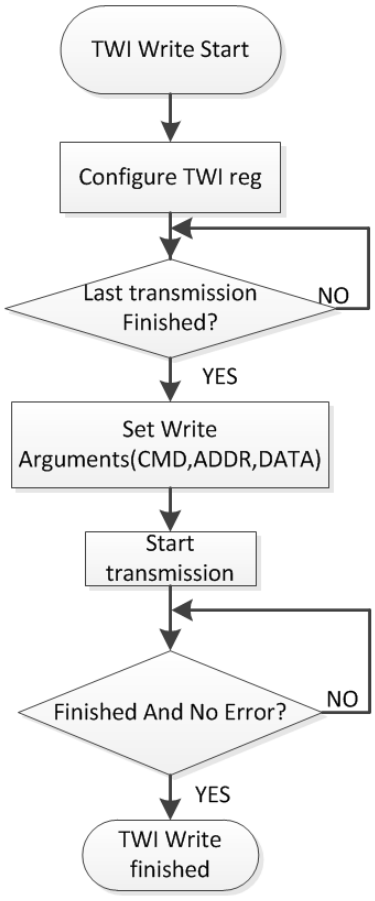


Figure10- 4. TWI Write Flow

Figure 10-5 shows a software operation flow of TWI read from device.

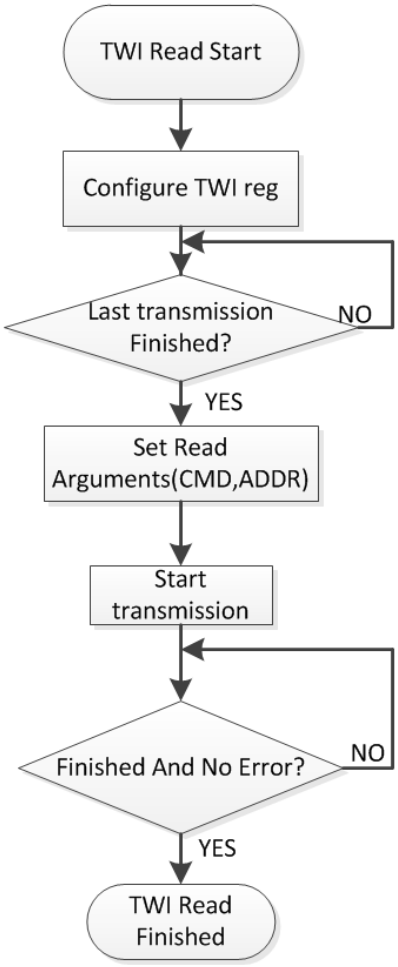


Figure10- 5. TWI Read Flow

10.1.5 Register List

Module Name	Base Address
TWI0	0x0500 2000
TWI1	0x0500 2400
TWI2	0x0500 2800
R_TWI	0x0708 1400

Register Name	Offset	Description
TWI_ADDR	0x0000	TWI Slave Address
TWI_XADDR	0x0004	TWI Extended Slave Address
TWI_DATA	0x0008	TWI Data Byte
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
TWI_EFR	0x001C	TWI Enhance Feature Register
TWI_LCR	0x0020	TWI Line Control Register

10.1.6 Register Description

10.1.6.1 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 7-bit addressing SLA6, SLA5, SLA4, SLA3, SLA2, SLA1, SLA0 10-bit addressing 1, 1, 1, 1, 0, SLAX[9:8]
0	R/W	0x0	GCE General Call Address Enable 0: Disable 1: Enable

NOTE

For 7-bit addressing:

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

For 10-bit addressing:

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

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

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

10.1.6.3 TWI Data Register(Default Value:0x0000_0000)

Offset: 0x0008			Register Name: TWI_DATA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/

7:0	R/W	0x0	TWI_DATA Data byte transmitted or received

10.1.6.4 TWI Control Register(Default Value:0x0000_0000)

Offset: 0x000C			Register Name: TWI_CNTR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R/W	0x0	INT_EN Interrupt Enable 0: The interrupt line always low 1: The interrupt line will go high when INT_FLAG is set.
6	R/W	0x0	BUS_EN TWI Bus Enable 0: The TWI bus ISDA/ISCL is ignored and the TWI Controller will not respond to any address on the bus 1: The TWI will respond to calls 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 M_STA is set to '1', 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 being 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 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 behave as 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 INT_FLAG is automatically set to '1' when any of 28 (out of the possible 29) states is entered (see 'STAT Register' below). The only state that does not set INT_FLAG is state F8h. If the INT_EN bit is set, the interrupt line goes high when IFLG is set to '1'. If the TWI is operating in slave mode, data transfer is suspended when INT_FLAG is set and the low period of the TWI bus clock line (SCL) is stretched until '1' is written to INT_FLAG. The TWI clock line is then released and the interrupt line goes low.
2	R/W	0x0	A_ACK Assert Acknowledge When A_ACK is set to '1', an Acknowledge (low level on SDA) will be sent during the acknowledge clock pulse on the TWI bus if: 1. Either the whole of a matching 7-bit slave address or the first or the second byte of a matching 10-bit slave address has been received. 2. The general call address has been received and the GCE bit in the ADDR register is set to '1'. 3. A data byte has been received in master or slave mode. When A_ACK is '0', a Not Acknowledge (high level on SDA) will be sent when a data byte is received in master or slave mode. If A_ACK is cleared to '0' in slave transmitter mode, the byte in the DATA register is assumed to be the 'last byte'. After this byte has been transmitted, the TWI will enter state C8h then return to the idle state (status code F8h) when INT_FLAG is cleared. The TWI will not respond as a slave unless A_ACK is set.
1:0	/	/	/

10.1.6.5 TWI Status Register(Default Value:0x0000_00F8)

Offset: 0x0010			**Register Name: TWI_STAT**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R	0xF8	STA Status Information Byte **Code Status** 0x00: Bus error 0x08: START condition transmitted 0x10: Repeated START condition transmitted 0x18: Address + Write bit transmitted, ACK received 0x20: Address + Write bit transmitted, ACK not received 0x28: Data byte transmitted in master mode, ACK received 0x30: Data byte transmitted in master mode, ACK not received

			0x38: Arbitration lost in address or data byte 0x40: Address + Read bit transmitted, ACK received 0x48: Address + Read bit transmitted, ACK not received 0x50: Data byte received in master mode, ACK transmitted 0x58: Data byte received in master mode, not ACK transmitted 0x60: Slave address + Write bit received, ACK transmitted 0x68: Arbitration lost in address as master, slave address + Write bit received, ACK transmitted 0x70: General Call address received, ACK transmitted 0x78: Arbitration lost in 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 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: 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

10.1.6.6 TWI Clock Register(Default Value:0x0000_0000)

Offset: 0x0014			**Register Name: TWI_CCR**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:7	/	/	/
6:3	R/W	0x0	CLK_M
2:0	R/W	0x0	CLK_N The TWI bus is sampled by the TWI at the frequency defined by F0: Fsamp = F 0 = Fin / 2^CLK_N The TWI OSCL output frequency, in master mode, is F1 / 10: F1 = F0 / (CLK_M + 1) Foscl = F1 / 10 = Fin / (2^CLK_N * (CLK_M + 1)*10) For Example： Fin = 48MHz (APB clock input) For 400 kHz full speed 2Wire, CLK_N = 2, CLK_M=2 F0 = 48MHz/2^2=12MHz, F1= F0/(10*(2+1)) = 0.4MHz

			For 100 kHz standard speed 2Wire, CLK_N=2, CLK_M=11 F0=48MHz/2^2=12MHz, F1=F0/(10*(11+1)) = 0.1MHz

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

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

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

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

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

Offset: 0x0020			Register Name: TWI_LCR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:6	/	/	/
5	R	0x1	SCL_STATE Current State of TWI_SCL 0 : Low 1 : High
4	R	0x1	SDA_STATE Current State of TWI_SDA 0 : Low 1 : High
3	R/W	0x1	SCL_CTL TWI_SCL Line State Control Bit When line control mode is enabled (bit[2] set), value of this bit decide 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 control 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 line control mode is enabled (bit[0] set), value of this bit decide 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 control by the value of bit[1]. 0 : Disable TWI_SDA line control mode 1 : Enable TWI_SDA line control mode

10.2 UART

10.2.1 Overview

The UART is used for serial communication with a peripheral, modem (data carrier equipment, DCE) or data set. Data is written from a master (CPU) over the APB bus to the UART and it is converted to serial form and transmitted to the destination device. Serial data is also received by the UART and stored for the master (CPU) to read back.

The UART contains registers to control the character length, baud rate, parity generation/checking, and interrupt generation. Although there is only one interrupt output signal from the UART, there are several prioritized interrupt types that can be responsible for its assertion. Each of the interrupt types can be separately enabled/ disabled with the control registers.

The UART has 16450 and 16550 modes of operation, which are compatible with a range of standard software drivers. In 16550 mode, transmit and receive operations are both buffered by FIFOs. In 16450 mode, these FIFOs are disabled.

The UART supports word lengths from five to eight bits, an optional parity bit and 1, 1 ½ or 2 stop bits, and is fully programmable by an AMBA APB CPU interface. A 16-bit programmable baud rate generator and an 8-bit scratch register are included, together with separate transmit and receive FIFOs. Eight modem control lines and a diagnostic loop-back mode are provided.

Interrupts can be generated for a range of TX Buffer/FIFO, RX Buffer/FIFO, Modem Status and Line Status conditions.

For integration in systems where Infrared SIR serial data format is required, the UART can be configured to have a software-programmable IrDA SIR Mode. If this mode is not selected, only the UART (RS232 standard) serial data format is available.

Features:

- Compatible with industry-standard 16550 UARTs
- 256 Bytes Transmit and Receive data FIFOs
- Capable of speed up to 5 Mbit/s
- Supports 5 to 8 data bits and 1/1.5/2 stop bits
- Supports Even, Odd or No Parity
- Supports DMA controller interface
- Supports Software/ Hardware Flow Control
- Supports IrDA 1.0 SIR
- Supports RS-485/9-bit mode

10.2.2 Block Diagram

Figure 10-6 shows a block diagram of the UART.

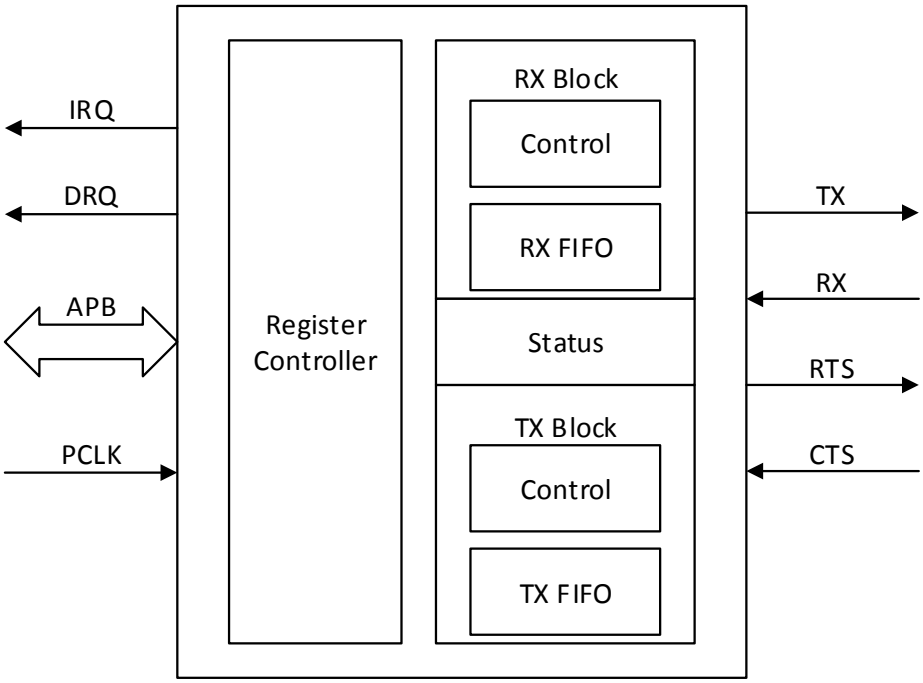


Figure10- 6. UART Block Diagram

10.2.3 Operations and Functional Descriptions

10.2.3.1 External Signals

Table 10-4 describes the external signals of UART.

Table10- 4. UART External Signals

Signal	Type	Description
UART0_TX	O	Serial Data Output
UART0_RX	I	Serial Data Input
UART1_TX	O	Serial Data Output
UART1_RX	I	Serial Data Input
UART1_CTS	I	Clear to Send
UART1_RTS	O	Request to Send
UART2_TX	O	Serial Data Output
UART2_RX	I	Serial Data Input
UART2_CTS	I	Clear to Send
UART2_RTS	O	Request to Send
UART3_TX	O	Serial Data Output
UART3_RX	I	Serial Data Input
UART3_CTS	I	Clear to Send
UART3_RTS	O	Request to Send
UART4_TX	O	Serial Data Output
UART4_RX	I	Serial Data Input
UART4_CTS	I	Clear to Send
UART4_RTS	O	Request to Send

R_UART_TX	O	Serial Data Output
R_UART_RX	I	Serial Data Input

10.2.3.2 Clock Sources

Table 10-5 describes the clock sources of UART.

Table10- 5. UART Clock Source

Clock Sources	Description
APB2_CLK	Clock of APB2

10.2.3.3 Typical Application

Figure 10-7 shows the application block diagram of UART.

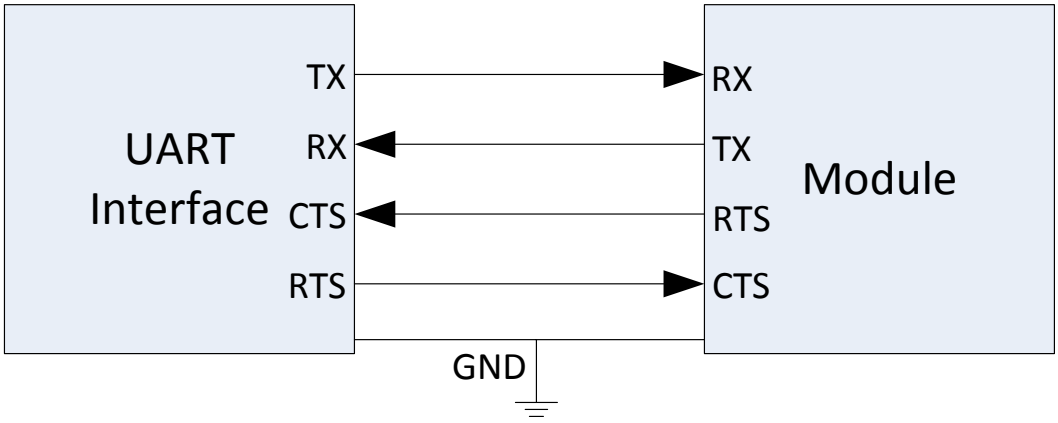


Figure10- 7. UART Application Diagram

10.2.3.4 UART Timing Diagram

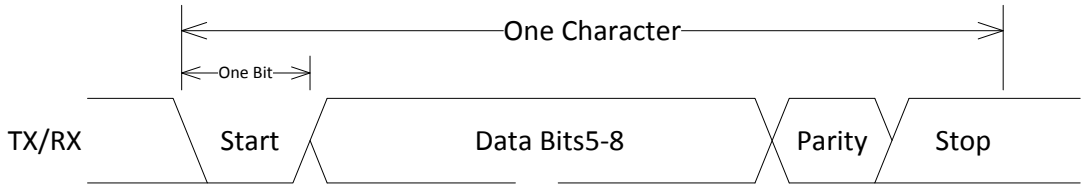


Figure10- 8. UART Serial Data Format

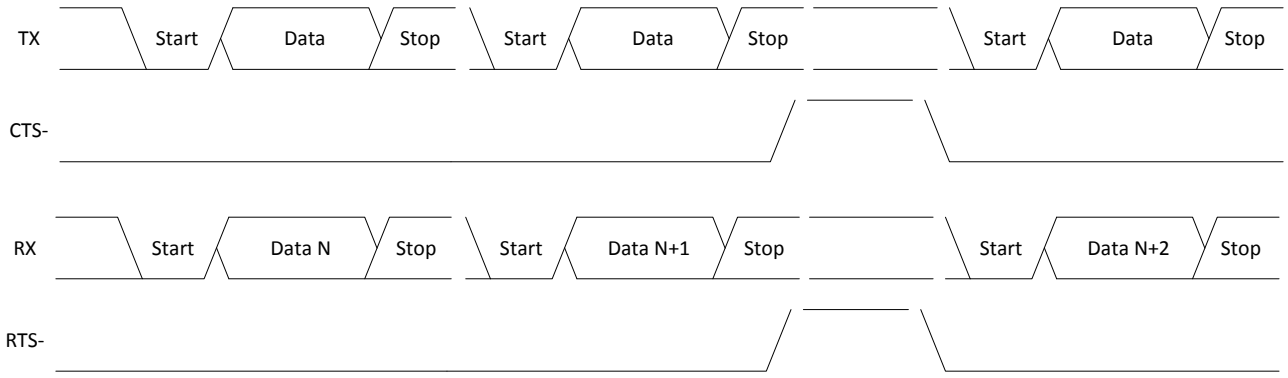


Figure10- 9. RTS/CTS Autoflow Control Timing

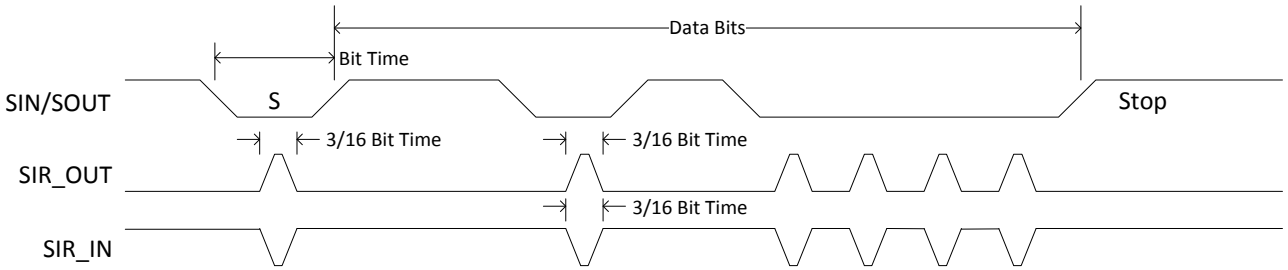


Figure10- 10. Serial IrDA Data Format

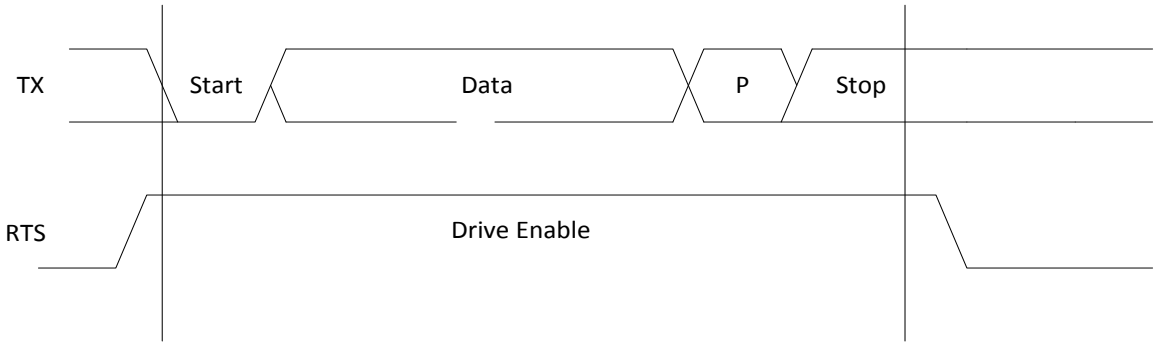


Figure10- 11. RS-485 Timing

10.2.3.5 UART Operating Mode

10.2.3.5.1 Basic Mode Setting

The UART_LCR register can set basic parameter of a data frame: data width, stop bit number, parity type.
A frame transfer of the UART includes the start signal, data signal, parity bit and stop signal. The LSB is transmitted first.
Start signal(start bit): It is the start flag of a data frame. According to UART protocol, the low level of TXD signal indicates the start of a data frame. When the UART does transmit data, the level need hold high.
Data signal(data bit): The data bit width can be configured as 5-bit,6-bit,7-bit,8-bit through different applications.
Parity bit: It is 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 TXD signal indicates the end of a data frame.

10.2.3.5.2 Baud Rate Setting

The baud rate is calculated as follows: Baud rate = SCLK /(16 * divisor).SCLK is usually APB2 and can be set in CCU. 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 different UART mode and error rate is as follows.

Table10- 6. UART Mode Baud and Error Rates

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

Table10- 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

Table10- 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

10.2.3.5.3 DLAB Setting

DLAB control bit (UART_LCR[7]) is the access control bit of divisor Latch register.
If DLAB is 0, then 0x00 offset address is TX/RX FIFO register, 0x04 offset address is IER register.
If DLAB is 1, then 0x00 offset address is DLL register, 0x04 offset address is DLH register.
When UART initial, divisor need be set. That is, writing 1 to DLAB can access the DLL and DLH register, after finished setting, writing 0 to DLAB can access the TX/RX FIFO register.

10.2.3.5.4 CHCFG_AT_BUSY Setting

The function of CHCFG_AT_BUSY(UART_HALT[1]) and CHANGE_UPDATE(UART_HALT[2]) are as follows.
CHCFG_AT_BUSY(configure at busy): Enable the bit, software can also set UART controller when UART is busy, such as the LCR,DLH,DLL register.
CHANGE_UPDATE(change update): If CHCFG_AT_BUSY is enabled, and CHANGE_UPDATE is written to 1, the configuration of UART controller can be updated. After completed update, the bit is cleared to 0 automatically.
Setting divisor, performs the following steps:
Step1 Write CHCFG_AT_BUSY to 1
Step2 Write DLAB to 1, and set DLH and DLL
Step3 Write CHANGE_UPDATE to update configuration. The bit is cleared to 0 automatically after completed update.

10.2.3.5.5 UART Busy

UART_USR[0] is a busy flag of UART controller or not.
When TX transmits data, or RX receives data,or TX FIFO is not empty, or RX FIFO is not empty, then the BUSY flag bit can be set to 1 by hardware, which indicates the UART controller is busy.

10.2.4 Programming Guidelines

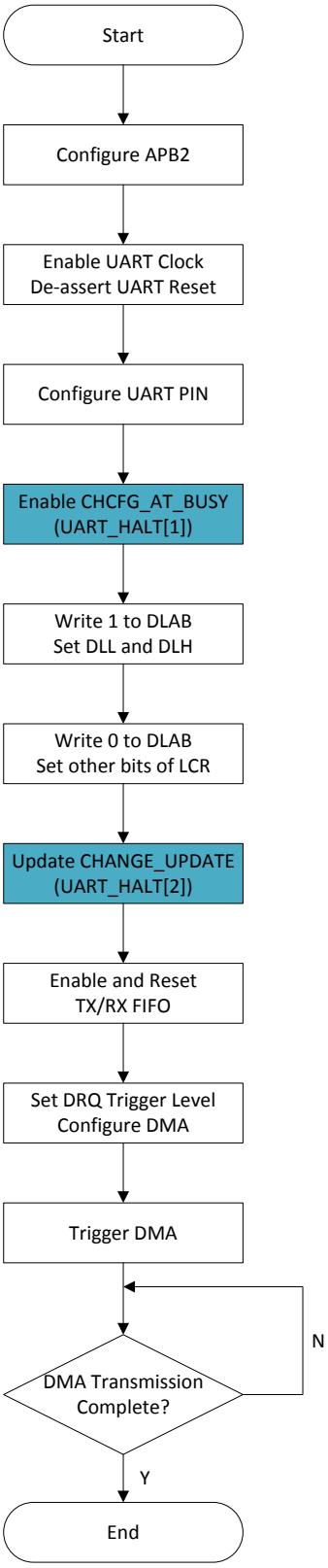


Figure10- 12. UART DRQ Flow Chart

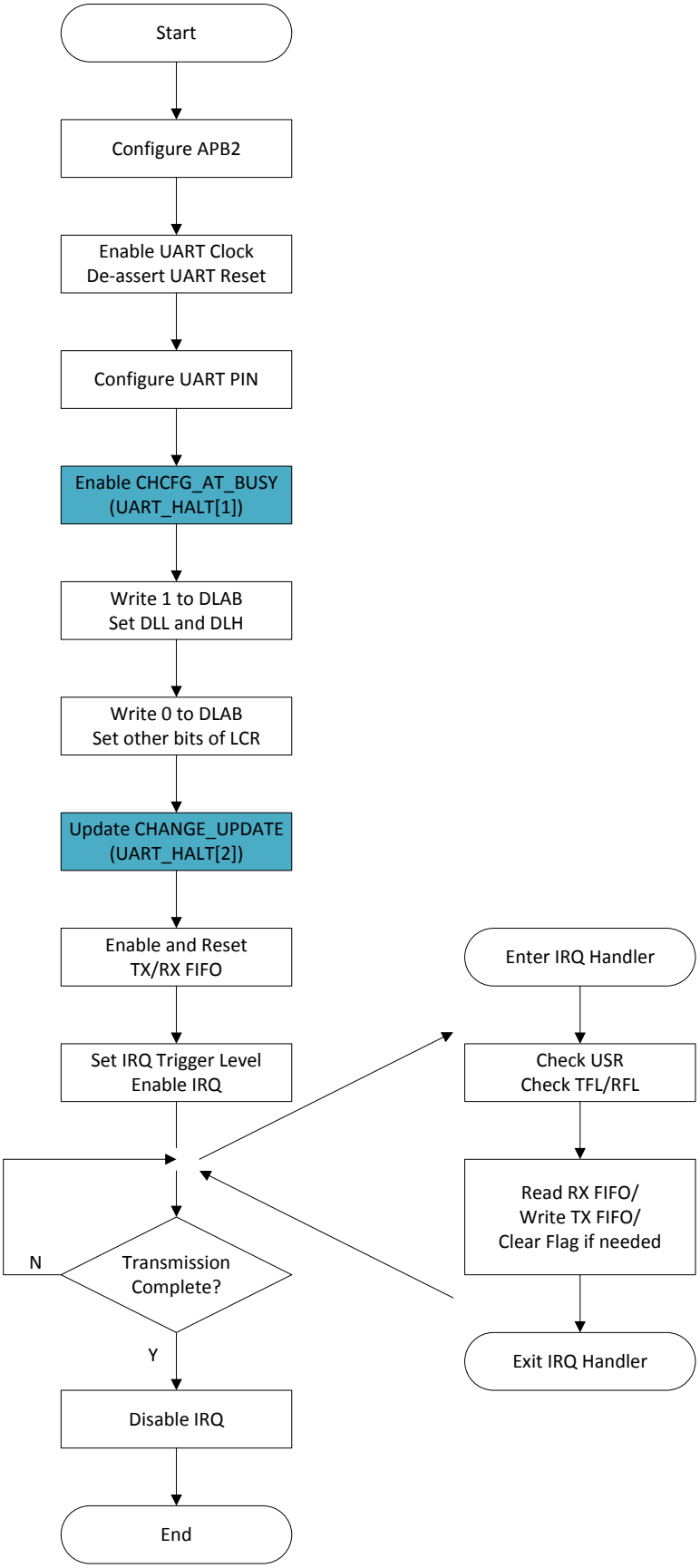


Figure10- 13. UART IRQ Flow Chart

10.2.5 Register List

Module Name	Base Address
UART0	0x05000000
UART1	0x05000400
UART2	0x05000800
UART3	0x05000C00
UART4	0x05001000
R_UART	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
UART_MCR	0x0010	UART Modem Control Register
UART_LSR	0x0014	UART Line Status Register
UART_MSR	0x0018	UART Modem Status Register
UART_SCH	0x001C	UART Scratch Register
UART_USR	0x007C	UART Status Register
UART_TFL	0x0080	UART Transmit FIFO Level Register
UART_RFL	0x0084	UART Receive FIFO Level Register
UART_HSK	0x0088	UART DMA Handshake Configuration Register
UART_HALT	0x00A4	UART Halt TX Register
UART_DBG_DLL	0x00B0	UART Debug DLL Register
UART_DBG_DLH	0x00B4	UART Debug DLH Register
UART_485_CTL	0x00C0	UART RS485 Control and Status Register
RS485_ADDR_MATCH	0x00C4	UART RS485 Address Match Register
BUS_IDLE_CHK	0x00C8	UART RS485 Bus Idle Check Register
TX_DLY	0x00CC	UART TX Delay Register

10.2.6 Register Description

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

Offset: 0x00			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 the Line Status Register (LCR) is set. If in FIFO mode and FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO is preserved, but any incoming data are lost and an overrun error occurs.

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

Offset: 0x00			Register Name: UART_THR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	W	0x0	THR Transmit Holding Register Data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If in FIFO mode and FIFOs are enabled (FCR[0] = 1) and THRE is set, 16 number of characters of data may be written to the THR before the FIFO is full. Any attempt to write data when the FIFO is full results in the write data being lost.

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

Offset: 0x00			Register Name: UART_DLL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R/W	0x0	DLL Divisor Latch Low Lower 8 bits of a 16-bit, read/write, Divisor Latch register that contains the baud rate divisor for the UART. This register may only be accessed when the DLAB bit (LCR[7]) is set and the UART is not busy (USR[0] is zero). The output baud rate is equal to the serial clock (sclk) frequency divided by sixteen times the value of the baud rate divisor, as follows: baud rate = (serial clock freq) / (16 * divisor). Note that with the Divisor Latch Registers (DLL and DLH) set to zero, the baud clock is disabled and no serial communications occur. Also, once the DLL is set, at least 8 clock cycles of the slowest UART clock should be allowed to pass before transmitting or receiving data.

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

Offset: 0x04			Register Name: UART_DLH
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/

7:0	R/W	0x0	DLH Divisor Latch High Upper 8 bits of a 16-bit, read/write, Divisor Latch register that contains the baud rate divisor for the UART. This register may only be accessed when the DLAB bit (LCR[7]) is set and the UART is not busy (USR[0] is zero). The output baud rate is equal to the serial clock (sclk) frequency divided by sixteen times the value of the baud rate divisor, as follows: baud rate = (serial clock freq) / (16 * divisor). Note that with the Divisor Latch Registers (DLL and DLH) set to zero, the baud clock is disabled and no serial communications occur. Also, once the DLH is set, at least 8 clock cycles of the slowest UART clock should be allowed to pass before transmitting or receiving data.

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

Offset: 0x04			**Register Name: UART_IER**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R/W	0x0	PTIME Programmable THRE Interrupt Mode Enable This is used to enable/disable the generation of THRE Interrupt. 0: Disable 1: Enable
6:5	/	/	/
4	R/W	0x0	RS485_INT_EN RS485 Interrupt Enable 0:Disable 1:Enable
3	R/W	0x0	EDSSI Enable Modem Status Interrupt This is used to enable/disable the generation of Modem Status Interrupt. This is the fourth highest priority interrupt. 0: Disable 1: Enable
2	R/W	0x0	ELSI Enable Receiver Line Status Interrupt This is used to enable/disable the generation of Receiver Line Status Interrupt. This is the highest priority interrupt. 0: Disable 1: Enable
1	R/W	0x0	ETBEI Enable Transmit Holding Register Empty Interrupt This is used to enable/disable the generation of Transmitter Holding Register Empty Interrupt. This is the third highest priority interrupt. 0: Disable 1: Enable
0	R/W	0x0	ERBFI Enable Received Data Available Interrupt This is used to enable/disable the generation of Received Data Available Interrupt and the Character Timeout Interrupt (if in FIFO mode and FIFOs enabled). These are the second highest priority interrupts. 0: Disable 1: Enable

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

Offset: 0x08			Register Name: UART_IIR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:6	R	0x0	FEFLAG FIFOs Enable Flag This is used to indicate whether the FIFOs are enabled or disabled. 00: Disable 11: Enable
5:4	/	/	/
3:0	R	0x1	IID Interrupt ID This indicates the highest priority pending interrupt which can be one of the following types: 0000: modem status 0001: no interrupt pending 0010: THR empty 0011:RS485 Interrupt 0100: received data available 0110: receiver line status 0111: busy detect 1100: character timeout Bit 3 indicates an interrupt can only occur when the FIFOs are enabled and used to distinguish a Character Timeout condition interrupt.

Interrupt ID	Priority Level	Interrupt Type	Interrupt Source	Interrupt Reset
0001	-	None	None	-
0110	Highest	Receiver line status	Overrun/parity/ framing errors or break interrupt	Reading the line status register
0011	Second	RS485 Interrupt	In RS485 mode, receives address data and match setting address	Writes 1 to addr flag to reset
0100	Third	Received data available	Receiver data available (non-FIFO mode or FIFOs disabled) or RCVR FIFO trigger level reached (FIFO mode and FIFOs enabled)	Reading the receiver buffer register (non-FIFO mode or FIFOs disabled) or the FIFO drops below the trigger level (FIFO mode and FIFOs enabled)
1100	Fourth	Character timeout indication	No characters in or out of the RCVR FIFO during the last 4 character times and there is at least 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 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	Clear to send or data set ready or	Reading the Modem status Register

		status	ring indicator or data carrier detect. Note that if auto flow control mode is enabled, a change in CTS (that is, DCTS set) does not cause an interrupt.	
0111	Seventh	Busy detect indication	UART_16550_COMPATIBLE = NO and master has tried to write to the Line Control Register while the UART is busy (USR[0] is set to one).	Reading the UART status register

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

Offset: 0x08			Register Name: UART_FCR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:6	W	0x0	RT RCVR Trigger This is used to select the trigger level in the receiver FIFO at which the Received Data Available Interrupt is generated. In auto flow control mode it is used to determine when the rts_n signal is de-asserted. It also determines when the dma_rx_req_n signal is asserted in certain modes of operation. 00: 1 character in the FIFO 01: FIFO ¼ full 10: FIFO ½ full 11: FIFO-2 less than full
5:4	W	0x0	TFT TX Empty Trigger This is used to select the empty threshold level at which the THRE Interrupts are generated when the mode is active. It also determines when the dma_tx_req_n signal is asserted when in certain modes of operation. 00: FIFO empty 01: 2 characters in the FIFO 10: FIFO ¼ full 11: FIFO ½ full
3	W	0x0	DMAM DMA Mode 0: Mode 0 In this mode, if PTE is high and TX FIFO is enabled, the TX DMA request will send when TFL is less than or equal to FIFO Trigger Level. If PTE is high and TX FIFO is disabled, the TX DMA request will send when THRE is empty. If PTE is low, the TX DMA request will send when the TX FIFO is empty. If dma_pte_rx is high and RX FIFO is enabled, the rx drq will send when RFL is equal to or more than FIFO Trigger Level. 1: Mode 1 In this mode, if TX FIFO is enabled and the PTE is high, the TX DMA request will send when TFL is less than or equal to FIFO Trigger Level. If PTE is low, the TX DMA request will send when TX FIFO is empty and the request stops only when TX FIFO is full.

			If RFL is equal to or more than FIFO Trigger Level, the rx drq will be set 1, in otherwise, it will be set 0.
2	W	0x0	XFIFOR XMIT FIFO Reset This resets the control portion of the transmit FIFO and treats the FIFO as empty. This also de-asserts the DMA TX request. It is 'self-clearing'. It is not necessary to clear this bit.
1	W	0x0	RFIFOR RCVR FIFO Reset This resets the control portion of the receive FIFO and treats the FIFO as empty. This also de-asserts the DMA RX request. It is 'self-clearing'. It is not necessary to clear this bit.
0	W	0x0	FIFOE Enable FIFOs This enables/disables the transmit (XMIT) and receive (RCVR) FIFOs. Whenever the value of this bit is changed both the XMIT and RCVR controller portion of FIFOs is reset.

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

Offset: 0x0C			**Register Name: UART_LCR**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R/W	0x0	DLAB Divisor Latch Access Bit It is writeable only when UART is not busy (USR[0] is zero) and always readable. This bit is used to enable reading and writing of the Divisor Latch register (DLL and DLH) to set the baud rate of the UART. This bit must be cleared after initial baud rate setup in order to access other registers. 0: Select RX Buffer Register (RBR) / TX Holding Register(THR) and Interrupt Enable Register (IER) 1: Select Divisor Latch LS Register (DLL) and Divisor Latch MS Register (DLM)
6	R/W	0x0	BC Break Control Bit This is used to cause a break condition to be transmitted to the receiving device. If set to one the serial output is forced to the spacing (logic 0) state. When not in Loopback Mode, as determined by MCR[4], the sout line is forced low until the Break bit is cleared. If SIR_MODE = Enabled and active (MCR[6] set to one) the sir_out_n line is continuously pulsed. When in Loopback Mode, the break condition is internally looped back to the receiver and the sir_out_n line is forced low.
5:4	R/W	0x0	EPS Even Parity Select It is writeable only when UART is not busy (USR[0] is zero) and always writable/readable. This is used to select between even and odd parity, when parity is enabled (PEN set to one). Setting the LCR[5] is unset to reverse the LCR[4]. 00: Odd Parity

			01: Even Parity 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, PEN(LCR[3]) must set to 1.**
3	R/W	0x0	PEN Parity Enable It is writeable only when UART is not busy (USR[0] is zero) and always readable. This bit is used to enable and disable parity generation and detection in transmitted and received serial character respectively. 0: Parity disabled 1: Parity enabled
2	R/W	0x0	STOP Number of stop bits It is writeable only when UART is not busy (USR[0] is zero) and always readable. This is used to select the number of stop bits per character that the peripheral transmits and receives. If set to zero, one stop bit is transmitted in the serial data. If set to one and the data bits are set to 5 (LCR[1:0] set to zero) one and a half stop bits is transmitted. Otherwise, two stop bits are transmitted. Note that regardless of the number of stop bits selected, the receiver checks only the first stop bit. 0: 1 stop bit 1: 1.5 stop bits when DLS (LCR[1:0]) is zero, else 2 stop bit
1:0	R/W	0x0	DLS Data Length Select It is writeable only when UART is not busy (USR[0] is zero) and always readable. This is used to select the number of data bits per character that the peripheral transmits and receives. The number of bit that may be selected areas follows: 00: 5 bits 01: 6 bits 10: 7 bits 11: 8 bits

10.2.6.9 UART Modem Control Register(Default Value: 0x0000_0000)

Offset: 0x10			**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:Reverse
5	R/W	0x0	AFCE Auto Flow Control Enable When FIFOs are enabled and the Auto Flow Control Enable (AFCE) bit is set, Auto Flow Control features are enabled. 0: Auto Flow Control Mode disabled

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

10.2.6.10 UART Line Status Register(Default Value: 0x0000_0060)

Offset:0x14			Register Name: UART_LSR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7	R	0x0	FIFOERR

			RX Data Error in FIFO When FIFOs are disabled, this bit is always 0. When FIFOs are enabled, this bit is set to 1 when there is at least one PE, FE, or BI in the RX FIFO. It is cleared by a read from the LSR register provided there are no subsequent errors in the FIFO.
6	R	0x1	TEMT Transmitter Empty If the FIFOs are disabled, this bit is set to "1" whenever the TX Holding Register and the TX Shift Register are empty. If the FIFOs are enabled, this bit is set whenever the TX FIFO and the TX Shift Register are empty. In both cases, this bit is cleared when a byte is written to the TX data channel.
5	R	0x1	THRE TX Holding Register Empty If the FIFOs are disabled, this bit is set to "1" whenever the TX Holding Register is empty and ready to accept new data and it is cleared when the CPU writes to the TX Holding Register. If the FIFOs are enabled, this bit is set to "1" whenever the TX FIFO is empty and it is cleared when at least one byte is written to the TX FIFO.
4	R	0x0	BI Break Interrupt This is used to indicate the detection of a break sequence on the serial input data. If in UART mode (SIR_MODE == Disabled), it is set whenever the serial input, 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 whenever 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 LSR clears the BI bit. In the non-FIFO mode, the BI indication occurs immediately and persists until the LSR is read.
3	RC	0x0	FE Framing Error This is used to indicate the occurrence of a framing error in the receiver. A framing error occurs when the receiver does not detect a valid STOP bit in the received data. In the FIFO mode, since the framing error is associated with a character received, it is revealed when the character with the framing error is at the top of the FIFO. When a framing error occurs, the UART tries to resynchronize. It does this by assuming that the error was due to the start bit of the next character and then continues receiving the other bit i.e. data, and/or parity and stop. It should be noted that the Framing Error (FE) bit (LSR[3]) is set if a break interrupt has occurred, as indicated by Break Interrupt (BI) bit (LSR[4]). 0: no framing error 1:framing error Reading the LSR clears the FE bit.
2	RC	0x0	PE Parity Error This is used to indicate the occurrence of a parity error in the receiver if the

			Parity Enable (PEN) bit (LCR[3]) is set. In the FIFO mode, since the parity error is associated with a character received, it is revealed when the character with the parity error arrives at the top of the FIFO. It should be noted that the Parity Error (PE) bit (LSR[2]) is set if a break interrupt has occurred, as indicated by Break Interrupt (BI) bit (LSR[4]). 0: no parity error 1: parity error Reading the LSR clears the PE bit.
1	RC	0x0	OE Overrun Error This occurs if a new data character was received before the previous data was read. In the non-FIFO mode, the OE bit is set when a new character arrives in the receiver before the previous character was read from the RBR. When this happens, the data in the RBR is overwritten. In the FIFO mode, an overrun error occurs when the FIFO is full and a new character arrives at the receiver. The data in the FIFO is retained and the data in the receive shift register is lost. 0: no overrun error 1: overrun error Reading the LSR clears the OE bit.
0	R	0x0	DR Data Ready This is used to indicate that the receiver contains at least one character in the RBR or the receiver FIFO. 0: no data ready 1: data ready This bit is cleared when the RBR is read in non-FIFO mode, or when the receiver FIFO is empty, in FIFO mode.

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

Offset: 0x18			**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)
6	R	0x0	RI Line State of Ring Indicator This is used to indicate the current state of the modem control line ri_n. This bit is the complement of ri_n. When the Ring Indicator input (ri_n) is asserted it is an indication that a telephone ringing signal has been received by the modem or data set. 0: ri_n input is de-asserted (logic 1) 1: ri_n input is asserted (logic 0)
5	R	0x0	DSR

			Line State of Data Set Ready This is used to indicate the current state of the modem control line dsr_n. This bit is the complement of dsr_n. When the Data Set Ready input (dsr_n) is asserted it is an indication that the modem or data set is ready to establish communications with UART. 0: dsr_n input is de-asserted (logic 1) 1: dsr_n input is asserted (logic 0) In Loopback Mode (MCR[4] set to one), DSR is the same as MCR[0] (DTR).
4	R	0x0	CTS Line State of Clear To Send This is used to indicate the current state of the modem control line cts_n. This bit is the complement of cts_n. When the Clear to Send input (cts_n) is asserted it is an indication that the modem or data set is ready to exchange data with UART. 0: cts_n input is de-asserted (logic 1) 1: cts_n input is asserted (logic 0) In Loopback Mode (MCR[4] = 1), CTS is the same as MCR[1] (RTS).
3	RC	0x0	DDCD Delta Data Carrier Detect This is used to indicate that the modem control line dcd_n has changed since the last time the MSR was read. 0: no change on dcd_n since last read of MSR 1: change on dcd_n since last read of MSR Reading the MSR clears the DDCD bit. **NOTE** **If the DDCD bit is not set and the dcd_n signal is asserted (low) and a reset occurs (software or otherwise), then the DDCD bit is set when the reset is removed if the dcd_n signal remains asserted.**
2	RC	0x0	TERI Trailing Edge Ring Indicator This is used to indicate that a change on the input ri_n (from an active-low to an inactive-high state) has occurred since the last time the MSR was read. 0: no change on ri_n since last read of MSR 1: change on ri_n since last read of MSR Reading the MSR clears the TERI bit.
1	RC	0x0	DDSR Delta Data Set Ready This is used to indicate that the modem control line dsr_n has changed since the last time the MSR was read. 0: no change on dsr_n since last read of MSR 1: change on dsr_n since last read of MSR Reading the MSR clears the DDSR bit. In Loopback Mode (MCR[4] = 1), DDSR reflects changes on MCR[0] (DTR). If the DDSR bit is not set and the dsr_n signal is asserted (low) and a reset occurs (software or otherwise), then the DDSR bit is set when the reset is removed if the dsr_n signal remains asserted.
0	RC	0x0	DCTS Delta Clear to Send This is used to indicate that the modem control line cts_n has changed since the last time the MSR was read. 0: no change on ctsdsr_n since last read of MSR 1: change on ctsdsr_n since last read of MSR

			Reading the MSR clears the DCTS bit. In Loopback Mode (MCR[4] = 1), DCTS reflects changes on MCR[1] (RTS). If the DCTS bit is not set and the cts_n signal is asserted (low) and a reset occurs (software or otherwise), then the DCTS bit is set when the reset is removed if the cts_n signal remains asserted.

10.2.6.12 UART Scratch Register(Default Value: 0x0000_0000)

Offset: 0x1C			Register Name: UART_SCH
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R/W	0x0	SCRATCH_REG Scratch Register This register is for programmers to use as a temporary storage space. It has no defined purpose in the UART.

10.2.6.13 UART Status Register(Default Value: 0x0000_0006)

Offset: 0x7C			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.
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 in 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

10.2.6.14 UART Transmit FIFO Level Register(Default Value: 0x0000_0000)

Offset: 0x80			Register Name: UART_TFL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:9	/	/	/
8:0	R	0x0	TFL Transmit FIFO Level This is indicates the number of data entries in the transmit FIFO.

10.2.6.15 UART Receive FIFO Level Register(Default Value: 0x0000_0000)

Offset: 0x84			Register Name: UART_RFL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:9	/	/	/
8:0	R	0x0	RFL Receive FIFO Level This is indicates the number of data entries in the receive FIFO.

10.2.6.16 UART DMA Handshake Configuration Register(Default Value: 0x0000_00E5)

Offset: 0x88			Register Name: UART_HSK
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R/W	0xE5	Handshake configuration 0xA5: DMA wait cycle mode 0xE5: DMA handshake mode

10.2.6.17 UART Halt TX Register(Default Value: 0x0000_0000)

Offset: 0xA4			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 PTE is set 1, when TFL is less than trig, send the DMA request. If PTE is set 0, when FIFO is empty, send the DMA request. The DMA request will stop when FIFO is full. In DMA0 mode, if PTE is set 1 and FIFO on, when TFL is less than trig, send DMA request. If PTE is set 1 and FIFO off, when THRE is empty, send DMA request. If PTE is set 0, when FIFO is empty, send DMA request.

6	R/W	0x0	DMA_PTE_RX The sending of RX_DRQ. In DMA1 mode, when RFL is more than or equal to trig or receive timeout, send DRQ. In DMA0 mode, if DMA_PTE_RX = 1 and FIFO on, when RFL is more than trig, send DRQ. In other cases, once the receive data is valid, send 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	/	/	/
2	R/WAC	0x0	CHANGE_UPDATE After the user uses HALT[1] to change the baud rate or LCR configuration, write 1 to update the configuration and wait this bit 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 baudrate register (DLH and DLL) when the UART is busy (USB[0] is 1). 1: Enable change when busy
0	R/W	0x0	HALT_TX Halt TX This register is use to halt transmissions for testing, so that the transmit FIFO can be filled by the master when FIFOs are implemented and enabled. 0 : Halt TX disabled 1 : Halt TX enabled  **NOTE** **If FIFOs are not enabled, the setting has no effect on operation.**

10.2.6.18 UART DBG DLL Register(Default Value: 0x0000_0000)

Offset: 0xB0			**Register Name: UART_DBG_DLL**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R/W	0x0	DEBUG DLL

10.2.6.19 UART DBG DLH Register(Default Value: 0x0000_0000)

Offset: 0xB4			**Register Name: UART_DBG_DLH**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R/W	0x0	DEBUG DLH

10.2.6.20 UART RS485 Control and Status Register(Default Value: 0x0000_0000)

Offset: 0xC0			Register Name: UART_485_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:7	/	/	Reserved
6	R/W1C	0x0	AAD_ADDR_F In AAD mode, when UART receives an address byte and the byte is the same as RS485_ADDR_MATCH, this bit will be set 1.If RS485 interrupt is enabled, the RS485 interrupt will arrive. Write 1 to clear this bit and reset the RS485 interrupt.
5	R/W1C	0x0	RS485_ADDR_DET_F This is a flag of the detecting of address bytes. When UART receives an address byte, this bit will be set 1.If the RS485 Interrupt is enabled, the RS485 interrupt will arrive. 1:An address byte is detected 0:No address byte is detected Write 1 to clear this bit and reset the RS485 interrupt.
4	/	/	reverse
3	R/W	0x0	RX_BF_ADDR In NMM mode, If set this bit to 1, UART will receive all the bytes into FIFO before receiving an address byte. If set to 0, it will not. 1:Receive 0:Not Receive
2	R/W	0x0	RX_AF_ADDR In NMM mode, if set this bit to 1, UART will receive all the bytes into FIFO after receiving an address byte. If set to 0, it will not. 1:Receive 0:Not Receive
1:0	R/W	0x0	RS485_SLAVE_MODE_SEL RS485 Slave Mode 00: Normal Multidrop Operation（NMM） 01: Auto Address Detection Operation（AAD） 10: reserved 11: reserved

10.2.6.21 UART RS485 Address Match Register(Default Value: 0x0000_0000)

Offset: 0xC4			Register Name: RS485_ADDR_MATCH
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R/W	0x0	ADDR_MATCH The matching address uses in AAD mode. It is only available for AAD.

10.2.6.22 UART RS485 Bus Idle Check Register(Default Value: 0x0000_0000)

Offset: 0xC8	Register Name: BUS_IDLE_CHK

Bit	Read/Write	Default/Hex	Description
31:8	/	/	/
7	R/W	0x0	BUS_IDLE_CHK_EN 1: Enable bus idle check function 0: Disable bus idle check function
6	R	0x0	BUS_STATUS The Flag of Bus Status 1:busy 0:idle
5:0	R	0x0	ADJ_TIME Bus Idle Time. The unit is 8*16*Tclk.

10.2.6.23 UART TX Delay Register(Default Value: 0x0000_0000)

Offset: 0xCC			Register Name: TX_DLY
Bit	**Read/Write**	**Default/Hex**	**Description**
31:8	/	/	/
7:0	R/W	0x0	DLY The delay time between the last stop bit and the next start bit. The unit is 16*Tclk. It is used to control the space between two bytes in TX.

10.3 RSB

10.3.1 Overview

The RSB(reduced serial bus) Host Controller is designed to communicate with RSB Device using two push-pull wires. It supports a simplified two wire protocol on a push-pull bus. The transfer speed can be up to 10MHz and the performance will be improved much. The RSB bus protocol is designed and implemented by the Allwinner Technology. Allwinner technology has the final interpretation of the IP.

The RSB has the following features:

- Supports industry-standard AMBA Peripheral Bus (APB) and it is fully compliant with the AMBA Specification, Revision 2.0.
- Speed up to 10MHz with ultra low power
- Push-Pull bus
- Supports host mode
- Programmable output delay of CD signal
- Parity check for address and data transmission
- Supports multi-devices

10.3.2 Block Diagram

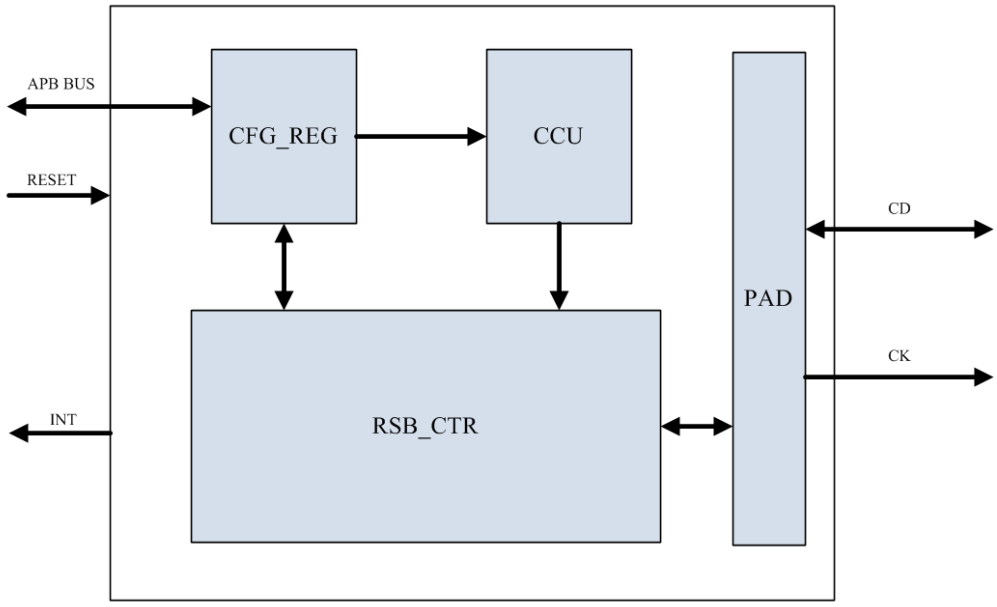


Figure10- 14. RSB Block Diagram

RESET: Module reset signal

INT: Module output interrupt signal

CFG_REG: Module configuration register

RSB_CTR: Packet encoding/decoding

CCU: Module clock controller unit

10.4 SPI

10.4.1 Overview

The SPI is a full-duplex, synchronous, serial communication interface which allows rapid data communication with software interrupts. The SPI controller contains one 64x8 receiver buffer (RXFIFO) and one 64x8 transmit buffer (TXFIFO). It can work at master mode and slave mode.

The SPI has the following features:

- Full-duplex synchronous serial interface
- 5 clock sources
- 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
- Interrupt or DMA support
- Supports Mode0, Mode1, Mode2 and Mode3
- Support the Maximum IO Rate: 100MHz, but the recommended production frequency is 40MHz ~ 80MHz
- Supports 3-wire/4-wire SPI
- Supports programmable serial data frame length: 0 bit to 32 bits
- Supports Standard SPI/Dual-Output/Dual-Input SPI/Dual I/O SPI and Quad-Output/Quad-Input SPI

10.4.2 Block Diagram

Figure 10-15 shows a block diagram of the SPI.

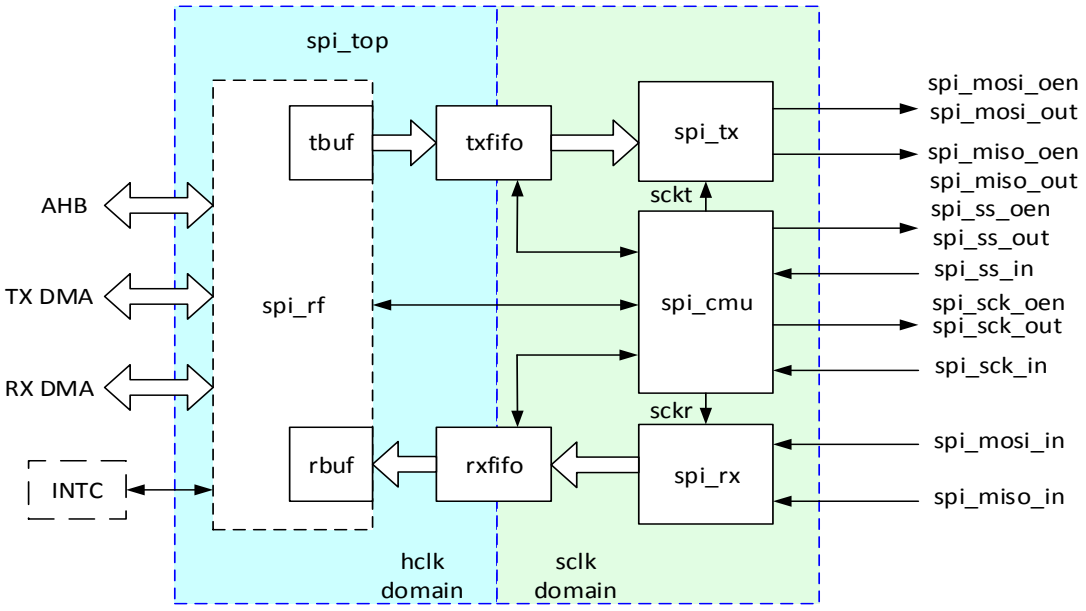


Figure10- 15. SPI Block Diagram

The SPI comprises with:

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 8bits,then the data is written into the rxfifo.

spi_rbuf: The block is used as converted the rxfifo data into read data length of AHB.
txfifo, rxfifo: For transmit and receive transfers, data transmitted from the SPI to the external serial device is written into the txfifo; data received from the external serial device into SPI is pushed into the rxfifo.
spi_cmu: Responsible for implementing SPI bus clock, chip select, internal sample and the generation of transfer clock.
spi_tx: Responsible for implementing SPI data transfer ,the interface of the internal txfifo and status register.
spi_rx: Responsible for implementing SPI data receive, the interface of the internal rxfifo and status register.

10.4.3 Operations and Functional Descriptions

10.4.3.1 External Signals

Table 10-9 describes the external signals of SPI. MOSI and MISO are bidirectional I/O, when SPI is configured as master device, CLK and CS are output pin; when SPI is configurable as slave device, CLK and CS are input pin. The unused SPI ports are used as General Purpose I/O ports.

Table10- 9. SPI External Signals

Signal	Description	Type
SPI0_CS0	SPI0 chip select signal0, low active	I/O
SPI0_CS1	SPI0 chip select signal1, low active	I/O
SPI0_CLK	SPI0 clock signal	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 active low, or serial data input and output for quad input or quad output	I/O
SPI0_HOLD	The HOLD pin is used to temporarily pause serial communication without deselecting or resetting the device. While the HOLD pin is asserted, transitions on the SCK pin and data on the SI pin will be ignored, or serial data input and output for quad input or quad output	I/O
SPI1_CS0	SPI1 chip select signal0, low active	I/O
SPI1_CS1	SPI1 chip select signal1, low active	I/O
SPI1_CLK	SPI1 clock signal	I/O
SPI1_MOSI	SPI1 master data out, slave data in	I/O
SPI1_MISO	SPI1 master data in, slave data out	I/O

10.4.3.2 Clock Sources

Each SPI controller gets five different clocks, users can select one of them to make SPI clock source. Table 10-10 describes the clock sources for SPI.

Table10- 10. SPI Clock Sources

Clock Sources	Description
OSC24M	24MHz Crystal
PLL_PERIPH0(1X)	Peripheral Clock, default value is 600MHz
PLL_PERIPH1(1X)	Peripheral Clock, default value is 600MHz
PLL_PERIPH0(2X)	Peripheral Clock, default value is 1200MHz
PLL_PERIPH1(2X)	Peripheral Clock, default value is 1200MHz

10.4.3.3 Typical Application

Figure 10-16 shows the application block diagram when the SPI master device is connected to a slave device.

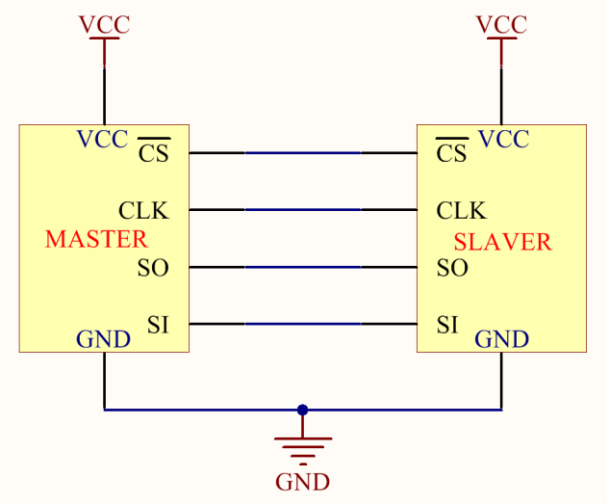


Figure10- 16. SPI Application Block Diagram

10.4.3.4 SPI Transmit Format

The SPI supports 4 different formats for data transfer. Software can select one of the four modes in which the SPI works by setting the bit1(Polarity) and bit0(Phase) of **SPI Transfer Control Register**. 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.

During Phase 0, Polarity 0 and Phase 1, Polarity 1 operations, output data changes on the falling clock edge and input data is shifted in on the rising edge.

During Phase 1, Polarity 0 and Phase 0, Polarity 1 operations, output data changes on the rising edges of the clock and is shifted in on falling edges.

The POL defines the signal polarity when SPI_SCLK is in idle state. The SPI_SCLK is high level when POL is '1' and it is low level when POL is '0'. The PHA decides whether the leading edge of SPI_SCLK is used for setup or sample data. The leading edge is used for setup data when PHA is '1' and for sample data when PHA is '0'. The four modes are listed in Table 10-11.

Table10- 11. SPI Transmit Format

SPI Mode	POL	PHA	Leading Edge	Trailing Edge
0	0	0	Rising, Sample	Falling, Setup
1	0	1	Rising, Setup	Falling, Sample
2	1	0	Falling, Sample	Rising, Setup
3	1	1	Failing, Setup	Rising, Sample

Figure 10-17 and Figure 10-18 describe four waveforms for SPI_SCLK.

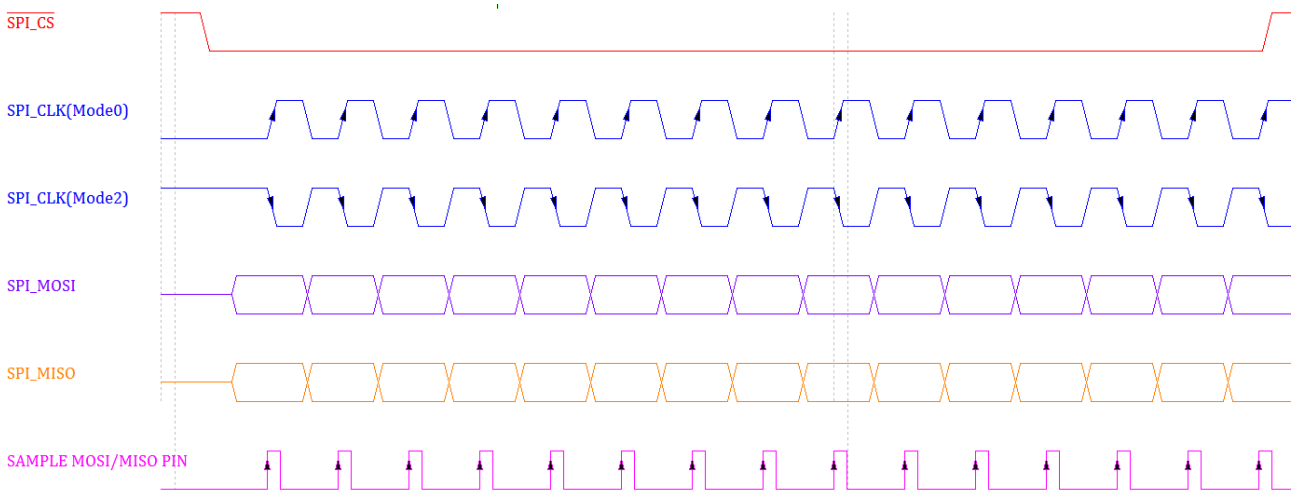


Figure10- 17. SPI Phase 0 Timing Diagram

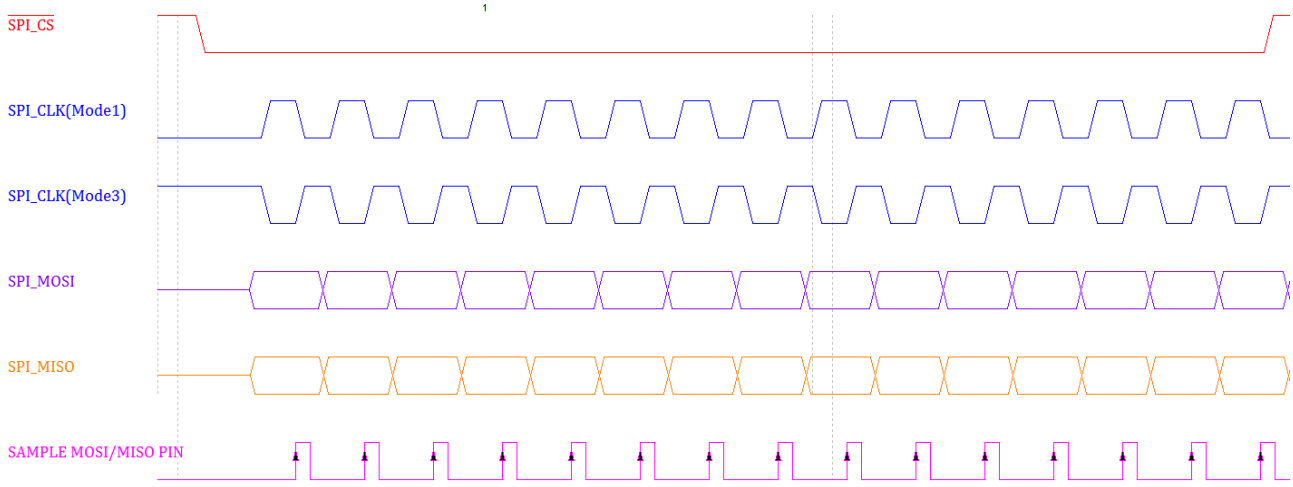


Figure10- 18. SPI Phase 1 Timing Diagram

10.4.3.5 SPI Master and Slave Mode

The SPI controller can be configured to a master or slave device. Master mode is selected by setting the **MODE** bit in the **SPI Global Control Register;** slave mode is selected by clearing the the **MODE** bit in the **SPI Global Control Register**. In master mode, SPI_CLK is generated and transmitted to external device, and data from the TX FIFO is transmitted on the MOSI pin, the data from slave is received on the MISO pin and sent to RX FIFO. Chip Select(SPI_SS) is active low signal. SPI_SS must be set low before data are transmitted or received. SPI_SS can be selected SPI auto control or software manual control. When using auto control, **SS_OWNER**(the bit 6 in the **SPI Transfer Control Register**) must be cleared(default value is 0);when using manual control, **SS_OWNER** must be set, Chip Select level is controlled by **SS_LEVEL** bit(the bit 7 in the **SPI Transfer Control Register**).

In slave mode, after software selects the **MODE** bit to '0',it waits for master initiate a transaction. When the master assertes SPI_SS and SPI_CLK is transmitted to the slave, the slave data is transmitted from TX FIFO on MISO pin and data from MOSI pin is received in RX FIFO.

10.4.3.6 SPI 3-Wire Mode

The SPI 3-wire mode is only valid when the SPI controller work in master mode, and selected when the **Work Mode Select(bit[1:0])** is equal to 0x2 in the **SPI Bit-Aligned Transfer Configure Register**. and in the 3-wire mode, the input data and the output data use the same single data line. The following figure describes this mode.

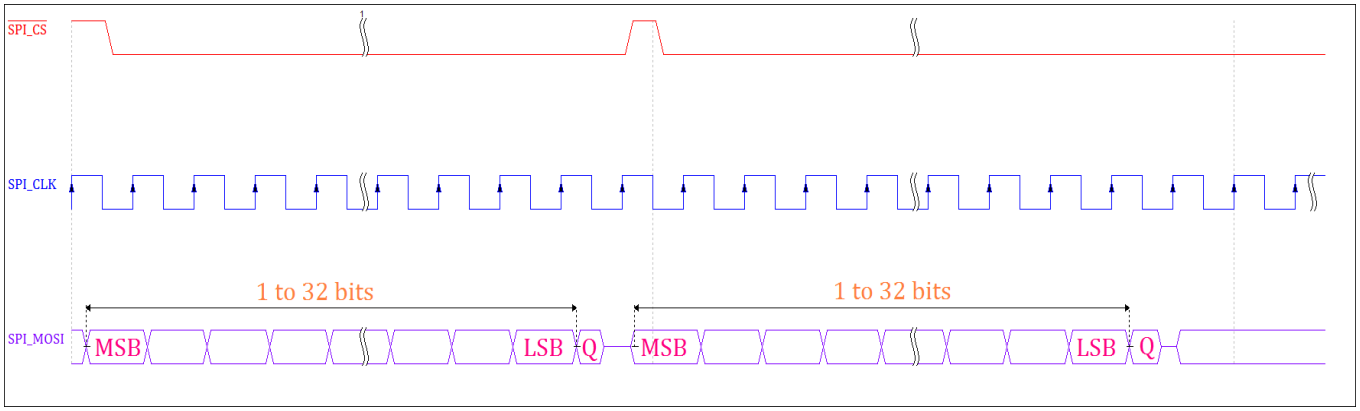


Figure10- 19. SPI 3-Wire Mode

10.4.3.7 SPI Dual Read Mode

The dual read mode(SPI x2) is selected when the **DRM**(bit28) is set in the **SPI Master Burst Control Counter Register**. Using the dual mode allows data to be transferred to or from the device at two times the rate of standard single mode SPI devices, data can be read at fast speed using two data bits(MOSI and MISO) at a time. The following figure describes the Dual Input/Dual Output SPI(Figure 10-20) and the Dual IO SPI(Figure 10-21).

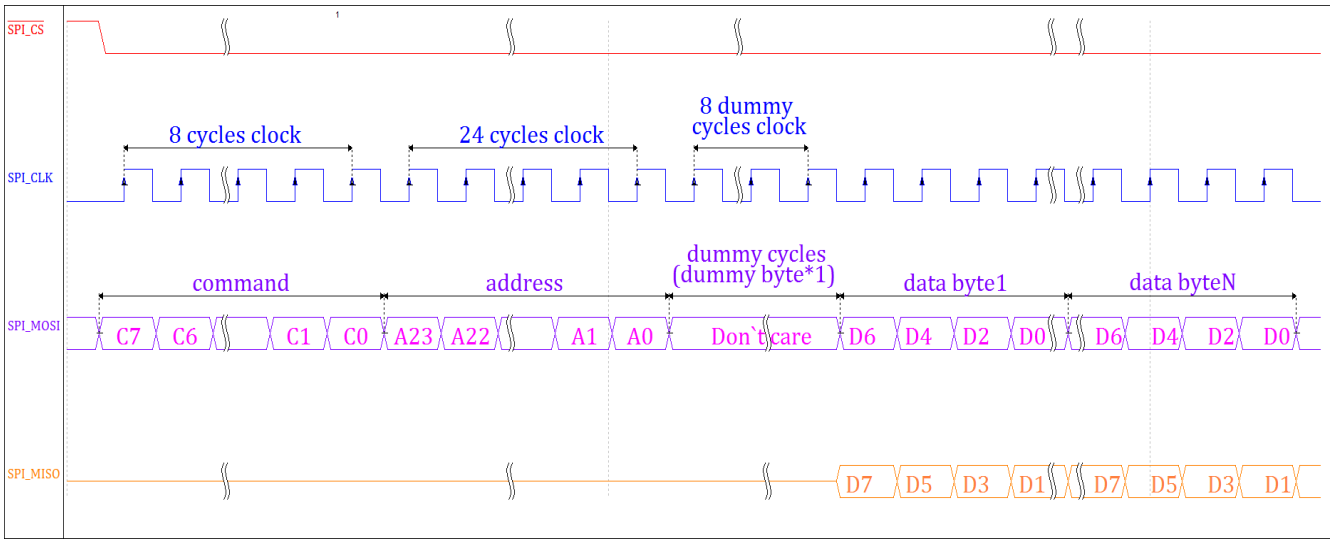


Figure10- 20. SPI Dual Read Mode

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

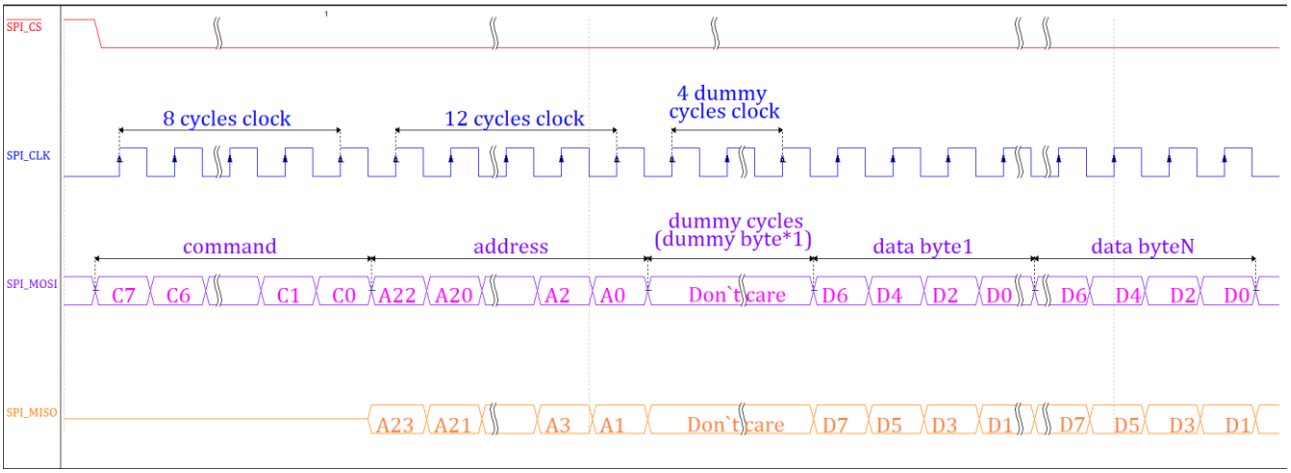


Figure10- 21. SPI Dual IO Mode

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

10.4.3.8 SPI Quad Mode

The quad read mode(SPI x4) is selected when the **Quad_EN**(bit29) is set in the **SPI Master Burst Control Counter Register**. Using the quad mode allows data to be transferred to or from the device at 4 times the rate of standard single mode SPI devices, data can be read at fast speed using four data bits(MOSI, MISO, IO2(WP#)and IO3(HOLD#)) at the same time. The following figure describes the Quad Input/Quad Output SPI.

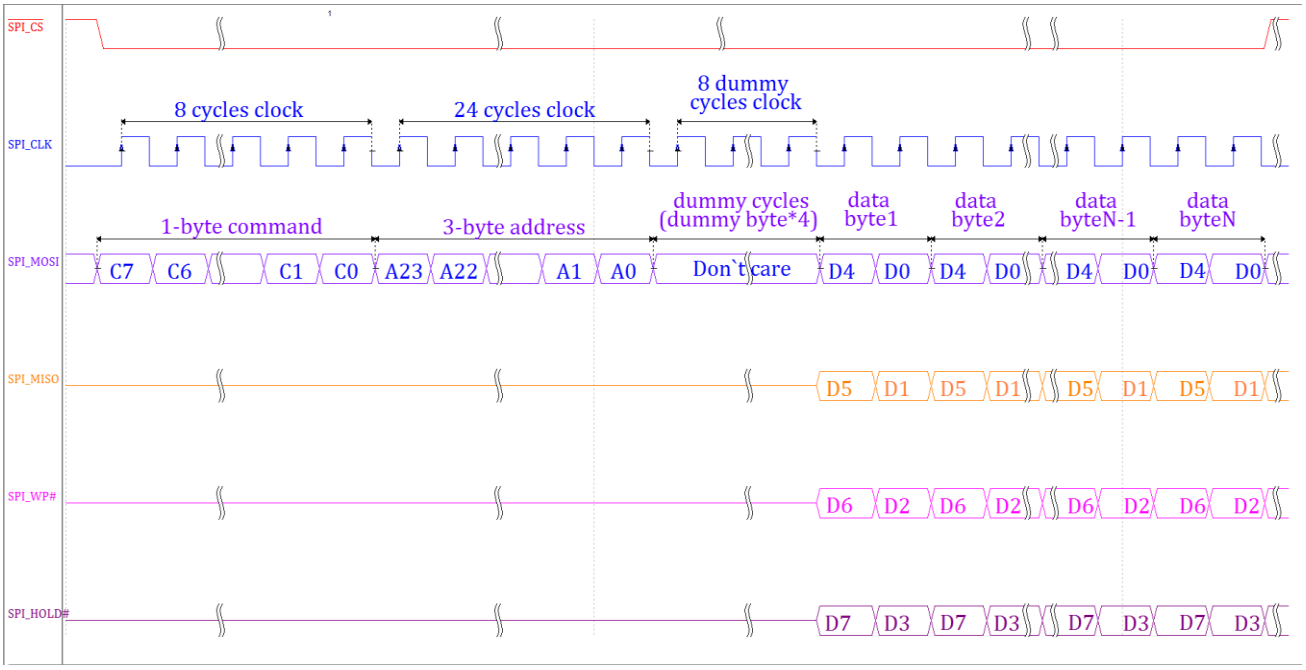


Figure10- 22. SPI Quad Read Mode

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

10.4.4 Programming Guidelines

10.4.4.1 CPU or DMA Operation

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

Write Data: CPU or DMA must write data on the register SPI_TXD, 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 an completed interrupt(the TC bit in SPI Interrupt Status Register) to the processor at the end of each transfer.

(1).CPU Mode

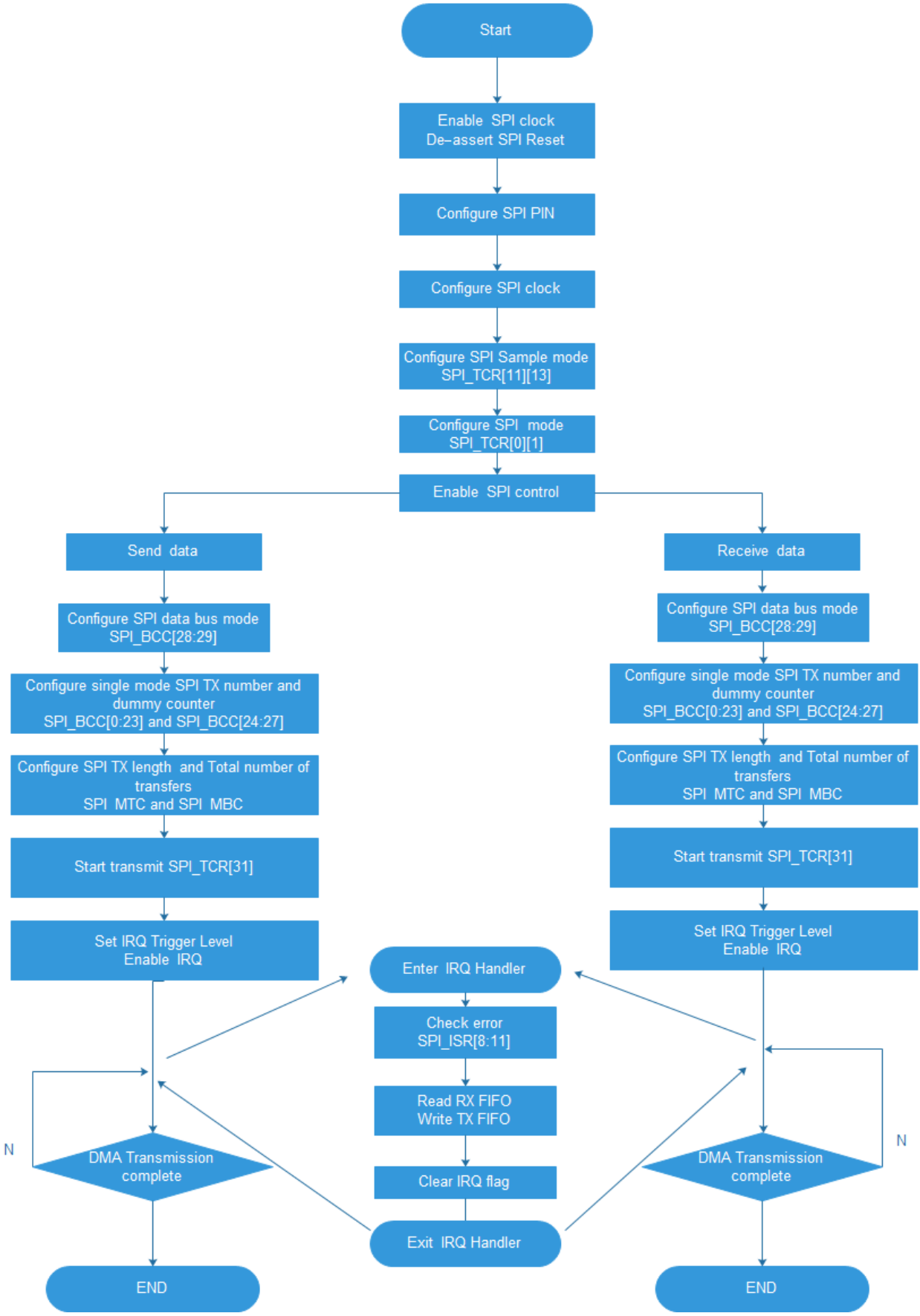


Figure10- 23. CPU Mode

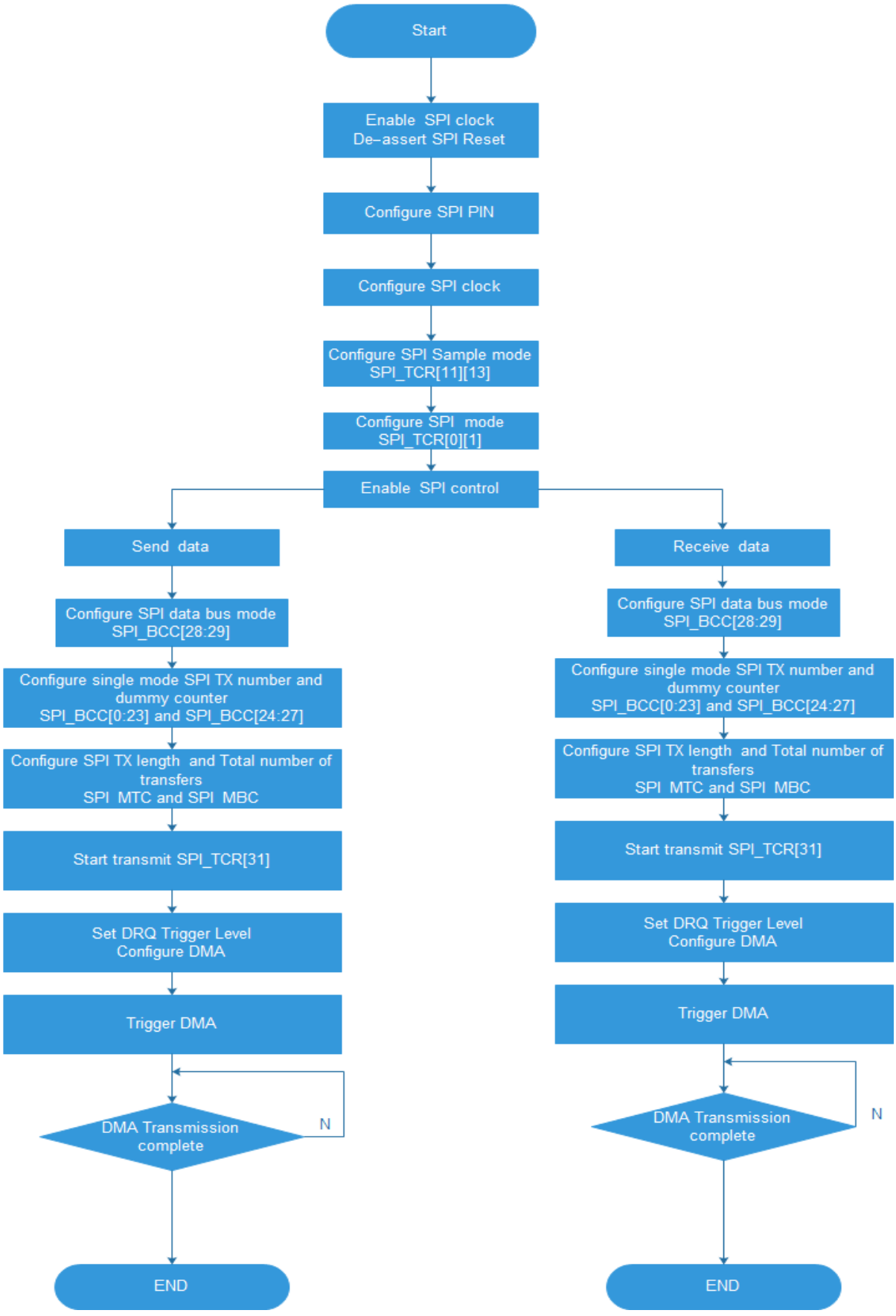


Figure10- 24. DMA Mode

10.4.4.2 Transmit/Receive Burst in Master Mode

In SPI Master mode, the transmit and receive burst(byte in unit) are configured before the SPI transfers serial data between the processor and external device. The transmit burst write in MWTC(bit[23:0]) of **SPI Master Transmit Counter Register.** The transmit burst in single mode before automatically sending dummy burst write in STC(bit[23:0]) of **SPI Master Burst Control Counter Register.** For dummy data, SPI controller can automatically sent before receive by writing DBC(bit[27:24]) in **SPI Master Burst Control Counter Register**. If users don't use SPI controller to sent dummy data automatically, then the dummy bursts are used as the transmit counters to write together in MWTC(bit[23:0]) of **SPI Master Transmit Counter Register.** In Master mode, the total burst numbers write in MBC(bit[23:0]) of **SPI Master Burst Counter Register.** When all transmit burst and receive burst are transferred, SPI contr oller will send an completed interrupt, at the same time, SPI controller will clear DBC,MWTC and MBC.

10.4.4.3 SPI Sample Mode and Run Clock Configuration

The SPI Controller runs at 3kHz~100MHz at its interface to external SPI devices. The internal SPI Clock should run at the same frequency as the outgoing clock in master mode. The SPI clock is selected different clock sources, SPI must configure different work mode. There are three work mode: normal sample mode, delay half cycle sample mode, delay one cycle sample mode. Delay half cycle sample mode is the default mode of SPI controller. When SPI runs at 40MHz or below 40MHz, SPI can work at normal sample mode or delay half cycle sample mode. When SPI runs over 75MHz,Set the **SDC** bit in **SPI Transfer Control Register** to '1' to make 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 propagating between master and slave. The different configuration of SPI sample mode shows in Table 10-12.

Table10- 12. SPI Sample Mode and Run Clock

SPI Sample Mode	SDM(bit13)	SDC(bit11)	Run Clock
normal sample	1	0	<=24MHz
delay half cycle sample	0	0	<=40MHz
delay one cycle sample	0	1	>=75MHz

10.4.5 Register List

Module Name	Base Address
SPI0	0x05010000
SPI1	0x05011000

Register Name	Offset	Description
SPI_GCR	0x0004	SPI Global Control Register
SPI_TCR	0x0008	SPI Transfer Control Register
SPI_IER	0x0010	SPI Interrupt Control Register
SPI_ISR	0x0014	SPI Interrupt Status Register
SPI_FCR	0x0018	SPI FIFO Control Register
SPI_FSR	0x001C	SPI FIFO Status Register
SPI_WCR	0x0020	SPI Wait Clock Counter Register
SPI_CCR	0x0024	SPI Clock Rate Control Register
SPI_MBC	0x0030	SPI Burst Counter Register

SPI_MTC	0x0034	SPI Transmit Counter Register
SPI_BCC	0x0038	SPI Burst Control Register
SPI_BATCR	0x003C	SPI Bit-Aligned Transfer Configure Register
SPI_3W_CCR	0x0040	SPI 3Wire Clock Configuration Register
SPI_TBR	0x0044	SPI TX Bit Register
SPI_RBR	0x0048	SPI RX Bit Register
SPI_NDMA_MODE_CTL	0x0088	SPI Normal DMA Mode Control Register
SPI_TXD	0x0200	SPI TX Data Register
SPI_RXD	0x0300	SPI RX Data Register

10.4.6 Register Description

10.4.6.1 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' has no effect.
30:8	/	/	/
7	R/W	0x1	TP_EN Transmit Pause Enable In master mode, it is used to control transmit state machine to stop smart burst sending when RX FIFO is full. 1: Stop transmit data when RXFIFO full 0: Normal operation, ignore RXFIFO status Cannot be written when XCH=1
6:2	/	/	/
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.

10.4.6.2 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.
30:15	/	/	/
14	R/W	0x0	SDDM Sending Data Delay Mode 0:Normal sending 1:Delay sending Set the bit to"1" to make the data that should be sent with a delay of half cycle of SPI_CLK in dual IO mode for SPI mode 0.
13	R/W	0x0	SDM Master Sample Data Mode 0: Delay sample mode 1: Normal sample mode In normal sample mode, SPI master samples the data at the correct edge for each SPI mode; In delay sample mode, SPI master samples data at the edge that is half cycle delayed by the correct edge defined in respective SPI mode.
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 of SPI_CLK. It is used in high speed read operation to reduce the error caused by the time delay of SPI_CLK propagating between master and slave. 0: Normal operation, do not delay internal read sample point 1: Delay internal read sample point 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 BC period 1: Discard unused SPI bursts, only fetching the SPI bursts during dummy burst period. The bursts number is specified by TC. Cannot be written when XCH=1.
7	R/W	0x1	SS_LEVEL When control SS signal manually (SPI_CTRL_REG.SS_CTRL==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, controller sends SS signal automatically with data together. When this bit is set to 1, software must manually write SPI_CTL_REG.SS_LEVEL to 1 or 0 to control the level of 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 wave form for the SPI_SSx signal. Only valid when SS_OWNER = 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.

10.4.6.3 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 valid state to invalid state 0: Disable 1: Enable
12	R/W	0x0	TC_INT_EN Transfer Completed Interrupt Enable 0: Disable 1: Enable
11	R/W	0x0	TF_UDR_INT_EN TXFIFO 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	/	/	/
2	R/W	0x0	RF_FUL_INT_EN RX FIFO Full Interrupt Enable 0: Disable 1: Enable
1	R/W	0x0	RX_EMP_INT_EN RX FIFO Empty Interrupt Enable 0: Disable 1: Enable
0	R/W	0x0	RF_RDY_INT_EN RX FIFO Ready Request Interrupt Enable 0: Disable 1: Enable

10.4.6.4 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 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 BC has been exchanged. In other condition, When set, this bit indicates that all the data in TXFIFO has been loaded in the Shift register, and the Shift register has shifted out all the bits. Writing 1 to this bit clears it. 0: Busy 1: Transfer completed
11	R/W1C	0x0	TF_UDF TXFIFO Underrun This bit is set when if the TXFIFO is underrun. Writing 1 to this bit clears it. 0: TXFIFO is not underrun 1: TXFIFO is underrun

10	R/W1C	0x0	TF_OVF TXFIFO Overflow This bit is set when if the TXFIFO is overflow. Writing 1 to this bit clears it. 0: TXFIFO is not overflow 1: TXFIFO is overflowed
9	R/W1C	0x0	RX_UDF RXFIFO Underrun When set, this bit indicates that RXFIFO has underrun. Writing 1 to this bit clears it.
8	R/W1C	0x0	RX_OVF RXFIFO Overflow When set, this bit indicates that RXFIFO has overflowed. Writing 1 to this bit clears it. 0: RXFIFO is available 1: RXFIFO is overflowed
7	/	/	/
6	R/W1C	0x0	TX_FULL TXFIFO Full This bit is set when if the TXFIFO is full . Writing 1 to this bit clears it. 0: TXFIFO is not Full 1: TXFIFO is Full
5	R/W1C	0x1	TX_EMP TXFIFO Empty This bit is set if the TXFIFO is empty. Writing 1 to this bit clears it. 0: TXFIFO contains one or more words. 1: TXFIFO is empty
4	R/W1C	0x1	TX_READY TXFIFO Ready 0: TX_WL > TX_TRIG_LEVEL 1: TX_WL <= TX_TRIG_LEVEL This bit is set any time if TX_WL <= TX_TRIG_LEVEL. Writing “1” to this bit clears it. TX_WL is the water level of RXFIFO.
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 set any time if RX_WL >= RX_TRIG_LEVEL. Writing “1” to this bit clears it. RX_WL is the water level of RXFIFO.

10.4.6.5 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.
30	R/W	0x0	TF_TEST_ENB TX Test Mode Enable 0: Disable 1: Enable In normal mode, TX FIFO can only be read by SPI controller, writing ‘1’ to this bit will switch TX FIFO read and write function to AHB bus. This bit is used to test the TX FIFO, donot set in normal operation and donot set RF_TEST and TF_TEST at the same time.
29:25	/	/	/
24	R/W	0x0	TF_ DRQ_EN TX FIFO DMA Request Enable 0: Disable 1: Enable
23:16	R/W	0x40	TX_TRIG_LEVEL TX FIFO Empty Request Trigger Level
15	R/WAC	0x0	RF_RST RXFIFO Reset 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, RX FIFO can only be written by SPI controller, writing ‘1’ to this bit will switch RX FIFO read and write function to AHB bus. This bit is used to test the RX FIFO, donot set in normal operation and donot 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

10.4.6.6 SPI FIFO Status Register(Default Value: 0x0000_0000)

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

10.4.6.7 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). 0: No wait states inserted

			n: n SPI_SCLK wait states inserted These bits control the number of wait states to be inserted before start dual data transfer in dual SPI mode. The SPI module counts SPI_SCLK by SWC for delaying next word data transfer. 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 next word data transfer. 0: No wait states inserted N: N SPI_SCLK wait states inserted

10.4.6.8 SPI Clock Control Register(Default Value: 0x0000_0002)

Offset: 0x0024			Register Name: SPI_CCR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:13	/	/	/
12	R/W	0x0	DRS Divide Rate Select (Master Mode Only) 0: Select Clock Divide Rate 1 1: Select Clock Divide Rate 2
11:8	R/W	0x0	CDR1_M Clock Divide Rate 1 (Master Mode Only) The SPI_SCLK is determined according to the following equation: SPI_CLK = Source_CLK / (2^CDR1_M).
7:0	R/W	0x2	CDR2_N Clock Divide Rate 2 (Master Mode Only) The SPI_SCLK is determined according to the following equation: SPI_CLK = Source_CLK / (2*(CDR2_N + 1)).

10.4.6.9 SPI Master Burst Counter Register(Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: SPI_MBC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	/	/	/
23:0	R/W	0x0	MBC Master Burst Counter In master mode, this field specifies the total burst number. 0: 0 burst 1: 1 burst … N: N bursts Total transfer data, include the TXD, RXD and dummy burst.

10.4.6.10 SPI Master Transmit Counter Register(Default Value: 0x0000_0000)

Offset: 0x0034			Register Name: SPI_MTC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	/	/	/
23:0	R/W	0x0	MWTC Master Write Transmit Counter In master mode, this field specifies the burst number that should be sent to TXFIFO before automatically sending dummy burst. For saving bus bandwidth, the dummy burst (all zero bits or all one bits) is sent by SPI Controller automatically. 0: 0 burst 1: 1 burst … N: N bursts

10.4.6.11 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 0: Quad mode disable 1: Quad mode enable Quad mode includes Quad-Input and Quad-Output.
28	R/W	0x0	DRM Master Dual Mode RX Enable 0: RX use single-bit mode 1: RX use dual mode Cannot be written when XCH=1; It is only valid when Quad_Mode_EN=0.
27:24	R/W	0x0	DBC Master Dummy Burst Counter In master mode, this field specifies the burst number that should be sent before receive in dual SPI mode. The data 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 single mode before automatically sending dummy burst. This is the first transmit counter in all bursts.

			0: 0 burst 1: 1 burst … N: N bursts Cannot be written when XCH=1

10.4.6.12 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.
30	R/W	0x0	MSMS Master Sample Standard 1: Standard Sample Mode 0: Delay Sample Mode In Standard Sample Mode, SPI master samples the data at the standard rising edge of SCLK for each SPI mode; In Delay Sample Mode, SPI master samples data at the edge that is half cycle delayed by the standard rising edge of 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: 0bit 000001: 1bit

			… 100000: 32bits Other values: reserved It is only valid when ***Work Mode Select==0x10/0x11,*** and cannot be written when TCE=1.
15:14	/	/	/
13:8	R/W	0x00	Configure the length of serial data frame(burst) of TX 000000: 0bit 000001: 1bit … 100000: 32bits 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, controller sends SS signal automatically with data together. When this bit is set to 1, software must manually write SPI_CTL_REG.SS_LEVEL to 1 or 0 to control the level of 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	/	/	/
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

10.4.6.13 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***.

10.4.6.14 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***.

10.4.6.15 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***.

10.4.6.16 SPI Normal DMA Mode Control Register(Default Value: 0x0000_00E5)

Offset: 0x0088			Register Name: NDFC_NDMA_MODE_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**

31:8	/	/	/
7:6	R/W	0x11	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	0: active fall do not care ack 1: active fall must after detect ack is high
4:0	R/W	0x05	Delay Cycles The counts of hold cycles from DMA last signal high to dma_active high

10.4.6.17 SPI TX Data Register(Default Value: 0x0000_0000)

Offset: 0x0200			Register Name: SPI_TXD
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R/W	0x0	TDATA Transmit Data This register can be accessed in byte, half-word or word unit by AHB. In byte accessing method, if there are rooms in TXFIFO, one burst data is written to TXFIFO and the depth is increased by 1. In half-word accessing method, two SPI burst data are written and the TXFIFO depth is increased by 2. In word accessing method, four SPI burst data are written and the TXFIFO depth is increased by 4. **NOTE** **This address is 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.**

10.4.6.18 SPI RX Data Register(Default Value: 0x0000_0000)

Offset: 0x0300			Register Name: SPI_RXD
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	RDATA Receive Data This register can be accessed in byte, half-word or word unit by AHB. In byte accessing method, if there are data in RXFIFO, the top word is returned and the RXFIFO depth is decreased by 1. In half-word accessing method, two SPI bursts are returned and the RXFIFO depth is decreased by 2. In 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.**

10.5 USB2.0 OTG

10.5.1 Overview

The USB2.0 OTG is a Dual-Role Device controller, which supports both device and host functions which can also be configured as a Host-only or Device-only controller, fully compliant with the USB 2.0 Specification. It can support 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 can support high-speed (HS, 480 Mbit/s), and full-speed (FS, 12 Mbit/s) in Device mode. Standard USB transceiver can be used through its UTMI+PHY Level3 interface. The UTMI+PHY interface is bidirectional with 8-bit data bus. For saving CPU bandwidth, USB-OTG DMA interface can support external DMA controller to take care of the data transfer between the memory and USB-OTG FIFO. The USB-OTG core also supports USB power saving functions.

Features:

- Complies 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
- Supports the UTMI+ Level 3 interface. The 8-bit bidirectional data buses are used
- Supports bi-directional endpoint0 for Control transfer
- Supports up to 10 User-Configurable Endpoints for Bulk, Isochronousl and Interrupt bi-directional transfers (Endpoint1, Endpoint2, Endpoint3, Endpoint4, Endpoint5)
- Supports up to (8KB+64Bytes) FIFO for 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 mode
- 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 EPs

10.5.2 Block Diagram

Figure 10-25 shows the block diagram of USB OTG Controller.

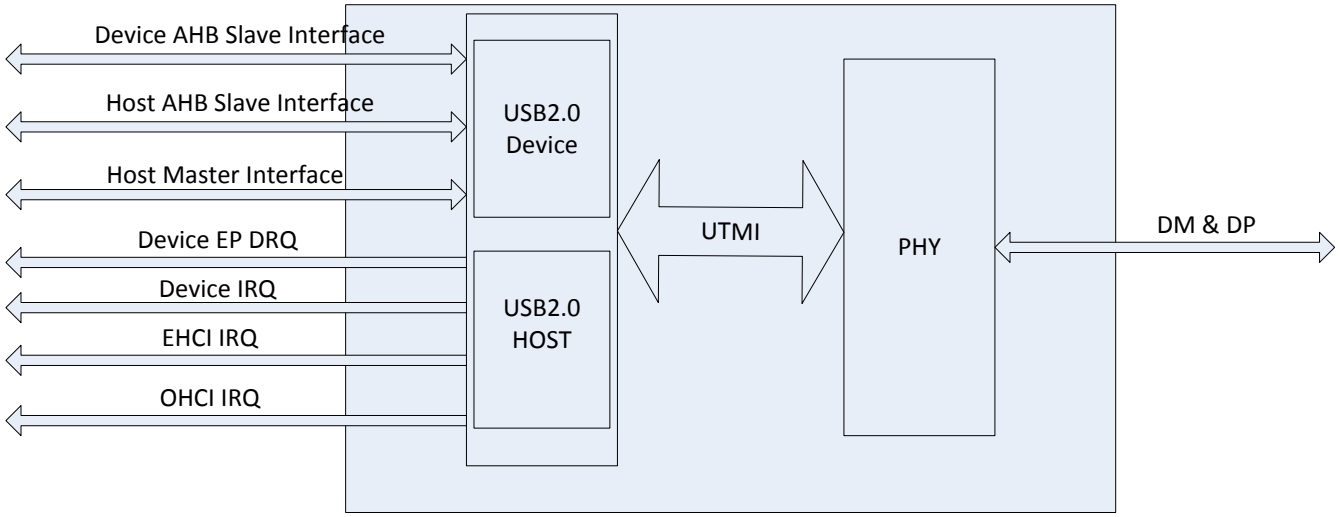


Figure10- 25. USB OTG Controller Block Diagram

10.5.3 External Signals

Table10- 13. USB2.0 OTG External Signals

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

10.6 USB2.0 Host Controller

10.6.1 Overview

USB2.0 Host Controller is fully compliant with the USB 2.0 specification, Enhanced Host Controller Interface (EHCI) Specification, Revision 1.0, and the Open Host Controller Interface (OHCI) Specification Release 1.0a. The controller supports high-speed, 480 Mbit/s transfers (40 times faster than USB 1.1 full-speed mode) using an EHCI Host Controller, as well as full and low speeds through one or more integrated OHCI Host Controllers.

The USB2.0 Host controller includes the following features:

- 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
- Including an internal DMA Controller for data transfer with memory
- Complies with Enhanced Host Controller Interface (EHCI) Specification, Version 1.0, and the Open Host Controller Interface (OHCI) Specification, Version 1.0a
- Supports High-Speed (HS, 480 Mbit/s), Full-Speed (FS, 12 Mbit/s), and Low-Speed (LS, 1.5 Mbit/s) Device
- Supports the UTMI+ Level 3 interface. The 8-bit bidirectional data buses are used
- Supports only 1 USB Root Port shared between EHCI and OHCI

10.6.2 Block Diagram

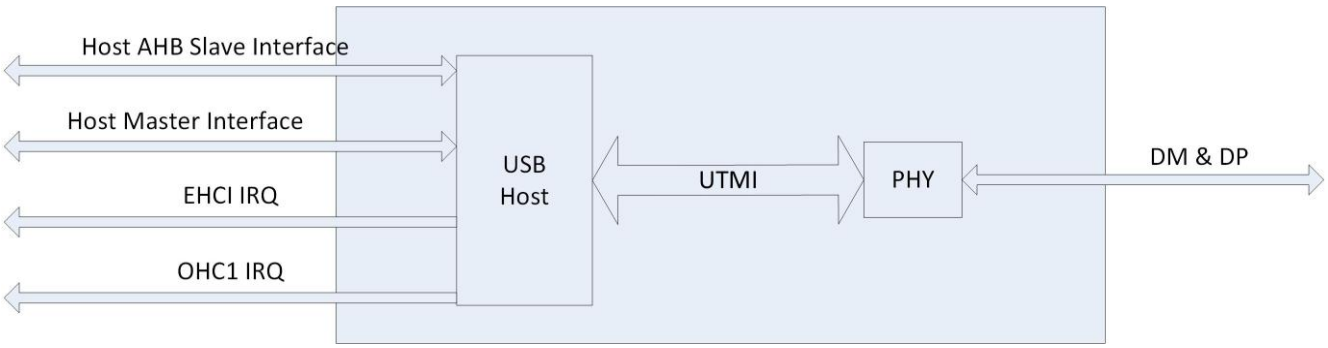


Figure10- 26. USB2.0 Host Controller Block Diagram

10.6.3 Operations and Functional Descriptions

10.6.3.1 External Signals

Table10- 14. USB2.0 Host External Signals

Signal	Description	Type
USB1-DP	USB2.0 Host differential signal postive	AI/O

USB1-DM	USB2.0 Host differential signal negative	AI/O

10.6.3.2 Clock and Reset

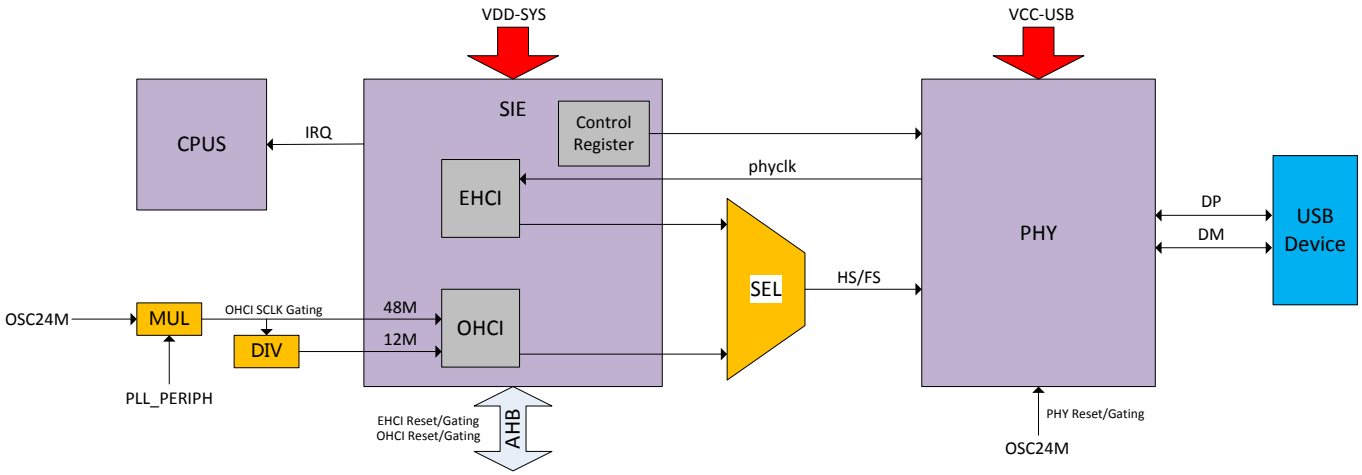


Figure10- 27. USB2.0 Host Clock Description

10.6.3.3 Function Implementation

Please refer to USB2.0 Specification, Enhanced Host Controller Interface (EHCI) Specification, Version 1.0, and the Open Host Controller Interface (OHCI) Specification, Version 1.0a.

10.6.4 Register List

Module Name	Base Address
USB2.0_HOST	0x05200000

Register Name	Offset	Description
EHCI Capability Register		
E_CAPLENGTH	0x000	EHCI Capability register Length Register
E_HCIVERSION	0x002	EHCI Host Interface Version Number Register
E_HCSPARAMS	0x004	EHCI Host Control Structural Parameter Register
E_HCCPARAMS	0x008	EHCI Host Control Capability Parameter Register
E_HCSPPORTROUTE	0x00C	EHCI Companion Port Route Description
EHCI Operational Register		
E_USBCMD	0x010	EHCI USB Command Register
E_USBSTS	0x014	EHCI USB Status Register
E_USBINTR	0x018	EHCI USB Interrupt Enable Register
E_FRINDEX	0x01C	EHCI USB Frame Index Register
E_CTRLDSSEGMENT	0x020	EHCI 4G Segment Selector Register
E_PERIODICLISTBASE	0x024	EHCI Frame List Base Address Register
E_ASYNCLISTADDR	0x028	EHCI Next Asynchronous List Address Register
E_CONFIGFLAG	0x050	EHCI Configured Flag Register

E_PORTSC	0x054	EHCI Port Status/Control Register
OHCI Control and Status Partition Register		
O_HcRevision	0x400	OHCI Revision Register
O_HcControl	0x404	OHCI Control Register
O_HcCommandStatus	0x408	OHCI Command Status Register
O_HcInterruptStatus	0x40C	OHCI Interrupt Status Register
O_HcInterruptEnable	0x410	OHCI Interrupt Enable Register
O_HcInterruptDisable	0x414	OHCI Interrupt Disable Register
OHCI Memory Pointer Partition Register		
O_HcHCCA	0x418	OHCI HCCA Base
O_HcPeriodCurrentED	0x41C	OHCI Period Current ED Base
O_HcControlHeadED	0x420	OHCI Control Head ED Base
O_HcControlCurrentED	0x424	OHCI Control Current ED Base
O_HcBulkHeadED	0x428	OHCI Bulk Head ED Base
O_HcBulkCurrentED	0x42c	OHCI Bulk Current ED Base
O_HcDoneHead	0x430	OHCI Done Head Base
OHCI Frame Counter Partition Register		
O_HcFmInterval	0x434	OHCI Frame Interval Register
O_HcFmRemaining	0x438	OHCI Frame Remaining Register
O_HcFmNumber	0x43C	OHCI Frame Number Register
O_HcPerioddicStart	0x440	OHCI Periodic Start Register
O_HcLSThreshold	0x444	OHCI LS Threshold Register
OHCI Root Hub Partition Register		
O_HcRhDescriptorA	0x448	OHCI Root Hub Descriptor Register A
O_HcRhDesriptorB	0x44C	OHCI Root Hub Descriptor Register B
O_HcRhStatus	0x450	OHCI Root Hub Status Register
O_HcRhPortStatus	0x454	OHCI Root Hub Port Status Register
HCI Controller and PHY Interface Register		
HCI Interface	0x800	HCI Interface Register
PHY Control	0x810	PHY Control Register
HSIC PHY tune1	0x81C	HSIC PHY Tune1 Register
HSIC PHY tune2	0x820	HSIC PHY Tune2 Register
HSIC PHY tune3	0x824	HSIC PHY Tune3 Register
HCI SIE Port Disable Control	0x828	HCI SIE Port Disable Control Register

10.6.4.1 EHCI Register Description

10.6.4.2 EHCI Identification Register(Default Value:0x10)

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

			beginning of the Operational Register Space.

10.6.4.3 EHCI Host Interface Version Number Register(Default Value:0x0100)

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

10.6.4.4 EHCI Host Control Structural Parameter Register(Default Value:0x0000_0004)

Offset: 0x0004			Register Name: HCSPARAMS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	/	/	/
23:20	R	0x0	Debug Port Number This register identifies which of the host controller ports is the debug port. The value is the port number (one based) of the debug port. This field will always be '0'.
19:16	/	/	/
15:12	R	0x0	Number of Companion Controller (N_CC) This field indicates the number of companion controllers associated with this USB2.0 host controller. A zero in this field indicates there are no companion host controllers. And a value larger than zero in this field indicates there are companion USB1.1 host controller(s). This field will always be '0'.
11:8	R	0x0	Number of Port per Companion Controller(N_PCC) This field indicates the number of ports supported per companion host controller host controller. It is used to indicate the port routing configuration to system software. This field will always fix with '0'.
7	R	0x0	Port Routing Rules This field indicates the method used by this implementation for how all ports are mapped to companion controllers. The value of this field has the following interpretation: Value: 0 — The first N_PCC ports are routed to the lowest numbered function companion host controller, the next N_PCC port are routed to the next lowest function companion controller, and so on. Value: 1 — The port routing is explicitly enumerated by the first N_PORTS elements of the HCSP-PORTTOUTE array.

			This field will always be '0'.
6:4	/	/	/
3:0	R	0x1	N_PORTS This field specifies the number of physical downstream ports implemented on this host controller. The value of this field determines how many port registers are addressable in the Operational Register Space. Valid values are in the range of 0x1 to 0x0f. This field is always 1.

10.6.4.5 EHCI Host Control Capability Parameter Register(Default Value:0x0000_0008)

Offset: 0x0008			**Register Name: HCCPARAMS**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:8	R	0x0	EHCI Extended Capabilities Pointer (EECP) This optional field indicates the existence of a capabilities list. A value of 00b indicates no extended capabilities are implemented. A non-zero value in this register indicates the offset in PCI configuration space of the first EHCI extended capabiliby. The pointer value must be 40h or greater if implemented to maintain to consistency of the PCI header defined for this calss of device. The value of this field is always '00b'.
7:4	R	0x0	Isochronous Scheduling Threshold This field indicates, relative to the current position of the executing host controller, where software can reliably update the isochronous schedule. When bit[7] is zero, the value of the least significant 3 bits indicates the number of micro-frames a host controller can hold a set of isochronous data structures(one or more) before flushing the state. When bit[7] is a one, then host software assumes the host controller may cache an isochronous data structure for an entire frame.
3	/	/	/
2	R	0x0	Asynchronous Schedule Park Capability If this bit is set to a one, then the host controller supports the park feature for high-speed queue heads in the Asynchronous Schedule. The feature can be disabled or enabled and set to a specific level by using the Asynchronous Schedule Park Mode Enable and Asynchronous Schedule Park Mode Count fields in the USBCMD register.
1	R	0x0	Programmable Frame List Flag If this bit is set to a zero, then system software must use a frame list length of 1024 elements with this host controller. The USBCMD register Frame List Size field is a read-only register and should be set to zero. If set to 1, then system software can specify and use the frame list in the USBCMD register Frame List Size field to cofigure the host controller. The frame list must always aligned on a 4K page boundary. This requirement ensures that the frame list is always physically contiguous.

0	/	/	/

10.6.4.6 EHCI Companion Port Route Description(Default Value:0x0000_0000)

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

10.6.4.7 EHCI USB Command Register(Default Value:0x0008_0000)

Offset: 0x0010			Register Name: USBCMD
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	/	/	/
23:16	R/W	0x08	Interrupt Threshold Control The value in this field is used by system software to select the maximum rate at which the host controller will issue interrupts. The only valid values are defined below: Value | Minimum Interrupt Interval 0x00 | Reserved 0x01 | 1 micro-frame 0x02 | 2 micro-frame 0x04 | 4 micro-frame 0x08 | 8 micro-frame(default, equates to 1 ms) 0x10 | 16 micro-frame(2ms) 0x20 | 32 micro-frame(4ms) 0x40 | 64 micro-frame(8ms) Any other value in this register yields undefined results. The default value in this field is 0x08 . Software modifications to this bit while HC Halted bit is equal to zero results in undefined behavior.

15:12	/	/	/
11	R	0x0	Asynchronous Schedule Park Mode Enable(OPTIONAL) If the Asynchronous Park Capability bit in the HCCPARAMS register is a one, then this bit defaults to a 1 and is R/W. Otherwise the bit must be a zero and is Read Only. Software uses this bit to enable or disable Park mode. When this bit is one, Park mode is enabled. When this bit is zero, Park mode is disabled.
10	/	/	/
9:8	R	0x0	Asynchronous Schedule Park Mode Count(OPTIONAL) Asynchronous Park Capability bit in the HCCPARAMS register is a one, Then this field defaults to 0x3 and is W/R. Otherwise it defaults to zero and is R. It contains a count of the number of successive transactions the host controller is allowed to execute from a high-speed queue head on the Asynchronous schedule before continuing traversal of the Asynchronous schedule. Valid value are 0x1 to 0x3.Software must not write a zero to this bit when Park Mode Enable is a one as it will result in undefined behavior.
7	R/W	0x0	Light Host Controller Reset(OPTIONAL) This control bit is not required. If implemented, it allows the driver to reset the EHCI controller without affecting the state of the ports or relationship to the companion host controllers. For example, the PORSTC registers should not be reset to their default values and the CF bit setting should not go to zero (retaining port ownership relationships). A host software read of this bit as zero indicates the Light Host Controller Reset has completed and it si safe for software to re-initialize the host controller. A host software read of this bit as a one indicates the Light Host
6	R/W	0x0	Interrupt on Async Advance Doorbell This bit is used as a doorbell by software to tell the host controller to issue an interrupt the next time it advances asynchronous schedule. Soft- Ware must write a 1 to this bit to ring the doorbell. When the host controller has evicted all appropriate cached schedule state, it sets the Interrupt on Async Advance status bit in the USBSTS. if the Interrupt on Async Advance Enable bit in the USBINTR register is a one then the host controller will assert an interrupt at the next interrupt threshold. The host controller sets this bit to a zero after it has set the Interrupt on Async Advance status bit in the USBSTS register to a one. Software should not write a one to this bit when the asynchronous schedule is disabled. Doing so will yield undefined results.
5	R/W	0x0	Asynchronous Schedule Enable This bit controls whether the host controller skips processing the Asynchronous Schedule. Values mean: Bit Value: 0 — Do not process the Asynchronous Schedule. Bit Value: 1 — Use the ASYNLISTADDR register to access the Asynchronous Schedule.

<table>
<tr><td></td><td></td><td></td><td>The default value of this field is ‘0b’.</td></tr>
<tr><td>4</td><td>R/W</td><td>0x0</td><td>Periodic Schedule Enable
This bit controls whether the host controller skips processing the Periodic Schedule. Values mean:<table><tr><td>Bit Value</td><td>Meaning</td></tr><tr><td>0</td><td>Do not process the Periodic Schedule.</td></tr><tr><td>1</td><td>Use the PERIODICLISTBASE register to access the Periodic Schedule.</td></tr></table>The default value of this field is ‘0b’.</td></tr>
<tr><td>3:2</td><td>R/W</td><td>0x0</td><td>Frame List Size
This field is R/W only if Programmable Frame List Flag in the HCCPARAMS registers is set to a one. This field specifies the size of the
Frame list. The size the frame list controls which bits in the Frame Index Register should be used for the Frame List Current index. Values mean:<table><tr><td>Bits</td><td>Meaning</td></tr><tr><td>00b</td><td>1024 elements(4096bytes)Default value</td></tr><tr><td>01b</td><td>512 elements(2048byts)</td></tr><tr><td>10b</td><td>256 elements(1024bytes)For resource-constrained condition</td></tr><tr><td>11b</td><td>reserved</td></tr></table>The default value is ‘00b’.</td></tr>
<tr><td>1</td><td>R/W</td><td>0x0</td><td>Host Controller Reset
This control bit is used by software to reset the host controller. The effects of this on Root Hub registers are similar to a Chip Hardware Reset.
When software writes a one to this bit, the Host Controller resets its internal pipelines, timers, counters, state machines, etc. to their initial value. Any transaction currently in progress on USB is immediately terminated. A USB reset is not driven on downstream ports.
All operational registers, including port registers and port state machines are set to their initial values. Port ownership reverts to the companion host controller(s). Software must reinitialize the host controller as described in Section 4.1 of the CHEI Specification in order to return the host controller to an operational state.
This bit is set to zero by the Host Controller when the reset process is complete. Software cannot terminate the reset process early by writing a zero to this register.
Software should not set this bit to a one when the HC Halted bit in the USBSTS register is a zero. Attempting to reset an actively running host controller will result in undefined behavior.</td></tr>
<tr><td>0</td><td>R/W</td><td>0x0</td><td>Run/Stop
When set to a 1, the Host Controller proceeds with execution of the schedule. When set to 0, the Host Controller completes the current and any actively pipelined transactions on the USB and then halts. The Host Controller must halt within 16 micro-frames after software clears this bit.
The HC Halted bit indicates when the Host Controller has finished its pending pipelined transactions and has entered the stopped state.
Software must not write a one to this field unless the Host Controller is in</td></tr>
</table>

			the Halt State. The default value is 0x0.

10.6.4.8 EHCI USB Status Register(Default Value:0x0000_1000)

Offset: 0x0014			Register Name: USBSTS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15	R	0x0	Asynchronous Schedule Status The bit reports the current real status of Asynchronous Schedule. If this bit is a zero then the status of the Asynchronous Schedule is disabled. If this bit is a one then the status of the Asynchronous Schedule is enabled. The Host Controller is not required to immediately disable or enable the Asynchronous Schedule when software transitions the Asynchronous Schedule Enable bit in the USBCMD register. When this bit and the Asynchronous Schedule Enable bit are the same value, the Asynchronous Schedule is either enabled (1) or disabled (0).
14	R	0x0	Periodic Schedule Status The bit reports the current real status of the Periodic Schedule. If this bit is a zero then the status of the Periodic Schedule is disabled. If this bit is a one then the status of the Periodic Schedule is enabled. The Host Controller is not required to *immediately* disable or enable the Periodic Schedule when software transitions the *Periodic Schedule Enable* bit in the USBCMD register. When this bit and the *Periodic Schedule Enable* bit are the same value, the Periodic Schedule is either enabled (1) or disabled (0).
13	R	0x0	Reclamation This is a read-only status bit, which is used to detect an empty asynchronous schedule.
12	R	0x1	HC Halted This bit is a zero whenever the Run/Stop bit is a one. The Host Controller Sets this bit to one after it has stopped executing as a result of the Run/Stop bit being set to 0, either by software or by the Host Controller Hardware (e.g. internal error). The default value is '1'.
11:6	/	/	/
5	R/W1C	0x0	Interrupt on Async Advance System software can force the host controller to issue an interrupt the next time the host controller advances the asynchronous schedule by writing a one to the Interrupt on Async Advance Doorbell bit in the USBCMD register. This status bit indicates the assertion of that interrupt source.
4	R/W1C	0x0	Host System Error The Host Controller set this bit to 1 when a serious error occurs during a host system access involving the Host Controller module. When this error occurs, the Host Controller clears the Run/Stop bit in the Command register to prevent further execution of the scheduled TDs.

3	R/W1C	0x0	Frame List Rollover The Host Controller sets this bit to a one when the Frame List Index rolls over from its maximum value to zero. The exact value at which the rollover occurs depends on the frame list size. For example, if the frame list size is 1024, the Frame Index Register rolls over every time FRINDEX [13] toggles. Similarly, if the size is 512, the Host Controller sets this bit to a one every time FRINDEX [12] toggles.
2	R/W1C	0x0	Port Change Detect The Host Controller sets this bit to a one when any port for which the Port Owner bit is set to zero has a change bit transition from a zero to a one or a Force Port Resume bit transition from a zero to a one as a result of a J-K transition detected on a suspended port. This bit will also be set as a result of the Connect Status Chang being set to a one after system software has relinquished ownership of a connected port by writing a one to a port's Port Owner bit.
1	R/W1C	0x0	USB Error Interrupt(USBERRINT) The Host Controller sets this bit to 1 when completion of USB transaction results in an error condition(e.g. error counter underflow).If the TD on which the error interrupt occurred also had its IOC bit set, both. This bit and USBINT bit are set.
0	R/W1C	0x0	USB Interrupt(USBINT) The Host Controller sets this bit to a one on the completion of a USB transaction, which results in the retirement of a Transfer Descriptor that had its IOC bit set. The Host Controller also sets this bit to 1 when a short packet is detected (actual number of bytes received was less than the expected number of bytes)

10.6.4.9 EHCI USB Interrupt Enable Register(Default Value:0x0000_0000)

Offset: 0x0018			**Register Name: USBINTR**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:6	/	/	/
5	R/W	0x0	Interrupt on Async Advance Enable When this bit is 1, and the Interrupt on Async Advance bit in the USBSTS register is 1, the host controller will issue an interrupt at the next interrupt threshold. The interrupt is acknowledged by software clearing the Interrupt on Async Advance bit.
4	R/W	0x0	Host System Error Enable When this bit is 1, and the Host System Error Status bit in the USBSTS register is 1, the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Host System Error bit.
3	R/W	0x0	Frame List Rollover Enable When this bit is 1, and the Frame List Rollover bit in the USBSTS register is 1, the host controller will issue an interrupt. The interrupt is acknowledged by

			software clearing the Frame List Rollover bit.
2	R/W	0x0	Port Change Interrupt Enable When this bit is 1, and the Port Chang Detect bit in the USBSTS register is 1, the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Port Chang Detect bit.
1	R/W	0x0	USB Error Interrupt Enable When this bit is 1, and the USBERRINT bit in the USBSTS register is 1,the host controller will issue an interrupt at the next interrupt threshold. The interrupt is acknowledged by software clearing the USBERRINT bit.
0	R/W	0x0	USB Interrupt Enable When this bit is 1, and the USBINT bit in the USBSTS register is 1,the host controller will issue an interrupt at the next interrupt threshold. The interrupt is acknowledged by software clearing the USBINT bit.

10.6.4.10 EHCI Frame Index Register(Default Value:0x0000_0000)

Offset: 0x001C			**Register Name: FRINDEX**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:14	/	/	/
13:0	R/W	0x0	Frame Index The value in this register increment at the end of each time frame (e.g. micro-frame).Bits[N:3] are used for the Frame List current index. It means that each location of the frame list is accessed 8 times(frames or Micro-frames) before moving to the next index. The following illustrates Values of N based on the value of the Frame List Size field in the USBCMD register.

USBCMD[Frame List Size]	Number Elements	N
00b	1024	12
01b	512	11
10b	256	10
11b	Reserved	

NOTE

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

10.6.4.11 EHCI Periodic Frame List Base Address Register(Default Value:0x0000_0000)

Offset: 0x0024			**Register Name: PERIODICLISTBASE**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:12	R/W	0x0	Base Address These bits correspond to memory address signals [31:12], respectively. This register contains the beginning address of the Periodic Frame List in the system memory. System software loads this register prior to starting the schedule execution

			by the Host Controller. The memory structure referenced by this physical memory pointer is assumed to be 4K byte aligned. The contents of this register are combined with the Frame Index Register (FRINDEX) to enable the Host Controller to step through the Periodic Frame List in sequence.
11:0	/	/	/

NOTE

Writes must be Dword Writes.

10.6.4.12 EHCI Current Asynchronous List Address Register(Default Value:0x0000_0000)

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

NOTE

Write must be DWord Writes.

10.6.4.13 EHCI Configure Flag Register(Default Value:0x0000_0000)

Offset: 0x0050			Register Name: CONFIGFLAG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	Configure Flag(CF) Host software sets this bit as the last action in its process of configuring the Host Controller. This bit controls the default port-routing control logic as follow: Value: 0 — Port routing control logic default-routs each port to an implementation dependent classic host controller. Value: 1 — Port routing control logic default-routs all ports to this host controller. The default value of this field is '0'.

NOTE

This register is not used in the normal implementation.

10.6.4.14 EHCI Port Status and Control Register(Default Value:0x0000_2000)

<table>
<tr><td colspan="3">Offset: 0x0054</td><td>Register Name: PORTSC</td></tr>
<tr><td>Bit</td><td>Read/Write</td><td>Default/Hex</td><td>Description</td></tr>
<tr><td>31:22</td><td>/</td><td>/</td><td>/</td></tr>
<tr><td>21</td><td>R/W</td><td>0x0</td><td>Wake on Disconnect Enable(WKDSCNNT_E)
Writing this bit to a one enables the port to be sensitive to device disconnects as wake-up events.
This field is zero if Port Power is zero.
The default value in this field is ‘0’.</td></tr>
<tr><td>20</td><td>R/W</td><td>0x0</td><td>Wake on Connect Enable(WKCNNT_E)
Writing this bit to a one enable the port to be sensitive to device connects as wake-up events.
This field is zero if Port Power is zero.
The default value in this field is ‘0’.</td></tr>
<tr><td>19:16</td><td>R/W</td><td>0x0</td><td>Port Test Control
The value in this field specifies the test mode of the port. The encoding of the test mode bits are as follow:
<table>
<tr><td>Bits</td><td>Test Mode</td></tr>
<tr><td>0000b</td><td>The port is NOT operating in a test mode.</td></tr>
<tr><td>0001b</td><td>Test J_STATE</td></tr>
<tr><td>0010b</td><td>Test K_STATE</td></tr>
<tr><td>0011b</td><td>Test SE0_NAK</td></tr>
<tr><td>0100b</td><td>Test Packet</td></tr>
<tr><td>0101b</td><td>Test FORCE_ENABLE</td></tr>
<tr><td>0110b-1111b</td><td>Reserved</td></tr>
</table>
The default value in this field is ‘0000b’.</td></tr>
<tr><td>15:14</td><td>/</td><td>/</td><td>/</td></tr>
<tr><td>13</td><td>R/W</td><td>0x1</td><td>Port Owner
This bit unconditionally goes to a 0b when the Configured bit in the CONFIGFLAG register makes a 0b to 1b transition. This bit unconditionally goes to 1b whenever the Configured bit is zero.
System software uses this field to release ownership of the port to selected host controller (in the event that the attached device is not a high-speed device).Software writes a one to this bit when the attached device is not a high-speed device. A one in this bit means that a companion host controller owns and controls the port.
Default Value = 1b.</td></tr>
<tr><td>12</td><td>/</td><td>/</td><td>/</td></tr>
<tr><td>11:10</td><td>R</td><td>0x0</td><td>Line Status
These bits reflect the current logical levels of the D+ (bit11) and D-(bit10) signal lines. These bits are used for detection of low-speed USB devices prior to port reset and enable sequence. This read only field is valid only when the port enable bit is zero and the current connect status bit is set to a one.</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>The encoding of the bits are:
<table>
<tr><td>Bit[11:10]</td><td>USB State</td><td>Interpretation</td></tr>
<tr><td>00b</td><td>SE0</td><td>Not Low-speed device, perform EHCI reset.</td></tr>
<tr><td>10b</td><td>J-state</td><td>Not Low-speed device, perform EHCI reset.</td></tr>
<tr><td>01b</td><td>K-state</td><td>Low-speed device, release ownership of port.</td></tr>
<tr><td>11b</td><td>Undefined</td><td>Not Low-speed device, perform EHCI reset.</td></tr>
</table>
This value of this field is undefined if Port Power is zero.</td></tr>
<tr><td>9</td><td>/</td><td>/</td><td>/</td></tr>
<tr><td>8</td><td>R/W</td><td>0x0</td><td>Port Reset
1=Port is in Reset. 0=Port is not in Reset. Default value = 0.
When software writes a one to this bit (from a zero), the bus reset sequence as defined in the USB Specification Revision 2.0 is started. Software writes a zero to this bit to terminate the bus reset sequence. Software must keep this bit at a one long enough to ensure the reset sequence, as specified in the USB Specification Revision 2.0, completes.
When software writes this bit to a one , it must also write a zero to the Port Enable bit.
Note that when software writes a zero to this bit there may be a delay before the bit status changes to a zero. The bit status will not read as a zero until after the reset has completed. If the port is in high-speed mode after reset is complete, the host controller will automatically enable this port (e.g. set the Port Enable bit to a one). A host controller must terminate the reset and stabilize the state of the port within 2 milliseconds of software transitioning this bit from a one to a zero. For example: if the port detects that the attached device is high-speed during reset, then the host controller must have the port in the enabled state with 2ms of software writing this bit to a zero.
The HC Halted bit in the USBSTS register should be a zero before software attempts to use this bit. The host controller may hold Port Reset asserted to a one when the HC Halted bit is a one.
This field is zero if Port Power is zero.</td></tr>
<tr><td>7</td><td>R/W</td><td>0x0</td><td>Suspend
Port Enabled Bit and Suspend bit of this register define the port states as follows:
<table>
<tr><td>Bits[Port Enables, Suspend]</td><td>Port State</td></tr>
<tr><td>0x</td><td>Disable</td></tr>
<tr><td>10</td><td>Enable</td></tr>
<tr><td>11</td><td>Suspend</td></tr>
</table>
When in suspend state, downstream propagation of data is blocked on this port, except for port reset. The blocking occurs at the end of the current transaction, if a transaction was in progress when this bit was written to 1. In the suspend state, the port is sensitive to resume detection. Not that the bit</td></tr>
</table>

			status does not change until the port is suspend and that there may be a delay in suspending a port if there is a transaction currently in progress on the USB. A write of zero to this bit is ignored by the host controller. The host controller will unconditionally set this bit to a zero when: ① Software sets the Force Port Resume bit to a zero(from a one). ② Software sets the Port Reset bit to a one(from a zero). If host software sets this bit to a one when the port is not enabled(i.e. Port enabled bit is a zero), the results are undefined. This field is zero if Port Power is zero. The default value in this field is '0'.
6	R/W	0x0	Force Port Resume 1 = Resume detected/driven on port. 0 = No resume (K-state) detected/ driven on port. Default value = 0. This functionality defined for manipulating this bit depends on the value of the Suspend bit. For example, if the port is not suspend and software transitions this bit to a one, then the effects on the bus are undefined. Software sets this bit to a 1 drive resume signaling. The Host Controller sets this bit to a 1 if a J-to-K transition is detected while the port is in the Suspend state. When this bit transitions to a one because a J-to-K transition is detected, the Port Change Detect bit in the USBSTS register is also set to a one. If software sets this bit to a one, the host controller must not set the Port Change Detect bit. Note that when the EHCI controller owns the port, the resume sequence follows the defined sequence documented in the USB Specification Revision 2.0. The resume signaling (Full-speed 'K') is driven on the port as long as this remains a one. Software must appropriately time the Resume and set this bit to a zero when the appropriate amount of time has elapsed. Writing a zero (from one) causes the port to return high-speed mode (forcing the bus below the port into a high-speed idle). This bit will remain a one until the port has switched to high-speed idle. The host controller must complete this transition within 2 milliseconds of software setting this bit to a zero. This field is zero if Port Power is zero.
5	R/W1C	0x0	Over-current Change Default = 0. This bit gets set to a one when there is a change to Over-current Active. Software clears this bit by writing a one to this bit position.
4	R	0x0	Over-current Active 0 = This port does not have an over-current condition. 1 = This port currently has an over-current condition. This bit will automatically transition from a one to a zero when the over current condition is removed. The default value of this bit is '0'.
3	R/W1C	0x0	Port Enable/Disable Change Default = 0. 1 = Port enabled/disabled status has changed. 0 = No change. For the root hub, this bit gets set to a one only when a port is disabled due to the appropriate conditions existing at the EOF2 point (See Chapter 11 of the USB Specification for the definition of a Port Error). Software clears this bit by

			writing a 1 to it. This field is zero if Port Power is zero.
2	R/W	0x0	Port Enabled/Disabled 1=Enable, 0=Disable. Ports can only be enabled by the host controller as a part of the reset and enable. Software cannot enable a port by writing a one to this field. The host controller will only set this bit to a one when the reset sequence determines that the attached device is a high-speed device. Ports can be disabled by either a fault condition(disconnect event or other fault condition) or by host software. Note that the bit status does not change until the port state actually changes. There may be a delay in disabling or enabling a port due to other host controller and bus events. When the port is disabled, downstream propagation of data is blocked on this port except for reset. The default value of this field is '0'. This field is zero if Port Power is zero.
1	R/W1C	0x0	Connect Status Change 1=Change in Current Connect Status, 0=No change, Default=0. Indicates a change has occurred in the port's Current Connect Status. The host controller sets this bit for all changes to the port device connect status, even if system software has not cleared an existing connect status change. For example, the insertion status changes twice before system software has cleared the changed condition, hub hardware will be "setting" an already-set bit. Software sets this bit to 0 by writing a 1 to it. This field is zero if Port Power is zero.
0	R	0x0	Current Connect Status Device is present on port when the value of this field is a one, and no device is present on port when the value of this field is a zero. This value reflects the current state of the port, and may not correspond directly to the event that caused the Connect Status Change(Bit 1) to be set. This field is zero if Port Power zero.

NOTE

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

10.6.5 OHCI Register Description

10.6.5.1 HcRevision Register(Default Value:0x10)

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.

10.6.5.2 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 00b USBReset 01b USBResume 10b USBOperational 11b USBSuspend 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.
5	R/W	R	0x0	BulkListEnable

<table>
<tr><td></td><td></td><td></td><td></td><td>This bit is set to enable the processing of the Bulk list in the next Frame. If cleared by HCD, processing of the Bulk list does not occur after the next SOF. HC checks this bit whenever it determines to process the list. When disabled, HCD may modify the list. If HcBulkCurrentED 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, 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
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>

10.6.5.3 HcCommandStatus Register(Default Value:0x0000_0000)

Offset: 0x0408			Register Name: HcCommandStatus
Bit	**Read/Write**	**Default/Hex**	**Description**

	HCD	HC		
31:18	/	/	0x0	Reserved
17:16	R	R/W	0x0	SchedulingOverrunCount These bits are incremented on each scheduling overrun error. It is initialized to 00b and wraps around at 11b. This will be incremented when a scheduling overrun is detected even if SchedulingOverrun in *HcInterruptStatus* has already been set. This is used by HCD to monitor any persistent scheduling problem.
15:4	/	/	/	/
3	R/W	R/W	0x0	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.
2	R/W	R/W	0x0	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. 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.

10.6.5.4 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 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.

10.6.5.5 HcInterruptEnable Register(Default Value:0x0000_0000)

Offset: 0x0410				Register Name: HcInterruptEnable
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.
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;

10.6.5.6 HcInterruptDisable Register(Default Value:0x0000_0000)

Offset: 0x0414				Register Name: HcInterruptDisable
Bit	**Read/Write**		**Default**	**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;
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;

10.6.5.7 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.

10.6.5.8 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.

10.6.5.9 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.

10.6.5.10 HcControlCurrentED Register

Offset: 0x424				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 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.

10.6.5.11 HcBulkHeadED Register(Default Value:0x0000_0000)

Offset: 0x428				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.

10.6.5.12 HcBulkCurrentED Register(Default Value:0x0000_0000)

Offset: 0x42C				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 initially set to zero to indicate the end of the Bulk list.
3:0	R	R	0x0	BulkCurrentED [3:0] Because the general TD length is 16 bytes, the memory structure for the TD must be aligned to a 16-byte boundary. So the lower bits in the PCED, through bit 0 to bit 3 must be zero in this field.

10.6.5.13 HcDoneHead Register(Default Value:0x0000_0000)

Offset: 0x430				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.

10.6.5.14 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 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.

10.6.5.15 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.

10.6.5.16 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 HC reads the first ED in that Frame. After writing to HCCA, HC will set the StartofFrame in *HcInterruptStatus.*

10.6.5.17 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.

10.6.5.18 HcLSThreshold Register(Default Value:0x0000_0628)

Offset: 0x444				Register Name: HcLSThreshold
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:12	/	/	/	Reserved
11:0	R/W	R	0x0628	LSThreshold This field contains a value which is compared to the FrameRemaining field prior to initiating a Low Speed transaction. The transaction is started only if FrameRemaining ^{3} this field. The value is calculated by HCD with the consideration of transmission and setup overhead.

10.6.5.19 HcRhDescriptorA Register(Default Value:0x0200_1201)

Offset: 0x448				Register Name: HcRhDescriptorA
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31:24	R/W	R	0x2	PowerOnToPowerGoodTime[POTPGT]

<table>
<tr><td></td><td></td><td></td><td></td><td>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.</td></tr>
<tr><td>23:13</td><td>/</td><td>/</td><td>/</td><td>/</td></tr>
<tr><td>12</td><td>R/W</td><td>R</td><td>0x1</td><td>NoOverCurrentProtection
This bit describes how the overcurrent status for the Root Hub ports are reported. When this bit is cleared, the OverCurrentProtectionMode field specifies global or per-port reporting.
<table><tr><td>0</td><td>Over-current status is reported collectively for all downstream ports.</td></tr><tr><td>1</td><td>No overcurrent protection supported.</td></tr></table></td></tr>
<tr><td>11</td><td>R/W</td><td>R</td><td>0x0</td><td>OverCurrentProtectionMode
This bit describes how the overcurrent status for the Root Hub ports are reported. At reset, these fields should reflect the same mode as PowerSwitchingMode. This field is valid only if the NoOverCurrentProtection field is cleared.
<table><tr><td>0</td><td>Over-current status is reported collectively for all downstream ports.</td></tr><tr><td>1</td><td>Over-current status is reported on per-port basis.</td></tr></table></td></tr>
<tr><td>10</td><td>R</td><td>R</td><td>0x0</td><td>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.</td></tr>
<tr><td>9</td><td>R/W</td><td>R</td><td>0x1</td><td>PowerSwitchingMode
This bit is used to specify how the power switching of the Root Hub ports is controlled. It is implementation-specific. This field is only valid if the NoPowerSwitching field is cleared.
<table><tr><td>0</td><td>All ports are powered at the same time.</td></tr><tr><td>1</td><td>Each port is powered individually. This mode allows port power to be controlled by either the global switch or per-port switching. If the PortPowerControlMask bit is set, the port responds only to port power commands (Set/ClearPortPower). If the port mask is cleared, then the port is controlled only by the global power switch (Set/ClearGlobalPower).</td></tr></table></td></tr>
<tr><td>8</td><td>R/W</td><td>R</td><td>0x0</td><td>NoPowerSwithcing
These bits are used to specify whether power switching is supported or ports are always powered. It is implementation-specific. When this bit is cleared, the PowerSwitchingMode specifies global or per-port switching.
<table><tr><td>0</td><td>Ports are power switched.</td></tr><tr><td>1</td><td>Ports are always powered on when the HC is powered on.</td></tr></table></td></tr>
<tr><td>7:0</td><td>R</td><td>R</td><td>0x01</td><td>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.</td></tr>
</table>

10.6.5.20 HcRhDescriptorB Register (Default Value:0x0000_0000)

Offset: 0x44c				Register Name: HcRhDescriptorB
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.

10.6.5.21 HcRhStatus Register(Default Value:0x0000_0000)

Offset: 0x450				Register Name: HcRhStatus Register
Bit	Read/Write		Default/Hex	Description
	HCD	HC		
31	W	R	0x0	(write)ClearRemoteWakeupEnable Write a ‘1’ clears DeviceRemoteWakeupEnable. Write 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; 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.

10.6.5.22 HcRhPortStatus Register(Default Value:0x0000_0100)

Offset: 0x454				**Register Name: HcRhPortStatus**
Bit	**Read/Write**		**Default/Hex**	**Description**
	HCD	**HC**		
31:21	/	/	0x0	Reserved
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

<table>
<tr><td></td><td></td><td></td><td></td><td>a ‘0’ has no effect.<table><tr><td>0</td><td>no change in PortOverCurrentIndicator</td></tr><tr><td>1</td><td>PortOverCurrentIndicator has changed</td></tr></table></td></tr>
<tr><td>18</td><td>R/W</td><td>R/W</td><td>0x0</td><td>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.<table><tr><td>0</td><td>resume is not completed</td></tr><tr><td>1</td><td>resume completed</td></tr></table></td></tr>
<tr><td>17</td><td>R/W</td><td>R/W</td><td>0x0</td><td>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.<table><tr><td>0</td><td>no change in PortEnableStatus</td></tr><tr><td>1</td><td>change in PortEnableStatus</td></tr></table></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.<table><tr><td>0</td><td>no change in PortEnableStatus</td></tr><tr><td>1</td><td>change in PortEnableStatus</td></tr></table>
  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.<table><tr><td>0</td><td>full speed device attached</td></tr><tr><td>1</td><td>low speed device attached</td></tr></table>
(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</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td></td><td>detected. HCD sets this bit by writing SetPortPower or SetGlobalPower. HCD clears this bit by writing ClearPortPower or ClearGlobalPower. Which power control switches are enabled is determined by PowerSwitchingMode and PortPortControlMask[NumberDownstreamPort]. In global switching mode(PowerSwitchingMode=0), only Set/ClearGlobalPower controls
this bit. In per-port power switching (PowerSwitchingMode=1), if the PortPowerControlMask[NDP] bit for the port is set, only
Set/ClearPortPower commands are enabled. If the mask is not set, only Set/ClearGlobalPower commands are enabled. When port power is disabled, CurrentConnectStatus, PortEnableStatus, PortSuspendStatus, and PortResetStatus should be reset.
<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</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td></td><td>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
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’ has no effect. The CurrentConnectStatus is not affected by any write.

This bit is always read as ‘1’ when the attached device is nonremovalble(DviceRemoveable[NumberDownstreamPort]).</td></tr>
</table>

10.6.6 HCI Controller and PHY Interface Description

10.6.6.1 HCI Interface Register(Default Value:0x1000_0000)

Offset: 0x800			Register Name: USB_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:29	/	/	Reserved.
28	R	0x1	DMA Transfer Status Enable 0: Disable 1: Enable
27:26	/	/	/
25	R/W	0x0	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	R/W	0x0	Simulation Mode 1: Set PHY in a non-driving mode so the EHCI can detect device connection, this is used only for simulation 0: No effect
23:21	/	/	/
20	R/W	0x0	EHCI HS Force Set 1 to this field force the ehci enter the high speed mode during bus reset. This field only valid when the bit 1 is set.
19:13	/	/	/
12	R/W	0x0	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	0x0	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	0x0	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	0x0	AHB Master Interface Burst Type INCR4 Enable 1: Use INCR4 when appropriate 0: Do not use INCR4,use other enabled INCRX or unspecified length burst INCR
8	R/W	0x0	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	0x0	ULPI Bypass Enable 1: Enable UTMI interface, disable ULPI interface 0: Enable ULPI interface, disable UTMI interface

10.6.6.2 HCI Control 3 Register(Default Value:0x0000_0000)

Offset: 0x808			**Register Name: HCI_CTRL3**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	Reserved.
16	R/W1C	0x1	Linestate Change Detect 0: Linestate change not dected. 1: Linestate change dected. Write '1' to clear.
15:4	/	/	Reserved.
3	R/W	0x0	Remote Wakeup Enable 1: Enable 0: Disable
2	/	/	Reserved.
1	R/W	0x0	Linestate Change Interrupt Enable 1: Enable 0: Disable
0	R/W	0	Linestate Change Detect Enable 1: Enable 0: Disable

10.6.6.3 PHY Control Register(Default Value: 0x0000_0002)

Offset: 0x810			**Register Name: PHY Control**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	FSEL PHY Clock Selection 0: PHY Clock is 24MHz 1: PHY Clock is 19.2MHz
15:8	/	/	/
7	R/W	0x0	LOOPBACKENB
6	R/W	0x0	IDPULLUP
5	R/W	0x0	VBUSVLDEXT (for phy vbus) 0: Invalid 1: Valid

4	R/W	0x0	VBUSVLDEXTSEL Internal signal has been tied to '1'. This bit has no effect.
3	R/W	0x0	SIDDQ 1: Write 1 to disable phy. 0: Write 0 to enable phy.
2	R/W	0x0	COMMONONN
1:0	R/W	0x0	VATESTENB

10.6.6.4 HSIC PHY Tune1 Register(Default Value: 0x0000_0010)

Offset: 0x81C			Register Name: HSIC_PHY_Tune1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:7	/	/	/
6:5	/	/	TXRPUTUNE
5:4	R/W	0x1	TXRPDTUNE
3:0	R/W	0x0	TXSRTUNE

10.6.6.5 HSIC PHY Tune2 Register(Default Value: 0x0000_0010)

Offset: 0x820			Register Name: HSIC_PHY_Tune2
Bit	**Read/Write**	**Default/Hex**	**Description**
31	/	/	BIST_EN
30	R/W	0x0	TESTBURNIN
29	R/W	0x0	TESTDATAOUTSEL
28	R/W	0x0	TESTCLK
27:24	R/W	0x0	TESTADDR
23:16	R/W	0x0	TESTDATAIN
15:4	R/W	0x1	SIDDQ
3:0	R/W	0x0	REFCLK DIV

10.6.6.6 HSIC PHY Tune3 Register(Default Value: 0x0000_0010)

Offset: 0x824			Register Name: HSIC PHY tune3 Register
Bit	**Read/Write**	**Default/Hex**	**Description**
31:6	/	/	/
5	R/W	0x0	HSIC BIST_ERROR
4	R/W	0x0	HSIC BIST_DONE
3:2	R/W	0x0	HSIC TESTDATA OUT[3:2]
1	R/W	0x1	Non_HSIC_MODE_BIST_ERROR testdata out[1]
0	R/W	0x0	Non_HSIC_MODE_BIST_DONE testdata out[0]

10.6.6.7 HCI SIE Port Disable Control Register(Default Value:0x1000_0000)

Offset: 0x828			Register Name: USB_SPDCR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	SE0 Status This bit is set when no-se0 is detected before SOF when bit[1:0] is 10b or 11b
15:2	/	/	/
1:0	R/W	0x0	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

10.7 Port Controller

10.7.1 Overview

The Port Controller can be configured with multi-functional input/output pins. All these ports can be configured as GPIO only if multiplexed functions not used. The total 6 group external PIO interrupt sources are supported and interrupt mode can be configured by software.

The features of Port Controller are as follows:

- 8 ports(PB,PC,PD,PE,PF,PG,PH,PL)
- Software control for each signal pin
- GPIO peripheral can produce interrupt
- Pull-up/Pull-down/no-Pull register control
- Control the direction of every signal
- 4 drive strengths in each operating mode
- Up to 58 interrupts
- Configurable interrupt edges

10.7.2 Block Diagram

The block diagram of port controller is shown in Figure 10-28.

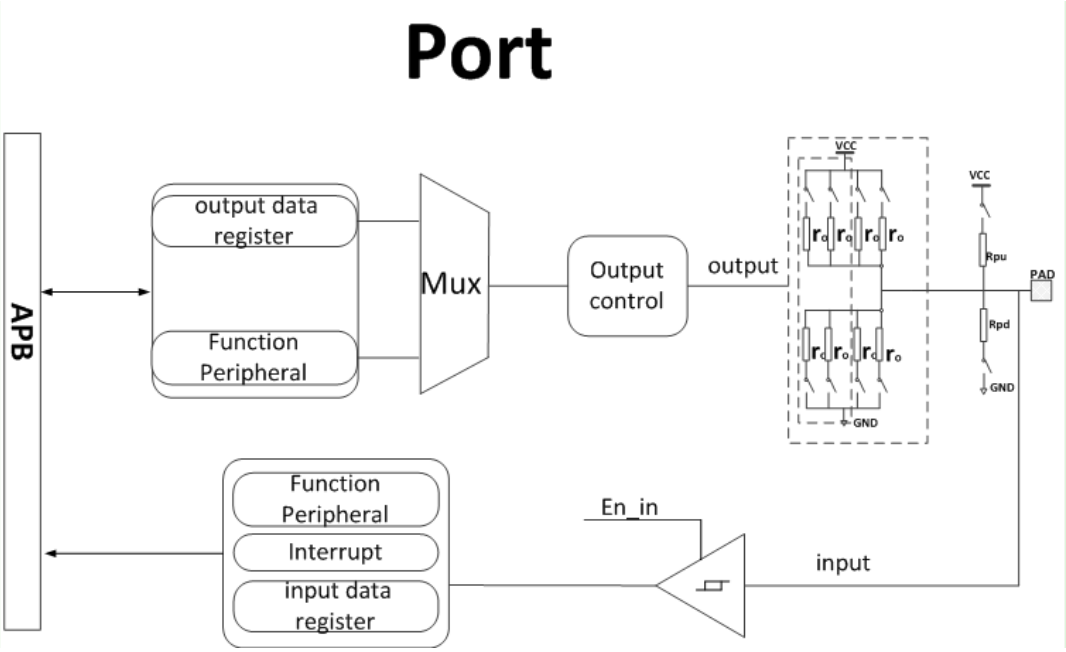


Figure10- 28. Port Controller Block Diagram

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

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

10.7.3 Operations and Functional Descriptions

10.7.3.1 Multi-function Port Table

The A50 includes 106 multi-functional input/output port pins.There are 8 ports as listed below:

Table10- 15. Multi-function Port Table

Port Name	Number of Pins	Input Driver	Output Driver	Multiplex Pins	Power
PB	11	Schmitt	CMOS	UART/JTAG/AIF/I2S/PWM	3.3V
PC	17	Schmitt	CMOS	NAND/SDC/SPI	1.8V/3.3V
PD	24	Schmitt	CMOS	RGB/LVDS/SPI/UART	1.8V/3.3V
PE	7	Schmitt	CMOS	TWI/PLL-LOCK-DBG/BIST_RESULT	1.8V/2.8/3.3V
PF	7	Schmitt	CMOS	SDC/UART/JTAG	1.8V/3.3V
PG	14	Schmitt	CMOS	SDC/UART/AIF/I2S	1.8/3.3V
PH	13	Schmitt	CMOS	TWI/UART/DMIC/SPI/CPU_CUR_W	3.3V
PL	13	Schmitt	CMOS	RSB/UART/TWI/JTAG/R_CPU_CUR_W	1.8V/3.3V

10.7.3.2 Port Functiom

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.

Table10- 16. 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 actual situation

N:: Forbid to configure

10.7.3.3 Pull up/down Logic

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

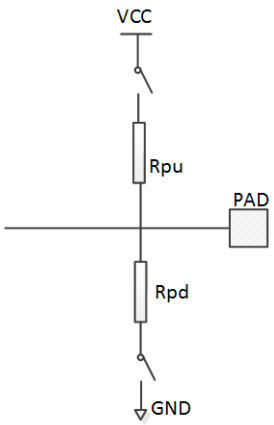


Figure10- 29. Pull up/down Logic

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

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

The pull-up/down of each IO is weak pull-up/down, the resistance is about 100kΩ.

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

10.7.3.4 Buffer Strength

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

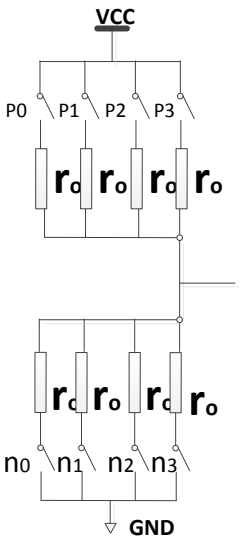


Figure10- 30. IO Buffer Strength Diagram

When output high level, the n0,n1,n2,n3 of NMOS is off, the p0,p1,p2,p3 of PMOS is on. When buffer strength is set to 0(buffer strength is weakest), only p0 is on, the output impedance is maximum ,the impedance value is r0. When buffer strength is set to 1, only p0 and p1 is on, the output impedance is equivalent to two r0 in parallel, the impedance value is r0/2.When buffer strength is 2, only 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, 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 buffer strength is set to 0(buffer strength is weakest), only n0 is on, the output impedance is maximum ,the impedance value is r0. When buffer strength is set to 1, only n0 and n1 is on, the output impedance is equivalent to two r0 in parallel, the impedance value is r0/2. When buffer strength is 2, only n0,n1 and n2 is on, the output impedance is equivalent to three r0 in parallel, the impedance value is r0/3. When buffer strength is 3, 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 output driver circuit and port is unconnected, driver configuration is invalid.

10.7.3.5 Interrupt

Each group IO has independent interrupt number.IO within group uses one interrupt number, when one IO generates interrupt, Port Controller sent interrupt request to GIC. External Interrupt Status Register is used to query which IO generates interrupt.

Interrupt trigger of GPIO supports the following trigger types.

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

External Interrupt Configure Register is used to configure trigger type.

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

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

10.7.4 CPUX Register List

Module Name	Base Address
GPIO	0x0300B000

Register Name	Offset	Description
Pn_CFG0	n*0x0024+0x00	Port n Configure Register 0 (n =1~7)
Pn_CFG1	n*0x0024+0x04	Port n Configure Register 1 (n =1~7)
Pn_CFG2	n*0x0024+0x08	Port n Configure Register 2 (n = 1~7)
Pn_CFG3	n*0x0024+0x0C	Port n Configure Register 3 (n = 1~7)
Pn_DAT	n*0x0024+0x10	Port n Data Register (n = 1~7)
Pn_DRV0	n*0x0024+0x14	Port n Multi-Driving Register 0 (n = 1~7)
Pn_DRV1	n*0x0024+0x18	Port n Multi-Driving Register 1 (n = 1~7)
Pn_PUL0	n*0x0024+0x1C	Port n Pull Register 0 (n = 1~7)
Pn_PUL1	n*0x0024+0x20	Port n Pull Register 1 (n = 1~7)
Pn_INT_CFG0	0x200+n*0x20+0x00	PIO Interrupt Configure Register 0(n =1,5,6,7)

Pn _INT_CFG1	0x200+n*0x20+0x04	PIO Interrupt Configure Register 1(n =1,5,6,7)
Pn _INT_CFG2	0x200+n*0x20+0x08	PIO Interrupt Configure Register 2(n =1,5,6,7)
Pn _INT_CFG3	0x200+n*0x20+0x0C	PIO Interrupt Configure Register 3(n =1,5,6,7)
Pn _INT_CTL	0x200+n*0x20+0x10	PIO Interrupt Control Register(n =1,5,6,7)
Pn _INT_STA	0x200+n*0x20+0x14	PIO Interrupt Status Register(n =1,5,6,7)
Pn _INT_DEB	0x200+n*0x20+0x18	PIO Interrupt Debounce Register(n =1,5,6,7)
PIO_POW_MOD_SEL	0x0340	PIO Group Withstand Voltage Mode Select Register
PIO_POW_Val	0x0348	PIO Group Power Value Register

10.7.5 CPUX Register Description

10.7.5.1 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 010:AIF2_DIN 100:Reserved 110:PB_EINT7	 001:Output 011:I2S0_DIN 101:Reserved 111:IO Disable
27	/	/	/	
26:24	R/W	0x7	PB6_SELECT 000:Input 010:AIF2_DOUT 100:Reserved 110:PB_EINT6	 001:Output 011:I2S0_DOUT 101:Reserved 111:IO Disable
23	/	/	/	
22:20	R/W	0x7	PB5_SELECT 000:Input 010:AIF2_BCLK 100:Reserved 110:PB_EINT5	 001:Output 011:I2S0_BCLK 101:Reserved 111:IO Disable
19	/	/	/	
18:16	R/W	0x7	PB4_SELECT 000:Input 010:AIF2_SYNC 100:Reserved 110:PB_EINT4	 001:Output 011:I2S0_LRCK 101:Reserved 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:Reserved 101:Reserved 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:Reserved 101:Reserved 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:Reserved 101:Reserved 111:IO Disable
3	/	/	/	
2:0	R/W	0x7	PB0_SELECT 000:Input 010:UART2_TX 100:JTAG_MS 110:PB_EINT0	001:Output 011:Reserved 101:Reserved 111:IO Disable

10.7.5.2 PB Configure Register 1 (Default Value: 0x0000_0777)

Offset: 0x0028			Register Name: PB_CFG1	
Bit	**Read/Write**	**Default/Hex**	**Description**	
31:11	/	/	/	
10:8	R/W	0x7	PB10_SELECT 000:Input 010:PWM1 100:UART0_RX 110:PB_EINT10	001:Output 011:CPU_CUR_W 101:Reserved 111:IO Disable
7	/	/	/	
6:4	R/W	0x7	PB9_SELECT 000:Input 010:Reserved 100:UART0_TX 110:PB_EINT9	001:Output 011:Reserved 101:Reserved 111:IO Disable
3	/	/	/	
2:0	R/W	0x7	PB8_SELECT 000:Input 010:Reserved 100:Reserved 110:PB_EINT8	001:Output 011:I2S0_MCLK 101:Reserved 111:IO Disable

10.7.5.3 PB Configure Register 2 (Default Value: 0x0000_0000)

Offset: 0x002C			Register Name: PB_CFG2
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

10.7.5.4 PB Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: PB_CFG3
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

10.7.5.5 PB Data Register (Default Value: 0x0000_0000)

Offset: 0x0034			Register Name: PB_DAT
Bit	Read/Write	Default/Hex	Description
31:11	/	/	/
10:0	R/W	0x0	PB_DAT If the port is configured as input, the corresponding bit is the pin state. If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.

10.7.5.6 PB Multi-Driving Register 0 (Default Value: 0x0015_5555)

Offset: 0x0038			Register Name: PB_DRV0
Bit	Read/Write	Default/Hex	Description
31:22	/	/	/
21:20	R/W	0x1	PB10_DRV PB10 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
19:18	R/W	0x1	PB9_DRV PB9 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
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 10: Level 2	01: Level 1 11: Level 3
13:12	R/W	0x1	PB6_DRV PB6 Multi-Driving Select 00: Level 0 10: Level 2	 01: Level 1 11: Level 3
11:10	R/W	0x1	PB5_DRV PB5 Multi-Driving Select 00: Level 0 10: Level 2	 01: Level 1 11: Level 3
9:8	R/W	0x1	PB4_DRV PB4 Multi-Driving Select 00: Level 0 10: Level 2	 01: Level 1 11: Level 3
7:6	R/W	0x1	PB3_DRV PB3 Multi-Driving Select 00: Level 0 10: Level 2	 01: Level 1 11: Level 3
5:4	R/W	0x1	PB2_DRV PB2 Multi-Driving Select 00: Level 0 10: Level 2	 01: Level 1 11: Level 3
3:2	R/W	0x1	PB1_DRV PB1 Multi-Driving Select 00: Level 0 10: Level 2	 01: Level 1 11: Level 3
1:0	R/W	0x1	PB0_DRV PB0 Multi-Driving Select 00: Level 0 10: Level 2	 01: Level 1 11: Level 3

10.7.5.7 PB Multi-Driving Register 1 (Default Value: 0x0000_0000)

Offset: 0x003C			Register Name: PB_DRV1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.8 PB Pull Register 0 (Default Value: 0x0000_0000)

Offset: 0x0040			Register Name: PB_PULL0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:22	/	/	/
21:20	R/W	0x0	PB10_PULL PB10 Pull-up/down Select

			00: Pull-up/down disable 10: Pull-down	01: Pull-up 11: Reserved
19:18	R/W	0x0	PB9_PULL PB9 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
17:16	R/W	0x0	PB8_PULL PB8 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
15:14	R/W	0x0	PB7_PULL PB7 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
13:12	R/W	0x0	PB6_PULL PB6 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
11:10	R/W	0x0	PB5_PULL PB5 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
9:8	R/W	0x0	PB4_PULL PB4 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
7:6	R/W	0x0	PB3_PULL PB3 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
5:4	R/W	0x0	PB2_PULL PB2 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
3:2	R/W	0x0	PB1_PULL PB1 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
1:0	R/W	0x0	PB0_PULL PB0 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved

10.7.5.9 PB Pull Register 1 (Default Value: 0x0000_0000)

Offset: 0x0044			Register Name: PB_PULL1
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

10.7.5.10 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 001:Output 010:NAND_RB1 011:Reserved 100:SPI0_CS1 101:Reserved 110:Reserved 111:IO Disable
27	/	/	/
26:24	R/W	0x7	PC6_SELECT 000:Input 001:Output 010:NAND_RB0 011:SDC2_CMD 100:Reserved 101:Reserved 110:Reserved 111:IO Disable
23	/	/	/
22:20	R/W	0x7	PC5_SELECT 000:Input 001:Output 010:NAND_RE 011:SDC2_CLK 100:Reserved 101:Reserved 110:Reserved 111:IO Disable
19	/	/	/
18:16	R/W	0x7	PC4_SELECT 000:Input 001:Output 010:NAND_CE0 011:Reserved 100:SPI0_MISO 101:Reserved 110:Reserved 111:IO Disable
15	/	/	/
14:12	R/W	0x7	PC3_SELECT 000:Input 001:Output 010:NAND_CE1 011:Reserved 100:SPI0_CS0 101:Reserved 110:Reserved 111:IO Disable
11	/	/	/
10:8	R/W	0x7	PC2_SELECT 000:Input 001:Output 010:NAND_CLE 011:Reserved 100:SPI0_MOSI 101:Reserved

			110:Reserved	111:IO Disable
7	/	/	/	
6:4	R/W	0x7	PC1_SELECT 000:Input 010:NAND_ALE 100:Reserved 110:Reserved	001:Output 011:SDC2_RST 101:Reserved 111:IO Disable
3	/	/	/	
2:0	R/W	0x7	PC0_SELECT 000:Input 010:NAND_WE 100:SPI0_CLK 110:Reserved	001:Output 011:SDC2_DS 101:Reserved 111:IO Disable

10.7.5.11 PC Configure Register 1 (Default Value: 0x7777_7777)

Offset: 0x004C			Register Name: PC_CFG1	
Bit	**Read/Write**	**Default/Hex**	**Description**	
31	/	/	/	
30:28	R/W	0x7	PC15_SELECT 000:Input 010:NAND_DQ1 100:SPI0_WP 110:Reserved	001:Output 011:SDC2_D2 101:Reserved 111:IO Disable
27	/	/	/	
26:24	R/W	0x7	PC14_SELECT 000:Input 010:NAND_DQ2 100:Reserved 110:Reserved	001:Output 011:SDC2_D6 101:Reserved 111:IO Disable
23	/	/	/	
22:20	R/W	0x7	PC13_SELECT 000:Input 010:NAND_DQ3 100:Reserved 110:Reserved	001:Output 011:SDC2_D1 101:Reserved 111:IO Disable
19	/	/	/	
18:16	R/W	0x7	PC12_SELECT 000:Input 010:NAND_DQS 100:Reserved 110:Reserved	001:Output 011:Reserved 101:Reserved 111:IO Disable
15	/	/	/	
14:12	R/W	0x7	PC11_SELECT 000:Input	001:Output

			010:NAND_DQ4	011:SDC2_D5
			100:Reserved	101:Reserved
			110:Reserved	111:IO Disable
11	/	/	/	
10:8	R/W	0x7	PC10_SELECT 000:Input 010:NAND_DQ5 100:Reserved 110:Reserved	 001:Output 011:SDC2_D0 101:Reserved 111:IO Disable
7	/	/	/	
6:4	R/W	0x7	PC9_SELECT 000:Input 010:NAND_DQ6 100:Reserved 110:Reserved	 001:Output 011:SDC2_D4 101:Reserved 111:IO Disable
3	/	/	/	
2:0	R/W	0x7	PC8_SELECT 000:Input 010:NAND_DQ7 100:Reserved 110:Reserved	 001:Output 011:SDC2_D3 101:Reserved 111:IO Disable

10.7.5.12 PC Configure Register 2 (Default Value: 0x0000_0007)

Offset: 0x0050			Register Name: PC_CFG2	
Bit	**Read/Write**	**Default/Hex**	**Description**	
31:3	/	/	/	
2:0	R/W	0x7	PC16_SELECT 000:Input 010:NAND_DQ0 100:SPI0_HOLD 110:Reserved	 001:Output 011:SDC2_D7 101:Reserved 111:IO Disable

10.7.5.13 PC Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x0054			Register Name: PC_CFG3
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.14 PC Data Register (Default Value: 0x0000_0000)

Offset: 0x0058			Register Name: PC_DAT
Bit	**Read/Write**	**Default/Hex**	**Description**

31:17	/	/	/
16:0	R/W	0x0	PC_DAT If the port is configured as input, the corresponding bit is the pin state. If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.

10.7.5.15 PC Multi-Driving Register 0 (Default Value: 0x5555_5555)

Offset: 0x005C			Register Name: PC_DRV0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	R/W	0x1	PC15_DRV PC15 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
29:28	R/W	0x1	PC14_DRV PC14 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
27:26	R/W	0x1	PC13_DRV PC13 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
25:24	R/W	0x1	PC12_DRV PC12 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
23:22	R/W	0x1	PC11_DRV PC11 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
21:20	R/W	0x1	PC10_DRV PC10 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
19:18	R/W	0x1	PC9_DRV PC9 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
17:16	R/W	0x1	PC8_DRV PC8 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
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
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

10.7.5.16 PC Multi-Driving Register 1 (Default Value: 0x0000_0001)

Offset: 0x0060			Register Name: PC_DRV1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1:0	R/W	0x1	PC16_DRV PC16 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

10.7.5.17 PC Pull Register 0 (Default Value: 0x4000_0440)

Offset: 0x0064			Register Name: PC_PULL0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	R/W	0x1	PC15_PULL PC15 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
29:28	R/W	0x0	PC14_PULL PC14 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
27:26	R/W	0x0	PC13_PULL PC13 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
25:24	R/W	0x0	PC12_PULL PC12 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
23:22	R/W	0x0	PC11_PULL PC11 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
21:20	R/W	0x0	PC10_PULL PC10 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
19:18	R/W	0x0	PC9_PULL PC9 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
17:16	R/W	0x0	PC8_PULL PC8 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
15:14	R/W	0x0	PC7_PULL PC7 Pull-up/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	0x1	PC3_PULL PC3 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
5:4	R/W	0x0	PC2_PULL PC2 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
3:2	R/W	0x0	PC1_PULL PC Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
1:0	R/W	0x0	PC0_PULL PC0 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved

10.7.5.18 PC Pull Register 1 (Default Value: 0x0000_0001)

Offset: 0x0068			Register Name: PC_PULL1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:2	/	/	/
1:0	R/W	0x1	PC16_PULL PC16 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved

10.7.5.19 PD Configure Register 0 (Default Value: 0x0000_0077)

Offset: 0x006C			Register Name: PD_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31	/	/	/
30:28	R/W	0x7	PD7_SELECT 000:Input 001:Output 010:LCD_D11 011:LVDS_CKN 100:Reserved 101:Reserved 110:Reserved 111:IO Disable
27	/	/	/

26:24	R/W	0x7	PD6_SELECT 000:Input 010:LCD_D10 100:Reserved 110:Reserved	 001:Output 011:LVDS_CKP 101:Reserved 111:IO Disable
23	/	/	/	
22:20	R/W	0x7	PD5_SELECT 000:Input 010:LCD_D7 100:Reserved 110:Reserved	 001:Output 011:LVDS_D2N 101:Reserved 111:IO Disable
19	/	/	/	
18:16	R/W	0x7	PD4_SELECT 000:Input 010:LCD_D6 100:Reserved 110:Reserved	 001:Output 011:LVDS_D2P 101:Reserved 111:IO Disable
15	/	/	/	
14:12	R/W	0x7	PD3_SELECT 000:Input 010:LCD_D5 100:Reserved 110:Reserved	 001:Output 011:LVDS_D1N 101:Reserved 111:IO Disable
11	/	/	/	
10:8	R/W	0x7	PD2_SELECT 000:Input 010:LCD_D4 100:Reserved 110:Reserved	 001:Output 011:LVDS_D1P 101:Reserved 111:IO Disable
7	/	/	/	
6:4	R/W	0x7	PD1_SELECT 000:Input 010:LCD_D3 100:Reserved 110:Reserved	 001:Output 011:LVDS_D0N 101:Reserved 111:IO Disable
3	/	/	/	
2:0	R/W	0x7	PD0_SELECT 000:Input 010:LCD_D2 100:Reserved 110:Reserved	 001:Output 011:LVDS_D0P 101:Reserved 111:IO Disable

10.7.5.20 PD Configure Register 1 (Default Value: 0x0000_0000)

Offset: 0x0070	Register Name: PD_CFG1

Bit	Read/Write	Default/Hex	Description
31	/	/	/
30:28	R/W	0x7	PD15_SELECT 000:Input 001:Output 010:LCD_D21 011:Reserved 100:UART3_RX 101:Reserved 110:Reserved 111:IO Disable
27	/	/	/
26:24	R/W	0x7	PD14_SELECT 000:Input 001:Output 010:LCD_D20 011:Reserved 100:UART3_TX 101:Reserved 110:Reserved 111:IO Disable
23	/	/	/
22:20	R/W	0x7	PD13_SELECT 000:Input 001:Output 010:LCD_D19 011:Reserved 100:SPI1_MISO 101:Reserved 110:Reserved 111:IO Disable
19	/	/	/
18:16	R/W	0x7	PD12_SELECT 000:Input 001:Output 010:LCD_D18 011:Reserved 100:SPI1_MOSI 101:Reserved 110:Reserved 111:IO Disable
15	/	/	/
14:12	R/W	0x7	PD11_SELECT 000:Input 001:Output 010:LCD_D15 011:Reserved 100:SPI1_CLK 101:Reserved 110:Reserved 111:IO Disable
11	/	/	/
10:8	R/W	0x7	PD10_SELECT 000:Input 001:Output 010:LCD_D14 011:Reserved 100:SPI1_CS 101:Reserved 110:Reserved 111:IO Disable
7	/	/	/
6:4	R/W	0x7	PD9_SELECT 000:Input 001:Output 010:LCD_D13 011:LVDS_D3N 100:Reserved 101:Reserved 110:Reserved 111:IO Disable
3	/	/	/
2:0	R/W	0x7	PD8_SELECT 000:Input 001:Output

			010:LCD_D12	011:LVDS_D3P
			100:Reserved	101:Reserved
			110:Reserved	111:IO Disable

10.7.5.21 PD Configure Register 2 (Default Value: 0x0000_0000)

Offset: 0x0074			Register Name: PD_CFG2	
Bit	**Read/Write**	**Default/Hex**	**Description**	
31	/	/	/	
30:28	R/W	0x7	PD23_SELECT 000:Input 010:PWM0 100:Reserved 110:Reserved	001:Output 011:Reserved 101:Reserved 111:IO Disable
27	/	/	/	
26:24	R/W	0x7	PD22_SELECT 000:Input 010:Reserved 100:Reserved 110:Reserved	001:Output 011:Reserved 101:Reserved 111:IO Disable
23	/	/	/	
22:20	R/W	0x7	PD21_SELECT 000:Input 010:LCD_VSYNC 100:UART4_CTS 110:Reserved	001:Output 011:Reserved 101:Reserved 111:IO Disable
19	/	/	/	
18:16	R/W	0x7	PD20_SELECT 000:Input 010:LCD_HSYNC 100:UART4_RTS 110:Reserved	001:Output 011:Reserved 101:Reserved 111:IO Disable
15	/	/	/	
14:12	R/W	0x7	PD19_SELECT 000:Input 010:LCD_DE 100:UART4_RX 110:Reserved	001:Output 011:Reserved 101:Reserved 111:IO Disable
11	/	/	/	
10:8	R/W	0x7	PD18_SELECT 000:Input 010:LCD_CLK 100:UART4_TX 110:Reserved	001:Output 011:Reserved 101:Reserved 111:IO Disable
7	/	/	/	

6:4	R/W	0x7	PD17_SELECT 000:Input 001:Output 010:LCD_D23 011:Reserved 100:UART3_CTS 101:Reserved 110:Reserved 111:IO Disable
3	/	/	/
2:0	R/W	0x7	PD16_SELECT 000:Input 001:Output 010:LCD_D22 011:Reserved 100:UART3_RTS 101:Reserved 110:Reserved 111:IO Disable

10.7.5.22 PD Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x0078			Register Name: PD_CFG3
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.23 PD Data Register (Default Value: 0x0000_0000)

Offset: 0x007C			Register Name: PD_DAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	/	/	/
23:0	R/W	0x0	PD_DAT If the port is configured as input, the corresponding bit is the pin state. If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.

10.7.5.24 PD Multi-Driving Register 0 (Default Value: 0x0000_0005)

Offset: 0x0080			Register Name: PD_DRV0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	R/W	0x1	PD15_DRV PD15 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
29:28	R/W	0x1	PD14_DRV PD14 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
27:26	R/W	0x1	PD13_DRV

			PD13 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
25:24	R/W	0x1	PD12_DRV PD12 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
23:22	R/W	0x1	PD11_DRV PD11 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
21:20	R/W	0x1	PD10_DRV PD10 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
19:18	R/W	0x1	PD9_DRV PD9 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
17:16	R/W	0x1	PD8_DRV PD8 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
15:14	R/W	0x1	PD7_DRV PD7 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
13:12	R/W	0x1	PD6_DRV PD6 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
11:10	R/W	0x1	PD5_DRV PD5 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
9:8	R/W	0x1	PD4_DRV PD4 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
7:6	R/W	0x1	PD3_DRV PD3 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
5:4	R/W	0x1	PD2_DRV PD2 Multi-Driving Select 00: Level 0 01: Level 1

			10: Level 2 11: Level 3
3:2	R/W	0x1	PD1_DRV PD1 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
1:0	R/W	0x1	PD0_DRV PD0 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

10.7.5.25 PD Multi-Driving Register 1 (Default Value: 0x0000_0000)

Offset: 0x84			Register Name: PD_DRV1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:14	R/W	0x1	PD23_DRV PD23 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
13:12	R/W	0x1	PD22_DRV PD22 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
11:10	R/W	0x1	PD21_DRV PD21 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
9:8	R/W	0x1	PD20_DRV PD20 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
7:6	R/W	0x1	PD19_DRV PD19 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
5:4	R/W	0x1	PD18_DRV PD18 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
3:2	R/W	0x1	PD17_DRV PD17 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
1:0	R/W	0x1	PD16_DRV PD16 Multi-Driving Select

			00: Level 0 10: Level 2	01: Level 1 11: Level 3

10.7.5.26 PD Pull Register 0 (Default Value: 0x0000_0000)

Offset: 0x0088			Register Name: PD_PULL0	
Bit	**Read/Write**	**Default/Hex**	**Description**	
31:30	R/W	0x0	PD15_PULL PD15 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
29:28	R/W	0x0	PD14_PULL PD14 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
27:26	R/W	0x0	PD13_PULL PD13 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
25:24	R/W	0x0	PD12_PULL PD12 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
23:22	R/W	0x0	PD11_PULL PD11 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
21:20	R/W	0x0	PD10_PULL PD10 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
19:18	R/W	0x0	PD9_PULL PD9 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
17:16	R/W	0x0	PD8_PULL PD8 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
15:14	R/W	0x0	PD7_PULL PD7 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
13:12	R/W	0x0	PD6_PULL PD6 Pull-up/down Select	

			00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
11:10	R/W	0x0	PD5_PULL PD5 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
9:8	R/W	0x0	PD4_PULL PD4 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
7:6	R/W	0x0	PD3_PULL PD3 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
5:4	R/W	0x0	PD2_PULL PD2 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
3:2	R/W	0x0	PD1_PULL PD1 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
1:0	R/W	0x0	PD0_PULL PD0 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved

10.7.5.27 PD Pull Register 1 (Default Value: 0x0000_0000)

Offset: 0x008C			Register Name: PD_PULL1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	/	/	/
15:14	R/W	0x0	PD23_PULL PD23 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
13:12	R/W	0x0	PD22_PULL PD22 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
11:10	R/W	0x0	PD21_PULL PD21 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
9:8	R/W	0x0	PD20_PULL

			PD20 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
7:6	R/W	0x0	PD19_PULL PD19 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
5:4	R/W	0x0	PD18_PULL PD18 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
3:2	R/W	0x0	PD17_PULL PD17 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
1:0	R/W	0x0	PD16_PULL PD16 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved

10.7.5.28 PE Configure Register 0 (Default Value: 0x7777_7777)

Offset: 0x0090			Register Name: PE_CFG0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:27	/	/	/
26:24	R/W	0x7	PE6_SELECT 000:Input 001:Output 010:Reserved 011:Reserved 100:Reserved 101:Reserved 110:Reserved 111:IO Disable
23	/	/	/
22:20	R/W	0x7	PE5_SELECT 000:Input 001:Output 010:Reserved 011:PLL_LOCK_DBG 100:Reserved 101:Reserved 110:Reserved 111:IO Disable
19	/	/	/
18:16	R/W	0x7	PE4_SELECT 000:Input 001:Output 010:Reserved 011:BIST_RESULT1 100:Reserved 101:Reserved 110:Reserved 111:IO Disable
15	/	/	/
14:12	R/W	0x7	PE3_SELECT 000:Input 001:Output

			010:Reserved 100:Reserved 110:Reserved	011:BIST_RESULT0 101:Reserved 111:IO Disable
11	/	/	/	
10:8	R/W	0x7	PE2_SELECT 000:Input 010:MIPI_SDA 100:Reserved 110:Reserved	 001:Output 011:TWI2_SDA 101:Reserved 111:IO Disable
7	/	/	/	
6:4	R/W	0x7	PE1_SELECT 000:Input 010:MIPI_SCK 100:Reserved 110:Reserved	 001:Output 011:TWI2_SCK 101:Reserved 111:IO Disable
3	/	/	/	
2:0	R/W	0x7	PE0_SELECT 000:Input 010:MIPI_MCLK 100:Reserved 110:Reserved	 001:Output 011:Reserved 101:Reserved 111:IO Disable

10.7.5.29 PE Configure Register 1 (Default Value: 0x7777_7777)

Offset: 0x0094			Register Name: PE_CFG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.30 PE Configure Register 2 (Default Value: 0x0000_0077)

Offset: 0x0098			Register Name: PE_CFG2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.31 PE Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x009C			Register Name: PE_CFG3
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.32 PE Data Register (Default Value: 0x0000_0000)

Offset: 0x00A0			Register Name: PE_DAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:7	/	/	/
6:0	R/W	0x0	PE_DAT If the port is configured as input, the corresponding bit is the pin state. If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.

10.7.5.33 PE Multi-Driving Register 0 (Default Value: 0x5555_5555)

Offset: 0x00A4			Register Name: PE_DRV0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:14	/	/	/
13:12	R/W	0x1	PE6_DRV PE6 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
11:10	R/W	0x1	PE5_DRV PE5 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
9:8	R/W	0x1	PE4_DRV PE4 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
7:6	R/W	0x1	PE3_DRV PE3 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
5:4	R/W	0x1	PE2_DRV PE2 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
3:2	R/W	0x1	PE1_DRV PE1 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
1:0	R/W	0x1	PE0_DRV PE0 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3

10.7.5.34 PE Multi-Driving Register 1 (Default Value: 0x0000_0005)

Offset: 0x00A8			Register Name: PE_DRV1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.35 PE PULL Register 0 (Default Value: 0x0000_0000)

Offset: 0x00AC			Register Name: PE_PULL0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:14	/	/	/
13:12	R/W	0x0	PE6_PULL PE6 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
11:10	R/W	0x0	PE5_PULL PE5 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
9:8	R/W	0x0	PE4_PULL PE4 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
7:6	R/W	0x0	PE3_PULL PE3 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
5:4	R/W	0x0	PE2_PULL PE2 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
3:2	R/W	0x0	PE1_PULL PE1 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
1:0	R/W	0x0	PE0_PULL PE0 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved

10.7.5.36 PE PULL Register 1 (Default Value: 0x0000_0000)

Offset: 0x00B0			Register Name: PE_PULL1
Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

10.7.5.37 PF Configure Register 0 (Default Value: 0x7777_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:Reserved 011:Reserved 100:Reserved 101:Reserved 110:PF_EINT6 111:IO Disable
23	/	/	/
22:20	R/W	0x7	PF5_SELECT 000:Input 001:Output 010:SDC0_D2 011:JTAG_CK 100:Reserved 101:Reserved 110:PF_EINT5 111:IO Disable
19	/	/	/
18:16	R/W	0x7	PF4_SELECT 000:Input 001:Output 010:SDC0_D3 011:UART0_RX 100:Reserved 101:Reserved 110:PF_EINT4 111:IO Disable
15	/	/	/
14:12	R/W	0x7	PF3_SELECT 000:Input 001:Output 010:SDC0_CMD 011:JTAG_DO 100:Reserved 101:Reserved 110:PF_EINT3 111:IO Disable
11	/	/	/
10:8	R/W	0x7	PF2_SELECT 000:Input 001:Output 010:SDC0_CLK 011:UART0_TX 100:Reserved 101:Reserved 110:PF_EINT2 111:IO Disable
7	/	/	/
6:4	R/W	0x7	PF1_SELECT 000:Input 001:Output 010:SDC0_D0 011:JTAG_DI 100:Reserved 101:Reserved

			110:PF_EINT1	111:IO Disable
3	/	/	/	
2:0	R/W	0x7	PF0_SELECT 000:Input 010:SDC0_D1 100:Reserved 110:PF_EINT0	 001:Output 011:JTAG_MS 101:Reserved 111:IO Disable

10.7.5.38 PF Configure Register 1 (Default Value: 0x0000_0000)

Offset: 0x00B8			Register Name: PF_CFG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.39 PF Configure Register 2 (Default Value: 0x0000_0000)

Offset: 0x00BC			Register Name: PF_CFG2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.40 PF Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x00C0			Register Name: PF_CFG3
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.41 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 input, the corresponding bit is the pin state. If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.

10.7.5.42 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
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

10.7.5.43 PF Multi-Driving Register 1 (Default Value: 0x0000_0000)

Offset: 0x00CC			Register Name: PF_DRV1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.44 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 10: Pull-down	01: Pull-up 11: Reserved
11:10	R/W	0x0	PF5_PULL PF5 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
9:8	R/W	0x0	PF4_PULL PF4 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
7:6	R/W	0x0	PF3_PULL PF3 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
5:4	R/W	0x0	PF2_PULL PF2 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
3:2	R/W	0x0	PF1_PULL PF1 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
1:0	R/W	0x0	PF0_PULL PF0 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved

10.7.5.45 PF Pull Register 1 (Default Value: 0x0000_0000)

Offset: 0x00D4			Register Name: PF_PULL1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.46 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 010:UART1_RX 100:Reserved 110:PG_EINT7	 001:Output 011:Reserved 101:Reserved 111:IO Disable
27	/	/	/	

26:24	R/W	0x7	PG6_SELECT 000:Input 010:UART1_TX 100:Reserved 110:PG_EINT6	001:Output 011:Reserved 101:Reserved 111:IO Disable
23	/	/	/	
22:20	R/W	0x7	PG5_SELECT 000:Input 010:SDC1_D3 100:Reserved 110:PG_EINT5	001:Output 011:Reserved 101:Reserved 111:IO Disable
19	/	/	/	
18:16	R/W	0x7	PG4_SELECT 000:Input 010:SDC1_D2 100:Reserved 110:PG_EINT4	001:Output 011:Reserved 101:Reserved 111:IO Disable
15	/	/	/	
14:12	R/W	0x7	PG3_SELECT 000:Input 010:SDC1_D1 100:Reserved 110:PG_EINT3	001:Output 011:Reserved 101:Reserved 111:IO Disable
11	/	/	/	
10:8	R/W	0x7	PG2_SELECT 000:Input 010:SDC1_D0 100:Reserved 110:PG_EINT2	001:Output 011:Reserved 101:Reserved 111:IO Disable
7	/	/	/	
6:4	R/W	0x7	PG1_SELECT 000:Input 010:SDC1_CMD 100:Reserved 110:PG_EINT1	001:Output 011:Reserved 101:Reserved 111:IO Disable
3	/	/	/	
2:0	R/W	0x7	PG0_SELECT 000:Input 010:SDC1_CLK 100:Reserved 110:PG_EINT0	001:Output 011:Reserved 101:Reserved 111:IO Disable

10.7.5.47 PG Configure Register 1 (Default Value: 0x0077_7777)

Offset: 0x00DC	Register Name: PG_CFG1

Bit	Read/Write	Default/Hex	Description
31:23	/	/	/
22:20	R/W	0x7	PG13_SELECT 000:Input 001:Output 010:AIF3_DIN 011:I2S1_DIN 100:Reserved 101:Reserved 110:PG_EINT13 111:IO Disable
19	/	/	/
18:16	R/W	0x7	PG12_SELECT 000:Input 001:Output 010:AIF3_DOUT 011:I2S1_DOUT 100:Reserved 101:Reserved 110:PG_EINT12 111:IO Disable
15	/	/	/
14:12	R/W	0x7	PG11_SELECT 000:Input 001:Output 010:AIF3_BCLK 011:I2S1_BCLK 100:Reserved 101:Reserved 110:PG_EINT11 111:IO Disable
11	/	/	/
10:8	R/W	0x7	PG10_SELECT 000:Input 001:Output 010:AIF3_SYNC 011:I2S1_LRCK 100:Reserved 101:Reserved 110:PG_EINT10 111:IO Disable
7	/	/	/
6:4	R/W	0x7	PG9_SELECT 000:Input 001:Output 010:UART1_CTS 011:I2S1_MCLK 100:Reserved 101:Reserved 110:PG_EINT9 111:IO Disable
3	/	/	/
2:0	R/W	0x7	PG8_SELECT 000:Input 001:Output 010:UART1_RTS 011:Reserved 100:Reserved 101:Reserved 110:PG_EINT8 111:IO Disable

10.7.5.48 PG Configure Register 2 (Default Value: 0x0000_0000)

Offset: 0x00E0			Register Name: PG_CFG2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.49 PG Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x00E4			Register Name: PG_CFG3
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.50 PG Data Register (Default Value: 0x0000_0000)

Offset: 0x00E8			Register Name: PG_DAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:14	/	/	/
13:0	R/W	0x0	PG_DAT If the port is configured as input, the corresponding bit is the pin state. If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.

10.7.5.51 PG Multi-Driving Register 0 (Default Value: 0x0555_5555)

Offset: 0x00EC			Register Name: PG_DRV0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	/	/	/
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
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
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

10.7.5.52 PG Multi-Driving Register 1 (Default Value: 0x0000_0000)

Offset: 0x00F0			Register Name: PG_DRV1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.53 PG Pull Register 0 (Default Value: 0x0000_0000)

Offset: 0x00F4			Register Name: PG_PULL0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	/	/	/
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
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 10: Pull-down	 01: Pull-up 11: Reserved
5:4	R/W	0x0	PG2_PULL PG2 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
3:2	R/W	0x0	PG1_PULL PG1 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
1:0	R/W	0x0	PG0_PULL PG0 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved

10.7.5.54 PG Pull Register 1 (Default Value: 0x0000_0000)

Offset: 0x00F8			Register Name: PG_PULL1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.55 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 010:UART3_CTS 100:Reserved 110:PH_EINT7	 001:Output 011:SPI1_MISO 101:Reserved 111:IO Disable
27	/	/	/	
26:24	R/W	0x7	PH6_SELECT 000:Input 010:UART3_RTS 100:Reserved 110:PH_EINT6	 001:Output 011:SPI1_MOSI 101:Reserved 111:IO Disable
23	/	/	/	
22:20	R/W	0x7	PH5_SELECT 000:Input 010:UART3_RX 100:Reserved 110:PH_EINT5	 001:Output 011:SPI1_CLK 101:Reserved 111:IO Disable

19	/	/	/	
18:16	R/W	0x7	PH4_SELECT 000:Input 010:UART3_TX 100:CPU_CUR_W 110:PH_EINT4	001:Output 011:SPI1_CS 101:Reserved 111:IO Disable
15	/	/	/	
14:12	R/W	0x7	PH3_SELECT 000:Input 010:TWI1_SDA 100:Reserved 110:PH_EINT3	001:Output 011:Reserved 101:Reserved 111:IO Disable
11	/	/	/	
10:8	R/W	0x7	PH2_SELECT 000:Input 010:TWI1_SCK 100:Reserved 110:PH_EINT2	001:Output 011:CPU_CUR_W 101:Reserved 111:IO Disable
7	/	/	/	
6:4	R/W	0x7	PH1_SELECT 000:Input 010:TWI0_SDA 100:Reserved 110:PH_EINT1	001:Output 011:Reserved 101:Reserved 111:IO Disable
3	/	/	/	
2:0	R/W	0x7	PH0_SELECT 000:Input 010:TWI0_SCK 100:Reserved 110:PH_EINT0	001:Output 011:Reserved 101:Reserved 111:IO Disable

10.7.5.56 PH Configure Register 1 (Default Value: 0x0007_7777)

Offset: 0x0100			Register Name: PH_CFG1	
Bit	**Read/Write**	**Default/Hex**	**Description**	
31:19	/	/	/	
18:16	R/W	0x7	PH12_SELECT 000:Input 010:DMIC_CLK 100:Reserved 110:PH_EINT12	001:Output 011:Reserved 101:Reserved 111:IO Disable
15	/	/	/	
14:12	R/W	0x7	PH11_SELECT 000:Input 010:DMIC_DATA0	001:Output 011:Reserved

			100:Reserved 110:PH_EINT11	101:Reserved 111:IO Disable
11	/	/	/	
10:8	R/W	0x7	PH10_SELECT 000:Input 010:DMIC_DATA1 100:Reserved 110:PH_EINT10	 001:Output 011:Reserved 101:Reserved 111:IO Disable
7	/	/	/	
6:4	R/W	0x7	PH9_SELECT 000:Input 010:DMIC_DATA2 100:Reserved 110:PH_EINT9	 001:Output 011:Reserved 101:Reserved 111:IO Disable
3	/	/	/	
2:0	R/W	0x7	PH8_SELECT 000:Input 010:DMIC_DATA3 100:Reserved 110:PH_EINT8	 001:Output 011:Reserved 101:Reserved 111:IO Disable

10.7.5.57 PH Configure Register 2 (Default Value: 0x0000_0000)

Offset: 0x104			Register Name: PH_CFG2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.58 PH Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x108			Register Name: PH_CFG3
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.59 PH Data Register (Default Value: 0x0000_0000)

Offset: 0x10C			Register Name: PH_DAT
Bit	**Read/Write**	**Default/Hex**	**Description**
31:13	/	/	/
12:0	R/W	0x0	PH_DAT If the port is configured as input, the corresponding bit is the pin state. If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined

			value will be read.

10.7.5.60 PH Multi-Driving Register 0 (Default Value: 0x0155_5555)

Offset: 0x110			Register Name: PH_DRV0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:26	/	/	/
25:24	R/W	0x1	PH12_DRV PH12 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
23:22	R/W	0x1	PH11_DRV PH11 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
21:20	R/W	0x1	PH10_DRV PH10 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
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
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

10.7.5.61 PH Multi-Driving Register 1 (Default Value: 0x0000_0000)

Offset: 0x0114			Register Name: PH_DRV1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.62 PH Pull Register 0 (Default Value: 0x0000_0000)

Offset: 0x0118			Register Name: PH_PULL0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:26	/	/	/
25:24	R/W	0x0	PH12_PULL PH12 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
23:22	R/W	0x0	PH11_PULL PH11 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
21:20	R/W	0x0	PH10_PULL PH10 Pull-up/down Select 00: Pull-up/down disable 01: Pull-up 10: Pull-down 11: Reserved
19:18	R/W	0x0	PH9_PULL PH9 Pull-up/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
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

10.7.5.63 PH Pull Register 1 (Default Value: 0x0000_0000)

Offset: 0x011C			Register Name: PH_PULL1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.5.64 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 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
27:24	R/W	0x0	EINT6_CFG External INT6 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
23:20	R/W	0x0	EINT5_CFG External INT5 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
19:16	R/W	0x0	EINT4_CFG External INT4 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
15:12	R/W	0x0	EINT3_CFG External INT3 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
11:8	R/W	0x0	EINT2_CFG External INT2 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level

			0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
7:4	R/W	0x0	EINT1_CFG External INT1 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
3:0	R/W	0x0	EINT0_CFG External INT0 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved

10.7.5.65 PB External Interrupt Configure Register 1 (Default Value: 0x0000_0000)

Offset: 0x0224			Register Name: PB_EINT_CFG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:12	/	/	/
11:8	R/W	0x0	EINT10_CFG External INT10 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
7:4	R/W	0x0	EINT9_CFG External INT9 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
3:0	R/W	0x0	EINT8_CFG External INT8 Mode 0000: Positive Edge 0001: Negative Edge

			0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved

10.7.5.66 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	/	/	/

10.7.5.67 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	/	/	/

10.7.5.68 PB External Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0230			Register Name: PB_EINT_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:11	/	/	/
10	R/W	0x0	EINT10_CTL External INT10 Enable 0: Disable 1: Enable
9	R/W	0x0	EINT9_CTL External INT9 Enable 0: Disable 1: Enable
8	R/W	0x0	EINT8_CTL External INT8 Enable 0: Disable 1: Enable
7	R/W	0x0	EINT7_CTL External INT7 Enable 0: Disable 1: Enable
6	R/W	0x0	EINT6_CTL External INT6 Enable 0: Disable 1: Enable
5	R/W	0x0	EINT5_CTL

			External INT5 Enable 0: Disable 1: Enable
4	R/W	0x0	EINT4_CTL External INT4 Enable 0: Disable 1: Enable
3	R/W	0x0	EINT3_CTL External INT3 Enable 0: Disable 1: Enable
2	R/W	0x0	EINT2_CTL External INT2 Enable 0: Disable 1: Enable
1	R/W	0x0	EINT1_CTL External INT1 Enable 0: Disable 1: Enable
0	R/W	0x0	EINT0_CTL External INT0 Enable 0: Disable 1: Enable

10.7.5.69 PB External Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0234			Register Name: PB_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
9	R/W	0x0	EINT9_STATUS External INT9 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
8	R/W	0x0	EINT8_STATUS External INT8 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
7	R/W	0x0	EINT7_STATUS

			External INT7 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
6	R/W	0x0	EINT6_STATUS External INT6 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
5	R/W	0x0	EINT5_STATUS External INT5 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
4	R/W	0x0	EINT4_STATUS External INT4 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
3	R/W	0x0	EINT3_STATUS External INT3 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
2	R/W	0x0	EINT2_STATUS External INT2 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
1	R/W	0x0	EINT1_STATUS External INT1 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
0	R/W	0x0	EINT0_STATUS External INT0 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

10.7.5.70 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

10.7.5.71 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 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
23:20	R/W	0x0	EINT5_CFG External INT5 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
19:16	R/W	0x0	EINT4_CFG External INT4 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
15:12	R/W	0x0	EINT3_CFG External INT3 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative)

			Others: Reserved
11:8	R/W	0x0	EINT2_CFG External INT2 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
7:4	R/W	0x0	EINT1_CFG External INT1 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
3:0	R/W	0x0	EINT0_CFG External INT0 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved

10.7.5.72 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	/	/	/

10.7.5.73 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	/	/	/

10.7.5.74 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	/	/	/

10.7.5.75 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
2	R/W	0x0	EINT2_CTL External INT2 Enable 0: Disable 1: Enable
1	R/W	0x0	EINT1_CTL External INT1 Enable 0: Disable 1: Enable
0	R/W	0x0	EINT0_CTL External INT0 Enable 0: Disable 1: Enable

10.7.5.76 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
3	R/W	0x0	EINT3_STATUS External INT3 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
2	R/W	0x0	EINT2_STATUS External INT2 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
1	R/W	0x0	EINT1_STATUS External INT1 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
0	R/W	0x0	EINT0_STATUS External INT0 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear

10.7.5.77 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

10.7.5.78 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 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
27:24	R/W	0x0	EINT6_CFG External INT6 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
23:20	R/W	0x0	EINT5_CFG External INT5 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
19:16	R/W	0x0	EINT4_CFG External INT4 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
15:12	R/W	0x0	EINT3_CFG External INT3 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved

11:8	R/W	0x0	EINT2_CFG External INT2 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
7:4	R/W	0x0	EINT1_CFG External INT1 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
3:0	R/W	0x0	EINT0_CFG External INT0 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved

10.7.5.79 PG External Interrupt Configure Register 1 (Default Value: 0x0000_0000)

Offset: 0x02C4			Register Name: PG_EINT_CFG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	/	/	/
23:20	R/W	0x0	EINT13_CFG External INT13 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
19:16	R/W	0x0	EINT12_CFG External INT12 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative)

			Others: Reserved
15:12	R/W	0x0	EINT11_CFG External INT11 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
11:8	R/W	0x0	EINT10_CFG External INT10 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
7:4	R/W	0x0	EINT9_CFG External INT9 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
3:0	R/W	0x0	EINT8_CFG External INT8 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved

10.7.5.80 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	/	/	/

10.7.5.81 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	/	/	/

10.7.5.82 PG External Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x02D0			Register Name: PG_EINT_CTL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:14	/	/	/
13	R/W	0x0	EINT13_CTL External INT13 Enable 0: Disable 1: Enable
12	R/W	0x0	EINT12_CTL External INT12 Enable 0: Disable 1: Enable
11	R/W	0x0	EINT11_CTL External INT11 Enable 0: Disable 1: Enable
10	R/W	0x0	EINT10_CTL External INT10 Enable 0: Disable 1: Enable
9	R/W	0x0	EINT9_CTL External INT9 Enable 0: Disable 1: Enable
8	R/W	0x0	EINT8_CTL External INT8 Enable 0: Disable 1: Enable
7	R/W	0x0	EINT7_CTL External INT7 Enable 0: Disable 1: Enable
6	R/W	0x0	EINT6_CTL External INT6 Enable 0: Disable 1: Enable
5	R/W	0x0	EINT5_CTL External INT5 Enable 0: Disable 1: Enable
4	R/W	0x0	EINT4_CTL External INT4 Enable

			0: Disable 1: Enable
3	R/W	0x0	EINT3_CTL External INT3 Enable 0: Disable 1: Enable
2	R/W	0x0	EINT2_CTL External INT2 Enable 0: Disable 1: Enable
1	R/W	0x0	EINT1_CTL External INT1 Enable 0: Disable 1: Enable
0	R/W	0x0	EINT0_CTL External INT0 Enable 0: Disable 1: Enable

10.7.5.83 PG External Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x02D4			Register Name: PG_EINT_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:14	/	/	/
13	R/W	0x0	EINT13_STATUS External INT13 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
12	R/W	0x0	EINT12_STATUS External INT12 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
11	R/W	0x0	EINT11_STATUS External INT11 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
10	R/W	0x0	EINT10_STATUS External INT10 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
9	R/W	0x0	EINT9_STATUS

			External INT9 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
8	R/W	0x0	EINT8_STATUS External INT8 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
7	R/W	0x0	EINT7_STATUS External INT7 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
6	R/W	0x0	EINT6_STATUS External INT6 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
5	R/W	0x0	EINT5_STATUS External INT5 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
4	R/W	0x0	EINT4_STATUS External INT4 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
3	R/W	0x0	EINT3_STATUS External INT3 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
2	R/W	0x0	EINT2_STATUS External INT2 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
1	R/W	0x0	EINT1_STATUS External INT1 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
0	R/W	0x0	EINT0_STATUS External INT0 Pending Bit

			0: No IRQ pending 1: IRQ pending Write '1' to clear

10.7.5.84 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

10.7.5.85 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 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
27:24	R/W	0x0	EINT6_CFG External INT6 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
23:20	R/W	0x0	EINT5_CFG External INT5 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level

			0100: Double Edge (Positive/ Negative) Others: Reserved
19:16	R/W	0x0	EINT4_CFG External INT4 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
15:12	R/W	0x0	EINT3_CFG External INT3 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
11:8	R/W	0x0	EINT2_CFG External INT2 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
7:4	R/W	0x0	EINT1_CFG External INT1 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
3:0	R/W	0x0	EINT0_CFG External INT0 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved

10.7.5.86 PH External Interrupt Configure Register 1 (Default Value: 0x0000_0000)

Offset: 0x02E4			Register Name: PH_EINT_CFG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:20	/	/	/
19:16	R/W	0x0	EINT12_CFG External INT12 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
15:12	R/W	0x0	EINT11_CFG External INT11 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
11:8	R/W	0x0	EINT10_CFG External INT10 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
7:4	R/W	0x0	EINT9_CFG External INT9 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
3:0	R/W	0x0	EINT8_CFG External INT8 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved

10.7.5.87 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	/	/	/

10.7.5.88 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	/	/	/

10.7.5.89 PH External Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x02F0			Register Name: PH_EINT_CTL
Bit	Read/Write	Default/Hex	Description
31:13	/	/	/
12	R/W	0x0	EINT12_CTL External INT12 Enable 0: Disable 1: Enable
11	R/W	0x0	EINT11_CTL External INT11 Enable 0: Disable 1: Enable
10	R/W	0x0	EINT10_CTL External INT10 Enable 0: Disable 1: Enable
9	R/W	0x0	EINT9_CTL External INT9 Enable 0: Disable 1: Enable
8	R/W	0x0	EINT8_CTL External INT8 Enable 0: Disable 1: Enable
7	R/W	0x0	EINT7_CTL External INT7 Enable 0: Disable 1: Enable
6	R/W	0x0	EINT6_CTL External INT6 Enable 0: Disable

			1: Enable
5	R/W	0x0	EINT5_CTL External INT5 Enable 0: Disable 1: Enable
4	R/W	0x0	EINT4_CTL External INT4 Enable 0: Disable 1: Enable
3	R/W	0x0	EINT3_CTL External INT3 Enable 0: Disable 1: Enable
2	R/W	0x0	EINT2_CTL External INT2 Enable 0: Disable 1: Enable
1	R/W	0x0	EINT1_CTL External INT1 Enable 0: Disable 1: Enable
0	R/W	0x0	EINT0_CTL External INT0 Enable 0: Disable 1: Enable

10.7.5.90 PH External Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x02F4			Register Name: PH_EINT_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:13	/	/	/
12	R/W	0x0	EINT12_STATUS External INT12 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
11	R/W	0x0	EINT11_STATUS External INT11 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
10	R/W	0x0	EINT10_STATUS External INT10 Pending Bit 0: No IRQ pending 1: IRQ pending

			Write ‘1’ to clear
9	R/W	0x0	EINT9_STATUS External INT9 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
8	R/W	0x0	EINT8_STATUS External INT8 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
7	R/W	0x0	EINT7_STATUS External INT7 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
6	R/W	0x0	EINT6_STATUS External INT6 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
5	R/W	0x0	EINT5_STATUS External INT5 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
4	R/W	0x0	EINT4_STATUS External INT4 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
3	R/W	0x0	EINT3_STATUS External INT3 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
2	R/W	0x0	EINT2_STATUS External INT2 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
1	R/W	0x0	EINT1_STATUS External INT1 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear

0	R/W	0x0	EINT0_STATUS External INT0 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

10.7.5.91 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

10.7.5.92 PIO Group Withstand Voltage Mode Select Register (Default Value: 0x0000_0000)

Offset: 0x0340			Register Name: PIO_POW_MOD_SEL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:13	/	/	/
12	R/W	0x0	VCC-IO POWER MODE Select 0: 3.3V 1: 1.8V
11:8	/	/	/
7	/	/	PH_POWER MODE Select invalid
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	R/W	0x0	PE_POWER MODE Select 0: 3.3V 1: 1.8V If PE_Port Power Source selects VCC-IO,this bit is invalid
3	R/W	0x0	PD_POWER MODE Select

			0: 3.3V 1: 1.8V If PD_Port Power Source selects VCC-IO,this bit is invalid
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	/	/	PB_POWER MODE Select invalid
0	/	/	/

NOTE

For 0x0340 register ,when the power domain of GPIO is larger than 1.8V, then the withstand voltage is set to 3.3V mode, that is, the corresponding register value is set to 0; when the power domain of GPIO is 1.8V,then the withstand voltage is set to 1.8V mode, that is, the corresponding register value is set to 1; when the power domain of GPIO is uncertain, then the withstand voltage is set to 3.3V mode, that is, the corresponding register value is set to 0.

10.7.5.93 PIO Group Power Value Register

Offset: 0x0348			**Register Name: PIO_POW_Val**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R	/	VCC-IO Power Value
15:8	/	/	/
7	/	/	PH_Port Power Value invalid
6	R	/	PG_Port Power Value If PG_Port Power Source selects VCC-IO,this bit is invalid
5	R	/	PF_Port Power Value If PF_Port Power Source selects VCC-IO,this bit is invalid
4	R	/	PE_Port Power Value If PE_Port Power Source selects VCC-IO,this bit is invalid
3	R	/	PD_Port Power Value If PD_Port Power Source selects VCC-IO,this bit is invalid
2	R	/	PC_Port Power Value If PC_Port Power Source selects VCC-IO,this bit is invalid
1	/	/	PB_Port Power Value invalid
0	/	/	/

NOTE

IO pressure mode is 3.3V when reading 0;IO pressure mode is 1.8V when reading 1.

10.7.6 CPUS Register List

Module Name	Base Address
R_GPIO	0x07022000

Register Name	Offset	Description
Pn_CFG0	n*0x0024+0x00	Port n Configure Register 0 (n =0)
Pn_CFG1	n*0x0024+0x04	Port n Configure Register 1 (n =0)
Pn_CFG2	n*0x0024+0x08	Port n Configure Register 2 (n =0)
Pn_CFG3	n*0x0024+0x0C	Port n Configure Register 3 (n =0)
Pn_DAT	n*0x0024+0x10	Port n Data Register (n =0)
Pn_DRV0	n*0x0024+0x14	Port n Multi-Driving Register 0 (n =0)
Pn_DRV1	n*0x0024+0x18	Port n Multi-Driving Register 1 (n =0)
Pn_PUL0	n*0x0024+0x1C	Port n Pull Register 0 (n =0)
Pn_PUL1	n*0x0024+0x20	Port n Pull Register 1 (n =0)
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_Val	0x0348	PIO Group Power Value Register

10.7.7 CPUS Register Description

10.7.7.1 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:R_JTAG_DI 011:Reserved 100:Reserved 101:Reserved 110:R_PL_EINT7 111:IO Disable
27	/	/	/
26:24	R/W	0x7	PL6_SELECT 000:Input 001:Output 010:R_JTAG_DO 011:Reserved 100:Reserved 101:Reserved

			110:R_PL_EINT6	111:IO Disable
23	/	/	/	
22:20	R/W	0x7	PL5_SELECT 000:Input 010:R_JTAG_CK 100:Reserved 110:R_PL_EINT5	001:Output 011:Reserved 101:Reserved 111:IO Disable
19	/	/	/	
18:16	R/W	0x7	PL4_SELECT 000:Input 010:R_JTAG_MS 100:Reserved 110:R_PL_EINT4	001:Output 011:Reserved 101:Reserved 111:IO Disable
15	/	/	/	
14:12	R/W	0x7	PL3_SELECT 000:Input 010:R_UART_RX 100:Reserved 110:R_PL_EINT3	001:Output 011:Reserved 101:Reserved 111:IO Disable
11	/	/	/	
10:8	R/W	0x7	PL2_SELECT 000:Input 010:R_UART_TX 100:Reserved 110:R_PL_EINT2	001:Output 011:Reserved 101:Reserved 111:IO Disable
7	/	/	/	
6:4	R/W	0x7	PL1_SELECT 000:Input 010:R_RSB_SDA 100:Reserved 110:R_PL_EINT1	001:Output 011:R_TWI_SDA 101:Reserved 111:IO Disable
3	/	/	/	
2:0	R/W	0x7	PL0_SELECT 000:Input 010:R_RSB_SCK 100:Reserved 110:R_PL_EINT0	001:Output 011:R_TWI_SCK 101:Reserved 111:IO Disable

10.7.7.2 PL Configure Register 1 (Default Value: 0x7777_7777)

Offset: 0x0004			Register Name: PL_CFG1	
Bit	**Read/Write**	**Default/Hex**	**Description**	
31:19	/	/	/	
18:16	R/W	0x7	PL12_SELECT 000:Input	001:Output

			010:Reserved 100:Reserved 110:R_PL_EINT12	011:Reserved 101:Reserved 111:IO Disable
15	/	/	/	
14:12	R/W	0x7	PL11_SELECT 000:Input 010:R_CPU_CUR_W 100:Reserved 110:R_PL_EINT11	 001:Output 011:Reserved 101:Reserved 111:IO Disable
11	/	/	/	
10:8	R/W	0x7	PL10_SELECT 000:Input 010:R_PWM 100:Reserved 110:R_PL_EINT10	 001:Output 011:Reserved 101:Reserved 111:IO Disable
7	/	/	/	
6:4	R/W	0x7	PL9_SELECT 000:Input 010:R_TWI_SDA 100:Reserved 110:R_PL_EINT9	 001:Output 011:Reserved 101:Reserved 111:IO Disable
3	/	/	/	
2:0	R/W	0x7	PL8_SELECT 000:Input 010:R_TWI_SCK 100:Reserved 110:R_PL_EINT8	 001:Output 011:Reserved 101:Reserved 111:IO Disable

10.7.7.3 PL Configure Register 2 (Default Value: 0x0000_7777)

Offset: 0x0008			Register Name: PL_CFG2
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.7.4 PL Configure Register 3 (Default Value: 0x0000_0000)

Offset: 0x000C			Register Name: PL_CFG3
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.7.5 PL Data Register (Default Value: 0x0000_0000)

Offset: 0x0010	Register Name: PL_DAT

Bit	Read/Write	Default/Hex	Description
31:13	/	/	/
12:0	R/W	0x0	PL_DAT If the port is configured as input, the corresponding bit is the pin state. If the port is configured as output, the pin state is the same as the corresponding bit. The read bit value is the value setup by software. If the port is configured as functional pin, the undefined value will be read.

10.7.7.6 PL Multi-Driving Register 0 (Default Value: 0x5555_5555)

Offset: 0x0014			Register Name: PL_DRV0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:26	/	/	/
25:24	R/W	0x1	PL12_DRV PL12 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
23:22	R/W	0x1	PL11_DRV PL11 Multi-Driving Select 00: Level 0 01: Level 1 10: Level 2 11: Level 3
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 10: Level 2	 01: Level 1 11: Level 3
7:6	R/W	0x1	PL3_DRV PL3 Multi-Driving Select 00: Level 0 10: Level 2	 01: Level 1 11: Level 3
5:4	R/W	0x1	PL2_DRV PL2 Multi-Driving Select 00: Level 0 10: Level 2	 01: Level 1 11: Level 3
3:2	R/W	0x1	PL1_DRV PL1 Multi-Driving Select 00: Level 0 10: Level 2	 01: Level 1 11: Level 3
1:0	R/W	0x1	PL0_DRV PL0 Multi-Driving Select 00: Level 0 10: Level 2	 01: Level 1 11: Level 3

10.7.7.7 PL Multi-Driving Register 1 (Default Value: 0x0000_0055)

Offset: 0x0018			Register Name: PL_DRV1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	/	/	/

10.7.7.8 PL Pull Register 0 (Default Value: 0x0000_0005)

Offset: 0x001C			Register Name: PL_PULL0
Bit	**Read/Write**	**Default/Hex**	**Description**
31:26	/	/	/
25:24	R/W	0x0	PL12_PULL PL12 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
23:22	R/W	0x0	PL11_PULL PL11 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
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 10: Pull-down	 01: Pull-up 11: Reserved
17:16	R/W	0x0	PL8_PULL PL8 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
15:14	R/W	0x0	PL7_PULL PL7 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
13:12	R/W	0x0	PL6_PULL PL6 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
11:10	R/W	0x0	PL5_PULL PL5 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
9:8	R/W	0x0	PL4_PULL PL4 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
7:6	R/W	0x0	PL3_PULL PL3 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
5:4	R/W	0x0	PL2_PULL PL2 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
3:2	R/W	0x1	PL1_PULL PL1 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved
1:0	R/W	0x1	PL0_PULL PL0 Pull-up/down Select 00: Pull-up/down disable 10: Pull-down	 01: Pull-up 11: Reserved

10.7.7.9 PL Pull Register 1 (Default Value: 0x0000_0000)

Offset: 0x20	Register Name: PL_PULL1

Bit	Read/Write	Default/Hex	Description
31:0	/	/	/

10.7.7.10 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 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
27:24	R/W	0x0	EINT6_CFG External INT6 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
23:20	R/W	0x0	EINT5_CFG External INT5 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
19:16	R/W	0x0	EINT4_CFG External INT4 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
15:12	R/W	0x0	EINT3_CFG External INT3 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level

			0100: Double Edge (Positive/ Negative) Others: Reserved
11:8	R/W	0x0	EINT2_CFG External INT2 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
7:4	R/W	0x0	EINT1_CFG External INT1 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
3:0	R/W	0x0	EINT0_CFG External INT0 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved

10.7.7.11 PL External Interrupt Configure Register 1 (Default Value: 0x0000_0000)

Offset: 0x0204			Register Name: PL_EINT_CFG1
Bit	**Read/Write**	**Default/Hex**	**Description**
31:20	/	/	/
19:16	R/W	0x0	EINT12_CFG External INT12 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
15:12	R/W	0x0	EINT11_CFG External INT11 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level

			0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
11:8	R/W	0x0	EINT10_CFG External INT10 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
7:4	R/W	0x0	EINT9_CFG External INT9 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved
3:0	R/W	0x0	EINT8_CFG External INT8 Mode 0000: Positive Edge 0001: Negative Edge 0010: High Level 0011: Low Level 0100: Double Edge (Positive/ Negative) Others: Reserved

10.7.7.12 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	/	/	/

10.7.7.13 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	/	/	/

10.7.7.14 PL External Interrupt Control Register (Default Value: 0x0000_0000)

Offset: 0x0210	Register Name: PL_EINT_CTL

Bit	Read/Write	Default/Hex	Description
31:13	/	/	/
12	R/W	0x0	EINT12_CTL External INT12 Enable 0: Disable 1: Enable
11	R/W	0x0	EINT11_CTL External INT11 Enable 0: Disable 1: Enable
10	R/W	0x0	EINT10_CTL External INT10 Enable 0: Disable 1: Enable
9	R/W	0x0	EINT9_CTL External INT9 Enable 0: Disable 1: Enable
8	R/W	0x0	EINT8_CTL External INT8 Enable 0: Disable 1: Enable
7	R/W	0x0	EINT7_CTL External INT7 Enable 0: Disable 1: Enable
6	R/W	0x0	EINT6_CTL External INT6 Enable 0: Disable 1: Enable
5	R/W	0x0	EINT5_CTL External INT5 Enable 0: Disable 1: Enable
4	R/W	0x0	EINT4_CTL External INT4 Enable 0: Disable 1: Enable
3	R/W	0x0	EINT3_CTL External INT3 Enable 0: Disable 1: Enable
2	R/W	0x0	EINT2_CTL External INT2 Enable 0: Disable 1: Enable

1	R/W	0x0	EINT1_CTL External INT1 Enable 0: Disable 1: Enable
0	R/W	0x0	EINT0_CTL External INT0 Enable 0: Disable 1: Enable

10.7.7.15 PL External Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0214			Register Name: PL_EINT_STATUS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:13	/	/	/
12	R/W	0x0	EINT12_STATUS External INT12 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
11	R/W	0x0	EINT11_STATUS External INT11 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
10	R/W	0x0	EINT10_STATUS External INT10 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
9	R/W	0x0	EINT9_STATUS External INT9 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
8	R/W	0x0	EINT8_STATUS External INT8 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
7	R/W	0x0	EINT7_STATUS External INT7 Pending Bit 0: No IRQ pending 1: IRQ pending Write ‘1’ to clear
6	R/W	0x0	EINT6_STATUS

			External INT6 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
5	R/W	0x0	EINT5_STATUS External INT5 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
4	R/W	0x0	EINT4_STATUS External INT4 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
3	R/W	0x0	EINT3_STATUS External INT3 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
2	R/W	0x0	EINT2_STATUS External INT2 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
1	R/W	0x0	EINT1_STATUS External INT1 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear
0	R/W	0x0	EINT0_STATUS External INT0 Pending Bit 0: No IRQ pending 1: IRQ pending Write '1' to clear

10.7.7.16 PL External Interrupt Debounce Register (Default Value: 0x0000_0000)

Offset: 0x0218			Register Name: PL_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

10.7.7.17 PIO Group Withstand Voltage Mode Select Register (Default Value: 0x0000_0000)

Offset: 0x0340			Register Name: PIO_POW_MOD_SEL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:13	/	/	/
12	R/W	0x0	VCC-IO POWER MODE Select 0: 3.3V 1: 1.8V
11:1	/	/	/
0	R/W	0x0	PL_POWER MODE Select 0: 3.3V 1: 1.8V If PL_Port Power Source selects VCC-IO,this bit is invalid

NOTE

For 0x0340 register ,when the power domain of GPIO is larger than 1.8V, then the withstand voltage is set to 3.3V mode, that is, the corresponding register value is set to 0; when the power domain of GPIO is 1.8V,then the withstand voltage is set to 1.8V mode, that is, the corresponding register value is set to 1; when the power domain of GPIO is uncertain, then the withstand voltage is set to 3.3V mode, that is, the corresponding register value is set to 0.

10.7.7.18 PIO Group Power Value Register

Offset: 0x0348			Register Name: PIO_POW_Val
Bit	**Read/Write**	**Default /Hex**	**Description**
31:17	/	/	/
16	R	/	VCC-IO Power Value
15:1	/	/	/
0	R	/	PL_Port Power Value If PL_ Port Power Source selects VCC-IO,this bit is invalid

NOTE

IO pressure mode is 3.3V when reading 0; IO pressure mode is 1.8V when reading 1.

10.8 GPADC

10.8.1 Overview

The GPADC is a 12-bit sampling analog to digital converter, and is a type of successive approximation register (SAR) converter. The GPADC has the following features:

- 8-bit effective SAR type A/D converter
- 12-bit resolution
- 64 FIFO depth of data register
- Power supply voltage: 1.8 V
- Analog input range: 0 V to 1.8 V
- Maximum sampling frequency: 1 MHz
- Supports self calibration
- Supports data compare and interrupt
- Supports three operation modes:
  - Single conversion mode
  - Continuous conversion mode
  - Burst conversion mode

10.8.2 Block Diagram

Figure 10-31 shows the block diagram of the GPADC.

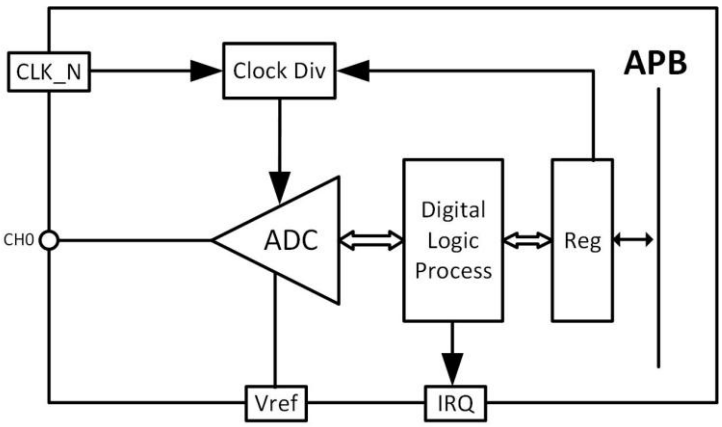


Figure10- 31. GPADC Block Diagram

10.8.3 Operations and Functional Descriptions

10.8.3.1 External Signals

Table 10-17 describes the external signals of GPADC.

Table10- 17. GPADC External Signals

Signal	Description	Type
GPADC	ADC Input	AI

10.8.3.2 Clock Sources

GPADC has one clock source. Table 10-18 describes the clock source for GPADC. Users can see **Clock Controller Unit(CCU)** for clock setting, configuration and gating information.

Table10- 18. GPADC Clock Sources

Clock Sources	Description
OSC24M	24MHz

10.8.3.3 GPADC Timing Requirement

CLK_IN = 24MHz.
CONV_TIME(Conversion Time) = 1/(24MHz/14Cycles) =0.583 (us).
TACQ>10RC. R is ADC sample circuit output impdence, C =6.4pF
ADC Sample Frequency > TACQ+CONV_TIME.

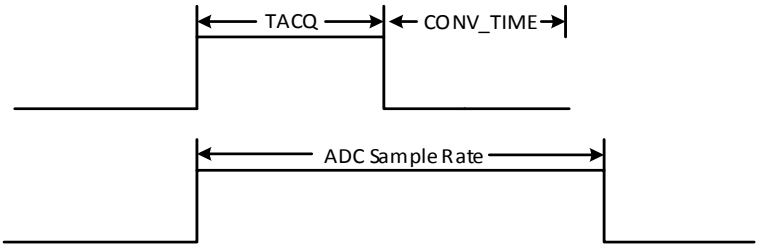


Figure10- 32. GPADC Timing Requirement

10.8.3.4 GPADC Work Mode

(1).Single conversion mode
GPADC completes one conversion in specified channel, the converted data is updated at the data register of corresponding channel.
(2).Continuous conversion mode
GPADC has continuous conversion in specified channel until the software stops, the converted data is updated at the data register of corresponding channel.
(3).Burst conversion mode
GPADC samples and converts in the specified channel, and sequentially stores the results in FIFO. The FIFO supports interrupt controller and the depth of FIFO is 64.

10.8.3.5 GPADC Formula

GPADC formula: Vin = GPADC_DATA * V_{REF} / 4095， where V_{REF}=AVCC.

10.8.4 Programming Guidelines

The GPADC is initialized as follows.

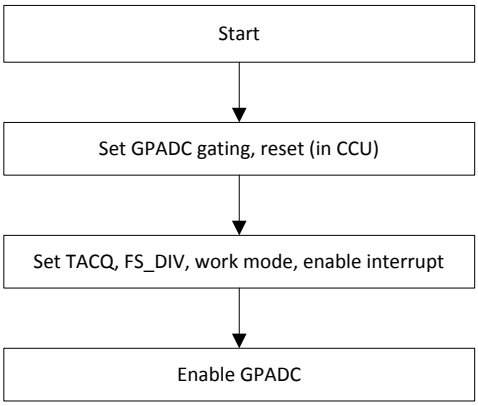


Figure10- 33. GPADC Initialization Process

(1).Query Mode

Step1: Write 1 to the bit[16] of **GPADC_BGR_REG** to dessert reset.

Step2: Write 1 to the bit[0] of **GPADC_BGR_REG** to open GPADC clock.

Step3: Write 0x2F to the bit[15:0] of **GP_SR_CON** to set ADC acquire time.

Step4: Write 0x1DF to the bit[31:16] of **GP_SR_CON** to set ADC sample frequency divider.

Step5: Write 0x2 to the bit[19:18] of **GP_CTRL** to set continuous conversion mode.

Step6: Write 1 to the bit[0] of **GP_CS_EN** to enable analog input channel select.

Step7: Write 1 to the bit[16] of **GP_CTRL** to enable ADC.

Step8: Read the bit[0] of **GP_DATA_INTS**, if the bit is 1, then data conversion is complete.

Step9: Read the bit[11:0] of **GP_CH0_DATA**, calculate voltage value based on GPADC formula.

(2).Interrupt Mode

Step1: Write 1 to the bit[16] of **GPADC_BGR_REG** to dessert reset.

Step2: Write 1 to the bit[0] of **GPADC_BGR_REG** to open GPADC clock.

Step3: Write 0x2F to the bit[15:0] of **GP_SR_CON** to set ADC acquire time.

Step4: Write 0x1DF to the bit[31:16] of **GP_SR_CON** to set ADC sample frequency divider.

Step5: Write 0x2 to the bit[19:18] of **GP_CTRL** to set continuous conversion mode.

Step6: Write 1 to the bit[0] of **GP_CS_EN** to enable analog input channel select.

Step7: Write 1 to the bit[0] of **GP_DATA_INTC** to enable GPADC data interrupt.

Step8: Set GIC interface based on IRQ 32, write 1 to the bit[0] of the **0x03021104** register .

Step9: Put interrupt handler address into interrupt vector table.

Step10: Write 1 to the bit16 of **GP_CTRL** to enable ADC function.

Step11: Read the bit[11:0] of **GP_CH0_DATA** from the interrupt handler, calculate voltage value based on GPADC formula.

10.8.5 Register List

Module Name	Base Address
GPADC	0x05070000

Register Name	Offset	Description
GPADC_SR_CON	0x0000	GPADC Sample Rate Configure Register
GPADC_CTRL	0x0004	GPADC Control Register
GPADC_CS_EN	0x0008	GPADC Compare and Select Enable Register
GPADC_FIFO_INTC	0x000C	GPADC FIFO Interrupt Control Register
GPADC_FIFO_INTS	0x0010	GPADC FIFO Interrupt Status Register
GPADC_FIFO_DATA	0x0014	GPADC FIFO Data Register
GPADC_CDATA	0x0018	GPADC Calibration Data Register
GPADC_DATAL_INTC	0x0020	GPADC Data Low Interrupt Configure Register
GPADC_DATAH_INTC	0x0024	GPADC Data High Interrupt Configure Register
GPADC_DATA_INTC	0x0028	GPADC Data Interrupt Configure Register
GPADC_DATAL_INTS	0x0030	GPADC Data Low Interrupt Status Register
GPADC_DATAH_INTS	0x0034	GPADC Data High Interrupt Status Register
GPADC_DATA_INTS	0x0038	GPADC Data Interrupt Status Register
GPADC_CH0_CMP_DATA	0x0040	GPADC CH0 Compare Data Register
GPADC_CH0_DATA	0x0080	GPADC CH0 Data Register

10.8.6 Register Description

10.8.6.1 GPADC Sample Rate Configure Register (Default Value: 0x01DF_002F)

Offset: 0x0000			Register Name: GPADC_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: 2us

10.8.6.2 GPADC Control Register (Default Value: 0x0000_0000)

Offset: 0x0004			Register Name: GPADC_CTRL
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	R/ W	0x0	ADC_FIRST_DLY ADC First Convert Delay Setting ADC conversion of each channel is delayed by N samples.

23	R/ W	0x1	ADC_AUTOCALI_EN ADC Auto Calibration
22	/	/	/
21:20	R/W	0x0	ADC_OP_BIAS ADC OP Bias Adjust the bandwidth of the ADC amplifier
19:18	R/W	0x0	GPADC Work Mode 00: Single conversion mode 01: Reserved 10: Continuous conversion mode 11: Burst conversion mode
17	R/W	0x0	ADC_CALI_EN ADC Calibration 1: Start Calibration, it is cleared to 0 after calibration
16	R/W	0x0	ADC_EN ADC Function Enable Before enable the bit, configure ADC parameters including the work mode and channel number,etc. 0: Disable 1: Enable
15:0	/	/	/

10.8.6.3 GPADC Compare and Select Enable Register (Default Value: 0x0000_0000)

Offset: 0x0008			**Register Name: GPADC_CS_EN**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:17	/	/	/
16	R/W	0x0	ADC_CH0_CMP_EN Channel 0 Compare Enable 0: Disable 1: Enable
15:1	/	/	/
0	R/W	0x0	ADC_CH0_SELECT Analog Input Channel 0 Select 0: Disable 1: Enable

10.8.6.4 GPADC FIFO Interrupt Control Register (Default Value: 0x0000_1F00)

Offset: 0x000C	**Register Name: GPADC_FIFO_INTC**

Bit	Read/Write	Default/Hex	Description
31:18	/	/	/
17	R/W	0x0	FIFO_OVERRUN_IRQ_EN ADC FIFO Overrun IRQ Enable 0: Disable 1: Enable
16	R/W	0x0	FIFO_DATA_IRQ_EN ADC FIFO Data Available IRQ Enable 0: Disable 1: Enable
15:14	/	/	/
13:8	R/W	0x1F	FIFO_TRIG_LEVEL Interrupt and DMA request trigger level for ADC Trigger Level = TXTL + 1
7:5	/	/	/
4	R/W1C	0x0	FIFO_FLUSH ADC FIFO Flush Write '1' to flush TX FIFO, clear automatically to '0'.
3:0	/	/	/

10.8.6.5 GPADC FIFO Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0010			Register Name: GPADC_FIFO_INTS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:18	/	/	/
17	R/W1C	0x0	FIFO_OVERRUN_PENDING ADC FIFO Over Run IRQ Pending 0: No Pending IRQ 1: FIFO Overrun Pending IRQ Write '1' to clear this interrupt or automatically clear if 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 interrupt condition fails.
15:14	/	/	/
13:8	R	0x0	RXA_CNT ADC FIFO available sample word counter
7:0	/	/	/

10.8.6.6 GPADC FIFO Data Register

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

10.8.6.7 GPADC Calibration Data Register (Default Value: 0x0000_0000)

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

10.8.6.8 GPADC Low Interrupt Configure Register (Default Value: 0x0000_0000)

Offset: 0x0020			Register Name: GPADC_DATAL_INTC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	CH0_LOW_IRQ_EN 0: Disable 1: Enable

10.8.6.9 GPADC High Interrupt Configure Register (Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: GPADC_DATAH_INTC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	CH0_HIG_IRQ_EN 0: Disable 1: Enable

10.8.6.10 GPADC DATA Interrupt Configure Register (Default Value: 0x0000_0000)

Offset: 0x0028			Register Name: GPADC_DATA_INTC
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W	0x0	CH0_DATA_IRQ_EN 0: Disable 1: Enable

10.8.6.11 GPADC Low Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0030			Register Name: GPADC_DATAL_INTS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W1C	0x0	CH0_LOW_PENGDING 1: Channel 0 Voltage Low Available Pending IRQ Write '1' to clear this interrupt or automatic clear if interrupt condition fails

10.8.6.12 GPADC High Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0034			Register Name: GPADC_DATAH_INTS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
0	R/W1C	0x0	CH0_HIG_PENGDING 0: NO Pending IRQ 1: Channel 0 Voltage High Available Pending IRQ Write '1' to clear this interrupt or automatic clear if interrupt condition fails

10.8.6.13 GPADC Data Interrupt Status Register (Default Value: 0x0000_0000)

Offset: 0x0038			Register Name: GPADC_DATA_INTS
Bit	**Read/Write**	**Default/Hex**	**Description**
31:1	/	/	/
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 automatic clear if interrupt condition fails

10.8.6.14 GPADC CH0 Compare Data Register (Default Value: 0x0BFF_0400)

Offset: 0x0040			Register Name: GPADC_CH0_CMP_DATA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:28	/	/	/
27:16	R/W	0xBFF	CH0_CMP_HIG_DATA Channel 0 Voltage High Value
15:12	/	/	/
11:0	R/W	0x400	CH0_CMP_LOW_DATA Channel 0 Voltage Low Value

10.8.6.15 GPADC CH0 Data Register (Default Value: 0x0000_0000)

Offset: 0x0080			Register Name: GPADC_CH0_DATA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:12	/	/	/
11:0	R	0x000	GP_CH0_DATA Channel 0 Data

10.9 LRADC

10.9.1 Overview

LRADC is 6-bit resolution ADC for Key application. The LRADC can work up to maximum conversion rate of 2KHz.

Features:

- Power Supply Voltage:1.8V, Reference Voltage:1.35V
- Interrupt Support
- Support Hold Key and General Key
- Support normal、continue and single work mode
- 6-bits Resolution, Sample Rate up to 2KHz
- Voltage input range between 0 to 1.35V

10.9.2 Block Diagram

Figure 10-34 shows a block diagram of the LRADC.

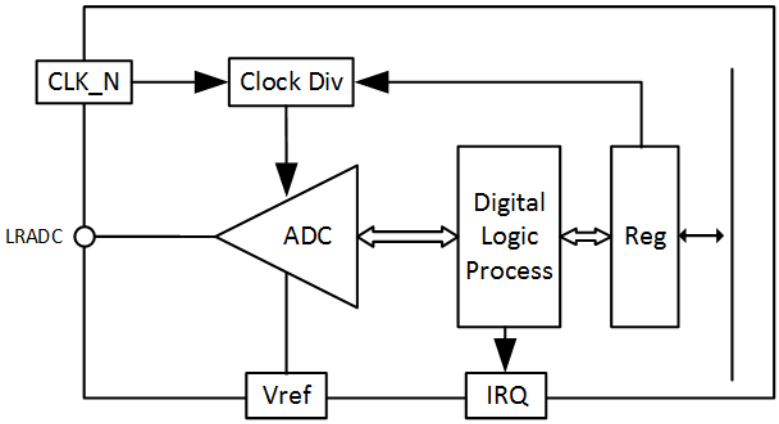


Figure10- 34. LRADC Block Diagram

10.9.3 Operations and Functional Descriptions

10.9.3.1 External Signals

Table 10-19 describes the external signal of LRADC.

Table10- 19. LRADC External Signal

Signal	Description	Type
LRADC	Analog Input	Analog

10.9.3.2 Clock Sources

Table 10-20 describes the clock source for LRADC.

Table10- 20. LRADC Clock Sources

Clock Sources	Description
LOSC	32.768KHz LOSC

10.9.3.3 LRADC Work Mode

(1) Normal Mode

ADC gathers 8 samples, the average of the 8 samples is updated in data register,and the data interrupt sign is enabled.It is sampled repeatedly according to this mode until ADC stop.

(2)Continue Mode

ADC gathers 8 samples every other 8*(N+1) sample cycle. The average 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_REG**).

(3)Single Mode

ADC gathers 8 samples, the average of the 8 samples is updated in data register, and the data interrupt sign is enabled, since then ADC stops sample.

10.9.3.4 Interrupt

Each channel has 5 interrupt sources and 5 interrupt enable controls, As shown in figure 10-35.

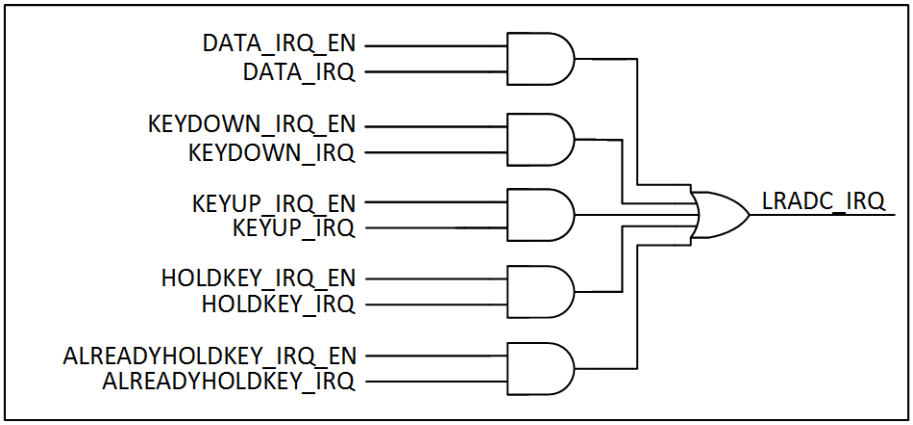


Figure10- 35. LRADC Interrupt Source

IRQ1 is generated when the input voltage is between LEVELA(LEVELA=1.35V) and LEVELB(LRADC_CTRL bit 5:4). IRQ2 is generated when the input voltage is lower than LEVELB. The control logic will generate Hold KEY interrupt when the control logic receives IRQ1 and does not receive IRQ2 for several time(LEVELA_B_CNT), otherwise DATA_IRQ interrupt is generated. Hold KEY is generally used for self-locking button. When self-locking button is locked and the control logic receives IRQ2, the control logic will generate **Already Hold KEY** interrupt

10.9.4 Programming Guide

LRADC Initial Process

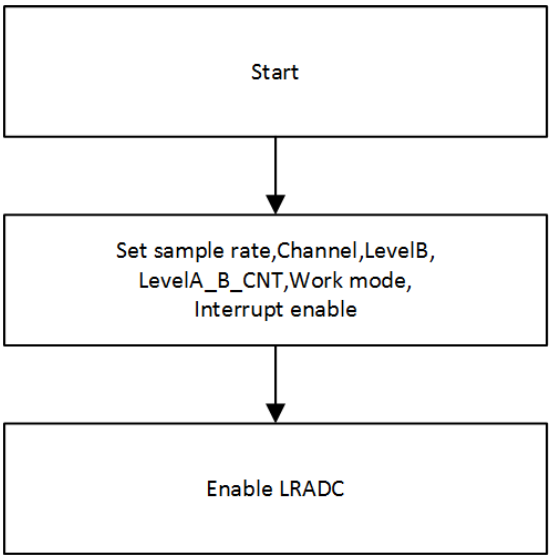


Figure10- 36. LRADC Initial Process

- Set CONTINUE_TIME_SELECT when LRADC works in Continue mode.
- The range of input voltage is from 0 to LEVELB (LRADC_CTRL bit 5:4).
- Calculation formula: LRADC_DATA = Vin/V_{REF}*63, V_{REF}=1.35V
- LRADC has 6-bit resolution, 1-bit offset error, 1-bit quantizing error. After LRADC calibrates 1-bit offset error,LRADC has 5-bit resolution.

10.9.5 LRADC Register List

Module Name	Base Address
LRADC	0x05070800

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_DATA0	0x000C	LRADC Data Register0
LRADC_REV	0x0100	LRADC Revision Register

10.9.6 LRADC Register Description

10.9.6.1 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
23:20	/	/	/
19:16	R/W	0x0	CONTINUE_TIME_SELECT

			Continue 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: Continue Mode
11:8	R/W	0x1	LEVELA_B_CNT Level A to Level B time threshold select, judge ADC convert value in 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: 0x3C (1.286V) 01: 0x39 (1.221V) 10: 0x36 (1.157V) 11: 0x33 (1.093V)
3: 2	R/W	0x2	LRADC_SAMPLE_RATE LRADC Sample Rate 00: 2KHz 01: 1KHz 10: 500Hz 11: 250Hz
1	/	/	/
0	R/W	0x0	LRADC_EN LRADC enable 0: Disable 1: Enable

10.9.6.2 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

10.9.6.3 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 general Key pull up, it the corresponding interrupt is enabled. 0: No IRQ 1: IRQ Pending Notes: 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 hold Key pull down and pull the general key down, if the corresponding interrupt is enabled. 0: No IRQ 1: IRQ Pending Notes: 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 Hold Key pull down, the status bit is set and the interrupt line is set if the corresponding interrupt is enabled. 0: NO IRQ 1: IRQ Pending Notes: 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 General Key pull down, the status bit is set and the interrupt line is set if the corresponding interrupt is enabled. 0: No IRQ 1: IRQ Pending Notes: Writing 1 to the bit will clear it and its corresponding interrupt if the interrupt is enabled.
0	R/W1C	0x0	ADC0_DATA_PENDING ADC0 Data IRQ Pending Bit 0: No IRQ 1: IRQ Pending Notes: Writing 1 to the bit will clear it and its corresponding interrupt if the interrupt is enabled.

10.9.6.4 LRADC Data Register0 (Default Value: 0x0000_003F)

Offset: 0x000C			**Register Name: LRADC_DATA0**
Bit	Read/Write	Default/Hex	Description
31:6	/	/	/
5:0	R	0x3F	LRADC0_DATA LRADC0 Data

10.10 PWM

10.10.1 Overview

The PWM is pulse width modulation module in the system domain. The PWM has the following features:

- Supports single pulse and cycle pulse output
- Supports frequency range from 0 to OSC (depend on the frequency of input oscillator : 24MHz / 19.2MHz / 38.4MHz)
- Adjustable duty cycle from 0% to 100%
- The minimum resolution is 1/65536

10.10.2 Block Diagram

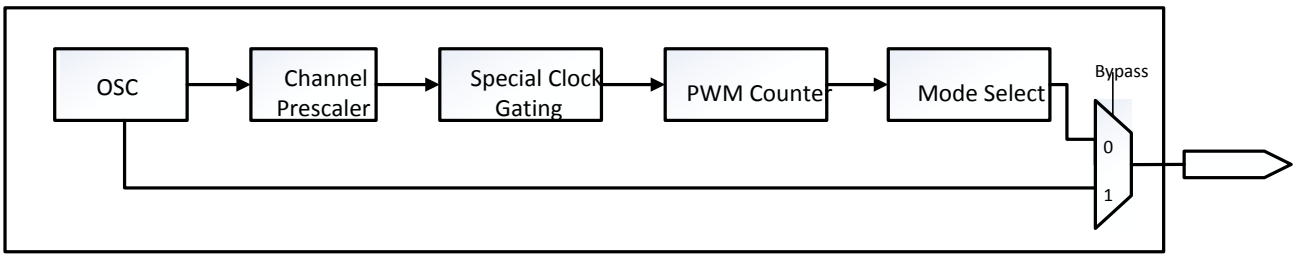


Figure10- 37. PWM Block Diagram

10.10.3 Operations and Functional Description

10.10.3.1 External Signal

The external signal of PWM has multiplexing function with other I/O pin. Configure multiplexing function of GPIO(General Purpose Input Output) by software in Port Controller module to use PWM.

10.10.3.2 Clock and Reset

The clock source of PWM is OSC24M, OSC19.2M, or OSC38.4MHz (depending on frequency of input oscillator). The PWM is hung on APB1 Bus. Ensure that open APB1 Bus gating and de-assert reset signal when accessed to the PWM.

10.10.3.3 Typical Application

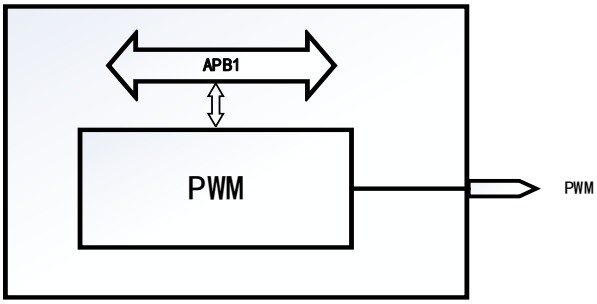


Figure10- 38. PWM Application Diagram

10.10.3.4 Function Implemention

10.10.3.4.1 Clock Control

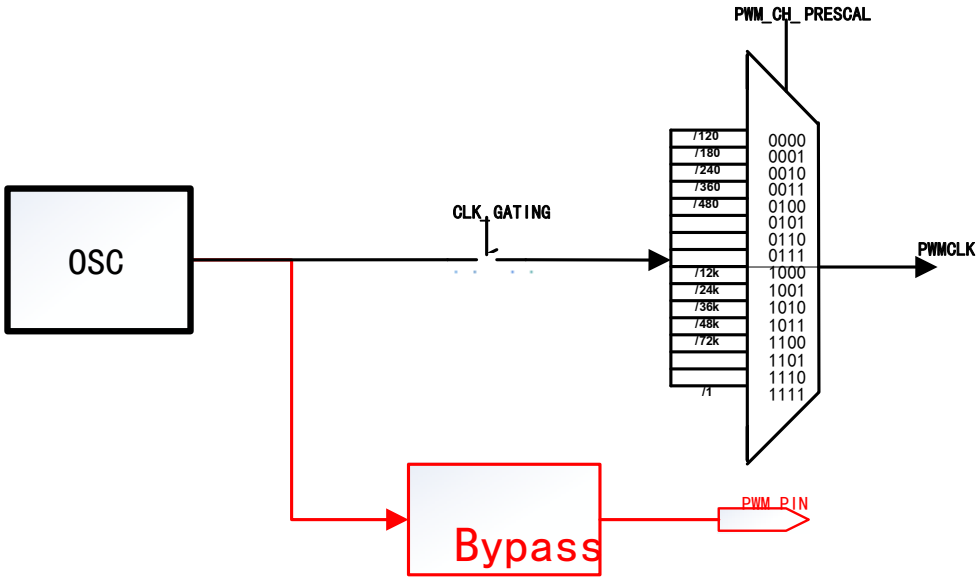


Figure10- 39. PWM Clock Control

The clock controller part of PWM includes clock switch (CLK_GATING), prescale factor selecting (PWM_CH_PRESCAL) and clock source bypass (PWM_BYPASS). Open CLK_GATING and configure prescale factor before used clock source, the output clock(PWMCLK) is delivered to the PWM module. Clock source is bypassed directly to the PWM output pin by clock source bypass part.

10.10.3.4.2 Output Mode

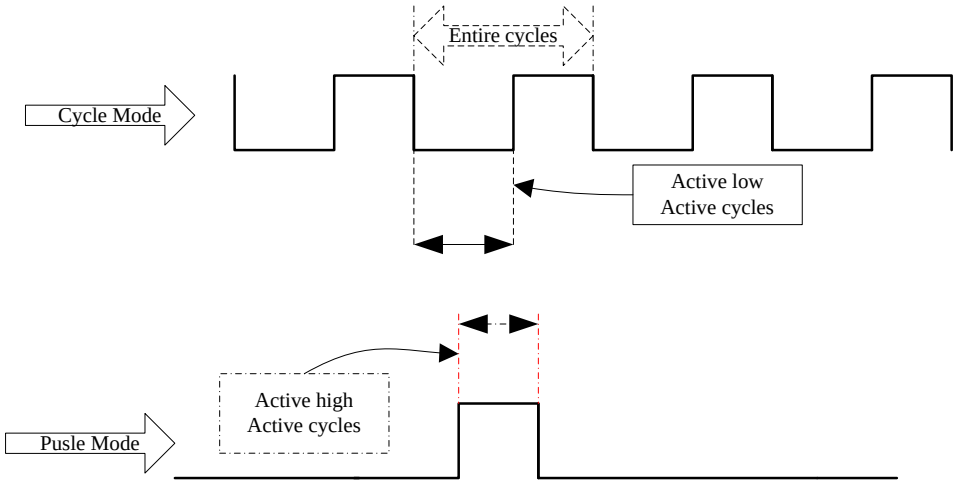


Figure10- 40. PWM Output Mode

PWM supports cycle mode and single pulse mode.

Cycle mode: When the value of up-counter reaches Entire Cycles, the value is decreased to 0 , PWM continues to count and outputs a continuous waveform.

Single pulse mode: When the value of up-counter reaches Entire Cycles, the value is decreased to 0, PWM stops counting and outputs a single pulse waveform.

10.10.3.4.3 Output Parameter

The period ,duty cycle and active state of PWM output waveform are decided by up-counter and comparator. The rule of comparator is as follows.

Cycle mode

Counter > (PWM_CH_Entire_Cycles - PWM_CH_Active_Cycles), output active state

Counter <= (PWM_CH_Entire_Cycles – PWM_CH_Active_Cycles), output inactive state

Active state can be set to high level or low level.

Output waveform period = （OSC/PWM_CH_PRESCAL）$^{-1}$ *PWM_CH_Entire_Cycles

Output active state time = （OSC/PWM_CH_PRESCAL）$^{-1}$ *PWM_CH_Active_Cycles

Output inactive state time = （OSC/PWM_CH_PRESCAL）$^{-1}$ * （PWM_CH_Entire_Cycles - PWM_CH_Active_Cycles）

Duty Cycle = (PWM_CH_Active_Cycles /PWM_CH_Entire_Cycle) * 100%

Single pulse mode

Pulse width = （OSC/PWM_CH_PRESCAL）$^{-1}$ *PWM_CH_ENTIREe_Active_Cycle

10.10.3.5 Operating Mode

10.10.3.5.1 Clock Configuration

Clock source bypass: After the bit 9 of **PWM_CTRL_REG** is enabled, PWM port can output OSC clock.
Clock Gating: Enable the bit6 of **PWM_CTRL_REG** to use PWM clock.
Prescale factor: Set the bit[3:0] of **PWM_CTRL_REG** to select corresponding prescale factor.

10.10.3.5.2 PWM Parameter

Select mode: Select cycle mode or pulse mode by the bit7 of **PWM_CTRL_REG**.
Output level: Select low level or high level by the bit5 of **PWM_CTRL_REG**.
Initial value: Set **PWM_CH_ENTIRE_ACT_CYS** and **PWM_CH_ENTIRE_CYS** by **PWM_CH_PERIOD**.

NOTE

The active cycles in PWM_CH_PERIOD must be less than or equal to entire cycles.
Check the bit28 of PWM_CTRL_REG before writing PWM_CH_PERIOD.

10.10.4 Programming Guidelines

For example,to output a cycle mode waveform with 5ms period and 50% duty ratio (active state is high level), performs the following steps.

Step1: Clock and parameter configuration: configure corresponding bit of **PWM_CTRL_REG** and **PWM_CH0_PERIOD** to set clock source bypass, clock gating enable, channel period, prescale factor, output mode and level, etc.

Step2: Enable PWM0.

10.10.5 Register List

Module Name	Base Address
PWM	0x0300 A000

Register Name	Offset	Description
PWM_CTRL_REG	0x0000	PWM Control Register
PWM_CH0_PERIOD	0x0004	PWM Channel 0 Period Register
PWM_CH1_PERIOD	0x0008	PWM Channel 1 Period Register

10.10.6 Register Description

10.10.6.1 PWM Control Register(Default Value: 0x0000_0000)

Offset: 0x0000			Register Name: PWM_CTRL_REG
Bit	**Read/Write**	**Default/Hex**	**Description**
31:30	/	/	/
29	R	0x0	PWM1_RDY PWM1 period register ready 0: PWM1 period register is ready to write 1: PWM1 period register is busy
28	R	0x0	PWM0_RDY PWM0 period register ready 0: PWM0 period register is ready to write 1: PWM0 period register is busy
27:25	/	/	/
24	R/W	0x0	PWM1_BYPASS PWM channel1 bypass enable If the bit is set to 1, PWM1's output is OSC. 0: Disable 1: Enable
23	R/W	0x0	PWM_CH1_PULSE_OUT_START PWM channel 1 pulse output start 0: No effect 1: Output 1 pulse. The pulse width should be according to the period 1 register[15:0],and the pulse state should be according to the active state. After the pulse is finished,the bit will be cleared automatically.
22	R/W	0x0	PWM_CH1_MODE PWM channel 1 mode 0: Cycle mode 1: Pulse mode
21	R/W	0x0	PWM_CH1_CLK_GATING Gating the Special Clock for PWM1 0: Mask 1: Pass
20	R/W	0x0	PWM_CH1_ACT_STATE. PWM channel 1 active state. 0: Low Level 1: High Level
19	R/W	0x0	PWM_CH1_EN PWM channel 1 enable 0: Disable 1: Enable
18:15	R/W	0x0	PWM_CH1_PRESCAL

			PWM channel 1 prescalar These bits should be setting before the PWM Channel 1 clock gate on. 0000: /120 0001: /180 0010: /240 0011: /360 0100: /480 0101: / 0110: / 0111: / 1000: /12k 1001: /24k 1010: /36k 1011: /48k 1100: /72k 1101: / 1110: / 1111: /1
14:10	/	/	/
9	R/W	0x0	
8	R/W	0x0	PWM_CH0_PUL_START PWM channel 0 pulse output start 0: No effect 1: Output 1 pulse. The pulse width should be according to the period 0 register[15:0],and the pulse state should be according to the active state. After the pulse is finished,the bit will be cleared automatically.
7	R/W	0x0	PWM_CHANNEL0_MODE 0: Cycle mode 1: Pulse mode
6	R/W	0x0	SCLK_CH0_GATING Gating the Special Clock for PWM0 0: Mask 1: Pass
5	R/W	0x0	PWM_CH0_ACT_STA PWM channel 0 active state 0: Low Level 1: High Level
4	R/W	0x0	PWM_CH0_EN PWM channel 0 enable 0: Disable 1: Enable
3:0	R/W	0x0	PWM_CH0_PRESCAL PWM channel 0 prescalar These bits should be setting before the PWM channel 0 clock gate on. 0000: /120

			0001: /180 0010: /240 0011: /360 0100: /480 0101: / 0110: / 0111: / 1000: /12k 1001: /24k 1010: /36k 1011: /48k 1100: /72k 1101: / 1110: / 1111: /1

10.10.6.2 PWM Channel 0 Period Register

Offset: 0x0004			Register Name: PWM_CH0_PERIOD
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	R/W	UDF	PWM_CH0_ENTIRE_CYS Number of the entire cycles in the PWM clock. 0 : 1 cycle 1: 2 cycles …… N: N+1 cycles If the register need to be modified dynamically, the PCLK should be faster than the PWM CLK(PWM CLK = OSC/pre-scale).
15:0	R/W	UDF	PWM_CH0_ENTIRE_ACT_CYS Number of the active cycles in the PWM clock. 0: 0 cycle 1: 1 cycles …… N : N cycles

NOTE

The active cycles should be no larger than the period cycles.

10.10.6.3 PWM Channel 1 Period Register

Offset: 0x0008			Register Name: PWM_CH1_PERIOD
Bit	**Read/Write**	**Default/Hex**	**Description**
31:16	R/W	UDF	PWM_CH1_ENTIRE_CYS Number of the entire cycles in the PWM clock.

			0 : 1 cycle 1: 2 cycles …… N: N+1 cycles If the register need to be modified dynamically, the PCLK should be faster than the PWM CLK(PWM CLK = OSC/pre-scale).
15:0	R/W	UDF	PWM_CH1_ENTIRE_ACT_CYS Number of the active cycles in the PWM clock. 0: 0 cycle 1: 1 cycles …… N : N cycles

Figures

11 Security System

11.1 Crypto Engine

11.1.1 Overview

The Crypto Engine(CE) module is one encryption/decryption algorithms accelerator. It supports kinds of symmetric, asymmetric, Hash, and RNG algorithms. There are two software interfaces for secure and non-secure world each. Algorithm control information is written in memory by task descriptor, then CE automatically reads it when executing request. It supports parallel requests from 4 channels each world, and has an internal DMA controller to transfer data between CE and memory. It supports parallel running for symmetric, Hash, asymmetric algorithms.

The CE module has the following features.

- Supports Symmetrical Algorithm: AES, DES, 3DES, XTS
- Supports Hash Algorithm: MD5, SHA, HMAC
- Supports Public Key Algorithm: RSA, ECC
- Supports RNG Algorithm: PRNG, TRNG
- Supports 128-bit, 192-bit and 256-bit key size for AES
- Supports 256-bit, 512-bit key for XTS
- AES supports ECB, CBC, CTR, CTS, CFB, OFB, CBC-MAC modes
- AES-CFB mode supports CFB1, CFB8, CFB64, CFB128
- AES-CTR supports CTR16, CTR32, CTR64, CTR128
- DES supports ECB, CBC, CTR, CBC-MAC mode
- DES-CTR supports CTR16, CTR32, CTR64 mode
- Supports SHA1, SHA224, SHA256, SHA384, SHA512 for SHA. Supports HMAC-SHA1, HMAC-SHA256 for HMAC. Supports multi-package mode for Hash algorithm
- MD5, SHA, HMAC are padded using hardware, if not last package, input should aligned with computation block, namely 512 bits or 1024 bits
- RSA supports 512/1024/2048/3072/4096 bits width
- ECC supports 160/224/256/384/521 bits width
- Supports 160 bits hardware PRNG with 175 bits seed. Output aligns with 5 words
- Supports 256 bits hardware TRNG. Output aligns with 8 words
- Supports secure and non-secure interfaces respectively, each world issues task request through its own interface
- Each world has 4 channels for software request, each channel has an interrupt control and status bit, and channels are independent with each other
- Supports task chain mode for each request. Task or task chain are executed at request order
- Symmetric, asymmetric, HASH ctrl logics are separate, but them can handle task simultaneously. Symmetric logic can select instantiate 2 suits at implementation time

- 8 scatter group(sg) are supported for both input and output data. sg size is input/output word number. DMA reads and write at word aligned
- DMA has multiple channel, each corresponding to one suit of algorithms

11.1.2 Block Diagram

The following figure shows the block diagram of Crypto Engine.

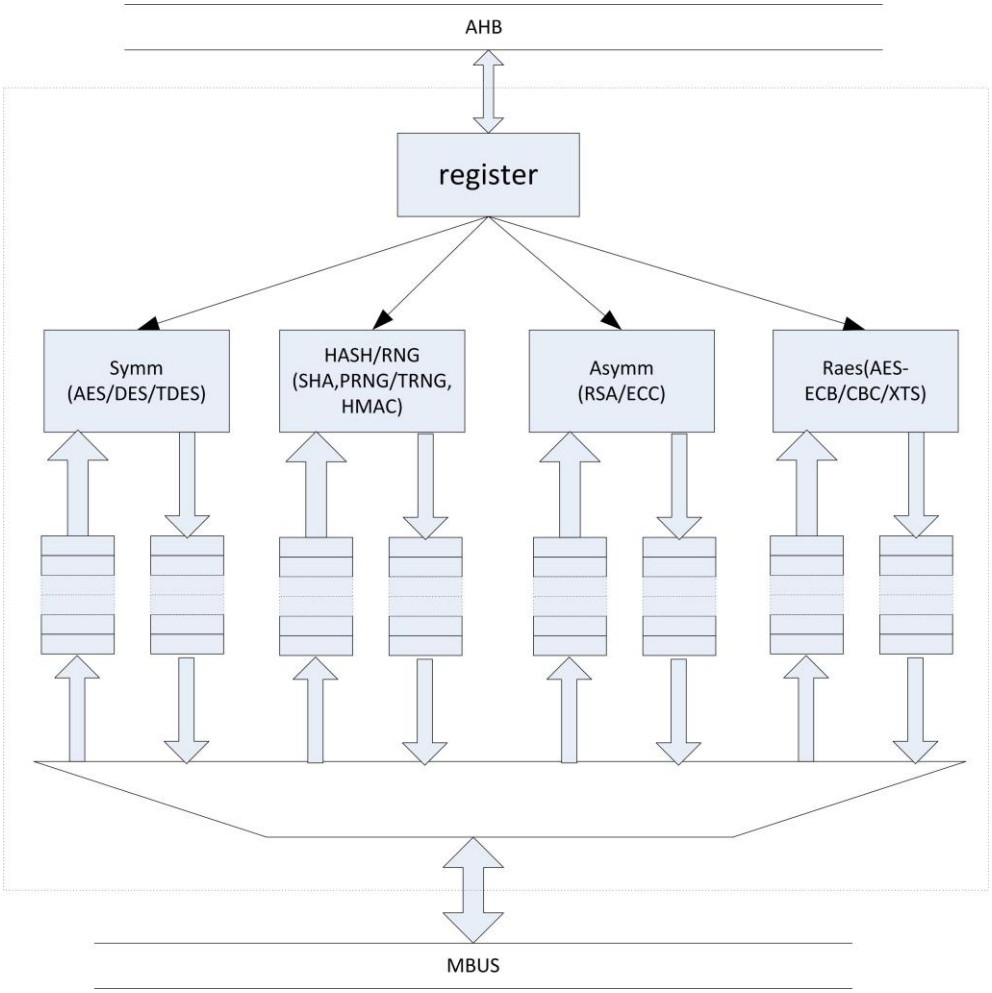


Figure11- 1. CE Block Diagram

Symm: symmetric; Asymm: asymmetric; Raes: AES mode.

Encryption/decryption algorithms are divided into 3 types: symmetric, HASH/RNG, asymmetric.

CE supports 4 channels in each world, and 3 algorithm types, each channel has its own tx and rx fifo,so the four channels can run in parallel.

CE module has AHB interface for task configuration and starting through registers with task descriptor mode.

There is a MBUS interface for data transfer. Reading task description, reading source data, writing destination data by using the interface.

11.1.3 Operations and Functional Descriptions

11.1.3.1 Crypto Engine Task Descriptor

Software make request through task descriptor, providing algorithm type, mode, key address, source/destination address and size, etc. The task descriptor is as follows.

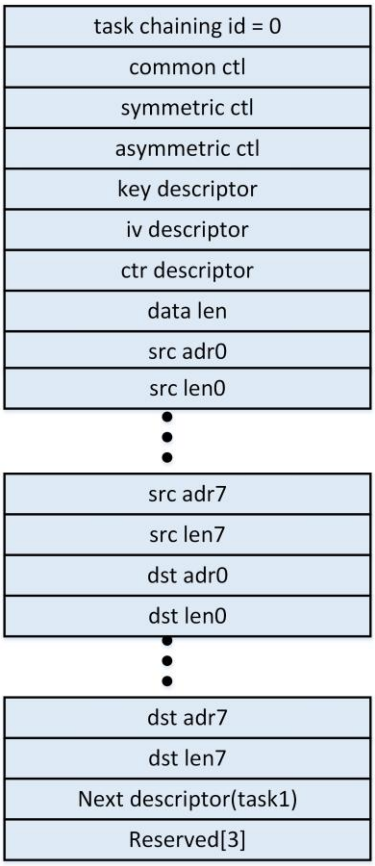


Figure11- 2. Crypto Engine Task Chaining

Task chaining id supports 0~3.

11.1.3.2 Task_descriptor_queue Common Control

Bit	Read/Write	Default/Hex	Description
31	R/W	0x0	Interrupt enable for current task 0: disable interrupt 1: enable interrupt
30:25	/	/	/
24:17	R/W	0x0	CBC_MAC_LEN The outcome bit length of CBC-MAC when in CBC-MAC mode.
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	HASH/HMAC plaintext last 0: not the last HASH/HMAC plaintext package, need not padding 1: the last HASH/HMAC plaintext package, need to padding
14:9	/	/	/
8	R/W	0x0	OP DIR Algorithm Operation Direction 0: Encryption 1: Decryption
7	/	/	/
6:0	R/W	0x0	Algorithm type 0x0: AES 0x1: DES 0x2: Triple DES (3DES) 0x3~0xf: reserved 0x10: MD5 0x11: SHA-1 0x12: SHA-224 0x13: SHA-256 0x14: SHA-384 0x15: SHA-512 0x16: HMAC-SHA1 0x17: HMAC-SHA256 0x18~0x1b: reserved 0x1c: TRNG 0x1d: PRNG 0x20: RSA 0x21: ECC 0x30: RAES Others: reserved

11.1.3.3 Task_descriptor_queue 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)
19:18	R/W	0x0	CFB_WIDTH For AES-CFB width 00: CFB1

			01: CFB8 10: CFB64 11: CFB128
17	R/W	0x0	PRNG_LD Load new 15bits key into lfsr for PRNG
16	R/W	0x0	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 >128bit).
15:14	/	/	/
13	R/W	0x0	XTS_LAST 0: not last block for XTS 1: last block for XTS
12	R/W	0x0	XTS_FIRST 0: not first block for XTS 1: first block for XTS
11:8	R/W	0x0	AES/DES/3DES/RAES modes. DES/3DES only supports ECB/CBC/CTR. RAES only supports ECB/CBC/XTS. operation mode for symmetric 0000: Electronic Code Book (ECB) mode 0001: Cipher Block Chaining (CBC) mode 0010: Counter (CTR) mode 0011: CipherText Stealing (CTS) mode 0100: Output feedback (OFB)mode 0101: Cipher feedback (CFB)mode 0110: CBC-MAC mode 1001: XTS 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
1:0	R/W	0x0	AES Key Size 00: 128-bit 01: 192-bit 10: 256-bit 11: Reserved

11.1.3.4 Task_descriptor_queue Asymmetric Control

Bit	Read/Write	Default/Hex	Description
31:21	/	/	/
20:16	R/W	0x0	PKC algorithm mode.

			For modular computation: 00000: modular exponent(RSA) 00001: modular add 00010: modular minus 00011: modular multiplication others: reserved For ECC: 00000: point add 00001: point double 00010: point multiplication 00011: point verification 00100: encryption 00101: decryption 00110: sign 00111: sign verify others: reserved
15:8	/	/	/
7:0	R/W	0x0	Asymmetric algorithms operation width field. It indicates how much width this request apply, as words.

key addr field is address for each algorithm's key, and for the total length address of Hash when last package, also for extension feature micro codes address. When indicate HASH's total length address, it must be 64bits data length with 64bits address align.

iv addr field is address for IV or modulus, or tweak value address for XTS.

ctr addr is address for next block's IV, and for HMAC K1 address.

src/dst sgX addr field indicates 32-bit address for source and destination data.

src/dst sgX size field indicates size for each sg respectively.

next task field should be set to 0 when no next task, else set to next task's descriptor.

11.1.3.5 Task Request

There are 4 steps for one task handling from software.

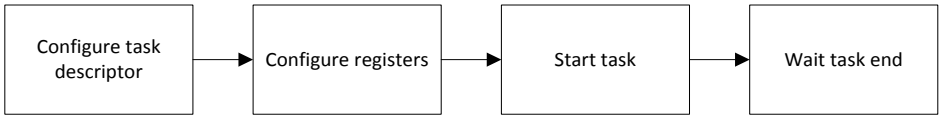


Figure11- 3. Task Request Process

Firstly, software should configure task descriptor in memory, including all fields in descriptor. Channel id corresponds to one channel in CE, from 0 to 3 for secure and non secure world respectively. According to algorithm type, 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. Source and destination sg address and size are set based on upper application. If there is another task concatenating after this task, then set its descriptor address at next descriptor field.

Secondly, software should set registers, including task descriptor address, interrupt control.

Thirdly, software read load register to ensure that the bit0 is zero, then starts request by pulling up the bit0 of the load register.

Lastly, wait interrupt status.

11.1.3.6 Data Length Setting

For HASH algorithm, data length field indicates valid source data bit number; for others indicates source data byte number.
For PRNG, data length should be 20 bytes aligned.
For TRNG should be 32 bytes aligned.
For data length of HASH algorithm should be 512/1024 bits aligned if current request is not the last data block, because of hardware padding.
Data size in source and destination sg is as words, whose value should corresponds with data length field, or else CE will report error and stop execution.

11.1.3.7 Security Operation

When CPU issues request to CE module, CE module will save the secure mode of CPU. When executing this request, this state bit works as access tag for inner and system resource. For HUK/RSSK/SSK from SID, only secure mode can access, or else these keys will be used as 0. For access to SID and keysram module through AHB bus, only secure mode can success, or else will read 0 or can not write. When issuing MBUS read and write requests, CE will use send this secure mode bit to BUS, so secure request can access secure and non secure space, but non secure request only can access non secure space.

11.1.3.8 Parallel Task

Algorithms are divided into 3 types: symmetric, HASH/RNG, asymmetric. Each type has a task queue with 8 elements for requests. Tasks in each queue are handled in sequence. Among these 3 types, task request and complete time are not sure. If one type uses the outcome of another type, software should make sure that start one type after another type is finished.
CE supports 4 channels in each world, and 3 algorithm types which can run in parallel. When software issues request, it first checks if load bit is low which means software can request. If load bit is high which means last request is not registered by CE, software should wait until load bit is low. If software makes several requests with the same type, these tasks will be executed in request sequence. If software makes several requests with different types, these tasks will be executed in parallel. Because parallel tasks would finish out of order, software should make different type request with different channel id, which results in generating different interrupt status bit.

11.1.3.9 PKC Microcode

PKC module supports RSA, ECC asymmetric algorithms in the form of microcode. It implements basic modular add, minus, multiplication, point add, point double, and logic computing, etc. Complete RSA/ECC encryption, decryption, sign, verify are implemented with these microcode.
Asymmetric algorithms RSA/ECC are implemented as microcode in PKC module. Asymmetric encryption, decryption, signature and verification operations are composed with certain fixed microcode with hardware.

11.1.3.10 PKC Configuration

Before starting PKC, task description must be configured. Parameters to PKC are assigned to source sg, outcome is put to destination sg.
For RSA, parameters should be at the order of key, modulus, plaintext, for example key assigned to source sg 0, modulus assigned to source sg 1, plaintext assigned to source sg 2.
For ECC point add P2 = P0 + P1, parameters should be at the order of p, a, P0x, P0y, P1x, P1y. Output is at the order of P2x, P2y.
For ECC point double P2 = 2*P0, parameters should be at the order of p, a, P0x, P0y. Output is at the order of P2x, P2y.
For ECC point multiplication P2 = k*P0, parameters should be at the order of p, k, P0x, P0y. Output is at the order of P2x, P2y.
For ECC point verification, parameters should be at the order of p, a, P0x, P0y, b. Output is 1 or 0.
For ECC encryption, parameters should be at the order of random k, p, a, Gx, Gy, Qx, Qy, m. Output is at the order of Rx, Ry, c.
For ECC decryption, parameters should be at the order of random k, p, a, Rx, Ry, c. Output is m.
For ECC signature, parameters should be at the order of random k, p, a, Gx, Gy, n, d, e. Output is at the order of r, s.
For ECC signature verification, parameters should be at the order of n, s, e, r, p, a, Gx, Gy, Qx, Qy, n, r. Output is 1 or 0.

11.1.3.11 Error Check

CE module includes error detection for task configuration, data computing error, and authentication invalid. When algorithm type in task description is read into module, CE will check if this type is supported through checking algorithm type field in common ctrl. If type value is out of scope, CE will issue interrupt signal and set error state. Each type has certain input and output data size. After getting task descriptor, input size and output size configuration will be checked to avoid size error. If size configuration is wrong, CE will issue interrupt signal and set error state. To protect keys be put into keysram from disclose, if request using RSSK is for AES decryption, and destination address is not in keysram space, CE would not execute this task. It will issue interrupt signal and set error state.

11.1.3.12 Clock Requirement

Clock Name	Description	Requirement
ahb_clk	AHB bus clock	24MHz ~ 200MHz
m_clk	MBUS clk	24MHz ~ 400MHz
ce_clk	CE work clock	24MHz ~ 300MHz

11.1.4 Register List

Module Name	Base Address
CE_NS	0x01904000
CE_S	0x01904800

Register Name	Offset	Description
CE_TDA	0x0000	Task Descriptor Address

CE_ICR	0x0008	Interrupt Control Register
CE_ISR	0x000C	Interrupt Status Register
CE_TLR	0x0010	Task Load Register
CE_TSR	0x0014	Task Status Register
CE_ESR	0x0018	Error Status Register
CE_SCSA	0x0024	Symmetric Algorithm DMA Current Source Address
CE_SCDA	0x0028	Symmetric Algorithm DMA Current Destination Address
CE_HCSA	0x0034	HASH Algorithm DMA Current Source Address
CE_HCDA	0x0038	HASH Algorithm DMA Current Destination Address
CE_ACSA	0x0044	Asymmetric Algorithm DMA Current Source Address
CE_ACDA	0x0048	Asymmetric Algorithm DMA Current Destination Address
CE_XCSA	0x0054	XTS Algorithm DMA Current Source Address
CE_XCDA	0x0058	XTS Algorithm DMA Current Destination Address

11.1.5 Register Description

11.1.5.1 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

11.1.5.2 CE Interrupt Control Register(Default Value: 0x0000_0000)

Offset: 0x0008			**Register Name: CE_ICR**
Bit	**Read/Write**	**Default/Hex**	**Description**
31:4	/	/	/
3:0	R/W	0x0	Task channel0-3 interrupt enable 0: interrupt disable 1: interrupt enable

11.1.5.3 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 channel0-3 end pending 0: not finished 1: finished It indicates if task has been completed . Write ‘1’ to clear it.

11.1.5.4 CE Task Load Register(Default Value: 0x0000_0000)

Offset: 0x0010			Register Name: CE_TLR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:15	/	/	/
14:8	R/W	0x0	Algorithm type, the same with type field in description common control.
7:1	/	/	/
0	R/W	0x0	Task Load When setting, CE can load the descriptor of task if task FIFO is not full.

11.1.5.5 CE Task Status Register(Default Value: 0x0000_0000)

Offset: 0x0014			Register Name: CE_TSR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:26	/	/	/
25:24	R	0x0	Indicate which channel in run for XTS algorithm. 00: task channel0 01: task channel1 10: task channel2 11: task channel3
23:18	/	/	/
17:16	R	0x0	Indicate which channel in run for asymmetric algorithm. 00: task channel0 01: task channel1 10: task channel2 11: task channel3
15:10	/	/	/
9:8	R	0x0	Indicate which channel in run for digest algorithm. 00: task channel0 01: task channel1 10: task channel2 11: task channel3
7:2	/	/	/
1:0	R	0x0	Indicate which channel in run for symmetric algotithm. 00: task channel0 01: task channel1 10: task channel2 11: task channel3

11.1.5.6 CE Error Status Register(Default Value: 0x0000_0000)

Offset: 0x0018			Register Name: CE_ESR
Bit	**Read/Write**	**Default/Hex**	**Description**
31:24	R/W1C	0x0	Task channel 3 error type. (the same for other channels)

			Bit 24: algorithm not support Bit 25: data length error Bit 26: keysram access error. Write '1' to clear. Bit 29: address invalid other: reserved
23:16	R/W1C	0x0	Task channel 2 error type. Bit 16: algorithm not support Bit 17: data length error Bit 18: keysram access error. Write '1' to clear. Bit 21: address invalid other: reserved
15:8	R/W1C	0x0	Task channel 1 error type. Bit 8: algorithm not support Bit 9: data length error Bit 10: keysram access error. Write '1' to clear. Bit 13: address invalid other: reserved
7:0	R/W1C	0x0	Task channel 0 error type. Bit 0: algorithm not support Bit 1: data length error Bit 2: keysram access error. Write '1' to clear. Bit 5: address invalid other: reserved

11.1.5.7 CE Symmetric Current Source Address Register(Default Value: 0x0000_0000)

Offset: 0x0024			Register Name: CE_SCSA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	Symmetric algorithm current source address DMA reads.

11.1.5.8 CE Symmetric Current Destination Address Register(Default Value: 0x0000_0000)

Offset: 0x0028			Register Name: CE_SCDA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	Symmetric algorithm current destination address DMA writes.

11.1.5.9 CE HASH Current Source Address Register(Default Value: 0x0000_0000)

Offset: 0x0034			Register Name: CE_HCSA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	HASH algorithm current source address DMA reads.

11.1.5.10 CE HASH Current Destination Address Register(Default Value: 0x0000_0000)

Offset: 0x0038			Register Name: CE_HCDA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	HASH algorithm current destination address DMA writes.

11.1.5.11 CE Asymmetric Current Source Address Register(Default Value: 0x0000_0000)

Offset: 0x0044			Register Name: CE_ACSA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	Asymmetric algorithm current source address DMA reads.

11.1.5.12 CE Asymmetric Current Destination Address Register(Default Value: 0x0000_0000)

Offset: 0x0048			Register Name: CE_ACDA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	Asymmetric algorithm current destination address DMA writes.

11.1.5.13 CE XTS Current Source Address Register(Default Value: 0x0000_0000)

Offset: 0x0054			Register Name: CE_XCSA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	XTS algorithm current source address DMA reads.

11.1.5.14 CE XTS Current Destination Address Register(Default Value: 0x0000_0000)

Offset: 0x0058			Register Name: CE_XCDA
Bit	**Read/Write**	**Default/Hex**	**Description**
31:0	R	0x0	XTS algorithm current destination address DMA writes.

11.2 Security ID

The Security ID(SID) is mainly used to program for SOC Efuse, which includes ChipID、HASH code an so on. The Efuse space of SOC is 0x0000 ~ 0x13F(Size is 2.5K bits)。 The SID Module has the following features：

- The module register is unsecure forever, Efuse has safety and non-safety.
- A Efuse can only be programed one time
- SID_SRAM is used to back up the information of Efuse.

Figures

Tables

12 Carrier, Storage and Baking Information

12.1 Carrier

12.1.1 Matrix Tray Information

Table 12-1 shows the A50 matrix tray carrier information.

Table12- 1. Matrix Tray Carrier Information

Item	Color	Size	Note
Tray	Black	315mm x 136mm x 7.62mm	133 Qty/Tray
Aluminum foil bags	Silvery white	540mm x 300mm x 0.14mm	Surface impedance:$10^9\Omega$ Vacuum packing Including HIC and desiccant Printing: RoHS symbol
Pearl cotton cushion(Vacuum bag)	White	12mm x 680mm x 185mm	
Pearl cotton cushion (The Gap between vacuum bag and inner box)	White	Left-Right:12mm x 180mm x 85mm Front-Back:12mm x 350mm x 70mm	
Inner Box	White	396mm x 196mm x 96mm	Printing: RoHS symbol 10 Tray/Inner box
Carton	White	420mm x 410mm x 320mm	6 Inner box/Carton

Table 12-2 shows the A50 packing quantity.

Table12- 2. Packing Quantity Information

Sample	Size(mm)	Qty/Tray	Tray/Inner Box	Full Inner Box Qty	Inner Box/Carton	Full Carton Qty
A50	12.8 x 12.3	133	10	1330	6	7980

Figure 12-1 shows tray dimension drawing of the A50.

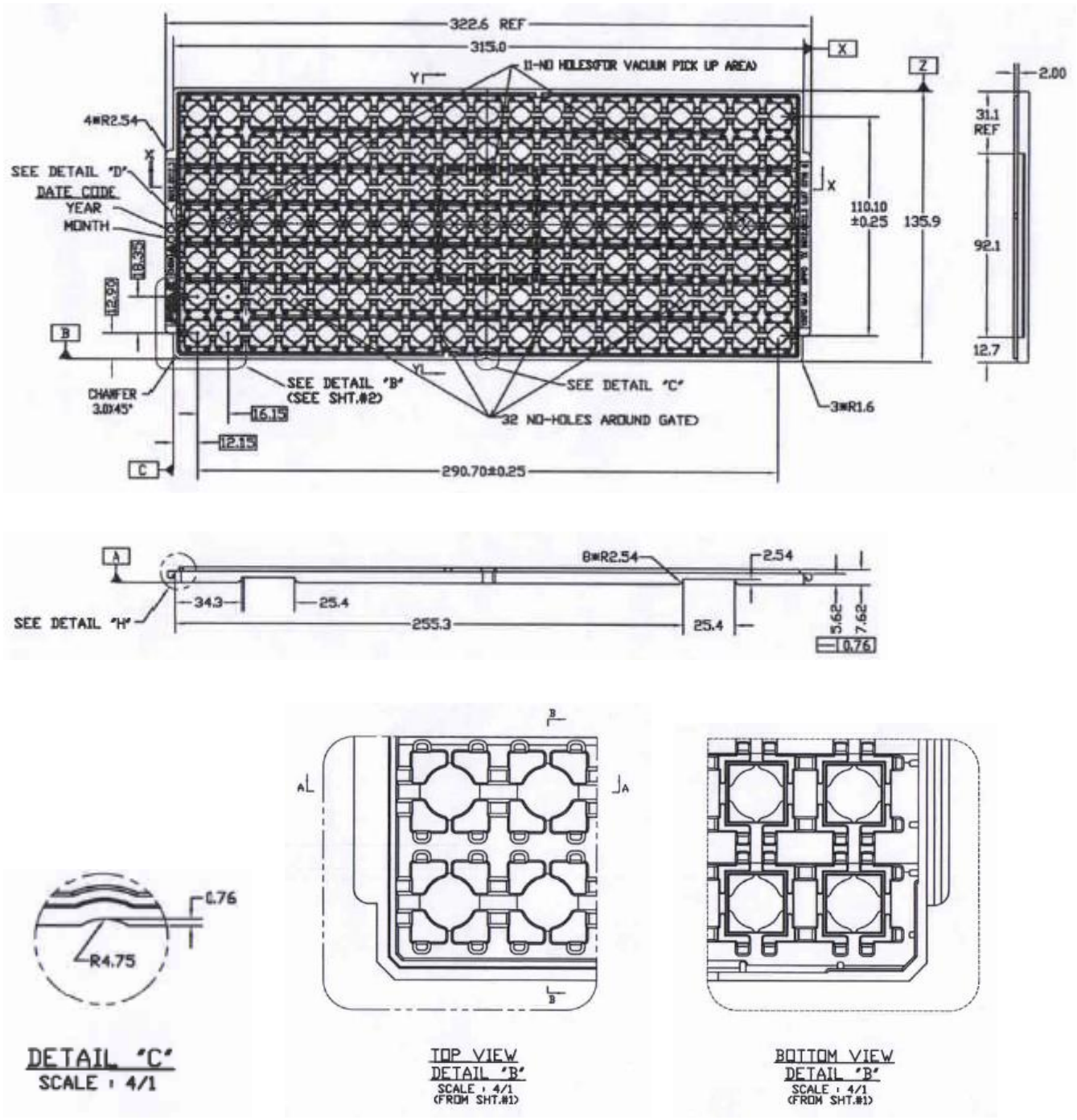


Figure12- 1. Tray Dimension Drawing

12.2 Storage

Reliability is affected if any condition specified in Section 12.2.2 and Section 12.2.3 has been exceeded.

12.2.1 Moisture Sensitivity Level(MSL)

A package's MSL indicates its ability to withstand exposure after it is removed from its shipment bag, a low MSL device

sample can be exposed on the factory floor longer than a high MSL device sample. All MSL are defined in Table 12-3.

Table12- 3. MSL Summary

MSL	Out-of-bag floor life	Comments
1	Unlimited	≤30°C / 85%RH
2	1 year	≤30°C / 60%RH
2a	4 weeks	≤30°C / 60%RH
3	168 hours	≤30°C / 60%RH
4	72 hours	≤30°C / 60%RH
5	48 hours	≤30°C / 60%RH
5a	24 hours	≤30°C / 60%RH
6	Time on Label(TOL)	≤30°C / 60%RH

NOTE

The A50 device samples are classified as MSL3.

12.2.2 Bagged Storage Conditions

The shelf life of the A50 device samples are defined in Table 12-4.

Table12- 4. Bagged Storage Conditions

Packing mode	Vacuum packing
Storage temperature	20°C ~26°C
Storage humidity	40%~60%RH
Shelf life	12 months

12.2.3 Out-of-bag Duration

It is defined by the device MSL rating, the out-of-bag duration of the A50 are as follows.

Table12- 5. Out-of-bag Duration

Storage temperature	20°C ~26°C
Storage humidity	40%~60%RH
Moisture sensitive level(MSL)	3
Floor life	168 hours

For no mention of storage rules in this document, please refer to the latest ***IPC/JEDEC J-STD-020C***.

12.3 Baking

It is not necessary to bake the A50 if the conditions specified in Section 12.2.2 and Section 12.2.3 have not been exceeded. It is necessary to bake the A50 if any condition specified in Section 12.2.2 and Section 12.2.3 has been exceeded.

It is necessary bake the A50 if the storage humidity condition has been exceeded, we recommend that the device sample removed from its shipment bag more than 2 days shall be baked to guarantee production.

Baking conditions: 125°C, 8 hours, nitrogen protection. Note that baking should not exceed 3 times.

Figures

Tables

13 Reflow Profile

All Allwinner chips provided for clients are lead-free RoHS-compliant products.

The reflow profile recommended in this document is a lead-free reflow profile that is suitable for pure lead-free technology of lead-free solder paste. If customers need to use lead solder paste, please contact with Allwinner FAE.

The lead-free reflow profile conditions are given in Table 13-1. The table is for reference only.

Table13- 1. Lead-free Reflow Profile Conditions

Profile Stage	Description	Symbol	High Temperature Condition Limits
Preheat	Initial ramp temperature range	A	25°C to 150°C
	Initial ramp rate	B	1.5~2.5°C/s
Soak	Soak temperature range	C	150°C to 190°C
	Soak time	D	80s to 110s
Reflow	Liquidus temperature	E	217°C
	Time above liquidus	F	60s to 90s
	Peak temperature	G	240°C to 250°C
Cool down	Cool down temperature rate	H	≤ 4°C/s

Figure 13-1 shows the typical lead-free reflow profile.

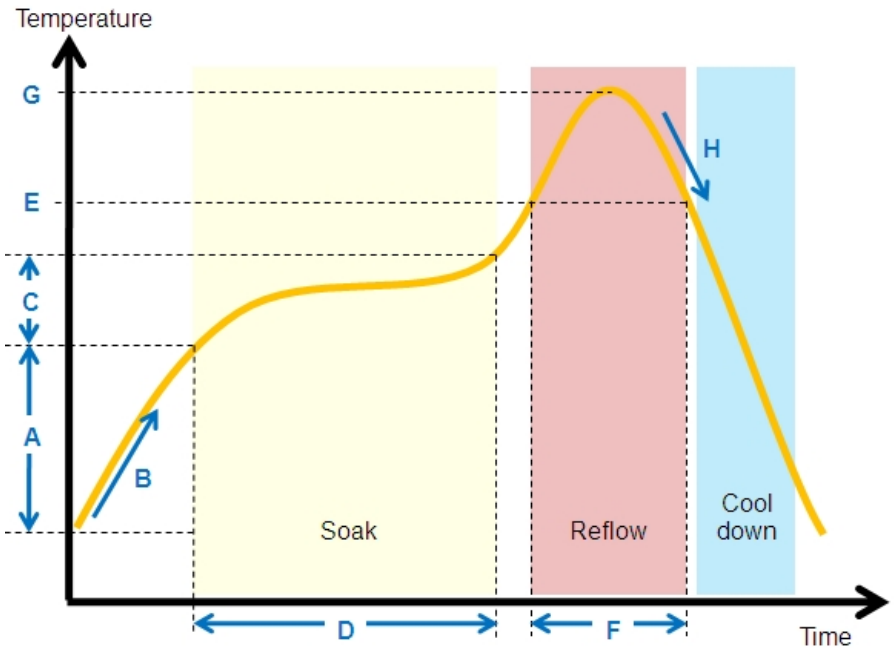


Figure13- 1. Typical Lead-free Reflow Profile

NOTE

The above reflow profile is solder joint testing result, it is for reference only, please adjust depending on actual production conditions.

The method of measuring the reflow soldering process is as follows.

Fix the thermocouple probe of the temperature measuring line at the connection point between the pin (solderable end) of the packaged device and the pad by using high-temperature solder wire or high-temperature tape, fix the packaged device at the pad by using high-temperature tape or other methods, and cover over the thermocouple probe. See Figure 13-2.

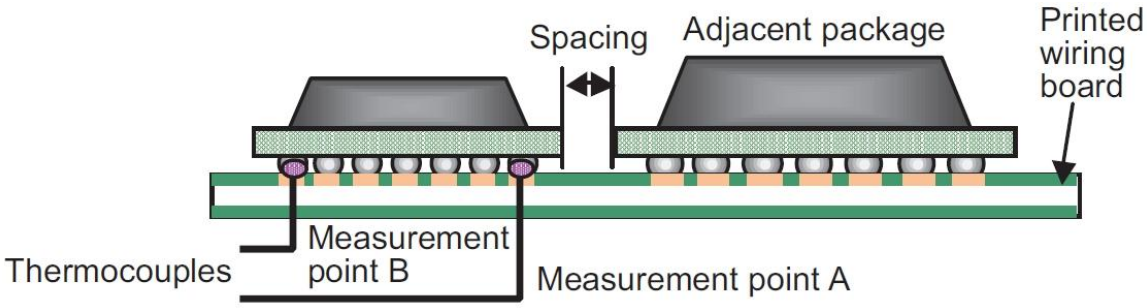


Figure13- 2. Measuring the Reflow Soldering Process

NOTE

To measure the temperature of QFP-packaged chip, place the temperature probe directly at the pin.

If possible, the more accurate measuring way is to drill the packaged device, or drill the PCB, and fix the thermocouple probe through the drilled hole at the pad.

Figures

14 Marking Information

Figure 14-1 shows the A50 marking.

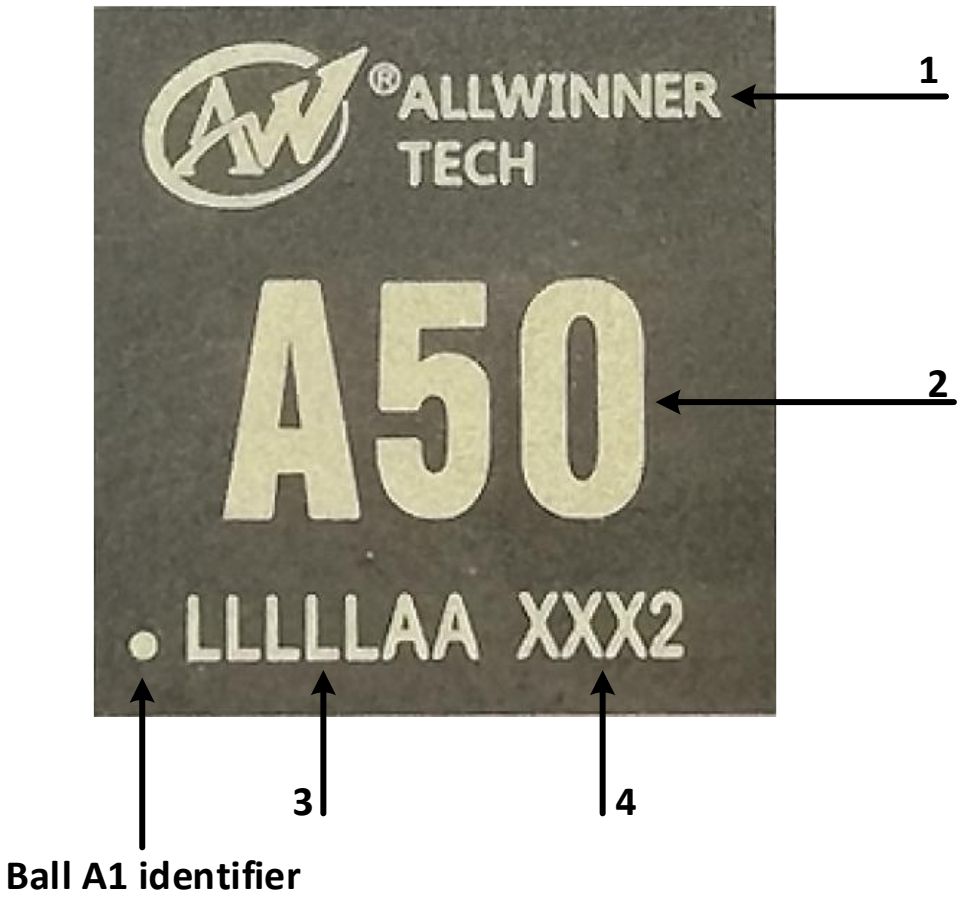


Figure14- 1. A50 Marking Information

Table 14-1 describes the A50 marking definitions.

Table14- 1. A50 Marking Definitions

No.	Marking	Description	Fixed/Dynamic
1	ALLWINNER TECH	Allwinner logo or name	Fixed
2	A50	Product name	Fixed
3	LLLLLAA	Lot number	Dynamic
4	XXX2	Data code	Dynamic

