

Samsung Exynos4210

RISC Microprocessor

**Public Version
August 2011**

User's Manual

SAMSUNG ELECTRONICS RESERVES THE RIGHT TO CHANGE PRODUCTS, INFORMATION AND SPECIFICATIONS WITHOUT NOTICE.

Products and specifications discussed herein are for reference purposes only. All information discussed herein is provided on an "AS IS" basis, without warranties of any kind.

This document and all information discussed herein remain the sole and exclusive property of Samsung Electronics. No license of any patent, copyright, mask work, trademark or any other intellectual property right is granted by one party to the other party under this document, by implication, estoppel or otherwise.

Samsung products are not intended for use in life support, critical care, medical, safety equipment, or similar applications where product failure could result in loss of life or personal or physical harm, or any military or defense application, or any governmental procurement to which special terms or provisions may apply.

For updates or additional information about Samsung products, contact your nearest Samsung office.

All brand names, trademarks and registered trademarks belong to their respective owners.

© 2011 Samsung Electronics Co., Ltd. All rights reserved.

Important Notice

While the information in this publication has been carefully checked and is believed to be entirely accurate at the time of publication, it is intended for reference purposes only. Samsung assumes no responsibility for possible errors or omissions, or for any consequences resulting from the use of the information contained herein.

Samsung reserves the right to make changes in its products or product specifications with the intent to improve function or design at any time and without notice and is not required to update this documentation to reflect such changes.

This publication does not convey to a purchaser of semiconductor devices described herein any license under the patent rights of Samsung or others.

Purchasers are responsible for their products and applications using Samsung components. Samsung makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Samsung assume any liability arising out of the application or use of any product or circuit and specifically disclaims any and all liability, including without limitation any consequential

or incidental damages.

"Typical" parameters can and do vary in different applications. All operating parameters, including "Typicals" must be validated for each customer application by the customer's technical experts.

Samsung products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, for other applications intended to support or sustain life, or for any other application in which the failure of the Samsung product could create a situation where personal injury or death may occur.

Should the Buyer purchase or use a Samsung product for any such unintended or unauthorized application, the Buyer shall indemnify and hold Samsung and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, expenses, and reasonable attorney fees arising out of, either directly or indirectly, any claim of personal injury or death that may be associated with such unintended or unauthorized use, even if such claim alleges that Samsung was negligent regarding the design or manufacture of said product.

Copyright © 2011 Samsung Electronics Co., Ltd.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electric or mechanical, by photocopying, recording, or otherwise, without the prior written consent of Samsung Electronics.

Samsung Electronics Co., Ltd.
San #24 Nongseo-Dong, Giheung-Gu
Yongin-City, Gyeonggi-Do, Korea 446-711

Contact Us: mobilesol.cs@samsung.com

Home Page: <http://www.samsungsemi.com>

Device Handling Guide

Precaution against Electrostatic Discharge

When handling semiconductor devices, be sure that the environment is protected against static electricity.

1. Operators should wear anti-static clothing and use earth band.
2. All objects that come in direct contact with devices should be made of materials that do not produce static electricity that would cause damage.
3. Equipment and work table must be earthed.
4. Ionizer is recommended to remove electron charge.

Contamination

Be sure to use semiconductor products in the environment that may not be exposed to dust or dirt adhesion.

Temperature/Humidity

Semiconductor devices are sensitive to environment temperature and humidity. High temperature or humidity may deteriorate semiconductor device's characteristics. Therefore avoid storage or use in such conditions.

Mechanical Shock

Care should be exercised not to apply excessive mechanical shock or force on semiconductor device.

Chemical

Do not expose semiconductor device to chemical because reaction to chemical may cause deterioration of device characteristics.

Light Protection

In case of non-EMC (Epoxy Molding Compound) package, do not expose semiconductor IC to strong light. It may cause device's malfunction. (But, some special products which utilize the light or have security function are excepted from this guide)

Radioactive, Cosmic and X-ray

Semiconductor devices can be influenced by radioactive, cosmic ray or X-ray. Radioactive, cosmic and X-ray may cause soft error during device operation. Therefore semiconductor devices must be shielded under environment that may be exposed to radioactive, cosmic ray or X-ray.

EMS (Electromagnetic Susceptibility)

Note that semiconductor device's characteristics may be affected by strong electromagnetic wave or magnetic field during operation under insufficient PCB circuit design for EMS.

Trademarks

Exynos, Exynos4210, FlexOneNAND, and OneNAND are trademarks of Samsung Electronics.

ARM, Jazelle, TrustZone, and Thumb are registered trademarks of ARM Limited. Cortex, ETM, ETB, Coresight, ISA, and Neon are trademarks of ARM Limited.

Java is a trademark of Sun Microsystems, Inc.

SD is a registered trademark of Toshiba Corporation.

MMC and eMMC are trademarks of MultiMediaCard Association.

JTAG is a registered trademark of JTAG Technologies, Inc.

Synopsys is a registered trademark of Synopsys, Inc.

I2S is a trademark of Phillips Electronics.

I²C is a trademark of Phillips Semiconductor Corp.

MIPI and Slimbus are registered trademarks of the Mobile Industry Processor Interface (MIPI) Alliance.

All other trademarks are the property of their respective owners.

Revision History

Revision No.	Date	Description	Author(s)
0.00	July 25, 2011	• Initial draft	IW Kong

Table of Contents

1 PRODUCT OVERVIEW	1-1
1.1 Introduction	1-1
1.2 Features	1-2
1.2.1 Core Subsystem	1-4
1.2.2 Memory Subsystem	1-5
1.2.3 Multimedia	1-6
1.2.4 Audio Subsystem.....	1-9
1.2.5 Connectivity	1-9
1.2.6 System Peripheral	1-12
1.3 Conventions	1-14
1.3.1 Register RW Conventions	1-14
1.3.2 Register Value Conventions	1-14
2 MEMORY MAP	2-1
2.1 Overview	2-1
2.2 SFR Base Address	2-2
3 CHIP ID.....	3-1
3.1 Overview	3-1
3.2 Register Description.....	3-2
3.2.1 Register Map Summary	3-2
4 GPIO CONTROL	4-1
4.1 Overview	4-1
4.2 Features	4-2
4.3 Functional Description	4-3
4.3.1 General Purpose Input/Output Block Diagram	4-3
4.4 Register Description.....	4-4
4.4.1 Register Map Summary	4-4
4.4.2 Part 1	4-19
4.4.3 Part 2	4-121
4.4.4 Part 3	4-224
5 CLOCK MANAGEMENT UNITS.....	5-1
5.1 Overview	5-1
5.2 Functional Description	5-2
5.2.1 Clock Declaration.....	5-2
5.2.2 Clock Values	5-4
5.2.3 Clock Generation	5-6
5.2.4 Clock Configuration Procedure.....	5-11
5.2.5 Special Clock Description	5-13
5.2.6 CLKOUT	5-15
5.2.7 I/O Description	5-17
5.3 Register Description.....	5-18
5.3.1 Register Map Summary	5-19

6 INTERRUPT CONTROLLER	6-1
6.1 Overview	6-1
6.2 Features	6-1
6.3 Functional Description	6-2
6.3.1 Security Extensions Support	6-2
6.3.2 Implementation-Specific Configurable Features	6-2
6.3.3 Terminology	6-3
6.3.4 Interrupt Source	6-3
6.3.5 Functional Overview of Generic Interrupt Controller	6-11
6.3.6 Interrupt Handling and Prioritization	6-11
6.4 Register Description	6-12
6.4.1 Register Map Summary	6-12
7 INTERRUPT COMBINER	7-1
7.1 Overview	7-1
7.2 Features	7-1
7.3 Functional Description	7-2
7.3.1 Block Diagram	7-2
7.3.2 Interrupt Sources	7-3
7.3.3 Interrupt Combiner Operation	7-7
7.4 Register Description	7-8
7.4.1 Register Map Summary	7-8
8 DMA CONTROLLER	8-1
8.1 Overview	8-1
8.2 Features	8-2
8.3 Functional Description	8-5
8.3.1 Instructions	8-5
8.4 Register Description	8-6
8.4.1 Register Map Summary for MDMA	8-6
8.4.2 Register Map Summary for PDMA0/PDMA1	8-10
9 SRAM CONTROLLER	9-1
9.1 Overview	9-1
9.2 Features	9-1
9.3 Functional Description	9-2
9.3.1 Block Diagram	9-2
9.3.2 Interface Port Description	9-2
9.3.3 nWAIT Pin Operation	9-3
9.3.4 Programmable Access Cycle	9-4
9.4 Register Description	9-5
9.4.1 Register Map Summary	9-5
10 NAND FLASH CONTROLLER	10-1
10.1 Overview	10-1
10.2 Features	10-1
10.3 Functional Description	10-2
10.3.1 Block Diagram	10-2
10.3.2 Interface Port Description	10-2
10.3.3 NAND Flash Memory Timing	10-3

10.3.4 Software Mode.....	10-4
10.3.5 Programming Constraints	10-14
10.4 Register Description	10-15
10.4.1 Register Map Summary	10-15
10.4.2 Nand Flash Interface and 1/4-bit ECC Registers	10-17
10.4.3 ECC Registers for 8, 12 and 16-bit ECC.....	10-28
11 PWM TIMER	11-1
11.1 Overview	11-1
11.2 Features	11-4
11.3 Functional Description	11-5
11.3.1 Interface Port Description	11-5
11.3.2 Prescaler and Divider	11-5
11.3.3 Basic Timer Operation.....	11-6
11.3.4 Auto-reload and Double Buffering	11-7
11.3.5 Timer Operation Example.....	11-8
11.3.6 Initialize Timer (Setting Manual-up Data and Inverter).....	11-9
11.3.7 PWM (Pulse Width Modulation).....	11-9
11.3.8 Output Level Control.....	11-10
11.3.9 Dead Zone Generator.....	11-11
11.4 Register Description.....	11-12
11.4.1 Register Map Summary	11-12
12 WATCHDOG TIMER.....	12-1
12.1 Overview	12-1
12.2 Features	12-1
12.3 Functional Description	12-2
12.3.1 Watchdog Timer Operation	12-2
12.3.2 WTDAT and WTCNT	12-2
12.3.3 WDT Start.....	12-2
12.3.4 Consideration of Debugging Environment.....	12-2
12.4 Register Description.....	12-3
12.4.1 Register Map Summary	12-3
13 UNIVERSAL ASYNCHRONOUS RECEIVER AND TRANSMITTER (UART)	
13-1	
13.1 Overview	13-1
13.2 Features	13-1
13.3 Functional Description	13-2
13.3.1 Block Diagram	13-2
13.3.2 Interface Port Description	13-3
13.3.3 Data Transmission.....	13-4
13.3.4 Data Reception	13-4
13.3.5 Auto Flow Control (AFC)	13-5
13.3.6 Example of Non Auto-Flow Control (Controlling nRTS and nCTS by Software).....	13-6
13.3.7 Tx/Rx FIFO Trigger Level and DMA Burst Size in DMA Mode	13-6
13.3.8 RS-232C Interface	13-6
13.3.9 Interrupt/DMA Request Generation	13-7
13.3.10 UART Error Status FIFO	13-8
13.3.11 UART Input Clock Description	13-12
13.4 Register Description.....	13-13

13.4.1 Register Map Summary	13-13
-----------------------------------	-------

14 INTER-INTEGRATED CIRCUIT (IIC) BUS INTERFACE14-1

14.1 Overview	14-1
14.2 Features	14-2
14.3 Functional Description	14-3
14.3.1 Block Diagram	14-3
14.3.2 Interface Port Description	14-3
14.3.3 IIC-Bus Interface Operation	14-4
14.3.4 Start and Stop Conditions	14-4
14.3.5 Data Transfer Format	14-5
14.3.6 ACK Signal Transmission	14-6
14.3.7 Read-Write Operation	14-7
14.3.8 Bus Arbitration Procedures	14-7
14.3.9 Abort Conditions	14-7
14.3.10 Configuring I'C-Bus	14-7
14.3.11 Flowcharts of Operations in Each Mode	14-8
14.4 Register Description.....	14-12
14.4.1 Register Map Summary	14-12

15 SERIAL PERIPHERAL INTERFACE (SPI).....15-1

15.1 Overview	15-1
15.2 Features	15-1
15.3 Functional Description	15-2
15.3.1 Interface Port Description	15-2
15.3.2 Operation of Serial Peripheral Interface	15-3
15.3.3 Setting Sequence of Special Function Register	15-6
15.3.4 SPI Input Clock Description	15-6
15.4 Register Description	15-7
15.4.1 Register Map Summary	15-7
15.4.2 Special Function Register.....	15-8

16 DISPLAY CONTROLLER16-1

16.1 Overview	16-1
16.2 Features	16-2
16.3 Functional Description	16-4
16.3.1 Brief Description of the Sub-Block	16-4
16.3.2 Data Flow	16-4
16.3.3 Overview of the Color Data	16-7
16.3.4 Palette Usage	16-22
16.3.5 Window Blending	16-24
16.3.6 Image Enhancement	16-33
16.3.7 VTIME Controller Operation	16-34
16.3.8 Setting of Commands	16-37
16.3.9 Virtual Display	16-40
16.3.10 RGB Interface Spec	16-41
16.3.11 LCD Indirect i80 System Interface	16-46
16.4 I/O Description	16-49
16.5 Register Description	16-50
16.5.1 Overview	16-50
16.5.2 Register Map Summary	16-51

16.5.3 Palette Memory.....	16-58
16.5.4 Control Registers	16-59
16.5.5 Palette Ram	16-129
16.5.6 Gamma Lookup Table	16-131
16.5.7 Shadow Windows Control	16-132
17 ADC & TOUCH SCREEN INTERFACE	17-1
17.1 Overview	17-1
17.2 Features	17-1
17.3 Functional Description	17-2
17.3.1 Block Diagram	17-2
17.3.2 Interface Port Description	17-3
17.3.3 A/D Conversion Time.....	17-3
17.3.4 Touch Screen Interface Mode	17-4
17.3.5 Standby Mode.....	17-6
17.3.6 Two Touch Screen Interfaces.....	17-6
17.3.7 ADC & Touch Screen Interface Input Clock Diagram	17-8
17.4 Register Description.....	17-9
17.4.1 Register Map Summary	17-9
18 KEYPAD INTERFACE.....	18-1
18.1 Overview	18-1
18.2 Features	18-1
18.3 Functional Description	18-2
18.3.1 Interface Port Description	18-2
18.3.2 Key Matrix Interface External Connection Guide	18-4
18.3.3 Debouncing Filter.....	18-5
18.3.4 Filter Clock.....	18-5
18.3.5 Wakeup Source	18-5
18.3.6 Keypad Scanning Procedure for software scan	18-6
18.3.7 Keypad Scanning Procedure for Hardware scan	18-10
18.4 Register Description.....	18-11
18.4.1 Register Map Summary	18-11

List of Figures

Figure Number	Title	Page Number
Figure 4-1	GPIO Block Diagram	4-3
Figure 5-1	Exynos4210 Clock Generation Circuit (CPU, BUS, DRAM Clocks).....	5-8
Figure 5-2	Exynos4210 Clock Generation Circuit (Special Clocks).....	5-10
Figure 5-3	Exynos4210 CLKOUT Control Logic	5-15
Figure 5-4	Exynos4210 Clock Controller Address Map	5-18
Figure 6-1	Interrupt Sources Connection	6-3
Figure 7-1	Block Diagram of External Interrupt Combiner	7-2
Figure 8-1	Two DMA Tops	8-1
Figure 9-1	Block Diagram of SROM Controller Introduction	9-2
Figure 9-2	SROM Controller nWAIT Timing Diagram.....	9-3
Figure 9-3	SROM Controller Read Timing Diagram	9-4
Figure 9-4	SROM Controller Write Timing Diagram	9-4
Figure 10-1	NAND Flash Controller Block Diagram.....	10-2
Figure 10-2	CLE and ALE Timing (TACLS = 1, TWRPH0 = 0, TWRPH1 = 0)	10-3
Figure 10-3	nWE and nRE Timing (TWRPH0 = 0, TWRPH1 = 0)	10-3
Figure 11-1	Simple Example of a PWM Cycle	11-2
Figure 11-2	PWM TIMER Clock Tree Diagram.....	11-3
Figure 11-3	Timer Operations	11-6
Figure 11-4	Example of Double Buffering Feature	11-7
Figure 11-5	Example of a Timer Operation.....	11-8
Figure 11-6	Example of PWM	11-9
Figure 11-7	Inverter On/Off	11-10
Figure 11-8	Waveform when a Deadzone Feature is Enabled	11-11
Figure 12-1	Watchdog Timer Block Diagram	12-2
Figure 13-1	Block Diagram of UART.....	13-2
Figure 13-2	UART AFC Interface.....	13-5
Figure 13-3	UART Receives the Five Characters Including Two Errors	13-9
Figure 13-4	IrDA Function Block Diagram	13-10
Figure 13-5	Serial I/O Frame Timing Diagram (Normal UART).....	13-10
Figure 13-6	Infra-Red Transmit Mode Frame Timing Diagram	13-11
Figure 13-7	Infra-Red Receive Mode Frame Timing Diagram.....	13-11
Figure 13-8	Input Clock Diagram for UART	13-12
Figure 13-9	nCTS and Delta CTS Timing Diagram	13-24
Figure 13-10	Block diagram of UNTSP, UINTP and UNTM	13-28
Figure 14-1	I ² C-Bus Block Diagram	14-3
Figure 14-2	Start and Stop Condition.....	14-4

Figure 14-3	I ² C-Bus Interface Data Format	14-5
Figure 14-4	Data Transfer on the I ² C-Bus	14-5
Figure 14-5	Acknowledge on the I ² C-Bus	14-6
Figure 14-6	Operations for Master/Transmitter Mode.....	14-8
Figure 14-7	Operations for Master/ Receiver Mode.....	14-9
Figure 14-8	Operations for Slave/ Transmitter Mode.....	14-10
Figure 14-9	Operations for Slave/Receiver Mode.....	14-11
 Figure 15-1	SPI Transfer Format	15-5
Figure 15-2	Input Clock Diagram for SPI	15-6
Figure 15-3	Auto Chip Select Mode Waveform (CPOL = 0, CPHA = 0, CH_WIDTH = Byte)	15-10
 Figure 16-1	Block Diagram of Display Controller	16-1
Figure 16-2	Block Diagram of the Data Flow	16-5
Figure 16-3	Block Diagram of the Interface	16-6
Figure 16-4	16BPP (5:6:5) Display Types.....	16-21
Figure 16-5	Blending Equation.....	16-26
Figure 16-6	Blending Diagram	16-28
Figure 16-7	Blending Factor Decision	16-29
Figure 16-8	Color-Key Function Configurations.....	16-30
Figure 16-9	Blending and Color-Key Function	16-31
Figure 16-10	Blending Decision Diagram	16-32
Figure 16-11	Image Enhancement Flow	16-33
Figure 16-12	Sending Command	16-38
Figure 16-13	Example of Scrolling in Virtual Display	16-40
Figure 16-14	LCD RGB Interface Timing	16-42
Figure 16-15	LCD RGB Interface Timing (RGB Parallel).....	16-43
Figure 16-16	LCD RGB Interface Timing (RGB Serial, Dummy Disable).....	16-44
Figure 16-17	LCD RGB Output Order.....	16-45
Figure 16-18	Indirect i80 System Interface WRITE Cycle Timing.....	16-47
 Figure 17-1	ADC and Touch Screen Interface Functional Block Diagram	17-2
Figure 17-2	ADC and Touch Screen Operation Signal.....	17-7
Figure 17-3	Input Clock Diagram for ADC & Touch Screen Interface	17-8
 Figure 18-1	Key Matrix Interface External Connection Guide.....	18-4
Figure 18-2	Internal Debouncing Filter Operation.....	18-5
Figure 18-3	Keypad Scanning Procedure	18-6
Figure 18-4	Keypad Scanning Procedure II	18-7
Figure 18-5	Keypad Scanning Procedure III	18-8
Figure 18-6	Keypad Scanning Procedure when the Two-key Pressed with Different Row	18-9

List of Tables

Table Number	Title	Page Number
Table 5-1	Operating Frequencies in Exynos4210.....	5-1
Table 5-2	APLL & MPLL PMS Value	5-5
Table 5-3	EPLL PMS Value	5-5
Table 5-4	VPLL PMS Value	5-5
Table 5-5	Special Clocks in Exynos4210.....	5-13
Table 5-6	I/O Clocks in Exynos4210.....	5-14
Table 5-7	CLKOUT Input Clock Selection Information	5-16
Table 6-1	GIC Configuration Values	6-2
Table 6-2	External GIC Interrupt Table (SPI[127:32]: Non-Combined Interrupt).....	6-4
Table 6-3	External GIC Interrupt Table (SPI[31:0]: Combined Interrupt).....	6-7
Table 6-4	External GIC Interrupt Table (PPI[15:0]).....	6-10
Table 7-1	Interrupt Groups for E-GIC.....	7-3
Table 8-1	DMA Request Mapping Table.....	8-2
Table 11-1	Minimum and Maximum Resolution Based on Prescaler and Clock Divider Values.....	11-5
Table 13-1	Interrupts in Connection with FIFO	13-7
Table 16-1	32BPP (8:8:8:8) Palette Data Format	16-22
Table 16-2	25BPP (A: 8:8:8) Palette Data Format.....	16-22
Table 16-3	19BPP (A: 6:6:6) Palette Data Format.....	16-23
Table 16-4	16BPP (A: 5:5:5) Palette Data Format.....	16-23
Table 16-5	Relation 16 bpp Between VCLK and CLKVAL (TFT, Frequency of Video Clock Source = 60 MHz)....	16-34
Table 16-6	i80 Output Mode	16-39
Table 16-7	Timing Reference Code (XY Definition).....	16-48
Table 18-1	Keypad Interface I/O Description	18-2

1 Product Overview

1.1 Introduction

Exynos4210 is a 32-bit RSIC cost-effective, low power, performance optimized and Cortex-A9 Dual Core based micro-processor solution for mobile phones and netbook applications.

Exynos4210 has an optimized interface to external memory capable of sustaining the demanding memory bandwidths required in communication services and memory interleaver and system MMU to support virtual addressing for bus masters. The memory system has dedicated DRAM ports and Static Memory port. The dedicated DRAM ports support DDR2, LPDDR2 and DDR3 interface for high bandwidth. Static Memory Port supports FlexOneNAND, NOR Flash, and ROM type external memory and components.

To reduce the total system cost and enhance the overall functionality, Exynos4210 includes many hardware peripherals such as TFT 24-bit true color LCD controller, Camera Interface, MIPI DSI, CSI, System Manager for power management, SATA interface, PCI Express interface (not available with POP option), embedded GPS, MIPI slimbus interface, five UARTs, 24-channel DMA, Timers, General I/O Ports, three I2S, S/PDIF, eight IIC-BUS interface, three HS-SPI, USB Host 2.0, USB 2.0 Device operating at high speed (480 Mbps), four SD Host and high-speed Multimedia Card Interface, and four PLLs for clock generation.

1.2 Features

The key features of Exynos4210 include:

- ARM Cortex-A9 based Dual CPU Subsystem with NEON
 - 32/32 KB I/D Cache, 1 MB L2 Cache
 - Operating frequency up to 1 GHz
- 64-bit Multi-layer bus architecture
 - Core-D domain for ARM Cortex-A9 Dual, CoreSight and external memory interface.
- Operating frequency up to 200 MHz
 - Global D- domain mainly for multimedia components and external storage interfaces.
- Operating frequency up to 100 MHz
 - Core-P, Global-P domain mainly for other system component such as system peripherals, peri DMAs, connectivity IPs, and Audio interfaces.
- Operating frequency up to 100 MHz
 - Audio domain for low power audio play
- Advanced power management for mobile applications
- 64 KB ROM for secure booting and 128 KB RAM for security function
- 8-bit ITU 601/ 656 Camera Interface supports horizontal size up to 4224 pixels for scaled and 8192 pixels for un-scaled resolution
- 2D Graphics Acceleration up to 200 Mpixels/s
- 1/2/4/ 8bpp Palletized or 8/16/ 24bpp Non-Palletized Color TFT recommend up to WXGA resolution
- TV-out and HDMI interface support for NTSC and PAL mode with image enhancer
- MIPI-DSI and MIPI-CSI interface support
- One AC-97 audio codec interface and 3-channel PCM serial audio interface
- Three 24-bit I2S interface support
- One Tx only S/PDIF interface support for digital audio
- Eight I²C interface support
- Three SPI support
- Five UART supports three Mbps ports for Bluetooth 2.0
- On-chip USB 2.0 Device supports high-speed (480 Mbps, on-chip transceiver)
- On-chip USB 2.0 Host support
- Two on-chip USB HSIC
- Asynchronous Modem Interface support
- Four SD/ SDIO/ HS-MMC interface support
- SATA 1.5 Gbps and 3.0 Gbps interface support

- 24-channel DMA controller (8 channels for Memory-to-memory DMA, 16 channels for Peripheral DMA)
- Supports 14 × 8 key matrix
- 10-channel 12-bit multiplexed ADC
- Configurable GPIOs
- Real time clock, PLL, timer with PWM and watch dog timer
- System timer support for accurate tick time in power down mode (Except sleep mode)
- Memory Subsystem
 - Asynchronous SRAM/ROM/NOR Interface with x8 or x16 data bus
 - NAND Interface with x8 data bus
 - Muxed OneNAND Interface with x16 data bus
 - DDR2 interface with x16 or x32 data bus (800 Mbps/pin DDR)
 - LPDDR2 interface (800 Mbps/ pin DDR)
 - DDR3 interface (800 Mbps/ pin DDR)
- Embedded GPS/AGPS.

1.2.1 Core Subsystem

The key features of main microprocessors include:

- The ARM Cortex-A9 MPCore (dual core) processor integrates the proven and highly successful ARM MPCore technology along with further enhancements to simplify and broaden the adoption of multi-core solutions.
- With the ability to scale in speed from 600 MHz to 1 GHz (or more), the ARM Cortex-A9 MPCore dual processor meets the requirements of power-optimized mobile devices, which require operation in low power and performance-optimized consumer applications require over 2000 Dhrystone MIPS.
- Other features of ARM Cortex-A9 MPCore dual core processor include:
 - Thumb-2 technology for greater performance, energy efficiency, and code density
 - NEON™ signal processing extensions
 - Jazelle RCT Java-acceleration technology
 - TrustZone technology for secure transactions and DRM
 - Floating-Point unit for significant acceleration for both single and double precision scalar Floating-Point operations
 - Optimized L1 caches for performance and power
 - Integrated 1 MB L2 Cache using standard compiled RAMs
 - Program Trace Macrocell and CoreSight
- Generic Interrupt Controller
 - Support for three interrupt types
 - Software Generated Interrupt (SGI)
 - Private Peripheral Interrupt (PPI)
 - Shared Peripheral Interrupt (SPI).
 - Programmable interrupts that enable to set the
 - Security state for an interrupt
 - Priority level of an interrupt
 - enabling or disabling of an interrupt
 - Processors that receive an interrupt.
 - Enhanced security features

1.2.2 Memory Subsystem

The key features of memory subsystem include:

- High bandwidth Memory Matrix subsystem
- Two independent external memory ports (1×16 Static Hybrid Memory port and 2×32 DRAM port)
- Matrix architecture increases the overall bandwidth with simultaneous access capability
 - SRAM/ ROM/ NOR Interface
 - x8 or x16 data bus
 - Address range support: 16-bit
 - Supports asynchronous interface
 - Supports byte and half-word access
 - OneNAND Interface
 - x16 data bus
 - Supports synchronous and asynchronous interface
 - Supports 2 chip selections
 - Supports 2 KB page mode for OneNAND and 4 KB page mode for Flex OneNAND
 - Supports dedicated DMA
 - NAND Interface
 - Supports industry standard NAND interface
 - x8 data bus
 - DDR2 Interface
 - x32 data bus with 800 Mbps/pin double data rate (DDR)
 - 1.8 V interface voltage
 - Density support up to 1-Gb per port (2CS, when 4 bank DDR2)
 - Density support up to 4-Gb per port (1CS, when 8 bank DDR2)
 - LPDDR2 interface
 - x32 data bus with up to 800 Mbps/pin
 - 1.2 V interface voltage
 - Density support up to 4-Gb per port (2CS)
 - DDR3 interface
 - x32 data bus with up to 800 Mbps/pin
 - 1.8/1.5 V interface voltage
 - Density support up to 4-Gb per port (2CS)

1.2.3 Multimedia

The key features of multimedia include:

- Camera Interface
 - Multiple input support
 - ITU-R BT 601/656 mode
 - DMA (AXI 64-bit interface) mode
 - MIPI (CSI) mode
 - Direct FIFO mode (from LCDC)
 - Multiple output support
 - DMA (AXI 64-bit interface) mode
 - Direct FIFO mode (to LCDC)
 - Digital Zoom In (DZI) capability
 - Multiple camera input support
 - Programmable polarity of video sync signals
 - Input horizontal size support up to 4224 pixels for scaled and 8192 pixels for un-scaled resolution
 - Image mirror and rotation (X-axis mirror, Y-axis mirror, 90°, 180°, and 270° rotation)
 - Various image formats generation
 - Capture frame control support
 - Image effect support
- JPEG Codec
 - Supports Compression/Decompression up to 65536 × 65536
 - Supported format of compression
 - Input raw image: YCbCr4:2:2 or RGB565
 - Output JPEG file: Baseline JPEG of YCbCr4:2:2 or YCbCr4:2:0
 - Supported format of decompression
 - Input JPEG file: Baseline JPEG of YCbCr4:4:4, YCbCr4:2:2, YCbCr4:2:0, or gray
 - Output raw image: YCbCr4:2:2 or YCbCr4:2:0
 - Supports general-purpose color-space converter
- 2D Graphic Engine
 - BitBLT
 - Supports maximum 8000x8000 image size
 - Window clipping, 90°/180°/270°/Rotation, X Flip/Y Flip
 - Reverse Addressing (X positive/negative, Y positive/negative)
 - Totally 4-operand raster operation (ROP4)
 - Alpha blending (fixed alpha value/per-pixel alpha value)
 - Arbitrary size pixel pattern drawing, Pattern cache
 - 16/24/32-bpp. Packed 24-bpp color format

- Analog TV interface
 - Out video format: NTSC-M/NTSC-J/NTSC4.43/PAL-B, D, G, H, I/PAL-M/PAL-N/PAL-Nc/PAL-60 compliant
 - Supported input format: ITU-R BT.601 (YCbCr 4: 4: 4)
 - Supports 480i and 576i resolutions
 - Supports Composite interface
- Digital TV Interface
 - High-definition Multimedia Interface (HDMI) 1.3
 - Supports up to 1080p 60Hz and 8-channel/112 kHz/24-bit audio
 - Supports 480p, 576p, 720p, 1080i (cannot support 480i)
 - Supports HDCP v1.1
- Rotator
 - Supported image format: YCbCr422 (Interleave), YCbCr420 (Non-interleave), RGB565 and RGB888 (Unpacked)
 - Supported rotate degree: 90, 180, 270, flip vertical, and flip horizontal
- Video processor: The video processor supports:
 - BOB/ 2D-IPC mode
 - Produces YCbCr 4: 4: 4 output to help the mixer blend video and graphics
 - 1/4X to 16X vertical scaling with 4-tap/16-phase polyphase filter
 - 1/4X to 16X horizontal scaling with 8-tap/16-phase polyphase filter
 - Pan and scan, Letterbox, and NTSC/PAL conversion using scaling
 - Flexible scaled video positioning within display area
 - 1/16 pixel resolution Pan and Scan modes
 - Flexible post video processing
 - Color saturation, Brightness/Contrast enhancement, Edge enhancement
 - Color space conversion between BT.601 and BT.709
 - Video input source size up to 1920 × 1080
- Video Mixer
 - The Video Mixer supports:
 - Overlapping and blending input video and graphic layers
 - 480i/p, 576i/p, 720p, and 1080i/p display size
 - Four layers (1 video layer, 2 graphic layer, and 1 background layer)

- TFT-LCD Interface

The TFT-LCD Interface supports:

- 24/18/16-bpp parallel RGB Interface LCD
- 8/6-bpp serial RGB Interface
- Dual i80 Interface LCD
- 1/ 2/4/8-bpp Palletized or 8/16/24-bpp Non-Palletized Color TFT
- Typical actual screen size: $1366 \times 768, 1024 \times 768, 800 \times 480, 640 \times 480, 320 \times 240, 160 \times 160$, and so on
- Virtual image up to 16M pixel (4K pixel x4K pixel)
- Five Window Layers for PIP or OSD
- Real-time overlay plane multiplexing
- Programmable OSD window positioning
- 16-level alpha blending

1.2.4 Audio Subsystem

The key features of audio subsystem include:

- Audio processing is progressed by Reconfigurable Processor (RP)
- Low power audio subsystem
 - 5.1 channel I2S with 32-bit-width 64-depth FIFO
 - 128 KB audio play output buffer
 - Hardware mixer mixes primary and secondary sounds

1.2.5 Connectivity

The key features of connectivity include:

- PCM Audio Interface
 - 16-bit mono audio interface
 - Master mode only
 - Supports three port PCM interface
- AC97 Audio Interface
 - Independent channels for stereo PCM In, stereo PCM Out, and mono MIC In
 - 16-bit stereo (2-channel) audio
 - Variable sampling rate AC97 Codec interface (48 kHz and below)
 - Supports AC97 Full Specification
- SPDIF Interface (Tx only)
 - Linear PCM up to 24-bit per sample support
 - Non-Linear PCM formats such as AC3, MPEG1, and MPEG2 support
 - 2 × 24-bit buffers that are alternately filled with data
- I2S Bus Interface
 - Three I2S-bus for audio-codec interface with DMA-based operation
 - Serial, 8/16/24-bit per channel data transfers
 - Supports I2S, MSB-justified, and LSB-justified data format
 - Supports PCM 5.1 channel
 - Various bit clock frequency and codec clock frequency support
 - 16, 24, 32, 48 fs of bit clock frequency
 - 256, 384, 512, 768 fs of codec clock
 - Supports one port for 5.1 channel I2S (in Audio Subsystem) and two ports for 2 channel I2S

- Modem Interface
 - Asynchronous direct/ indirect 16-bit SRAM-style interface
 - On-chip 16 KB dual-ported SRAM buffer for direct interface
- I2C Bus Interface
 - Eight Multi-Master IIC-Bus
 - Serial, 8-bit oriented and bi-directional data transfers can be made at up to 100 Kbit/s in the standard mode
 - Up to 400 Kbit/s in the fast mode
- MIPI-Slimbus Interface
 - 6 ports. Each port has 16 entry FIFO with 32-bit width
- UART
 - Five UART with DMA-based or interrupt-based operation
 - Supports 5-bit, 6-bit, 7-bit, or 8-bit serial data transmit/ receive
 - Rx/Tx independent 256nbyte FIFO for UART0, 64 byte FIFO for UART1 and 16 byte FIFO for UART2/3/4
 - Programmable baud rate
 - Supports IrDA 1.0 SIR (115.2 Kbps) mode
 - Loop back mode for testing
 - Non-integer clock divides in Baud clock generation
- USB 2.0 Device
 - Complies to USB 2.0 Specification (Revision 1.0a)
 - Supports high-speed up to 480 Mbps
 - On-chip USB transceiver
- USB Host 2.0
 - Complies with the USB Host 2.0
 - Supports high-speed up to 480 Mbps
 - On-chip USB transceiver
- HS-MMC/SDIO Interface
 - Multimedia Card Protocol version 4.0 compatible (HS-MMC)
 - SD Memory Card Protocol version 2.0 compatible
 - DMA based or Interrupt based operation
 - 128 word FIFO for Tx/ Rx
 - Four ports HS-MMC or four ports SDIO
- SATA Controller
 - Compatible with the Serial ATA standard.
- PCI Express (not available with POP option)
 - Compatible with version 2.0 standard.

- SPI Interface
 - Complies with three Serial Peripheral Interface Protocol version 2.11
 - Rx/ Tx independent 64-word FIFO for SPI0 and 16-word FIFO for SPI1
 - DMA-based or interrupt-based operation
- GPIO

1.2.6 System Peripheral

The key features of system peripheral include:

- Real Time Clock
 - Full clock features: sec, min, hour, date, day, month, and year
 - 32.768 kHz operation
 - Alarm interrupt
 - Time-tick interrupt
- PLL
 - Four on-chip PLLs, APLL/MPLL/EPLL/VPLL
 - APLL generates ARM core and MSYS clocks
 - MPLL generates a system bus clock and special clocks
 - EPLL generates special clocks
 - VPLL generates clocks for video interface
- Keypad
 - 14 × 8 Key Matrix support
 - Provides internal de-bounce filter
- Timer with Pulse Width Modulation
 - Five channel 32-bit internal timer with interrupt-based operation
 - Three channel 32-bit Timer with PWM
 - Programmable duty cycle, frequency, and polarity
 - Dead-zone generation
 - Supports external clock source
- System timer
 - Accurate timer provides exact 1ms tick at any power mode except sleep
- Interrupt interval can be changed without stopping reference tick timer DMA
 - Micro-code programming based DMA
 - The specific instruction set provides flexibility to program DMA transfers
 - Supports linked list DMA function
 - Supports three enhanced built-in DMA with eight channels per DMA, so the total number of channels supported are 24
 - Supports one Memory-to-memory type optimized DMA and two Peripheral-to-memory type optimized DMA
 - M2M DMA supports up to 16 burst and P2M DMA supports up to 8 burst
- A/D Converter and Touch Screen Interface
 - 10 channel multiplexed ADC
 - Maximum 500 Ksamples/sec and 12-bit resolution

- Watch Dog Timer
- Power Management
 - Clock-gating control for components
 - Various low power modes are available such as AFTR, LPA, Deep Stop, and Sleep modes
 - Wake up sources in sleep mode are external interrupts, RTC alarm, Tick timer, HDMI CEC [\(1\)](#), and the key interface.
 - Deep Stop mode's wakes up sources are MMC, Touch screen interface, system timer [\(2\)](#), and entire wake up sources of Sleep mode.
 - AFTR and LPA mode's wakes up sources are 5.1 channel I2S and entire wake up source of Deep-Stop mode.

NOTE:

1. HDMIC CEC Wakeup works only when main oscillator clock is enabled in system-level power-down mode.
2. When oscillator clock is turned off in system-level power mode, RTCCLK should be the reference clock or main oscillator clock is enabled.

1.3 Conventions

1.3.1 Register RW Conventions

Symbol	Definition	Description
R	Read Only	The application has permission to read the Register field. Writes to read-only fields have no effect.
W	Write Only	The application has permission to write in the Register field.
R/W	Read & Write	The application has permission to read and writes in the Register field. The application sets this field by writing 1'b1 and clears it by writing 1'b0.
R/WC	Read & Write to clear	The application has permission to read and writes in the Register field. The application clears this field by writing 1'b1. A register write of 1'b0 has no effect on this field.
R/WS	Read & Write to set	The application has permission to read and writes in the Register field. The application sets this field by writing 1'b1. A register write of 1'b0 has no effect on this field.

1.3.2 Register Value Conventions

Expression	Description
x	Undefined bit
X	Undefined multiple bits
?	Undefined, but depends on the device or pin status
Device dependent	The value depends on the device
Pin value	The value depends on the pin status

2 Memory Map

2.1 Overview

Base Address	Limit Address	Size	Description
0x0000_0000	0x0001_0000	64 KB	iROM
0x0200_0000	0x0201_0000	64 KB	iROM (mirror of 0x0 – 0x10000)
0x0202_0000	0x0204_0000	128 KB	iRAM
0x0300_0000	0x0303_0000	192 KB	SRP-RAM or general purpose.
0x0381_0000	0x0383_0000	–	AudioSS's SFR region
0x0400_0000	0x0402_0000	128 KB	SROMC's Bank0
0x0500_0000	0x0502_0000	128 KB	SROMC's Bank1
0x0600_0000	0x0602_0000	128 KB	SROMC's Bank2
0x0700_0000	0x0702_0000	128 KB	SROMC's Bank3
0x0800_0000	0x0C00_0000	64 MB	PCI-e memory
0x0C00_0000	0x0C40_0000	–	OneNANDC's channel 0/1
0x0C60_0000	0x0CD0_0000	–	OneNANDC's SFR region
0x0CE0_0000	0x0D00_0000	–	NFCON's SFR region
0x1000_0000	0x1400_0000	–	SFR region
0x4000_0000	0xA000_0000	1.5 GB	DMC-0's memory
0xA000_0000	0x0000_0000	1.5 GB	DMC-1's memory

2.2 SFR Base Address

Base Address	IP	Base Address	IP
0x1000_0000	CHIPID	0x106A_0000	—
0x1001_0000	SYSREG	0x106B_0000	—
0x1002_0000	PMU	0x106C_0000	—
0x1003_0000	CMU_TOP_PART	0x1080_0000	sMDMA0
0x1004_0000	CMU_DMC_PART	0x1081_0000	nsMDMA0
0x1005_0000	SYSTimer	0x1083_0000	SSS
0x1006_0000	WDT	0x1088_0000	Coresight
0x1007_0000	RTC	0x1089_0000	Coresight
0x100A_0000	KEYIF	0x108A_0000	—
0x100B_0000	HDMI_CEC	0x10A0_0000	AXI_ACPX
0x1010_0000	SECKEY	0x10A4_0000	SMMU MDMA
0x1011_0000	TZPC0	0x10A5_0000	SMMU SSS
0x1012_0000	TZPC1	0x10AA_0000	QE MDMA
0x1013_0000	TZPC2	0x10AB_0000	QE SSS
0x1014_0000	TZPC3	0x10AC_0000	—
0x1015_0000	TZPC4	0x10AD_0000	—
0x1016_0000	TZPC5	0x10AE_0000	—
0x1017_0000	—	0x10AF_0000	—
0x1040_0000	DMC0	0x10E0_0000	AXI_AUDIO_X
0x1041_0000	DMC1	0x10E1_0000	—
0x1044_0000	Int_combiner	0x1100_0000	GPIO_right
0x1045_0000	—	0x1101_0000	—
0x1046_0000	IEM_IEC	0x1120_0000	AXI_GDR
0x1047_0000	IEM_AP	0x1121_0000	AXI_GPR
0x1048_0000	GIC_controller	0x1122_0000	—
0x1049_0000	GIC_distributor	0x1123_0000	—
0x104C_0000	Core_timers	0x1124_0000	ASYNCAXI_CAMIF
0x1050_0000	MPCore Private region	0x1125_0000	ASYNCAXI_LCD0
0x1051_0000	—	0x1126_0000	ASYNCAXI_LCD1
0x1060_0000	AXI_DMCD	0x1127_0000	ASYNCAXI_FSYSD
0x1062_0000	AXI_DMCSFRX	0x1129_0000	ASYNCAXI_MAUDIO
0x1064_0000	ASYNCAXI_GDL_DMCD	0x112A_0000	—
0x1065_0000	ASYNCAXI_GDR_DMCD	0x112B_0000	—
0x1068_0000	QE CPU	0x1140_0000	GPIO_left
0x1141_0000	—	0x11C9_0000	—

Base Address	IP	Base Address	IP
0x1160_0000	AXI_GDL	0x11CA_0000	–
0x1161_0000	AXI GPL	0x11CB_0000	–
0x1162_0000	–	0x11E0_0000	AXI_LCD0X
0x1163_0000	–	0x11E1_0000	–
0x1164_0000	ASYNCAXI IMAGE	0x11E2_0000	SMMUFIMD0
0x1165_0000	ASYNCAXI_TV	0x11E3_0000	–
0x1166_0000	ASYNCAXI_MFC_L	0x11E4_0000	–
0x116A_0000	–	0x11E5_0000	–
0x116B_0000	–	0x1200_0000	FIMD1
0x1180_0000	FIMC0	0x1201_0000	–
0x1181_0000	FIMC1	0x1202_0000	MIE1
0x1182_0000	FIMC2	0x1203_0000	–
0x1183_0000	FIMC3	0x1204_0000	–
0x1184_0000	JPEG	0x1208_0000	MIPI_DSI1
0x1188_0000	MIPI_CSI0	0x1209_0000	–
0x1189_0000	MIPI_CSI1	0x120A_0000	–
0x118A_0000	–	0x120B_0000	–
0x11A0_0000	AXI_CAMX	0x1220_0000	AXI_LCD1X
0x11A1_0000	–	0x1221_0000	–
0x11A2_0000	SMMUFIMC0	0x1222_0000	SMMUFIMD1
0x11A3_0000	SMMUFIMC1	0x1223_0000	–
0x11A4_0000	SMMUFIMC2	0x1224_0000	PPMU_LCD1
0x11A5_0000	SMMUFIMC3	0x1225_0000	–
0x11A6_0000	SMMUJPEG	0x1240_0000	PCIe
0x11A7_0000	QEFIMC0	0x1241_0000	–
0x11A8_0000	QEFIMC1	0x1248_0000	USBDEV_LINK
0x11A9_0000	QEFIMC2	0x1249_0000	–
0x11AA_0000	QEFIMC3	0x124A_0000	–
0x11AC_0000	–	0x124B_0000	–
0x11AD_0000	–	0x124C_0000	–
0x11C0_0000	FIMD0	0x1250_0000	TSI
0x11C1_0000	–	0x1251_0000	SDMMC0
0x11C2_0000	MIE0	0x1252_0000	SDMMC1
0x11C3_0000	–	0x1253_0000	SDMMC2
0x11C4_0000	–	0x1254_0000	SDMMC3
0x11C8_0000	MIPI_DSI0	0x1255_0000	SDMMC4
0x1256_0000	SATA	0x12A4_0000	SMMUDMA2

Base Address	IP	Base Address	IP
0x1257_0000	SROMC	0x12A5_0000	–
0x1258_0000	USBHOST_EHCI	0x12A6_0000	QEG2D
0x1259_0000	USBHOST_OHCI	0x12A7_0000	QERotator
0x125A_0000	–	0x12A8_0000	QEMDMA2
0x125B_0000	USBPHY_CON	0x12A9_0000	–
0x125C_0000	PCIePHY	0x12AA_0000	–
0x125D_0000	SATA PHY	0x12AB_0000	–
0x125E_0000	–	0x12C0_0000	VP
0x125F_0000	–	0x12C1_0000	Mixer
0x1260_0000	AXI_FSYSD	0x12C2_0000	TVENC
0x1261_0000	AXI_FSYSS	0x12C3_0000	–
0x1262_0000	SMMUPCIe	0x12D0_0000	HDMI0
0x1263_0000	–	0x12D1_0000	HDMI1
0x1264_0000	–	0x12D2_0000	HDMI2
0x1268_0000	PDMA0	0x12D3_0000	HDMI3
0x1269_0000	PDMA1	0x12D4_0000	HDMI4
0x126A_0000	PCIePHY_ctrl	0x12D5_0000	HDMI5
0x126B_0000	SATAPHY_Ctrl	0x12D6_0000	HDMI6
0x126C_0000	–	0x12D7_0000	–
0x126D_0000	–	0x12E0_0000	AXI_TVX
0x126E_0000	–	0x12E1_0000	–
0x126F_0000	–	0x12E2_0000	SMMUTV
0x1270_0000	–	0x12E3_0000	–
0x1271_0000	–	0x12E4_0000	–
0x1272_0000	–	0x12E5_0000	–
0x1284_0000	sMDMA2	0x1362_0000	SMMUMFC_L
0x1285_0000	nsMDMA2	0x1363_0000	SMMUMFC_R
0x1286_0000	–	0x1366_0000	–
0x12A0_0000	AXI_IMGX	0x1367_0000	–
0x12A1_0000	–	0x1368_0000	–
0x12A2_0000	SMMUG2D	0x1380_0000	UART0
0x12A3_0000	SMMURotator	0x1381_0000	UART1
0x1382_0000	UART2		
0x1383_0000	UART3		
0x1384_0000	UART4		
0x1385_0000	–		
0x1386_0000	I2C0		

Base Address	IP	Base Address	IP
0x1387_0000	I2C1		
0x1388_0000	I2C2		
0x1389_0000	I2C3		
0x138A_0000	I2C4		
0x138B_0000	I2C5		
0x138C_0000	I2C6		
0x138D_0000	I2C7		
0x138E_0000	I2CHDMI		
0x1391_0000	TSADC		
0x1392_0000	SPI0		
0x1393_0000	SPI1		
0x1394_0000	SPI2		
0x1395_0000	–		
0x1396_0000	I2S1		
0x1397_0000	I2S2		
0x1398_0000	PCM1		
0x1399_0000	PCM2		
0x139A_0000	AC97		
0x139B_0000	SPDIF		
0x139C_0000	–		
0x139D_0000	PWMTimer		
0x13A0_0000	ModemIF		

3 Chip ID

3.1 Overview

The Exynos4210 includes a Chip ID block for the software (SW) that sends and receives APB interface signals to the bus system.

NOTE: ARM architecture version of Exynos4210 is ARMv7 r2p7. But when you read Main ID register in CP15, it may show ARMv7 r2p1.

3.2 Register Description

3.2.1 Register Map Summary

- Base Address: 0x1000_0000

Register	Offset	Description	Reset Value
Product ID Register			
PRO_ID	0x0000	Product information (ID, package, revision)	0x4321_0211
PKG_ID	0x0004	Package information (POP Type, package)	–
OP_MODE	0x0008	[7:0] = XOM[7:0]	–
LOT_ID_0	0x0014	Lot & wafer information {ID[31:0]}	–
LOT_ID_1	0x0018	Lot & wafer information {16'b0, ID[47:32]}	–

- Base Address: 0x1010_0000

Register	Offset	Description	Reset Value
Package ID Register			
ROOTKEY0	0x0000	Root key[31:0]	–
ROOTKEY1	0x0004	Root key[63:32]	–
ROOTKEY2	0x0008	Root key[95:64]	–
ROOTKEY3	0x000C	Root key[127:96]	–

3.2.1.1 PRO_ID

- Address = 0x1000_0000, Reset Value = 0x4320_0XXX

Name	Bit	Type	Description	Reset Value
Product ID	[31:12]	R	Product ID	0x43210
RSVD	[11:10]	R	Reserved	0x0
Package	[9:8]	R	Package Information	0x2
MainRev	[7:4]	R	Main Revision Number	0x1
SubRev	[3:0]	R	Sub Revision Number	0x1

NOTE: PRO_ID register[7:0] depends on the e-fuse ROM value. As power on sequence is progressing, the e-fuse ROM values are loaded to the registers. It can read the loaded current e-fuse ROM values. An e-fuse ROM has main and sub revision numbers.

3.2.1.2 PKG_ID

- Address = 0x1000_0004, Reset Value = Undefined

Name	Bit	Type	Description	Reset Value
Ids_ARM	[31:24]	R	Static current of VDD_ARM @1.1 V	–
Ids_INT	[23:17]	R	Static current of VDD_INT @1.1 V	–
S_CODE	[16:12]	R	Speed Information	–
Package ID	[11:0]	R	Package information (POP Type, package)	–

NOTE: PKG_ID register [31:0] depends on the e-fuse ROM value.

4 GPIO Control

This chapter describes the General Purpose Input/Output (GPIO).

4.1 Overview

Exynos4210 includes 252 multi-functional input/output port pins and 156 memory port pins. There are 37 general port groups and 2 memory port groups as listed below:

- GPA0,GPA1: 14 in/out ports - 3xUART with flow control, UART without flow control, and/ or 2xI2C
- GPB: 8 in/out ports - 2xSPI and/or I2C and/ or IEM
- GPC0,GPC1: 10 in/out ports - 2xI2S, and/or 2xPCM, and/or AC97, SPDIF, I2C, and/ or SPI
- GPD0,GPD1: 8 in/out ports - PWM, I2C, and/or LCD I/F, I2C
- GPE0,GPE1,GPE2,GPE3,GPE4: 35 in/ out ports - Modem I/F, and/or CAM I/F, and/or TS I/F, and/ or LCD I/F, and/ or Trace I/F
- GPF0, GPF1,GPF2,GPF3: 30 in/ out ports - LCD I/F
- GPJ0, GPJ1: 13 in/out ports - CAM I/F
- GPK0, GPK1,GPK2,GPK3: 28 in/out ports - 4xMMC(4-bit MMC), and/ or 2xMMC(8-bit MMC)
- GPL0, GPL1: 11 in/out ports - GPS I/F
- GPL2: 8 in/ out ports - GPS debugging I/F or Key pad I/F
- GPX0, GPX1, GPX2, GPX3: 32 in/out ports - External wake-up, and/ or Key pad I/F

NOTE: These are in ALIVE region.

- GPZ: 7 in/out ports - Low Power I2S and/or PCM
- GPY0, GPY1, GPY2: 16 in/ out ports - Control signals of EBI (SROM, NF, OneNAND)
- GPY3,GPY4,GPY5,GPY6: 32 in/ out memory ports - EBI (For more information about EBI configuration, refer to Chapter 5, and 6)
- MP1_0 – MP1_9: 78 DRAM1 ports

NOTE: GPIO registers does not control these ports.

- MP2_0 – MP2_9: 78 DRAM2 ports

NOTE: GPIO registers does not control these ports.

- ETC0,ETC1,ETC6: 18 in/out ETC ports - JTAG, SLIMBUS, RESET, CLOCK
-

Caution: Do not leave a port in Input Pull-up/down disable state, when the port is not used or connected to an input pin without Pull-up/down. It may cause unexpected state and leakage current.

When a port is used as output function, Pull-up/down must be disabled.

4.2 Features

The key features of GPIO include:

- Controls 172 External Interrupts
- Controls 32 External Wake-up Interrupts
- 252 multi-functional input/ output ports
- Controls pin states in Sleep Mode except GPX0, GPX1, GPX2, and GPX3 (GPX^{*} pins are alive-pads)

4.3 Functional Description

4.3.1 General Purpose Input/Output Block Diagram

GPIO consists of two parts, namely, alive-part and off-part. In Alive-part power is supplied on sleep mode, but in off-part it is not the same. Therefore, the registers in alive-part keep their values during sleep mode.

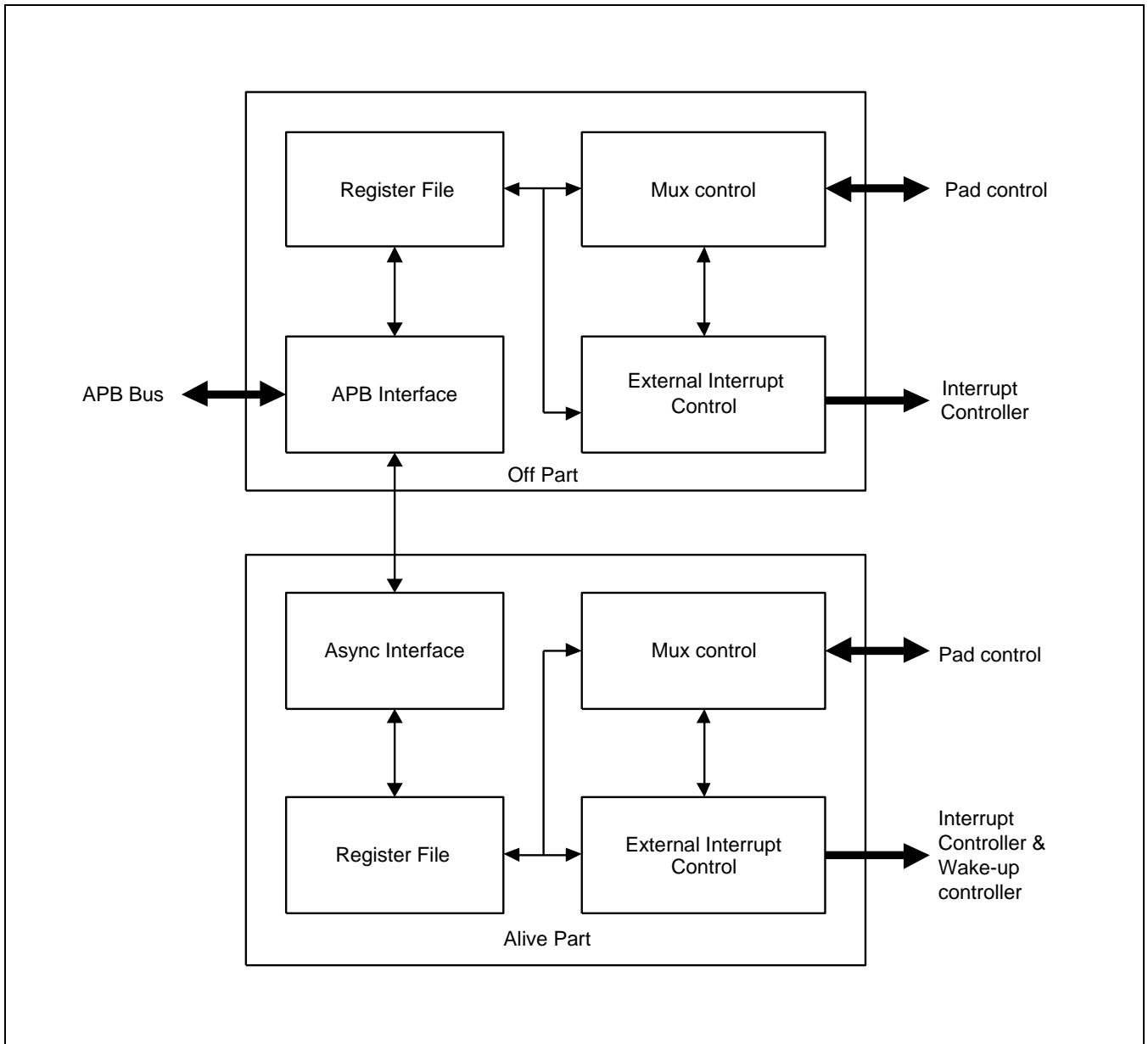


Figure 4-1 GPIO Block Diagram

4.4 Register Description

4.4.1 Register Map Summary

- Base Address: 0x1140_0000

Register	Offset	Description	Reset Value
Part 1			
GPA0CON	0x0000	Port group GPA0 configuration register	0x0000_0000
GPA0DAT	0x0004	Port group GPA0 data register	0x00
GPA0PUD	0x0008	Port group GPA0 Pull-up/down Register	0x5555
GPA0DRV	0x000C	Port group GPA0 drive strength control register	0x00_0000
GPA0CONPDN	0x0010	Port group GPA0 power down mode configuration Register	0x0000
GPA0PUDPDN	0x0014	Port group GPA0 power down mode pull-up/down register	0x0000
GPA1CON	0x0020	Port group GPA1 configuration register	0x0000_0000
GPA1DAT	0x0024	Port group GPA1 data register	0x00
GPA1PUD	0x0028	Port group GPA1 pull-up/down register	0x0555
GPA1DRV	0x002C	Port group GPA1 drive strength control register	0x00_0000
GPA1CONPDN	0x0030	Port group GPA1 power down mode configuration register	0x0000
GPA1PUDPDN	0x0034	Port group GPA1 power down mode pull-up/down register	0x0000
GPBCON	0x0040	Port group GPB configuration register	0x0000_0000
GPBDAT	0x0044	Port group GPB data register	0x00
GPBPUD	0x0048	Port group GPB pull-up/down register	0x5555
GPBDRV	0x004C	Port group GPB drive strength control register	0x00_0000
GPBCONPDN	0x0050	Port group GPB power down mode configuration register	0x0000
GPBPUDPDN	0x0054	Port group GPB power down mode pull-up/down register	0x0000
GPC0CON	0x0060	Port group GPC0 configuration register	0x0000_0000
GPC0DAT	0x0064	Port group GPC0 data register	0x00
GPC0PUD	0x0068	Port group GPC0 pull-up/down register	0x0155
GPC0DRV	0x006C	Port group GPC0 drive strength control register	0x00_0000
GPC0CONPDN	0x0070	Port group GPC0 power down mode configuration register	0x0000
GPC0PUDPDN	0x0074	Port group GPC0 power down mode pull-up/down register	0x0000
GPC1CON	0x0080	Port group GPC1 configuration register	0x0000_0000
GPC1DAT	0x0084	Port group GPC1 data register	0x00

Register	Offset	Description	Reset Value
GPC1PUD	0x0088	Port group GPC1 pull-up/down register	0x0155
GPC1DRV	0x008C	Port group GPC1 drive strength control register	0x00_0000
GPC1CONPDN	0x0090	Port group GPC1 power down mode configuration register	0x0000
GPC1PUDPDN	0x0094	Port group GPC1 power down mode pull-up/down register	0x0000
GPD0CON	0x00A0	Port group GPD0 configuration register	0x0000_0000
GPD0DAT	0x00A4	Port group GPD0 data register	0x00
GPD0PUD	0x00A8	Port group GPD0 pull-up/down register	0x0055
GPD0DRV	0x00AC	Port group GPD0 drive strength control register	0x00_0000
GPD0CONPDN	0x00B0	Port group GPD0 power down mode configuration register	0x0000
GPD0PUDPDN	0x00B4	Port group GPD0 power down mode pull-up/down register	0x0000
GPD1CON	0x00C0	Port group GPD1 configuration register	0x0000_0000
GPD1DAT	0x00C4	Port group GPD1 data register	0x00
GPD1PUD	0x00C8	Port group GPD1 pull-up/down register	0x0055
GPD1DRV	0x00CC	Port group GPD1 drive strength control register	0x00_0000
GPD1CONPDN	0x00D0	Port group GPD1 power down mode configuration register	0x0000
GPD1PUDPDN	0x00D4	Port group GPD1 power down mode pull-up/down register	0x0000
GPE0CON	0x00E0	Port group GPE0 configuration register	0x0000_0000
GPE0DAT	0x00E4	Port group GPE0 data register	0x00
GPE0PUD	0x00E8	Port group GPE0 pull-up/down register	0x0155
GPE0DRV	0x00EC	Port group GPE0 drive strength control register	0x00_0000
GPE0CONPDN	0x00F0	Port group GPE0 power down mode configuration register	0x0000
GPE0PUDPDN	0x00F4	Port group GPE0 power down mode pull-up/down register	0x0000
GPE1CON	0x0100	Port group GPE1 configuration register	0x0000_0000
GPE1DAT	0x0104	Port group GPE1 data register	0x00
GPE1PUD	0x0108	Port group GPE1 pull-up/down register	0x5555
GPE1DRV	0x010C	Port group GPE1 drive strength control register	0x00_0000
GPE1CONPDN	0x0110	Port group GPE1 power down mode configuration register	0x0000
GPE1PUDPDN	0x0114	Port group GPE1 power down mode pull-up/down register	0x0000
GPE2CON	0x0120	Port group GPE2 configuration register	0x0000_0000

Register	Offset	Description	Reset Value
GPE2DAT	0x0124	Port group GPE2 data register	0x00
GPE2PUD	0x0128	Port group GPE2 pull-up/down register	0x0555
GPE2DRV	0x012C	Port group GPE2 drive strength control register	0x00_0000
GPE2CONPDN	0x0130	Port group GPE2 power down mode configuration register	0x0000
GPE2PUDPDN	0x0134	Port group GPE2 power down mode pull-up/down register	0x0000
GPE3CON	0x0140	Port group GPE3 configuration register	0x0000_0000
GPE3DAT	0x0144	Port group GPE3 data register	0x00
GPE3PUD	0x0148	Port group GPE3 pull-up/down register	0x5555
GPE3DRV	0x014C	Port group GPE3 drive strength control register	0x00_0000
GPE3CONPDN	0x0150	Port group GPE3 power down mode configuration register	0x0000
GPE3PUDPDN	0x0154	Port group GPE3 power down mode pull-up/down register	0x0000
GPE4CON	0x0160	Port group GPE4 configuration register	0x0000_0000
GPE4DAT	0x0164	Port group GPE4 data register	0x00
GPE4PUD	0x0168	Port group GPE4 pull-up/down register	0x5555
GPE4DRV	0x016C	Port group GPE4 drive strength control register	0x00_0000
GPE4CONPDN	0x0170	Port group GPE4 power down mode configuration register	0x0000
GPE4PUDPDN	0x0174	Port group GPE4 power down mode pull-up/down register	0x0000
GPF0CON	0x0180	Port group GPF0 configuration register	0x0000_0000
GPF0DAT	0x0184	Port group GPF0 data register	0x00
GPF0PUD	0x0188	Port group GPF0 pull-up/down register	0x5555
GPF0DRV	0x018C	Port group GPF0 drive strength control register	0x00_0000
GPF0CONPDN	0x0190	Port group GPF0 power down mode configuration register	0x0000
GPF0PUDPDN	0x0194	Port group GPF0 power down mode pull-up/down register	0x0000
GPF1CON	0x01A0	Port group GPF1 configuration register	0x0000_0000
GPF1DAT	0x01A4	Port group GPF1 data register	0x00
GPF1PUD	0x01A8	Port group GPF1 pull-up/down register	0x5555
GPF1DRV	0x01AC	Port group GPF1 drive strength control register	0x00_0000
GPF1CONPDN	0x01B0	Port group GPF1 power down mode configuration register	0x0000
GPF1PUDPDN	0x01B4	Port group GPF1 power down mode pull-up/down register	0x0000

Register	Offset	Description	Reset Value
GPF2CON	0x01C0	Port group GPF2 configuration register	0x0000_0000
GPF2DAT	0x01C4	Port group GPF2 data register	0x00
GPF2PUD	0x01C8	Port group GPF2 pull-up/down register	0x5555
GPF2DRV	0x01CC	Port group GPF2 drive strength control register	0x00_0000
GPF2CONPDN	0x01D0	Port group GPF2 power down mode configuration register	0x0000
GPF2PUDPDN	0x01D4	Port group GPF2 power down mode pull-up/down register	0x0000
GPF3CON	0x01E0	Port group GPF3 configuration register	0x0000_0000
GPF3DAT	0x01E4	Port group GPF3 data register	0x00
GPF3PUD	0x01E8	Port group GPF3 pull-up/down register	0x0555
GPF3DRV	0x01EC	Port group GPF3 drive strength control register	0x00_0000
GPF3CONPDN	0x01F0	Port group GPF3 power down mode configuration register	0x0000
GPF3PUDPDN	0x01F4	Port group GPF3 power down mode pull-up/down register	0x0000
ETC0PUD	0x0208	Port group ETC0 pull-up/down register	0x0400
ETC0DRV	0x020C	Port group ETC0 drive strength control register	0x00_0000
ETC1PUD	0x0228	Port group ETC1 pull-up/down register	0x0005
ETC1DRV	0x022C	Port group ETC1 drive strength control register	0x00_0000
EXT_INT1_CON	0x0700	External interrupt EXT_INT1 configuration register	0x0000_0000
EXT_INT2_CON	0x0704	External interrupt EXT_INT2 configuration register	0x0000_0000
EXT_INT3_CON	0x0708	External interrupt EXT_INT3 configuration register	0x0000_0000
EXT_INT4_CON	0x070C	External interrupt EXT_INT4 configuration register	0x0000_0000
EXT_INT5_CON	0x0710	External interrupt EXT_INT5 configuration register	0x0000_0000
EXT_INT6_CON	0x0714	External interrupt EXT_INT6 configuration register	0x0000_0000
EXT_INT7_CON	0x0718	External interrupt EXT_INT7 configuration register	0x0000_0000
EXT_INT8_CON	0x071C	External interrupt EXT_INT8 configuration register	0x0000_0000
EXT_INT9_CON	0x0720	External interrupt EXT_INT9 configuration register	0x0000_0000
EXT_INT10_CON	0x0724	External interrupt EXT_INT10 configuration register	0x0000_0000
EXT_INT11_CON	0x0728	External interrupt EXT_INT11 configuration register	0x0000_0000
EXT_INT12_CON	0x072C	External interrupt EXT_INT12 configuration register	0x0000_0000
EXT_INT13_CON	0x0730	External interrupt EXT_INT13 configuration register	0x0000_0000
EXT_INT14_CON	0x0734	External interrupt EXT_INT14 configuration register	0x0000_0000
EXT_INT15_CON	0x0738	External interrupt EXT_INT15 configuration register	0x0000_0000
EXT_INT16_CON	0x073C	External interrupt EXT_INT16 configuration register	0x0000_0000
EXT_INT1_FLTCNO	0x0800	External interrupt EXT_INT1 filter configuration register 0	0x0000_0000

Register	Offset	Description	Reset Value
EXT_INT1_FLTCON1	0x0804	External interrupt EXT_INT1 filter configuration register 1	0x0000_0000
EXT_INT2_FLTCON0	0x0808	External interrupt EXT_INT2 filter configuration register 0	0x0000_0000
EXT_INT2_FLTCON1	0x080C	External interrupt EXT_INT2 filter configuration register 1	0x0000_0000
EXT_INT3_FLTCON0	0x0810	External interrupt EXT_INT3 filter configuration register 0	0x0000_0000
EXT_INT3_FLTCON1	0x0814	External interrupt EXT_INT3 filter configuration register 1	0x0000_0000
EXT_INT4_FLTCON0	0x0818	External interrupt EXT_INT4 filter configuration register 0	0x0000_0000
EXT_INT4_FLTCON1	0x081C	External interrupt EXT_INT4 filter configuration register 1	0x0000_0000
EXT_INT5_FLTCON0	0x0820	External interrupt EXT_INT5 filter configuration register 0	0x0000_0000
EXT_INT5_FLTCON1	0x0824	External interrupt EXT_INT5 filter configuration register 1	0x0000_0000
EXT_INT6_FLTCON0	0x0828	External interrupt EXT_INT6 Filter Configuration Register 0	0x0000_0000
EXT_INT6_FLTCON1	0x082C	External Interrupt EXT_INT6 filter configuration register 1	0x0000_0000
EXT_INT7_FLTCON0	0x0830	External interrupt EXT_INT7 filter configuration register 0	0x0000_0000
EXT_INT7_FLTCON1	0x0834	External interrupt EXT_INT7 filter configuration register 1	0x0000_0000
EXT_INT8_FLTCON0	0x0838	External interrupt EXT_INT8 filter configuration register 0	0x0000_0000
EXT_INT8_FLTCON1	0x083C	External interrupt EXT_INT8 filter configuration register 1	0x0000_0000
EXT_INT9_FLTCON0	0x0840	External interrupt EXT_INT9 filter configuration register 0	0x0000_0000
EXT_INT9_FLTCON1	0x0844	External interrupt EXT_INT9 filter configuration register 1	0x0000_0000
EXT_INT10_FLTCON0	0x0848	External interrupt EXT_INT10 filter configuration register 0	0x0000_0000
EXT_INT10_FLTCON1	0x084C	External interrupt EXT_INT10 filter configuration register 1	0x0000_0000
EXT_INT11_FLTCON0	0x0850	External interrupt EXT_INT11 filter configuration register 0	0x0000_0000
EXT_INT11_FLTCON1	0x0854	External interrupt EXT_INT11 filter configuration register 1	0x0000_0000
EXT_INT12_FLTCON0	0x0858	External interrupt EXT_INT12 filter configuration	0x0000_0000

Register	Offset	Description	Reset Value
		register 0	
EXT_INT12_FLTCON1	0x085C	External interrupt EXT_INT12 filter configuration register 1	0x0000_0000
EXT_INT13_FLTCON0	0x0860	External interrupt EXT_INT13 filter configuration register 0	0x0000_0000
EXT_INT13_FLTCON1	0x0864	External interrupt EXT_INT13 filter configuration register 1	0x0000_0000
EXT_INT14_FLTCON0	0x0868	External interrupt EXT_INT14 filter configuration register 0	0x0000_0000
EXT_INT14_FLTCON1	0x086C	External interrupt EXT_INT14 filter configuration register 1	0x0000_0000
EXT_INT15_FLTCON0	0x0870	External interrupt EXT_INT15 filter configuration register 0	0x0000_0000
EXT_INT15_FLTCON1	0x0874	External interrupt EXT_INT15 filter configuration register 1	0x0000_0000
EXT_INT16_FLTCON0	0x0878	External interrupt EXT_INT16 filter configuration register 0	0x0000_0000
EXT_INT16_FLTCON1	0x087C	External interrupt EXT_INT16 filter configuration register 1	0x0000_0000
EXT_INT1_MASK	0x0900	External interrupt EXT_INT1 mask register	0x0000_00FF
EXT_INT2_MASK	0x0904	External interrupt EXT_INT2 mask register	0x0000_003F
EXT_INT3_MASK	0x0908	External interrupt EXT_INT3 mask register	0x0000_00FF
EXT_INT4_MASK	0x090C	External interrupt EXT_INT4 mask register	0x0000_001F
EXT_INT5_MASK	0x0910	External interrupt EXT_INT5 mask register	0x0000_001F
EXT_INT6_MASK	0x0914	External interrupt EXT_INT6 mask register	0x0000_000F
EXT_INT7_MASK	0x0918	External interrupt EXT_INT7 mask register	0x0000_000F
EXT_INT8_MASK	0x091C	External interrupt EXT_INT8 mask register	0x0000_001F
EXT_INT9_MASK	0x0920	External interrupt EXT_INT9 mask register	0x0000_00FF
EXT_INT10_MASK	0x0924	External interrupt EXT_INT10 mask register	0x0000_003F
EXT_INT11_MASK	0x0928	External interrupt EXT_INT11 mask register	0x0000_00FF
EXT_INT12_MASK	0x092C	External interrupt EXT_INT12 mask register	0x0000_00FF
EXT_INT13_MASK	0x0930	External interrupt EXT_INT13 mask register	0x0000_00FF
EXT_INT14_MASK	0x0934	External interrupt EXT_INT14 mask register	0x0000_00FF
EXT_INT15_MASK	0x0938	External interrupt EXT_INT15 mask register	0x0000_00FF
EXT_INT16_MASK	0x093C	External interrupt EXT_INT16 mask register	0x0000_003F
EXT_INT1_PEND	0x0A00	External interrupt EXT_INT1 pending register	0x0000_0000
EXT_INT2_PEND	0x0A04	External interrupt EXT_INT2 pending register	0x0000_0000
EXT_INT3_PEND	0x0A08	External interrupt EXT_INT3 pending register	0x0000_0000
EXT_INT4_PEND	0x0A0C	External interrupt EXT_INT4 pending register	0x0000_0000

Register	Offset	Description	Reset Value
EXT_INT5_PEND	0x0A10	External interrupt EXT_INT5 pending register	0x0000_0000
EXT_INT6_PEND	0x0A14	External interrupt EXT_INT6 pending register	0x0000_0000
EXT_INT7_PEND	0x0A18	External interrupt EXT_INT7 pending register	0x0000_0000
EXT_INT8_PEND	0x0A1C	External interrupt EXT_INT8 pending register	0x0000_0000
EXT_INT9_PEND	0x0A20	External interrupt EXT_INT9 pending register	0x0000_0000
EXT_INT10_PEND	0x0A24	External interrupt EXT_INT10 pending register	0x0000_0000
EXT_INT11_PEND	0x0A28	External interrupt EXT_INT11 pending register	0x0000_0000
EXT_INT12_PEND	0x0A2C	External interrupt EXT_INT12 pending register	0x0000_0000
EXT_INT13_PEND	0x0A30	External interrupt EXT_INT13 pending register	0x0000_0000
EXT_INT14_PEND	0x0A34	External interrupt EXT_INT14 pending register	0x0000_0000
EXT_INT15_PEND	0x0A38	External interrupt EXT_INT15 pending register	0x0000_0000
EXT_INT16_PEND	0x0A3C	External interrupt EXT_INT16 pending register	0x0000_0000
EXT_INT_SERVICE_XA	0x0B08	Current service register	0x0000_0000
EXT_INT_SERVICE_PEND_XA	0x0B0C	Current service pending register	0x0000_0000
EXT_INT_GRPFXPRI_XA	0x0B10	External interrupt group fixed priority control register	0x0000_0000
EXT_INT1_FIXPRI	0x0B14	External interrupt 1 fixed priority control register	0x0000_0000
EXT_INT2_FIXPRI	0x0B18	External interrupt 2 fixed priority control register	0x0000_0000
EXT_INT3_FIXPRI	0x0B1C	External interrupt 3 fixed priority control register	0x0000_0000
EXT_INT4_FIXPRI	0x0B20	External interrupt 4 fixed priority control register	0x0000_0000
EXT_INT5_FIXPRI	0x0B24	External interrupt 5 fixed priority control register	0x0000_0000
EXT_INT6_FIXPRI	0x0B28	External interrupt 6 fixed priority control register	0x0000_0000
EXT_INT7_FIXPRI	0x0B2C	External interrupt 7 fixed priority control register	0x0000_0000
EXT_INT8_FIXPRI	0x0B30	External interrupt 8 fixed priority control register	0x0000_0000
EXT_INT9_FIXPRI	0x0B34	External interrupt 9 fixed priority control register	0x0000_0000
EXT_INT10_FIXPRI	0x0B38	External interrupt 10 fixed priority control register	0x0000_0000
EXT_INT11_FIXPRI	0x0B3C	External interrupt 11 fixed priority control register	0x0000_0000
EXT_INT12_FIXPRI	0x0B40	External interrupt 12 fixed priority control register	0x0000_0000
EXT_INT13_FIXPRI	0x0B44	External interrupt 13 fixed priority control register	0x0000_0000
EXT_INT14_FIXPRI	0x0B48	External interrupt 14 fixed priority control register	0x0000_0000
EXT_INT15_FIXPRI	0x0B4C	External interrupt 15 fixed priority control register	0x0000_0000
EXT_INT16_FIXPRI	0x0B50	External interrupt 16 fixed priority control register	0x0000_0000

- Base Address: 0x1100_0000

Register	Offset	Description	Reset Value
Part 2			
GPJ0CON	0x0000	Port group GPJ0 configuration register	0x0000_0000
GPJ0DAT	0x0004	Port group GPJ0 data register	0x00
GPJ0PUD	0x0008	Port group GPJ0 pull-up/down register	0x5555
GPJ0DRV	0x000C	Port group GPJ0 drive strength control register	0x00_0000
GPJ0CONPDN	0x0010	Port group GPJ0 power down mode configuration register	0x0000
GPJ0PUDPDN	0x0014	Port group GPJ0 power down mode pull-up/down register	0x0000
GPJ1CON	0x0020	Port group GPJ1 configuration register	0x0000_0000
GPJ1DAT	0x0024	Port Group GPJ1 data register	0x00
GPJ1PUD	0x0028	Port Group GPJ1 pull-up/down register	0x0155
GPJ1DRV	0x002C	Port Group GPJ1 drive strength control register	0x00_0000
GPJ1CONPDN	0x0030	Port Group GPJ1 power down mode configuration register	0x0000
GPJ1PUDPDN	0x0034	Port Group GPJ1 power down mode pull-up/down register	0x0000
GPK0CON	0x0040	Port group GPK0 configuration register	0x0000_0000
GPK0DAT	0x0044	Port group GPK0 data register	0x00
GPK0PUD	0x0048	Port group GPK0 pull-up/down register	0x1555
GPK0DRV	0x004C	Port group GPK0 drive strength control register	0x00_2AAA
GPK0CONPDN	0x0050	Port group GPK0 power down mode configuration register	0x0000
GPK0PUDPDN	0x0054	Port group GPK0 power down mode pull-up/down register	0x0000
GPK1CON	0x0060	Port group GPK1 configuration register	0x0000_0000
GPK1DAT	0x0064	Port group GPK1 data register	0x00
GPK1PUD	0x0068	Port group GPK1 pull-up/down register	0x1555
GPK1DRV	0x006C	Port group GPK1 drive strength control register	0x00_0000
GPK1CONPDN	0x0070	Port group GPK1 power down mode configuration register	0x0000
GPK1PUDPDN	0x0074	Port group GPK1 power down mode pull-up/down register	0x0000
GPK2CON	0x0080	Port group GPK2 configuration register	0x0000_0000
GPK2DAT	0x0084	Port group GPK2 data register	0x00
GPK2PUD	0x0088	Port group GPK2 pull-up/down register	0x1555
GPK2DRV	0x008C	Port group GPK2 drive strength control register	0x00_0000
GPK2CONPDN	0x0090	Port group GPK2 power down mode configuration	0x0000

Register	Offset	Description	Reset Value
		register	
GPK2PUDPDN	0x0094	Port group GPK2 power down mode pull-up/down register	0x0000
GPK3CON	0x00A0	Port group GPK3 configuration register	0x0000_0000
GPK3DAT	0x00A4	Port group GPK3 data register	0x00
GPK3PUD	0x00A8	Port group GPK3 pull-up/down register	0x1555
GPK3DRV	0x00AC	Port group GPK3 drive strength control register	0x00_0000
GPK3CONPDN	0x00B0	Port group GPK3 power down mode configuration register	0x0000
GPK3PUDPDN	0x00B4	Port group GPK3 power down mode pull-up/down register	0x0000
GPL0CON	0x00C0	Port group GPL0 configuration register	0x0000_0000
GPL0DAT	0x00C4	Port group GPL0 data register	0x00
GPL0PUD	0x00C8	Port group GPL0 pull-up/down register	0x5555
GPL0DRV	0x00CC	Port group GPL0 drive strength control register	0x00_0000
GPL0CONPDN	0x00D0	Port group GPL0 power down mode configuration register	0x0000
GPL0PUDPDN	0x00D4	Port group GPL0 power down mode pull-up/down register	0x0000
GPL1CON	0x00E0	Port group GPL1 configuration register	0x0000_0000
GPL1DAT	0x00E4	Port group GPL1 data register	0x00
GPL1PUD	0x00E8	Port group GPL1 pull-up/down register	0x0015
GPL1DRV	0x00EC	Port group GPL1 drive strength control register	0x00_0000
GPL1CONPDN	0x00F0	Port group GPL1 power down mode configuration register	0x0000
GPL1PUDPDN	0x00F4	Port group GPL1 power down mode pull-up/down register	0x0000
GPL2CON	0x0100	Port group GPL2 configuration register	0x0000_0000
GPL2DAT	0x0104	Port group GPL2 data register	0x00
GPL2PUD	0x0108	Port group GPL2 pull-up/down register	0x5555
GPL2DRV	0x010C	Port group GPL2 drive strength control register	0x00_0000
GPL2CONPDN	0x0110	Port group GPL2 power down mode configuration register	0x0000
GPL2PUDPDN	0x0114	Port group GPL2 power down mode pull-up/down register	0x0000
GPY0CON	0x0120	Port group GPY0 configuration register	0x0022_5522
GPY0DAT	0x0124	Port group GPY0 data register	0x00
GPY0PUD	0x0128	Port group GPY0 pull-up/down register	0x0000
GPY0DRV	0x012C	Port group GPY0 drive strength control register	0x00_0AAA

Register	Offset	Description	Reset Value
GPY0CONPDN	0x0130	Port group GPY0 power down mode configuration register	0x0000
GPY0PUDPDN	0x0134	Port group GPY0 power down mode pull-up/down register	0x0000
GPY1CON	0x0140	Port group GPY1 configuration register	0x0000_2222
GPY1DAT	0x0144	Port group GPY1 data register	0x00
GPY1PUD	0x0148	Port group GPY1 pull-up/down register	0x0000
GPY1DRV	0x014C	Port group GPY1 drive strength control register	0x00_00AA
GPY1CONPDN	0x0150	Port group GPY1 power down mode configuration register	0x0000
GPY1PUDPDN	0x0154	Port group GPY1 power down mode pull-up/down register	0x0000
GPY2CON	0x0160	Port group GPY2 configuration register	0x0025_5555
GPY2DAT	0x0164	Port group GPY2 data register	0x00
GPY2PUD	0x0168	Port group GPY2 pull-up/down register	0x0000
GPY2DRV	0x016C	Port group GPY2 drive strength control register	0x00_0AAA
GPY2CONPDN	0x0170	Port group GPY2 power down mode configuration register	0x0000
GPY2PUDPDN	0x0174	Port group GPY2 power down mode pull-up/down register	0x0000
GPY3CON	0x0180	Port group GPY3 configuration register	0x2222_2222
GPY3DAT	0x0184	Port group GPY3 data register	0x00
GPY3PUD	0x0188	Port group GPY3 pull-up/down register	0x0000
GPY3DRV	0x018C	Port group GPY3 drive strength control register	0x00_AAAA
GPY3CONPDN	0x0190	Port group GPY3 power down mode configuration register	0x0000
GPY3PUDPDN	0x0194	Port group GPY3 power down mode pull-up/down register	0x0000
GPY4CON	0x01A0	Port group GPY4 configuration register	0x2222_2222
GPY4DAT	0x01A4	Port group GPY4 data register	0x00
GPY4PUD	0x01A8	Port group GPY4 pull-up/down register	0x0000
GPY4DRV	0x01AC	Port group GPY4 drive strength control register	0x00_AAAA
GPY4CONPDN	0x01B0	Port group GPY4 power down mode configuration register	0x0000
GPY4PUDPDN	0x01B4	Port group GPY4 power down mode pull-up/down register	0x0000
GPY5CON	0x01C0	Port group GPY5 configuration register	0x2222_2222
GPY5DAT	0x01C4	Port group GPY5 data register	0x00
GPY5PUD	0x01C8	Port group GPY5 pull-up/down register	0x0000

Register	Offset	Description	Reset Value
GPY5DRV	0x01CC	Port group GPY5 drive strength control register	0x00_AAAA
GPY5CONPDN	0x01D0	Port group GPY5 power down mode configuration register	0x0000
GPY5PUDPDN	0x01D4	Port group GPY5 power down mode pull-up/down register	0x0000
GPY6CON	0x01E0	Port group GPY6 configuration register	0x2222_2222
GPY6DAT	0x01E4	Port group GPY6 data register	0x00
GPY6PUD	0x01E8	Port group GPY6 pull-up/down register	0x0000
GPY6DRV	0x01EC	Port group GPY6 drive strength control register	0x00_AAAA
GPY6CONPDN	0x01F0	Port group GPY6 power down mode configuration register	0x0000
GPY6PUDPDN	0x01F4	Port group GPY6 power down mode pull-up/down register	0x0000
ETC6PUD	0x0228	Port group ETC6 pull-up/down register	0xC0C0
ETC6DRV	0x022C	Port group ETC6 drive strength control register	0x00_0000
EXT_INT21_CON	0x0700	External interrupt EXT_INT21 configuration register	0x0000_0000
EXT_INT22_CON	0x0704	External interrupt EXT_INT22 configuration register	0x0000_0000
EXT_INT23_CON	0x0708	External interrupt EXT_INT23 configuration register	0x0000_0000
EXT_INT24_CON	0x070C	External interrupt EXT_INT24 configuration register	0x0000_0000
EXT_INT25_CON	0x0710	External interrupt EXT_INT25 configuration register	0x0000_0000
EXT_INT26_CON	0x0714	External interrupt EXT_INT26 configuration register	0x0000_0000
EXT_INT27_CON	0x0718	External interrupt EXT_INT27 configuration register	0x0000_0000
EXT_INT28_CON	0x071C	External interrupt EXT_INT28 configuration register	0x0000_0000
EXT_INT29_CON	0x0720	External interrupt EXT_INT29 configuration register	0x0000_0000
EXT_INT21_FLTCON0	0x0800	External interrupt EXT_INT21 filter configuration register 0	0x0000_0000
EXT_INT21_FLTCON1	0x0804	External interrupt EXT_INT21 filter configuration register 1	0x0000_0000
EXT_INT22_FLTCON0	0x0808	External interrupt EXT_INT22 filter configuration register 0	0x0000_0000
EXT_INT22_FLTCON1	0x080C	External interrupt EXT_INT22 filter configuration register 1	0x0000_0000
EXT_INT23_FLTCON0	0x0810	External interrupt EXT_INT23 filter configuration register 0	0x0000_0000
EXT_INT23_FLTCON1	0x0814	External interrupt EXT_INT23 filter configuration register 1	0x0000_0000
EXT_INT24_FLTCON0	0x0818	External interrupt EXT_INT24 filter configuration register 0	0x0000_0000
EXT_INT24_FLTCON1	0x081C	External interrupt EXT_INT24 filter configuration register 1	0x0000_0000

Register	Offset	Description	Reset Value
EXT_INT25_FLTCON0	0x0820	External interrupt EXT_INT25 filter configuration register 0	0x0000_0000
EXT_INT25_FLTCON1	0x0824	External interrupt EXT_INT25 filter configuration register 1	0x0000_0000
EXT_INT26_FLTCON0	0x0828	External interrupt EXT_INT26 filter configuration register 0	0x0000_0000
EXT_INT26_FLTCON1	0x082C	External interrupt EXT_INT26 filter configuration register 1	0x0000_0000
EXT_INT27_FLTCON0	0x0830	External interrupt EXT_INT27 filter configuration register 0	0x0000_0000
EXT_INT27_FLTCON1	0x0834	External interrupt EXT_INT27 filter configuration register 1	0x0000_0000
EXT_INT28_FLTCON0	0x0838	External interrupt EXT_INT28 filter configuration register 0	0x0000_0000
EXT_INT28_FLTCON1	0x083C	External interrupt EXT_INT28 filter configuration register 1	0x0000_0000
EXT_INT29_FLTCON0	0x0840	External interrupt EXT_INT29 filter configuration register 0	0x0000_0000
EXT_INT29_FLTCON1	0x0844	External interrupt EXT_INT29 filter configuration register 1	0x0000_0000
EXT_INT21_MASK	0x0900	External interrupt EXT_INT21 mask register	0x0000_00FF
EXT_INT22_MASK	0x0904	External interrupt EXT_INT22 mask register	0x0000_001F
EXT_INT23_MASK	0x0908	External interrupt EXT_INT23 mask register	0x0000_007F
EXT_INT24_MASK	0x090C	External interrupt EXT_INT24 mask register	0x0000_007F
EXT_INT25_MASK	0x0910	External interrupt EXT_INT25 mask register	0x0000_007F
EXT_INT26_MASK	0x0914	External interrupt EXT_INT26 mask register	0x0000_007F
EXT_INT27_MASK	0x0918	External interrupt EXT_INT27 mask register	0x0000_00FF
EXT_INT28_MASK	0x091C	External interrupt EXT_INT28 mask register	0x0000_0007
EXT_INT29_MASK	0x0920	External interrupt EXT_INT29 mask register	0x0000_00FF
EXT_INT21_PEND	0x0A00	External interrupt EXT_INT21 pending register	0x0000_0000
EXT_INT22_PEND	0x0A04	External interrupt EXT_INT22 pending register	0x0000_0000
EXT_INT23_PEND	0x0A08	External interrupt EXT_INT23 pending register	0x0000_0000
EXT_INT24_PEND	0xA0C	External interrupt EXT_INT24 pending register	0x0000_0000
EXT_INT25_PEND	0x0A10	External interrupt EXT_INT25 pending register	0x0000_0000
EXT_INT26_PEND	0x0A14	External interrupt EXT_INT26 pending register	0x0000_0000
EXT_INT27_PEND	0x0A18	External interrupt EXT_INT27 pending register	0x0000_0000
EXT_INT28_PEND	0x0A1C	External interrupt EXT_INT28 pending register	0x0000_0000
EXT_INT29_PEND	0x0A20	External interrupt EXT_INT29 pending register	0x0000_0000
EXT_INT_SERVICE_XB	0xB08	Current service register	0x0000_0000

Register	Offset	Description	Reset Value
EXT_INT_SERVICE_PEND_XB	0x0B0C	Current service pending register	0x0000_0000
EXT_INT_GRPFIXPRI_XB	0x0B10	External interrupt group fixed priority control register	0x0000_0000
EXT_INT21_FIXPRI	0x0B14	External interrupt 1 fixed priority control register	0x0000_0000
EXT_INT22_FIXPRI	0x0B18	External interrupt 2 fixed priority control register	0x0000_0000
EXT_INT23_FIXPRI	0x0B1C	External interrupt 3 fixed priority control register	0x0000_0000
EXT_INT24_FIXPRI	0x0B20	External interrupt 4 fixed priority control register	0x0000_0000
EXT_INT25_FIXPRI	0x0B24	External interrupt 5 fixed priority control register	0x0000_0000
EXT_INT26_FIXPRI	0x0B28	External interrupt 6 fixed priority control register	0x0000_0000
EXT_INT27_FIXPRI	0x0B2C	External interrupt 7 fixed priority control register	0x0000_0000
EXT_INT28_FIXPRI	0x0B30	External interrupt 8 fixed priority control register	0x0000_0000
EXT_INT29_FIXPRI	0x0B34	External interrupt 9 fixed priority control register	0x0000_0000
GPX0CON	0x0C00	Port group GPX0 configuration register	0x0000_0000
GPX0DAT	0x0C04	Port group GPX0 data register	0x00
GPX0PUD	0x0C08	Port group GPX0 pull-up/down register	0x5555
GPX0DRV	0x0C0C	Port group GPX0 drive strength control register	0x00_0000
GPX1CON	0x0C20	Port group GPX1 configuration register	0x0000_0000
GPX1DAT	0x0C24	Port group GPX1 data register	0x00
GPX1PUD	0x0C28	Port group GPX1 pull-up/down register	0x5555
GPX1DRV	0x0C2C	Port group GPX1 drive strength control register	0x00_0000
GPX2CON	0x0C40	Port group GPX2 configuration register	0x0000_0000
GPX2DAT	0x0C44	Port group GPX2 data register	0x00
GPX2PUD	0x0C48	Port group GPX2 pull-up/down register	0x5555
GPX2DRV	0x0C4C	Port group GPX2 drive strength control register	0x00_0000
GPX3CON	0x0C60	Port group GPX3 configuration register	0x0000_0000
GPX3DAT	0x0C64	Port group GPX3 data register	0x00
GPX3PUD	0x0C68	Port group GPX3 pull-up/down register	0x5555
GPX3DRV	0x0C6C	Port group GPX3 drive strength control register	0x00_0000
WAKEUP_INT0_CON	0x0E00	External interrupt WAKEUP_INT0 configuration register	0x0000_0000
WAKEUP_INT1_CON	0x0E04	External interrupt WAKEUP_INT1 configuration register	0x0000_0000
WAKEUP_INT2_CON	0x0E08	External interrupt WAKEUP_INT2 configuration register	0x0000_0000
WAKEUP_INT3_CON	0x0E0C	External interrupt WAKEUP_INT3 configuration register	0x0000_0000
WAKEUP_INT0_FLTCON0	0x0E80	External interrupt WAKEUP_INT0 filter configuration register 0	0x8080_8080

Register	Offset	Description	Reset Value
WAKEUP_INT0_FLTCON1	0x0E84	External interrupt WAKEUP_INT0 filter configuration register 1	0x8080_8080
WAKEUP_INT1_FLTCON0	0x0E88	External interrupt WAKEUP_INT1 filter configuration register 0	0x8080_8080
WAKEUP_INT1_FLTCON1	0x0E8C	External interrupt WAKEUP_INT1 filter configuration register 1	0x8080_8080
WAKEUP_INT2_FLTCON0	0x0E90	External interrupt WAKEUP_INT2 filter configuration register 0	0x8080_8080
WAKEUP_INT2_FLTCON1	0x0E94	External interrupt WAKEUP_INT2 filter configuration register 1	0x8080_8080
WAKEUP_INT3_FLTCON0	0x0E98	External interrupt WAKEUP_INT3 filter configuration register 0	0x8080_8080
WAKEUP_INT3_FLTCON1	0x0E9C	External interrupt WAKEUP_INT3 filter configuration register 1	0x8080_8080
WAKEUP_INT0_MASK	0x0F00	External interrupt WAKEUP_INT0 mask register	0x0000_00FF
WAKEUP_INT1_MASK	0x0F04	External interrupt WAKEUP_INT1 mask register	0x0000_00FF
WAKEUP_INT2_MASK	0x0F08	External interrupt WAKEUP_INT2 mask register	0x0000_00FF
WAKEUP_INT3_MASK	0x0F0C	External interrupt WAKEUP_INT3 mask register	0x0000_00FF
WAKEUP_INT0_PEND	0x0F40	External interrupt WAKEUP_INT0 pending register	0x0000_0000
WAKEUP_INT1_PEND	0x0F44	External interrupt WAKEUP_INT1 pending register	0x0000_0000
WAKEUP_INT2_PEND	0x0F48	External interrupt WAKEUP_INT2 pending register	0x0000_0000
WAKEUP_INT3_PEND	0x0F4C	External interrupt WAKEUP_INT3 pending register	0x0000_0000

- Base Address: 0x0386_0000

Register	Offset	Description	Reset Value
Part 3			
GPZCON	0x0000	Port group GPZ configuration register	0x0000_0000
GPZDAT	0x0004	Port group GPZ data register	0x00
GPZPUD	0x0008	Port group GPZ pull-up/down register	0x1555
GPZDRV	0x000C	Port group GPZ drive strength control register	0x00_0000
GPZCONPDN	0x0010	Port group GPZ power down mode configuration register	0x0000
GPZPUDPDN	0x0014	Port group GPZ power down mode pull-up/down register	0x0000

4.4.2 Part 1

For the following SFRs, setting the value does not take effect immediately. It takes at least 800 APB clocks for the value to take effect after the SFR is actually changed:

GPA0PUD, GPA0DRV, GPA1PUD, GPA1DRV, GPBPUD, GPBDRV, GPC0PUD, GPC0DRV, GPC1PUD, GPC1DRV, GPD0PUD, GPD0DRV, GPD1PUD, GPD1DRV, GPE0PUD, GPE0DRV, GPE1PUD, GPE1DRV, GPE2PUD, GPE2DRV, GPE3PUD, GPE3DRV, GPE4PUD, GPE4DRV, GPF0PUD, GPF0DRV, GPF1PUD, GPF1DRV, GPF2PUD, GPF2DRV, GPF3PUD, GPF3DRV.

4.4.2.1 GPA0CON

- Address = 0x1140_0000, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPA0CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = UART_0_RXD 0x3 – 0xE = Reserved 0xF = EXT_INT1[0]	0x00
GPA0CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = UART_0_TXD 0x3 – 0xE = Reserved 0xF = EXT_INT1[1]	0x00
GPA0CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = UART_0_CTSn 0x3 – 0xE = Reserved 0xF = EXT_INT1[2]	0x00
GPA0CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = UART_0_RTSn 0x3 – 0xE = Reserved 0xF = EXT_INT1[3]	0x00
GPA0CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = UART_1_RXD 0x3 – 0xE = Reserved 0xF = EXT_INT1[4]	0x00
GPA0CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = UART_1_TXD 0x3 – 0xE = Reserved 0xF = EXT_INT1[5]	0x00
GPA0CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = UART_1_CTSn 0x3 = I2C_2_SDA 0x4 – 0xE = Reserved 0xF = EXT_INT1[6]	0x00
GPA0CON[7]	[31:28]	RW	0x0 = Input 0x1 = Output 0x2 = UART_1_RTSn 0x3 = I2C_2_SCL 0x4 – 0xE = Reserved 0xF = EXT_INT1[7]	0x00

4.4.2.2 GPA0DAT

- Address = 0x1140_0004, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPA0DAT[7:0]	[7:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.2.3 GPA0PUD

- Address = 0x1140_0008, Reset Value = 0x5555

Name	Bit	Type	Description	Reset Value
GPA0PUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/ down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x5555

4.4.2.4 GPA0DRV

- Address = 0x1140_000C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
GPA0DRV[n]	[2n+1:2n] n=0 – 7	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.2.5 GPA0CONPDN

- Address = 0x1140_0010, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPA0[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.2.6 GPA0PUPDN

- Address = 0x1140_0014, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPA0[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.2.7 GPA1CON

- Address = 0x1140_0020, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPA1CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = UART_2_RXD 0x3 = Reserved 0x4 = UART_AUDIO_RXD 0x5 – 0xE = Reserved 0xF = EXT_INT2[0]	0x00
GPA1CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = UART_2_TXD 0x3 = Reserved 0x4 = UART_AUDIO_TXD 0x5 – 0xE = Reserved 0xF = EXT_INT2[1]	0x00
GPA1CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = UART_2_CTSn 0x3 = I2C_3_SDA 0x4 – 0xE = Reserved 0xF = EXT_INT2[2]	0x00
GPA1CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = UART_2_RTSn 0x3 = I2C_3_SCL 0x4 – 0xE = Reserved 0xF = EXT_INT2[3]	0x00
GPA1CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = UART_3_RXD 0x3 = Reserved 0x4 = UART_AUDIO_RXD 0x5 – 0xE = Reserved, 0xF = EXT_INT2[4]	0x00
GPA1CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output	0x00

Name	Bit	Type	Description	Reset Value
			0x2 = UART_3_TXD 0x3 = Reserved 0x4 = UART_AUDIO_TXD 0x5 – 0xE = Reserved 0xF = EXT_INT2[5]	

4.4.2.8 GPA1CON

- Address = 0x1140_0024, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPA1DAT[5:0]	[5:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.2.9 GPA1PUD

- Address = 0x1140_0028, Reset Value = 0x0555

Name	Bit	Type	Description	Reset Value
GPA1PUD[n]	[2n+1:2n] n=0 – 5	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x0555

4.4.2.10 GPA1DRV

- Address = 0x1140_002C, Reset Value = 0x00_00000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
GPA1DRV[n]	[2n+1:2n] n=0 – 5	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.2.11 GPA1CONPDN

- Address = 0x1140_0030, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPA1[n]	[2n+1:2n] n=0 – 5	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.2.12 GPA1PUDPDN

- Address = 0x1140_0034, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPA1[n]	[2n+1:2n] n=0 – 5	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.2.13 GPBCON

- Address = 0x1140_0040, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPBCON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = SPI_0_CLK 0x3 – 0xE = Reserved 0xF = EXT_INT3[0]	0x00
GPBCON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = SPI_0_nSS 0x3 – 0xE = Reserved 0xF = EXT_INT3[1]	0x00
GPBCON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = SPI_0_MIS 0x3 = I2C_4_SDA 0x4 – 0xE = Reserved, 0xF = EXT_INT3[2]	0x00
GPBCON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = SPI_0_MOSI 0x3 = I2C_4_SCL 0x4 – 0xE = Reserved 0xF = EXT_INT3[3]	0x00
GPBCON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = SPI_1_CLK 0x3= Reserved	0x00

Name	Bit	Type	Description	Reset Value
			0x4 = IEM_SCLK 0x5 – 0xE = Reserved 0xF = EXT_INT3[4]	
GPBCON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = SPI_1_nSS 0x3= Reserved 0x4 = IEM_SPWI 0x5 – 0xE = Reserved 0xF = EXT_INT3[5]	0x00
GPBCON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = SPI_1_MISO 0x3 = I2C_5_SDA 0x4 – 0xE = Reserved 0xF = EXT_INT3[6]	0x00
GPBCON[7]	[31:28]	RW	0x0 = Input 0x1 = Output 0x2 = SPI_1_MOSI 0x3 = I2C_5_SCL 0x4 ~ 0xE = Reserved 0xF = EXT_INT3[7]	0x00

4.4.2.14 GPBDAT

- Address = 0x1140_0044, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPBDAT[7:0]	[7:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.2.15 GPBPUD

- Address = 0x1140_0048, Reset Value = 0x5555

Name	Bit	Type	Description	Reset Value
GPBPUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x5555

4.4.2.16 GPBDRV

- Address = 0x1140_004C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
GPBDRV[n]	[2n+1:2n] n=0 – 7	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.2.17 GPBCONPDN

- Address = 0x1140_0050, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPB[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.2.18 GPBPUPDN

- Address = 0x1140_0054, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPB[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.2.19 GPC0CON

- Address = 0x1140_0060, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPC0CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = I2S_1_SCLK 0x3 = PCM_1_SCLK 0x4 = AC97BITCLK 0x5 – 0xE = Reserved 0xF = EXT_INT4[0]	0x00
GPC0CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = I2S_1_CDCLK 0x3 = PCM_1_EXTCLK 0x4 = AC97RESETn 0x5 – 0xE = Reserved	0x00

Name	Bit	Type	Description	Reset Value
			0xF = EXT_INT4[1]	
GPC0CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = I2S_1_LRCK 0x3 = PCM_1_FSYNC 0x4 = AC97SYNC 0x5 – 0xE = Reserved 0xF = EXT_INT4[2]	0x00
GPC0CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = I2S_1_SDI 0x3 = PCM_1_SIN 0x4 = AC97SDI 0x5 – 0xE = Reserved 0xF = EXT_INT4[3]	0x00
GPC0CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = I2S_1_SDO 0x3 = PCM_1_SOUT 0x4 = AC97SDO 0x5 – 0xE = Reserved 0xF = EXT_INT4[4]	0x00

4.4.2.20 GPC0DAT

- Address = 0x1140_0064, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPC0DAT[4:0]	[4:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.2.21 GPC0PUD

- Address = 0x1140_0068, Reset Value = 0x0155

Name	Bit	Type	Description	Reset Value
GPC0PUD[n]	[2n+1:2n] n=0 – 4	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x0155

4.4.2.22 GPC0DRV

- Address = 0x1140_006C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
GPC0DRV[n] n=0 – 4	[2n+1:2n]	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.2.23 GPC0CONPDN

- Address = 0x1140_0070, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPC0[n] n=0 – 4	[2n+1:2n]	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.2.24 GPC0PUPD�N

- Address = 0x1140_0074, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPC0[n] n=0 – 4	[2n+1:2n]	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.2.25 GPC1CON

- Address = 0x1140_0080, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPC1CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = I2S_2_SCLK 0x3 = PCM_2_SCLK 0x4 = SPDIF_0_OUT 0x5 – 0xE = Reserved 0xF = EXT_INT5[0]	0x00
GPC1CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = I2S_2_CDCLK 0x3 = PCM_2_EXTCLK 0x4 = SPDIF_EXTCLK 0x5 = SPI_2_CLK 0x6 – 0xE = Reserved 0xF = EXT_INT5[1]	0x00
GPC1CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = I2S_2_LRCK 0x3 = PCM_2_FSYNC 0x4= Reserved 0x5 = SPI_2_nSS 0x6 – 0xE = Reserved 0xF = EXT_INT5[2]	0x00
GPC1CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = I2S_2_SDI 0x3 = PCM_2_SIN 0x4 = I2C_6_SDA 0x5 = SPI_2_MISO 0x6 – 0xE = Reserved 0xF = EXT_INT5[3]	0x00
GPC1CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = I2S_2_SDO 0x3 = PCM_2_SOUT 0x4 = I2C_6_SCL 0x5 = SPI_2_MOSI 0x6 – 0xE = Reserved 0xF = EXT_INT5[4]	0x00

4.4.2.26 GPC1DAT

- Address = 0x1140_0084, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPC1DAT[4:0]	[4:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.2.27 GPC1PUD

- Address = 0x1140_0088, Reset Value = 0x0155

Name	Bit	Type	Description	Reset Value
GPC1PUD[n]	[2n+1:2n] n=0 – 4	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x0155

4.4.2.28 GPC1DRV

- Address = 0x1140_008C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
GPC1DRV[n]	[2n+1:2n] n=0 – 4	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.2.29 GPC1CONPDN

- Address = 0x1140_0090, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPC1[n]	[2n+1:2n] n=0 – 4	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.2.30 GPC1PUDPDN

- Address = 0x1140_0094, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPC1[n]	[2n+1:2n] n=0 – 4	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.2.31 GPD0CON

- Address = 0x1140_00A0, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPD0CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = TOUT_0 0x3 = LCD_FRM 0x4 – 0xE = Reserved 0xF = EXT_INT6[0]	0x00
GPD0CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = TOUT_1 0x3 = LCD_PWM 0x4 – 0xE = Reserved 0xF = EXT_INT6[1]	0x00
GPD0CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = TOUT_2 0x3 = I2C_7_SDA 0x4 – 0xE = Reserved 0xF = EXT_INT6[2]	0x00
GPD0CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = TOUT_3 0x3 = I2C_7_SCL 0x4 – 0xE = Reserved 0xF = EXT_INT6[3]	0x00

4.4.2.32 GPD0DAT

- Address = 0x1140_00A0, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPD0DAT[3:0]	[3:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.2.33 GPD0PUD

- Address = 0x1140_00A8, Reset Value = 0x0055

Name	Bit	Type	Description	Reset Value
GPD0PUD[n]	[2n+1:2n] n=0 – 3	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x0055

4.4.2.34 GPD0DRV

- Address = 0x1140_00AC, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
GPD0DRV[n]	[2n+1:2n] n=0 – 3	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.2.35 GPD0CONPDN

- Address = 0x1140_00B0, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPD0[n]	[2n+1:2n] n=0 – 3	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.2.36 GPD0PUPDN

- Address = 0x1140_00B4, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPD0[n]	[2n+1:2n] n=0 – 3	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.2.37 GPD1CON

- Address = 0x1140_00C0, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPD1CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = I2C_0_SDA 0x3 = MIPI_BYT_CLK 0x4 – 0xE = Reserved 0xF = EXT_INT7[0]	0x00
GPD1CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = I2C_0_SCL 0x3 = MIPI_ESC_CLK 0x4 – 0xE = Reserved 0xF = EXT_INT7[1]	0x00
GPD1CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = I2C_1_SDA 0x3 = MIPI_2L_BYT_CLK 0x4 – 0xE = Reserved 0xF = EXT_INT7[2]	0x00
GPD1CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = I2C_1_SCL 0x3 = MIPI_2L_ESC_CLK 0x4 – 0xE = Reserved 0xF = EXT_INT7[3]	0x00

4.4.2.38 GPD1DAT

- Address = 0x1140_00C4, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPD1DAT[3:0]	[3:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.2.39 GPD1PUD

- Address = 0x1140_00C8, Reset Value = 0x0055

Name	Bit	Type	Description	Reset Value
GPD1PUD[n]	[2n+1:2n] n=0 – 3	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x0055

4.4.2.40 GPD1DRV

- Address = 0x1140_00CC, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
GPD1DRV[n]	[2n+1:2n] n=0 – 3	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.2.41 GPD1CONPDN

- Address = 0x1140_00D0, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPD1[n]	[2n+1:2n] n=0 – 3	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.2.42 GPD1PUDPDN

- Address = 0x1140_00D4, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPD1[n]	[2n+1:2n] n=0 – 3	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.2.43 GPE0CON

- Address = 0x1140_00E0, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPE0CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_WEn 0x3 = CAM_B_PCLK 0x4 = TS_CLK 0x5 = LCD_B_VCLK 0x6 = TraceClk 0x7 – 0xE = Reserved 0xF = EXT_INT8[0]	0x00
GPE0CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_CSn 0x3 = CAM_B_VSYNC 0x4 = TS_SYNC 0x5 = LCD_B_FRM 0x6 – 0xE = Reserved 0xF = EXT_INT8[1]	0x00
GPE0CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_Rn 0x3 = CAM_B_HREF 0x4 = TS_VAL 0x5 = LCD_B_HSYNC 0x6 = TraceCtl 0x7 – 0xE = Reserved 0xF = EXT_INT8[2]	0x00
GPE0CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_IRQn 0x3 = CAM_B_FIELD 0x4 = TS_DATA 0x5 = LCD_B_VSYNC 0x6 = TraceData[0] 0x7 – 0xE = Reserved 0xF = EXT_INT8[3]	0x00

Name	Bit	Type	Description	Reset Value
GPE0CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_ADVN 0x3 = CAM_B_CLKOUT 0x4 = TS_ERROR 0x5 = LCD_B_VDEN 0x6 = TraceData[1] 0x7 – 0xE = Reserved 0xF = EXT_INT8[4]	0x00

4.4.2.44 GPE0DAT

- Address = 0x1140_00E4, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPE0DAT[4:0]	[4:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.2.45 GPE0PUD

- Address = 0x1140_00E8, Reset Value = 0x0155

Name	Bit	Type	Description	Reset Value
GPE0PUD[n]	[2n+1:2n] n=0 – 4	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x0155

4.4.2.46 GPE0DRV

- Address = 0x1140_00EC, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
GPE0DRV[n]	[2n+1:2n] n=0 – 4	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.2.47 GPE0CONPDN

- Address = 0x1140_00F0, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPE0[n]	[2n+1:2n] n=0 – 4	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.2.48 GPE0PUDPDN

- Address = 0x1140_00F4, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPE0[n]	[2n+1:2n] n=0 – 4	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.2.49 GPE1CON

- Address = 0x1140_0100, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPE1CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_ADDR[0] 0x3 = CAM_B_DATA[0] 0x4 = Reserved 0x5 = LCD_B_VD[0] 0x6 = TraceData[2] 0x7 – 0xE = Reserved 0xF = EXT_INT9[0]	0x00
GPE1CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_ADDR[1] 0x3 = CAM_B_DATA[1] 0x4 = Reserved 0x5 = LCD_B_VD[1] 0x6 = TraceData[3] 0x7 – 0xE = Reserved 0xF = EXT_INT9[1]	0x00
GPE1CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output, 0x2 = MDM_ADDR[2], 0x3 = CAM_B_DATA[2], 0x4 = Reserved, 0x5 = LCD_B_VD[2], 0x6 = TraceData[4],	0x00

Name	Bit	Type	Description	Reset Value
			0x7 – 0xE = Reserved, 0xF = EXT_INT9[2]	
GPE1CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_ADDR[3] 0x3 = CAM_B_DATA[3] 0x4 = Reserved 0x5 = LCD_B_VD[3] 0x6 = TraceData[5] 0x7 – 0xE = Reserved 0xF = EXT_INT9[3]	0x00
GPE1CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_ADDR[4] 0x3 = CAM_B_DATA[4] 0x4 = Reserved 0x5 = LCD_B_VD[4] 0x6 = TraceData[6] 0x7 – 0xE = Reserved 0xF = EXT_INT9[4]	0x00
GPE1CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_ADDR[5] 0x3 = CAM_B_DATA[5] 0x4 = Reserved 0x5 = LCD_B_VD[5] 0x6 = TraceData[7] 0x7 – 0xE = Reserved 0xF = EXT_INT9[5]	0x00
GPE1CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_ADDR[6] 0x3 = CAM_B_DATA[6] 0x4 = Reserved 0x5 = LCD_B_VD[6] 0x6 = TraceData[8] 0x7 – 0xE = Reserved 0xF = EXT_INT9[6]	0x00
GPE1CON[7]	[31:28]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_ADDR[7] 0x3 = CAM_B_DATA[7] 0x4 = Reserved 0x5 = LCD_B_VD[7] 0x6 = TraceData[9] 0x7 – 0xE = Reserved 0xF = EXT_INT9[7]	0x00

4.4.2.50 GPE1DAT

- Address = 0x1140_0104, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPE1DAT[7:0]	[7:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.2.51 GPE1PUD

- Address = 0x1140_0108, Reset Value = 0x5555

Name	Bit	Type	Description	Reset Value
GPE1PUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x5555

4.4.2.52 GPE1DRV

- Address = 0x1140_010C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
GPE1DRV[n]	[2n+1:2n] n=0 – 7	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.2.53 GPE1CONPDN

- Address = 0x1140_0110, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPE1[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.2.54 GPE1PUDPDN

- Address = 0x1140_0114, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPE1[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.2.55 GPE2CON

- Address = 0x1140_0120, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPE2CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_ADDR[8] 0x3 = Reserved 0x4 = Reserved 0x5 = LCD_B_VD[8] 0x6 = TraceData[10] 0x7 – 0xE = Reserved 0xF = EXT_INT10[0]	0x00
GPE2CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_ADDR[9] 0x3 = Reserved 0x4 = Reserved 0x5 = LCD_B_VD[9] 0x6 = TraceData[11] 0x7 – 0xE = Reserved 0xF = EXT_INT10[1]	0x00
GPE2CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_ADDR[10] 0x3 = Reserved 0x4 = Reserved 0x5 = LCD_B_VD[10] 0x6 = TraceData[12] 0x7 – 0xE = Reserved 0xF = EXT_INT10[2]	0x00
GPE2CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_ADDR[11] 0x3 = Reserved 0x4 = Reserved 0x5 = LCD_B_VD[11] 0x6 = TraceData[13] 0x7 – 0xE = Reserved	0x00

Name	Bit	Type	Description	Reset Value
			0xF = EXT_INT10[3]	
GPE2CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_ADDR[12] 0x3 = Reserved 0x4 = Reserved 0x5 = LCD_B_VD[12] 0x6 = TraceData[14] 0x7 – 0xE = Reserved 0xF = EXT_INT10[4]	0x00
GPE2CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_ADDR[13] 0x3 = Reserved 0x4 = Reserved 0x5 = LCD_B_VD[13] 0x6 = TraceData[15] 0x7 – 0xE = Reserved 0xF = EXT_INT10[5]	0x00

4.4.2.56 GPE2DAT

- Address = 0x1140_0124, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPE2DAT[5:0]	[5:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.2.57 GPE2PUD

- Address = 0x1140_0128, Reset Value = 0x0555

Name	Bit	Type	Description	Reset Value
GPE2PUD[n]	[2n+1:2n] n=0 – 5	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x0555

4.4.2.58 GPE2DRV

- Address = 0x1140_012C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
GPE2DRV[n]	[2n+1:2n] n=0 – 5	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.2.59 GPE2CONPDN

- Address = 0x1140_0130, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPE2[n]	[2n+1:2n] n=0 – 5	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.2.60 GPE2PUPD�N

- Address = 0x1140_0134, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPE2[n]	[2n+1:2n] n=0 – 5	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.2.61 GPE3CON

- Address = 0x1140_0140, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPE3CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_DATA[0] 0x3 = Reserved 0x4 = Reserved 0x5 = LCD_B_VD[14] 0x6 = TraceData[16] 0x7 – 0xE = Reserved 0xF = EXT_INT11[0]	0x00
GPE3CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_DATA[1] 0x3 = Reserved	0x00

Name	Bit	Type	Description	Reset Value
			0x4 = Reserved 0x5 = LCD_B_VD[15] 0x6 = TraceData[17] 0x7 – 0xE = Reserved 0xF = EXT_INT11[1]	
GPE3CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_DATA[2] 0x3 = Reserved 0x4 = Reserved 0x5 = LCD_B_VD[16] 0x6 = TraceData[18] 0x7 – 0xE = Reserved 0xF = EXT_INT11[2]	0x00
GPE3CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_DATA[3] 0x3 = Reserved 0x4 = Reserved 0x5 = LCD_B_VD[17] 0x6 = TraceData[19] 0x7 – 0xE = Reserved 0xF = EXT_INT11[3]	0x00
GPE3CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_DATA[4] 0x3 = Reserved 0x4 = Reserved 0x5 = LCD_B_VD[18] 0x6 = TraceData[20] 0x7 – 0xE = Reserved 0xF = EXT_INT11[4]	0x00
GPE3CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_DATA[5] 0x3 = Reserved 0x4 = Reserved 0x5 = LCD_B_VD[19] 0x6 = TraceData[21] 0x7 – 0xE = Reserved 0xF = EXT_INT11[5]	0x00
GPE3CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_DATA[6] 0x3 = Reserved 0x4 = Reserved 0x5 = LCD_B_VD[20] 0x6 = TraceData[22] 0x7 – 0xE = Reserved 0xF = EXT_INT11[6]	0x00

Name	Bit	Type	Description	Reset Value
GPE3CON[7]	[31:28]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_DATA[7] 0x3 = Reserved 0x4 = Reserved 0x5 = LCD_B_VD[21] 0x6 = TraceData[23] 0x7 – 0xE = Reserved 0xF = EXT_INT11[7]	0x00

4.4.2.62 GPE3DAT

- Address = 0x1140_0144, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPE3DAT[7:0]	[7:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.2.63 GPE3PUD

- Address = 0x1140_0148, Reset Value = 0x5555

Name	Bit	Type	Description	Reset Value
GPE3PUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x5555

4.4.2.64 GPE3DRV

- Address = 0x1140_014C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
GPE3DRV[n]	[2n+1:2n] n=0 – 7	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.2.65 GPE3CONPDN

- Address = 0x1140_0150, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPE3[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.2.66 GPE3PUDPDN

- Address = 0x1140_0154, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPE3[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.2.67 GPE4CON

- Address = 0x1140_0160, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPE4CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_DATA[8] 0x3 = Reserved 0x4 = Reserved 0x5 = LCD_B_VD[22] 0x6 = TraceData[24] 0x7 – 0xE = Reserved 0xF = EXT_INT12[0]	0x00
GPE4CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_DATA[9] 0x3 = Reserved 0x4 = Reserved 0x5 = LCD_B_VD[23] 0x6 = TraceData[25] 0x7 – 0xE = Reserved 0xF = EXT_INT12[1]	0x00
GPE4CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_DATA[10] 0x3 = Reserved 0x4 = Reserved 0x5 = VSYNC_LDI_B 0x6 = TraceData[26]	0x00

Name	Bit	Type	Description	Reset Value
			0x7 – 0xE = Reserved 0xF = EXT_INT12[2]	
GPE4CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_DATA[11] 0x3 = Reserved 0x4 = Reserved 0x5 = SYS_OE_B 0x6 = TraceData[27] 0x7 – 0xE = Reserved 0xF = EXT_INT12[3]	0x00
GPE4CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_DATA[12] 0x3 = Reserved 0x4 = Reserved 0x5 = LCD_B_PWM 0x6 = TraceData[28] 0x7 – 0xE = Reserved 0xF = EXT_INT12[4]	0x00
GPE4CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_DATA[13] 0x3 = Reserved 0x4 = Reserved 0x5 = Reserved 0x6 = TraceData[29] 0x7 – 0xE = Reserved 0xF = EXT_INT12[5]	0x00
GPE4CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_DATA[14] 0x3 = Reserved 0x4 = Reserved 0x5 = Reserved 0x6 = TraceData[30] 0x7 – 0xE = Reserved 0xF = EXT_INT12[6]	0x00
GPE4CON[7]	[31:28]	RW	0x0 = Input 0x1 = Output 0x2 = MDM_DATA[15] 0x3 = Reserved 0x4 = Reserved 0x5 = Reserved 0x6 = TraceData[31] 0x7 – 0xE = Reserved 0xF = EXT_INT12[7]	0x00

4.4.2.68 GPE4DAT

- Address = 0x1140_0164, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPE4DAT[7:0]	[7:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.2.69 GPE4PUD

- Address = 0x1140_0168, Reset Value = 0x5555

Name	Bit	Type	Description	Reset Value
GPE4PUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x5555

4.4.2.70 GPE4DRV

- Address = 0x1140_016C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
GPE4DRV[n]	[2n+1:2n] n=0 – 7	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.2.71 GPE4CONPDN

- Address = 0x1140_0170, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPE4[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.2.72 GPE4PUDPDN

- Address = 0x1140_0174, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPE4[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.2.73 GPF0CON

- Address = 0x1140_0180, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPF0CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_HSYNC 0x3 – 0xE = Reserved 0xF = EXT_INT13[0]	0x00
GPF0CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VSYNC 0x3 – 0xE = Reserved 0xF = EXT_INT13[1]	0x00
GPF0CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VDEN 0x3 – 0xE = Reserved 0xF = EXT_INT13[2]	0x00
GPF0CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VCLK 0x3 – 0xE = Reserved 0xF = EXT_INT13[3]	0x00
GPF0CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[0] 0x3 – 0xE = Reserved 0xF = EXT_INT13[4]	0x00
GPF0CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[1] 0x3 – 0xE = Reserved 0xF = EXT_INT13[5]	0x00
GPF0CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[2] 0x3 – 0xE = Reserved	0x00

Name	Bit	Type	Description	Reset Value
			0xF = EXT_INT13[6]	
GPF0CON[7]	[31:28]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[3] 0x3 – 0xE = Reserved 0xF = EXT_INT13[7]	0x00

4.4.2.74 GPF0DAT

- Address = 0x1140_0184, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPF0DAT[7:0]	[7:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.2.75 GPF0PUD

- Address = 0x1140_0188, Reset Value = 0x5555

Name	Bit	Type	Description	Reset Value
GPF0PUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x5555

4.4.2.76 GPF0DRV

- Address = 0x1140_018C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
GPF0DRV[n]	[2n+1:2n] n=0 – 7	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.2.77 GPF0CONPDN

- Address = 0x1140_0190, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPF0[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.2.78 GPF0PUDPDN

- Address = 0x1140_0194, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPF0[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.2.79 GPF1CON

- Address = 0x1140_01A0, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPF1CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[4] 0x3 – 0xE = Reserved 0xF = EXT_INT14[0]	0x00
GPF1CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[5] 0x3 – 0xE = Reserved 0xF = EXT_INT14[1]	0x00
GPF1CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[6] 0x3 – 0xE = Reserved 0xF = EXT_INT14[2]	0x00
GPF1CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[7] 0x3 – 0xE = Reserved 0xF = EXT_INT14[3]	0x00
GPF1CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[8] 0x3 – 0xE = Reserved	0x00

Name	Bit	Type	Description	Reset Value
			0xF = EXT_INT14[4]	
GPF1CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[9] 0x3 – 0xE = Reserved 0xF = EXT_INT14[5]	0x00
GPF1CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[10] 0x3 – 0xE = Reserved 0xF = EXT_INT14[6]	0x00
GPF1CON[7]	[31:28]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[11] 0x3 – 0xE = Reserved 0xF = EXT_INT14[7]	0x00

4.4.2.80 GPF1DAT

- Address = 0x1140_01A4, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPF1DAT[7:0]	[7:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.2.81 GPF1PUD

- Address = 0x1140_01A8, Reset Value = 0x5555

Name	Bit	Type	Description	Reset Value
GPF1PUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x5555

4.4.2.82 GPF1DRV

- Address = 0x1140_01AC, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
GPF1DRV[n] n=0 – 7	[2n+1:2n]	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.2.83 GPF1CONPDN

- Address = 0x1140_01B0, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPF1[n] n=0 – 7	[2n+1:2n]	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.2.84 GPF1PUPDN

- Address = 0x1140_01B4, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPF1[n] n=0 – 7	[2n+1:2n]	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.2.85 GPF2CON

- Address = 0x1140_01C0, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPF2CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[12] 0x3 – 0xE = Reserved 0xF = EXT_INT15[0]	0x00
GPF2CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[13] 0x3 – 0xE = Reserved 0xF = EXT_INT15[1]	0x00
GPF2CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[14] 0x3 – 0xE = Reserved 0xF = EXT_INT15[2]	0x00
GPF2CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[15] 0x3 – 0xE = Reserved 0xF = EXT_INT15[3]	0x00
GPF2CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[16] 0x3 – 0xE = Reserved 0xF = EXT_INT15[4]	0x00
GPF2CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[17] 0x3 – 0xE = Reserved 0xF = EXT_INT15[5]	0x00
GPF2CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[18] 0x3 – 0xE = Reserved 0xF = EXT_INT15[6]	0x00
GPF2CON[7]	[31:28]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[19] 0x3 – 0xE = Reserved 0xF = EXT_INT15[7]	0x00

4.4.2.86 GPF2DAT

- Address = 0x1140_01C4, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPF2DAT[7:0]	[7:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.2.87 GPF2PUD

- Address = 0x1140_01C8, Reset Value = 0x5555

Name	Bit	Type	Description	Reset Value
GPF2PUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x5555

4.4.2.88 GPF2DRV

- Address = 0x1140_01CC, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
GPF2DRV[n]	[2n+1:2n] n=0 – 7	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.2.89 GPF2CONPDN

- Address = 0x1140_01D0, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPF2[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.2.90 GPF2PUDPDN

- Address = 0x1140_01D4, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPF2[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.2.91 GPF3CON

- Address = 0x1140_01E0, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPF3CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[20] 0x3 ~ 0xE = Reserved 0xF = EXT_INT16[0]	0x00
GPF3CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[21] 0x3 ~ 0xE = Reserved 0xF = EXT_INT16[1]	0x00
GPF3CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[22] 0x3 ~ 0xE = Reserved 0xF = EXT_INT16[2]	0x00
GPF3CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = LCD_VD[23] 0x3 ~ 0xE = Reserved 0xF = EXT_INT16[3]	0x00
GPF3CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = VSYNC_LDI 0x3 ~ 0xE = Reserved 0xF = EXT_INT16[4]	0x00
GPF3CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = SYS_OE 0x3 ~ 0xE = Reserved 0xF = EXT_INT16[5]	0x00

4.4.2.92 GPF3DAT

- Address = 0x1140_01E4, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPF3DAT[5:0]	[5:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.2.93 GPF3PUD

- Address = 0x1140_01E8, Reset Value = 0x0555

Name	Bit	Type	Description	Reset Value
GPF3PUD[n]	[2n+1:2n] n=0 – 5	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x0555

4.4.2.94 GPF3DRV

- Address = 0x1140_01EC, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
GPF3DRV[n]	[2n+1:2n] n=0 – 5	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.2.95 GPF3CONPDN

- Address = 0x1140_01F0, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPF3[n]	[2n+1:2n] n=0 – 5	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.2.96 GPF3PUDPDN

- Address = 0x1140_01F4, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPF3[n]	[2n+1:2n] n=0 – 5	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.2.97 ETC0PUD

- Address = 0x1140_0208, Reset Value = 0x0400

Name	Bit	Type	Description	Reset Value
ETC0PUD[n]	[2n+1:2n] n=0 – 5	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x0400

ETC0PUD[9:8] controls XjTDO. /ETC0PUD[11:10] controls XjDBGSEL.

4.4.2.98 ETC0DRV

- Address = 0x1140_020C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
ETC0DRV[n]	[2n+1:2n] n=0 – 5	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

ETC0DRV[1:0] controls XjTRSTn.

ETC0DRV[3:2] controls XjTMS.

ETC0DRV[5:4] controls XjTCK.

ETC0DRV[7:6] controls XjTDI.

ETC0DRV[9:8] controls XjTDO.

ETC0DRV[11:10] controls XjDBGSEL.

4.4.2.99 ETC1PUD

- Address = 0x1140_0228, Reset Value = 0x0005

Name	Bit	Type	Description	Reset Value
ETC1PUD[n]	[2n+1:2n] n=0 – 1	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x0005

ETC1PUD[1:0] controls XsbusDATA.

ETC1PUD[3:2] controls XsbusCLK.

4.4.2.100 ETC1DRV

- Address = 0x1140_022C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
ETC1DRV[n]	[2n+1:2n] n=0 – 1	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

ETC1DRV[1:0] controls XsbusDATA.

ETC1DRV[3:2] controls XsbusCLK.

4.4.2.101 EXT_INT1CON

- Address = 0x1140_0700, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31]	-	Reserved	0x0
EXT_INT1_CON[7]	[30:28]	RW	Setting the signaling method of EXT_INT1[7] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[27]	-	Reserved	0x0
EXT_INT1_CON[6]	[26:24]	RW	Setting the signaling method of EXT_INT1[6] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered	0x0

Name	Bit	Type	Description	Reset Value
			0x5 – 0x7 = Reserved	
RSVD	[23]	-	Reserved	0x0
EXT_INT1_CON[5]	[22:20]	RW	Setting the signaling method of EXT_INT1[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
EXT_INT1_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT1[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT1_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT1[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT1_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT1[2] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	-	Reserved	0x0
EXT_INT1_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT1[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT1_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT1[0] 0x0 = Low level 0x1 = High level	0x0

Name	Bit	Type	Description	Reset Value
			0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	

4.4.2.102 EXT_INT2CON

- Address = 0x1140_0704, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	–	Reserved	0x00
RSVD	[23]	–	Reserved	0x0
EXT_INT2_CON[5]	[22:20]	RW	Setting the signaling method of EXT_INT2[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	–	Reserved	0x0
EXT_INT2_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT2[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	–	Reserved	0x0
EXT_INT2_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT2[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	–	Reserved	0x0
EXT_INT2_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT2[2] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	–	Reserved	0x0
EXT_INT2_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT2[1] 0x0 = Low level	0x0

Name	Bit	Type	Description	Reset Value
			0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	
RSVD	[3]	–	Reserved	0x0
EXT_INT2_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT2[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.2.103 EXT_INT3CON

- Address = 0x1140_0708, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31]	-	Reserved	0x0
EXT_INT3_CON[7]	[30:28]	RW	Setting the signaling method of EXT_INT3[7] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[27]	-	Reserved	0x0
EXT_INT3_CON[6]	[26:24]	RW	Setting the signaling method of EXT_INT3[6] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[23]	-	Reserved	0x0
EXT_INT3_CON[5]	[22:20]	RW	Setting the signaling method of EXT_INT3[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
EXT_INT3_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT3[4] 0x0 = Low level, 0x1 = High level, 0x2 = Falling edge triggered, 0x3 = Rising edge triggered, 0x4 = Both edge triggered, 0x5 ~ 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT3_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT3[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT3_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT3[2]	0x0

Name	Bit	Type	Description	Reset Value
			0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	
RSVD	[7]	-	Reserved	0x0
EXT_INT3_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT3[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT3_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT3[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.2.104 EXT_INT4CON

- Address = 0x1140_070C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:20]	-	Reserved	0x000
RSVD	[19]	-	Reserved	0x0
EXT_INT4_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT4[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT4_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT4[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT4_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT4[2] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	-	Reserved	0x0
EXT_INT4_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT4[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT4_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT4[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.2.105 EXT_INT5CON

- Address = 0x1140_0710, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:20]	-	Reserved	0x000
RSVD	[19]	-	Reserved	0x0
EXT_INT5_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT5[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT5_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT5[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT5_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT5[2] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	-	Reserved	0x0
EXT_INT5_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT5[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT5_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT5[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.2.106 EXT_INT6CON

- Address = 0x1140_0714, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	-	Reserved	0x0000
RSVD	[15]	-	Reserved	0x0
EXT_INT6_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT6[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT6_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT6[2] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	-	Reserved	0x0
EXT_INT6_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT6[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT6_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT6[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.2.107 EXT_INT7CON

- Address = 0x1140_0718, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	–	Reserved	0x0000
RSVD	[15]	–	Reserved	0x0
EXT_INT7_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT7[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	–	Reserved	0x0
EXT_INT7_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT7[2] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	–	Reserved	0x0
EXT_INT7_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT7[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	–	Reserved	0x0
EXT_INT7_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT7[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.2.108 EXT_INT8CON

- Address = 0x1140_071C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:20]	-	Reserved	0x000
RSVD	[19]	-	Reserved	0x0
EXT_INT8_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT8[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT8_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT8[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT8_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT8[2] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	-	Reserved	0x0
EXT_INT8_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT8[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT8_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT8[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.2.109 EXT_INT9CON

- Address = 0x1140_0720, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31]	-	Reserved	0x0
EXT_INT9_CON[7]	[30:28]	RW	Setting the signaling method of EXT_INT9[7] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[27]	-	Reserved	0x0
EXT_INT9_CON[6]	[26:24]	RW	Setting the signaling method of EXT_INT9[6] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[23]	-	Reserved	0x0
EXT_INT9_CON[5]	[22:20]	RW	Setting the signaling method of EXT_INT9[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
EXT_INT9_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT9[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT9_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT9[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT9_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT9[2]	0x0

Name	Bit	Type	Description	Reset Value
			0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	
RSVD	[7]	-	Reserved	0x0
EXT_INT9_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT9[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT9_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT9[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.2.110 EXT_INT10CON

- Address = 0x1140_0724, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	-	Reserved	0x00
RSVD	[23]	-	Reserved	0x0
EXT_INT10_CON[5]	[22:20]	RW	Setting the signaling method of EXT_INT10[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
EXT_INT10_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT10[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT10_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT10[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT10_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT10[2] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	-	Reserved	0x0
EXT_INT10_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT10[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0

Name	Bit	Type	Description	Reset Value
EXT_INT10_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT10[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.2.111 EXT_INT11CON

- Address = 0x1140_0728, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31]	-	Reserved	0x0
EXT_INT11_CON[7]	[30:28]	RW	Setting the signaling method of EXT_INT11[7] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[27]	-	Reserved	0x0
EXT_INT11_CON[6]	[26:24]	RW	Setting the signaling method of EXT_INT11[6] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[23]	-	Reserved	0x0
EXT_INT11_CON[5]	[22:20]	RW	Setting the signaling method of EXT_INT11[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
EXT_INT11_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT11[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT11_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT11[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT11_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT11[2]	0x0

Name	Bit	Type	Description	Reset Value
			0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	
RSVD	[7]	-	Reserved	0x0
EXT_INT11_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT11[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT11_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT11[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.2.112 EXT_INT12CON

- Address = 0x1140_072C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31]	-	Reserved	0x0
EXT_INT12_CON[7]	[30:28]	RW	Setting the signaling method of EXT_INT12[7] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[27]	-	Reserved	0x0
EXT_INT12_CON[6]	[26:24]	RW	Setting the signaling method of EXT_INT12[6] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[23]	-	Reserved	0x0
EXT_INT12_CON[5]	[22:20]	RW	Setting the signaling method of EXT_INT12[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
EXT_INT12_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT12[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT12_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT12[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT12_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT12[2] 0x0 = Low level	0x0

Name	Bit	Type	Description	Reset Value
			0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	
RSVD	[7]	-	Reserved	0x0
EXT_INT12_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT12[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT12_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT12[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.2.113 EXT_INT13CON

- Address = 0x1140_0730, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31]	-	Reserved	0x0
EXT_INT13_CON[7]	[30:28]	RW	Setting the signaling method of EXT_INT13[7] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[27]	-	Reserved	0x0
EXT_INT13_CON[6]	[26:24]	RW	Setting the signaling method of EXT_INT13[6] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[23]	-	Reserved	0x0
EXT_INT13_CON[5]	[22:20]	RW	Setting the signaling method of EXT_INT13[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
EXT_INT13_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT13[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT13_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT13[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT13_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT13[2]	0x0

Name	Bit	Type	Description	Reset Value
			0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	
RSVD	[7]	-	Reserved	0x0
EXT_INT13_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT13[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT13_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT13[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.2.114 EXT_INT14CON

- Address = 0x1140_0734, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31]	-	Reserved	0x0
EXT_INT14_CON[7]	[30:28]	RW	Setting the signaling method of EXT_INT14[7] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[27]	-	Reserved	0x0
EXT_INT14_CON[6]	[26:24]	RW	Setting the signaling method of EXT_INT14[6] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[23]	-	Reserved	0x0
EXT_INT14_CON[5]	[22:20]	RW	Setting the signaling method of EXT_INT14[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
EXT_INT14_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT14[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT14_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT14[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT14_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT14[2]	0x0

Name	Bit	Type	Description	Reset Value
			0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	
RSVD	[7]	-	Reserved	0x0
EXT_INT14_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT14[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT14_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT14[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.2.115 EXT_INT15CON

- Address = 0x1140_0738, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31]	-	Reserved	0x0
EXT_INT15_CON[7]	[30:28]	RW	Setting the signaling method of EXT_INT15[7] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[27]	-	Reserved	0x0
EXT_INT15_CON[6]	[26:24]	RW	Setting the signaling method of EXT_INT15[6] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[23]	-	Reserved	0x0
EXT_INT15_CON[5]	[22:20]	RW	Setting the signaling method of EXT_INT15[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
EXT_INT15_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT15[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT15_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT15[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT15_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT15[2]	0x0

Name	Bit	Type	Description	Reset Value
			0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	
RSVD	[7]	-	Reserved	0x0
EXT_INT15_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT15[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT15_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT15[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.2.116 EXT_INT16CON

- Address = 0x1140_073C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	-	Reserved	0x00
RSVD	[23]	-	Reserved	0x0
EXT_INT16_CON[5]	[22:20]	RW	Setting the signaling method of EXT_INT16[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
EXT_INT16_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT16[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT16_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT16[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT16_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT16[2] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	-	Reserved	0x0
EXT_INT16_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT16[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0

Name	Bit	Type	Description	Reset Value
EXT_INT16_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT16[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 ~ 0x7 = Reserved	0x0

4.4.2.117 EXT_INT1_FLTCON0

- Address = 0x1140_0800, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN1[3]	[31]	RW	Filter Enable for EXT_INT1[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH1[3]	[30:24]	RW	Filtering width of EXT_INT1[3]	0x00
FLTEN1[2]	[23]	RW	Filter Enable for EXT_INT1[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH1[2]	[22:16]	RW	Filtering width of EXT_INT1[2]	0x00
FLTEN1[1]	[15]	RW	Filter Enable for EXT_INT1[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH1[1]	[14:8]	RW	Filtering width of EXT_INT1[1]	0x00
FLTEN1[0]	[7]	RW	Filter Enable for EXT_INT1[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH1[0]	[6:0]	RW	Filtering width of EXT_INT1[0]	0x00

4.4.2.118 EXT_INT1_FLTCON1

- Address = 0x1140_0804, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN1[7]	[31]	RW	Filter Enable for EXT_INT1[7] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH1[7]	[30:24]	RW	Filtering width of EXT_INT1[7]	0x00
FLTEN1[6]	[23]	RW	Filter Enable for EXT_INT1[6] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH1[6]	[22:16]	RW	Filtering width of EXT_INT1[6]	0x00
FLTEN1[5]	[15]	RW	Filter Enable for EXT_INT1[5] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH1[5]	[14:8]	RW	Filtering width of EXT_INT1[5]	0x00
FLTEN1[4]	[7]	RW	Filter Enable for EXT_INT1[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH1[4]	[6:0]	RW	Filtering width of EXT_INT1[4]	0x00

4.4.2.119 EXT_INT2_FLTCON0

- Address = 0x1140_0808, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN2[3]	[31]	RW	Filter Enable for EXT_INT2[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH2[3]	[30:24]	RW	Filtering width of EXT_INT2[3]	0x00
FLTEN2[2]	[23]	RW	Filter Enable for EXT_INT2[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH2[2]	[22:16]	RW	Filtering width of EXT_INT2[2]	0x00
FLTEN2[1]	[15]	RW	Filter Enable for EXT_INT2[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH2[1]	[14:8]	RW	Filtering width of EXT_INT2[1]	0x00
FLTEN2[0]	[7]	RW	Filter Enable for EXT_INT2[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH2[0]	[6:0]	RW	Filtering width of EXT_INT2[0]	0x00

4.4.2.120 EXT_INT2_FLTCON1

- Address = 0x1140_080C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	-	Reserved	0x0000
FLTEN2[5]	[15]	RW	Filter Enable for EXT_INT2[5] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH2[5]	[14:8]	RW	Filtering width of EXT_INT2[5]	0x00
FLTEN2[4]	[7]	RW	Filter Enable for EXT_INT2[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH2[4]	[6:0]	RW	Filtering width of EXT_INT2[4]	0x00

4.4.2.121 EXT_INT3_FLTCON0

- Address = 0x1140_0810, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN3[3]	[31]	RW	Filter Enable for EXT_INT3[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH3[3]	[30:24]	RW	Filtering width of EXT_INT3[3]	0x00
FLTEN3[2]	[23]	RW	Filter Enable for EXT_INT3[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH3[2]	[22:16]	RW	Filtering width of EXT_INT3[2]	0x00
FLTEN3[1]	[15]	RW	Filter Enable for EXT_INT3[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH3[1]	[14:8]	RW	Filtering width of EXT_INT3[1]	0x00
FLTEN3[0]	[7]	RW	Filter Enable for EXT_INT3[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH3[0]	[6:0]	RW	Filtering width of EXT_INT3[0]	0x00

4.4.2.122 EXT_INT3_FLTCON1

- Address = 0x1140_0814, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN3[7]	[31]	RW	Filter Enable for EXT_INT3[7] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH3[7]	[30:24]	RW	Filtering width of EXT_INT3[7]	0x00
FLTEN3[6]	[23]	RW	Filter Enable for EXT_INT3[6] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH3[6]	[22:16]	RW	Filtering width of EXT_INT3[6]	0x00
FLTEN3[5]	[15]	RW	Filter Enable for EXT_INT3[5] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH3[5]	[14:8]	RW	Filtering width of EXT_INT3[5]	0x00
FLTEN3[4]	[7]	RW	Filter Enable for EXT_INT3[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH3[4]	[6:0]	RW	Filtering width of EXT_INT3[4]	0x00

4.4.2.123 EXT_INT4_FLTCON0

- Address = 0x1140_0818, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN4[3]	[31]	RW	Filter Enable for EXT_INT4[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH4[3]	[30:24]	RW	Filtering width of EXT_INT4[3]	0x00
FLTEN4[2]	[23]	RW	Filter Enable for EXT_INT4[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH4[2]	[22:16]	RW	Filtering width of EXT_INT4[2]	0x00
FLTEN4[1]	[15]	RW	Filter Enable for EXT_INT4[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH4[1]	[14:8]	RW	Filtering width of EXT_INT4[1]	0x00
FLTEN4[0]	[7]	RW	Filter Enable for EXT_INT4[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH4[0]	[6:0]	RW	Filtering width of EXT_INT4[0]	0x00

4.4.2.124 EXT_INT4_FLTCON1

- Address = 0x1140_081C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	—	Reserved	0x000000
FLTEN4[4]	[7]	RW	Filter Enable for EXT_INT4[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH4[4]	[6:0]	RW	Filtering width of EXT_INT4[4]	0x00

4.4.2.125 EXT_INT5_FLTCON0

- Address = 0x1140_0820, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN5[3]	[31]	RW	Filter Enable for EXT_INT5[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH5[3]	[30:24]	RW	Filtering width of EXT_INT5[3]	0x00
FLTEN5[2]	[23]	RW	Filter Enable for EXT_INT5[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH5[2]	[22:16]	RW	Filtering width of EXT_INT5[2]	0x00
FLTEN5[1]	[15]	RW	Filter Enable for EXT_INT5[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH5[1]	[14:8]	RW	Filtering width of EXT_INT5[1]	0x00
FLTEN5[0]	[7]	RW	Filter Enable for EXT_INT5[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH5[0]	[6:0]	RW	Filtering width of EXT_INT5[0]	0x00

4.4.2.126 EXT_INT5_FLTCON1

- Address = 0x1140_0824, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	—	Reserved	0x000000
FLTEN5[4]	[7]	RW	Filter Enable for EXT_INT5[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH5[4]	[6:0]	RW	Filtering width of EXT_INT5[4]	0x00

4.4.2.127 EXT_INT6_FLTCON0

- Address = 0x1140_0828, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN6[3]	[31]	RW	Filter Enable for EXT_INT6[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH6[3]	[30:24]	RW	Filtering width of EXT_INT6[3]	0x00
FLTEN6[2]	[23]	RW	Filter Enable for EXT_INT6[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH6[2]	[22:16]	RW	Filtering width of EXT_INT6[2]	0x00
FLTEN6[1]	[15]	RW	Filter Enable for EXT_INT6[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH6[1]	[14:8]	RW	Filtering width of EXT_INT6[1]	0x00
FLTEN6[0]	[7]	RW	Filter Enable for EXT_INT6[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH6[0]	[6:0]	RW	Filtering width of EXT_INT6[0]	0x00

4.4.2.128 EXT_INT6_FLTCON1

- Address = 0x1140_082C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:0]	-	Reserved	0x00000000

4.4.2.129 EXT_INT7_FLTCON0

- Address = 0x1140_0830, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN7[3]	[31]	RW	Filter Enable for EXT_INT7[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH7[3]	[30:24]	RW	Filtering width of EXT_INT7[3]	0x00
FLTEN7[2]	[23]	RW	Filter Enable for EXT_INT7[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH7[2]	[22:16]	RW	Filtering width of EXT_INT7[2]	0x00
FLTEN7[1]	[15]	RW	Filter Enable for EXT_INT7[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH7[1]	[14:8]	RW	Filtering width of EXT_INT7[1]	0x00
FLTEN7[0]	[7]	RW	Filter Enable for EXT_INT7[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH7[0]	[6:0]	RW	Filtering width of EXT_INT7[0]	0x00

4.4.2.130 EXT_INT7_FLTCON1

- Address = 0x1140_0834, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:0]	-	Reserved	0x00000000

4.4.2.131 EXT_INT8_FLTCON0

- Address = 0x1140_0838, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN8[3]	[31]	RW	Filter Enable for EXT_INT8[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH8[3]	[30:24]	RW	Filtering width of EXT_INT8[3]	0x00
FLTEN8[2]	[23]	RW	Filter Enable for EXT_INT8[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH8[2]	[22:16]	RW	Filtering width of EXT_INT8[2]	0x00
FLTEN8[1]	[15]	RW	Filter Enable for EXT_INT8[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH8[1]	[14:8]	RW	Filtering width of EXT_INT8[1]	0x00
FLTEN8[0]	[7]	RW	Filter Enable for EXT_INT8[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH8[0]	[6:0]	RW	Filtering width of EXT_INT8[0]	0x00

4.4.2.132 EXT_INT8_FLTCON1

- Address = 0x1140_083C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x000000
FLTEN8[4]	[7]	RW	Filter Enable for EXT_INT8[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH8[4]	[6:0]	RW	Filtering width of EXT_INT8[4]	0x00

4.4.2.133 EXT_INT9_FLTCON0

- Address = 0x1140_0840, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN9[3]	[31]	RW	Filter Enable for EXT_INT9[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH9[3]	[30:24]	RW	Filtering width of EXT_INT9[3]	0x00
FLTEN9[2]	[23]	RW	Filter Enable for EXT_INT9[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH9[2]	[22:16]	RW	Filtering width of EXT_INT9[2]	0x00
FLTEN9[1]	[15]	RW	Filter Enable for EXT_INT9[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH9[1]	[14:8]	RW	Filtering width of EXT_INT9[1]	0x00
FLTEN9[0]	[7]	RW	Filter Enable for EXT_INT9[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH9[0]	[6:0]	RW	Filtering width of EXT_INT9[0]	0x00

4.4.2.134 EXT_INT9_FLTCON1

- Address = 0x1140_0844, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN9[7]	[31]	RW	Filter Enable for EXT_INT9[7] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH9[7]	[30:24]	RW	Filtering width of EXT_INT9[7]	0x00
FLTEN9[6]	[23]	RW	Filter Enable for EXT_INT9[6] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH9[6]	[22:16]	RW	Filtering width of EXT_INT9[6]	0x00
FLTEN9[5]	[15]	RW	Filter Enable for EXT_INT9[5] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH9[5]	[14:8]	RW	Filtering width of EXT_INT9[5]	0x00
FLTEN9[4]	[7]	RW	Filter Enable for EXT_INT9[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH9[4]	[6:0]	RW	Filtering width of EXT_INT9[4]	0x00

4.4.2.135 EXT_INT10_FLTCON0

- Address = 0x1140_0848, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN10[3]	[31]	RW	Filter Enable for EXT_INT10[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH10[3]	[30:24]	RW	Filtering width of EXT_INT10[3]	0x00
FLTEN10[2]	[23]	RW	Filter Enable for EXT_INT10[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH10[2]	[22:16]	RW	Filtering width of EXT_INT10[2]	0x00
FLTEN10[1]	[15]	RW	Filter Enable for EXT_INT10[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH10[1]	[14:8]	RW	Filtering width of EXT_INT10[1]	0x00
FLTEN10[0]	[7]	RW	Filter Enable for EXT_INT10[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH10[0]	[6:0]	RW	Filtering width of EXT_INT10[0]	0x00

4.4.2.136 EXT_INT10_FLTCON1

- Address = 0x1140_084C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	–	Reserved	0x0000
FLTEN10[5]	[15]	RW	Filter Enable for EXT_INT10[5] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH10[5]	[14:8]	RW	Filtering width of EXT_INT10[5]	0x00
FLTEN10[4]	[7]	RW	Filter Enable for EXT_INT10[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH10[4]	[6:0]	RW	Filtering width of EXT_INT10[4]	0x00

4.4.2.137 EXT_INT11_FLTCON0

- Address = 0x1140_0850, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN11[3]	[31]	RW	Filter Enable for EXT_INT11[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH11[3]	[30:24]	RW	Filtering width of EXT_INT11[3]	0x00
FLTEN11[2]	[23]	RW	Filter Enable for EXT_INT11[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH11[2]	[22:16]	RW	Filtering width of EXT_INT11[2]	0x00
FLTEN11[1]	[15]	RW	Filter Enable for EXT_INT11[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH11[1]	[14:8]	RW	Filtering width of EXT_INT11[1]	0x00
FLTEN11[0]	[7]	RW	Filter Enable for EXT_INT11[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH11[0]	[6:0]	RW	Filtering width of EXT_INT11[0]	0x00

4.4.2.138 EXT_INT11_FLTCON1

- Address = 0x1140_0854, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN11[7]	[31]	RW	Filter Enable for EXT_INT11[7] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH11[7]	[30:24]	RW	Filtering width of EXT_INT11[7]	0x00
FLTEN11[6]	[23]	RW	Filter Enable for EXT_INT11[6] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH11[6]	[22:16]	RW	Filtering width of EXT_INT11[6]	0x00
FLTEN11[5]	[15]	RW	Filter Enable for EXT_INT11[5] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH11[5]	[14:8]	RW	Filtering width of EXT_INT11[5]	0x00
FLTEN11[4]	[7]	RW	Filter Enable for EXT_INT11[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH11[4]	[6:0]	RW	Filtering width of EXT_INT11[4]	0x00

4.4.2.139 EXT_INT12_FLTCON0

- Address = 0x1140_0858, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN12[3]	[31]	RW	Filter Enable for EXT_INT12[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH12[3]	[30:24]	RW	Filtering width of EXT_INT12[3]	0x00
FLTEN12[2]	[23]	RW	Filter Enable for EXT_INT12[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH12[2]	[22:16]	RW	Filtering width of EXT_INT12[2]	0x00
FLTEN12[1]	[15]	RW	Filter Enable for EXT_INT12[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH12[1]	[14:8]	RW	Filtering width of EXT_INT12[1]	0x00
FLTEN12[0]	[7]	RW	Filter Enable for EXT_INT12[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH12[0]	[6:0]	RW	Filtering width of EXT_INT12[0]	0x00

4.4.2.140 EXT_INT12_FLTCON1

- Address = 0x1140_085C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN12[7]	[31]	RW	Filter Enable for EXT_INT12[7] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH12[7]	[30:24]	RW	Filtering width of EXT_INT12[7]	0x00
FLTEN12[6]	[23]	RW	Filter Enable for EXT_INT12[6] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH12[6]	[22:16]	RW	Filtering width of EXT_INT12[6]	0x00
FLTEN12[5]	[15]	RW	Filter Enable for EXT_INT12[5] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH12[5]	[14:8]	RW	Filtering width of EXT_INT12[5]	0x00
FLTEN12[4]	[7]	RW	Filter Enable for EXT_INT12[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH12[4]	[6:0]	RW	Filtering width of EXT_INT12[4]	0x00

4.4.2.141 EXT_INT13_FLTCON0

- Address = 0x1140_0860, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN13[3]	[31]	RW	Filter Enable for EXT_INT13[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH13[3]	[30:24]	RW	Filtering width of EXT_INT13[3]	0x00
FLTEN13[2]	[23]	RW	Filter Enable for EXT_INT13[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH13[2]	[22:16]	RW	Filtering width of EXT_INT13[2]	0x00
FLTEN13[1]	[15]	RW	Filter Enable for EXT_INT13[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH13[1]	[14:8]	RW	Filtering width of EXT_INT13[1]	0x00
FLTEN13[0]	[7]	RW	Filter Enable for EXT_INT13[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH13[0]	[6:0]	RW	Filtering width of EXT_INT13[0]	0x00

4.4.2.142 EXT_INT13_FLTCON1

- Address = 0x1140_0864, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN13[7]	[31]	RW	Filter Enable for EXT_INT13[7] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH13[7]	[30:24]	RW	Filtering width of EXT_INT13[7]	0x00
FLTEN13[6]	[23]	RW	Filter Enable for EXT_INT13[6] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH13[6]	[22:16]	RW	Filtering width of EXT_INT13[6]	0x00
FLTEN13[5]	[15]	RW	Filter Enable for EXT_INT13[5] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH13[5]	[14:8]	RW	Filtering width of EXT_INT13[5]	0x00
FLTEN13[4]	[7]	RW	Filter Enable for EXT_INT13[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH13[4]	[6:0]	RW	Filtering width of EXT_INT13[4]	0x00

4.4.2.143 EXT_INT14_FLTCON0

- Address = 0x1140_0868, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN14[3]	[31]	RW	Filter Enable for EXT_INT14[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH14[3]	[30:24]	RW	Filtering width of EXT_INT14[3]	0x00
FLTEN14[2]	[23]	RW	Filter Enable for EXT_INT14[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH14[2]	[22:16]	RW	Filtering width of EXT_INT14[2]	0x00
FLTEN14[1]	[15]	RW	Filter Enable for EXT_INT14[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH14[1]	[14:8]	RW	Filtering width of EXT_INT14[1]	0x00
FLTEN14[0]	[7]	RW	Filter Enable for EXT_INT14[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH14[0]	[6:0]	RW	Filtering width of EXT_INT14[0]	0x00

4.4.2.144 EXT_INT14_FLTCON1

- Address = 0x1140_086C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN14[7]	[31]	RW	Filter Enable for EXT_INT14[7] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH14[7]	[30:24]	RW	Filtering width of EXT_INT14[7]	0x00
FLTEN14[6]	[23]	RW	Filter Enable for EXT_INT14[6] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH14[6]	[22:16]	RW	Filtering width of EXT_INT14[6]	0x00
FLTEN14[5]	[15]	RW	Filter Enable for EXT_INT14[5] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH14[5]	[14:8]	RW	Filtering width of EXT_INT14[5]	0x00
FLTEN14[4]	[7]	RW	Filter Enable for EXT_INT14[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH14[4]	[6:0]	RW	Filtering width of EXT_INT14[4]	0x00

4.4.2.145 EXT_INT15_FLTCON0

- Address = 0x1140_0870, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN15[3]	[31]	RW	Filter Enable for EXT_INT15[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH15[3]	[30:24]	RW	Filtering width of EXT_INT15[3]	0x00
FLTEN15[2]	[23]	RW	Filter Enable for EXT_INT15[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH15[2]	[22:16]	RW	Filtering width of EXT_INT15[2]	0x00
FLTEN15[1]	[15]	RW	Filter Enable for EXT_INT15[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH15[1]	[14:8]	RW	Filtering width of EXT_INT15[1]	0x00
FLTEN15[0]	[7]	RW	Filter Enable for EXT_INT15[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH15[0]	[6:0]	RW	Filtering width of EXT_INT15[0]	0x00

4.4.2.146 EXT_INT15_FLTCON1

- Address = 0x1140_0874, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN15[7]	[31]	RW	Filter Enable for EXT_INT15[7] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH15[7]	[30:24]	RW	Filtering width of EXT_INT15[7]	0x00
FLTEN15[6]	[23]	RW	Filter Enable for EXT_INT15[6] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH15[6]	[22:16]	RW	Filtering width of EXT_INT15[6]	0x00
FLTEN15[5]	[15]	RW	Filter Enable for EXT_INT15[5] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH15[5]	[14:8]	RW	Filtering width of EXT_INT15[5]	0x00
FLTEN15[4]	[7]	RW	Filter Enable for EXT_INT15[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH15[4]	[6:0]	RW	Filtering width of EXT_INT15[4]	0x00

4.4.2.147 EXT_INT16_FLTCON0

- Address = 0x1140_0878, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN16[3]	[31]	RW	Filter Enable for EXT_INT16[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH16[3]	[30:24]	RW	Filtering width of EXT_INT16[3]	0x00
FLTEN16[2]	[23]	RW	Filter Enable for EXT_INT16[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH16[2]	[22:16]	RW	Filtering width of EXT_INT16[2]	0x00
FLTEN16[1]	[15]	RW	Filter Enable for EXT_INT16[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH16[1]	[14:8]	RW	Filtering width of EXT_INT16[1]	0x00
FLTEN16[0]	[7]	RW	Filter Enable for EXT_INT16[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH16[0]	[6:0]	RW	Filtering width of EXT_INT16[0]	0x00

4.4.2.148 EXT_INT16_FLTCON1

- Address = 0x1140_087C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	–	Reserved	0x0000
FLTEN16[5]	[15]	RW	Filter Enable for EXT_INT16[5] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH16[5]	[14:8]	RW	Filtering width of EXT_INT16[5]	0x00
FLTEN16[4]	[7]	RW	Filter Enable for EXT_INT16[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH16[4]	[6:0]	RW	Filtering width of EXT_INT16[4]	0x00

4.4.2.149 EXT_INT1_MASK

- Address = 0x1140_0900, Reset Value = 0x0000_00FF

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x0000000
EXT_INT1_MASK[7]	[7]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT1_MASK[6]	[6]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT1_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT1_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT1_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT1_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT1_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT1_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.2.150 EXT_INT2_MASK

- Address = 0x1140_0904, Reset Value = 0x0000_003F

Name	Bit	Type	Description	Reset Value
RSVD	[31:6]	-	Reserved	0x00000000
EXT_INT2_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT2_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT2_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT2_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT2_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT2_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.2.151 EXT_INT3_MASK

- Address = 0x1140_0908, Reset Value = 0x0000_00FF

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x0000000
EXT_INT3_MASK[7]	[7]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT3_MASK[6]	[6]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT3_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT3_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT3_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT3_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT3_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT3_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.2.152 EXT_INT4_MASK

- Address = 0x1140_090C, Reset Value = 0x0000_001F

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	-	Reserved	0x00000000
EXT_INT4_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT4_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT4_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT4_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT4_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.2.153 EXT_INT5_MASK

- Address = 0x1140_0910, Reset Value = 0x0000_001F

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	-	Reserved	0x00000000
EXT_INT5_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT5_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT5_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT5_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT5_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.2.154 EXT_INT6_MASK

- Address = 0x1140_0914, Reset Value = 0x0000_000F

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	-	Reserved	0x00000000
EXT_INT6_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT6_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT6_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT6_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.2.155 EXT_INT7_MASK

- Address = 0x1140_0918, Reset Value = 0x0000_000F

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	-	Reserved	0x00000000
EXT_INT7_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT7_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT7_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT7_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.2.156 EXT_INT8_MASK

- Address = 0x1140_091C, Reset Value = 0x0000_001F

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	-	Reserved	0x00000000
EXT_INT8_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT8_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT8_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT8_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT8_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.2.157 EXT_INT9_MASK

- Address = 0x1140_0920, Reset Value = 0x0000_00FF

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x0000000
EXT_INT9_MASK[7]	[7]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT9_MASK[6]	[6]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT9_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT9_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT9_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT9_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT9_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT9_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.2.158 EXT_INT10_MASK

- Address = 0x1140_0924, Reset Value = 0x0000_003F

Name	Bit	Type	Description	Reset Value
RSVD	[31:6]	-	Reserved	0x00000000
EXT_INT10_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT10_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT10_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT10_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT10_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT10_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.2.159 EXT_INT11_MASK

- Address = 0x1140_0928, Reset Value = 0x0000_00FF

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x000000
EXT_INT11_MASK[7]	[7]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT11_MASK[6]	[6]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT11_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT11_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT11_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT11_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT11_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT11_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.2.160 EXT_INT12_MASK

- Address = 0x1140_092C, Reset Value = 0x0000_00FF

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x000000
EXT_INT12_MASK[7]	[7]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT12_MASK[6]	[6]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT12_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT12_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT12_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT12_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT12_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT12_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.2.161 EXT_INT13_MASK

- Address = 0x1140_0930, Reset Value = 0x0000_00FF

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x000000
EXT_INT13_MASK[7]	[7]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT13_MASK[6]	[6]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT13_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT13_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT13_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT13_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT13_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT13_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.2.162 EXT_INT14_MASK

- Address = 0x1140_0934, Reset Value = 0x0000_00FF

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x000000
EXT_INT14_MASK[7]	[7]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT14_MASK[6]	[6]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT14_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT14_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT14_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT14_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT14_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT14_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.2.163 EXT_INT15_MASK

- Address = 0x1140_0938, Reset Value = 0x0000_00FF

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x000000
EXT_INT15_MASK[7]	[7]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT15_MASK[6]	[6]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT15_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT15_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT15_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT15_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT15_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT15_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.2.164 EXT_INT16_MASK

- Address = 0x1140_093C, Reset Value = 0x0000_003F

Name	Bit	Type	Description	Reset Value
RSVD	[31:6]	-	Reserved	0x00000000
EXT_INT16_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT16_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT16_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT16_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT16_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT16_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.2.165 EXT_INT1_PEND

- Address = 0x1140_0A00, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x00000000
EXT_INT1_PEND[7]	[7]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT1_PEND[6]	[6]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT1_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT1_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT1_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT1_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT1_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT1_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.2.166 EXT_INT2_PEND

- Address = 0x1140_0A04, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:6]	-	Reserved	0x00000000
EXT_INT2_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT2_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT2_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT2_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT2_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT2_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.2.167 EXT_INT3_PEND

- Address = 0x1140_0A08, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x00000000
EXT_INT3_PEND[7]	[7]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT3_PEND[6]	[6]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT3_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT3_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT3_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT3_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT3_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT3_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.2.168 WAKEUP_INT_PEND

- Address = 0x1140_0A0C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	-	Reserved	0x00000000
WAKEUP_INT_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.2.169 EXT_INT5_PEND

- Address = 0x1140_0A10, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	-	Reserved	0x00000000
EXT_INT5_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT5_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT5_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT5_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT5_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.2.170 EXT_INT6_PEND

- Address = 0x1140_0A14, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	-	Reserved	0x00000000
EXT_INT6_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT6_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT6_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT6_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.2.171 EXT_INT7_PEND

- Address = 0x1140_0A18, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	-	Reserved	0x00000000
EXT_INT7_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT7_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT7_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT7_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.2.172 EXT_INT8_PEND

- Address = 0x1140_0A1C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	-	Reserved	0x00000000
EXT_INT8_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT8_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT8_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT8_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT8_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.2.173 EXT_INT9_PEND

- Address = 0x1140_0A20, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x0000000
EXT_INT9_PEND[7]	[7]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT9_PEND[6]	[6]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT9_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT9_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT9_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT9_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT9_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT9_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.2.174 EXT_INT10_PEND

- Address = 0x1140_0A24, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:6]	-	Reserved	0x00000000
EXT_INT10_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT10_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT10_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT10_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT10_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT10_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.2.175 EXT_INT11_PEND

- Address = 0x1140_0A28, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x0000000
EXT_INT11_PEND[7]	[7]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT11_PEND[6]	[6]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT11_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT11_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT11_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT11_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT11_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT11_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.2.176 EXT_INT12_PEND

- Address = 0x1140_0A2C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x0000000
EXT_INT12_PEND[7]	[7]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT12_PEND[6]	[6]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT12_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT12_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT12_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT12_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT12_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT12_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.2.177 EXT_INT13_PEND

- Address = 0x1140_0A30, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x0000000
EXT_INT13_PEND[7]	[7]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT13_PEND[6]	[6]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT13_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT13_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT13_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT13_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT13_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT13_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.2.178 EXT_INT14_PEND

- Address = 0x1140_0A34, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x0000000
EXT_INT14_PEND[7]	[7]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT14_PEND[6]	[6]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT14_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT14_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT14_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT14_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT14_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT14_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.2.179 EXT_INT15_PEND

- Address = 0x1140_0A38, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	RSVD	Reserved	0x0000000
EXT_INT15_PEND[7]	[7]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT15_PEND[6]	[6]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT15_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT15_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT15_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT15_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT15_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT15_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.2.180 EXT_INT16_PEND

- Address = 0x1140_0A3C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:6]	—	Reserved	0x00000000
EXT_INT16_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT16_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT16_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT16_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT16_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT16_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.2.181 EXT_INT_SERVICE_XA

- Address = 0x1140_0B08, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	RW	Reserved	0x00000000
SVC_Group_Num	[7:3]	RW	EXT_INT Service group number	0x00
SVC_Num	[2:0]	RW	Interrupt number to be serviced	0x0

4.4.2.182 EXT_INT_SERVICE_PEND_XA

- Address = 0x1140_0B0C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	RW	Reserved	0x00000000
SVC_PEND	[7:0]	RW	0x0 = Not occur 0x1 = Occur interrupt	0x00

4.4.2.183 EXT_INT_GRPFIXPRI_XA

- Address = 0x1140_0B10, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	-	Reserved	0x00000000
Highest_GRP_NUM	[3:0]	RW	Group number of the highest priority when fixed group priority mode: 0 (EXT_INT1) – 15 (EXT_INT16)	0x00

4.4.2.184 EXT_INT1_FIXPRI

- Address = 0x1140_0B14, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 1 (EXT_INT1) when fixed priority mode: 0 – 7	0x0

4.4.2.185 EXT_INT2_FIXPRI

- Address = 0x1140_0B18, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 2 (EXT_INT2) when fixed priority mode: 0 – 7	0x0

4.4.2.186 EXT_INT3_FIXPRI

- Address = 0x1140_0B1C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 3 (EXT_INT3) when fixed priority mode: 0 – 7	0x0

4.4.2.187 WAKEUP_INT_FIXPRI

- Address = 0x1140_0B20, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 4 (WAKEUP_INT) when fixed priority mode: 0 – 7	0x0

4.4.2.188 EXT_INT5_FIXPRI

- Address = 0x1140_0B24, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 5 (EXT_INT5) when fixed priority mode: 0 – 7	0x0

4.4.2.189 EXT_INT6_FIXPRI

- Address = 0x1140_0B28, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 6 (EXT_INT6) when fixed priority mode: 0 – 7	0x0

4.4.2.190 EXT_INT7_FIXPRI

- Address = 0x1140_0B2C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 7 (EXT_INT7) when fixed priority mode: 0 – 7	0x0

4.4.2.191 EXT_INT8_FIXPRI

- Address = 0x1140_0B30, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 8 (EXT_INT8) when fixed priority mode: 0 – 7	0x0

4.4.2.192 EXT_INT9_FIXPRI

- Address = 0x1140_0B34, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 9 (EXT_INT9) when fixed priority mode: 0 – 7	0x0

4.4.2.193 EXT_INT10_FIXPRI

- Address = 0x1140_0B38, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 10 (EXT_INT10) when fixed priority mode: 0 – 7	0x0

4.4.2.194 EXT_INT11_FIXPRI

- Address = 0x1140_0B3C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 11 (EXT_INT11) when fixed priority mode: 0 – 7	0x0

4.4.2.195 EXT_INT12_FIXPRI

- Address = 0x1140_0B40, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 12 (EXT_INT12) when fixed priority mode: 0 – 7	0x0

4.4.2.196 EXT_INT13_FIXPRI

- Address = 0x1140_0B44, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 13 (EXT_INT13) when fixed priority mode: 0 – 7	0x0

4.4.2.197 EXT_INT14_FIXPRI

- Address = 0x1140_0B48, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 14 (EXT_INT14) when fixed priority mode: 0 – 7	0x0

4.4.2.198 EXT_INT15_FIXPRI

- Address = 0x1140_0B4C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 15 (EXT_INT15) when fixed priority mode: 0 – 7	0x0

4.4.2.199 EXT_INT16_FIXPRI

- Address = 0x1140_0B50, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 16 (EXT_INT16) when fixed priority mode: 0~7	0x0

4.4.3 Part 2

For the following SFRs, setting the value does not take effect immediately. It takes at least 800 APB clocks for the value to take effect after the SFR is actually changed: GPJ0PUD, GPJ0DRV, GPJ1PUD, GPJ1DRV, GPK0PUD, GPK0DRV, GPK1PUD, GPK1DRV, GPK2PUD, GPK2DRV, GPK3PUD, GPK3DRV, GPL0PUD, GPL0DRV, GPL1PUD, GPL1DRV, GPL2PUD, GPL2DRV, GPY0PUD, GPY0DRV, GPY1PUD, GPY1DRV, GPY2PUD, GPY2DRV, GPY3PUD, GPY3DRV, GPY4PUD, GPY4DRV, GPY5PUD, GPY5DRV, GPY6PUD, GPY6DRV

4.4.3.1 GPJ0CON

- Address = 0x1100_0000, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPJ0CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = CAM_A_PCLK 0x3 – 0xE = Reserved 0xF = EXT_INT21[0]	0x00
GPJ0CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = CAM_A_VSYNC 0x3 – 0xE = Reserved 0xF = EXT_INT21[1]	0x00
GPJ0CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = CAM_A_HREF 0x3 – 0xE = Reserved 0xF = EXT_INT21[2]	0x00
GPJ0CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = CAM_A_DATA[0] 0x3 – 0xE = Reserved 0xF = EXT_INT21[3]	0x00
GPJ0CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = CAM_A_DATA[1] 0x3 – 0xE = Reserved 0xF = EXT_INT21[4]	0x00
GPJ0CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = CAM_A_DATA[2] 0x3 – 0xE = Reserved 0xF = EXT_INT21[5]	0x00
GPJ0CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = CAM_A_DATA[3] 0x3 – 0xE = Reserved 0xF = EXT_INT21[6]	0x00
GPJ0CON[7]	[31:28]	RW	0x0 = Input 0x1 = Output 0x2 = CAM_A_DATA[4] 0x3 – 0xE = Reserved 0xF = EXT_INT21[7]	0x00

4.4.3.2 GPJ0DAT

- Address = 0x1100_0004, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPJ0DAT[7:0]	[7:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.3 GPJ0PUD

- Address = 0x1100_0008, Reset Value = 0x5555

Name	Bit	Type	Description	Reset Value
GPJ0PUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x5555

4.4.3.4 GPJ0DRV

- Address = 0x1100_000C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
GPJ0DRV[n]	[2n+1:2n] n=0 – 7	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.3.5 GPJ0CONPDN

- Address = 0x1100_0010, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPJ0[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.3.6 GPJ0PUPDN

- Address = 0x1100_0014, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPJ0[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.3.7 GPJ1CON

- Address = 0x1100_0020, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPJ1CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = CAM_A_DATA[5] 0x3 – 0xE = Reserved 0xF = EXT_INT22[0]	0x00
GPJ1CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = CAM_A_DATA[6] 0x3 – 0xE = Reserved 0xF = EXT_INT22[1]	0x00
GPJ1CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = CAM_A_DATA[7] 0x3 – 0xE = Reserved 0xF = EXT_INT22[2]	0x00
GPJ1CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = CAM_A_CLKOUT 0x3 – 0xE = Reserved 0xF = EXT_INT22[3]	0x00
GPJ1CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = CAM_A_FIELD 0x3 – 0xE = Reserved 0xF = EXT_INT22[4]	0x00

4.4.3.8 GPJ1DAT

- Address = 0x1100_0024, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPJ1DAT[4:0]	[4:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.9 GPJ1PUD

- Address = 0x1100_0028, Reset Value = 0x0155

Name	Bit	Type	Description	Reset Value
GPJ1PUD[n]	[2n+1:2n] n=0 – 4	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x0155

4.4.3.10 GPJ1DRV

- Address = 0x1100_002C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
GPJ1DRV[n]	[2n+1:2n] n=0 – 4	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.3.11 GPJ1CONPDN

- Address = 0x1100_0030, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPJ1[n]	[2n+1:2n] n=0 – 4	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.3.12 GPJ1PUDPDN

- Address = 0x1100_0034, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPJ1[n]	[2n+1:2n] n=0 – 4	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.3.13 GPK0CON

- Address = 0x1100_0040, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPK0CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = SD_0_CLK 0x3 = SD_4_CLK 0x4 – 0xE = Reserved 0xF = EXT_INT23[0]	0x00
GPK0CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = SD_0_CMD 0x3 = SD_4_CMD 0x4 – 0xE = Reserved 0xF = EXT_INT23[1]	0x00
GPK0CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = SD_0_CDn 0x3 = SD_4_CDn 0x4 – 0xE = Reserved 0xF = EXT_INT23[2]	0x00
GPK0CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = SD_0_DATA[0] 0x3 = SD_4_DATA[0] 0x4 – 0xE = Reserved 0xF = EXT_INT23[3]	0x00
GPK0CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = SD_0_DATA[1] 0x3 = SD_4_DATA[1] 0x4 – 0xE = Reserved 0xF = EXT_INT23[4]	0x00
GPK0CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = SD_0_DATA[2] 0x3 = SD_4_DATA[2] 0x4 – 0xE = Reserved	0x00

Name	Bit	Type	Description	Reset Value
			0xF = EXT_INT23[5]	
GPK0CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = SD_0_DATA[3] 0x3 = SD_4_DATA[3] 0x4 – 0xE = Reserved 0xF = EXT_INT23[6]	0x00

4.4.3.14 GPK0DAT

- Address = 0x1100_0044, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPK0DAT[6:0]	[6:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.15 GPK0PUD

- Address = 0x1100_0048, Reset Value = 0x1555

Name	Bit	Type	Description	Reset Value
GPK0PUD[n]	[2n+1:2n] n=0 – 6	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x1555

4.4.3.16 GPK0DRV

- Address = 0x1100_004C, Reset Value = 0x00_2AAA

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
GPK0DRV[n]	[2n+1:2n] n=0 – 6	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_2AAA

4.4.3.17 GPK0CONPDN

- Address = 0x1100_0050, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPK0[n]	[2n+1:2n] n=0 – 6	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.3.18 GPK0PUDPDN

- Address = 0x1100_0054, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPK0[n]	[2n+1:2n] n=0 – 6	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.3.19 GPK1CON

- Address = 0x1100_0060, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPK1CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = SD_1_CLK 0x3 – 0xE = Reserved 0xF = EXT_INT24[0]	0x00
GPK1CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = SD_1_CMD 0x3 – 0xE = Reserved 0xF = EXT_INT24[1]	0x00
GPK1CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = SD_1_CDn 0x3 = SATA_P0_ACT_LED 0x4 = SD_4_nRESET_OUT 0x5 – 0xE = Reserved 0xF = EXT_INT24[2]	0x00
GPK1CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = SD_1_DATA[0] 0x3 = SD_0_DATA[4] 0x4 = SD_4_DATA[4] 0x5 – 0xE = Reserved 0xF = EXT_INT24[3]	0x00

Name	Bit	Type	Description	Reset Value
GPK1CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = SD_1_DATA[1] 0x3 = SD_0_DATA[5] 0x4 = SD_4_DATA[5] 0x5 – 0xE = Reserved 0xF = EXT_INT24[4]	0x00
GPK1CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = SD_1_DATA[2] 0x3 = SD_0_DATA[6] 0x4 = SD_4_DATA[6] 0x5 – 0xE = Reserved 0xF = EXT_INT24[5]	0x00
GPK1CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = SD_1_DATA[3] 0x3 = SD_0_DATA[7] 0x4 = SD_4_DATA[7] 0x5 – 0xE = Reserved 0xF = EXT_INT24[6]	0x00

4.4.3.20 GPK1DAT

- Address = 0x1100_0064, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPK1DAT[6:0]	[6:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.21 GPK1PUD

- Address = 0x1100_0068, Reset Value = 0x1555

Name	Bit	Type	Description	Reset Value
GPK1PUD[n]	[2n+1:2n] n=0 – 6	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x1555

4.4.3.22 GPK1DRV

- Address = 0x1100_006C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
GPK1DRV[n]	[2n+1:2n] n=0 – 6	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.3.23 GPK1CONPDN

- Address = 0x1100_0070, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPK1[n]	[2n+1:2n] n=0 – 6	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.3.24 GPK1PUPDN

- Address = 0x1100_0074, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPK1[n]	[2n+1:2n] n=0 – 6	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.3.25 GPK2CON

- Address = 0x1100_0080, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPK2CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = SD_2_CLK 0x3 – 0xE = Reserved 0xF = EXT_INT25[0]	0x00
GPK2CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = SD_2_CMD 0x3 – 0xE = Reserved 0xF = EXT_INT25[1]	0x00
GPK2CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = SD_2_CDn 0x3 – 0xE = Reserved 0xF = EXT_INT25[2]	0x00
GPK2CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = SD_2_DATA[0] 0x3 – 0xE = Reserved 0xF = EXT_INT25[3]	0x00
GPK2CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = SD_2_DATA[1] 0x3 – 0xE = Reserved 0xF = EXT_INT25[4]	0x00
GPK2CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = SD_2_DATA[2] 0x3 – 0xE = Reserved 0xF = EXT_INT25[5]	0x00
GPK2CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = SD_2_DATA[3] 0x3 – 0xE = Reserved	0x00

Name	Bit	Type	Description	Reset Value
			0xF = EXT_INT25[6]	

4.4.3.26 GPK2DAT

- Address = 0x1100_0084, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPK2DAT[6:0]	[6:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.27 GPK2PUD

- Address = 0x1100_0088, Reset Value = 0x1555

Name	Bit	Type	Description	Reset Value
GPK2PUD[n]	[2n+1:2n] n=0 – 6	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x1555

4.4.3.28 GPK2DRV

- Address = 0x1100_008C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
GPK2DRV[n]	[2n+1:2n] n=0 – 6	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.3.29 GPK2CONPDN

- Address = 0x1100_0090, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPK2[n]	[2n+1:2n] n=0 – 6	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.3.30 GPK2PUPDN

- Address = 0x1100_0094, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPK2[n]	[2n+1:2n] n=0 – 6	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.3.31 GPK3CON

- Address = 0x1100_00A0, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPK3CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = SD_3_CLK 0x3 – 0xE = Reserved 0xF = EXT_INT26[0]	0x00
GPK3CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = SD_3_CMD 0x3 – 0xE = Reserved 0xF = EXT_INT26[1]	0x00
GPK3CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = SD_3_CDn 0x3 – 0xE = Reserved 0xF = EXT_INT26[2]	0x00
GPK3CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = SD_3_DATA[0] 0x3 = SD_2_DATA[4] 0x4 – 0xE = Reserved 0xF = EXT_INT26[3]	0x00
GPK3CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = SD_3_DATA[1] 0x3 = SD_2_DATA[5] 0x4 – 0xE = Reserved 0xF = EXT_INT26[4]	0x00
GPK3CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = SD_3_DATA[2] 0x3 = SD_2_DATA[6] 0x4 – 0xE = Reserved 0xF = EXT_INT26[5]	0x00
GPK3CON[6]	[27:24]	RW	0x0 = Input	0x00

Name	Bit	Type	Description	Reset Value
			0x1 = Output 0x2 = SD_3_DATA[3] 0x3 = SD_2_DATA[7] 0x4 – 0xE = Reserved 0xF = EXT_INT26[6]	

4.4.3.32 GPK3DAT

- Address = 0x1100_00A4, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPK3DAT[6:0]	[6:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.33 GPK3PUD

- Address = 0x1100_00A8, Reset Value = 0x1555

Name	Bit	Type	Description	Reset Value
GPK3PUD[n]	[2n+1:2n] n=0 – 6	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserve 0x3 = Pull-up enabled	0x1555

4.4.3.34 GPK3DRV

- Address = 0x1100_00AC, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
GPK3DRV[n]	[2n+1:2n] n=0 – 6	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.3.35 GPK3CONPDN

- Address = 0x1100_00B0, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPK3[n]	[2n+1:2n] n=0 – 6	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.3.36 GPK3PUDPDN

- Address = 0x1100_00B4, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPK3[n]	[2n+1:2n] n=0 – 6	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.3.37 GPL0CON

- Address = 0x1100_00C0, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPL0CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = GNSS_SYNC 0x3 – 0xE = Reserved 0xF = EXT_INT27[0]	0x00
GPL0CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = GNSS_ISIGN 0x3 – 0xE = Reserved 0xF = EXT_INT27[1]	0x00
GPL0CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = GNSS_IMAG 0x3 – 0xE = Reserved 0xF = EXT_INT27[2]	0x00
GPL0CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = GNSS_QSIGN 0x3 – 0xE = Reserved 0xF = EXT_INT27[3]	0x00
GPL0CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = GNSS_QMAG 0x3 – 0xE = Reserved	0x00

Name	Bit	Type	Description	Reset Value
			0xF = EXT_INT27[4]	
GPL0CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = GNSS_MCLK 0x3 – 0xE = Reserved 0xF = EXT_INT27[5]	0x00
GPL0CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = GNSS_RF_RSTN 0x3 – 0xE = Reserved 0xF = EXT_INT27[6]	0x00
GPL0CON[7]	[31:28]	RW	0x0 = Input 0x1 = Output 0x2 = GNSS_CLKREQ 0x3 – 0xE = Reserved 0xF = EXT_INT27[7]	0x00

4.4.3.38 GPL0DAT

- Address = 0x1100_00C4, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPL0DAT[7:0]	[7:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.39 GPL0PUD

- Address = 0x1100_00C8, Reset Value = 0x5555

Name	Bit	Type	Description	Reset Value
GPL0PUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x5555

4.4.3.40 GPL0DRV

- Address = 0x1100_00CC, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
GPL0DRV[n]	[2n+1:2n] n=0 – 7	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.3.41 GPL0CONPDN

- Address = 0x1100_00D0, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPL0[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.3.42 GPL0PUPDPDN

- Address = 0x1100_00D4, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPL0[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.3.43 GPL1CON

- Address = 0x1100_00E0, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPL1CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = GNSS_SCL 0x3 – 0xE = Reserved 0xF = EXT_INT28[0]	0x00
GPL1CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = GNSS_SDA 0x3 – 0xE = Reserved 0xF = EXT_INT28[1]	0x00
GPL1CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = GNSS_EPOCH 0x3 – 0xE = Reserved 0xF = EXT_INT28[2]	0x00

4.4.3.44 GPL1DAT

- Address = 0x1100_00E4, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPL1DAT[2:0]	[2:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.45 GPL1PUD

- Address = 0x1100_00E8, Reset Value = 0x0015

Name	Bit	Type	Description	Reset Value
GPL1PUD[n]	[2n+1:2n] n=0 – 2	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x0015

4.4.3.46 GPL1DRV

- Address = 0x1100_00EC, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
GPL1DRV[n]	[2n+1:2n] n=0 – 2	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.3.47 GPL1CONPDN

- Address = 0x1100_00F0, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPL1[n]	[2n+1:2n] n=0 – 2	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.3.48 GPL1PUPD�

- Address = 0x1100_00F4, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPL1[n]	[2n+1:2n] n=0 – 2	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.3.49 GPL2CON

- Address = 0x1100_0100, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPL2CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = GNSS_GPIO[0] 0x3 = KP_COL[0] 0x4 – 0xE = Reserved 0xF = EXT_INT29[0]	0x00
GPL2CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = GNSS_GPIO[1] 0x3 = KP_COL[1] 0x4 – 0xE = Reserved 0xF = EXT_INT29[1]	0x00
GPL2CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = GNSS_GPIO[2] 0x3 = KP_COL[2] 0x4 – 0xE = Reserved 0xF = EXT_INT29[2]	0x00
GPL2CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = GNSS_GPIO[3] 0x3 = KP_COL[3] 0x4 – 0xE = Reserved 0xF = EXT_INT29[3]	0x00
GPL2CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = GNSS_GPIO[4] 0x3 = KP_COL[4] 0x4 – 0xE = Reserved 0xF = EXT_INT29[4]	0x00
GPL2CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = GNSS_GPIO[5] 0x3 = KP_COL[5] 0x4 – 0xE = Reserved 0xF = EXT_INT29[5]	0x00
GPL2CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = GNSS_GPIO[6] 0x3 = KP_COL[6] 0x4 – 0xE = Reserved 0xF = EXT_INT29[6]	0x00
GPL2CON[7]	[31:28]	RW	0x0 = Input 0x1 = Output 0x2 = GNSS_GPIO[7]	0x00

Name	Bit	Type	Description	Reset Value
			0x3 = KP_COL[7] 0x4 – 0xE = Reserved 0xF = EXT_INT29[7]	

4.4.3.50 GPL2DAT

- Address = 0x1100_0104, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPL2DAT[7:0]	[7:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.51 GPL2PUD

- Address = 0x1100_0108, Reset Value = 0x5555

Name	Bit	Type	Description	Reset Value
GPL2PUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x5555

4.4.3.52 GPL2DRV

- Address = 0x1100_010C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
GPL2DRV[n]	[2n+1:2n] n=0 – 7	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.3.53 GPL2CONPDN

- Address = 0x1100_0110, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPL2[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.3.54 GPL2PUDPDN

- Address = 0x1100_0114, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPL2[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.3.55 GPY0CON

- Address = 0x1100_0120, Reset Value = 0x0022_5522

Name	Bit	Type	Description	Reset Value
GPY0CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = SROM_CS _n [0] 0x3 = NF_CS _n [2] 0x4 – 0xE = Reserved 0xF = –	0x02
GPY0CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = SROM_CS _n [1] 0x3 = NF_CS _n [3] 0x4 – 0xE = Reserved 0xF = –	0x02
GPY0CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = SROM_CS _n [2] 0x3 = NF_CS _n [0] 0x4 = Reserved 0x5 = OND_CS _n [0] 0x6 – 0xE = Reserved 0xF = –	0x05
GPY0CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = SROM_CS _n [3] 0x3 = NF_CS _n [1] 0x4 = Reserved 0x5 = OND_CS _n [1] 0x6 – 0xE = Reserved 0xF = –	0x05
GPY0CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_OEn 0x3 – 0xE = Reserved 0xF = –	0x02
GPY0CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output	0x02

Name	Bit	Type	Description	Reset Value
			0x2 = EBI_WEn 0x3 – 0xE = Reserved 0xF = –	

4.4.3.56 GPY0DAT

- Address = 0x1100_0124, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPY0DAT[5:0]	[5:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.57 GPY0PUD

- Address = 0x1100_0128, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY0PUD[n]	[2n+1:2n] n=0 – 5	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x0000

4.4.3.58 GPY0DRV

- Address = 0x1100_012C, Reset Value = 0x00_0AAA

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
GPY0DRV[n]	[2n+1:2n] n=0 – 5	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0AAA

4.4.3.59 GPY0CONPDN

- Address = 0x1100_0130, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY0[n]	[2n+1:2n] n=0 – 5	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.3.60 GPY0PUDPDN

- Address = 0x1100_0134, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY0[n]	[2n+1:2n] n=0 – 5	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.3.61 GPY1CON

- Address = 0x1100_0140, Reset Value = 0x0000_2222

Name	Bit	Type	Description	Reset Value
GPY1CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_BEn[0] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY1CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_BEn[1] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY1CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = SROM_WAITn 0x3 – 0xE = Reserved 0xF = –	0x02
GPY1CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_DATA_RDn 0x3 – 0xE = Reserved 0xF = –	0x02

4.4.3.62 GPY1DAT

- Address = 0x1100_0144, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPY1DAT[3:0]	[3:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.63 GPY1PUD

- Address = 0x1100_0148, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY1PUD[n]	[2n+1:2n] n=0 – 3	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x0000

4.4.3.64 GPY1DRV

- Address = 0x1100_014C, Reset Value = 0x00_00AA

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
GPY1DRV[n]	[2n+1:2n] n=0 – 3	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_00AA

4.4.3.65 GPY1CONPDN

- Address = 0x1100_0150, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY1[n]	[2n+1:2n] n=0 – 3	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.3.66 GPY1PUPDPDN

- Address = 0x1100_0154, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY1[n]	[2n+1:2n] n=0~3	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.3.67 GPY2CON

- Address = 0x1100_0160, Reset Value = 0x0025_5555

Name	Bit	Type	Description	Reset Value
GPY2CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = NF_CLE 0x3 = Reserved 0x4 = Reserved 0x5 = OND_ADDRVALID 0x6 – 0xE = Reserved 0xF = –	0x05
GPY2CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = NF_ALE 0x3 = Reserved 0x4 = Reserved 0x5 = OND_SMCLK 0x6 – 0xE = Reserved 0xF = –	0x05
GPY2CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = NF_RnB[0] 0x3 = Reserved 0x4 = Reserved 0x5 = OND_INT[0] 0x6 – 0xE = Reserved 0xF = –	0x05
GPY2CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = NF_RnB[1] 0x3 = Reserved 0x4 = Reserved 0x5 = OND_INT[1] 0x6 – 0xE = Reserved 0xF = –	0x05
GPY2CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = NF_RnB[2] 0x3 = Reserved 0x4 = Reserved 0x5 = OND_RPn 0x6 – 0xE = Reserved 0xF = –	0x05
GPY2CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = NF_RnB[3] 0x3 – 0xE = Reserved 0xF = –	0x02

4.4.3.68 GPY2DAT

- Address = 0x1100_0164, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPY2DAT[5:0]	[5:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.69 GPY2PUD

- Address = 0x1100_0168, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY2PUD[n]	[2n+1:2n] n=0 – 5	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x0000

4.4.3.70 GPY2DRV

- Address = 0x1100_016C, Reset Value = 0x00_0AAA

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
GPY2DRV[n]	[2n+1:2n] n=0 – 5	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0AAA

4.4.3.71 GPY2CONPDN

- Address = 0x1100_0170, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY2[n]	[2n+1:2n] n=0 – 5	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.3.72 GPY2PUDPDN

- Address = 0x1100_0174, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY2[n]	[2n+1:2n] n=0 – 5	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.3.73 GPY3CON

- Address = 0x1100_0180, Reset Value = 0x2222_2222

Name	Bit	Type	Description	Reset Value
GPY3CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_ADDR[0] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY3CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_ADDR[1] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY3CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_ADDR[2] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY3CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_ADDR[3] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY3CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_ADDR[4] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY3CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_ADDR[5] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY3CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_ADDR[6] 0x3 – 0xE = Reserved	0x02

Name	Bit	Type	Description	Reset Value
			0xF = -	
GPY3CON[7]	[31:28]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_ADDR[7] 0x3 – 0xE = Reserved 0xF = -	0x02

4.4.3.74 GPY3DAT

- Address = 0x1100_0184, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPY3DAT[7:0]	[7:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.75 GPY3PUD

- Address = 0x1100_0188, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY3PUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x0000

4.4.3.76 GPY3DRV

- Address = 0x1100_018C, Reset Value = 0x00_AAAA

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
GPY3DRV[n]	[2n+1:2n] n=0 – 7	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_AAAA

4.4.3.77 GPY3CONPDN

- Address = 0x1100_0190, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY3[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.3.78 GPY3PUDPDN

- Address = 0x1100_0194, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY3[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.3.79 GPY4CON

- Address = 0x1100_01A0, Reset Value = 0x2222_2222

Name	Bit	Type	Description	Reset Value
GPY4CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_ADDR[8] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY4CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_ADDR[9] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY4CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_ADDR[10] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY4CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_ADDR[11] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY4CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_ADDR[12] 0x3 – 0xE = Reserved	0x02

Name	Bit	Type	Description	Reset Value
			0xF = –	
GPY4CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_ADDR[13] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY4CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_ADDR[14] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY4CON[7]	[31:28]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_ADDR[15] 0x3 – 0xE = Reserved 0xF = –	0x02

4.4.3.80 GPy4DAT

- Address = 0x1100_01A4, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPy4DAT[7:0]	[7:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.81 GPy4PUD

- Address = 0x1100_01A8, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPy4PUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x0000

4.4.3.82 GPY4DRV

- Address = 0x1100_01AC, Reset Value = 0x00_AAAA

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
GPY4DRV[n]	[2n+1:2n] n=0 – 7	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_AAAA

4.4.3.83 GPY4CONPDN

- Address = 0x1100_01B0, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY4[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.3.84 GPY4PUPDWN

- Address = 0x1100_01B4, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY4[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.3.85 GPY5CON

- Address = 0x1100_01C0, Reset Value = 0x2222_2222

Name	Bit	Type	Description	Reset Value
GPY5CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_DATA[0] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY5CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_DATA[1] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY5CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_DATA[2] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY5CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_DATA[3] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY5CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_DATA[4] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY5CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_DATA[5] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY5CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_DATA[6] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY5CON[7]	[31:28]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_DATA[7] 0x3 – 0xE = Reserved 0xF = –	0x02

4.4.3.86 GPY5DAT

- Address = 0x1100_01C4, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPY5DAT[7:0]	[7:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.87 GPY5PUD

- Address = 0x1100_01C8, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY5PUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x0000

4.4.3.88 GPY5DRV

- Address = 0x1100_01CC, Reset Value = 0x00_AAAA

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
GPY5DRV[n]	[2n+1:2n] n=0 – 7	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_AAAA

4.4.3.89 GPY5CONPDN

- Address = 0x1100_01D0, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY5[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.3.90 GPY5PUDPDN

- Address = 0x1100_01D4, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY5[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.3.91 GPY6CON

- Address = 0x1100_01E0, Reset Value = 0x2222_2222

Name	Bit	Type	Description	Reset Value
GPY6CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_DATA[8] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY6CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_DATA[9] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY6CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_DATA[10] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY6CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_DATA[11] 0x3 – 0xE = Reserved 0xF = --	0x02
GPY6CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_DATA[12] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY6CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_DATA[13] 0x3 – 0xE = Reserved 0xF = –	0x02
GPY6CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_DATA[14] 0x3 – 0xE = Reserved	0x02

Name	Bit	Type	Description	Reset Value
			0xF = -	
GPY6CON[7]	[31:28]	RW	0x0 = Input 0x1 = Output 0x2 = EBI_DATA[15] 0x3 – 0xE = Reserved 0xF = -	0x02

4.4.3.92 GPY6DAT

- Address = 0x1100_01E4, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPY6DAT[7:0]	[7:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.93 GPY6PUD

- Address = 0x1100_01E8, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY6PUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x0000

4.4.3.94 GPY6DRV

- Address = 0x1100_01EC, Reset Value = 0x00_AAAA

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
GPY6DRV[n]	[2n+1:2n] n=0 – 7	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_AAAA

4.4.3.95 GPY6CONPDN

- Address = 0x1100_01F0, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY6[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.3.96 GPY6PUDPDN

- Address = 0x1100_01F4, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPY6[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

4.4.3.97 ETC6PUD

- Address = 0x1100_0228, Reset Value = 0xC0C0

Name	Bit	Type	Description	Reset Value
ETC6PUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0xC0C0

ETC6PUD[11:10] controls XuotgDRVVBUS.

ETC6PUD[13:12] controls XuhostPWREN.

ETC6PUD[15:14] controls XuhostOVERCUR.

4.4.3.98 ETC6DRV

- Address = 0x1100_022C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
ETC6DRV[n]	[2n+1:2n] n=0 – 7	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

ETC6DRV[3:2] controls XCLKOUT.

ETC6DRV[11:10] controls XuotgDRVVBUS.

ETC6DRV[13:12] controls XuhostPWREN.

ETC6DRV[15:14] controls XuhostOVERCUR.

4.4.3.99 EXT_INT21CON

- Address = 0x1100_0700, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31]	–	Reserved	0x0
EXT_INT21_CON[7]	[30:28]	RW	Setting the signaling method of EXT_INT21[7] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[27]	–	Reserved	0x0
EXT_INT21_CON[6]	[26:24]	RW	Setting the signaling method of EXT_INT21[6] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[23]	–	Reserved	0x0
EXT_INT21_CON[5]	[22:20]	RW	Setting the signaling method of EXT_INT21[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	–	Reserved	0x0

Name	Bit	Type	Description	Reset Value
EXT_INT21_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT21[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT21_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT21[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT21_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT21[2] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	-	Reserved	0x0
EXT_INT21_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT21[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT21_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT21[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.3.100 EXT_INT22CON

- Address = 0x1100_0704, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:20]	-	Reserved	0x000
RSVD	[19]	-	Reserved	0x0
EXT_INT22_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT22[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT22_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT22[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT22_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT22[2] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	-	Reserved	0x0
EXT_INT22_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT22[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT22_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT22[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.3.101 EXT_INT23CON

- Address = 0x1100_0708, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:28]	-	Reserved	0x0
RSVD	[27]	-	Reserved	0x0
EXT_INT23_CON[6]	[26:24]	RW	Setting the signaling method of EXT_INT23[6] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[23]	-	Reserved	0x0
EXT_INT23_CON[5]	[22:20]	RW	Setting the signaling method of EXT_INT23[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
EXT_INT23_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT23[4] 0x0 = Low level, 0x1 = High level, 0x2 = Falling edge triggered, 0x3 = Rising edge triggered, 0x4 = Both edge triggered, 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT23_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT23[3] 0x0 = Low level, 0x1 = High level, 0x2 = Falling edge triggered, 0x3 = Rising edge triggered, 0x4 = Both edge triggered, 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT23_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT23[2] 0x0 = Low level, 0x1 = High level, 0x2 = Falling edge triggered, 0x3 = Rising edge triggered, 0x4 = Both edge triggered, 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	-	Reserved	0x0

Name	Bit	Type	Description	Reset Value
EXT_INT23_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT23[1] 0x0 = Low level, 0x1 = High level, 0x2 = Falling edge triggered, 0x3 = Rising edge triggered, 0x4 = Both edge triggered, 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT23_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT23[0] 0x0 = Low level, 0x1 = High level, 0x2 = Falling edge triggered, 0x3 = Rising edge triggered, 0x4 = Both edge triggered, 0x5 – 0x7 = Reserved	0x0

4.4.3.102 EXT_INT24CON

- Address = 0x1100_070C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:28]	-	Reserved	0x0
RSVD	[27]	-	Reserved	0x0
EXT_INT24_CON[6]	[26:24]	RW	Setting the signaling method of EXT_INT24[6] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[23]	-	Reserved	0x0
EXT_INT24_CON[5]	[22:20]	RW	Setting the signaling method of EXT_INT24[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
EXT_INT24_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT24[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT24_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT24[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT24_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT24[2] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	-	Reserved	0x0

Name	Bit	Type	Description	Reset Value
EXT_INT24_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT24[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT24_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT24[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.3.103 EXT_INT25CON

- Address = 0x1100_0710, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:28]	-	Reserved	0x0
RSVD	[27]	-	Reserved	0x0
EXT_INT25_CON[6]	[26:24]	RW	Setting the signaling method of EXT_INT25[6] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[23]	-	Reserved	0x0
EXT_INT25_CON[5]	[22:20]	RW	Setting the signaling method of EXT_INT25[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
EXT_INT25_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT25[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT25_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT25[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT25_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT25[2] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	-	Reserved	0x0

Name	Bit	Type	Description	Reset Value
EXT_INT25_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT25[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT25_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT25[0] 0x0 = Low level, 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.3.104 EXT_INT26CON

- Address = 0x1100_0714, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:28]	-	Reserved	0x0
RSVD	[27]	-	Reserved	0x0
EXT_INT26_CON[6]	[26:24]	RW	Setting the signaling method of EXT_INT26[6] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[23]	-	Reserved	0x0
EXT_INT26_CON[5]	[22:20]	RW	Setting the signaling method of EXT_INT26[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
EXT_INT26_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT26[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT26_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT26[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT26_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT26[2] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	-	Reserved	0x0

Name	Bit	Type	Description	Reset Value
EXT_INT26_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT26[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT26_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT26[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.3.105 EXT_INT27CON

- Address = 0x1100_0718, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31]	-	Reserved	0x0
EXT_INT27_CON[7]	[30:28]	RW	Setting the signaling method of EXT_INT27[7] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[27]	-	Reserved	0x0
EXT_INT27_CON[6]	[26:24]	RW	Setting the signaling method of EXT_INT27[6] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[23]	-	Reserved	0x0
EXT_INT27_CON[5]	[22:20]	RW	Setting the signaling method of EXT_INT27[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
EXT_INT27_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT27[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT27_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT27[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT27_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT27[2]	0x0

Name	Bit	Type	Description	Reset Value
			0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	
RSVD	[7]	-	Reserved	0x0
EXT_INT27_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT27[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT27_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT27[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.3.106 EXT_INT28CON

- Address = 0x1100_071C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:12]	–	Reserved	0x00000
RSVD	[11]	–	Reserved	0x0
EXT_INT28_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT28[2] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	–	Reserved	0x0
EXT_INT28_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT28[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	–	Reserved	0x0
EXT_INT28_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT28[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.3.107 EXT_INT29CON

- Address = 0x1100_0720, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31]	-	Reserved	0x0
EXT_INT29_CON[7]	[30:28]	RW	Setting the signaling method of EXT_INT29[7] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[27]	-	Reserved	0x0
EXT_INT29_CON[6]	[26:24]	RW	Setting the signaling method of EXT_INT29[6] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[23]	-	Reserved	0x0
EXT_INT29_CON[5]	[22:20]	RW	Setting the signaling method of EXT_INT29[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
EXT_INT29_CON[4]	[18:16]	RW	Setting the signaling method of EXT_INT29[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
EXT_INT29_CON[3]	[14:12]	RW	Setting the signaling method of EXT_INT29[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[11]	-	Reserved	0x0
EXT_INT29_CON[2]	[10:8]	RW	Setting the signaling method of EXT_INT29[2]	0x0

Name	Bit	Type	Description	Reset Value
			0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	
RSVD	[7]	-	Reserved	0x0
EXT_INT29_CON[1]	[6:4]	RW	Setting the signaling method of EXT_INT29[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
EXT_INT29_CON[0]	[2:0]	RW	Setting the signaling method of EXT_INT29[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.3.108 EXT_INT21_FLTCON0

- Address = 0x1100_0800, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN1[3]	[31]	RW	Filter Enable for EXT_INT21[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH1[3]	[30:24]	RW	Filtering width of EXT_INT21[3]	0x00
FLTEN1[2]	[23]	RW	Filter Enable for EXT_INT21[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH1[2]	[22:16]	RW	Filtering width of EXT_INT21[2]	0x00
FLTEN1[1]	[15]	RW	Filter Enable for EXT_INT21[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH1[1]	[14:8]	RW	Filtering width of EXT_INT21[1]	0x00
FLTEN1[0]	[7]	RW	Filter Enable for EXT_INT21[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH1[0]	[6:0]	RW	Filtering width of EXT_INT21[0]	0x00

4.4.3.109 EXT_INT21_FLTCON1

- Address = 0x1100_0804, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN1[7]	[31]	RW	Filter Enable for EXT_INT21[7] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH1[7]	[30:24]	RW	Filtering width of EXT_INT21[7]	0x00
FLTEN1[6]	[23]	RW	Filter Enable for EXT_INT21[6] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH1[6]	[22:16]	RW	Filtering width of EXT_INT21[6]	0x00
FLTEN1[5]	[15]	RW	Filter Enable for EXT_INT21[5] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH1[5]	[14:8]	RW	Filtering width of EXT_INT21[5]	0x00
FLTEN1[4]	[7]	RW	Filter Enable for EXT_INT21[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH1[4]	[6:0]	RW	Filtering width of EXT_INT21[4]	0x00

4.4.3.110 EXT_INT22_FLTCON0

- Address = 0x1100_0808, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN2[3]	[31]	RW	Filter Enable for EXT_INT22[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH2[3]	[30:24]	RW	Filtering width of EXT_INT22[3]	0x00
FLTEN2[2]	[23]	RW	Filter Enable for EXT_INT22[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH2[2]	[22:16]	RW	Filtering width of EXT_INT22[2]	0x00
FLTEN2[1]	[15]	RW	Filter Enable for EXT_INT22[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH2[1]	[14:8]	RW	Filtering width of EXT_INT22[1]	0x00
FLTEN2[0]	[7]	RW	Filter Enable for EXT_INT22[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH2[0]	[6:0]	RW	Filtering width of EXT_INT22[0]	0x00

4.4.3.111 EXT_INT22_FLTCON1

- Address = 0x1100_080C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x000000
FLTEN2[4]	[7]	RW	Filter Enable for EXT_INT22[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH2[4]	[6:0]	RW	Filtering width of EXT_INT22[4]	0x00

4.4.3.112 EXT_INT23_FLTCON0

- Address = 0x1100_0810, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN3[3]	[31]	RW	Filter Enable for EXT_INT23[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH3[3]	[30:24]	RW	Filtering width of EXT_INT23[3]	0x00
FLTEN3[2]	[23]	RW	Filter Enable for EXT_INT23[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH3[2]	[22:16]	RW	Filtering width of EXT_INT23[2]	0x00
FLTEN3[1]	[15]	RW	Filter Enable for EXT_INT23[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH3[1]	[14:8]	RW	Filtering width of EXT_INT23[1]	0x00
FLTEN3[0]	[7]	RW	Filter Enable for EXT_INT23[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH3[0]	[6:0]	RW	Filtering width of EXT_INT23[0]	0x00

4.4.3.113 EXT_INT23_FLTCON1

- Address = 0x1100_0814, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	-	Reserved	0x00
FLTEN3[6]	[23]	RW	Filter Enable for EXT_INT23[6] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH3[6]	[22:16]	RW	Filtering width of EXT_INT23[6]	0x00
FLTEN3[5]	[15]	RW	Filter Enable for EXT_INT23[5] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH3[5]	[14:8]	RW	Filtering width of EXT_INT23[5]	0x00
FLTEN3[4]	[7]	RW	Filter Enable for EXT_INT23[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH3[4]	[6:0]	RW	Filtering width of EXT_INT23[4]	0x00

4.4.3.114 EXT_INT24_FLTCON0

- Address = 0x1100_0818, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN4[3]	[31]	RW	Filter Enable for EXT_INT24[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH4[3]	[30:24]	RW	Filtering width of EXT_INT24[3]	0x00
FLTEN4[2]	[23]	RW	Filter Enable for EXT_INT24[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH4[2]	[22:16]	RW	Filtering width of EXT_INT24[2]	0x00
FLTEN4[1]	[15]	RW	Filter Enable for EXT_INT24[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH4[1]	[14:8]	RW	Filtering width of EXT_INT24[1]	0x00
FLTEN4[0]	[7]	RW	Filter Enable for EXT_INT24[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH4[0]	[6:0]	RW	Filtering width of EXT_INT24[0]	0x00

4.4.3.115 EXT_INT24_FLTCON1

- Address = 0x1100_081C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	-	Reserved	0x00
FLTEN4[6]	[23]	RW	Filter Enable for EXT_INT24[6] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH4[6]	[22:16]	RW	Filtering width of EXT_INT24[6]	0x00
FLTEN4[5]	[15]	RW	Filter Enable for EXT_INT24[5] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH4[5]	[14:8]	RW	Filtering width of EXT_INT24[5]	0x00
FLTEN4[4]	[7]	RW	Filter Enable for EXT_INT24[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH4[4]	[6:0]	RW	Filtering width of EXT_INT24[4]	0x00

4.4.3.116 EXT_INT25_FLTCON0

- Address = 0x1100_0820, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN5[3]	[31]	RW	Filter Enable for EXT_INT25[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH5[3]	[30:24]	RW	Filtering width of EXT_INT25[3]	0x00
FLTEN5[2]	[23]	RW	Filter Enable for EXT_INT25[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH5[2]	[22:16]	RW	Filtering width of EXT_INT25[2]	0x00
FLTEN5[1]	[15]	RW	Filter Enable for EXT_INT25[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH5[1]	[14:8]	RW	Filtering width of EXT_INT25[1]	0x00
FLTEN5[0]	[7]	RW	Filter Enable for EXT_INT25[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH5[0]	[6:0]	RW	Filtering width of EXT_INT25[0]	0x00

4.4.3.117 EXT_INT25_FLTCON1

- Address = 0x1100_0824, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	-	Reserved	0x00
FLTEN5[6]	[23]	RW	Filter Enable for EXT_INT25[6] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH5[6]	[22:16]	RW	Filtering width of EXT_INT25[6]	0x00
FLTEN5[5]	[15]	RW	Filter Enable for EXT_INT25[5] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH5[5]	[14:8]	RW	Filtering width of EXT_INT25[5]	0x00
FLTEN5[4]	[7]	RW	Filter Enable for EXT_INT25[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH5[4]	[6:0]	RW	Filtering width of EXT_INT25[4]	0x00

4.4.3.118 EXT_INT26_FLTCON0

- Address = 0x1100_0828, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN6[3]	[31]	RW	Filter Enable for EXT_INT26[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH6[3]	[30:24]	RW	Filtering width of EXT_INT26[3]	0x00
FLTEN6[2]	[23]	RW	Filter Enable for EXT_INT26[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH6[2]	[22:16]	RW	Filtering width of EXT_INT26[2]	0x00
FLTEN6[1]	[15]	RW	Filter Enable for EXT_INT26[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH6[1]	[14:8]	RW	Filtering width of EXT_INT26[1]	0x00
FLTEN6[0]	[7]	RW	Filter Enable for EXT_INT26[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH6[0]	[6:0]	RW	Filtering width of EXT_INT26[0]	0x00

4.4.3.119 EXT_INT26_FLTCON1

- Address = 0x1100_082C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	-	Reserved	0x00
FLTEN6[6]	[23]	RW	Filter Enable for EXT_INT26[6] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH6[6]	[22:16]	RW	Filtering width of EXT_INT26[6]	0x00
FLTEN6[5]	[15]	RW	Filter Enable for EXT_INT26[5] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH6[5]	[14:8]	RW	Filtering width of EXT_INT26[5]	0x00
FLTEN6[4]	[7]	RW	Filter Enable for EXT_INT26[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH6[4]	[6:0]	RW	Filtering width of EXT_INT26[4]	0x00

4.4.3.120 EXT_INT27_FLTCON0

- Address = 0x1100_0830, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN7[3]	[31]	RW	Filter Enable for EXT_INT27[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH7[3]	[30:24]	RW	Filtering width of EXT_INT27[3]	0x00
FLTEN7[2]	[23]	RW	Filter Enable for EXT_INT27[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH7[2]	[22:16]	RW	Filtering width of EXT_INT27[2]	0x00
FLTEN7[1]	[15]	RW	Filter Enable for EXT_INT27[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH7[1]	[14:8]	RW	Filtering width of EXT_INT27[1]	0x00
FLTEN7[0]	[7]	RW	Filter Enable for EXT_INT27[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH7[0]	[6:0]	RW	Filtering width of EXT_INT27[0]	0x00

4.4.3.121 EXT_INT27_FLTCON1

- Address = 0x1100_0834, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN7[7]	[31]	RW	Filter Enable for EXT_INT27[7] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH7[7]	[30:24]	RW	Filtering width of EXT_INT27[7]	0x00
FLTEN7[6]	[23]	RW	Filter Enable for EXT_INT27[6] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH7[6]	[22:16]	RW	Filtering width of EXT_INT27[6]	0x00
FLTEN7[5]	[15]	RW	Filter Enable for EXT_INT27[5] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH7[5]	[14:8]	RW	Filtering width of EXT_INT27[5]	0x00
FLTEN7[4]	[7]	RW	Filter Enable for EXT_INT27[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH7[4]	[6:0]	RW	Filtering width of EXT_INT27[4]	0x00

4.4.3.122 EXT_INT28_FLTCON0

- Address = 0x1100_0838, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	-	Reserved	0x00
FLTEN8[2]	[23]	RW	Filter Enable for EXT_INT28[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH8[2]	[22:16]	RW	Filtering width of EXT_INT28[2]	0x00
FLTEN8[1]	[15]	RW	Filter Enable for EXT_INT28[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH8[1]	[14:8]	RW	Filtering width of EXT_INT28[1]	0x00
FLTEN8[0]	[7]	RW	Filter Enable for EXT_INT28[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH8[0]	[6:0]	RW	Filtering width of EXT_INT28[0]	0x00

4.4.3.123 EXT_INT28_FLTCON1

- Address = 0x1100_083C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:0]	-	Reserved	0x00000000

4.4.3.124 EXT_INT29_FLTCON0

- Address = 0x1100_0840, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN9[3]	[31]	RW	Filter Enable for EXT_INT29[3] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH9[3]	[30:24]	RW	Filtering width of EXT_INT29[3]	0x00
FLTEN9[2]	[23]	RW	Filter Enable for EXT_INT29[2] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH9[2]	[22:16]	RW	Filtering width of EXT_INT29[2]	0x00
FLTEN9[1]	[15]	RW	Filter Enable for EXT_INT29[1] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH9[1]	[14:8]	RW	Filtering width of EXT_INT29[1]	0x00
FLTEN9[0]	[7]	RW	Filter Enable for EXT_INT29[0] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH9[0]	[6:0]	RW	Filtering width of EXT_INT29[0]	0x00

4.4.3.125 EXT_INT29_FLTCON1

- Address = 0x1100_0844, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
FLTEN9[7]	[31]	RW	Filter Enable for EXT_INT29[7] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH9[7]	[30:24]	RW	Filtering width of EXT_INT29[7]	0x00
FLTEN9[6]	[23]	RW	Filter Enable for EXT_INT29[6] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH9[6]	[22:16]	RW	Filtering width of EXT_INT29[6]	0x00
FLTEN9[5]	[15]	RW	Filter Enable for EXT_INT29[5] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH9[5]	[14:8]	RW	Filtering width of EXT_INT29[5]	0x00
FLTEN9[4]	[7]	RW	Filter Enable for EXT_INT29[4] 0x0 = Disables 0x1 = Enabled	0x0
FLTWIDTH9[4]	[6:0]	RW	Filtering width of EXT_INT29[4]	0x00

4.4.3.126 EXT_INT21_MASK

- Address = 0x1100_0900, Reset Value = 0x0000_00FF

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x0000000
EXT_INT21_MASK[7]	[7]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT21_MASK[6]	[6]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT21_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT21_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT21_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT21_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT21_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT21_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.3.127 EXT_INT22_MASK

- Address = 0x1100_0904, Reset Value = 0x0000_001F

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	-	Reserved	0x00000000
EXT_INT22_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT22_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT22_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT22_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT22_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.3.128 EXT_INT23_MASK

- Address = 0x1100_0908, Reset Value = 0x0000_007F

Name	Bit	Type	Description	Reset Value
RSVD	[31:7]	-	Reserved	0x00000000
EXT_INT23_MASK[6]	[6]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT23_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT23_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT23_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT23_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT23_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT23_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.3.129 EXT_INT24_MASK

- Address = 0x1100_090C, Reset Value = 0x0000_007F

Name	Bit	Type	Description	Reset Value
RSVD	[31:7]	-	Reserved	0x00000000
EXT_INT24_MASK[6]	[6]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT24_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT24_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT24_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT24_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT24_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT24_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.3.130 EXT_INT25_MASK

- Address = 0x1100_0910, Reset Value = 0x0000_007F

Name	Bit	Type	Description	Reset Value
RSVD	[31:7]	-	Reserved	0x00000000
EXT_INT25_MASK[6]	[6]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT25_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT25_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT25_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT25_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT25_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT25_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.3.131 EXT_INT26_MASK

- Address = 0x1100_0914, Reset Value = 0x0000_007F

Name	Bit	Type	Description	Reset Value
RSVD	[31:7]	-	Reserved	0x00000000
EXT_INT26_MASK[6]	[6]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT26_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT26_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT26_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT26_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT26_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT26_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.3.132 EXT_INT27_MASK

- Address = 0x1100_0918, Reset Value = 0x0000_00FF

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x000000
EXT_INT27_MASK[7]	[7]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT27_MASK[6]	[6]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT27_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT27_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT27_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT27_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT27_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT27_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.3.133 EXT_INT28_MASK

- Address = 0x1100_091C, Reset Value = 0x0000_0007

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
EXT_INT28_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT28_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT28_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.3.134 EXT_INT29_MASK

- Address = 0x1100_0920, Reset Value = 0x0000_00FF

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x000000
EXT_INT29_MASK[7]	[7]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT29_MASK[6]	[6]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT29_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT29_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT29_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT29_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT29_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
EXT_INT29_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.3.135 EXT_INT21_PEND

- Address = 0x1100_0A00, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x000000
EXT_INT21_PEND[7]	[7]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT21_PEND[6]	[6]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT21_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT21_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT21_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT21_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT21_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT21_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.3.136 EXT_INT22_PEND

- Address = 0x1100_0A04, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	-	Reserved	0x00000000
EXT_INT22_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT22_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT22_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT22_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT22_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.3.137 EXT_INT23_PEND

- Address = 0x1100_0A08, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:7]	-	Reserved	0x00000000
EXT_INT23_PEND[6]	[6]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT23_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT23_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT23_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT23_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT23_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT23_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.3.138 EXT_INT24_PEND

- Address = 0x1100_0A0C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:7]	-	Reserved	0x00000000
EXT_INT24_PEND[6]	[6]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT24_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT24_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT24_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT24_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT24_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT24_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.3.139 EXT_INT25_PEND

- Address = 0x1100_0A10, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:7]	-	Reserved	0x00000000
EXT_INT25_PEND[6]	[6]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT25_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT25_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT25_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT25_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT25_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT25_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.3.140 EXT_INT26_PEND

- Address = 0x1100_0A14, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:7]	-	Reserved	0x00000000
EXT_INT26_PEND[6]	[6]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT26_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT26_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT26_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT26_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT26_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT26_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.3.141 EXT_INT27_PEND

- Address = 0x1100_0A18, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x00000000
EXT_INT27_PEND[7]	[7]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT27_PEND[6]	[6]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT27_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT27_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT27_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT27_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT27_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT27_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.3.142 EXT_INT28_PEND

- Address = 0x1100_0A1C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
EXT_INT28_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT28_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT28_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.3.143 EXT_INT29_PEND

- Address = 0x1100_0A20, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x000000
EXT_INT29_PEND[7]	[7]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT29_PEND[6]	[6]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT29_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT29_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT29_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT29_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT29_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
EXT_INT29_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.3.144 EXT_INT_SERVICE_XB

- Address = 0x1100_0B08, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	RW	Reserved	0x00000000
SVC_Group_Num	[7:3]	RW	EXT_INT Service group number	0x00
SVC_Num	[2:0]	RW	Interrupt number to be serviced	0x0

4.4.3.145 EXT_INT_SERVICE_PEND_XB

- Address = 0x1100_0B0C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	RW	Reserved	0x00000000
SVC_PEND	[7:0]	RW	0x0 = Not occur 0x1 = Occur interrupt	0x00

4.4.3.146 EXT_INT_GRPFIXPRI_XB

- Address = 0x1100_0B10, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	-	Reserved	0x00000000
Highest_GRP_NUM	[3:0]	RW	Group number of the highest priority when fixed group priority mode: 0 (EXT_INT21) – 8 (EXT_INT29)	0x00

4.4.3.147 EXT_INT21_FIXPRI

- Address = 0x1100_0B14, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 1 (EXT_INT21) when fixed priority mode: 0 – 7	0x0

4.4.3.148 EXT_INT22_FIXPRI

- Address = 0x1100_0B18, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 2 (EXT_INT22) when fixed priority mode: 0 – 7	0x0

4.4.3.149 EXT_INT23_FIXPRI

- Address = 0x1100_0B1C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 3 (EXT_INT23) when fixed priority mode: 0 – 7	0x0

4.4.3.150 EXT_INT24_FIXPRI

- Address = 0x1100_0B20, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 4 (EXT_INT24) when fixed priority mode: 0 – 7	0x0

4.4.3.151 EXT_INT25_FIXPRI

- Address = 0x1100_0B24, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 5 (EXT_INT25) when fixed priority mode: 0 – 7	0x0

4.4.3.152 EXT_INT26_FIXPRI

- Address = 0x1100_0B28, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 6 (EXT_INT26) when fixed priority mode: 0 – 7	0x0

4.4.3.153 EXT_INT27_FIXPRI

- Address = 0x1100_0B2C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 7 (EXT_INT27) when fixed priority mode: 0 – 7	0x0

4.4.3.154 EXT_INT28_FIXPRI

- Address = 0x1100_0B30, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 8 (EXT_INT28) when fixed priority mode: 0 – 7	0x0

4.4.3.155 EXT_INT29_FIXPRI

- Address = 0x1100_0B34, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x00000000
Highest_EINT_NUM	[2:0]	RW	Interrupt number of the highest priority in External Interrupt Group 9 (EXT_INT29) when fixed priority mode: 0 – 7	0x0

4.4.3.156 GPX0CON

- Address = 0x1100_0C00, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPX0CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = AUD_TCK 0x4 = GNSS_TCK 0x5 – 0xE = Reserved 0xF = WAKEUP_INT0[0]	0x00
GPX0CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = AUD_TMS 0x4 = GNSS_TMS 0x5 – 0xE = Reserved 0xF = WAKEUP_INT0[1]	0x00
GPX0CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = AUD_TDI 0x4 = GNSS_TDI 0x5 – 0xE = Reserved 0xF = WAKEUP_INT0[2]	0x00
GPX0CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = AUD_TDO 0x4 = GNSS_TDO 0x5 – 0xE = Reserved 0xF = WAKEUP_INT0[3]	0x00
GPX0CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = AUD_TRSTn 0x4 = GNSS_TRSTn 0x5 – 0xE = Reserved 0xF = WAKEUP_INT0[4]	0x00
GPX0CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 – 0xE = Reserved 0xF = WAKEUP_INT0[5]	0x00
GPX0CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 – 0xE = Reserved 0xF = WAKEUP_INT0[6]	0x00
GPX0CON[7]	[31:28]	RW	0x0 = Input 0x1 = Output	0x00

Name	Bit	Type	Description	Reset Value
			0x2 – 0xE = Reserved 0xF = WAKEUP_INT0[7]	

4.4.3.157 GPX0DAT

- Address = 0x1100_0C04, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPX0DAT[7:0]	[7:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.158 GPX0PUD

- Address = 0x1100_0C08, Reset Value = 0x5555

Name	Bit	Type	Description	Reset Value
GPX0PUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x5555

4.4.3.159 GPX0DRV

- Address = 0x1100_0C0C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
GPX0DRV[n]	[2n+1:2n] n=0 – 7	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.3.160 GPX1CON

- Address = 0x1100_0C20, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPX1CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_COL[0] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT1[0]	0x00
GPX1CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_COL[1] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT1[1]	0x00
GPX1CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_COL[2] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT1[2]	0x00
GPX1CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_COL[3] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT1[3]	0x00
GPX1CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_COL[4] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT1[4]	0x00
GPX1CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_COL[5] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT1[5]	0x00
GPX1CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_COL[6] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT1[6]	0x00
GPX1CON[7]	[31:28]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved	0x00

Name	Bit	Type	Description	Reset Value
			0x3 = KP_COL[7] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT1[7]	

4.4.3.161 GPX1DAT

- Address = 0x1100_0C24, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPX1DAT[7:0]	[7:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.162 GPX1PUD

- Address = 0x1100_0C28, Reset Value = 0x5555

Name	Bit	Type	Description	Reset Value
GPX1PUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x5555

4.4.3.163 GPX1DRV

- Address = 0x1100_0C2C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
GPX1DRV[n]	[2n+1:2n] n=0 – 7	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.3.164 GPX2CON

- Address = 0x1100_0C40, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPX2CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_ROW[0] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT2[0]	0x00
GPX2CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_ROW[1] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT2[1]	0x00
GPX2CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_ROW[2] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT2[2]	0x00
GPX2CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_ROW[3] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT2[3]	0x00
GPX2CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_ROW[4] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT2[4]	0x00
GPX2CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_ROW[5] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT2[5]	0x00
GPX2CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_ROW[6] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT2[6]	0x00
GPX2CON[7]	[31:28]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved	0x00

Name	Bit	Type	Description	Reset Value
			0x3 = KP_ROW[7] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT2[7]	

4.4.3.165 GPX2DAT

- Address = 0x1100_0C44, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPX2DAT[7:0]	[7:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.166 GPX2PUD

- Address = 0x1100_0C48, Reset Value = 0x5555

Name	Bit	Type	Description	Reset Value
GPX2PUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x5555

4.4.3.167 GPX2DRV

- Address = 0x1100_0C4C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
GPX2DRV[n]	[2n+1:2n] n=0 – 7	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.3.168 GPX3CON

- Address = 0x1100_0C60, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPX3CON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_ROW[8] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT3[0]	0x00
GPX3CON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_ROW[9] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT3[1]	0x00
GPX3CON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_ROW[10] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT3[2]	0x00
GPX3CON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_ROW[11] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT3[3]	0x00
GPX3CON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_ROW[12] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT3[4]	0x00
GPX3CON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = KP_ROW[13] 0x4 – 0xE = Reserved 0xF = WAKEUP_INT3[5]	0x00
GPX3CON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = HDMI_CEC 0x4 – 0xE = Reserved 0xF = WAKEUP_INT3[6]	0x00
GPX3CON[7]	[31:28]	RW	0x0 = Input 0x1 = Output 0x2 = Reserved 0x3 = HDMI_HPD	0x00

Name	Bit	Type	Description	Reset Value
			0x4 – 0xE = Reserved 0xF = WAKEUP_INT3[7]	

4.4.3.169 GPX3DAT

- Address = 0x1100_0C64, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPX3DAT[7:0]	[7:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.3.170 GPX3PUD

- Address = Base Address + 0C68, Reset Value = 0x5555

Name	Bit	Type	Description	Reset Value
GPX3PUD[n]	[2n+1:2n] n=0 – 7	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x5555

4.4.3.171 GPX3DRV

- Address = 0x1100_0C6C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
	[23:16]	RW	Reserved(should be zero)	0x00
GPX3DRV[n]	[2n+1:2n] n=0 – 7	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.3.172 WAKEUP_INT0CON

- Address = 0x1100_0E00, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31]	-	Reserved	0x0
WAKEUP_INT0_CON[7]	[30:28]	RW	Setting the signaling method of WAKEUP_INT0[7] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[27]	-	Reserved	0x0
WAKEUP_INT0_CON[6]	[26:24]	RW	Setting the signaling method of WAKEUP_INT0[6] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[23]	-	Reserved	0x0
WAKEUP_INT0_CON[5]	[22:20]	RW	Setting the signaling method of WAKEUP_INT0[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
WAKEUP_INT0_CON[4]	[18:16]	RW	Setting the signaling method of WAKEUP_INT0[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
WAKEUP_INT0_CON[3]	[14:12]	RW	Setting the signaling method of WAKEUP_INT0[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered	0x0

Name	Bit	Type	Description	Reset Value
			0x4 = Both edge triggered 0x5 – 0x7 = Reserved	
RSVD	[11]	-	Reserved	0x0
WAKEUP_INT0_CON[2]	[10:8]	RW	Setting the signaling method of WAKEUP_INT0[2] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	-	Reserved	0x0
WAKEUP_INT0_CON[1]	[6:4]	RW	Setting the signaling method of WAKEUP_INT0[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
WAKEUP_INT0_CON[0]	[2:0]	RW	Setting the signaling method of WAKEUP_INT0[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.3.173 WAKEUP_INT1CON

- Address = 0x1100_0E04, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31]	-	Reserved	0x0
WAKEUP_INT1_CON[7]	[30:28]	RW	Setting the signaling method of WAKEUP_INT1[7] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[27]	-	Reserved	0x0
WAKEUP_INT1_CON[6]	[26:24]	RW	Setting the signaling method of WAKEUP_INT1[6] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[23]	-	Reserved	0x0
WAKEUP_INT1_CON[5]	[22:20]	RW	Setting the signaling method of WAKEUP_INT1[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
WAKEUP_INT1_CON[4]	[18:16]	RW	Setting the signaling method of WAKEUP_INT1[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
WAKEUP_INT1_CON[3]	[14:12]	RW	Setting the signaling method of WAKEUP_INT1[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered	0x0

Name	Bit	Type	Description	Reset Value
			0x4 = Both edge triggered 0x5 – 0x7 = Reserved	
RSVD	[11]	-	Reserved	0x0
WAKEUP_INT1_CON[2]	[10:8]	RW	Setting the signaling method of WAKEUP_INT1[2] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	-	Reserved	0x0
WAKEUP_INT1_CON[1]	[6:4]	RW	Setting the signaling method of WAKEUP_INT1[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
WAKEUP_INT1_CON[0]	[2:0]	RW	Setting the signaling method of WAKEUP_INT1[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.3.174 WAKEUP_INT2CON

- Address = 0x1100_0E08, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31]	-	Reserved	0x0
WAKEUP_INT2_CON[7]	[30:28]	RW	Setting the signaling method of WAKEUP_INT2[7] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 ~ 0x7 = Reserved	0x0
RSVD	[27]	-	Reserved	0x0
WAKEUP_INT2_CON[6]	[26:24]	RW	Setting the signaling method of WAKEUP_INT2[6] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 ~ 0x7 = Reserved	0x0
RSVD	[23]	-	Reserved	0x0
WAKEUP_INT2_CON[5]	[22:20]	RW	Setting the signaling method of WAKEUP_INT2[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 ~ 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
WAKEUP_INT2_CON[4]	[18:16]	RW	Setting the signaling method of WAKEUP_INT2[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 ~ 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
WAKEUP_INT2_CON[3]	[14:12]	RW	Setting the signaling method of WAKEUP_INT2[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered	0x0

Name	Bit	Type	Description	Reset Value
			0x5 – 0x7 = Reserved	
RSVD	[11]	-	Reserved	0x0
WAKEUP_INT2_CON[2]	[10:8]	RW	Setting the signaling method of WAKEUP_INT2[2] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	-	Reserved	0x0
WAKEUP_INT2_CON[1]	[6:4]	RW	Setting the signaling method of WAKEUP_INT2[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
WAKEUP_INT2_CON[0]	[2:0]	RW	Setting the signaling method of WAKEUP_INT2[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.3.175 WAKEUP_INT3CON

- Address = 0x1100_0E0C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31]	-	Reserved	0x0
WAKEUP_INT3_CON[7]	[30:28]	RW	Setting the signaling method of WAKEUP_INT3[7] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[27]	-	Reserved	0x0
WAKEUP_INT3_CON[6]	[26:24]	RW	Setting the signaling method of WAKEUP_INT3[6] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[23]	-	Reserved	0x0
WAKEUP_INT3_CON[5]	[22:20]	RW	Setting the signaling method of WAKEUP_INT3[5] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[19]	-	Reserved	0x0
WAKEUP_INT3_CON[4]	[18:16]	RW	Setting the signaling method of WAKEUP_INT3[4] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[15]	-	Reserved	0x0
WAKEUP_INT3_CON[3]	[14:12]	RW	Setting the signaling method of WAKEUP_INT3[3] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered	0x0

Name	Bit	Type	Description	Reset Value
			0x4 = Both edge triggered 0x5 – 0x7 = Reserved	
RSVD	[11]	-	Reserved	0x0
WAKEUP_INT3_CON[2]	[10:8]	RW	Setting the signaling method of WAKEUP_INT3[2] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[7]	-	Reserved	0x0
WAKEUP_INT3_CON[1]	[6:4]	RW	Setting the signaling method of WAKEUP_INT3[1] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0
RSVD	[3]	-	Reserved	0x0
WAKEUP_INT3_CON[0]	[2:0]	RW	Setting the signaling method of WAKEUP_INT3[0] 0x0 = Low level 0x1 = High level 0x2 = Falling edge triggered 0x3 = Rising edge triggered 0x4 = Both edge triggered 0x5 – 0x7 = Reserved	0x0

4.4.3.176 WAKEUP_INT0_FLTCON0

- Address = 0x1100_0E80, Reset Value = 0x8080_8080

Name	Bit	Type	Description	Reset Value
FLTEN10[3]	[31]	RW	Filter Enable for WAKEUP_INT0[3] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL10[3]	[30]	RW	Filter Selection for WAKEUP_INT0[3] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH10[3]	[29:24]	RW	Filtering width of WAKEUP_INT0[3] This value is valid when FLTSEL10 (of WAKEUP_INT0) is 0x1.	0x00
FLTEN10[2]	[23]	RW	Filter Enable for WAKEUP_INT0[2] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL10[2]	[22]	RW	Filter Selection for WAKEUP_INT0[2] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH10[2]	[21:16]	RW	Filtering width of WAKEUP_INT0[2] This value is valid when FLTSEL10 (of WAKEUP_INT0) is 0x1.	0x00
FLTEN10[1]	[15]	RW	Filter Enable for WAKEUP_INT0[1] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL10[1]	[14]	RW	Filter Selection for WAKEUP_INT0[1] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH10[1]	[13:8]	RW	Filtering width of WAKEUP_INT0[1] This value is valid when FLTSEL10 (of WAKEUP_INT0) is 0x1.	0x00
FLTEN10[0]	[7]	RW	Filter Enable for WAKEUP_INT0[0] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL10[0]	[6]	RW	Filter Selection for WAKEUP_INT0[0] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH10[0]	[5:0]	RW	Filtering width of WAKEUP_INT0[0] This value is valid when FLTSEL10 (of WAKEUP_INT0) is 0x1.	0x00

4.4.3.177 WAKEUP_INT0_FLTCON1

- Address = 0x1100_0E84, Reset Value = 0x8080_8080

Name	Bit	Type	Description	Reset Value
FLTEN10[7]	[31]	RW	Filter Enable for WAKEUP_INT0[7] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL10[7]	[30]	RW	Filter Selection for WAKEUP_INT0[7] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH10[7]	[29:24]	RW	Filtering width of WAKEUP_INT0[7] This value is valid when FLTSEL10 (of WAKEUP_INT0) is 0x1.	0x00
FLTEN10[6]	[23]	RW	Filter Enable for WAKEUP_INT0[6] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL10[6]	[22]	RW	Filter Selection for WAKEUP_INT0[6] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH10[6]	[21:16]	RW	Filtering width of WAKEUP_INT0[6] This value is valid when FLTSEL10 (of WAKEUP_INT0) is 0x1.	0x00
FLTEN10[5]	[15]	RW	Filter Enable for WAKEUP_INT0[5] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL10[5]	[14]	RW	Filter Selection for WAKEUP_INT0[5] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH10[5]	[13:8]	RW	Filtering width of WAKEUP_INT0[5] This value is valid when FLTSEL10 (of WAKEUP_INT0) is 0x1.	0x00
FLTEN10[4]	[7]	RW	Filter Enable for WAKEUP_INT0[4] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL10[4]	[6]	RW	Filter Selection for WAKEUP_INT0[4] 0x0 = Delay filter 0x1 = Digital filter(Clock count)	0x0
FLTWIDTH10[4]	[5:0]	RW	Filtering width of WAKEUP_INT0[4] This value is valid when FLTSEL10 (of WAKEUP_INT0) is 0x1.	0x00

4.4.3.178 WAKEUP_INT1_FLTCON0

- Address = 0x1100_0E88, Reset Value = 0x8080_8080

Name	Bit	Type	Description	Reset Value
FLTEN11[3]	[31]	RW	Filter Enable for WAKEUP_INT1[3] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL11[3]	[30]	RW	Filter Selection for WAKEUP_INT1[3] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH11[3]	[29:24]	RW	Filtering width of WAKEUP_INT1[3] This value is valid when FLTSEL11 (of WAKEUP_INT1) is 0x1.	0x00
FLTEN11[2]	[23]	RW	Filter Enable for WAKEUP_INT1[2] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL11[2]	[22]	RW	Filter Selection for WAKEUP_INT1[2] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH11[2]	[21:16]	RW	Filtering width of WAKEUP_INT1[2] This value is valid when FLTSEL11 (of WAKEUP_INT1) is 0x1.	0x00
FLTEN11[1]	[15]	RW	Filter Enable for WAKEUP_INT1[1] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL11[1]	[14]	RW	Filter Selection for WAKEUP_INT1[1] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH11[1]	[13:8]	RW	Filtering width of WAKEUP_INT1[1] This value is valid when FLTSEL11 (of WAKEUP_INT1) is 0x1.	0x00
FLTEN11[0]	[7]	RW	Filter Enable for WAKEUP_INT1[0] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL11[0]	[6]	RW	Filter Selection for WAKEUP_INT1[0] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH11[0]	[5:0]	RW	Filtering width of WAKEUP_INT1[0] This value is valid when FLTSEL11 (of WAKEUP_INT1) is 0x1.	0x00

4.4.3.179 WAKEUP_INT1_FLTCON1

- Address = 0x1100_0E8C, Reset Value = 0x8080_8080

Name	Bit	Type	Description	Reset Value
FLTEN11[7]	[31]	RW	Filter Enable for WAKEUP_INT1[7] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL11[7]	[30]	RW	Filter Selection for WAKEUP_INT1[7] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH11[7]	[29:24]	RW	Filtering width of WAKEUP_INT1[7] This value is valid when FLTSEL11 (of WAKEUP_INT1) is 0x1.	0x00
FLTEN11[6]	[23]	RW	Filter Enable for WAKEUP_INT1[6] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL11[6]	[22]	RW	Filter Selection for WAKEUP_INT1[6] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH11[6]	[21:16]	RW	Filtering width of WAKEUP_INT1[6] This value is valid when FLTSEL11 (of WAKEUP_INT1) is 0x1.	0x00
FLTEN11[5]	[15]	RW	Filter Enable for WAKEUP_INT1[5] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL11[5]	[14]	RW	Filter Selection for WAKEUP_INT1[5] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH11[5]	[13:8]	RW	Filtering width of WAKEUP_INT1[5] This value is valid when FLTSEL11 (of WAKEUP_INT1) is 0x1.	0x00
FLTEN11[4]	[7]	RW	Filter Enable for WAKEUP_INT1[4] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL11[4]	[6]	RW	Filter Selection for WAKEUP_INT1[4] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH11[4]	[5:0]	RW	Filtering width of WAKEUP_INT1[4] This value is valid when FLTSEL11 (of WAKEUP_INT1) is 0x1.	0x00

4.4.3.180 WAKEUP_INT2_FLTCON0

- Address = 0x1100_0E90, Reset Value = 0x8080_8080

Name	Bit	Type	Description	Reset Value
FLTEN12[3]	[31]	RW	Filter Enable for WAKEUP_INT2[3] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL12[3]	[30]	RW	Filter Selection for WAKEUP_INT2[3] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH12[3]	[29:24]	RW	Filtering width of WAKEUP_INT2[3] This value is valid when FLTSEL12 (of WAKEUP_INT2) is 0x1.	0x00
FLTEN12[2]	[23]	RW	Filter Enable for WAKEUP_INT2[2] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL12[2]	[22]	RW	Filter Selection for WAKEUP_INT2[2] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH12[2]	[21:16]	RW	Filtering width of WAKEUP_INT2[2] This value is valid when FLTSEL12 (of WAKEUP_INT2) is 0x1.	0x00
FLTEN12[1]	[15]	RW	Filter Enable for WAKEUP_INT2[1] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL12[1]	[14]	RW	Filter Selection for WAKEUP_INT2[1] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH12[1]	[13:8]	RW	Filtering width of WAKEUP_INT2[1] This value is valid when FLTSEL12 (of WAKEUP_INT2) is 0x1.	0x00
FLTEN12[0]	[7]	RW	Filter Enable for WAKEUP_INT2[0] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL12[0]	[6]	RW	Filter Selection for WAKEUP_INT2[0] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH12[0]	[5:0]	RW	Filtering width of WAKEUP_INT2[0] This value is valid when FLTSEL12 (of WAKEUP_INT2) is 0x1.	0x00

4.4.3.181 WAKEUP_INT2_FLTCON1

- Address = 0x1100_0E94, Reset Value = 0x8080_8080

Name	Bit	Type	Description	Reset Value
FLTEN12[7]	[31]	RW	Filter Enable for WAKEUP_INT2[7] 0x0 = disables 0x1 = enabled	0x1
FLTSEL12[7]	[30]	RW	Filter Selection for WAKEUP_INT2[7] 0x0 = delay filter 0x1 = digital filter (clock count)	0x0
FLTWIDTH12[7]	[29:24]	RW	Filtering width of WAKEUP_INT2[7] This value is valid when FLTSEL12 (of WAKEUP_INT2) is 0x1.	0x00
FLTEN12[6]	[23]	RW	Filter Enable for WAKEUP_INT2[6] 0x0 = disables 0x1 = enabled	0x1
FLTSEL12[6]	[22]	RW	Filter Selection for WAKEUP_INT2[6] 0x0 = delay filter 0x1 = digital filter (clock count)	0x0
FLTWIDTH12[6]	[21:16]	RW	Filtering width of WAKEUP_INT2[6] This value is valid when FLTSEL12 (of WAKEUP_INT2) is 0x1.	0x00
FLTEN12[5]	[15]	RW	Filter Enable for WAKEUP_INT2[5] 0x0 = disables 0x1 = enabled	0x1
FLTSEL12[5]	[14]	RW	Filter Selection for WAKEUP_INT2[5] 0x0 = delay filter 0x1 = digital filter(clock count)	0x0
FLTWIDTH12[5]	[13:8]	RW	Filtering width of WAKEUP_INT2[5] This value is valid when FLTSEL12 (of WAKEUP_INT2) is 0x1.	0x00
FLTEN12[4]	[7]	RW	Filter Enable for WAKEUP_INT2[4] 0x0 = disables 0x1 = enabled	0x1
FLTSEL12[4]	[6]	RW	Filter Selection for WAKEUP_INT2[4] 0x0 = delay filter 0x1 = digital filter(clock count)	0x0
FLTWIDTH12[4]	[5:0]	RW	Filtering width of WAKEUP_INT2[4] This value is valid when FLTSEL12 (of WAKEUP_INT2) is 0x1.	0x00

4.4.3.182 WAKEUP_INT3_FLTCON0

- Address = 0x1100_0E98, Reset Value = 0x8080_8080

Name	Bit	Type	Description	Reset Value
FLTEN13[3]	[31]	RW	Filter Enable for WAKEUP_INT3[3] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL13[3]	[30]	RW	Filter Selection for WAKEUP_INT3[3] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH13[3]	[29:24]	RW	Filtering width of WAKEUP_INT3[3] This value is valid when FLTSEL13 (of WAKEUP_INT3) is 0x1.	0x00
FLTEN13[2]	[23]	RW	Filter Enable for WAKEUP_INT3[2] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL13[2]	[22]	RW	Filter Selection for WAKEUP_INT3[2] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH13[2]	[21:16]	RW	Filtering width of WAKEUP_INT3[2] This value is valid when FLTSEL13 (of WAKEUP_INT3) is 0x1.	0x00
FLTEN13[1]	[15]	RW	Filter Enable for WAKEUP_INT3[1] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL13[1]	[14]	RW	Filter Selection for WAKEUP_INT3[1] 0x0 = Delay filter 0x1 = Digital filter(Clock count)	0x0
FLTWIDTH13[1]	[13:8]	RW	Filtering width of WAKEUP_INT3[1] This value is valid when FLTSEL13 (of WAKEUP_INT3) is 0x1.	0x00
FLTEN13[0]	[7]	RW	Filter Enable for WAKEUP_INT3[0] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL13[0]	[6]	RW	Filter Selection for WAKEUP_INT3[0] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH13[0]	[5:0]	RW	Filtering width of WAKEUP_INT3[0] This value is valid when FLTSEL13 (of WAKEUP_INT3) is 0x1.	0x00

4.4.3.183 WAKEUP_INT3_FLTCON1

- Address = 0x1100_0E9C, Reset Value = 0x8080_8080

Name	Bit	Type	Description	Reset Value
FLTEN13[7]	[31]	RW	Filter Enable for WAKEUP_INT3[7] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL13[7]	[30]	RW	Filter Selection for WAKEUP_INT3[7] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH13[7]	[29:24]	RW	Filtering width of WAKEUP_INT3[7] This value is valid when FLTSEL13 (of WAKEUP_INT3) is 0x1.	0x00
FLTEN13[6]	[23]	RW	Filter Enable for WAKEUP_INT3[6] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL13[6]	[22]	RW	Filter Selection for WAKEUP_INT3[6] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH13[6]	[21:16]	RW	Filtering width of WAKEUP_INT3[6] This value is valid when FLTSEL13 (of WAKEUP_INT3) is 0x1.	0x00
FLTEN13[5]	[15]	RW	Filter Enable for WAKEUP_INT3[5] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL13[5]	[14]	RW	Filter Selection for WAKEUP_INT3[5] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH13[5]	[13:8]	RW	Filtering width of WAKEUP_INT3[5] This value is valid when FLTSEL13 (of WAKEUP_INT3) is 0x1.	0x00
FLTEN13[4]	[7]	RW	Filter Enable for WAKEUP_INT3[4] 0x0 = Disables 0x1 = Enabled	0x1
FLTSEL13[4]	[6]	RW	Filter Selection for WAKEUP_INT3[4] 0x0 = Delay filter 0x1 = Digital filter (Clock count)	0x0
FLTWIDTH13[4]	[5:0]	RW	Filtering width of WAKEUP_INT3[4] This value is valid when FLTSEL13 (of WAKEUP_INT3) is 0x1.	0x00

4.4.3.184 WAKEUP_INT0_MASK

- Address = 0x1100_0F00, Reset Value = 0x0000_00FF

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	—	Reserved	0x000000
WAKEUP_INT0_MASK[7]	[7]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT0_MASK[6]	[6]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT0_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT0_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT0_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT0_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT0_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT0_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.3.185 WAKEUP_INT1_MASK

- Address = 0x1100_0F04, Reset Value = 0x0000_00FF

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	—	Reserved	0x000000
WAKEUP_INT1_MASK[7]	[7]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT1_MASK[6]	[6]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT1_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT1_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT1_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT1_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT1_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT1_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.3.186 WAKEUP_INT2_MASK

- Address = 0x1100_0F08, Reset Value = 0x0000_00FF

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	—	Reserved	0x000000
WAKEUP_INT2_MASK[7]	[7]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT2_MASK[6]	[6]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT2_MASK[5]	[5]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT2_MASK[4]	[4]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT2_MASK[3]	[3]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT2_MASK[2]	[2]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT2_MASK[1]	[1]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1
WAKEUP_INT2_MASK[0]	[0]	RW	0x0 = Enable Interrupt 0x1 = Masked	0x1

4.4.3.187 WAKEUP_INT3_MASK

- Address = 0x1100_0F0C, Reset Value = 0x0000_00FF

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	—	Reserved	0x000000
WAKEUP_INT3_MASK[7]	[7]	RW	0x0 = Enable Interrupt, 0x1 = Masked	0x1
WAKEUP_INT3_MASK[6]	[6]	RW	0x0 = Enable Interrupt, 0x1 = Masked	0x1
WAKEUP_INT3_MASK[5]	[5]	RW	0x0 = Enable Interrupt, 0x1 = Masked	0x1
WAKEUP_INT3_MASK[4]	[4]	RW	0x0 = Enable Interrupt, 0x1 = Masked	0x1
WAKEUP_INT3_MASK[3]	[3]	RW	0x0 = Enable Interrupt, 0x1 = Masked	0x1
WAKEUP_INT3_MASK[2]	[2]	RW	0x0 = Enable Interrupt, 0x1 = Masked	0x1
WAKEUP_INT3_MASK[1]	[1]	RW	0x0 = Enable Interrupt, 0x1 = Masked	0x1
WAKEUP_INT3_MASK[0]	[0]	RW	0x0 = Enable Interrupt, 0x1 = Masked	0x1

4.4.3.188 WAKEUP_INT0_PEND

- Address = 0x1100_0F40, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	—	Reserved	0x000000
WAKEUP_INT0_PEND[7]	[7]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT0_PEND[6]	[6]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT0_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT0_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT0_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT0_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT0_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT0_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.3.189 WAKEUP_INT1_PEND

- Address = 0x1100_0F44, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	—	Reserved	0x000000
WAKEUP_INT1_PEND[7]	[7]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT1_PEND[6]	[6]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT1_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT1_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT1_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT1_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT1_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT1_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.3.190 WAKEUP_INT2_PEND

- Address = 0x1100_0F48, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	—	Reserved	0x000000
WAKEUP_INT2_PEND[7]	[7]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT2_PEND[6]	[6]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT2_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT2_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT2_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT2_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT2_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT2_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.3.191 WAKEUP_INT3_PEND

- Address = 0x1100_0F4C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	—	Reserved	0x000000
WAKEUP_INT3_PEND[7]	[7]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT3_PEND[6]	[6]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT3_PEND[5]	[5]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT3_PEND[4]	[4]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT3_PEND[3]	[3]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT3_PEND[2]	[2]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT3_PEND[1]	[1]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0
WAKEUP_INT3_PEND[0]	[0]	RWX	0x0 = Not occur 0x1 = Occur interrupt	0x0

4.4.4 Part 3

4.4.4.1 GPZCON

- Address = 0x0386_0000, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
GPZCON[0]	[3:0]	RW	0x0 = Input 0x1 = Output 0x2 = I2S_0_SCLK 0x3 = PCM_0_SCLK 0x4 – 0xF = Reserved	0x00
GPZCON[1]	[7:4]	RW	0x0 = Input 0x1 = Output 0x2 = I2S_0_CDCLK 0x3 = PCM_0_EXTCLK 0x4 – 0xF = Reserved	0x00
GPZCON[2]	[11:8]	RW	0x0 = Input 0x1 = Output 0x2 = I2S_0_LRCK 0x3 = PCM_0_FSYNC 0x4 – 0xF = Reserved	0x00
GPZCON[3]	[15:12]	RW	0x0 = Input 0x1 = Output 0x2 = I2S_0_SDIN 0x3 = PCM_0_SDIN 0x4 – 0xF = Reserved	0x00
GPZCON[4]	[19:16]	RW	0x0 = Input 0x1 = Output 0x2 = I2S_0_SDO[0] 0x3 = PCM_0_SOUT 0x4 – 0xF = Reserved	0x00
GPZCON[5]	[23:20]	RW	0x0 = Input 0x1 = Output 0x2 = I2S_0_SDO[1] 0x3 = ST_TICK 0x4 – 0xF = Reserved	0x00
GPZCON[6]	[27:24]	RW	0x0 = Input 0x1 = Output 0x2 = I2S_0_SDO[2] 0x3 = ST_INT 0x4 – 0xF = Reserved	0x00

4.4.4.2 GPZDAT

- Address = 0x0386_0004, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
GPZDAT[6:0]	[6:0]	RWX	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.	0x00

4.4.4.3 GPZPUD

- Address = 0x0386_0008, Reset Value = 0x1555

Name	Bit	Type	Description	Reset Value
GPZPUD[n]	[2n+1:2n] n=0 – 6	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x1555

4.4.4.4 GPZDRV

- Address = 0x0386_000C, Reset Value = 0x00_0000

Name	Bit	Type	Description	Reset Value
RSVD	[23:16]	RW	Reserved (should be zero)	0x00
GPZDRV[n]	[2n+1:2n] n=0 – 6	RW	0x0 = 1x 0x2 = 2x 0x1 = 3x 0x3 = 4x	0x00_0000

4.4.4.5 GPZCONPDN

- Address = 0x0386_0010, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPZ[n]	[2n+1:2n] n=0 – 6	RW	0x0 = Output 0 0x1 = Output 1 0x2 = Input 0x3 = Previous state	0x00

4.4.4.6 GPZPUDPDN

- Address = 0x0386_0014, Reset Value = 0x0000

Name	Bit	Type	Description	Reset Value
GPZ[n]	[2n+1:2n] n=0 – 6	RW	0x0 = Pull-up/down disabled 0x1 = Pull-down enabled 0x2 = Reserved 0x3 = Pull-up enabled	0x00

5 Clock Management Units

5.1 Overview

This chapter describes the clock management units (CMUs) of Exynos4210. CMUs control Phase Locked Loops (PLLs) and generate system clocks for CPU and buses and function clocks for individual IPs in Exynos4210. They also communicate with the power management unit (PMU) in order to stop clocks before entering certain low power mode to reduce power consumption by minimizing clock toggling.

Table 5-1 Operating Frequencies in Exynos4210

Function Block	Description	Typical Operating Frequency
CPU	Cortex-A9 MPCore (Dual Core)	1 GHz@1.2 V, 800 MHz@1.1 V
	CoreSight	200 MHz/100 MHz
DMC	DMC0 and DMC1	400 MHz
	MIU	200 MHz
SSS	Security Sub-System	200 MHz
LEFTBUS	Data Bus / Peripheral Bus	200 MHz/100 MHz
RIGHTBUS	Data Bus / Peripheral Bus	200 MHz/100 MHz
G3D	3D Graphics Engine	400 MHz@1.2 V, 330 MHz@1.1 V
MFC	Multi-format Codec	200 MHz
IMAGE	2D Graphics Engine, Rotator, MDMA	200 MHz (266 MHz for G2D)
LCD0	FIMD0, MIE0, MIPI DSI0	160 MHz
LCD1	FIMD1, MIE1, MIPI DSI1	160 MHz
CAM	FIMC0~3, JPEG	160 MHz
TV	VP, MIXER, TVENC	160 MHz
FSYS	USB, PCIe, SATA, SDMMC, TSI, OneNANDC, SROMC, PDMA0, PDMA1, NFCON	133 MHz
GPS	GPS	133 MHz
MAUDIO	AudioSS, iROM , iRAM	100 MHz
PERI-L	UART, I2C, TSADC, SPI, I2S, PCM, SPDIF, PWM, MODEM I/F	100 MHz
PERI-R	CHIPID, SYSREG, PMU/CMU/TMU Bus I/F, MCTimer, WDT, RTC, KEYIF, SECKEY, TZPC	100 MHz

NOTE: Refer to the Audio Subsystem chapter for more details on MAUDIO block clocks.

5.2 Functional Description

5.2.1 Clock Declaration

The top-level clocks in Exynos4210 include:

- Clocks from clock pads, that is, XXTI, and XUSBXTI.
- Clocks from CMUs (for instance, ARMCLK, ACLK, HCLK, and SCLK)
- Clocks from HDMI-PHY

5.2.1.1 Clocks from Clock Pads

The following clocks are provided by clock pads. However, one may disable crystal clock pads.

- **XXTI:** Specifies a clock from the XXTI pin. The input frequency ranges from 12 to 50 MHz. When XXTI is not used, It should be pulled-down.
- **XUSBXTI:** Specifies a clock from crystal pads of the XusbXTI and XusbXTO pins. XusbXTI and XusbXTO use wide range OSC pads. This clock is supplied to APLL, MPLL, VPLL, ELL, and USB-PHY. For more information on USB PHY clock, refer to Chapter 36 "USB HOST" and 37 "USB DEVICE". 24 MHz should be used because the iROM code was designed based on the 24 MHz input clock. A 5Mohms parallel resister between XusbXTI and XusbXTO is required.

Caution: XXTI should be used only for testing purposes.

5.2.1.2 Clocks from CMU

CMUs generate internal clocks with intermediate frequencies using clocks from the clock pads (i.e., XXTI, and XUSBXTI), four PLLs (i.e., APLL, MPPLL, EPLL, and VPLL), and USB PHY and HDMI PHY clocks. Some of these clocks can be selected, pre-scaled, and provided to the corresponding modules.

To generate internal clocks, the following components are used.

- APLL can generate 30 MHz – 1030 MHz.
- MPPLL can generate 30 MHz – 1030 MHz.
- EPLL can generate 10 MHz – 600 MHz.
- VPLL can generate 10 MHz – 600 MHz. VPLL generates 54 MHz video clock or 330 MHz G3D clock.
- The USB Device PHY uses XUSBXTI to generate 30 MHz and 48 MHz
- The HDMI PHY uses XUSBXTI to generate 54 MHz

In typical Exynos4210 applications,

- APLL is used for Cortex-A9, CoreSight, and HPM.
- MPPLL is used for DMC, system bus, and other peripherals.
- G3D uses MPPLL/VPLL.
- EPLL is used for I2S's.

The Clock Controllers allow bypassing of PLLs for slow clock. Moreover, they can gate clocks in each block for power reduction.

5.2.2 Clock Values

Values for high-performance operation:

- freq (ARMCLK) = 1000 MHz@1.2 V, 800 MHz@1.1 V
- freq (ACLK_COREM0) = 250 MHz@1.2 V, 200 MHz@1.1 V
- freq (ACLK_COREM1) = 125 MHz@1.2 V, 100 MHz@1.1 V
- freq (PERIPHCLK) = 250 MHz@1.2 V, 200 MHz@1.1 V
- freq (ATCLK) = 200 MHz
- freq (PCLK_DBG) = 100 MHz
- freq (SCLK_DMC) = 400 MHz
- freq (ACLK_DMCD) = 200 MHz
- freq (ACLK_DMCP) = 100 MHz
- freq (ACLK_ACP) = 200 MHz
- freq (PCLK_ACP) = 100 MHz
- freq (ACLK_GDL) = 200 MHz
- freq (ACLK_GPL) = 100 MHz
- freq (ACLK_GDR) = 200 MHz
- freq (ACLK_GPR) = 100 MHz
- freq (ACLK_200) = 200 MHz
- freq (ACLK_100) = 100 MHz
- freq (ACLK_160) = 160 MHz
- freq (ACLK_133) = 133 MHz
- freq (SCLK_ONENAND) = 133 MHz, 160 MHz

Caution: The ratio between freq. (SCLK_DMC) and freq. (ACLK_DMCD) should be 2:1. This ratio should not be changed while DMC is running.

5.2.2.1 Recommended PMS Value for APLL & MPLL

Table 5-2 APLL & MPLL PMS Value

FIN (MHz)	Target FOUT (MHz)	P	M	S	AFC_ENB	AFC	FVCO (MHz)	FOUT (MHz)
24	200	6	200	3	0	0	1600	200
24	400	6	200	2	0	0	1600	400
24	500	6	250	2	0	0	2000	500
24	800	6	200	1	0	0	1600	800
24	1000	6	250	1	0	0	2000	1000
24	1200	3	150	1	0	0	2400	1200

NOTE: Although there is an equation for choosing PMS values, we strongly recommend only the values in the above table. If you have to use other values, please contact us.

5.2.2.2 Recommended PMS Value for EPLL

Table 5-3 EPLL PMS Value

FIN (MHz)	Target FOUT (MHz)	VSEL	P	M	S	K	FVCO (MHz)	FOUT (MHz)
24	180.0000	0	3	45	1	0	360.000	180.000
24	192.0000	0	3	48	1	0	384.000	192.000
24	49.1520	0	3	49	3	9961	393.21594	49.15199
24	45.1584	0	3	45	3	10381	361.26721	45.15840
24	73.7280	1	3	73	3	47710	589.82397	73.72800
24	67.7376	1	4	90	3	20762	541.9008	67.73760

NOTE: Although there is an equation for choosing PMS values, we strongly recommend only the values in the above table. If you have to use other values, please contact us.

5.2.2.3 Recommended PMS Value for VPLL

Table 5-4 VPLL PMS Value

FIN (MHz)	Target FOUT (MHz)	P	M	S	K	MFR	MRR	VSEL	SSCG_EN
24	54	3	53	3	1024	0	17	0	0
	330	2	53	1	2048	1	1	1	0

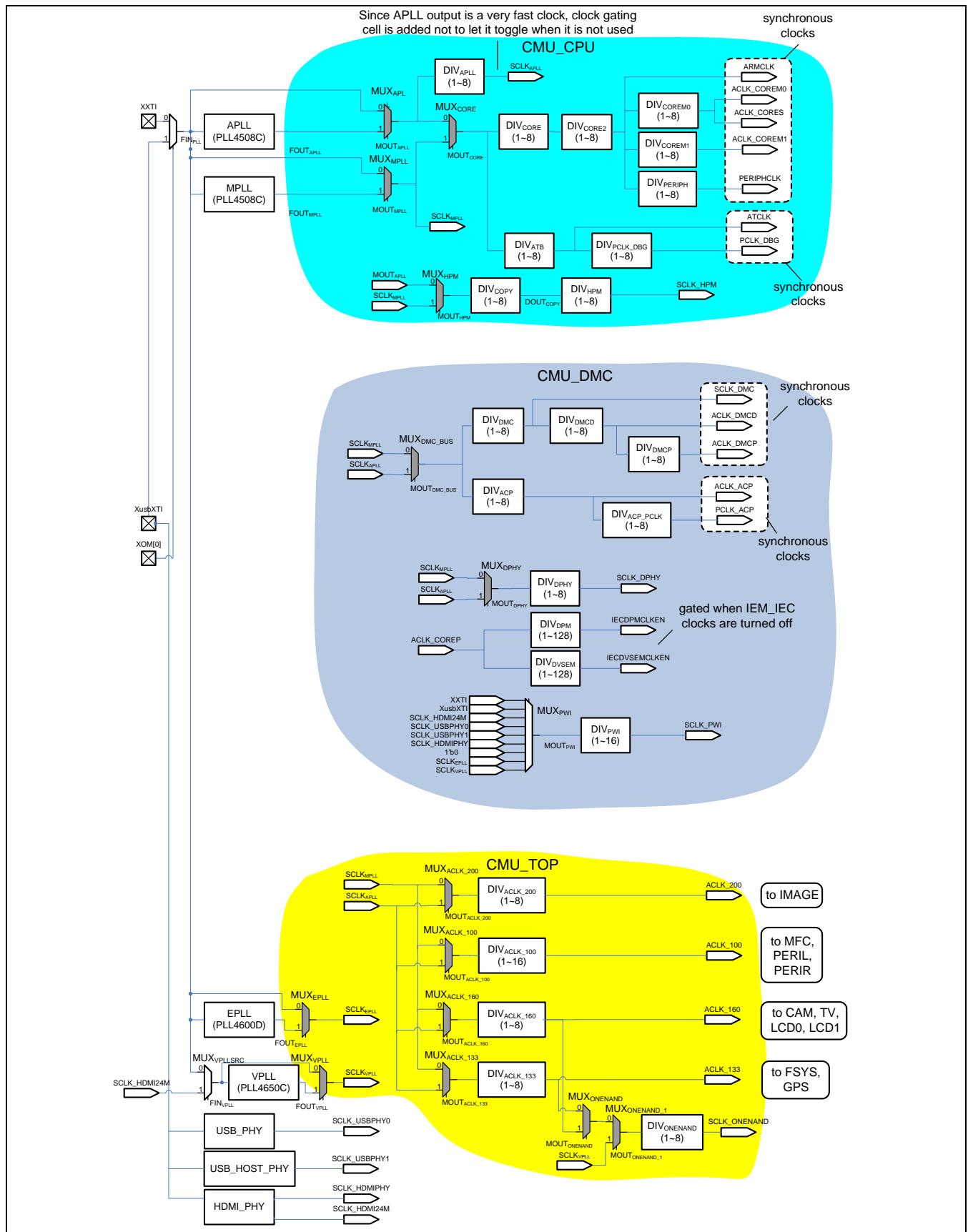
NOTE: Although there is an equation for choosing PMS values, we strongly recommend only the values in the above table. If you have to use other values, please contact us.

5.2.3 Clock Generation

[Figure 5-1](#) and [Figure 5-2](#) show block diagram of the clock generation logic. An external crystal clock is connected to the oscillation amplifier. The PLL converts low input frequency to high-frequency clock required by Exynos4210. The clock generator block also includes a built-in logic to stabilize the clock frequency after each system reset, since clock takes time for stabilizing.

[Figure 5-1](#) and [Figure 5-2](#) also show two types of clock mux. Clock mux in grey color represents glitch-free clock mux, which is free of glitches while clock selection is changed. Clock mux in white color represents non-glitch-free clock mux, which can suffer from glitches when changing clock sources. Care must be taken in using each clock mux. For glitch-free mux, it should be guaranteed that all clock sources are running when clock selection is changed from one to the other. If that's not the case, clock changing is not finished fully, resulting clock output may have unknown states. For non-glitch-free clock mux, it is possible to have a glitch when clock selection is changed. To prevent glitch signals, it is recommended to disable output of a non-glitch-free mux before trying to change clock sources. After clock changing is completed, users can re-enable output of the non-glitch-free clock mux so that there will be no glitches resulting from clock changes. Masking output of non-glitch-free muxes is handled by clock source mask control registers, starting with the keyword CLK_SRC_MASK.

Clock dividers shown in [Figure 5-1](#) and [Figure 5-2](#) indicate possible dividing value in parentheses. Those diving values can be changed by clock divider registers at run-time. Some clock dividers may have only one dividing value and user is not allowed to change the dividing value. For such dividers, i.e., denoted as "/2", there is no corresponding field in clock divider registers.



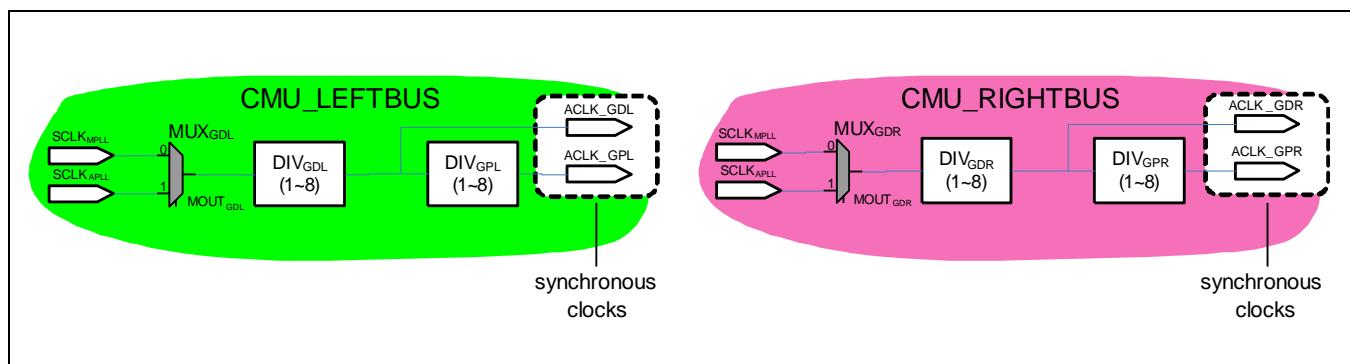
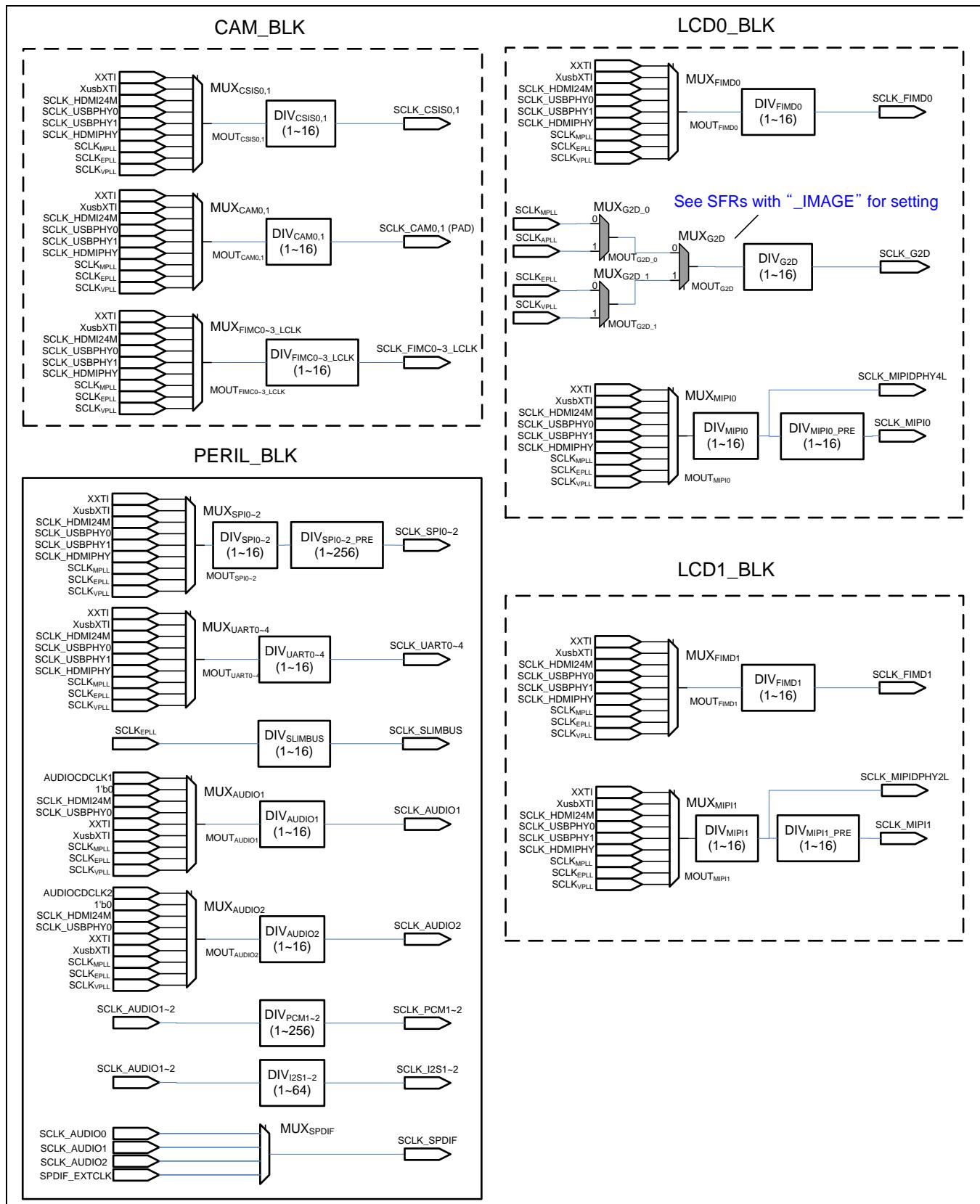


Figure 5-1 Exynos4210 Clock Generation Circuit (CPU, BUS, DRAM Clocks)



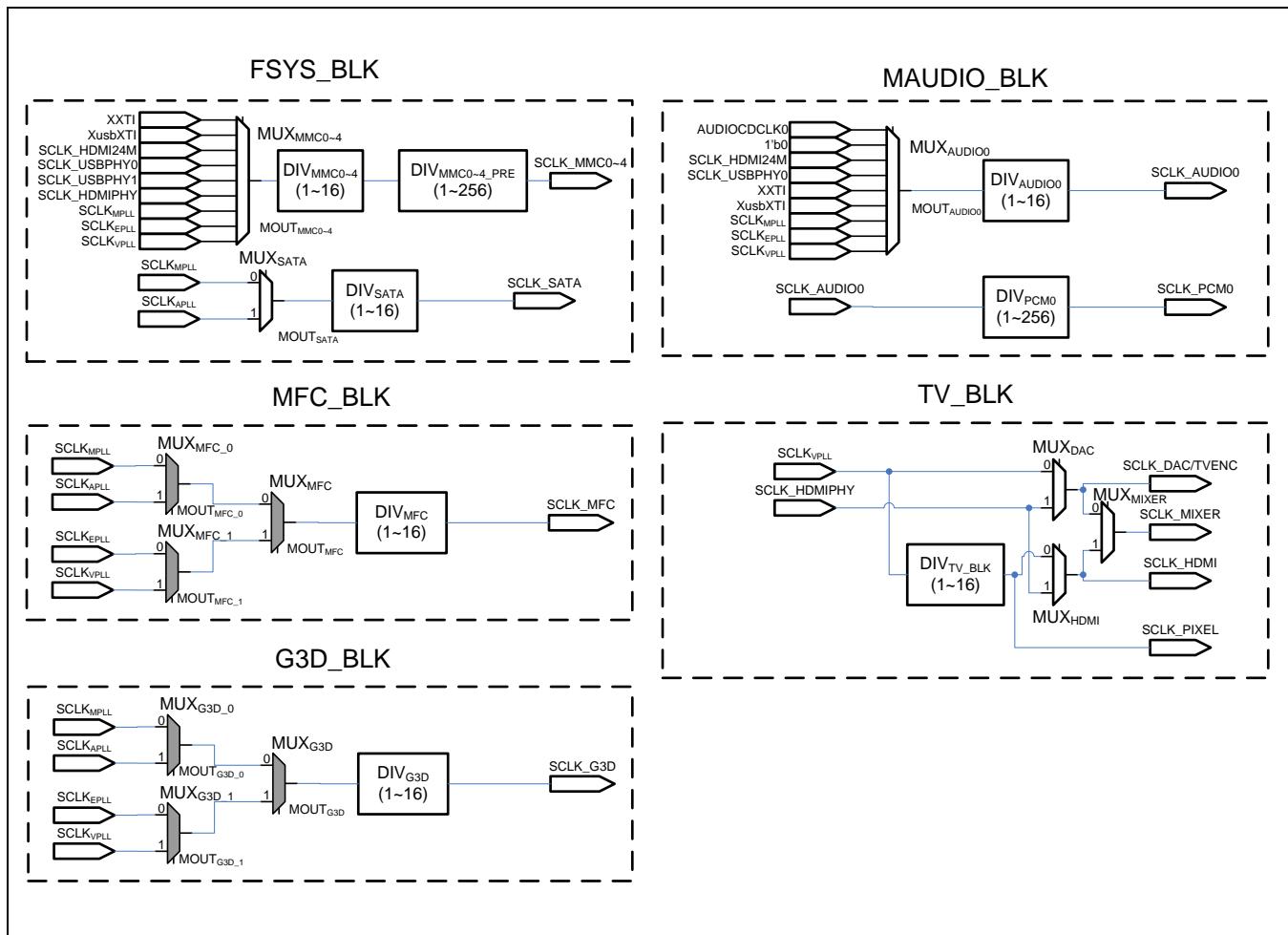


Figure 5-2 Exynos4210 Clock Generation Circuit (Special Clocks)

Caution: In the above figures, muxes with grey color are glitch-free. For glitch-free clock muxes, it should be guaranteed that all clock sources are running when clock selection is changed. For clock dividers, it should be guaranteed that input clock is running when divider value is changed

5.2.4 Clock Configuration Procedure

Below rules should be obeyed when changing clock configuration.

- All inputs of a glitch-free mux must run.
- When a PLL is turned off, one should not select the output of PLL.

Basic SFR configuration flows:

Change the system clock divider values

CLK_DIV_CPU0[31:0] = target value0;

CLK_DIV_DMC0[31:0] = target value1;

CLK_DIV_TOP[31:0] = target value2;

CLK_DIV_LEFTBUS[31:0] = target value3;

CLK_DIV_RIGHTBUS[31:0] = target value4;

Change the divider values for special clocks by setting CLK_DIV_XXX SFRs in CMU_TOP

CLK_DIV_XXX[31:0] = target value;

Change PLL PMS values

Set PMS values; // Set PDIV, MDIV, and SDIV values
(Refer to (A, M, E, V)PLL_CON0 SFRs)

Change other PLL control values

(A,M,E,V)PLL_CON1[31:0] = target value; // Set K, AFC, MRR, MFR values if necessary
(Refer to (A, M, E, V)PLL_CON1 SFRs)

Turn on a PLL

(A,M,E,V)PLL_CON0[31] = 1; // Turn on a PLL (Refer to (A, M, E, V)PLL_CON0 SFRs)

wait_lock_time; // Wait until the PLL is locked

MUX_(A, M, E, V)PLL_SEL = 1; // Select the PLL output clock instead of input reference clock,
after PLL output clock is stabilized. (Refer to CLK_SRC_CPU
SFR for APLL and MPLL, CLK_SRC_TOP0 for EPLL and VPLL)
Once a PLL is turned on, do not turn it off.

5.2.4.1 Clock Gating

Exynos4210 can disable the clock operation of each IP if it is not required thereby reducing the dynamic power.

There are two types of clock gating control registers to disable/enable clock operation, namely:

- Clock gating control register for function block
- Clock gating control register for IP

The above two registers are ANDed together to generate a final clock gating enable signal. As a result, if either of the two register field is turned OFF, the resulting clock is stopped. For instance, in order to stop clocks that are provided to MIXER module, one may set CLK_MIXER field in CLK_GATE_IP_TV register to 0 or CLK_TV field in CLK_GATE_BLOCK register to 0. Note, for latter case, all clocks in TV block, not only MIXER clocks, are turned off.

Caution: It should be guaranteed that S/W does not access IPs whose clock is gated. It may cause system failure

5.2.4.2 Clock Diving

Whenever clock divider control register is changed, it is recommended to check clock divider status registers before using the new clock output. This guarantees the corresponding divider finishes changing to a new dividing value before its output is used by other modules.

5.2.5 Special Clock Description

5.2.5.1 Special Clock Table

Table 5-5 Special Clocks in Exynos4210

Name	Description	Range	Source
SCLK_ONENAND	ONENAND operating clock	~ 160 MHz	ACLK_160, ACLK_133
SCLK_G3D	G3D core operating clock	~ 330 MHz@1.1 V ~ 400 MHz@1.2 V	SCLK _{APLL} , SCLK _{MPLL} , SCLK _{EPLL} , SCLK _{VPLL}
SCLK_G2D	G2D core operating clock	~ 266 MHz	SCLK _{APLL} , SCLK _{MPLL} , SCLK _{EPLL} , SCLK _{VPLL}
SCLK_MFC	MFC core operating clock	~ 200 MHz	SCLK _{APLL} , SCLK _{MPLL} , SCLK _{EPLL} , SCLK _{VPLL}
SCLK_CAM0,1	Reference clock for external CAM device	CAM spec	All possible clock sources
SCLK_CSIS0,1	CSIS operating clock	~ 160 MHz	All possible clock sources
SCLK_FIMC_LCLK0~3	FIMC core operating clock	~ 160 MHz	All possible clock sources
SCLK_FIMD0,1	FIMD operating clock	~ 100 MHz	All possible clock sources
SCLK_MIPI0,1	MIPI DSIM clock	~ 100 MHz	All possible clock sources
SCLK_MIPIDPHY2L	MIPI DPHY 2 Lane clock	~ 800 MHz	All possible clock sources
SCLK_MIPIDPHY4L	MIPI DPHY 4 Lane clock	~ 800 MHz	All possible clock sources
SCLK_TVENC	TVENC/ DAC clock	54 MHz	SCLK _{VPLL} , HDMI PHY output
SCLK_DAC	DAC clock	54 MHz	SCLK _{VPLL} , HDMI PHY output
SCLK_MIXER	MIXER clock	54 MHz (TV) ~ 148.5 MHz (HDMI)	SCLK _{VPLL} , HDMI PHY output
SCLK_HDMI	HDMI LINK clock	~ 148.5 MHz	All possible clock sources
SCLK_PIXEL	HDMI PIXEL clock	~ 148.5 MHz	All possible clock sources
SCLK_SPDIF	SPDIF operating clock	~ 83 MHz	SCLK_AUDIO0,1,2
SCLK_MMC0~4	HSMMC operating clock	~ 50 MHz	All possible clock sources
SCLK_SATA	SATA clock	~ 66.7 MHz	SCLK _{APLL} , SCLK _{MPLL}
SCLK_USBPHY0	USB Device clock	48 MHz	USB Device PHY clock out
SCLK_USBPHY1	USB HOST clock	48 MHz	USB HOST PHY clock out
SCLK_AUDIO0,1,2	AUDIO operating clock (I2S)	~ 100 MHz	All possible clock sources, AUDIOCDCLKx
SCLK_PCM0,1,2	AUDIO operating clock (PCM)	~ 5 MHz	SCLK_AUDIO0,1,2
SCLK_PWI	IEM APC operating clock	6 ~ 30 MHz	All possible clock sources
PLL reference clock	KEY I/F or TSADC filter clock	~ 24 MHz	XUSBXTI
SCLK_SPI0,1,2	SPI operating clock	~ 100 MHz	All possible clock sources
SCLK_UART0~4	UART operating clock	~ 200 MHz	All possible clock sources

- All possible clock sources include XXTI, XUSBXTI, SCLK_HDMI24M, SCLK_USBPHY, SCLK_HDMIPHY, SCLK_{MPLL}, SCLK_{EPLL}, and SCLK_{VPLL}.
- XXTI and XUSBXTI refer to external crystal, respectively.
- SCLK_USBPHY refers to USB PHY 48 MHz output clock.
- SCLK_HDMI24M refers to HDMI PHY (24 MHz reference clock for XUSBXTI) output.
- SCLK_HDMIPHY refers to HDMI PHY (PIXEL_CLKO) output clock.
- SCLK_{MPLL}, SCLK_{EPLL}, and SCLK_{VPLL} refer to the output clock of MPLL, EPLL, and VPLL, respectively.

Table 5-6 I/O Clocks in Exynos4210

Name	I/O	PAD	GPIO Function	Range	Description
IOCLK_AC97	IN	Xi2s1SCLK	Func2: AC97BITCLK	12.288 MHz	AC97 bit clock
IOCLK_I2S0,1,2	IN	Xi2s0CDCLK Xi2s1CDCLK Xpcm2EXTCLK	Func0: I2S_0_CDCLK Func0: I2S_1_CDCLK Func2: I2S_2_CDCLK	to 83.4 MHz	I2S CODEC clock
IOCLK_PCM0,1,2	IN	Xi2s0CDCLK Xi2s1CDCLK Xpcm2EXTCLK	Func1: PCM_0_EXTCLK Func1: PCM_1_EXTCLK Func0: PCM_2_EXTCLK	to 83.4 MHz	PCM CODEC clock
IOCLK_SPDIF	IN	Xpcm2EXTCLK	Func1: SPDIF_EXTCLK	36.864 MHz	SPDIF input clock

5.2.6 CLKOUT

User can monitor certain clocks in Exynos4210 using XCLKOUT port. Each of the five CMUs contains CLKOUT control logic where one of the clocks in that CMU is selected and divided if necessary. The generated CLKOUT signal from each CMU is fed to PMU and muxed with other clock signals (For the SFR description for the mux of PMU, refer to the PMU chapter).

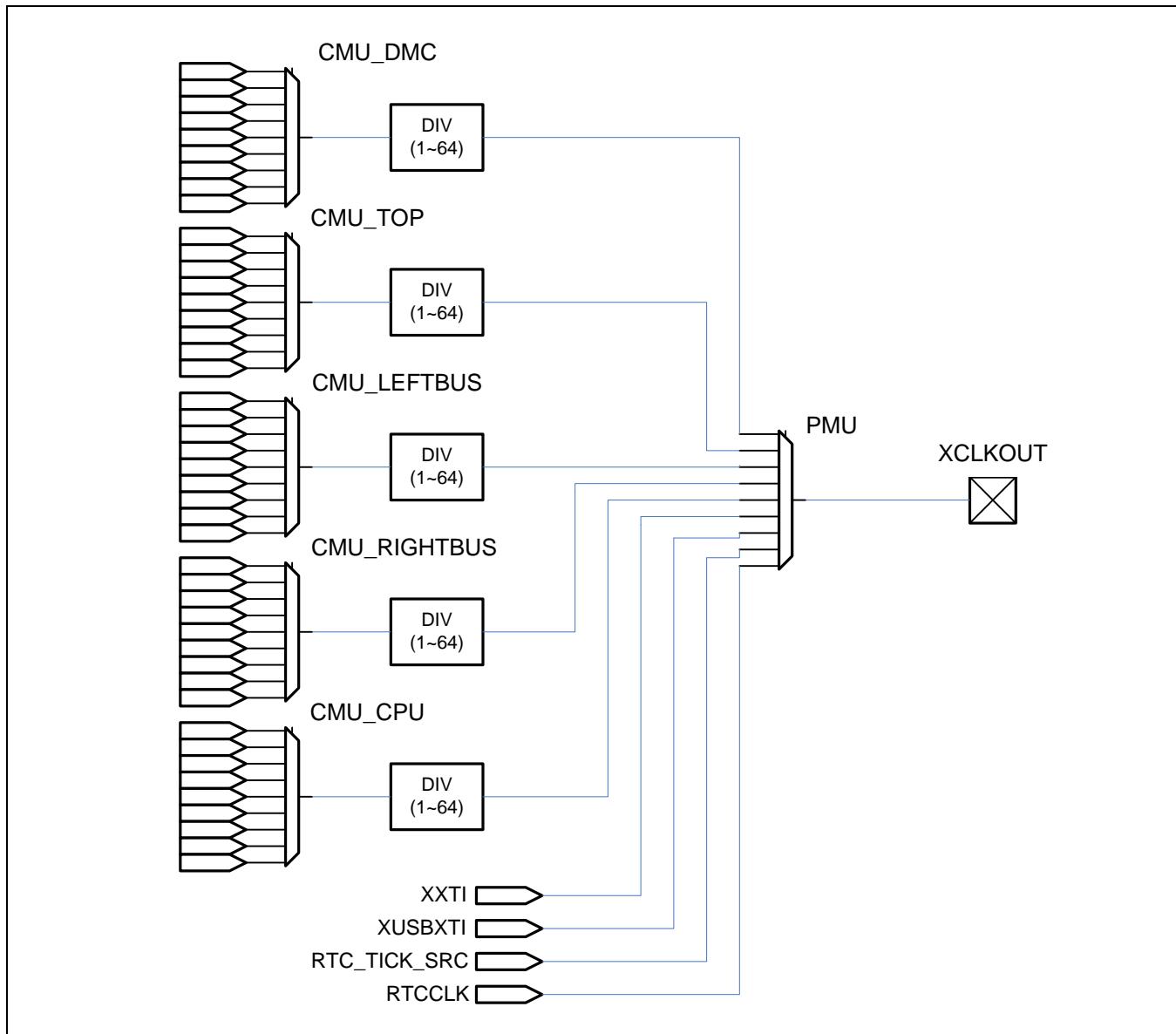


Figure 5-3 Exynos4210 CLKOUT Control Logic

Table 5-7 CLKOUT Input Clock Selection Information

-	CMU_CPU	CMU_DMC	CMU_TOP	CMU_RIGHTBUS	CMU_LEFTBUS	PMU
0	APLL_FOUT/2	ACLK_DMCD	EPLL_FOUT	SCLK_MPLL/2	SCLK_MPLL/2	CMU_DMC
1	Reserved	ACLK_DMCP	VPLL_FOUT	SCLK_APPLL/2	SCLK_APPLL/2	CMU_TOP
2	MPLL_FOUT/2	ACLK_ACP	SCLK_HDMI24M	ACLK_GDR	ACLK_GDL	CMU_LEFTBUS
3	Reserved	PCLK_ACP	SCLK_USBPHY0	ACLK_GPR	ACLK GPL	CMU_RIGHTBUS
4	ARMCLK/2	SCLK_DMC	SCLK_USBPHY1	–	–	CMU_CPU
5	ACLK_COREM0	SCLK_DPHY	SCLK_HDMIPHY	–	–	XXTI
6	ACLK_COREM1	Reserved	AUDIOCDCLK0	–	–	XUSBXTI
7	ACLK_CORES	SCLK_PWI	AUDIOCDCLK1	–	–	RTC_TICK_SRC
8	ATCLK	–	AUDIOCDCLK2	–	–	RTCCLK
9	PERIPHCLK	–	SPDIF_EXTCLK	–	–	–
10	PCLK_DBG	–	ACLK_160	–	–	–
11	SCLK_HPM	–	ACLK_133	–	–	–
12	–	–	ACLK_200	–	–	–
13	–	–	ACLK_100	–	–	–
14	–	–	SCLK_MFC	–	–	–
15	–	–	SCLK_G3D	–	–	–
16	–	–	SCLK_G2D	–	–	–
17	–	–	CAM_A_PCLK	–	–	–
18	–	–	CAM_B_PCLK	–	–	–
19	–	–	S_RXBYTECLKHS0_2L	–	–	–
20	–	–	S_RXBYTECLKHS0_4L	–	–	–

5.2.7 I/O Description

Signal	I/O	Description	Pad	Type
XXTI	Input	External oscillator pad	XXTI	dedicated
XUSBXTI	Input	Input pad for crystal	XusbXTI	dedicated
XUSBXTO	Output	Output pad for crystal	XusbXTO	dedicated
EPLLFILTER	Input/Output	Pad for EPLL loop filter capacitor	XEPLLFILTER	dedicated
VPLLFILTER	Input/Output	Pad for VPLL loop filter capacitor	XVPLLFILTER	dedicated
CLKOUT	Output	Clock out pad	XCLKOUT	dedicated

5.3 Register Description

Clock controller controls PLLs and clock generation units. This section describes how to control these parts using Special Functional Registers (SFRs) in the clock controller. Do not change any reserved area. Changing value of reserved area may lead to unexpected behavior.

[Figure 5-4](#) shows the address map of Exynos4210 clock controller. There are five CMUs, i.e., CMU_LEFTBUS, CMU_RIGHTBUS, CMU_TOP, CMU_DMC, and CMU_CPU, in Exynos4210 and each CMU uses 16 KB address space for SFRs. The internal structure of the address space for each CMU is same for all CMUs and it is as shown in the right part of [Figure 5-4](#). It is broken down into the following categories:

- 0x000 to 0x1FF is used for PLL control: PLL lock time and control
- 0x200 to 0x4FF is used for mux control: Mux selection, output masking, and status
- 0x500 to 0x6FF is used for clock division: Divider ratio and status
- 0x700 to 0x8FF is reserved and user is not allowed to access the region.
- 0x900 to 0x9FF is used for clock gating control: Clock gating of IPs and function blocks
- 0xA00 to 0xAFF is used for CLKOUT: CLKOUT input clock selection and divider ratio

NOTE: For CMU_LEFTBUS and CMU_RIGHTBUS, CLK_GATE_IP_XXX registers are located at 0x800.

In addition, some CMUs use addresses beyond 0xAFF for other functions such as CPU and IEM control in CMU_CPU. See register description for more details.

In [Figure 5-4](#), XXX in the register name shall be replaced with the function block name, i.e., LEFTBUS, RIGHTBUS, TOP, CAM, TV, MFC, G3D, IMAGE, LCD0, LCD1, MAUDIO, FSYS, PERIL, and PERIR.

0x1003_4000	CMU_LEFTBUS	0x000	xPLL_LOCK	PLL lock time
		0x100	xPLL_CON	PLL control
		0x200	CLK_SRC_XXX	Mux selection
		0x300	CLK_SRC_MASK_XXX	Mux output masking
		0x400	CLK_MUX_STAT_XXX	Mux status
		0x500	CLK_DIV_XXX	Divider ratio
0x1003_8000	CMU_RIGHTBUS	0x600	CLK_DIV_STAT_XXX	Divider status
		0x700	Reserved	
		0x800	Reserved	
		0x900	CLK_GATE_IP_XXX	IP clock gating
		0xA00	CLKOUT_CMU_XXX	CLKOUT control

* XXX : function block name

Figure 5-4 Exynos4210 Clock Controller Address Map

5.3.1 Register Map Summary

- Base Address: 0x1003_0000

Register	Offset	Description	Reset Value
CLK_SRC_LEFTBUS	0x4200	Select clock source for CMU_LEFTBUS	0x0000_0000
CLK_MUX_STAT_LEFTBUS	0x4400	Clock MUX status for CMU_LEFTBUS	0x0000_0001
CLK_DIV_LEFTBUS	0x4500	Set clock divider ratio for CMU_LEFTBUS	0x0000_0000
CLK_DIV_STAT_LEFTBUS	0x4600	Clock divider status for CMU_LEFTBUS	0x0000_0000
CLK_GATE_IP_LEFTBUS	0x4800	Control IP clock gating for LEFTBUS_BLK	0xFFFF_FFFF
CLKOUT_CMU_LEFTBUS	0x4A00	CLKOUT control register	0x0001_0000
CLKOUT_CMU_LEFTBUS_DIV_STAT	0x4A04	Clock divider status for CLKOUT	0x0000_0000
CLK_SRC_RIGHTBUS	0x8200	Select clock source for CMU_RIGHTBUS	0x0000_0000
CLK_MUX_STAT_RIGHTBUS	0x8400	Clock MUX status for CMU_RIGHTBUS	0x0000_0001
CLK_DIV_RIGHTBUS	0x8500	Set clock divider ratio for CMU_RIGHTBUS	0x0000_0000
CLK_DIV_STAT_RIGHTBUS	0x8600	Clock divider status for CMU_RIGHTBUS	0x0000_0000
CLK_GATE_IP_RIGHTBUS	0x8800	Control IP clock gating for RIGHTBUS_BLK	0xFFFF_FFFF
CLKOUT_CMU_RIGHTBUS	0x8A00	CLKOUT control register	0x0001_0000
CLKOUT_CMU_RIGHTBUS_DIV_STAT	0x8A04	Clock divider status for CLKOUT	0x0000_0000
EPLL_LOCK	0xC010	Control PLL locking period for EPLL.	0x0000_0FFF
VPLL_LOCK	0xC020	Control PLL locking period for VPLL.	0x0000_0FFF
EPLL_CON0	0xC110	Control PLL output frequency for EPLL.	0x0030_0301
EPLL_CON1	0xC114	Control PLL output frequency for EPLL.	0x0000_0000
VPLL_CON0	0xC120	Control PLL output frequency for VPLL.	0x0024_0201
VPLL_CON1	0xC124	Control PLL output frequency for VPLL.	0x6601_0464
CLK_SRC_TOP0	0xC210	Select clock source for CMU_TOP	0x0000_0000
CLK_SRC_TOP1	0xC214	Select clock source for CMU_TOP	0x0000_0000
CLK_SRC_CAM	0xC220	Select clock source CAM_BLK	0x1111_1111
CLK_SRC_TV	0xC224	Select clock source for TV_BLK	0x0000_0000
CLK_SRC_MFC	0xC228	Select clock source for MFC_BLK	0x0000_0000
CLK_SRC_G3D	0xC22C	Select clock source for G3D_BLK	0x0000_0000
CLK_SRC_IMAGE	0xC230	Select clock source for IMAGE_BLK	0x0000_0000
CLK_SRC_LCD0	0xC234	Select clock source for LCD_BLK	0x0000_1111
CLK_SRC_LCD1	0xC238	Select clock source for LCD_BLK	0x0000_1111
CLK_SRC_AUDIO	0xC23C	Select clock source for AUDIO_BLK	0x0000_0005
CLK_SRC_FSYS	0xC240	Select clock source for FSYS_BLK	0x0001_1111
CLK_SRC_PERIL0	0xC250	Select clock source for connectivity IPs	0x0001_1111
CLK_SRC_PERIL1	0xC254	Select clock source for connectivity IPs	0x0111_0055

Register	Offset	Description	Reset Value
CLK_SRC_MASK_TOP	0xC310	Clock source mask for CMU_TOP	0x0000_0001
CLK_SRC_MASK_CAM	0xC320	Clock source mask for CAM_BLK	0x1111_1111
CLK_SRC_MASK_TV	0xC324	Clock source mask for TV_BLK	0x0000_0111
CLK_SRC_MASK_LCD0	0xC334	Clock source mask for LCD0_BLK	0x0000_1111
CLK_SRC_MASK_LCD1	0xC338	Clock source mask for LCD1_BLK	0x0000_1111
CLK_SRC_MASK_MAUDIO	0xC33C	Clock source mask for AUDIO_BLK	0x0000_0001
CLK_SRC_MASK_FSYS	0xC340	Clock source mask for FSYS_BLK	0x0101_1111
CLK_SRC_MASK_PERIL0	0xC350	Clock source mask for PERIL_BLK	0x0001_1111
CLK_SRC_MASK_PERIL1	0xC354	Clock source mask for PERIL_BLK	0x0111_0111
CLK_MUX_STAT_TOP	0xC410	Clock MUX status for CMU_TOP	0x1111_1111
CLK_MUX_STAT_MFC	0xC428	Clock MUX status for MFC_BLK	0x0000_0111
CLK_MUX_STAT_G3D	0xC42C	Clock MUX status for G3D_BLK	0x0000_0111
CLK_MUX_STAT_IMAGE	0xC430	Clock MUX status for IMAGE_BLK	0x0000_0111
CLK_DIV_TOP	0xC510	Set clock divider ratio for CMU_TOP	0x0000_0000
CLK_DIV_CAM	0xC520	Set clock divider ratio for CAM_BLK	0x0000_0000
CLK_DIV_TV	0xC524	Set clock divider ratio for TV_BLK	0x0000_0000
CLK_DIV_MFC	0xC528	Set clock divider ratio for MFC_BLK	0x0000_0000
CLK_DIV_G3D	0xC52C	Set clock divider ratio for G3D_BLK	0x0000_0000
CLK_DIV_IMAGE	0xC530	Set clock divider ratio for IMAGE_BLK	0x0000_0000
CLK_DIV_LCD0	0xC534	Set clock divider ratio for LCD0_BLK	0x0070_0000
CLK_DIV_LCD1	0xC538	Set clock divider ratio for LCD1_BLK	0x0070_0000
CLK_DIV_MAUDIO	0xC53C	Set clock divider ratio for AUDIO_BLK	0x0000_0000
CLK_DIV_FSYS0	0xC540	Set clock divider ratio for FSYS_BLK	0x00B0_0000
CLK_DIV_FSYS1	0xC544	Set clock divider ratio for FSYS_BLK	0x0000_0000
CLK_DIV_FSYS2	0xC548	Set clock divider ratio for FSYS_BLK	0x0000_0000
CLK_DIV_FSYS3	0xC54C	Set clock divider ratio for FSYS_BLK	0x0000_0000
CLK_DIV_PERIL0	0xC550	Set clock divider ratio for PERIL_BLK	0x0000_0000
CLK_DIV_PERIL1	0xC554	Set clock divider ratio for PERIL_BLK	0x0000_0000
CLK_DIV_PERIL2	0xC558	Set clock divider ratio for PERIL_BLK	0x0000_0000
CLK_DIV_PERIL3	0xC55C	Set clock divider ratio for PERIL_BLK	0x0000_0000
CLK_DIV_PERIL4	0xC560	Set clock divider ratio for PERIL_BLK	0x0000_0000
CLK_DIV_PERIL5	0xC564	Set clock divider ratio for PERIL_BLK	0x0000_0000
CLKDIV2_RATIO	0xC580	Set PCLK divider ratio in FSYS, CAM, LCD0, LCD1, TV, GPS block	0x0111_1111
CLK_DIV_STAT_TOP	0xC610	Clock divider status for CMU_TOP	0x0000_0000
CLK_DIV_STAT_CAM	0xC620	Clock divider status for CAM_BLK	0x0000_0000
CLK_DIV_STAT_TV	0xC624	Clock divider status for TV_BLK	0x0000_0000

Register	Offset	Description	Reset Value
CLK_DIV_STAT_MFC	0xC628	Clock divider status for MFC_BLK	0x0000_0000
CLK_DIV_STAT_G3D	0xC62C	Clock divider status for G3D_BLK	0x0000_0000
CLK_DIV_STAT_IMAGE	0xC630	Clock divider status for IMAGE_BLK	0x0000_0000
CLK_DIV_STAT_LCD0	0xC634	Clock divider status for LCD0_BLK	0x0000_0000
CLK_DIV_STAT_LCD1	0xC638	Clock divider status for LCD1_BLK	0x0000_0000
CLK_DIV_STAT_AUDIO	0xC63C	Clock divider status for AUDIO_BLK	0x0000_0000
CLK_DIV_STAT_FSYS0	0xC640	Clock divider status for FSYS_BLK	0x0000_0000
CLK_DIV_STAT_FSYS1	0xC644	Clock divider status for FSYS_BLK	0x0000_0000
CLK_DIV_STAT_FSYS2	0xC648	Clock divider status for FSYS_BLK	0x0000_0000
CLK_DIV_STAT_FSYS3	0xC64C	Clock divider status for FSYS_BLK	0x0000_0000
CLK_DIV_STAT_PERILO	0xC650	Clock divider status for PERIL_BLK	0x0000_0000
CLK_DIV_STAT_PERIL1	0xC654	Clock divider status for PERIL_BLK	0x0000_0000
CLK_DIV_STAT_PERIL2	0xC658	Clock divider status for PERIL_BLK	0x0000_0000
CLK_DIV_STAT_PERIL3	0xC65C	Clock divider status for PERIL_BLK	0x0000_0000
CLK_DIV_STAT_PERIL4	0xC660	Clock divider status for PERIL_BLK	0x0000_0000
CLK_DIV_STAT_PERIL5	0xC664	Clock divider status for PERIL_BLK	0x0000_0000
CLKDIV2_STAT	0xC680	PCLK divider status for FSYS, CAM, LCD0, LCD1, TV block	0x0000_0000
CLK_GATE_SCLK_CAM	0xC820	Control special clock gating for CAM_BLK	0xFFFF_FFFF
CLK_GATE_IP_CAM	0xC920	Control IP clock gating for CAM_BLK	0xFFFF_FFFF
CLK_GATE_IP_TV	0xC924	Control IP clock gating for TV_BLK	0xFFFF_FFFF
CLK_GATE_IP_MFC	0xC928	Control IP clock gating for MFC_BLK	0xFFFF_FFFF
CLK_GATE_IP_G3D	0xC92C	Control IP clock gating for G3D_BLK	0xFFFF_FFFF
CLK_GATE_IP_IMAGE	0xC930	Control IP clock gating for IMAGE_BLK	0xFFFF_FFFF
CLK_GATE_IP_LCD0	0xC934	Control IP clock gating for LCD0_BLK	0xFFFF_FFFF
CLK_GATE_IP_LCD1	0xC938	Control IP clock gating for LCD1_BLK	0xFFFF_FFFF
CLK_GATE_IP_FSYS	0xC940	Control IP clock gating for FSYS_BLK	0xFFFF_FFFF
CLK_GATE_IP_GPS	0xC94C	Control IP clock gating for GPS_BLK	0xFFFF_FFFF
CLK_GATE_IP_PERIL	0xC950	Control IP clock gating for PERIL_BLK	0xFFFF_FFFF
CLK_GATE_IP_PERIR	0xC960	Control IP clock gating for PERIR_BLK	0xFFFF_FFFF
CLK_GATE_BLOCK	0xC970	Control block clock gating	0xFFFF_FFFF
CLKOUT_CMU_TOP	0xCA00	CLKOUT control register	0x0001_0000
CLKOUT_CMU_TOP_DIV_STAT	0xCA04	Clock divider status for CLKOUT	0x0000_0000

- Base Address: 0x1004_0000

Register	Offset	Description	Reset Value
CLK_SRC_DMC	0x0200	Select clock source for CMU_DMC	0x0001_0000
CLK_SRC_MASK_DMC	0x0300	Clock source mask for DMC_BLK	0x0001_0000
CLK_MUX_STAT_DMC	0x0400	Clock MUX status for CMU_DMC	0x1110_0110
CLK_DIV_DMC0	0x0500	Set clock divider ratio for CMU_DMC	0x0000_0000
CLK_DIV_DMC1	0x0504	Set clock divider ratio for CMU_DMC	0x0000_0000
CLK_DIV_STAT_DMC0	0x0600	Clock divider status for CMU_DMC	0x0000_0000
CLK_DIV_STAT_DMC1	0x0604	Clock divider status for CMU_DMC	0x0000_0000
CLK_GATE_IP_DMC	0x0900	Control IP clock gating for DMC_BLK	0xFFFF_FFFF
CLKOUT_CMU_DMC	0x0A00	CLKOUT control register	0x0001_0000
CLKOUT_CMU_DMC_DIV_STAT	0x0A04	Clock divider status for CLKOUT	0x0000_0000
DCGIDX_MAP0	0x1000	DCG index map 0	0xFFFF_FFFF
DCGIDX_MAP1	0x1004	DCG index map 1	0xFFFF_FFFF
DCGIDX_MAP2	0x1008	DCG index map 2	0xFFFF_FFFF
DCGPERF_MAP0	0x1020	DCG performance map 0	0xFFFF_FFFF
DCGPERF_MAP1	0x1024	DCG performance map 1	0xFFFF_FFFF
DVCIDX_MAP	0x1040	DVC index map	0x0OFF_FFFF
FREQ_CPU	0x1060	Maximum frequency of CPU	0x0000_0000
FREQ_DPM	0x1064	Frequency of DPM accumulators	0x0000_0000
DVSEMCLK_EN	0x1080	DVS emulation clock enable	0x0000_0000
MAXPERF	0x1084	MAX performance enable	0x0000_0000
APLL_LOCK	0x4000	Control PLL locking period for APPLL.	0x0000_0FFF
MPLL_LOCK	0x4008	Control PLL locking period for MPLL.	0x0000_0FFF
APLL_CON0	0x4100	Control PLL output frequency for APPLL.	0x00C8_0601
APLL_CON1	0x4104	Control PLL AFC	0x0000_001C
MPLL_CON0	0x4108	Control PLL output frequency for MPLL.	0x00C8_0601
MPLL_CON1	0x410C	Control PLL AFC	0x0000_001C
CLK_SRC_CPU	0x4200	Select clock source for CMU_CPU	0x0000_0000
CLK_MUX_STAT_CPU	0x4400	Clock MUX status for CMU_CPU	0x0011_0101
CLK_DIV_CPU0	0x4500	Set clock divider ratio for CMU_CPU	0x0000_0000
CLK_DIV_CPU1	0x4504	Set clock divider ratio for CMU_CPU	0x0000_0000
CLK_DIV_STAT_CPU0	0x4600	Clock divider status for CMU_CPU	0x0000_0000
CLK_DIV_STAT_CPU1	0x4604	Clock divider status for CMU_CPU	0x0000_0000
CLK_GATE_SCLK_CPU	0x4800	Control special clock gating for CMU_CPU	0xFFFF_FFFF
CLK_GATE_IP_CPU	0x4900	Control IP clock gating for CMU_CPU	0xFFFF_FFFF
CLKOUT_CMU_CPU	0x4A00	CLKOUT control register	0x0001_0000

Register	Offset	Description	Reset Value
CLKOUT_CMU_CPU_DIV_STAT	0x4A04	Clock divider status for CLKOUT	0x0000_0000
ARMCLK_STOPCTRL	0x5000	ARM Clock Stop Control Register. Cycle is counted by SCLK_APLL	0x0000_0044
ATCLK_STOPCTRL	0x5004	ATCLK Stop Control Register. Cycle is counted by SCLK_APLL	0x0000_0044
PARITYFAIL_STATUS	0x5010	PARITYFAIL status register	0x0000_0000
PARITYFAIL_CLEAR	0x5014	PARITYFAIL status register	0x0000_0000
PWR_CTRL	0x5020	Power control register	0x0000_0033
APLL_CON0_L8	0x5100	APLL control (performance level-8)	0x00C8_0601
APLL_CON0_L7	0x5104	APLL control (performance level-7)	0x00C8_0601
APLL_CON0_L6	0x5108	APLL control (performance level-6)	0x00C8_0601
APLL_CON0_L5	0x510C	APLL control (performance level-5)	0x00C8_0601
APLL_CON0_L4	0x5110	APLL control (performance level-4)	0x00C8_0601
APLL_CON0_L3	0x5114	APLL control (performance level-3)	0x00C8_0601
APLL_CON0_L2	0x5118	APLL control (performance level-2)	0x00C8_0601
APLL_CON0_L1	0x511C	APLL control (performance level-1)	0x00C8_0601
IEM_CONTROL	0x5120	Control IEM	0x0000_0000
APLL_CON1_L8	0x5200	Control PLL AFC (performance level-1)	0x0000_0000
APLL_CON1_L7	0x5204	Control PLL AFC (performance level-7)	0x0000_0000
APLL_CON1_L6	0x5208	Control PLL AFC (performance level-6)	0x0000_0000
APLL_CON1_L5	0x520C	Control PLL AFC (performance level-5)	0x0000_0000
APLL_CON1_L4	0x5210	Control PLL AFC (performance level-4)	0x0000_0000
APLL_CON1_L3	0x5214	Control PLL AFC (performance level-3)	0x0000_0000
APLL_CON1_L2	0x5218	Control PLL AFC (performance level-2)	0x0000_0000
APLL_CON1_L1	0x521C	Control PLL AFC (performance level-1)	0x0000_0000
CLKDIV_IEM_L8	0x5300	Clock divider for IEM (performance level-8)	0x0000_0000
CLKDIV_IEM_L7	0x5304	Clock divider for IEM (performance level-7)	0x0000_0000
CLKDIV_IEM_L6	0x5308	Clock divider for IEM (performance level-6)	0x0000_0000
CLKDIV_IEM_L5	0x530C	Clock divider for IEM (performance level-5)	0x0000_0000
CLKDIV_IEM_L4	0x5310	Clock divider for IEM (performance level-4)	0x0000_0000
CLKDIV_IEM_L3	0x5314	Clock divider for IEM (performance level-3)	0x0000_0000
CLKDIV_IEM_L2	0x5318	Clock divider for IEM (performance level-2)	0x0000_0000
CLKDIV_IEM_L1	0x531C	Clock divider for IEM (performance level-1)	0x0000_0000

SFRs consist of five parts.

The SFRs with address 0x1003_4000 to 0x1003_7FFF controls clock-related logics for LEFTBUS block. They control clock source selection, clock divider ratio, and clock gating.

The SFRs with address 0x1003_8000 to 0x1003_BFFF controls clock-related logics for RIGHTBUS block. They control clock source selection, clock divider ratio, and clock gating.

The SFRs with address 0x1003_C000 to 0x1003_FFFF controls clock-related logics for MFC, G3D, TV, LCD0, LCD1, CAM, FSYS, PERIL, and PERIR blocks. They control EPLL and VPLL, clock source selection, clock divider ratio, and clock gating.

The SFRs with address 0x1004_0000 to 0x1004_3FFF controls clock-related logics for DMC block. They control clock source selection, clock divider ratio, and clock gating.

The SFRs with address 0x1004_4000 to 0x1004_7FFF controls clock-related logics for CPU block. They control APLL and MPLL, clock source selection, clock divider ratio, CPU-related logics, and IEM.

5.3.1.1 CLK_SRC_LEFTBUS

- Address = 0x1003_4200, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	-	Reserved	0x0
MUX_GDL_SEL	[0]	RW	Control MUXGDL 0 = SCLKMPPLL 1 = SCLKAPLL	0x0

5.3.1.2 CLK_MUX_STAT_LEFTBUS

- Address = 0x1003_4400, Reset Value = 0x0000_0001

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	-	Reserved	0x0
GDL_SEL	[2:0]	R	Selection signal status of MUXGDL 001 = SCLKMPPLL 010 = SCLKAPLL 1xx = On changing	0x1

5.3.1.3 CLK_DIV_LEFTBUS

- Address = 0x1003_4500, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:7]	-	Reserved	0x0
GPL_RATIO	[6:4]	RW	DIVGPL clock divider ratio ACLK_GPL = MOUTGPL/(GPL_RATIO + 1)	0x0
RSVD	[3]	-	Reserved	0x0
GDL_RATIO	[2:0]	RW	DIVGDL clock divider ratio ACLK_GDL = MOUTGDL/(GDL_RATIO + 1)	0x0

5.3.1.4 CLK_DIV_STAT_LEFTBUS

- Address = 0x1003_4600, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	-	Reserved	0x0
DIV_GPL	[4]	R	DIVGPL status 0 = Stable 1 = Divider is changing	0x0
RSVD	[3:1]	-	Reserved	0x0
DIV_GDL	[0]	R	DIVGDL status 0 = Stable 1 = Divider is changing	0x0

5.3.1.5 CLK_GATE_IP_LEFTBUS

- Address = 0x1003_4800, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:2]	-	Reserved	0x3FFFFFFF
CLK_PPMULEFT	[1]	RW	Gating all clocks for PPMULEFT 0 = Mask 1 = Pass	0x1
RSVD	[0]	-	Reserved	0x1

5.3.1.6 CLKOUT_CMU_LEFTBUS

- Address = 0x1003_4A00, Reset Value = 0x0001_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:17]	-	Reserved	0x0
ENB_CLKOUT	[16]	RW	Enable CLKOUT 0 = Disable 1 = Enable	0x1
RSVD	[15:14]	-	Reserved	0x0
DIV_RATIO	[13:8]	RW	Divide ratio (Divide ratio = DIV_RATIO + 1)	0x0
RSVD	[7:5]	-	Reserved	0x0
MUX_SEL	[4:0]	RW	00000 = SCLK_MPLL/2 00001 = SCLK_APLL/2 00010 = ACLK_GDL 00011 = ACLK_GPL	0x0

5.3.1.7 CLKOUT_CMU_LEFTBUS_DIV_STAT

- Address = 0x1003_4A04, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	—	Reserved	0x0
DIV_STAT	[0]	R	DIVCLKOUT status 0 = Stable 1 = Divider is changing	0x0

5.3.1.8 CLK_SRC_RIGHTBUS

- Address = 0x1003_8200, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	—	Reserved	0x0
MUX_GDR_SEL	[0]	RW	Control MUXGDR 0 = SCLKMPLL 1 = SCLKAPLL	0x0

5.3.1.9 CLK_MUX_STAT_RIGHTBUS

- Address = 0x1003_8400, Reset Value = 0x0000_0001

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	—	Reserved	0x0
GDR_SEL	[2:0]	R	Selection signal status of MUXGDR 001 = SCLKMPLL 010 = SCLKAPLL 1xx = On changing	0x1

5.3.1.10 CLK_DIV_RIGHTBUS

- Address = 0x1003_8500, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:7]	—	Reserved	0x0
GPR_RATIO	[6:4]	RW	DIVGPR clock divider ratio ACLK_GPR = MOUTGPR / (GPR_RATIO + 1)	0x0
RSVD	[3]	—	Reserved	0x0
GDR_RATIO	[2:0]	RW	DIVGDR clock divider ratio ACLK_GDR = MOUTGDR / (GDR_RATIO + 1)	0x0

5.3.1.11 CLK_DIV_STAT_RIGHTBUS

- Address = 0x1003_8600, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	–	Reserved	0x0
DIV_GPR	[4]	R	DIVGPR status 0 = Stable 1 = Divider is changing	0x0
RSVD	[3:1]	–	Reserved	0x0
DIV_GDR	[0]	R	DIVGDR status 0 = Stable 1 = Divider is changing	0x0

5.3.1.12 CLK_GATE_IP_RIGHTBUS

- Address = 0x1003_8800, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:2]	–	Reserved	0x3FFFFFFF
CLK_PPMURIGHT	[1]	RW	Gating all clocks for PPMURIGHT 0 = Mask 1 = Pass	0x1
RSVD	[0]	–	Reserved	0x1

5.3.1.13 CLKOUT_CMU_RIGHTBUS

- Address = 0x1003_8A00, Reset Value = 0x0001_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:17]	–	Reserved	0x0
ENB_CLKOUT	[16]	RW	Enable CLKOUT 0 = Disable 1 = Enable	0x1
RSVD	[15:14]	–	Reserved	0x0
DIV_RATIO	[13:8]	RW	Divide ratio (Divide ratio = DIV_RATIO + 1)	0x0
RSVD	[7:5]	–	Reserved	0x0
MUX_SEL	[4:0]	RW	00000 = SCLK_MPLL/2 00001 = SCLK_APLL/2 00010 = ACLK_GDR 00011 = ACLK_GPR	0x0

5.3.1.14 CLKOUT_CMU_RIGHTBUS_DIV_STAT

- Address = 0x1003_8A04, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	–	Reserved	0x0
DIV_STAT	[0]	R	DIVCLKOUT status 0 = Stable 1 = Divider is changing	0x0

5.3.1.15 EPLL_LOCK

- Address = 0x1003_C010, Reset Value = 0x0000_0FFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	–	Reserved	0x0
PLL_LOCKTIME	[15:0]	RW	Required period to generate a stable clock output	0xFFFF

5.3.1.16 VPLL_LOCK

- Address = 0x1003_C020, Reset Value = 0x0000_0FFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	–	Reserved	0x0
PLL_LOCKTIME	[15:0]	RW	Required period to generate a stable clock output	0xFFFF

Lock time of PLL is related Fref which is FIN/PDIV

- 1) Lock time of EPLL is maximum 3000 cycles of Fref
300 us@Fref = 10 MHz, 750 us@Fref = 4 MHz, 375 us@Fref = 8 MHz
- 2) Lock time of VPLL is maximum 750 us

The maximum lock time means that the waiting time until locking takes at least 750 us in the worst case regardless of FREF. Therefore, the user of this PLL must wait for more than 750 us unconditionally before the PLL is locked. (Waiting time before locking \geq 750 us)

5.3.1.17 EPLL_CON0

- Address = 0x1003_C110, Reset Value = 0x0030_0301

Name	Bit	Type	Description	Reset Value
ENABLE	[31]	RW	PLL enable control 0 = Disable 1 = Enable	0x0
RSVD	[30]	-	Reserved	0x0
LOCKED	[29]	R	PLL locking indication 0 = Unlocked 1 = Locked	0x0
RSVD	[28]	-	Reserved	0x0
VSEL	[27]	RW	VCO frequency range selection VSEL = 0 for 330 MHz <= FVCO <= 460 MHz VSEL = 1 for 460 MHz <= FVCO <= 660 MHz	0x0
RSVD	[26:25]	-	Reserved	0x0
MDIV	[24:16]	RW	PLL M divide value	0x30
RSVD	[15:14]	-	Reserved	0x0
PDIV	[13:8]	RW	PLL P divide value	0x3
RSVD	[7:3]	-	Reserved	0x0
SDIV	[2:0]	RW	PLL S divide value	0x1

The reset value of EPLL_CON0 generates 192 MHz output clock for the input clock frequency of 24 MHz.

Equation to calculate the output frequency:

$$F_{OUT} = (MDIV + K/65536) \times F_{IN} / (PDIV \times 2^{SDIV})$$

where MDIV, PDIV, SDIV, and K must meet the following conditions:

PDIV: $1 \leq PDIV \leq 63$

MDIV: $16 \leq MDIV \leq 511$

SDIV: $0 \leq SDIV \leq 5$

K: $0 \leq K \leq 65535$

F_{ref} ($= F_{IN} / PDIV$): $4 \text{ MHz} \leq F_{ref} \leq 30 \text{ MHz}$

FVCO ($= MDIV \times F_{IN} / PDIV$):

$330 \text{ MHz} \leq F_{VCO} \leq 460 \text{ MHz}$ when VSEL = LOW

$460 \text{ MHz} \leq F_{VCO} \leq 660 \text{ MHz}$ when VSEL = HIGH

F_{OUT} : $12 \text{ MHz} \leq F_{OUT} \leq 660 \text{ MHz}$

Don't set the value PDIV or MDIV to all zeros.

Refer to [5.2.2.2 Recommended PMS Value for EPLL](#) for recommended PMS values.

5.3.1.18 EPLL_CON1

- Address = 0x1003_C114, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	–	Reserved	0x0
K	[15:0]	RW	PLL DSM (Delta-Sigma Modulator)	0x0

Refer to [5.2.2.2 Recommended PMS Value for EPLL](#) for recommended K value.

5.3.1.19 VPLL_CON0

- Address = 0x1003_C120, Reset Value = 0x0024_0201

Name	Bit	Type	Description	Reset Value
ENABLE	[31]	RW	PLL enable control 0 = Disable 1 = Enable	0x0
RSVD	[30]	-	Reserved	0x0
LOCKED	[29]	R	PLL locking indication 0 = Unlocked 1 = Locked	0x0
RSVD	[28]	-	Reserved	0x0
VSEL	[27]	RW	VCO frequency range selection VSEL = 0 for 330 MHz <= FVCO <= 460 MHz VSEL = 1 for 460 MHz <= FVCO <= 660 MHz	0x0
RSVD	[26:25]	-	Reserved	0x0
MDIV	[24:16]	RW	PLL M divide value	0x24
RSVD	[15:14]	-	Reserved	0x0
PDIV	[13:8]	RW	PLL P divide value	0x2
RSVD	[7:3]	-	Reserved	0x0
SDIV	[2:0]	RW	PLL S divide value	0x1

The reset value of VPLL_CON0 generates 222.75 MHz output clock for the input clock frequency of 24 MHz.

Equation to calculate the output frequency:

$$F_{OUT} = (MDIV + K/1024) \times F_{IN} / (PDIV \times 2^{SDIV})$$

where MDIV, PDIV, SDIV, and K must meet the following conditions:

PDIV: $1 \leq PDIV \leq 63$

MDIV: $23 \leq MDIV \leq 511$

SDIV: $0 \leq SDIV \leq 5$

K: $0 \leq K \leq 3072$

Fref (= FIN / PDIV): $2 \text{ MHz} \leq F_{ref} \leq 30 \text{ MHz}$

FVCO (= MDIV × FIN / PDIV):

$330 \text{ MHz} \leq F_{VCO} \leq 460 \text{ MHz}$ when VSEL = LOW

$460 \text{ MHz} \leq F_{VCO} \leq 660 \text{ MHz}$ when VSEL = HIGH

FOUT: $11 \text{ MHz} \leq F_{OUT} \leq 660 \text{ MHz}$

Don't set the value PDIV or MDIV to all zeros.

Refer to [5.2.2.3 Recommended PMS Value for VPLL](#)

5.3.1.20 VPLL_CON1

- Address = 0x1003_C124, Reset Value = 0x6601_0464

Name	Bit	Type	Description	Reset Value
SSCG_EN	[31]	RW	0 = Dithered mode is disabled 1 = Dithered mode is enabled	0x0
SEL_PF	[30:29]	RW	Modulation Method Control 00 = Down spread 01 = Up spread 1x = Center spread	0x3
MRR	[28:24]	RW	Modulation Rate Control	0x6
RSVD	[23:22]	-	Reserved	0x0
MFR	[21:16]	RW	Modulation Frequency Control	0x1
RSVD	[15:12]	-	Reserved	0x0
K	[11:0]	RW	PLL DSM (Delta-Sigma Modulator)	0x464

Modulation Frequency, MF, is calculated by the equation:

$$MF = 1 / [4 / (FIN / PDIV) \times 32 \times (MFR + 1)]$$

Modulation Rate, MR, is calculated by the equation:

$$MR = [(32 \times (MFR+1) \times MRR) / 1024] / MDIV \times 2$$

where MFR and MRR must meet the following conditions:

$$MFR: 0 \leq MFR \leq 47$$

$$MRR: 1 \leq MRR \leq 31$$

$$1 \leq MRR \times (MFR+1) \leq 48$$

5.3.1.21 CLK_SRC_TOP0

- Address = 0x1003_C210, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:29]	–	Reserved	0x0
MUX_ONENAND_SEL	[28]	RW	Control MUXONENAND 0 = ACLK_133 1 = ACLK_160	0x0
RSVD	[27:25]	–	Reserved	0x0
MUX_ACLK_133_SEL	[24]	RW	Control MUXACLK_133 0 = SCLKMPPLL 1 = SCLKAPLL	0x0
RSVD	[23:21]	–	Reserved	0x0
MUX_ACLK_160_SEL	[20]	RW	Control MUXACLK_160 0 = SCLKMPPLL 1 = SCLKAPLL	0x0
RSVD	[19:17]	–	Reserved	0x0
MUX_ACLK_100_SEL	[16]	RW	Control MUXACLK_100 0 = SCLKMPPLL 1 = SCLKAPLL	0x0
RSVD	[15:13]	–	Reserved	0x0
MUX_ACLK_200_SEL	[12]	RW	Control MUXACLK_200 0 = SCLKMPPLL 1 = SCLKAPLL	0x0
RSVD	[11:9]	–	Reserved	0x0
MUX_VPLL_SEL	[8]	RW	Control MUXVPLL 0 = FINPLL 1 = FOUTVPLL	0x0
RSVD	[7:5]	–	Reserved	0x0
MUX_EPLL_SEL	[4]	RW	Control MUXEPLL 0 = FINPLL 1 = FOUTEPLL	0x0
RSVD	[3:1]	–	Reserved	0x0
MUX_ONENAND_1_SEL	[0]	RW	Control MUXONENAND_1 0 = MOUTONENAND 1 = SCLKVPLL	0x0

5.3.1.22 CLK_SRC_TOP1

- Address = 0x1003_C214, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	–	Reserved	0x0
VPLLSRC_SEL	[0]	RW	Control MUXVPLLSRC 0 = FINPLL 1 = SCLKHDMI24M	0x0

5.3.1.23 CLK_SRC_CAM

- Address = 0x1003_C220, Reset Value = 0x1111_1111

Name	Bit	Type	Description	Reset Value
CSIS1_SEL	[31:28]	RW	Control MUXCSIS1, which is the source clock of CSIS1 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1
CSIS0_SEL	[27:24]	RW	Control MUXCSIS0, which is the source clock of CSIS0 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1
CAM1_SEL	[23:20]	RW	Control MUXCAM1, which is the source clock of CAM_B_CLKOUT 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1

Name	Bit	Type	Description	Reset Value
CAM0_SEL	[19:16]	RW	Control MUXCAM0, which is the source clock of CAM_A_CLKOUT 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1
FIMC3_LCLK_SEL	[15:12]	RW	Control MUXFIMC3_LCLK, which is the source clock of FIMC3 local clock 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1
FIMC2_LCLK_SEL	[11:8]	RW	Control MUXFIMC2_LCLK, which is the source clock of FIMC2 local clock 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1
FIMC1_LCLK_SEL	[7:4]	RW	Control MUXFIMC1_LCLK, which is the source clock of FIMC1 local clock 0000 = XXTI 0001 = XusbXTI, 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1
FIMC0_LCLK_SEL	[3:0]	RW	Control MUXFIMC0_LCLK, which is the source clock of FIMC0 local clock	0x1

Name	Bit	Type	Description	Reset Value
			0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	

5.3.1.24 CLK_SRC_TV

- Address = 0x1003_C224, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:9]	–	Reserved	0x0
DAC_SEL	[8]	RW	Control MUXDAC, which is the source clock of TVENC and DAC 0 = SCLKVPLL 1 = SCLK_HDMIPHY	0x0
RSVD	[7:5]	–	Reserved	0x0
MIXER_SEL	[4]	RW	Control MUXMIXER, which is the source clock of MIXER 0 = SCLK_DAC 1 = SCLK_HDMI	0x0
RSVD	[3:1]	–	Reserved	0x0
HDMI_SEL	[0]	RW	Control MUXHDMI, which is the source clock of HDMI link 0 = SCLK_PIXEL 1 = SCLK_HDMIPHY	0x0

5.3.1.25 CLK_SRC_MFC

- Address = 0x1003_C228, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:9]	–	Reserved	0x0
MFC_SEL	[8]	RW	Control MUXMFC, which is the source clock of MFC core 0 = MOUTMFC_0 1 = MOUTMFC_1	0x0
RSVD	[7:5]	–	Reserved	0x0
MFC_1_SEL	[4]	RW	Control MUXMFC_1, which is the source clock of MFC core 0 = SCLKEPLL 1 = SCLKVPLL	0x0
RSVD	[3:1]	–	Reserved	0x0
MFC_0_SEL	[0]	RW	Control MUXMFC_0, which is the source clock of MFC core 0 = SCLKMPPLL 1 = SCLKAPLL	0x0

5.3.1.26 CLK_SRC_G3D

- Address = 0x1003_C22C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:9]	–	Reserved	0x0
G3D_SEL	[8]	RW	Control MUXG3D, which is the source clock of G3D core 0 = MOUTG3D_0 1 = MOUTG3D_1	0x0
RSVD	[7:5]	–	Reserved	0x0
G3D_1_SEL	[4]	RW	Control MUXG3D_1, which is the source clock of G3D core 0 = SCLKEPLL 1 = SCLKVPLL	0x0
RSVD	[3:1]	–	Reserved	0x0
G3D_0_SEL	[0]	RW	Control MUXG3D_0, which is the source clock of G3D core 0 = SCLKMPLL 1 = SCLKAPLL	0x0

5.3.1.27 CLK_SRC_IMAGE

- Address = 0x1003_C230, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:9]	–	Reserved	0x0
G2D_SEL	[8]	RW	Control MUXG2D, which is the source clock of G2D core 0 = MOUTG2D_0 1 = MOUTG2D_1	0x0
RSVD	[7:5]	–	Reserved	0x0
G2D_1_SEL	[4]	RW	Control MUXG2D_1, which is the source clock of G2D core 0 = SCLKEPLL 1 = SCLKVPLL	0x0
RSVD	[3:1]	–	Reserved	0x0
G2D_0_SEL	[0]	RW	Control MUXG2D_0, which is the source clock of G2D core 0 = SCLKMPLL 1 = SCLKAPLL	0x0

5.3.1.28 CLK_SRC_LCD0

- Address = 0x1003_C234, Reset Value = 0x0000_1111

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	-	Reserved	0x0
MIPIO_SEL	[15:12]	RW	Control MUXMIPIO, which is the source clock of MIPI_DSIM0 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPH 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1
RSVD	[11:8]	RW	Reserved	0x1
RSVD	[7:4]	RW	Reserved	0x1
FIMD0_SEL	[3:0]	RW	Control MUXFIMD0, which is the source clock of FIMD0 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPH 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1

5.3.1.29 CLK_SRC_LCD1

- Address = 0x1003_C238, Reset Value = 0x0000_1111

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	-	Reserved	0x0
MIPI1_SEL	[15:12]	RW	Control MUXMIP1, which is the source clock of MIPI_DSIM1 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPH 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1
RSVD	[11:8]	RW	Reserved	0x1
RSVD	[7:4]	RW	Reserved	0x1
FIMD1_SEL	[3:0]	RW	Control MUXFIMD1, which is the source clock of FIMD1 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPH 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1

5.3.1.30 CLK_SRC_MAUDIO

- Address = 0x1003_C23C, Reset Value = 0x0000_0005

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	-	Reserved	0x0
AUDIO0_SEL	[3:0]	RW	Control MUXAUDIO0, which is the source clock of AUDIO0 0000 = AUDIOCDCLK0 0001 = reserved 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = XXTI 0101 = XusbXTI 0110 = SCLKMPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x5

5.3.1.31 CLK_SRC_FSYS

- Address = 0x1003_C240, Reset Value = 0x0001_1111

Name	Bit	Type	Description	Reset Value
RSVD	[31:25]	—	Reserved	0x0
SATA_SEL	[24]	RW	Control MUXSATA, which is the source clock of SATA 0 = SCLKMPPLL 1 = SCLKAPLL	0x0
RSVD	[23:20]	—	Reserved	0x0
MMC4_SEL	[19:16]	RW	Control MUXMMC4, which is the source clock of MMC4 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1
MMC3_SEL	[15:12]	RW	Control MUXMMC3, which is the source clock of MMC3 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1
MMC2_SEL	[11:8]	RW	Control MUXMMC2, which is the source clock of MMC2 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1
MMC1_SEL	[7:4]	RW	Control MUXMMC1, which is the source clock of MMC1 0000 = XXTI 0001 = XusbXTI	0x1

Name	Bit	Type	Description	Reset Value
			0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	
MMC0_SEL	[3:0]	RW	Control MUXMMC0, which is the source clock of MMC0 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1

5.3.1.32 CLK_SRC_PERILO

- Address = 0x1003_C250, Reset Value = 0x0001_1111

Name	Bit	Type	Description	Reset Value
RSVD	[31:20]	-	Reserved (should be 1'b1)	0x0
UART4_SEL	[19:16]	RW	Control MUXUART4, which is the source clock of UART4 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1
UART3_SEL	[15:12]	RW	Control MUXUART3, which is the source clock of UART3 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1
UART2_SEL	[11:8]	RW	Control MUXUART2, which is the source clock of UART2 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1
UART1_SEL	[7:4]	RW	Control MUXUART1, which is the source clock of UART1 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL	0x1

Name	Bit	Type	Description	Reset Value
			1000 = SCLKVPLL Others = Reserved	
UART0_SEL	[3:0]	RW	Control MUXUART0, which is the source clock of UART0 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1

5.3.1.33 CLK_SRC_PERIL1

- Address = 0x1003_C254, Reset Value = 0x0111_0055

Name	Bit	Type	Description	Reset Value
RSVD	[31:28]	-	Reserved	0x0
SPI2_SEL	[27:24]	RW	Control MUXSPI2, which is the source clock of SPI2 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1
SPI1_SEL	[23:20]	RW	Control MUXSPI1, which is the source clock of SPI1 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1
SPI0_SEL	[19:16]	RW	Control MUXSPI0, which is the source clock of SPI0 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x1
RSVD	[15:10]	-	Reserved	0x0
SPDIF_SEL	[9:8]	RW	Control MUXSPDIF, which is the source clock of SPDIF 00 = SCLK_AUDIO0 01 = SCLK_AUDIO1 10 = SCLK_AUDIO2 11 = SPDIF_EXTCLK	0x0
AUDIO2_SEL	[7:4]	RW	Control MUXAUDIO2, which is the source clock of AUDIO2	0x5

Name	Bit	Type	Description	Reset Value
			0000 = AUDIOCDCLK2 0001 = reserved 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = XXTI 0101 = XusbXTI 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	
AUDIO1_SEL	[3:0]	RW	Control MUXAUDIO1, which is the source clock of AUDIO1 0000 = AUDIOCDCLK1 0001 = reserved 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = XXTI 0101 = XusbXTI 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others = Reserved	0x5

5.3.1.34 CLK_SRC_MASK_TOP

- Address = 0x1003_C310, Reset Value = 0x0000_0001

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	-	Reserved	0x0
VPLLSRC_MASK	[0]	RW	Mask output clock of MUXVPLLSRC 0 = Mask 1 = Unmask	0x1

5.3.1.35 CLK_SRC_MASK_CAM

- Address = 0x1003_C320, Reset Value = 0x1111_1111

Name	Bit	Type	Description	Reset Value
RSVD	[31:29]	–	Reserved	0x0
CSIS1_MASK	[28]	RW	Mask output clock of MUXCSIS1 0 = Mask 1 = Unmask	0x1
RSVD	[27:25]	–	Reserved	0x0
CSIS0_MASK	[24]	RW	Mask output clock of MUXCSIS0 0 = Mask 1 = Unmask	0x1
RSVD	[23:21]	–	Reserved	0x0
CAM1_MASK	[20]	RW	Mask output clock of MUXCAM1 0 = Mask 1 = Unmask	0x1
RSVD	[19:17]	–	Reserved	0x0
CAM0_MASK	[16]	RW	Mask output clock of MUXCAM0 0 = Mask 1 = Unmask	0x1
RSVD	[15:13]	–	Reserved	0x0
FIMC3_LCLK_MASK	[12]	RW	Mask output clock of MUXFIMC3_LCLK 0 = Mask 1 = Unmask	0x1
RSVD	[11:9]	–	Reserved	0x0
FIMC2_LCLK_MASK	[8]	RW	Mask output clock of MUXFIMC2_LCLK 0 = Mask 1 = Unmask	0x1
RSVD	[7:5]	–	Reserved	0x0
FIMC1_LCLK_MASK	[4]	RW	Mask output clock of MUXFIMC1_LCLK 0 = Mask 1 = Unmask	0x1
RSVD	[3:1]	–	Reserved	0x0
FIMC0_LCLK_MASK	[0]	RW	Mask output clock of MUXFIMC0_LCLK 0 = Mask 1 = Unmask	0x1

5.3.1.36 CLK_SRC_MASK_TV

- Address = 0x1003_C324, Reset Value = 0x0000_0111

Name	Bit	Type	Description	Reset Value
RSVD	[31:9]	–	Reserved	0x0
DAC_MASK	[8]	RW	Mask output clock of MUXDAC 0 = Mask 1 = Unmask	0x1
RSVD	[7:5]	–	Reserved	0x0
MIXER_MASK	[4]	RW	Mask output clock of MUXMIXER 0 = Mask 1 = Unmask	0x1
RSVD	[3:1]	–	Reserved	0x0
HDMI_MASK	[0]	RW	Mask output clock of MUXHDMI 0 = Mask 1 = Unmask	0x1

5.3.1.37 CLK_SRC_MASK_LCD0

- Address = 0x1003_C334, Reset Value = 0x0000_1111

Name	Bit	Type	Description	Reset Value
RSVD	[31:13]	–	Reserved	0x0
MIPIO_MASK	[12]	RW	Mask output clock of MUXMIPIO 0 = Mask 1 = Unmask	0x1
RSVD	[11:9]	–	Reserved	0x0
RSVD	[8]	RW	Reserved	0x1
RSVD	[7:5]	–	Reserved	0x0
RSVD	[4]	RW	Reserved	0x1
RSVD	[3:1]	–	Reserved	0x0
FIMD0_MASK	[0]	RW	Mask output clock of MUXFIMD0 0 = Mask 1 = Unmask	0x1

5.3.1.38 CLK_SRC_MASK_LCD1

- Address = 0x1003_C338, Reset Value = 0x0000_1111

Name	Bit	Type	Description	Reset Value
RSVD	[31:13]	–	Reserved	0x0
MIPI1_MASK	[12]	RW	Mask output clock of MUXMIPI1 0 = Mask 1 = Unmask	0x1
RSVD	[11:9]	–	Reserved	0x0
RSVD	[8]	RW	Reserved	0x1
RSVD	[7:5]	–	Reserved	0x0
RSVD	[4]	RW	Reserved	0x1
RSVD	[3:1]	–	Reserved	0x0
FIMD1_MASK	[0]	RW	Mask output clock of MUXFIMD1 0 = Mask 1 = Unmask	0x1

5.3.1.39 CLK_SRC_MASK_MAUDIO

- Address = 0x1003_C33C, Reset Value = 0x0000_0001

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	–	Reserved	0x0
AUDIO0_MASK	[0]	RW	Mask output clock of MUXAUDIO0 0 = Mask 1 = Unmask	0x1

5.3.1.40 CLK_SRC_MASK_FSYS

- Address = 0x1003_C340, Reset Value = 0x0101_1111

Name	Bit	Type	Description	Reset Value
RSVD	[31:25]	–	Reserved	0x0
SATA_MASK	[24]	RW	Mask output clock of MUXSATA 0 = Mask 1 = Unmask	0x1
RSVD	[23:17]	–	Reserved	0x0
MMC4_MASK	[16]	RW	Mask output clock of MUXMMC4 0 = Mask 1 = Unmask	0x1
RSVD	[15:13]	–	Reserved	0x0
MMC3_MASK	[12]	RW	Mask output clock of MUXMMC3 0 = Mask 1 = Unmask	0x1
RSVD	[11:9]	–	Reserved	0x0
MMC2_MASK	[8]	RW	Mask output clock of MUXMMC2 0 = Mask 1 = Unmask	0x1
RSVD	[7:5]	–	Reserved	0x0
MMC1_MASK	[4]	RW	Mask output clock of MUXMMC1 0 = Mask 1 = Unmask	0x1
RSVD	[3:1]	–	Reserved	0x0
MMC0_MASK	[0]	RW	Mask output clock of MUXMMC0 0 = Mask 1 = Unmask	0x1

5.3.1.41 CLK_SRC_MASK_PERI0

- Address = 0x1003_C350, Reset Value = 0x0001_1111

Name	Bit	Type	Description	Reset Value
RSVD	[31:17]	–	Reserved	0x0
UART4_MASK	[16]	RW	Mask output clock of MUXUART4 0 = Mask 1 = Unmask	0x1
RSVD	[15:13]	–	Reserved	0x0
UART3_MASK	[12]	RW	Mask output clock of MUXUART3 0 = Mask 1 = Unmask	0x1
RSVD	[11:9]	–	Reserved	0x0
UART2_MASK	[8]	RW	Mask output clock of MUXUART2 0 = Mask 1 = Unmask	0x1
RSVD	[7:5]	–	Reserved	0x0
UART1_MASK	[4]	RW	Mask output clock of MUXUART1 0 = Mask 1 = Unmask	0x1
RSVD	[3:1]	–	Reserved	0x0
UART0_MASK	[0]	RW	Mask output clock of MUXUART0 0 = Mask 1 = Unmask	0x1

5.3.1.42 CLK_SRC_MASK_PERI1

- Address = 0x1003_C354, Reset Value = 0x0111_0111

Name	Bit	Type	Description	Reset Value
RSVD	[31:25]	–	Reserved	0x0
SPI2_MASK	[24]	RW	Mask output clock of MUXSPI2 0 = Mask 1 = Unmask	0x1
RSVD	[23:21]	–	Reserved	0x0
SPI1_MASK	[20]	RW	Mask output clock of MUXSPI1 0 = Mask 1 = Unmask	0x1
RSVD	[19:17]	–	Reserved	0x0
SPI0_MASK	[16]	RW	Mask output clock of MUXSPI0 0 = Mask 1 = Unmask	0x1
RSVD	[15:9]	–	Reserved	0x0
SPDIF_MASK	[8]	RW	Mask output clock of MUXSPDIF 0 = Mask 1 = Unmask	0x1
RSVD	[7:5]	–	Reserved	0x0
AUDIO2_MASK	[4]	RW	Mask output clock of MUXAUDIO2 0 = Mask 1 = Unmask	0x1
RSVD	[3:1]	–	Reserved	0x0
AUDIO1_MASK	[0]	RW	Mask output clock of MUXAUDIO1 0 = Mask 1 = Unmask	0x1

5.3.1.43 CLK_MUX_STAT_TOP

- Address = 0x1003_C410, Reset Value = 0x1111_1111

Name	Bit	Type	Description	Reset Value
RSVD	[31]	–	Reserved	0x0
ONENAND_SEL	[30:28]	R	Selection signal status of MUXONENAND 001 = DOUT133 010 = DOUT166 1xx = On changing	0x1
RSVD	[27]	–	Reserved	0x0
ACLK_133_SEL	[26:24]	R	Selection signal status of MUXACLK_133 001 = SCLKMPLL 010 = SCLKAPLL 1xx = On changing	0x1
RSVD	[23]	–	Reserved	0x0
ACLK_160_SEL	[22:20]	R	Selection signal status of MUXACLK_160 001 = SCLKMPLL 010 = SCLKAPLL 1xx = On changing	0x1
RSVD	[19]	–	Reserved	0x0
ACLK_100_SEL	[18:16]	R	Selection signal status of MUXACLK_100 001 = SCLKMPLL 010 = SCLKAPLL 1xx = On changing	0x1
RSVD	[15]	–	Reserved	0x0
ACLK_200_SEL	[14:12]	R	Selection signal status of MUXACLK_200 001 = SCLKMPLL 010 = SCLKAPLL 1xx = On changing	0x1
RSVD	[11]	–	Reserved	0x0
VPLL_SEL	[10:8]	R	Selection signal status of MUXVPLL 001 = FINVPLL 010 = FOUTVPLL 1xx = On changing	0x1
RSVD	[7]	–	Reserved	0x0
EPLL_SEL	[6:4]	R	Selection signal status of MUXEPLL 001 = FINPLL 010 = FOUTEPLL 1xx = On changing	0x1
RSVD	[3]	–	Reserved	0x0
ONENAND_1_SEL	[2:0]	R	Selection signal status of MUXONENAND_1 001 = MOUTONENAND 010 = SCLKVPLL 1xx = On changing	0x1

5.3.1.44 CLK_MUX_STAT_MFC

- Address = 0x1003_C428, Reset Value = 0x0000_0111

Name	Bit	Type	Description	Reset Value
RSVD	[31:11]	–	Reserved	0x0
MFC_SEL	[10:8]	R	Selection signal status of MUXMFC 001 = MOUTMFC_0 010 = MOUTMFC_1 1xx = On changing	0x1
RSVD	[7]	–	Reserved	0x0
MFC_1_SEL	[6:4]	R	Selection signal status of MUXMFC_1 001 = SCLKEPLL 010 = SCLKVPLL 1xx = On changing	0x1
RSVD	[3]	–	Reserved	0x0
MFC_0_SEL	[2:0]	R	Selection signal status of MUXMFC_0 001 = SCLKMPLL 010 = SCLKAPLL 1xx = On changing	0x1

5.3.1.45 CLK_MUX_STAT_G3D

- Address = 0x1003_C42C, Reset Value = 0x0000_0111

Name	Bit	Type	Description	Reset Value
RSVD	[31:11]	–	Reserved	0x0
G3D_SEL	[10:8]	R	Selection signal status of MUXG3D 001 = MOUTG3D_0 010 = MOUTG3D_1 1xx = On changing	0x1
RSVD	[7]	–	Reserved	0x0
G3D_1_SEL	[6:4]	R	Selection signal status of MUXG3D_1 001 = SCLKEPLL 010 = SCLKVPLL 1xx = On changing	0x1
RSVD	[3]	–	Reserved	0x0
G3D_0_SEL	[2:0]	R	Selection signal status of MUXG3D_0 001 = SCLKMPLL 010 = SCLKAPLL 1xx = On changing	0x1

5.3.1.46 CLK_MUX_STAT_IMAGE

- Address = 0x1003_C430, Reset Value = 0x0000_0111

Name	Bit	Type	Description	Reset Value
RSVD	[31:11]	–	Reserved	0x0
G2D_SEL	[10:8]	R	Selection signal status of MUXG2D 001 = MOUTG2D_0 010 = MOUTG2D_1 1xx = On changing	0x1
RSVD	[7]	–	Reserved	0x0
G2D_1_SEL	[6:4]	R	Selection signal status of MUXG2D_1 001 = SCLKEPLL 010 = SCLKVPLL 1xx = On changing	0x1
RSVD	[3]	–	Reserved	0x0
G2D_0_SEL	[2:0]	R	Selection signal status of MUXG2D_0 001 = SCLKMPLL 010 = SCLKAPLL 1xx = On changing	0x1

5.3.1.47 CLK_DIV_TOP

- Address = 0x1003_C510, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:19]	–	Reserved	0x0
ONENAND_RATIO	[18:16]	RW	DIVONENAND clock divider ratio SCLK_ONENAND = MOUTONENAND / (ONENAND_RATIO + 1)	0x0
RSVD	[15]	–	Reserved	0x0
ACLK_133_RATIO	[14:12]	RW	DIVACLK_133 clock divider ratio ACLK133 = MOUTACLK_133 / (ACLK_133_RATIO + 1)	0x0
RSVD	[11]	–	Reserved	0x0
ACLK_160_RATIO	[10:8]	RW	DIVACLK_160 clock divider ratio ACLK166 = MOUTACLK_160 / (ACLK_160_RATIO + 1)	0x0
ACLK_100_RATIO	[7:4]	RW	DIVACLK_100 clock divider ratio ACLK_100 = MOUTACLK_100 / (ACLK_100_RATIO + 1)	0x0
RSVD	[3]	–	Reserved	0x0
ACLK_200_RATIO	[2:0]	RW	DIVACLK_200 clock divider ratio ACLK_200 = MOUTACLK_200 / (ACLK_200_RATIO + 1)	0x0

5.3.1.48 CLK_DIV_CAM

- Address = 0x1003_C520, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
CSIS1_RATIO	[31:28]	RW	DIVCSIS1 clock divider ratio SCLK_CSIS1 = MOUTCSIS1 / (CSIS1_RATIO + 1)	0x0
CSIS0_RATIO	[27:24]	RW	DIVCSIS0 clock divider ratio SCLK_CSIS0 = MOUTCSIS0 / (CSIS0_RATIO + 1)	0x0
CAM1_RATIO	[23:20]	RW	DIVCAM1 clock divider ratio SCLK_CAM1 = MOUTCAM1 / (CAM1_RATIO + 1)	0x0
CAM0_RATIO	[19:16]	RW	DIVCAM0 clock divider ratio SCLK_CAM0 = MOUTCAM0 / (CAM0_RATIO + 1)	0x0
FIMC3_LCLK_RATIO	[15:12]	RW	DIVFIMC3_LCLK clock divider ratio SCLKFIMC3_LCLK= MOUTFIMC3_LCLK / (FIMC3_LCLK_RATIO + 1)	0x0
FIMC2_LCLK_RATIO	[11:8]	RW	DIVFIMC2_LCLK clock divider ratio SCLKFIMC2_LCLK= MOUTFIMC2_LCLK / (FIMC2_LCLK_RATIO + 1)	0x0
FIMC1_LCLK_RATIO	[7:4]	RW	DIVFIMC1_LCLK clock divider ratio SCLKFIMC1_LCLK= MOUTFIMC1_LCLK / (FIMC1_LCLK_RATIO + 1)	0x0
FIMC0_LCLK_RATIO	[3:0]	RW	DIVFIMC0_LCLK clock divider ratio SCLKFIMC0_LCLK= MOUTFIMC0_LCLK / (FIMC0_LCLK_RATIO + 1)	0x0

5.3.1.49 CLK_DIV_TV

- Address = 0x1003_C524, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	-	Reserved	0x0
TV_BLK_RATIO	[3:0]	RW	DIVTV_BLK clock divider ratio SCLK_PIXEL= SCLKVPLL/ (TV_BLK_RATIO + 1)	0x0

5.3.1.50 CLK_DIV_MFC

- Address = 0x1003_C528, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	-	Reserved	0x0
MFC_RATIO	[3:0]	RW	DIVMFC clock divider ratio SCLK_MFC= MOUTMFC/(MFC_RATIO +1)	0x0

5.3.1.51 CLK_DIV_G3D

- Address = 0x1003_C52C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	—	Reserved	0x0
G3D_RATIO	[3:0]	RW	DIVG3D clock divider ratio SCLK_G3D= MOUTG3D / (G3D_RATIO + 1)	0x0

5.3.1.52 CLK_DIV_IMAGE

- Address = 0x1003_C530, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	—	Reserved	0x0
G2D_RATIO	[3:0]	RW	DIVG2D clock divider ratio SCLK_G2D= MOUTG2D / (G2D_RATIO + 1)	0x0

5.3.1.53 CLK_DIV_LCD0

- Address = 0x1003_C534, Reset Value = 0x0070_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	—	Reserved	0x0
MIPI0_PRE_RATIO	[23:20]	RW	DIVMIPI0_PRE clock divider ratio SCLK_MIPI0 = DOUTMIPI0 / (MIPI0_PRE_RATIO + 1)	0x7
MIPI0_RATIO	[19:16]	RW	DIVMIPI0 clock divider ratio SCLK_MIPIDPHY4L = MOUTMIPI0 / (MIPI0_RATIO + 1)	0x0
RSVD	[15:12]	RW	Reserved	0x0
RSVD	[11:8]	RW	Reserved	0x0
RSVD	[7:4]	RW	Reserved	0x0
FIMD0_RATIO	[3:0]	RW	DIVFIMD0 clock divider ratio SCLK_FIMD0 = MOUTFIMD0 / (FIMD0_RATIO + 1)	0x0

5.3.1.54 CLK_DIV_LCD1

- Address = 0x1003_C538, Reset Value = 0x0070_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	–	Reserved	0x0
MIPI1_PRE_RATIO	[23:20]	RW	DIVMIP1_PRE clock divider ratio SCLK_MIPI1 = DOUTMIP1 / (MIPI1_PRE_RATIO + 1)	0x7
MIPI1_RATIO	[19:16]	RW	DIVMIP1 clock divider ratio SCLK_MIPIDPHY2L = MOUTMIP1 / (MIPI1_RATIO + 1)	0x0
RSVD	[15:12]	RW	Reserved	0x0
RSVD	[11:8]	RW	Reserved	0x0
RSVD	[7:4]	RW	Reserved	0x0
FIMD1_RATIO	[3:0]	RW	DIVFIMD1 clock divider ratio SCLK_FIMD1 = MOUTFIMD1 / (FIMD1_RATIO + 1)	0x0

5.3.1.55 CLK_DIV_MAUDIO

- Address = 0x1003_C53C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:12]	–	Reserved	0x0
PCM0_RATIO	[11:4]	RW	DIVPCM0 clock divider ratio SCLK_PCM0 = SCLK_AUDIO0 / (PCM0_RATIO + 1)	0x0
AUDIO0_RATIO	[3:0]	RW	DIVAUDIO0 clock divider ratio SCLK_AUDIO0 = MOUTAUDIO0 / (AUDIO0_RATIO + 1)	0x0

5.3.1.56 CLK_DIV_FSYS0

- Address = 0x1003_C540, Reset Value = 0x00B0_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	–	Reserved	0x0
SATA_RATIO	[23:20]	RW	DIVSATA clock divider ratio SCLK_SATA = MOUTSATA / (SATA_RATIO + 1)	0xB
RSVD	[19:0]	–	Reserved	0x0

5.3.1.57 CLK_DIV_FSYS1

- Address = 0x1003_C544, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
MMC1_PRE_RATIO	[31:24]	RW	DIVMMC1_PRE clock divider ratio SCLK_MMC1 = DOUTMMC1 / (MMC1_PRE_RATIO + 1)	0x0
RSVD	[23:20]	–	Reserved	0x0
MMC1_RATIO	[19:16]	RW	DIVMMC1 clock divider ratio DOUTMMC1 = MOUTMMC1 / (MMC1_RATIO + 1)	0x0
MMC0_PRE_RATIO	[15:8]	RW	DIVMMC0_PRE clock divider ratio SCLK_MMC0 = DOUTMMC0 / (MMC0_PRE_RATIO + 1)	0x0
RSVD	[7:4]	–	Reserved	0x0
MMC0_RATIO	[3:0]	RW	DIVMMC0 clock divider ratio DOUTMMC0 = MOUTMMC0 / (MMC0_RATIO + 1)	0x0

5.3.1.58 CLK_DIV_FSYS2

- Address = 0x1003_C548, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
MMC3_PRE_RATIO	[31:24]	RW	DIVMMC3_PRE clock divider ratio SCLK_MMC3 = DOUTMMC3 / (MMC3_PRE_RATIO + 1)	0x0
RSVD	[23:20]	-	Reserved	0x0
MMC3_RATIO	[19:16]	RW	DIVMMC3 clock divider ratio DOUTMMC3 = MOUTMMC3 / (MMC3_RATIO + 1)	0x0
MMC2_PRE_RATIO	[15:8]	RW	DIVMMC2_PRE clock divider ratio SCLK_MMC2 = DOUTMMC2 / (MMC2_PRE_RATIO + 1)	0x0
RSVD	[7:4]	-	Reserved	0x0
MMC2_RATIO	[3:0]	RW	DIVMMC2 clock divider ratio DOUTMMC2 = MOUTMMC2 / (MMC2_RATIO + 1)	0x0

5.3.1.59 CLK_DIV_FSYS3

- Address = 0x1003_C54C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	-	Reserved	0x0
MMC4_PRE_RATIO	[15:8]	RW	DIVMMC4_PRE clock divider ratio SCLK_MMC4 = DOUTMMC4 / (MMC4_PRE_RATIO + 1)	0x0
RSVD	[7:4]	-	Reserved	0x0
MMC4_RATIO	[3:0]	RW	DIVMMC4 clock divider ratio DOUTMMC4 = MOUTMMC4 / (MMC4_RATIO + 1)	0x0

5.3.1.60 CLK_DIV_PERILO

- Address = 0x1003_C550, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:20]	–	Reserved	0x0
UART4_RATIO	[19:16]	RW	DIVUART4 clock divider ratio SCLK_UART4 = MOUTUART4 / (UART4_RATIO + 1)	0x0
UART3_RATIO	[15:12]	RW	DIVUART3 clock divider ratio SCLK_UART3 = MOUTUART3 / (UART3_RATIO + 1)	0x0
UART2_RATIO	[11:8]	RW	DIVUART2 clock divider ratio SCLK_UART2 = MOUTUART2 / (UART2_RATIO + 1)	0x0
UART1_RATIO	[7:4]	RW	DIVUART1 clock divider ratio SCLK_UART1 = MOUTUART1 / (UART1_RATIO + 1)	0x0
UART0_RATIO	[3:0]	RW	DIVUART0 clock divider ratio SCLK_UART0 = MOUTUART0 / (UART0_RATIO + 1)	0x0

5.3.1.61 CLK_DIV_PERIL1

- Address = 0x1003_C554, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
SPI1_PRE_RATIO	[31:24]	RW	DIVSPI1_PRE clock divider ratio SCLK_SPI1 = DOUTSPI1 / (SPI1_PRE_RATIO + 1)	0x0
RSVD	[23:20]	–	Reserved	0x0
SPI1_RATIO	[19:16]	RW	DIVSPI1 clock divider ratio DOUTSPI1 = MOUTSPI1 / (SPI1_RATIO + 1)	0x0
SPI0_PRE_RATIO	[15:8]	RW	DIVSPI0_PRE clock divider ratio SCLK_SPI0 = DOUTSPI0 / (SPI0_PRE_RATIO + 1)	0x0
RSVD	[7:4]	–	Reserved	0x0
SPI0_RATIO	[3:0]	RW	DIVSPI0 clock divider ratio DOUTSPI0 = MOUTSPI0 / (SPI0_RATIO + 1)	0x0

5.3.1.62 CLK_DIV_PERIL2

- Address = 0x1003_C558, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	–	Reserved	0x0
SPI2_PRE_RATIO	[15:8]	RW	DIVSPI2_PRE clock divider ratio SCLK_SPI2 = DOUTSPI2 / (SPI2_PRE_RATIO + 1)	0x0
RSVD	[7:4]	–	Reserved	0x0
SPI2_RATIO	[3:0]	RW	DIVSPI2 clock divider ratio DOUTSPI2 = MOUTSPI2 / (SPI2_RATIO + 1)	0x0

5.3.1.63 CLK_DIV_PERIL3

- Address = 0x1003_C55C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	–	Reserved	0x0
SLIMBUS_RATIO	[7:4]	RW	DIVSLIMBUS clock divider ratio SCLK_SLIMBUS = SCLKEPLL / (SLIMBUS_RATIO + 1)	0x0
RSVD	[3:0]	–	Reserved	0x0

5.3.1.64 CLK_DIV_PERIL4

- Address = 0x1003_C560, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:28]	–	Reserved	0x0
PCM2_RATIO	[27:20]	RW	DIVPCM2 clock divider ratio SCLK_PCM2 = SCLK_AUDIO2 / (PCM2_RATIO + 1)	0x0
AUDIO2_RATIO	[19:16]	RW	DIVAUDIO2 clock divider ratio SCLK_AUDIO2 = MOUTAUDIO2 / (AUDIO2_RATIO + 1)	0x0
RSVD	[15:12]	–	Reserved	0x0
PCM1_RATIO	[11:4]	RW	DIVPCM1 clock divider ratio SCLK_PCM1 = SCLK_AUDIO1 / (PCM1_RATIO + 1)	0x0
AUDIO1_RATIO	[3:0]	RW	DIVAUDIO1 clock divider ratio SCLK_AUDIO1 = MOUTAUDIO1 / (AUDIO1_RATIO + 1)	0x0

5.3.1.65 CLK_DIV_PERIL5

- Address = 0x1003_C564, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:14]	–	Reserved	0x0
I2S2_RATIO	[13:8]	RW	DIVI2S2 clock divider ratio SCLK_I2S2 = SCLK_AUDIO2 / (I2S2_RATIO + 1)	0x0
RSVD	[7:6]	–	Reserved	0x0
I2S1_RATIO	[5:0]	RW	DIVI2S1 clock divider ratio SCLK_I2S1 = SCLK_AUDIO1 / (I2S1_RATIO + 1)	0x0

5.3.1.66 CLKDIV2_RATIO

- Address = 0x1003_C580, Reset Value = 0x0111_1111

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	—	Reserved	0x0
GPS_BLK	[25:24]	RW	PCLK divider ratio in GPS_BLK 0 = Reserved 1 = Divide by 2 2 = Divide by 3 3 = Divide by 4	0x1
RSVD	[23:22]	—	Reserved	0x0
TV_BLK	[21:20]	RW	PCLK divider ratio in TV_BLK 0 = Reserved 1 = Divide by 2 2 = Divide by 3 3 = Divide by 4	0x1
RSVD	[19:18]	—	Reserved	0x0
LCD1_BLK	[17:16]	RW	PCLK divider ratio in LCD1_BLK 0 = Reserved 1 = Divide by 2 2 = Divide by 3 3 = Divide by 4	0x1
RSVD	[15:14]	—	Reserved	0x0
LCD0_BLK	[13:12]	RW	PCLK divider ratio in LCD0_BLK for 160MHz domain 0 = Reserved 1 = Divide by 2 2 = Divide by 3 3 = Divide by 4	0x1
RSVD	[11:10]	—	Reserved	0x0
IMG_BLK	[9:8]	RW	PCLK divider ratio in LCD0_BLK for 200MHz domain 0 = Reserved 1 = Divide by 2 2 = Divide by 3 3 = Divide by 4	0x1
RSVD	[7:6]	—	Reserved	0x0
CAM_BLK	[5:4]	RW	PCLK divider ratio in CAM_BLK 0 = Reserved 1 = Divide by 2 2 = Divide by 3 3 = Divide by 4	0x1
RSVD	[3:2]	—	Reserved	0x0
FSYS_BLK	[1:0]	RW	PCLK divider ratio in FSYS_BLK 0 = Reserved 1 = Divide by 2 2 = Divide by 3	0x1

Name	Bit	Type	Description	Reset Value
			3 = Divide by 4	

5.3.1.67 CLK_DIV_STAT_TOP

- Address = 0x1003_C610, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:17]	—	Reserved	0x0
DIV_ONENAND	[16]	R	DIVONENAND status 0 = Stable 1 = Divider is changing	0x0
RSVD	[15:13]	—	Reserved	0x0
DIV_ACLK_133	[12]	R	DIVACLK_133 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[11:9]	—	Reserved	0x0
DIV_ACLK_160	[8]	R	DIVACLK_160 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[7:5]	—	Reserved	0x0
DIV_ACLK_100	[4]	R	DIVACLK_100 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[3:1]	—	Reserved	0x0
DIV_ACLK_200	[0]	R	DIVACLK_200 status 0 = Stable 1 = Divider is changing	0x0

5.3.1.68 CLK_DIV_STAT_CAM

- Address = 0x1003_C620, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:29]	–	Reserved	0x0
DIV_CSIS1	[28]	R	DIVCSIS1 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[27:25]	–	Reserved	0x0
DIV_CSIS0	[24]	R	DIVCSIS0 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[23:21]	–	Reserved	0x0
DIV_CAM1	[20]	R	DIVCAM1 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[19:17]	–	Reserved	0x0
DIV_CAM0	[16]	R	DIVCAM0 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[15:13]	–	Reserved	0x0
DIV_FIMC3_LCLK	[12]	R	DIVFIMC3_LCLK status 0 = Stable 1 = Divider is changing	0x0
RSVD	[11:9]	–	Reserved	0x0
DIV_FIMC2_LCLK	[8]	R	DIVFIMC2_LCLK status 0 = Stable 1 = Divider is changing	0x0
RSVD	[7:5]	–	Reserved	0x0
DIV_FIMC1_LCLK	[4]	R	DIVFIMC1_LCLK status 0 = Stable 1 = Divider is changing	0x0
RSVD	[3:1]	–	Reserved	0x0
DIV_FIMC0_LCLK	[0]	R	DIVFIMC0_LCLK status 0 = Stable 1 = Divider is changing	0x0

5.3.1.69 CLK_DIV_STAT_TV

- Address = 0x1003_C624, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	–	Reserved	0x0
DIV_TV_BLK	[0]	R	DIVTV_BLK status 0 = Stable 1 = Divider is changing	0x0

5.3.1.70 CLK_DIV_STAT_MFC

- Address = 0x1003_C628, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	–	Reserved	0x0
DIV_MFC	[0]	R	DIVMFC status 0 = Stable 1 = Divider is changing	0x0

5.3.1.71 CLK_DIV_STAT_G3D

- Address = 0x1003_C62C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	–	Reserved	0x0
DIV_G3D	[0]	R	DIVG3D status 0 = Stable 1 = Divider is changing	0x0

5.3.1.72 CLK_DIV_STAT_IMAGE

- Address = 0x1003_C630, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	–	Reserved	0x0
DIV_G2D	[0]	R	DIVG2D status 0 = Stable 1 = Divider is changing	0x0

5.3.1.73 CLK_DIV_STAT_LCD0

- Address = 0x1003_C634, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:21]	–	Reserved	0x0
DIV_MIPI0_PRE	[20]	R	DIVMIPI0_PRE status 0 = Stable 1 = Divider is changing	0x0
RSVD	[19:17]	–	Reserved	0x0
DIV_MIPI0	[16]	R	DIVMIPI0 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[15:13]	–	Reserved	0x0
RSVD	[12]	R	Reserved	0x0
RSVD	[11:9]	–	Reserved	0x0
RSVD	[8]	R	Reserved	0x0
RSVD	[7:5]	–	Reserved	0x0
RSVD	[4]	R	Reserved	0x0
RSVD	[3:1]	–	Reserved	0x0
DIV_FIMD0	[0]	R	DIVFIMD0 status 0 = Stable 1 = Divider is changing	0x0

5.3.1.74 CLK_DIV_STAT_LCD1

- Address = 0x1003_C638, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:21]	–	Reserved	0x0
DIV_MIPI1_PRE	[20]	R	DIVMIPI1_PRE status 0 = Stable 1 = Divider is changing	0x0
RSVD	[19:17]	–	Reserved	0x0
DIV_MIPI1	[16]	R	DIVMIPI1 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[15:13]	–	Reserved	0x0
RSVD	[12]	R	Reserved	0x0
RSVD	[11:9]	–	Reserved	0x0
RSVD	[8]	R	Reserved	0x0
RSVD	[7:5]	–	Reserved	0x0
RSVD	[4]	R	Reserved	0x0
RSVD	[3:1]	–	Reserved	0x0
DIV_FIMD1	[0]	R	DIVFIMD1 status 0 = Stable 1 = Divider is changing	0x0

5.3.1.75 CLK_DIV_STAT_MAUDIO

- Address = 0x1003_C63C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	–	Reserved	0x0
DIV_PCM0	[4]	R	DIVPCM0 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[3:1]	–	Reserved	0x0
DIV_AUDIO0	[0]	R	DIVAUDIO0 status 0 = Stable 1 = Divider is changing	0x0

5.3.1.76 CLK_DIV_STAT_FSYS0

- Address = 0x1003_C640, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:21]	–	Reserved	0x0
DIV_SATA	[20]	R	DIVSATA status 0 = Stable 1 = Divider is changing	0x0
RSVD	[19:0]	–	Reserved	0x0

5.3.1.77 CLK_DIV_STAT_FSYS1

- Address = 0x1003_C644, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:25]	–	Reserved	0x0
DIV_MMC1_PRE	[24]	R	DIVMMC1_PRE status 0 = Stable 1 = Divider is changing	0x0
RSVD	[23:17]	–	Reserved	0x0
DIV_MMC1	[16]	R	DIVMMC1 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[15:9]	–	Reserved	0x0
DIV_MMC0_PRE	[8]	R	DIVMMC0_PRE status 0 = Stable 1 = Divider is changing	0x0
RSVD	[7:1]	–	Reserved	0x0
DIV_MMC0	[0]	R	DIVMMC0 status 0 = Stable 1 = Divider is changing	0x0

5.3.1.78 CLK_DIV_STAT_FSYS2

- Address = 0x1003_C648, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:25]	–	Reserved	0x0
DIV_MMC3_PRE	[24]	R	DIVMMC3_PRE status 0 = Stable 1 = Divider is changing	0x0
RSVD	[23:17]	–	Reserved	0x0
DIV_MMC3	[16]	R	DIVMMC3 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[15:9]	–	Reserved	0x0
DIV_MMC2_PRE	[8]	R	DIVMMC2_PRE status 0 = Stable 1 = Divider is changing	0x0
RSVD	[7:1]	–	Reserved	0x0
DIV_MMC2	[0]	R	DIVMMC2 status 0 = Stable 1 = Divider is changing	0x0

5.3.1.79 CLK_DIV_STAT_FSYS3

- Address = 0x1003_C64C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:9]	–	Reserved	0x0
DIV_MMC4_PRE	[8]	R	DIVMMC4_PRE status 0 = Stable 1 = Divider is changing	0x0
RSVD	[7:1]	–	Reserved	0x0
DIV_MMC4	[0]	R	DIVMMC4 status 0 = Stable 1 = Divider is changing	0x0

5.3.1.80 CLK_DIV_STAT_PERI0

- Address = 0x1003_C650, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:17]	–	Reserved	0x0
DIV_UART4	[16]	R	DIVUART4 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[15:13]	–	Reserved	0x0
DIV_UART3	[12]	R	DIVUART3 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[11:9]	–	Reserved	0x0
DIV_UART2	[8]	R	DIVUART2 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[7:5]	–	Reserved	0x0
DIV_UART1	[4]	R	DIVUART1 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[3:1]	–	Reserved	0x0
DIV_UART0	[0]	R	DIVUART0 status 0 = Stable 1 = Divider is changing	0x0

5.3.1.81 CLK_DIV_STAT_PERIL1

- Address = 0x1003_C654, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:25]	–	Reserved	0x0
DIV_SPI1_PRE	[24]	R	DIVSPI1_PRE status 0 = Stable 1 = Divider is changing	0x0
RSVD	[23:17]	–	Reserved	0x0
DIV_SPI1	[16]	R	DIVSPI1 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[15:9]	–	Reserved	0x0
DIV_SPI0_PRE	[8]	R	DIVSPI0_PRE status 0 = Stable 1 = Divider is changing	0x0
RSVD	[7:1]	–	Reserved	0x0
DIV_SPI0	[0]	R	DIVSPI0 status 0 = Stable 1 = Divider is changing	0x0

5.3.1.82 CLK_DIV_STAT_PERIL2

- Address = 0x1003_C658, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:9]	–	Reserved	0x0
DIV_SPI2_PRE	[8]	R	DIVSPI2_PRE status 0 = Stable 1 = Divider is changing	0x0
RSVD	[7:1]	–	Reserved	0x0
DIV_SPI2	[0]	R	DIVSPI2 status 0 = Stable 1 = Divider is changing	0x0

5.3.1.83 CLK_DIV_STAT_PERIL3

- Address = 0x1003_C65C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	–	Reserved	0x0
DIV_SLIMBUS	[4]	R	DIVSLIMBUS status 0 = Stable 1 = Divider is changing	0x0
RSVD	[3:0]	–	Reserved	0x0

5.3.1.84 CLK_DIV_STAT_PERIL4

- Address = 0x1003_C660, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:21]	–	Reserved	0x0
DIV_PCM2	[20]	R	DIVPCM2 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[19:17]	–	Reserved	0x0
DIV_AUDIO2	[16]	R	DIVAUDIO2 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[15:5]	–	Reserved	0x0
DIV_PCM1	[4]	R	DIVPCM1 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[3:1]	–	Reserved	0x0
DIV_AUDIO1	[0]	R	DIVAUDIO1 status 0 = Stable 1 = Divider is changing	0x0

5.3.1.85 CLK_DIV_STAT_PERIL5

- Address = 0x1003_C664, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:9]	–	Reserved	0x0
DIV_I2S2	[8]	R	DIVI2S2 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[7:1]	–	Reserved	0x0
DIV_I2S1	[0]	R	DIVI2S1 status 0 = Stable 1 = Divider is changing	0x0

5.3.1.86 CLKDIV2_STAT

- Address = 0x1003_C680, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:25]	–	Reserved	0x0
GPS_BLK	[24]	R	PCLK divider status in TV_BLK 0 = Stable 1 = Divider is changing	0x0
RSVD	[23:21]	–	Reserved	0x0
TV_BLK	[20]	R	PCLK divider status in TV_BLK 0 = Stable 1 = Divider is changing	0x0
RSVD	[19:17]	–	Reserved	0x0
LCD1_BLK	[16]	R	PCLK divider status in LCD1_BLK 0 = Stable 1 = Divider is changing	0x0
RSVD	[15:13]	–	Reserved	0x0
LCD0_BLK	[12]	R	PCLK divider status in LCD0_BLK for 160MHz domain 0 = Stable 1 = Divider is changing	0x0
RSVD	[11:9]	–	Reserved	0x0
IMG_BLK	[8]	R	PCLK divider status in LCD0_BLK for 200MHz domain 0 = Stable 1 = Divider is changing	0x0
RSVD	[7:5]	–	Reserved	0x0
CAM_BLK	[4]	R	PCLK divider status in CAM_BLK 0 = Stable 1 = Divider is changing	0x0
RSVD	[3:1]	–	Reserved	0x0
FSYS_BLK	[0]	R	PCLK divider status in FSYS_BLK 0 = Stable 1 = Divider is changing	0x0

5.3.1.87 CLK_GATE_SCLK_CAM

- Address = 0x1003_C820, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	–	Reserved	0xFFFFFFFF
RSVD	[7:6]	RW	Reserved (should be 2'b11)	0x3
SCLK_CAM1	[5]	RW	Gating special clock for CAM1 0 = Mask 1 = Pass	0x1
SCLK_CAM0	[4]	RW	Gating special clock for CAM0 0 = Mask 1 = Pass	0x1
RSVD	[3:0]	RW	Reserved (should be 2'b1111)	0xF

5.3.1.88 CLK_GATE_IP_CAM

- Address = 0x1003_C920, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:19]	–	Reserved	0x1FFF
CLK_PIXELASYNC_M1	[18]	RW	Gating all clocks for PIXELASYNCM1 0 = Mask 1 = Pass	0x1
CLK_PIXELASYNC_M0	[17]	RW	Gating all clocks for PIXELASYNCM0 0 = Mask 1 = Pass	0x1
CLK_PPMUCAMIF	[16]	RW	Gating all clocks for PPMUCAMIF 0 = Mask 1 = Pass	0x1
CLK_QEFIMC3	[15]	RW	Gating all clocks for QEFIMC3 0 = Mask 1 = Pass	0x1
CLK_QEFIMC2	[14]	RW	Gating all clocks for QEFIMC2 0 = Mask 1 = Pass	0x1
CLK_QEFIMC1	[13]	RW	Gating all clocks for QEFIMC1 0 = Mask 1 = Pass	0x1
CLK_QEFIMC0	[12]	RW	Gating all clocks for QEFIMC0 0 = Mask 1 = Pass	0x1
CLK_SMMUJPEG	[11]	RW	Gating all clocks for SMMUJPEG 0 = Mask 1 = Pass	0x1
CLK_SMMUFIMC3	[10]	RW	Gating all clocks for SMMUFIMC3 0 = Mask	0x1

Name	Bit	Type	Description	Reset Value
			1 = Pass Gating all clocks for SMMUFIMC2 0 = Mask 1 = Pass	
CLK_SMMUFIMC2	[9]	RW	Gating all clocks for SMMUFIMC2 0 = Mask 1 = Pass	0x1
CLK_SMMUFIMC1	[8]	RW	Gating all clocks for SMMUFIMC1 0 = Mask 1 = Pass	0x1
CLK_SMMUFIMC0	[7]	RW	Gating all clocks for SMMUFIMC0 0 = Mask 1 = Pass	0x1
CLK_JPEG	[6]	RW	Gating all clocks for JPEG 0 = Mask 1 = Pass	0x1
CLK_CSIS1	[5]	RW	Gating all clocks for CSIS1 0 = Mask 1 = Pass	0x1
CLK_CSIS0	[4]	RW	Gating all clocks for CSIS0 0 = Mask 1 = Pass	0x1
CLK_FIMC3	[3]	RW	Gating all clocks for FIMC3 0 = Mask 1 = Pass	0x1
CLK_FIMC2	[2]	RW	Gating all clocks for FIMC2 0 = Mask 1 = Pass	0x1
CLK_FIMC1	[1]	RW	Gating all clocks for FIMC1 0 = Mask 1 = Pass	0x1
CLK_FIMC0	[0]	RW	Gating all clocks for FIMC0 0 = Mask 1 = Pass	0x1

5.3.1.89 CLK_GATE_IP_TV

- Address = 0x1003_C924, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:6]	–	Reserved	0x3FFFFFFF
CLK_PPMUTV	[5]	RW	Gating all clocks for PPMUTV 0 = Mask 1 = Pass	0x1
CLK_SMMUTV	[4]	RW	Gating all clocks for SMMUTV 0 = Mask 1 = Pass	0x1
CLK_HDMI	[3]	RW	Gating all clocks for HDMI link 0 = Mask 1 = Pass	0x1
CLK_TVENC	[2]	RW	Gating all clocks for TVENC 0 = Mask 1 = Pass	0x1
CLK_MIXER	[1]	RW	Gating all clocks for MIXER 0 = Mask 1 = Pass	0x1
CLK_VP	[0]	RW	Gating all clocks for VP 0 = Mask 1 = Pass	0x1

5.3.1.90 CLK_GATE_IP_MFC

- Address = 0x1003_C928, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	–	Reserved	0x7FFFFFFF
CLK_PPMUMFC_R	[4]	RW	Gating all clocks for PPMUMFC_R 0 = Mask 1 = Pass	0x1
CLK_PPMUMFC_L	[3]	RW	Gating all clocks for PPMUMFC_L 0 = Mask 1 = Pass	0x1
CLK_SMMUMFC_R	[2]	RW	Gating all clocks for SMMUMFC_R 0 = Mask 1 = Pass	0x1
CLK_SMMUMFC_L	[1]	RW	Gating all clocks for SMMUMFC_L 0 = Mask 1 = Pass	0x1
CLK_MFC	[0]	RW	Gating all clocks for MFC 0 = Mask 1 = Pass	0x1

5.3.1.91 CLK_GATE_IP_G3D

- Address = 0x1003_C92C, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:3]	–	Reserved	0x1FFFFFFF
CLK_QEG3D	[2]	RW	Gating all clocks for QEG3D 0 = Mask 1 = Pass	0x1
CLK_PPMUG3D	[1]	RW	Gating all clocks for PPMUG3D 0 = Mask 1 = Pass	0x1
CLK_G3D	[0]	RW	Gating all clocks for G3D 0 = Mask 1 = Pass	0x1

5.3.1.92 CLK_GATE_IP_IMAGE

- Address = 0x1003_C930, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:10]	–	Reserved	0x3FFFFF
CLK_PPMUIMAGE	[9]	RW	Gating all clocks for PPMUIMAGE 0 = Mask 1 = Pass	0x1
CLK_QEMDMA	[8]	RW	Gating all clocks for QEMDMA 0 = Mask 1 = Pass	0x1
CLK_QEROTATOR	[7]	RW	Gating all clocks for QEROTATOR 0 = Mask 1 = Pass	0x1
CLK_QEG2D	[6]	RW	Gating all clocks for QEG2D 0 = Mask 1 = Pass	0x1
CLK_SMMUIMDMA	[5]	RW	Gating all clocks for SMMUIMDMA 0 = Mask 1 = Pass	0x1
CLK_SMMUROTATOR	[4]	RW	Gating all clocks for SMMUROTATOR 0 = Mask 1 = Pass	0x1
CLK_SMMUG2D	[3]	RW	Gating all clocks for SMMUG2D 0 = Mask 1 = Pass	0x1
CLK_MDMA	[2]	RW	Gating all clocks for MDMA 0 = Mask 1 = Pass	0x1
CLK_ROTATOR	[1]	RW	Gating all clocks for ROTATOR 0 = Mask 1 = Pass	0x1
CLK_G2D	[0]	RW	Gating all clocks for G2D 0 = Mask 1 = Pass	0x1

5.3.1.93 CLK_GATE_IP_LCD0

- Address = 0x1003_C934, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:6]	–	Reserved	0x3FFFFFFF
CLK_PPMULCD0	[5]	RW	Gating all clocks for PPMULCD0 0 = Mask 1 = Pass	0x1
CLK_SMMUFIMD0	[4]	RW	Gating all clocks for SMMUFIMD0 0 = Mask 1 = Pass	0x1
CLK_DSIM0	[3]	RW	Gating all clocks for DSIM0 0 = Mask 1 = Pass	0x1
RSVD	[2]	RW	Reserved	0x1
CLK_MIE0	[1]	RW	Gating all clocks for MIE0 0 = Mask 1 = Pass	0x1
CLK_FIMD0	[0]	RW	Gating all clocks for FIMD0 0 = Mask 1 = Pass	0x1

5.3.1.94 CLK_GATE_IP_LCD1

- Address = 0x1003_C938, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:6]	–	Reserved	0x3FFFFFFF
CLK_PPMULCD1	[5]	RW	Gating all clocks for PPMULCD1 0 = Mask 1 = Pass	0x1
CLK_SMMUFIMD1	[4]	RW	Gating all clocks for SMMUFIMD1 0 = Mask 1 = Pass	0x1
CLK_DSIM1	[3]	RW	Gating all clocks for DSIM1 0 = Mask 1 = Pass	0x1
RSVD	[2]	RW	Reserved	0x1
CLK_MIE1	[1]	RW	Gating all clocks for MIE1 0 = Mask 1 = Pass	0x1
CLK_FIMD1	[0]	RW	Gating all clocks for FIMD1 0 = Mask 1 = Pass	0x1

5.3.1.95 CLK_GATE_IP_FSYS

- Address = 0x1003_C940, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:19]	–	Reserved	0x1FFF
CLK_SMMUPCIE	[18]	RW	Gating all clocks for SMMUPCIE 0 = Mask 1 = Pass	0x1
CLK_PPMUFILE	[17]	RW	Gating all clocks for PPMUFILE 0 = Mask 1 = Pass	0x1
CLK_NFCON	[16]	RW	Gating all clocks for NFCON 0 = Mask 1 = Pass	0x1
CLK_ONENAND	[15]	RW	Gating all clocks for ONENAND 0 = Mask 1 = Pass	0x1
CLK_PCIE	[14]	RW	Gating all clocks for PCI express 0 = Mask 1 = Pass	0x1
CLK_USBDEVICE	[13]	RW	Gating all clocks for USB Device 0 = Mask 1 = Pass	0x1
CLK_USBHOST	[12]	RW	Gating all clocks for USB HOST 0 = Mask 1 = Pass	0x1
CLK_SROMC	[11]	RW	Gating all clocks for SROM 0 = Mask 1 = Pass	0x1
CLK_SATA	[10]	RW	Gating all clocks for SATA 0 = Mask 1 = Pass	0x1
CLK_SDMMC4	[9]	RW	Gating all clocks for SDMMC4 0 = Mask 1 = Pass	0x1
CLK_SDMMC3	[8]	RW	Gating all clocks for SDMMC3 0 = Mask 1 = Pass	0x1
CLK_SDMMC2	[7]	RW	Gating all clocks for SDMMC2 0 = Mask 1 = Pass	0x1
CLK_SDMMC1	[6]	RW	Gating all clocks for SDMMC1 0 = Mask 1 = Pass	0x1
CLK_SDMMC0	[5]	RW	Gating all clocks for SDMMC0 0 = Mask 1 = Pass	0x1

Name	Bit	Type	Description	Reset Value
CLK_TSI	[4]	RW	Gating all clocks for TSI 0 = Mask 1 = Pass	0x1
CLK_SATAPHY	[3]	RW	Gating all clocks for SATAPHY 0 = Mask 1 = Pass	0x1
CLK_PCIEPHY	[2]	RW	Gating all clocks for PCIEPHY 0 = Mask 1 = Pass	0x1
CLK_PDMA1	[1]	RW	Gating all clocks for PDMA1 0 = Mask 1 = Pass	0x1
CLK_PDMA0	[0]	RW	Gating all clocks for PDMA0 0 = Mask 1 = Pass	0x1

5.3.1.96 CLK_GATE_IP_GPS

- Address = 0x1003_C94C, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:2]	-	Reserved	0x3FFFFFFF
CLK_SMMUGPS	[1]	RW	Gating all clocks for SMMUGPS 0 = Mask 1 = Pass	0x1
CLK_GPS	[0]	RW	Gating all clocks for GPS 0 = Mask 1 = Pass	0x1

5.3.1.97 CLK_GATE_IP_PERIL

- Address = 0x1003_C950, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:29]	–	Reserved	0x7
CLK_MODEMIF	[28]	RW	Gating all clocks for MODEMIF 0 = Mask 1 = Pass	0x1
CLK_AC97	[27]	RW	Gating all clocks for AC97 0 = Mask 1 = Pass	0x1
CLK_SPDIF	[26]	RW	Gating all clocks for SPDIF 0 = Mask 1 = Pass	0x1
CLK_SLIMBUS	[25]	RW	Gating all clocks for Slimbus 0 = Mask 1 = Pass	0x1
CLK_PWM	[24]	RW	Gating all clocks for PWM 0 = Mask 1 = Pass	0x1
CLK_PCM2	[23]	RW	Gating all clocks for PCM2 0 = Mask 1 = Pass	0x1
CLK_PCM1	[22]	RW	Gating all clocks for PCM1 0 = Mask 1 = Pass	0x1
CLK_I2S2	[21]	RW	Gating all clocks for I2S2 0 = Mask 1 = Pass	0x1
CLK_I2S1	[20]	RW	Gating all clocks for I2S1 0 = Mask 1 = Pass	0x1
RSVD	[19]	–	Reserved	0x1
CLK_SPI2	[18]	RW	Gating all clocks for SPI2 0 = Mask 1 = Pass	0x1
CLK_SPI1	[17]	RW	Gating all clocks for SPI1 0 = Mask 1 = Pass	0x1
CLK_SPI0	[16]	RW	Gating all clocks for SPI0 0 = Mask 1 = Pass	0x1
CLK_TSADC	[15]	RW	Gating all clocks for TSADC 0 = Mask 1 = Pass	0x1
CLK_I2CHDMI	[14]	RW	Gating all clocks for I2CHDMI 0 = Mask	0x1

Name	Bit	Type	Description	Reset Value
			1 = Pass Gating all clocks for I2C7 0 = Mask 1 = Pass	
CLK_I2C7	[13]	RW	Gating all clocks for I2C7 0 = Mask 1 = Pass	0x1
CLK_I2C6	[12]	RW	Gating all clocks for I2C6 0 = Mask 1 = Pass	0x1
CLK_I2C5	[11]	RW	Gating all clocks for I2C5 0 = Mask 1 = Pass	0x1
CLK_I2C4	[10]	RW	Gating all clocks for I2C4 0 = Mask 1 = Pass	0x1
CLK_I2C3	[9]	RW	Gating all clocks for I2C3 0 = Mask 1 = Pass	0x1
CLK_I2C2	[8]	RW	Gating all clocks for I2C2 0 = Mask 1 = Pass	0x1
CLK_I2C1	[7]	RW	Gating all clocks for I2C1 0 = Mask 1 = Pass	0x1
CLK_I2C0	[6]	RW	Gating all clocks for I2C0 0 = Mask 1 = Pass	0x1
RSVD	[5]	—	Reserved	0x1
CLK_UART4	[4]	RW	Gating all clocks for UART4 0 = Mask 1 = Pass	0x1
CLK_UART3	[3]	RW	Gating all clocks for UART3 0 = Mask 1 = Pass	0x1
CLK_UART2	[2]	RW	Gating all clocks for UART2 0 = Mask 1 = Pass	0x1
CLK_UART1	[1]	RW	Gating all clocks for UART1 0 = Mask 1 = Pass	0x1
CLK_UART0	[0]	RW	Gating all clocks for UART0 0 = Mask 1 = Pass	0x1

5.3.1.98 CLK_GATE_IP_PERIR

- Address = 0x1003_C960, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:18]	–	Reserved	0x3FFF
CLK_TMU_APBIF	[17]	RW	Gating all clocks for TMU_APBIF 0 = Mask 1 = Pass	0x1
CLK_KEYIF	[16]	RW	Gating all clocks for KEYIF 0 = Mask 1 = Pass	0x1
CLK_RTC	[15]	RW	Gating all clocks for RTC 0 = Mask 1 = Pass	0x1
CLK_WDT	[14]	RW	Gating all clocks for WDT 0 = Mask 1 = Pass	0x1
CLK_MCT	[13]	RW	Gating all clocks for Multi-Core Timer 0 = Mask 1 = Pass	0x1
CLK_SECKEY	[12]	RW	Gating all clocks for SECKEY 0 = Mask 1 = Pass	0x1
CLK_HDMI_CEC	[11]	RW	Gating all clocks for HDMI_CEC 0 = Mask 1 = Pass	0x1
CLK_TZPC5	[10]	RW	Gating all clocks for TZPC5 0 = Mask 1 = Pass	0x1
CLK_TZPC4	[9]	RW	Gating all clocks for TZPC4 0 = Mask 1 = Pass	0x1
CLK_TZPC3	[8]	RW	Gating all clocks for TZPC3 0 = Mask 1 = Pass	0x1
CLK_TZPC2	[7]	RW	Gating all clocks for TZPC2 0 = Mask 1 = Pass	0x1
CLK_TZPC1	[6]	RW	Gating all clocks for TZPC1 0 = Mask 1 = Pass	0x1
CLK_TZPC0	[5]	RW	Gating all clocks for TZPC0 0 = Mask 1 = Pass	0x1
CLK_CMU_DMCPART	[4]	RW	Gating all clocks for CMU_DMCPART 0 = Mask 1 = Pass	0x1

Name	Bit	Type	Description	Reset Value
RSVD	[3]	—	Reserved (should be 1'b1)	0x1
CLK_PMU_APBIF	[2]	RW	Gating all clocks for PMU_APBIF 0 = Mask 1 = Pass	0x1
CLK_SYSREG	[1]	RW	Gating all clocks for SYSREG 0 = Mask 1 = Pass	0x1
CLK_CHIP_ID	[0]	RW	Gating all clocks for CHIP ID 0 = Mask 1 = Pass	0x1

5.3.1.99 CLK_GATE_BLOCK

- Address = 0x1003_C970, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	—	Reserved	0xFFFFFFFF
CLK_GPS	[7]	RW	Gating all clocks for GPS_BLK (GPS) 0 = Mask 1 = Pass	0x1
RSVD	[6]	—	Reserved	0x1
CLK_LCD1	[5]	RW	Gating all clocks for LCD1_BLK (FIMD1, MIE1, DSIM1) 0 = Mask 1 = Pass	0x1
CLK_LCD0	[4]	RW	Gating all clocks for LCD0_BLK (FIMD0, MIE0, DSIM0) 0 = Mask 1 = Pass	0x1
CLK_G3D	[3]	RW	Gating all clocks for G3D_BLK (G3D) 0 = Mask 1 = Pass	0x1
CLK_MFC	[2]	RW	Gating all clocks for MFC_BLK (MFC) 0 = Mask 1 = Pass	0x1
CLK_TV	[1]	RW	Gating all clocks for TV_BLK (VP, MIXER, TVENC, HDMI) 0 = Mask 1 = Pass	0x1
CLK_CAM	[0]	RW	Gating all clocks for CAM_BLK (FIMC0,1,2,3) 0 = Mask 1 = Pass	0x1

5.3.1.100 CLKOUT_CMU_TOP

- Address = 0x1003_CA00, Reset Value = 0x0001_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:17]	–	Reserved	0x0
ENB_CLKOUT	[16]	RW	Enable CLKOUT 0 = Disable 1 = Enable	0x1
RSVD	[15:14]	–	Reserved	0x0
DIV_RATIO	[13:8]	RW	Divide ratio (Divide ratio = DIV_RATIO + 1)	0x0
RSVD	[7:5]	–	Reserved	0x0
MUX_SEL	[4:0]	RW	00000 = EPLL_FOUT 00001 = VPLL_FOUT 00010 = SCLK_HDMI24M 00011 = SCLK_USBPHY0 00100 = SCLK_USBPHY1 00101 = SCLK_HDMIPHY 00110 = AUDIOCDCLK0 00111 = AUDIOCDCLK1 01000 = AUDIOCDCLK2 01001 = SPDIF_EXTCLK 01010 = ACLK_160 01011 = ACLK_133 01100 = ACLK_200 01101 = ACLK_100 01110 = SCLK_MFC 01111 = SCLK_G3D 10000 = SCLK_G2D 10001 = CAM_A_PCLK 10010 = CAM_B_PCLK 10011 = S_RXBYTECLKHS0_2L 10100 = S_RXBYTECLKHS0_4L	0x0

5.3.1.101 CLKOUT_CMU_TOP_DIV_STAT

- Address = 0x1003_CA04, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	–	Reserved	0x0
DIV_STAT	[0]	R	DIVCLKOUT status 0 = Stable 1 = Divider is changing	0x0

5.3.1.102 CLK_SRC_DMC

- Address = 0x1004_0200, Reset Value = 0x0001_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:29]	–	Reserved	0x0
RSVD	[28]	RW	Reserved	0x0
RSVD	[27:25]	–	Reserved	0x0
RSVD	[24]	RW	Reserved	0x0
RSVD	[23:21]	–	Reserved	0x0
RSVD	[20]	RW	Reserved	0x0
MUX_PWI_SEL	[19:16]	RW	Control MUXPWI, which is the source clock of PWI 0000 = XXTI 0001 = XusbXTI 0010 = SCLK_HDMI24M 0011 = SCLK_USBPHY0 0100 = SCLK_USBPHY1 0101 = SCLK_HDMIPHY 0110 = SCLKMPPLL 0111 = SCLKEPLL 1000 = SCLKVPLL Others: Reserved	0x1
RSVD	[15:9]	–	Reserved	0x0
MUX_DPHY_SEL	[8]	RW	Control MUXDPHY 0 = SCLKMPPLL 1 = SCLKAPLL	0x0
RSVD	[7:5]	–	Reserved	0x0
MUX_DMC_BUS_SEL	[4]	RW	Control MUXDMC_BUS 0 = SCLKMPPLL 1 = SCLKAPLL	0x0
RSVD	[3:0]	–	Reserved	0x0

5.3.1.103 CLK_SRC_MASK_DMC

- Address = 0x1004_0300, Reset Value = 0x0001_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:17]	–	Reserved	0x0
PWI_MASK	[16]	RW	Mask output clock of MUXPWI 0 = Mask 1 = Unmask	0x1
RSVD	[15:0]	–	Reserved	0x0

5.3.1.104 CLK_MUX_STAT_DMC

- Address = 0x1004_0400, Reset Value = 0x1110_0110

Name	Bit	Type	Description	Reset Value
RSVD	[31]	–	Reserved	0x0
RSVD	[30:28]	–	Reserved	0x1
RSVD	[27]	–	Reserved	0x0
RSVD	[26:24]	–	Reserved	0x1
RSVD	[23]	–	Reserved	0x0
RSVD	[22:20]	–	Reserved	0x1
RSVD	[19:11]	–	Reserved	0x0
DPHY_SEL	[10:8]	R	Selection signal status of MUXDMC0 001 = SCLKMPLL 010 = SCLKAPLL 1xx: On changing	0x1
RSVD	[7]	–	Reserved	0x0
DMC_BUS_SEL	[6:4]	R	Selection signal status of MUXDMC_BUS 001 = SCLKMPLL 010 = SCLKAPLL 1xx: On changing	0x1
RSVD	[3:0]	–	Reserved	0x0

5.3.1.105 CLK_DIV_DMC0

- Address = 0x1004_0500, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:23]	–	Reserved	0x0
DMCP_RATIO	[22:20]	RW	DIVCK133 clock divider ratio ACLK_DMCP = ACLK_DMCD / (DMCP_RATIO + 1)	0x0
RSVD	[19]	–	Reserved	0x0
DMCD_RATIO	[18:16]	RW	DIVDMCD clock divider ratio ACLK_DMCD = DOUTDMC / (DMCD_RATIO + 1)	0x0
RSVD	[15]	–	Reserved	0x0
DMC_RATIO	[14:12]	RW	DIVDMC clock divider ratio DOUTDMC = MOUTDMC_BUS / (DMC_RATIO + 1)	0x0
RSVD	[11]	–	Reserved	0x0
DPHY_RATIO	[10:8]	RW	DIVDPHY clock divider ratio SCLK_DPHY = MOUTDPHY / (DPHY_RATIO + 1)	0x0
RSVD	[7]	–	Reserved	0x0
ACP_PCLK_RATIO	[6:4]	RW	DIVACP clock divider ratio PCLK_ACP = ACLK_ACP / (ACP_PCLK_RATIO + 1)	0x0
RSVD	[3]	–	Reserved	0x0
ACP_RATIO	[2:0]	RW	DIVACP clock divider ratio ACLK_ACP = MOUTDMC_BUS / (ACP_RATIO + 1)	0x0

5.3.1.106 CLK_DIV_DMC1

- Address = 0x1004_0504, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31]	–	Reserved	0x0
DPM_RATIO	[30:24]	RW	DIVDPM clock divider ratio which decides frequency of DPM channel clock.	0x0
RSVD	[23]	–	Reserved	0x0
DVSEM_RATIO	[22:16]	RW	DIVDVSEM clock divider ratio which decides frequency for PWM frame time slot in DVS emulation mode.	0x0
RSVD	[15:12]	–	Reserved	0x0
PWI_RATIO	[11:8]	RW	DIVPWI clock divider ratio SCLK_PWI = MOUTPWI / (PWI_RATIO + 1)	0x0
RSVD	[7:4]	–	Reserved	0x0
RSVD	[3:0]	RW	Reserved	0x0

5.3.1.107 CLK_DIV_STAT_DMC0

- Address = 0x1004_0600, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:21]	–	Reserved	0x0
DIV_DMCP	[20]	R	DIVDMCP status 0 = Stable 1 = Divider is changing	0x0
RSVD	[19:17]	–	Reserved	0x0
DIV_DMCD	[16]	R	DIVDMCD status 0 = Stable 1 = Divider is changing	0x0
RSVD	[15:13]	–	Reserved	0x0
DIV_DMC	[12]	R	DIVDMC status 0 = Stable 1 = Divider is changing	0x0
RSVD	[11:9]	–	Reserved	0x0
DIV_DPHY	[8]	R	DIVDPHY status 0 = Stable 1 = Divider is changing	0x0
RSVD	[7:5]	–	Reserved	0x0
DIV_ACP_PCLK	[4]	R	DIVACP_PCLK status 0 = Stable 1 = Divider is changing	0x0
RSVD	[3:1]	–	Reserved	0x0
DIV_ACP	[0]	R	DIVACP status 0 = Stable 1 = Divider is changing	0x0

5.3.1.108 CLK_DIV_STAT_DMC1

- Address = 0x1004_0604, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:25]	–	Reserved	0x0
DIV_DPM	[24]	R	DIVDPM status 0 = Stable 1 = Divider is changing	0x0
RSVD	[23:17]	–	Reserved	0x0
DIV_DVSEM	[16]	R	DIVDVSEM status 0 = Stable 1 = Divider is changing	0x0
RSVD	[15:9]	–	Reserved	0x0
DIV_PWI	[8]	R	DIVPWI status 0 = Stable 1 = Divider is changing	0x0
RSVD	[7:1]	–	Reserved	0x0
RSVD	[0]	–	Reserved	0x0

5.3.1.109 CLK_GATE_IP_DMC

- Address = 0x1004_0900, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	–	Reserved	0x7FF
RSVD	[25:23]	RW	Reserved	0x7
RSVD	[22:21]	–	Reserved (should be 2'b11)	0x3
CLK_GIC	[20]	RW	Gating all clocks for GIC 0 = Mask 1 = Pass	0x1
RSVD	[19]	–	Reserved	0x1
CLK_IEM_IEC	[18]	RW	Gating all clocks for IEM IEC 0 = Mask 1 = Pass	0x1
CLK_IEM_AP	[17]	RW	Gating all clocks for IEM APC 0 = Mask 1 = Pass	0x1
CLK_PPMUACP	[16]	RW	Gating all clocks for PPMUCPU 0 = Mask 1 = Pass	0x1
CLK_QESSS	[15]	RW	Gating all clocks for QESSS 0 = Mask 1 = Pass	0x1
RSVD	[14]	–	Reserved	0x1
CLK_ID_REMAPPER	[13]	RW	Gating all clocks for ID_REMAPPER 0 = Mask 1 = Pass	0x1
CLK_SMMUSSS	[12]	RW	Gating all clocks for SMMUSSS 0 = Mask 1 = Pass	0x1
RSVD	[11]	–	Reserved	0x1
CLK_PPMUCPU	[10]	RW	Gating all clocks for PPMUCPU 0 = Mask 1 = Pass	0x1
RSVD	[9:8]	–	Reserved	0x3
CLK_QECPU	[7]	RW	Gating all clocks for QECPUs 0 = Mask 1 = Pass	0x1
RSVD	[6:5]	–	Reserved	0x3
CLK_SSS	[4]	RW	Gating all clocks for SSS 0 = Mask 1 = Pass	0x1
RSVD	[3]	–	Reserved	0x1
CLK_INT_COMB	[2]	RW	Gating all clocks for INT_COMB 0 = Mask	0x1

Name	Bit	Type	Description	Reset Value
			1 = Pass Gating all clocks for DMC1 0 = Mask 1 = Pass	
CLK_DMC1	[1]	RW	Gating all clocks for DMC1 0 = Mask 1 = Pass	0x1
CLK_DMC0	[0]	RW	Gating all clocks for DMC0 0 = Mask 1 = Pass	0x1

Caution: When CLK_DMC0 or CLK_DMC1 field in CLK_GATE_IP_DMC register is set to zero, it is not allowed to access registers of PPMU_DMC0 or PPMU_DMC1, respectively.

5.3.1.110 CLKOUT_CMU_DMC

- Address = 0x1004_0A00, Reset Value = 0x0001_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:17]	–	Reserved	0x0
ENB_CLKOUT	[16]	RW	Enable CLKOUT 0 = Disable 1 = Enable	0x1
RSVD	[15:14]	–	Reserved	0x0
DIV_RATIO	[13:8]	RW	Divide ratio (Divide ratio = DIV_RATIO + 1)	0x0
RSVD	[7:5]	–	Reserved	0x0
MUX_SEL	[4:0]	RW	00000 = ACLK_DMCD 00001 = ACLK_DMCP 00010 = ACLK_ACP 00011 = PCLK_ACP 00100 = SCLK_DMC 00101 = SCLK_DPHY 00110 = Reserved 00111 = SCLK_PWI 01000 = Reserved	0x0

5.3.1.111 CLKOUT_CMU_DMC_DIV_STAT

- Address = 0x1004_0A04, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	–	Reserved	0x0
DIV_STAT	[0]	R	DIVCLKOUT status 0 = Stable 1 = Divider is changing	0x0

5.3.1.112 DCGIDX_MAP0

- Address = 0x1004_1000, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
DCGIDX_MAP0	[31:0]	RW	IEC configuration for DCG index map[31:0]	0xFFFFFFFF

5.3.1.113 DCGIDX_MAP1

- Address = 0x1004_1004, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
DCGIDX_MAP1	[31:0]	RW	IEC configuration for DCG index map[63:32]	0xFFFFFFFF

5.3.1.114 DCGIDX_MAP2

- Address = 0x1004_1008, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
DCGIDX_MAP2	[31:0]	RW	IEC configuration for DCG index map[95:64]	0xFFFFFFFF

5.3.1.115 DCGPERF_MAP0

- Address = 0x1004_1020, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
DCGPERF_MAP0	[31:0]	RW	DCG performance map[31:0]	0xFFFFFFFF

5.3.1.116 DCGPERF_MAP1

- Address = 0x1004_1024, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
DCGPERF_MAP1	[31:0]	RW	DCG performance map[63:32]	0xFFFFFFFF

5.3.1.117 DVCIDX_MAP

- Address = 0x1004_1040, Reset Value = 0x00FF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	—	Reserved	0x0
DCGPERF_MAP0	[23:0]	RW	IEC configuration for DVC index map[23:0]	0xFFFF

5.3.1.118 FREQ_CPU

- Address = 0x1004_1060, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	–	Reserved	0x0
FREQ_CPU	[23:0]	RW	Maximum frequency of CPU in KHz	0x0

5.3.1.119 FREQ_DPM

- Address = 0x1004_1064, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	–	Reserved	0x0
FREQ_DPM	[23:0]	RW	Maximum frequency of DPM accumulators	0x0

5.3.1.120 DVSEMCLK_EN

- Address = 0x1004_1080, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	–	Reserved	0x0
DVSEMCLK_EN	[0]	RW	DVS emulation clock enable	0x0

5.3.1.121 MAXPERF

- Address = 0x1004_1084, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	–	Reserved	0x0
MAXPERF_EN	[0]	RW	MAX performance enable 0 = Disable 1 = Enable	0x0

5.3.1.122 APLL_LOCK

- Address = 0x1004_4000, Reset Value = 0x0000_0FFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	–	Reserved	0x0
PLL_LOCKTIME	[15:0]	RW	Required period to generate a stable clock output	0xFFFF

5.3.1.123 MPLL_LOCK

- Address = 0x1004_4008, Reset Value = 0x0000_0FFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	—	Reserved	0x0
PLL_LOCKTIME	[15:0]	RW	Required period to generate a stable clock output	0xFFFF

Lock time of PLL is related Fref which is FIN / PDIV.

- 1) Lock time of APLL and MPLL is changed as the frequency of Fref.
 $<30\mu s @ Fref. = 6 \text{ MHz}/8 \text{ MHz}$, $<40\mu s @ Fref. = 4 \text{ MHz}$, $<60\mu s @ Fref. = 2 \text{ MHz}$, $<100\mu s @ Fref. = 1 \text{ MHz}$

5.3.1.124 APLL_CON0

- Address = 0x1004_4100, Reset Value = 0x00C8_0601

Name	Bit	Type	Description	Reset Value
ENABLE	[31]	RW	PLL enable control 0 = Disable 1 = Enable	0x0
RSVD	[30]	-	Reserved	0x0
LOCKED	[29]	R	PLL locking indication 0 = Unlocked 1 = Locked	0x0
RSVD	[28]	-	Reserved	0x0
FSEL	[27]	RWX	Monitoring frequency select pin 0 = FVCO_OUT = FREF 1 = FVCO_OUT = FVCO	0x0
RSVD	[26]	-	Reserved	0x0
MDIV	[25:16]	RWX	PLL M divide value	0xC8
RSVD	[15:14]	-	Reserved	0x0
PDIV	[13:8]	RWX	PLL P divide value	0x6
RSVD	[7:3]	-	Reserved	0x0
SDIV	[2:0]	RWX	PLL S divide value	0x1

The reset value of APLL_CON0 generates 800 MHz output clock for the input clock frequency of 24 MHz.

Equation to calculate the output frequency:

$$F_{OUT} = MDIV \times FIN / (PDIV \times 2^{SDIV-1})$$

where MDIV, PDIV, SDIV for APLL and MPLL must meet the following conditions:

PDIV: $1 \leq PDIV \leq 63$

MDIV: $64 \leq MDIV \leq 1023$

SDIV: $1 \leq SDIV \leq 5$

Fref (= FIN / PDIV): $1 \text{ MHz} \leq Fref \leq 12 \text{ MHz}$

FVCO (= $2 \times MDIV \times FIN / PDIV$): $1000 \text{ MHz} \leq FVCO \leq 2060 \text{ MHz}$

Refer to [5.2.2.1 Recommended PMS Value for APLL & MPLL](#) for recommended PMS values.

5.3.1.125 APLL_CON1

- Address = 0x1004_4104, Reset Value = 0x0000_001C

Name	Bit	Type	Description	Reset Value
AFC_ENB	[31]	RWX	Decides whether AFC is enabled or not. Active low 0 = AFC is enabled 1 = AFC is disabled	0x0
RSVD	[30:5]	—	Reserved	0x0
AFC	[4:0]	RWX	AFC value	0x1C

AFC(Adaptive Frequency Calibrator) automatically selects adaptive frequency curve of VCO using switched current bank for wide range, high phase noise (or Jitter), and fast lock time.

Refer to [5.2.2.1 Recommended PMS Value for APLL & MPLL](#) for recommended AFC_ENB and AFC values.

5.3.1.126 MPLL_CON0

- Address = 0x1004_4108, Reset Value = 0x00C8_0601

Name	Bit	Type	Description	Reset Value
ENABLE	[31]	RW	PLL enable control 0 = Disable 1 = Enable	0x0
RSVD	[30]	-	Reserved	0x0
LOCKED	[29]	R	PLL locking indication 0 = Unlocked 1 = Locked	0x0
RSVD	[28]	-	Reserved	0x0
FSEL	[27]	RW	Monitoring frequency select pin 0 = FVCO_OUT = FREF 1 = FVCO_OUT = FVCO	0x0
RSVD	[26]	-	Reserved	0x0
MDIV	[25:16]	RW	PLL M divide value	0xC8
RSVD	[15:14]	-	Reserved	0x0
PDIV	[13:8]	RW	PLL P divide value	0x6
RSVD	[7:3]	-	Reserved	0x0
SDIV	[2:0]	RW	PLL S divide value	0x1

The reset value of MPLL_CON0 generates 800 MHz output clock for the input clock frequency of 24 MHz.

Equation to calculate the output frequency:

$$F_{OUT} = MDIV \times FIN / (PDIV \times 2^{SDIV-1})$$

where MDIV, PDIV, SDIV for APLL and MPLL must meet the following conditions:

PDIV: $1 \leq PDIV \leq 63$

MDIV: $64 \leq MDIV \leq 1023$

SDIV: $1 \leq SDIV \leq 5$

Fref (= FIN / PDIV): $1 \text{ MHz} \leq Fref \leq 12 \text{ MHz}$

FVCO (= $2 \times MDIV \times FIN / PDIV$): $1000 \text{ MHz} \leq FVCO \leq 2060 \text{ MHz}$

Refer to [5.2.2.1 Recommended PMS Value for APLL & MPLL](#) for recommended PMS values.

5.3.1.127 MPLL_CON1

- Address = 0x1004_410C, Reset Value = 0x0000_001C

Name	Bit	Type	Description	Reset Value
AFC_ENB	[31]	RW	Decides whether AFC is enabled or not. Active low 0 = AFC is enabled 1 = AFC is disabled	0x0
RSVD	[30:5]	—	Reserved	0x0
AFC	[4:0]	RW	AFC value	0x1C

Refer to [5.2.2.1 Recommended PMS Value for APLL & MPLL](#) for recommended AFC_ENB and AFC values.

5.3.1.128 CLK_SRC_CPU

- Address = 0x1004_4200, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:21]	–	Reserved	0x0
MUX_HPM_SEL	[20]	RW	Control MUXHPM 0 = MOUTAPLL 1 = SCLKMPPLL	0x0
RSVD	[19:17]	–	Reserved	0x0
MUX_CORE_SEL	[16]	RW	Control MUXCORE 0 = MOUTAPLL 1 = SCLKMPPLL	0x0
RSVD	[15:9]	–	Reserved	0x0
MUX_MPPLL_SEL	[8]	RW	Control MUXMPPLL 0 = FINPLL 1 = MOUTMPPLLFOU	0x0
RSVD	[7:1]	–	Reserved	0x0
MUX_APPLL_SEL	[0]	RW	Control MUXAPPLL 0 = FINPLL 1 = MOUTAPPLLFOU	0x0

5.3.1.129 CLK_MUX_STAT_CPU

- Address = 0x1004_4400, Reset Value = 0x0011_0101

Name	Bit	Type	Description	Reset Value
RSVD	[31:23]	–	Reserved	0x0
HPM_SEL	[22:20]	R	Selection signal status of MUXHPM 001 = MOUTAPLL 010 = SCLKMPPLL 1xx: On changing	0x1
RSVD	[19]	–	Reserved	0x0
CORE_SEL	[18:16]	R	Selection signal status of MUXCORE 001 = MOUTAPLL 010 = SCLKMPPLL 1xx: On changing	0x1
RSVD	[15:11]	–	Reserved	0x0
MPLL_SEL	[10:8]	R	Selection signal status of MUXMPPLL 001 = FINPLL 010 = MOUTMPPLLFOU 1xx: On changing	0x1
RSVD	[7:3]	–	Reserved	0x0
APPLL_SEL	[2:0]	R	Selection signal status of MUXAPPLL 001 = FINPLL 010 = MOUTAPPLLFOU 1xx: On changing	0x1

5.3.1.130 CLK_DIV_CPU0

- Address = 0x1004_4500, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31]	—	Reserved	0x0
CORE2_RATIO	[30:28]	RW	DIVCORE2 clock divider ratio ARMCLK = DOUTCORE / (CORE2_RATIO + 1)	0x0
RSVD	[27]	—	Reserved	0x0
APLL_RATIO	[26:24]	RW	DIVAPLL clock divider ratio SCLKAPLL = MOUTAPLL / (APLL_RATIO + 1)	0x0
RSVD	[23]	—	Reserved	0x0
PCLK_DBG_RATIO	[22:20]	RW	DIVPCLK_DBG clock divider ratio PCLK_DBG = ATCLK / (PCLK_DBG_RATIO + 1)	0x0
RSVD	[19]	—	Reserved	0x0
ATB_RATIO	[18:16]	RW	DIVATB clock divider ratio ATCLK = MOUTCORE / (ATB_RATIO + 1)	0x0
RSVD	[15]	—	Reserved	0x0
PERIPH_RATIO	[14:12]	RW	DIVPERIPH clock divider ratio PERIPHCLK = DOUTCORE / (PERIPH_RATIO + 1)	0x0
RSVD	[11]	—	Reserved	0x0
COREM1_RATIO	[10:8]	RW	DIVCOREM1 clock divider ratio ACLK_COREM1 = ARMCLK / (COREM1_RATIO + 1)	0x0
RSVD	[7]	—	Reserved	0x0
COREM0_RATIO	[6:4]	RW	DIVCOREM0 clock divider ratio ACLK_COREM0 = ARMCLK / (COREM0_RATIO + 1)	0x0
RSVD	[3]	—	Reserved	0x0
CORE_RATIO	[2:0]	RWX	DIVCORE clock divider ratio ARMCLK = MOUTCORE / (CORE_RATIO + 1)	0x0

5.3.1.131 CLK_DIV_CPU1

- Address = 0x1004_4504, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:7]	—	Reserved	0x0
HPM_RATIO	[6:4]	RWX	DIVHPM clock divider ratio SCLK_HPM = DOUTCOPY / (HPM_RATIO + 1)	0x0
RSVD	[3]	—	Reserved	0x0
COPY_RATIO	[2:0]	RWX	DIVCOPY clock divider ratio DOUTCOPY = MOUTHPM / (COPY_RATIO + 1)	0x0

5.3.1.132 CLK_DIV_STAT_CPU0

- Address = 0x1004_4600, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:29]	–	Reserved	0x0
DIV_CORE2	[28]	R	DIVCORE2 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[27:25]	–	Reserved	0x0
DIV_APPLL	[24]	R	DIVAPPLL status 0 = Stable 1 = Divider is changing	0x0
RSVD	[23:21]	–	Reserved	0x0
DIV_PCLK_DBG	[20]	R	DIVPCLK_DBG status 0 = Stable 1 = Divider is changing	0x0
RSVD	[19:17]	–	Reserved	0x0
DIV_ATB	[16]	R	DIVATB status 0 = Stable 1 = Divider is changing	0x0
RSVD	[15:13]	–	Reserved	0x0
DIV_PERIPH	[12]	R	DIVPERIPH status 0 = Stable 1 = Divider is changing	0x0
RSVD	[11:9]	–	Reserved	0x0
DIV_COREM1	[8]	R	DIVCOREM1 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[7:5]	–	Reserved	0x0
DIV_COREM0	[4]	R	DIVCOREM0 status 0 = Stable 1 = Divider is changing	0x0
RSVD	[3:1]	–	Reserved	0x0
DIV_CORE	[0]	R	DIVCORE status 0 = Stable 1 = Divider is changing	0x0

5.3.1.133 CLK_DIV_STAT_CPU1

- Address = 0x1004_4604, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	—	Reserved	0x0
DIV_HPM	[4]	R	DIVHPM status 0 = Stable 1 = Divider is changing	0x0
RSVD	[3:1]	—	Reserved	0x0
DIV_COPY	[0]	R	DIVCOPY status 0 = Stable 1 = Divider is changing	0x0

5.3.1.134 CLK_GATE_SCLK_CPU

- Address = 0x1004_4800, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	—	Reserved	0xFFFF_FFF7
SCLK_APOLL	[0]	RW	Gating SCLK_APOLL 0 = Mask 1 = Pass	0x1

5.3.1.135 CLK_GATE_IP_CPU

- Address = 0x1004_4900, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:2]	—	Reserved	0xFFFF_FFF3
CLK_CSSYS	[1]	RW	Gating all clocks for CoreSight and SecureJTAG 0 = Mask 1 = Pass	0x1
CLK_HPM	[0]	RW	Gating all clocks for HPM 0 = Mask 1 = Pass	0x1

5.3.1.136 CLKOUT_CMU_CPU

- Address = 0x1004_4A00, Reset Value = 0x0001_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:17]	–	Reserved	0x0
ENB_CLKOUT	[16]	RW	Enable CLKOUT 0 = Disable 1 = Enable	0x1
RSVD	[15:14]	–	Reserved	0x0
DIV_RATIO	[13:8]	RW	Divide ratio (Divide ratio = DIV_RATIO + 1)	0x0
RSVD	[7:5]	–	Reserved	0x0
MUX_SEL	[4:0]	RW	00000 = APLL_FOUT/2 00001 = Reserved 00010 = MPLL_FOUT/2 00011 = Reserved 00100 = ARMCLK/2 00101 = ACLK_COREM0 00110 = ACLK_COREM1 00111 = ACLK_CORES 01000 = ATCLK 01001 = PERIPHCLK 01010 = PCLK_DBG 01011 = SCLK_HPM	0x0

5.3.1.137 CLKOUT_CMU_CPU_DIV_STAT

- Address = 0x1004_4A04, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	–	Reserved	0x0
DIV_STAT	[0]	R	DIVCLKOUT status 0 = Stable 1 = Divider is changing	0x0

5.3.1.138 ARMCLK_STOPCTRL

- Address = 0x1004_5000, Reset Value = 0x0000_0044

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	–	Reserved	0x0
PRE_WAIT_CNT	[7:4]	RW	Clock freeze cycle before ARM clamp(CLAMPCORE0,CLAMPCORE1,CLAMPICO REOUT,CLAMPL2_0,CLAMPL2_1) or reset signal (nCPURESET,nDBGRESET,nSCURESET,L2nRESET,nWDRESET,nPERIPHRESET,nPTMRESET) transition	0x4
POST_WAIT_CNT	[3:0]	RW	Clock freeze cycle after ARM clamp(CLAMPCORE0,CLAMPCORE1,CLAMPICO REOUT,CLAMPL2_0,CLAMPL2_1) or reset signal (nCPURESET,nDBGRESET,nSCURESET,L2nRESET,nWDRESET,nPERIPHRESET,nPTMRESET) transition	0x4

5.3.1.139 ATCLK_STOPCTRL

- Address = 0x1004_5004, Reset Value = 0x0000_0044

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	–	Reserved	0x0
PRE_WAIT_CNT	[7:4]	RW	Clock freeze cycle before ATRESETn, nPRESETDBG, CSSYS_nRESET signal transition	0x4
POST_WAIT_CNT	[3:0]	RW	Clock freeze cycle after ATRESETn, nPRESETDBG, CSSYS_nRESET signal transition	0x4

5.3.1.140 PARITYFAIL_STATUS

- Address = 0x1004_5010, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:18]	–	Reserved	0x0
PARITYFAILSCU	[17:16]	R	Parity output pin from SCU tag RAMs. ORed output from each Cortex-A9 processor present in the design	0x0
PARITYFAIL1	[15:8]	R	Parity output pin from the RAM array for CPU1 Indicates a parity fail 0 = No parity fail 1 = Parity fail Bit 7 BTAC parity error Bit 6 GHB parity error Bit 5 Instruction tag RAM parity error Bit 4 Instruction data RAM parity error Bit 3 Main TLB parity error Bit 2 D outer RAM parity error Bit 1 Data tag RAM parity error Bit 0 Data data RAM parity error	0x0
PARITYFAIL0	[7:0]	R	Parity output pin from the RAM array for CPU0 Indicates a parity fail 0 = No parity fail 1 = Parity fail Bit 7 BTAC parity error Bit 6 GHB parity error Bit 5 Instruction tag RAM parity error Bit 4 Instruction data RAM parity error Bit 3 Main TLB parity error Bit 2 D outer RAM parity error Bit 1 Data tag RAM parity error Bit 0 Data data RAM parity error	0x0

5.3.1.141 PARITYFAIL_CLEAR

- Address = 0x1004_5014, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:18]	–	Reserved	0x0
PARITYFAILSCU	[17:16]	RWX	Parity output pin from SCU tag RAMs. ORed output from each Cortex-A9 processor present in the design	0x0
PARITYFAIL1	[15:8]	RWX	Parity output pin from the RAM array for CPU1 Indicates a parity fail 0 = No parity fail 1 = Parity fail Bit 7 BTAC parity error Bit 6 GHB parity error Bit 5 Instruction tag RAM parity error Bit 4 Instruction data RAM parity error Bit 3 Main TLB parity error Bit 2 D outer RAM parity error Bit 1 Data tag RAM parity error Bit 0 Data data RAM parity error	0x0
PARITYFAIL0	[7:0]	RWX	Parity output pin from the RAM array for CPU0 Indicates a parity fail 0 = No parity fail 1 = Parity fail Bit 7 BTAC parity error Bit 6 GHB parity error Bit 5 Instruction tag RAM parity error Bit 4 Instruction data RAM parity error Bit 3 Main TLB parity error Bit 2 D outer RAM parity error Bit 1 Data tag RAM parity error Bit 0 Data data RAM parity error	0x0

5.3.1.142 PWR_CTRL

- Address = 0x1004_5020, Reset Value = 0x0000_0033

Name	Bit	Type	Description	Reset Value
RSVD	[31]	–	Reserved	0x0
CORE2_RATIO	[30:28]	RW	DIVCORE2 clock divider ratio when both ARM cores are in Wait For Interrupt/Event state	0x0
RSVD	[27:21]	–	Reserved	0x0
CSCLK_AUTO_ENB_IN_DEBUG	[20]	RW	Force CoreSight clocks to toggle when debugger is attached 0 = Disable 1 = Enable	0x0
RSVD	[19]	–	Reserved	0x0
CORE_RATIO	[18:16]	RW	DIVCORE clock divider ratio when both ARM cores are in Wait For Interrupt/Event state	0x0
RSVD	[15:10]	–	Reserved	0x0
DIVCORE2_DOWN_ENB	[9]	RW	Enable ARMCLK down feature with both ARM cores in IDLE mode for DIVCORE2 0 = Disable 1 = Enable	0x0
DIVCORE_DOWN_ENB	[8]	RW	Enable ARMCLK down feature with both ARM cores in IDLE mode for DIVCORE 0 = Disable 1 = Enable	0x0
RSVD	[7:6]	–	Reserved	0x0
USE_STANDBYWFE_ARM_CORE1	[5]	RW	Use ARM CORE1 STANDBYWFE to change ARMCLK frequency in ARM IDLE state	0x1
USE_STANDBYWFE_ARM_CORE0	[4]	RW	Use ARM CORE0 STANDBYWFE to change ARMCLK frequency in ARM IDLE state	0x1
RSVD	[3:2]	–	Reserved	0x0
USE_STANDBYWFI_ARM_CORE1	[1]	RW	Use ARM CORE1 STANDBYWFI to change ARMCLK frequency in ARM IDLE state	0x1
USE_STANDBYWFI_ARM_CORE0	[0]	RW	Use ARM CORE0 STANDBYWFI to change ARMCLK frequency in ARM IDLE state	0x1

5.3.1.143 APLL_CON0_L8

- Address = 0x1004_5100, Reset Value = 0x00C8_0601

Name	Bit	Type	Description	Reset Value
RSVD	[31:28]	—	Reserved	0x0
FSEL	[27]	RW	APLL FSEL value	0x0
RSVD	[26]	—	Reserved	0x0
MDIV	[25:16]	RW	APLL M divide value	0xC8
RSVD	[15:14]	—	Reserved	0x0
PDIV	[13:8]	RW	APLL P divide value	0x6
RSVD	[7:3]	—	Reserved	0x0
SDIV	[2:0]	RW	APLL S divide value	0x1

5.3.1.144 APLL_CON0_L7

- Address = 0x1004_5104, Reset Value = 0x00C8_0601

Name	Bit	Type	Description	Reset Value
RSVD	[31:28]	—	Reserved	0x0
FSEL	[27]	RW	APLL FSEL value	0x0
RSVD	[26]	—	Reserved	0x0
MDIV	[25:16]	RW	APLL M divide value	0xC8
RSVD	[15:14]	—	Reserved	0x0
PDIV	[13:8]	RW	APLL P divide value	0x6
RSVD	[7:3]	—	Reserved	0x0
SDIV	[2:0]	RW	APLL S divide value	0x1

5.3.1.145 APLL_CON0_L6

- Address = 0x1004_5108, Reset Value = 0x00C8_0601

Name	Bit	Type	Description	Reset Value
RSVD	[31:28]	—	Reserved	0x0
FSEL	[27]	RW	APLL FSEL value	0x0
RSVD	[26]	—	Reserved	0x0
MDIV	[25:16]	RW	APLL M divide value	0xC8
RSVD	[15:14]	—	Reserved	0x0
PDIV	[13:8]	RW	APLL P divide value	0x6
RSVD	[7:3]	—	Reserved	0x0
SDIV	[2:0]	RW	APLL S divide value	0x1

5.3.1.146 APLL_CON0_L5

- Address = 0x1004_510C, Reset Value = 0x00C8_0601

Name	Bit	Type	Description	Reset Value
RSVD	[31:28]	—	Reserved	0x0
FSEL	[27]	RW	APLL FSEL value	0x0
RSVD	[26]	—	Reserved	0x0
MDIV	[25:16]	RW	APLL M divide value	0xC8
RSVD	[15:14]	—	Reserved	0x0
PDIV	[13:8]	RW	APLL P divide value	0x6
RSVD	[7:3]	—	Reserved	0x0
SDIV	[2:0]	RW	APLL S divide value	0x1

5.3.1.147 APLL_CON0_L4

- Address = 0x1004_5110, Reset Value = 0x00C8_0601

Name	Bit	Type	Description	Reset Value
RSVD	[31:28]	—	Reserved	0x0
FSEL	[27]	RW	APLL FSEL value	0x0
RSVD	[26]	—	Reserved	0x0
MDIV	[25:16]	RW	APLL M divide value	0xC8
RSVD	[15:14]	—	Reserved	0x0
PDIV	[13:8]	RW	APLL P divide value	0x6
RSVD	[7:3]	—	Reserved	0x0
SDIV	[2:0]	RW	APLL S divide value	0x1

5.3.1.148 APLL_CON0_L3

- Address = 0x1004_5114, Reset Value = 0x00C8_0601

Name	Bit	Type	Description	Reset Value
RSVD	[31:28]	—	Reserved	0x0
FSEL	[27]	RW	APLL FSEL value	0x0
RSVD	[26]	—	Reserved	0x0
MDIV	[25:16]	RW	APLL M divide value	0xC8
RSVD	[15:14]	—	Reserved	0x0
PDIV	[13:8]	RW	APLL P divide value	0x6
RSVD	[7:3]	—	Reserved	0x0
SDIV	[2:0]	RW	APLL S divide value	0x1

5.3.1.149 APLL_CON0_L2

- Address = 0x1004_5118, Reset Value = 0x00C8_0601

Name	Bit	Type	Description	Reset Value
RSVD	[31:28]	–	Reserved	0x0
FSEL	[27]	RW	APLL FSEL value	0x0
RSVD	[26]	–	Reserved	0x0
MDIV	[25:16]	RW	APLL M divide value	0xC8
RSVD	[15:14]	–	Reserved	0x0
PDIV	[13:8]	RW	APLL P divide value	0x6
RSVD	[7:3]	–	Reserved	0x0
SDIV	[2:0]	RW	APLL S divide value	0x1

5.3.1.150 APLL_CON0_L1

- Address = 0x1004_511C, Reset Value = 0x00C8_0601

Name	Bit	Type	Description	Reset Value
RSVD	[31:28]	–	Reserved	0x0
FSEL	[27]	RW	APLL FSEL value	0x0
RSVD	[26]	–	Reserved	0x0
MDIV	[25:16]	RW	APLL M divide value	0xC8
RSVD	[15:14]	–	Reserved	0x0
PDIV	[13:8]	RW	APLL P divide value	0x6
RSVD	[7:3]	–	Reserved	0x0
SDIV	[2:0]	RW	APLL S divide value	0x1

5.3.1.151 IEM_CONTROL

- Address = 0x1004_5120, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	-	Reserved	0x0
SET_LEVEL	[23:16]	RW	Selects performance level 0xFF = Performance level-8 0x7F = Performance level-7 0x3F = Performance level-6 0x1F = Performance level-5 0x0F = Performance level-4 0x07 = Performance level-3 0x03 = Performance level-1 0x01 = Performance level-1	0x0
RSVD	[15:1]	-	Reserved	0x0
ENABLE_CHG_MUX_SEL	[4]	RW	Enable the feature to change MUXCORE selection from APLL to MPLL when PLL setting changes by IEM 0 = Disable 1 = Enable	0x0
RSVD	[3:1]	-	Reserved	0x0
ENABLE_IEM	[0]	RW	Enables IEM	0x0

5.3.1.152 APLL_CON1_L8

- Address = 0x1004_5200, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
AFC_ENB	[31]	RW	Decides whether AFC is enabled or not. Active low 0 = AFC is enabled 1 = AFC is disabled	0x0
RSVD	[30:5]	-	Reserved	0x0
AFC	[4:0]	RW	AFC value	0x0

5.3.1.153 APLL_CON1_L7

- Address = 0x1004_5204, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
AFC_ENB	[31]	RW	Decides whether AFC is enabled or not. Active low 0 = AFC is enabled 1 = AFC is disabled	0x0
RSVD	[30:5]	-	Reserved	0x0
AFC	[4:0]	RW	AFC value	0x0

5.3.1.154 APLL_CON1_L6

- Address = 0x1004_5208, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
AFC_ENB	[31]	RW	Decides whether AFC is enabled or not. Active low 0 = AFC is enabled 1 = AFC is disabled	0x0
RSVD	[30:5]	—	Reserved	0x0
AFC	[4:0]	RW	AFC value	0x0

5.3.1.155 APLL_CON1_L5

- Address = 0x1004_520C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
AFC_ENB	[31]	RW	Decides whether AFC is enabled or not. Active low 0 = AFC is enabled 1 = AFC is disabled	0x0
RSVD	[30:5]	—	Reserved	0x0
AFC	[4:0]	RW	AFC value	0x0

5.3.1.156 APLL_CON1_L4

- Address = 0x1004_5210, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
AFC_ENB	[31]	RW	Decides whether AFC is enabled or not. Active low 0 = AFC is enabled 1 = AFC is disabled	0x0
RSVD	[30:5]	—	Reserved	0x0
AFC	[4:0]	RW	AFC value	0x0

5.3.1.157 APLL_CON1_L3

- Address = 0x1004_5214, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
AFC_ENB	[31]	RW	Decides whether AFC is enabled or not. Active low 0 = AFC is enabled 1 = AFC is disabled	0x0
RSVD	[30:5]	—	Reserved	0x0
AFC	[4:0]	RW	AFC value	0x0

5.3.1.158 APLL_CON1_L2

- Address = 0x1004_5218, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
AFC_ENB	[31]	RW	Decides whether AFC is enabled or not. Active low 0 = AFC is enabled 1 = AFC is disabled	0x0
RSVD	[30:5]	—	Reserved	0x0
AFC	[4:0]	RW	AFC value	0x0

5.3.1.159 APLL_CON1_L1

- Address = 0x1004_521C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
AFC_ENB	[31]	RW	Decides whether AFC is enabled or not. Active low 0 = AFC is enabled 1 = AFC is disabled	0x0
RSVD	[30:5]	—	Reserved	0x0
AFC	[4:0]	RW	AFC value	0x0

5.3.1.160 CLKDIV_IEM_L8

- Address = 0x1004_5300, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:23]	—	Reserved	0x0
HPM_RATIO	[22:20]	RW	DIVHPM clock divider ratio SCLK_HPM = DOUTCOPY / (HPM_RATIO + 1)	0x0
RSVD	[19]	—	Reserved	0x0
COPY_RATIO	[18:16]	RW	DIVCOPY clock divider ratio DOUTCOPY = MOUTHPM / (COPY_RATIO + 1)	0x0
RSVD	[15:4]	—	Reserved	0x0
CORE_RATIO	[3:0]	RW	DIVAPLL clock divider ratio ARMCLK = MOUTCORE / (CORE_RATIO + 1)	0x0

5.3.1.161 CLKDIV_IEM_L7

- Address = 0x1004_5304, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:23]	–	Reserved	0x0
HPM_RATIO	[22:20]	RW	DIVHPM clock divider ratio SCLK_HPM = DOUTCOPY / (HPM_RATIO + 1)	0x0
RSVD	[19]	–	Reserved	0x0
COPY_RATIO	[18:16]	RW	DIVCOPY clock divider ratio DOUTCOPY = MOUTHPM / (COPY_RATIO + 1)	0x0
RSVD	[15:4]	–	Reserved	0x0
CORE_RATIO	[3:0]	RW	DIVAPLL clock divider ratio ARMCLK = MOUTCORE / (CORE_RATIO + 1)	0x0

5.3.1.162 CLKDIV_IEM_L6

- Address = 0x1004_5308, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:23]	–	Reserved	0x0
HPM_RATIO	[22:20]	RW	DIVHPM clock divider ratio SCLK_HPM = DOUTCOPY / (HPM_RATIO + 1)	0x0
RSVD	[19]	–	Reserved	0x0
COPY_RATIO	[18:16]	RW	DIVCOPY clock divider ratio DOUTCOPY = MOUTHPM / (COPY_RATIO + 1)	0x0
RSVD	[15:4]	–	Reserved	0x0
CORE_RATIO	[3:0]	RW	DIVAPLL clock divider ratio ARMCLK = MOUTCORE / (CORE_RATIO + 1)	0x0

5.3.1.163 CLKDIV_IEM_L5

- Address = 0x1004_530C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:23]	–	Reserved	0x0
HPM_RATIO	[22:20]	RW	DIVHMPM clock divider ratio SCLK_HPM = DOUTCOPY / (HPM_RATIO + 1)	0x0
RSVD	[19]	–	Reserved	0x0
COPY_RATIO	[18:16]	RW	DIVCOPY clock divider ratio DOUTCOPY = MOUTHMPM / (COPY_RATIO + 1)	0x0
RSVD	[15:4]	–	Reserved	0x0
CORE_RATIO	[3:0]	RW	DIVAPLL clock divider ratio ARMCLK = MOUTCORE / (CORE_RATIO + 1)	0x0

5.3.1.164 CLKDIV_IEM_L4

- Address = 0x1004_5310, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:23]	–	Reserved	0x0
HPM_RATIO	[22:20]	RW	DIVHMPM clock divider ratio SCLK_HPM = DOUTCOPY/(HPM_RATIO + 1)	0x0
RSVD	[19]	–	Reserved	0x0
COPY_RATIO	[18:16]	RW	DIVCOPY clock divider ratio DOUTCOPY = MOUTHMPM/(COPY_RATIO + 1)	0x0
RSVD	[15:4]	–	Reserved	0x0
CORE_RATIO	[3:0]	RW	DIVAPLL clock divider ratio ARMCLK = MOUTCORE/(CORE_RATIO + 1)	0x0

5.3.1.165 CLKDIV_IEM_L3

- Address = 0x1004_5314, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:23]	–	Reserved	0x0
HPM_RATIO	[22:20]	RW	DIVHPM clock divider ratio SCLK_HPM = DOUTCOPY/(HPM_RATIO + 1)	0x0
RSVD	[19]	–	Reserved	0x0
COPY_RATIO	[18:16]	RW	DIVCOPY clock divider ratio DOUTCOPY = MOUTHPM/(COPY_RATIO + 1)	0x0
RSVD	[15:4]	–	Reserved	0x0
CORE_RATIO	[3:0]	RW	DIVAPLL clock divider ratio ARMCLK = MOUTCORE/(CORE_RATIO + 1)	0x0

5.3.1.166 CLKDIV_IEM_L2

- Address = 0x1004_5318, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:23]	–	Reserved	0x0
HPM_RATIO	[22:20]	RW	DIVHPM clock divider ratio SCLK_HPM = DOUTCOPY/(HPM_RATIO + 1)	0x0
RSVD	[19]	–	Reserved	0x0
COPY_RATIO	[18:16]	RW	DIVCOPY clock divider ratio DOUTCOPY = MOUTHPM/(COPY_RATIO + 1)	0x0
RSVD	[15:4]	–	Reserved	0x0
CORE_RATIO	[3:0]	RW	DIVAPLL clock divider ratio ARMCLK = MOUTCORE/(CORE_RATIO + 1)	0x0

5.3.1.167 CLKDIV_IEM_L1

- Address = 0x1004_531C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:23]	–	Reserved	0x0
HPM_RATIO	[22:20]	RW	DIVHPM clock divider ratio SCLK_HPM = DOUTCOPY/(HPM_RATIO + 1)	0x0
RSVD	[19]	–	Reserved	0x0
COPY_RATIO	[18:16]	RW	DIVCOPY clock divider ratio DOUTCOPY = MOUTHPM/(COPY_RATIO + 1)	0x0
RSVD	[15:4]	–	Reserved	0x0
CORE_RATIO	[3:0]	RW	DIVAPLL clock divider ratio ARMCLK = MOUTCORE/(CORE_RATIO + 1)	0x0

6 Interrupt Controller

6.1 Overview

This chapter describes the interrupt controller of Exynos4210, which is also called as generic interrupt controller (GIC). The GIC is a centralized resource for supporting and managing interrupts in the system.

6.2 Features

Exynos4210 adopts PrimeCell Generic Interrupt Controller (GIC PL390) as a centralized resource for supporting and managing interrupts in a system. For GIC details, Please refer to the following ARM documents:

- PrimeCell Generic Interrupt Controller (PL390) Technical Reference Manual Revision r0p0
- ARM Generic Interrupt Controller Architecture Specification version 1.0

6.3 Functional Description

6.3.1 Security Extensions Support

Please refer to the GIC PL390 technical reference manual.

6.3.2 Implementation-Specific Configurable Features

During implementation of the GIC, the Exynos4210 GIC configuration is as follows:

- Total 160 interrupts including Software Generated Interrupts (SGIs), Private Peripheral Interrupts (PPIs) and Shared Peripheral Interrupts (SPIs) are supported.
- For SPI, maximal $32 \times 4 = 128$ interrupt requests shall be serviced.
- See the following the configuration table

Table 6-1 GIC Configuration Values

Items	Configuration Values
AMBA Protocol	AXI
Software Generated Interrupts (SGI)	16
Private Peripheral Interrupts (PPI)	8
Shared Peripheral Interrupts (SPI)	128
Priority Level	256
Legacy interrupt Support	No
Number of CPUs	2
CPU Interface AXI ID Width	10
Distributer AXI ID Width	10
Security Domains	2 (Supports TrustZone technology)
Lockable SPIs	31
Legacy dialog	– (Legacy interrupts are not used)
SGI Register Level Selection	0xf (default value)
SPI Register Level Selection	0x3ff (default value)
PPI Register Level Selection	0xf (default value)
PPI sensitivity	ppi_cx[0] – ppi_cx[5]: edge ppi_cx[6] – ppi_cx[10]: level ppi_cx[11]: edge ppi_cx[12]: level ppi_cx[13] – ppi_cx[14]:edge ppi_cx[15]: level
PPI Registering	Synchronized (for all PPI)
SPI Registering	Synchronized (for all SPI)

6.3.3 Terminology

Please refer to the GIC PL390 technical reference manual.

6.3.4 Interrupt Source

6.3.4.1 Interrupt Sources Connection

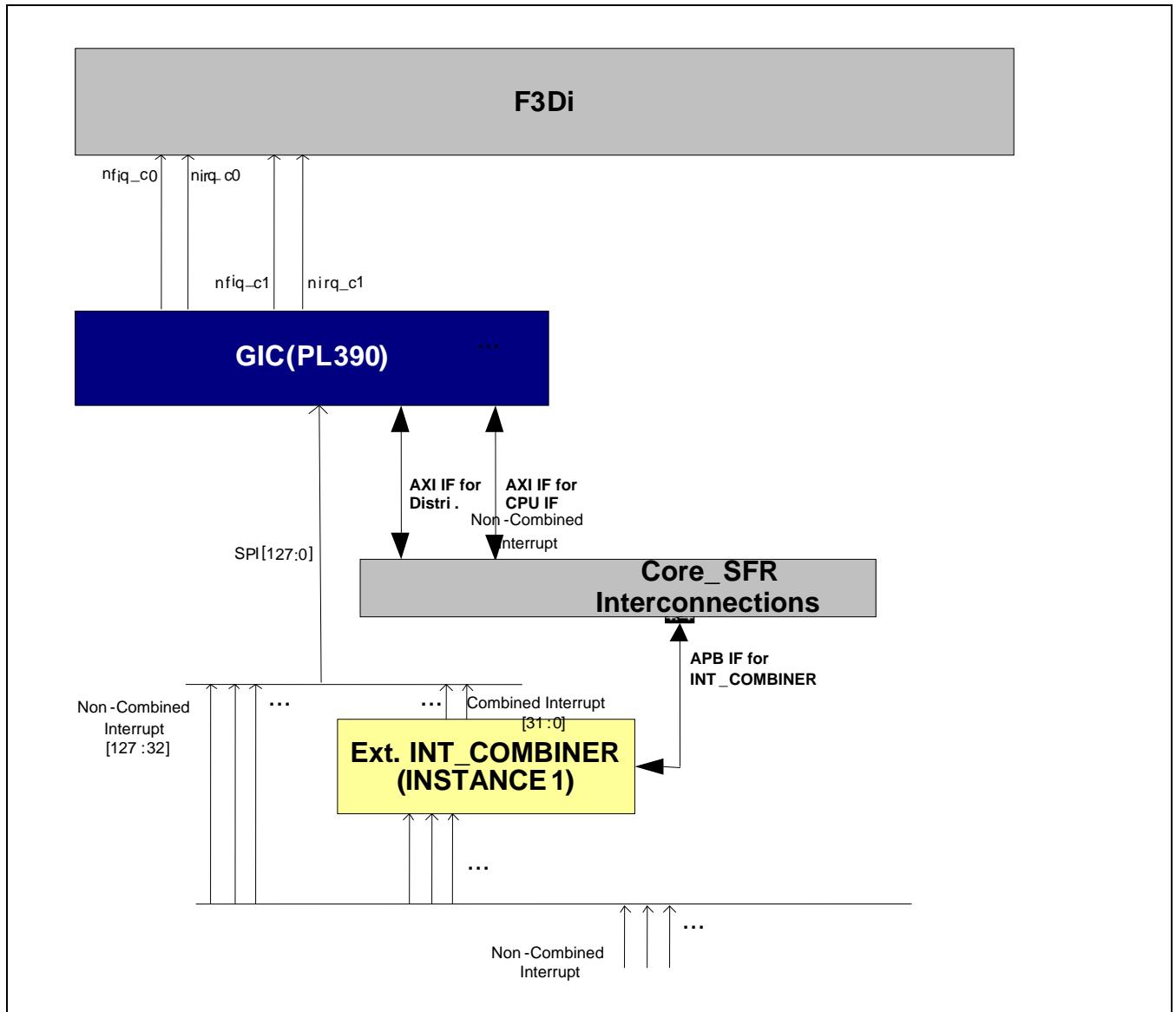


Figure 6-1 Interrupt Sources Connection

6.3.4.2 E-GIC Interrupt Table

Total 160 interrupts including Software Generated Interrupts (SGIs[15:0],ID[15:0]), Private Peripheral Interrupts (PPIs[15:0],ID[31:16]) and Shared Peripheral Interrupts (SPIs[127:0],ID[159:32]) are supported. For SPI, maximal $32 \times 4 = 128$ interrupt requests shall be serviced. See the following table.

Table 6-2 External GIC Interrupt Table (SPI[127:32]: Non-Combined Interrupt)

SPI Port No	ID	Interrupt Source	Source Block
127	159	G3D_IRQGP	G3D
126	158	G3D_IRQPP3	G3D
125	157	G3D_IRQPP2	G3D
124	156	G3D_IRQPP1	G3D
123	155	G3D_IRQPP0	G3D
122	154	G3D_IRQGPMMU	G3D
121	153	G3D_IRQPPMMU3	G3D
120	152	G3D_IRQPPMMU2	G3D
119	151	G3D_IRQPPMMU1	G3D
118	150	G3D_IRQPPMMU0	G3D
117	149	G3D_IRQPMU	G3D
116	148	SATA	
115	147	TSI	
114	146	CEC	
113	145	SLIMBUS	
112	144	INTFEEDCTRL_SSS	
111	143	GPS	
110	142	PMU	ALIVE
109	141	KEYPAD	
108	140	PEN1	
107	139	ADC1	
106	138	PEN0	
105	137	ADC0	
104	136	SPDIF	
103	135	PCM2	
102	134	PCM1	
101	133	PCM0	
100	132	AC97	
99	131	I2S2	
98	130	I2S1	
97	129	I2S0	
96	128	AUDIO_SS	

SPI Port No	ID	Interrupt Source	Source Block
95	127	TVENC	
94	126	MFC	
93	125	HDMI_I2C	
92	124	HDMI	
91	123	MIXER	
90	122	PCIe	
89	121	2D	
88	120	JPEG	
87	119	FIMC3	
86	118	FIMC2	
85	117	FIMC1	
84	116	FIMC0	
83	115	ROTATOR	
82	114	ONENAND_AUDI	
81	113	MIPI_DSI_2LANE	
80	112	MIPI_CSI_2LANE	
79	111	MIPI_DSI_4LANE	
78	110	MIPI_CSI_4LANE	
77	109	SDMMC	SDMMC (synopsys)
76	108	HSMMC3	
75	107	HSMMC2	
74	106	HSMMC1	
73	105	HSMMC0	
72	104	MODEMIF	
71	103	USB_DEVICE	
70	102	USB_HOST	
69	101	MCT_G1	MCT
68	100	SPI2	
67	99	SPI1	
66	98	SPI0	
65	97	I2C7	
64	96	I2C6	
63	95	I2C5	
62	94	I2C4	
61	93	I2C3	
60	92	I2C2	
59	91	I2C1	

SPI Port No	ID	Interrupt Source	Source Block
58	90	I2C0	
57	89	MCT_G0	MCT
56	88	UART4	
55	87	UART3	
54	86	UART2	
53	85	UART1	
52	84	UART0	
51	83	NFC	
50	82	IEM_IEC	
49	81	IEM_APPC	
48	80	MCT_L1	MCT
47	79	GPIO_XA	
46	78	GPIO_XB	
45	77	RTC_TIC	
44	76	RTC_ALARM	
43	75	WDT	
42	74	MCT_L0	MCT
41	73	TIMER4	
40	72	TIMER3	
39	71	TIMER2	
38	70	TIMER1	
37	69	TIMER0	
36	68	PDMA1	
35	67	PDMA0	
34	66	MDMA_LCD0	
33	65	Reserved	
32	64	EXT_INT16 to 31	External Interrupt

Table 6-3 External GIC Interrupt Table (SPI[31:0]: Combined Interrupt)

SPI Port No	ID	INT_E_Combiner	Interrupt Source	Source Block
31	63	INTG31[0](bypass)	EXT_INT15	External Interrupt
30	62	INTG30[0](bypass)	EXT_INT14	External Interrupt
29	61	INTG29[0](bypass)	EXT_INT13	External Interrupt
28	60	INTG28[0](bypass)	EXT_INT12	External Interrupt
27	59	INTG27[0](bypass)	EXT_INT11	External Interrupt
26	58	INTG26[0](bypass)	EXT_INT10	External Interrupt
25	57	INTG25[0](bypass)	EXT_INT9	External Interrupt
24	56	INTG24[0](bypass)	EXT_INT8	External Interrupt
23	55	INTG23[0](bypass)	EXT_INT7	External Interrupt
22	54	INTG22[0](bypass)	EXT_INT6	External Interrupt
21	53	INTG21[0](bypass)	EXT_INT5	External Interrupt
20	52	INTG20[0](bypass)	EXT_INT4	External Interrupt
19	51	INTG19[0](bypass)	EXT_INT3	External Interrupt
18	50	INTG18[0](bypass)	EXT_INT2	External Interrupt
17	49	INTG17[0](bypass)	EXT_INT1	External Interrupt
16	48	INTG16[0](bypass)	EXT_INT0	External Interrupt
15	47	INTG15[7]	DECERRINTR	F3DI
		INTG15[6]	SLVERRINTR	F3DI
		INTG15[5]	ERRDINTR	F3DI
		INTG15[4]	ERRTINTR	F3DI
		INTG15[3]	ERRWDINTR	F3DI
		INTG15[2]	ERRWTINTR	F3DI
		INTG15[1]	ECNTRINTR	F3DI
		INTG15[0]	SCUEVABORT	F3DI
14	46	INTG14[6]	CPU_nIRQOUT[1]	F3DI
13	45	INTG13[5]	CPU_nIRQOUT[0]	F3DI
12	44	INTG12[7]	MCT_G3	MCT
		INTG12[6]	MCT_G2	MCT
		INTG12[5]	MCT_G1	MCT
		INTG12[4]	MCT_G0	MCT
		INTG12[3]	LCD1[3]	
		INTG12[2]	LCD1[2]	
		INTG12[1]	LCD1[1]	
		INTG12[0]	LCD1[0]	
11	43	INTG11[7]	LCD1[3]	
		INTG11[6]	LCD1[2]	

SPI Port No	ID	INT_E_Combiner	Interrupt Source	Source Block
		INTG11[5]	LCD1[1]	
		INTG11[4]	LCD1[0]	
		INTG11[3]	LCD0[3]	
		INTG11[2]	LCD0[2]	
		INTG11[1]	LCD0[1]	
		INTG11[0]	LCD0[0]	
10	42	INTG10[7]	DMC1_PPC_PEREV_M	DMC1
		INTG10[6]	DMC1_PPC_PEREV_A	DMC1
		INTG10[5]	DMC0_PPC_PEREV_M	DMC0
		INTG10[4]	DMC0_PPC_PEREV_A	DMC0
		INTG10[3]	MIU	MIU
		INTG10[2]	L2CCINTR	F3DI
		INTG10[1]	RP_TIMER	
9	41	INTG9[6]	PPMU_MFC_M1	PPMU for MFC_M1
		INTG9[5]	PPMU_MFC_M0	PPMU for MFC_M0
		INTG9[4]	PPMU_3D	PPMU for 3D
		INTG9[3]	PPMU_TV_M0	PPMU for TV_M0
		INTG9[2]	PPMU_FILE_D_M0	PPMU for FILE_D_M0
		INTG9[1]	PPMU_LCD1	PPMU for LCD1
		INTG9[0]	PPMU_LCD0	PPMU for LCD0
8	40	INTG8[7]	PPMU_IMAGE_M0	PPMU for IMAGE_M0
		INTG8[6]	PPMU_CAMIF_M0	PPMU for CAMIF_M0
		INTG8[5]	PPMU_D_right_M0	PPMU for D_right_M0
		INTG8[4]	PPMU_D_lefr_M0	PPMU for D_left_M0
		INTG8[3]	PPMU_ACP0_M0	PPMU for ACP0_M0
		INTG8[2]	PPMU_Core_D_S0	PPMU for Core_D_S0
		INTG8[1]	PPMU_Core_D_M1	PPMU for Core_D_M1
		INTG8[0]	PPMU_Core_D_M0	PPMU for Core_D_M0
7	39	INTG7[7]	SYSMMU_PCIE[1]	System MMU for PCIe
		INTG7[6]	SYSMMU_MFC_M1[1]	System MMU for MFC_M1
		INTG7[5]	SYSMMU_MFC_M0[1]	System MMU for MFC_M0
		INTG7[4]	SYSMMU_TV_M0[1]	System MMU for TV_M0
		INTG7[3]	SYSMMU_LCD1_M1[1]	System MMU for LCD1_M1
		INTG7[2]	SYSMMU_LCD0_M0[1]	System MMU for LCD0_M0
		INTG7[1]	SYSMMU_MDMA1[1]	System MMU for MDMA1 (LCD0)
		INTG7[0]	SYSMMU_Rotator[1]	System MMU for Rotator
6	38	INTG6[7]	SYSMMU_2D[1]	System MMU for 2D

SPI Port No	ID	INT_E_Combiner	Interrupt Source	Source Block
		INTG6[6]	SYSMMU_JPEG[1]	System MMU for JPEG
		INTG6[5]	SYSMMU_FIMC3[1]	System MMU for FIMC3
		INTG6[4]	SYSMMU_FIMC2[1]	System MMU for FIMC2
		INTG6[3]	SYSMMU_FIMC1[1]	System MMU for FIMC1
		INTG6[2]	SYSMMU_FIMC0[1]	System MMU for FIMC0
		INTG6[1]	SYSMMU_SSS[1]	System MMU for SSS
		INTG6[0]	SYSMMU_MDMA0[1]	System MMU for MDMA0 (CORE)
5	37	INTG5[7]	SYSMMU_PCIE[0]	System MMU for PCIe
		INTG5[6]	SYSMMU_MFC_M1[0]	System MMU for MFC_M1
		INTG5[5]	SYSMMU_MFC_M0[0]	System MMU for MFC_M0
		INTG5[4]	SYSMMU_TV_M0[0]	System MMU for TV_M0
		INTG5[3]	SYSMMU_LCD1_M1[0]	System MMU for LCD1_M1
		INTG5[2]	SYSMMU_LCD0_M0[0]	System MMU for LCD0_M0
		INTG5[1]	SYSMMU_MDMA1[0]	System MMU for MDMA1 (LCD0)
		INTG5[0]	SYSMMU_Rotator[0]	System MMU for Rotator
4	36	INTG4[7]	SYSMMU_2D[0]	System MMU for 2D
		INTG4[6]	SYSMMU_JPEG[0]	System MMU for JPEG
		INTG4[5]	SYSMMU_FIMC3[0]	System MMU for FIMC3
		INTG4[4]	SYSMMU_FIMC2[0]	System MMU for FIMC2
		INTG4[3]	SYSMMU_FIMC1[0]	System MMU for FIMC1
		INTG4[2]	SYSMMU_FIMC0[0]	System MMU for FIMC0
		INTG4[1]	SYSMMU_SSS[0]	System MMU for SSS
		INTG4[0]	SYSMMU_MDMA0[0]	System MMU for MDMA0 (CORE)
3	35	INTG3[4]	TMU	TMU
		INTG3[3]	nCTIIRQ[1]	F3Di CTI interrupt for CPU1
		INTG3[2]	PMUIRQ[1]	F3Di PMU interrupt from CPU1
		INTG3[1]	PARITYFAILSCU[1]	Parity fail for SCU from CPU1
		INTG3[0]	PARITYFAIL1	L1 parity fail for CPU1
2	34	INTG2[4]	TMU	TMU
		INTG2[3]	nCTIIRQ[0]	F3Di CTI interrupt for CPU0
		INTG2[2]	PMUIRQ[0]	F3Di PMU interrupt from CPU0
		INTG2[1]	PARITYFAILSCU[0]	Parity fail for SCU from CPU0
		INTG2[0]	PARITYFAIL0	L1 parity fail for CPU0
1	33	Reserved		
0	32	Reserved		

Table 6-4 External GIC Interrupt Table (PPI[15:0])

PPI Port No	ID	Interrupt Source	Source Block
15	31	—	
14	30	—	
13	29	—	
12	28	—	
11	27	—	
10	26	—	
9	25	—	
8	24	—	
7	23	—	
6	22	—	
5	21	—	
4	20	—	
3	19	—	
2	18	—	
1	17	—	
0	16	—	

6.3.5 Functional Overview of Generic Interrupt Controller

Please refer to the GIC PL390 technical reference manual.

6.3.6 Interrupt Handling and Prioritization

Please refer to the GIC PL390 technical reference manual.

6.4 Register Description

6.4.1 Register Map Summary

- Base Address: 0x1048_0000 (CPU Interface)

Register	Offset	Description	Reset Value
ICCICR_CPU0	0x0000	CPU Interface Control Register	0x0000_0000
ICCPMR_CPU0	0x0004	Interrupt Priority Mask Register	0x0000_0000
ICCBPR_CPU0	0x0008	Binary Point Register	0x0000_0000
ICCIAR_CPU0	0x000C	Interrupt Acknowledge Register	0x0000_03FF
ICCEOIR_CPU0	0x0010	End of Interrupt Register	Undefined
ICCRPR_CPU0	0x0014	Running Priority Register	0x0000_00FF
ICCHPIR_CPU0	0x0018	Highest Pending Interrupt Register	0x0000_03FF
ICCABPR_CPU0	0x001C	Aliased Binary Point Register	0x0000_0000
INTEG_EN_C_CPU0	0x0040	Integration Test Enable Register	0x0000_0000
INTERRUPT_OUT_CPU0	0x0044	Interrupt Output Register	0x0000_0000
ICCIIDR	0x00FC	CPU Interface Identification Register	0x3901_043B
ICCICR_CPU1	0x8000	CPU Interface Control Register	0x0000_0000
ICCPMR_CPU1	0x8004	Interrupt Priority Mask Register	0x0000_0000
ICCBPR_CPU1	0x8008	Binary Point Register	0x0000_0000
ICCIAR_CPU1	0x800C	Interrupt Acknowledge Register	0x0000_03FF
ICCEOIR_CPU1	0x8010	End of Interrupt Register	Undefined
ICCRPR_CPU1	0x8014	Running Priority Register	0x0000_00FF
ICCHPIR_CPU1	0x8018	Highest Pending Interrupt Register	0x0000_03FF
ICCABPR_CPU1	0x801C	Aliased Binary Point Register	0x0000_0000
INTEG_C_EN_CPU1	0x8040	Integration Test Enable Register	0x0000_0000
INTERRUPT_OUT_CPU1	0x8044	Interrupt Output Register	0x0000_0000

- Base Address: 0x1049_0000 (Distributor)

Register	Offset	Description	Reset Value
ICDDCR	0x0000	Distributor Control Register	0x0000_0000
ICDICTR	0x0004	Interrupt Controller Type Register	0x0000_FC24
ICDIIDR	0x0008	Distributor Implementer Identification Register	0x0000_043B
ICDISR0_CPU0	0x0080	Interrupt Security Registers (SGI,PPI)	0x0000_0000
ICDISR1	0x0084	Interrupt Security Registers (SPI[31:0])	0x0000_0000
ICDISR2	0x0088	Interrupt Security Registers (SPI[63:32])	0x0000_0000
ICDISR3	0x008C	Interrupt Security Registers (SPI[95:64])	0x0000_0000
ICDISR4	0x0090	Interrupt Security Registers (SPI[127:96])	0x0000_0000
ICDISER0_CPU0	0x0100	Interrupt Set-Enable Register (SGI,PPI)	0x0000_FFFF
ICDISER1	0x0104	Interrupt Set-Enable Register (SPI[31:0])	0x0000_0000
ICDISER2	0x0108	Interrupt Set-Enable Register (SPI[63:32])	0x0000_0000
ICDISER3	0x010C	Interrupt Set-Enable Register (SPI[95:64])	0x0000_0000
ICDISER4	0x0110	Interrupt Set-Enable Register (SPI[127:96])	0x0000_0000
ICDICER0_CPU0	0x0180	Interrupt Clear-Enable Register (SGI,PPI)	0x0000_FFFF
ICDICER1	0x0184	Interrupt Clear-Enable Register (SPI[31:0])	0x0000_0000
ICDICER2	0x0188	Interrupt Clear-Enable Register (SPI[63:32])	0x0000_0000
ICDICER3	0x018C	Interrupt Clear-Enable Register (SPI[95:64])	0x0000_0000
ICDICER4	0x0190	Interrupt Clear-Enable Register (SPI[127:96])	0x0000_0000
ICDISPR0_CPU0	0x0200	Interrupt Pending-Set Register (SGI,PPI)	0x0000_0000
ICDISPR1	0x0204	Interrupt Pending-Set Register (SPI[31:0])	0x0000_0000
ICDISPR2	0x0208	Interrupt Pending-Set Register (SPI[63:32])	0x0000_0000
ICDISPR3	0x020C	Interrupt Pending-Set Register (SPI[95:64])	0x0000_0000
ICDISPR4	0x0210	Interrupt Pending-Set Register (SPI[127:96])	0x0000_0000
ICDICPR0_CPU0	0x0280	Interrupt Pending-Clear Register (SGI,PPI)	0x0000_0000
ICDICPR1	0x0284	Interrupt Pending-Clear Register (SPI[31:0])	0x0000_0000
ICDICPR2	0x0288	Interrupt Pending-Clear Register (SPI[63:32])	0x0000_0000
ICDICPR3	0x028C	Interrupt Pending-Clear Register (SPI[95:64])	0x0000_0000
ICDICPR4	0x0290	Interrupt Pending-Clear Register(SPI[127:96])	0x0000_0000
ICDABR0_CPU0	0x0300	Active Bit Register (SGI, PPI)	0x0000_0000
ICDABR1	0x0304	Active Bit Register (SPI[31:0])	0x0000_0000
ICDABR2	0x0308	Active Bit Register (SPI[63:32])	0x0000_0000
ICDABR3	0x030C	Active Bit Register (SPI[95:64])	0x0000_0000
ICDABR4	0x0310	Active Bit Register (SPI[127:96])	0x0000_0000
ICDIPR0_CPU0	0x0400	Priority Level Register (SGI[3:0])	0x0000_0000
ICDIPR1_CPU0	0x0404	Priority Level Register (SGI[7:4])	0x0000_0000
ICDIPR2_CPU0	0x0408	Priority Level Register (SGI[11:8])	0x0000_0000

Register	Offset	Description	Reset Value
ICDIPR3_CPU0	0x040C	Priority Level Register (SGI[15:12])	0x0000_0000
ICDIPR4_CPU0	0x0410	Priority Level Register (PPI[3:0])	0x0000_0000
ICDIPR5_CPU0	0x0414	Priority Level Register (PPI[7:4])	0x0000_0000
ICDIPR6_CPU0	0x0418	Priority Level Register (PPI[11:8])	0x0000_0000
ICDIPR7_CPU0	0x041C	Priority Level Register (PPI[15:12])	0x0000_0000
ICDIPR8	0x0420	Priority Level Register (SPI[3:0])	0x0000_0000
ICDIPR9	0x0424	Priority Level Register (SPI[7:4])	0x0000_0000
ICDIPR10	0x0428	Priority Level Register (SPI[11:8])	0x0000_0000
ICDIPR11	0x042C	Priority Level Register (SPI[15:12])	0x0000_0000
ICDIPR12	0x0430	Priority Level Register (SPI[19:16])	0x0000_0000
ICDIPR13	0x0434	Priority Level Register (SPI[23:20])	0x0000_0000
ICDIPR14	0x0438	Priority Level Register (SPI[27:24])	0x0000_0000
ICDIPR15	0x043C	Priority Level Register (SPI[31:28])	0x0000_0000
ICDIPR16	0x0440	Priority Level Register (SPI[35:32])	0x0000_0000
ICDIPR17	0x0444	Priority Level Register (SPI[39:36])	0x0000_0000
ICDIPR18	0x0448	Priority Level Register (SPI[43:40])	0x0000_0000
ICDIPR19	0x044C	Priority Level Register (SPI[47:44])	0x0000_0000
ICDIPR20	0x0450	Priority Level Register (SPI[51:48])	0x0000_0000
ICDIPR21	0x0454	Priority Level Register (SPI[55:52])	0x0000_0000
ICDIPR22	0x0458	Priority Level Register (SPI[59:56])	0x0000_0000
ICDIPR23	0x045C	Priority Level Register (SPI[63:60])	0x0000_0000
ICDIPR24	0x0460	Priority Level Register (SPI[67:64])	0x0000_0000
ICDIPR25	0x0464	Priority Level Register (SPI[71:68])	0x0000_0000
ICDIPR26	0x0468	Priority Level Register (SPI[75:72])	0x0000_0000
ICDIPR27	0x046C	Priority Level Register (SPI[79:76])	0x0000_0000
ICDIPR28	0x0470	Priority Level Register (SPI[83:80])	0x0000_0000
ICDIPR29	0x0474	Priority Level Register (SPI[87:84])	0x0000_0000
ICDIPR30	0x0478	Priority Level Register (SPI[91:98])	0x0000_0000
ICDIPR31	0x047C	Priority Level Register (SPI[95:92])	0x0000_0000
ICDIPR32	0x0480	Priority Level Register (SPI[99:96])	0x0000_0000
ICDIPR33	0x0484	Priority Level Register (SPI[103:100])	0x0000_0000
ICDIPR34	0x0488	Priority Level Register (SPI[107:104])	0x0000_0000
ICDIPR35	0x048C	Priority Level Register (SPI[111:108])	0x0000_0000
ICDIPR36	0x0490	Priority Level Register (SPI[115:112])	0x0000_0000
ICDIPR37	0x0494	Priority Level Register (SPI[119:116])	0x0000_0000
ICDIPR38	0x0498	Priority Level Register (SPI[123:120])	0x0000_0000
ICDIPR39	0x049C	Priority Level Register (SPI[127:124])	0x0000_0000

Register	Offset	Description	Reset Value
ICDIPTR0_CPU0	0x0800	Processor Targets Register (SGI[3:0])	0x0101_0101
ICDIPTR1_CPU0	0x0804	Processor Targets Register (SGI[7:4])	0x0101_0101
ICDIPTR2_CPU0	0x0808	Processor Targets Register (SGI[11:8])	0x0101_0101
ICDIPTR3_CPU0	0x080C	Processor Targets Register (SGI[15:12])	0x0101_0101
ICDIPTR4_CPU0	0x0810	Processor Targets Register (PPI[3:0])	0x0101_0101
ICDIPTR5_CPU0	0x0814	Processor Targets Register (PPI[7:4])	0x0101_0101
ICDIPTR6_CPU0	0x0818	Processor Targets Register (PPI[11:8])	0x0101_0101
ICDIPTR7_CPU0	0x081C	Processor Targets Register (PPI[15:12])	0x0101_0101
ICDIPTR8	0x0820	Processor Targets Register (SPI[3:0])	0x0000_0000
ICDIPTR9	0x0824	Processor Targets Register (SPI[7:4])	0x0000_0000
ICDIPTR10	0x0828	Processor Targets Register (SPI[11:8])	0x0000_0000
ICDIPTR11	0x082C	Processor Targets Register (SPI[15:12])	0x0000_0000
ICDIPTR12	0x0830	Processor Targets Register (SPI[19:16])	0x0000_0000
ICDIPTR13	0x0834	Processor Targets Register (SPI[23:20])	0x0000_0000
ICDIPTR14	0x0838	Processor Targets Register (SPI[27:24])	0x0000_0000
ICDIPTR15	0x083C	Processor Targets Register (SPI[31:28])	0x0000_0000
ICDIPTR16	0x0840	Processor Targets Register (SPI[35:32])	0x0000_0000
ICDIPTR17	0x0844	Processor Targets Register (SPI[39:36])	0x0000_0000
ICDIPTR18	0x0848	Processor Targets Register (SPI[43:40])	0x0000_0000
ICDIPTR19	0x084C	Processor Targets Register (SPI[47:44])	0x0000_0000
ICDIPTR20	0x0850	Processor Targets Register (SPI[51:48])	0x0000_0000
ICDIPTR21	0x0854	Processor Targets Register (SPI[55:52])	0x0000_0000
ICDIPTR22	0x0858	Processor Targets Register (SPI[59:56])	0x0000_0000
ICDIPTR23	0x085C	Processor Targets Register (SPI[63:60])	0x0000_0000
ICDIPTR24	0x0860	Processor Targets Register (SPI[67:64])	0x0000_0000
ICDIPTR25	0x0864	Processor Targets Register (SPI[71:68])	0x0000_0000
ICDIPTR26	0x0868	Processor Targets Register (SPI[75:72])	0x0000_0000
ICDIPTR27	0x086C	Processor Targets Register (SPI[79:76])	0x0000_0000
ICDIPTR28	0x0870	Processor Targets Register (SPI[83:80])	0x0000_0000
ICDIPTR29	0x0874	Processor Targets Register (SPI[87:84])	0x0000_0000
ICDIPTR30	0x0878	Processor Targets Register (SPI[91:98])	0x0000_0000
ICDIPTR31	0x087C	Processor Targets Register (SPI[95:92])	0x0000_0000
ICDIPTR32	0x0880	Processor Targets Register (SPI[99:96])	0x0000_0000
ICDIPTR33	0x0884	Processor Targets Register (SPI[103:100])	0x0000_0000
ICDIPTR34	0x0888	Processor Targets Register (SPI[107:104])	0x0000_0000
ICDIPTR35	0x088C	Processor Targets Register (SPI[111:108])	0x0000_0000
ICDIPTR36	0x0890	Processor Targets Register (SPI[115:112])	0x0000_0000

Register	Offset	Description	Reset Value
ICDIPTR37	0x0894	Processor Targets Register (SPI[119:116])	0x0000_0000
ICDIPTR38	0x0898	Processor Targets Register (SPI[123:120])	0x0000_0000
ICDIPTR39	0x089C	Processor Targets Register (SPI[127:124])	0x0000_0000
ICDICFR0_CPU0	0x0C00	Interrupt Configuration Register(SGI[15:0])	0xAAAA_AAAA
ICDICFR1_CPU0	0x0C04	Interrupt Configuration Register(PPI[15:0])	0x7DD5_5FFF
ICDICFR2	0x0C08	Interrupt Configuration Register(SPI[15:0])	0x5555_5555
ICDICFR3	0x0C0C	Interrupt Configuration Register(SPI[31:16])	0x5555_5555
ICDICFR4	0x0C10	Interrupt Configuration Register(SPI[47:32])	0x5555_5555
ICDICFR5	0x0C14	Interrupt Configuration Register(SPI[63:48])	0x5555_5555
ICDICFR6	0x0C18	Interrupt Configuration Register(SPI[79:64])	0x5555_5555
ICDICFR7	0x0C1C	Interrupt Configuration Register(SPI[95:80])	0x5555_5555
ICDICFR8	0x0C20	Interrupt Configuration Register(SPI[111:95])	0x5555_5555
ICDICFR9	0x0C24	Interrupt Configuration Register(SPI[127:112])	0x5555_5555
PPI_STATUS_CPU0	0x0D00	PPI Status Register	0x0000_0000
SPI_STATUS0	0x0D04	SPI[31:0] Status Register	0x0000_0000
SPI_STATUS1	0x0D08	SPI[63:32] Status Register	0x0000_0000
SPI_STATUS2	0x0D0C	SPI[95:64] Status Register	0x0000_0000
SPI_STATUS3	0x0D10	SPI[127:96] Status Register	0x0000_0000
ICDSGIR	0x0F00	Software Generated Interrupt Register	Undefined
ICDISR0_CPU1	0x8080	Interrupt Security Registers (SGI,PPI)	0x0000_0000
ICDISER0_CPU1	0x8100	Interrupt Set-Enable Register (SGI,PPI)	0x0000_FFFF
ICDICER0_CPU1	0x8180	Interrupt Clear-Enable Register (SGI,PPI)	0x0000_FFFF
ICDISPR0_CPU1	0x8200	Interrupt Pending-Set Register (SGI,PPI)	0x0000_0000
ICDICPR0_CPU1	0x8280	Interrupt Pending-Clear Register (SGI,PPI)	0x0000_0000
ICDABR0_CPU1	0x8300	Active Status Register (SGI, PPI)	0x0000_0000
ICDIPR0_CPU1	0x8400	Priority Level Register (SGI[3:0])	0x0000_0000
ICDIPR1_CPU1	0x8404	Priority Level Register (SGI[7:4])	0x0000_0000
ICDIPR2_CPU1	0x8408	Priority Level Register (SGI[11:8])	0x0000_0000
ICDIPR3_CPU1	0x840C	Priority Level Register (SGI[15:12])	0x0000_0000
ICDIPR4_CPU1	0x8410	Priority Level Register (PPI[3:0])	0x0000_0000
ICDIPR5_CPU1	0x8414	Priority Level Register (PPI[7:4])	0x0000_0000
ICDIPR6_CPU1	0x8418	Priority Level Register (PPI[11:8])	0x0000_0000
ICDIPR7_CPU1	0x841C	Priority Level Register (PPI[15:12])	0x0000_0000
ICDIPTR0_CPU1	0x8800	Processor Targets Register (SGI[3:0])	0x0202_0202
ICDIPTR1_CPU1	0x8804	Processor Targets Register (SGI[7:4])	0x0202_0202
ICDIPTR2_CPU1	0x8808	Processor Targets Register (SGI[11:8])	0x0202_0202
ICDIPTR3_CPU1	0x880C	Processor Targets Register (SGI[15:12])	0x0202_0202

Register	Offset	Description	Reset Value
ICDIPTR4_CPU1	0x8810	Processor Targets Register (PPI[3:0])	0x0202_0202
ICDIPTR5_CPU1	0x8814	Processor Targets Register (PPI[7:4])	0x0202_0202
ICDIPTR6_CPU1	0x8818	Processor Targets Register (PPI[11:8])	0x0202_0202
ICDIPTR7_CPU1	0x881C	Processor Targets Register (PPI[15:12])	0x0202_0202
ICDICFR0_CPU1	0x8C00	Interrupt Configuration Register(SGI[15:0])	0xAAAA_AAAA
ICDICFR1_CPU1	0x8C04	Interrupt Configuration Register(PPI[15:0])	0x7DD5_5FFF
PPI_STATUS_CPU1	0x8D00	PPI Status Register	0x0000_0000

NOTE: Please refer to the GIC Architecture Specification document for detailed register descriptions.

7 Interrupt Combiner

7.1 Overview

The interrupt controller in Exynos4210 consists of a generic interrupt controller (GIC) and an interrupt combiner. The interrupt combiner combines several interrupt sources as a group. The several interrupt requests in a group makes a single request signal. The interrupt input of GIC consists of the group interrupt requests from the interrupt combiner and uncombined interrupt sources.

7.2 Features

- 109 interrupt source inputs for the external interrupt combiner.
- 32 group interrupt outputs for the external interrupt combiner.
- Enable/Mask of each interrupt source in a group.
- Status of each interrupt source in a group before interrupt masking.
- Status of each interrupt source in a group after interrupt masking.
- Status of each group interrupt output after interrupt masking and combining.

7.3 Functional Description

7.3.1 Block Diagram

The external interrupt combiner combines some interrupts source into 32 group interrupt request outputs, which are connected to the inputs of the GIC unit outside the F3Di unit.

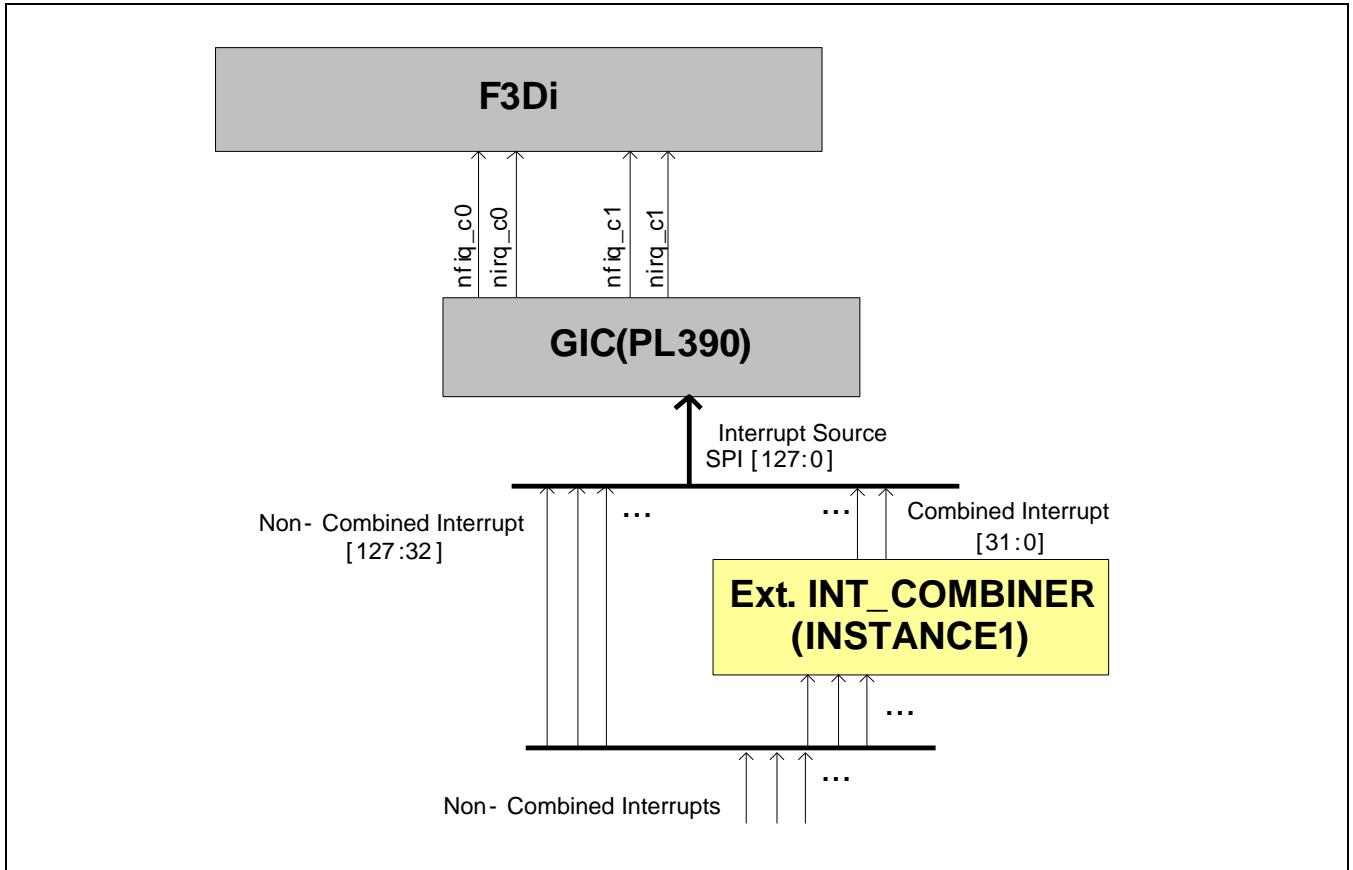


Figure 7-1 Block Diagram of External Interrupt Combiner

7.3.2 Interrupt Sources

Table 7-1 Interrupt Groups for E-GIC

Combiner Group ID	Combined Interrupt Source Name	Bit	Interrupt Source	Source Block
INTG0	Reserved	[7]	Reserved	Reserved
		[6]	Reserved	
		[5]	Reserved	
		[4]	Reserved	
		[3]	Reserved	Reserved
		[2]	Reserved	
		[1]	Reserved	
		[0]	Reserved	
INTG1	GLOBAL_TIMER	[7]	G3_IRQ	Multi-Core Timer
		[6]	G2_IRQ	
		[5]	G1_IRQ	
		[4]	G0_IRQ	
		[3]	Reserved	Reserved
		[2]	Reserved	
		[1]	Reserved	
		[0]	Reserved	
INTG2	TMU/ F3DiCTI0/ PARITYFAIL0/ PMU0	[4]	TMU	TMU
		[3]	nCTIIRQ[0]	CPU0
		[2]	PMUIRQ[0]	
		[1]	PARITYFAILSCU[0]	
		[0]	PARITYFAIL0	
INTG3	TMU/F3DiCTI1/ PARITYFAIL1/ PMU1	[4]	TMU	TMU
		[3]	nCTIIRQ[1]	CPU1
		[2]	PMUIRQ[1]	
		[1]	PARITYFAILSC[1]	
		[0]	PARITYFAIL1	
INTG4	SYSMMU[7:0]	[7]	SYSMMU_2D[0]	System MMU
		[6]	SYSMMU_JPEG[0]	
		[5]	SYSMMU_FIMC3[0]	
		[4]	SYSMMU_FIMC2[0]	
		[3]	SYSMMU_FIMC1[0]	
		[2]	SYSMMU_FIMC0[0]	
		[1]	SYSMMU_SSS[0]	
		[0]	SYSMMU_MDMA0[0]	

Combiner Group ID	Combined Interrupt Source Name	Bit	Interrupt Source	Source Block
INTG5	SYSMMU[15:8]	[7]	SYSMMU_PCIE[0]	System MMU
		[6]	SYSMMU_MFC_M1[0]	
		[5]	SYSMMU_MFC_M0[0]	
		[4]	SYSMMU_TV_M0[0]	
		[3]	SYSMMU_LCD1_M1[0]	
		[2]	SYSMMU_LCD0_M0[0]	
		[1]	SYSMMU_MDMA1[0]	
		[0]	SYSMMU_ROTATOR[0]	
INTG6	SYSMMU[23:16]	[7]	SYSMMU_2D[1]	System MMU
		[6]	SYSMMU_JPEG[1]	
		[5]	SYSMMU_FIMC3[1]	
		[4]	SYSMMU_FIMC2[1]	
		[3]	SYSMMU_FIMC1[1]	
		[2]	SYSMMU_FIMC0[1]	
		[1]	SYSMMU_SSS[1]	
		[0]	SYSMMU_MDMA0[1]	
INTG7	SYSMMU[31:24]	[7]	SYSMMU_PCIE[1]	System MMU
		[6]	SYSMMU_MFC_M1[1]	
		[5]	SYSMMU_MFC_M0[1]	
		[4]	SYSMMU_TV_M0[1]	
		[3]	SYSMMU_LCD1_M1[1]	
		[2]	SYSMMU_LCD0_M0[1]	
		[1]	SYSMMU_MDMA1[1]	
		[0]	SYSMMU_ROTATOR[1]	
INTG8	PPMU [7:0]	[7]	PPMU_IMAGE_M0	PPMU
		[6]	PPMU_CAMIF_M0	
		[5]	PPMU_D_RIGHT_M0	
		[4]	PPMU_D_LEFT_M0	
		[3]	PPMU_ACP0_M0	
		[2]	PPMU_CORE_D_S0	
		[1]	PPMU_CORE_D_M1	
		[0]	PPMU_CORE_D_M0	
INTG9	PPMU [14:8]	[6]	PPMU_MFC_M1	PPMU
		[5]	PPMU_MFC_M0	
		[4]	PPMU_3D	
		[3]	PPMU_TV_M0	

Combiner Group ID	Combined Interrupt Source Name	Bit	Interrupt Source	Source Block
		[2]	PPMU_FILE_D_M0	
		[1]	PPMU_LCD1	
		[0]	PPMU_LCD0	
INTG10	DMC1/ DMC0/ MIU/ L2CACHE/ RP_TIMER/ GPIO_AUDIO	[7]	DMC1_PPC_PEREV_M	DMC1
		[6]	DMC1_PPC_PEREV_A	
		[5]	DMC0_PPC_PEREV_M	DMC0
		[4]	DMC0_PPC_PEREV_A	
		[3]	MIU	MIU
		[2]	L2CACHE	L2 Cache
		[1]	RP_TIMER	RP
		[0]	GPIO_AUDIO	GPIO_AUDIO
INTG11	LCD1/0	[7]	LCD1[3]	LCD1
		[6]	LCD1[2]	
		[5]	LCD1[1]	
		[4]	LCD1[0]	
		[3]	LCD0[3]	LCD0
		[2]	LCD0[2]	
		[1]	LCD0[1]	
		[0]	LCD0[0]	
INTG12	GLOBAL_TIMER/ LCD1	[7]	G3_IRQ	Multi-Core Timer
		[6]	G2_IRQ	
		[5]	G1_IRQ	
		[4]	G0_IRQ	
		[3]	LCD1[3]	LCD1
		[2]	LCD1[2]	
		[1]	LCD1[1]	
		[0]	LCD1[0]	
INTG13	G3D_MMU	[5]	Reserved (CPU_nIRQOUT[0])	CPU/ G3D
		[4]	Reserved (G3D_GPMMU)	
		[3]	Reserved (G3D_PPMMU3)	
		[2]	Reserved (G3D_PPMMU2)	
		[1]	Reserved (G3D_PPMMU1)	
		[0]	Reserved (G3D_PPMMU0)	
INTG14	G3D_PMU/ PP/ GP	[6]	Reserved (CPU_nIRQOUT[1])	CPU/ G3D
		[5]	Reserved (G3D_PMU)	
		[4]	Reserved (G3D_GP)	

Combiner Group ID	Combined Interrupt Source Name	Bit	Interrupt Source	Source Block
		[3]	Reserved (G3D_PP3)	
		[2]	Reserved (G3D_PP2)	
		[1]	Reserved (G3D_PP1)	
		[0]	Reserved (G3D_PP0)	
INTG15	BUS_ERROR/ SCUEVABORT	[7]	DECERRINTR	
		[6]	SLVERRINTR	
		[5]	ERRRDINTR	
		[4]	ERRRTINTR	
		[3]	ERRWDINTR	
		[2]	ERRWTINTR	
		[1]	ECNTRINTR	
		[0]	SCUEVABORT	
INTG16 – NTG31	Reserved	[0]	Reserved	Reserved

7.3.3 Interrupt Combiner Operation

An interrupt source in an interrupt group is controlled by an interrupt enable bit. The interrupt enable bits are controlled by IESRn (IIESRn) registers and IECRn (IIECRn) registers. IESRn (IIESRn) register can toggle an interrupt bit to "1". If you write '1' to a bit position on IESRn, the corresponding bit on the interrupt enable bits are set to "1". On the other hand, IECRn (IIECRn) register can toggle an interrupt enable bit to "0". If you write "1" to a bit position on IECRn, the corresponding bit on the interrupt enable bits is cleared to "0". This feature will make it easy to address resource sharing issues in a multi-processor system.

There are several interrupt sources in an interrupt group. If an interrupt enable bit is "0", the corresponding interrupt is masked. All the interrupt sources in an interrupt group, which includes masked interrupt sources, are ORed to form a combined interrupt request signal. This combined group interrupt request output signal is connected to an input of a GIC.

Each interrupt source status before it is masked by an interrupt enable bit can be shown by reading ISTRn (IISTRn) register. The combined group interrupt request output signal can be shown by reading CIPSR0 (IICIPSRn) register.

7.4 Register Description

7.4.1 Register Map Summary

- Base Address: 0x1044_0000

Register	Offset	Description	Reset Value
For E-GIC			
IESR0	0x0000	Interrupt enable set register for group 0 to 3	0x00000000
IECR0	0x0004	Interrupt enable clear register for group 0 to 3	0x00000000
ISTR0	0x0008	Interrupt status register for group 0 to 3	Undefined
IMSR0	0x000C	Interrupt masked status register for group 0 to 3	Undefined
IESR1	0x0010	Interrupt enable set register for group 4 to 7	0x00000000
IECR1	0x0014	Interrupt enable clear register for group 4 to 7	0x00000000
ISTR1	0x0018	Interrupt status register for group 4 to 7	Undefined
IMSR1	0x001C	Interrupt masked status register for group 4 to 7	Undefined
IESR2	0x0020	Interrupt enable set register for group 8 to 11	0x00000000
IECR2	0x0024	Interrupt enable clear register for group 8 to 11	0x00000000
ISTR2	0x0028	Interrupt status register for group 8 to 11	Undefined
IMSR2	0x002C	Interrupt masked status register for group 8 to 11	Undefined
IESR3	0x0030	Interrupt enable set register for group 12 to 15	0x00000000
IECR3	0x0034	Interrupt enable clear register for group 12 to 15	0x00000000
ISTR3	0x0038	Interrupt masked status register for group 12 to 15	Undefined
IMSR3	0x003C	Interrupt status register for group 12 to 15	Undefined
CIPSR0	0x0100	Combined interrupt pending status0	Undefined

7.4.1.1 IESR0

- Address = 0x1044_0000, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:29]	-	Reserved	0x0
TMU	[28]	RW	Set the corresponding interrupt enable bit to "1". If the interrupt enable bit is set, the interrupt request will be served.	0
nCTIIRQ[1]	[27]	RW		0
PMUIRQ[1]	[26]	RW		0
PARITYFAILSCU[1]	[25]	RW	Write) 0 = The current setting is unchanged. 1 = Set the interrupt enable bit to "1".	0
PARITYFAIL1	[24]	RW	Read) The current interrupt enable bit. 0 = Masked. 1 = Enabled.	0
RSVD	[23:21]	-	Reserved	0x0
TMU	[20]	RW	Set the corresponding interrupt enable bit to "1". If the interrupt enable bit is set, the interrupt request will be served.	0
nCTIIRQ[0]	[19]	RW		0
PMUIRQ[0]	[18]	RW		0
PARITYFAILSCU[0]	[17]	RW	Write) 0 = The current setting is unchanged. 1 = Set the interrupt enable bit to "1".	0
PARITYFAIL0	[16]	RW	Read) The current interrupt enable bit. 0 = Masked. 1 = Enabled.	0
G3_IRQ	[15]	RW	Set the corresponding interrupt enable bit to "1". If the interrupt enable bit is set, the interrupt request will be served.	0
G2_IRQ	[14]	RW		0
G1_IRQ	[13]	RW		0
G0_IRQ	[12]	RW	Write) 0 = The current setting is unchanged. 1 = Set the interrupt enable bit to "1".	0
RSVD	[11]	RW		0
RSVD	[10]	RW	Read) The current interrupt enable bit.	0
RSVD	[9]	RW	0 = Masked. 1 = Enabled.	0
RSVD	[8]	RW		0
RSVD	[7]	RW	Set the corresponding interrupt enable bit to "1". If the interrupt enable bit is set, the interrupt request will be served.	0
RSVD	[6]	RW		0
RSVD	[5]	RW		0
RSVD	[4]	RW	Write) 0 = The current setting is unchanged. 1 = Set the interrupt enable bit to "1".	0
RSVD	[3]	RW		0
RSVD	[2]	RW	Read) The current interrupt enable bit.	0
RSVD	[1]	RW	0 = Masked. 1 = Enabled.	0
RSVD	[0]	RW		0

7.4.1.2 IECR0

- Address = 0x1044_0004, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:29]	-	Reserved	0x0
TMU	[28]	RW	Clear the corresponding interrupt enable bit to "0". If the interrupt enable bit is cleared, the interrupt will be masked.	0
nCTIIRQ[1]	[27]	RW		0
PMUIRQ[1]	[26]	RW		0
PARITYFAILSCU[1]	[25]	RW	Write) 0 = The current setting is unchanged. 1 = Clear the interrupt enable bit to "0".	0
PARITYFAIL1	[24]	RW	Read) The current interrupt enable bit. 0 = Masked. 1 = Enabled.	0
RSVD	[23:21]	-	Reserved	0x0
TMU	[20]	RW	Clear the corresponding interrupt enable bit to "0". If the interrupt enable bit is cleared, the interrupt will be masked.	0
nCTIIRQ[0]	[19]	RW		0
PMUIRQ[0]	[18]	RW		0
PARITYFAILSCU[0]	[17]	RW	Write) 0 = The current setting is unchanged. 1 = Clear the interrupt enable bit to "0".	0
PARITYFAIL0	[16]	RW	Read) The current interrupt enable bit. 0 = Masked. 1 = Enabled.	0
G3_IRQ	[15]	RW		0
G2_IRQ	[14]	RW	Clear the corresponding interrupt enable bit to '0'. If the interrupt enable bit is cleared, the interrupt will be masked.	0
G1_IRQ	[13]	RW		0
G0_IRQ	[12]	RW	Write) 1 = Clear the interrupt enable bit to '0'. 0 = The current setting is unchanged.	0
RSVD	[11]	RW		0
RSVD	[10]	RW	Read) The current interrupt enable bit. 1 = Enabled. 0 = Masked.	0
RSVD	[9]	RW		0
RSVD	[8]	RW		0
RSVD	[7]	RW		0
RSVD	[6]	RW	Clear the corresponding interrupt enable bit to "0". If the interrupt enable bit is cleared, the interrupt will be masked.	0
RSVD	[5]	RW		0
RSVD	[4]	RW	Write) 0 = The current setting is unchanged. 1 = Clear the interrupt enable bit to "0".	0
RSVD	[3]	RW		0
RSVD	[2]	RW	Read) The current interrupt enable bit.	0
RSVD	[1]	RW	0 = Masked. 1 = Enabled.	0
RSVD	[0]	RW		0

7.4.1.3 ISTR0

- Address = 0x1044_0008, Reset Value = Undefined

Name	Bit	Type	Description	Reset Value
RSVD	[31:29]	—	Reserved	0x0
TMU	[28]	R		—
nCTIIRQ[1]	[27]	R		—
PMUIRQ[1]	[26]	R		—
PARITYFAILSCU[1]	[25]	R	Interrupt pending status. The corresponding interrupt enable bit does not affect this pending status. 0 = The interrupt is not pending. 1 = The interrupt is pending.	—
PARITYFAIL1	[24]	R		—
RSVD	[23:21]	—	Reserved	0x0
TMU	[20]	R		—
nCTIIRQ[0]	[19]	R		—
PMUIRQ[0]	[18]	R		—
PARITYFAILSCU[0]	[17]	R	Interrupt pending status. The corresponding interrupt enable bit does not affect this pending status. 0 = The interrupt is not pending. 1 = The interrupt is pending.	—
PARITYFAIL0	[16]	R		—
G3_IRQ	[15]	R		—
G2_IRQ	[14]	R		—
G1_IRQ	[13]	R		—
G0_IRQ	[12]	R	Interrupt pending status. The corresponding interrupt enable bit does not affect this pending status.	—
RSVD	[11]	R	0 = The interrupt is not pending. 1 = The interrupt is pending.	—
RSVD	[10]	R		—
RSVD	[9]	R		—
RSVD	[8]	R		—
RSVD	[7]	R		—
RSVD	[6]	R		—
RSVD	[5]	R		—
RSVD	[4]	R	Interrupt pending status. The corresponding interrupt enable bit does not affect this pending status.	—
RSVD	[3]	R	0 = The interrupt is not pending. 1 = The interrupt is pending.	—
RSVD	[2]	R		—
RSVD	[1]	R		—
RSVD	[0]	R		—

7.4.1.4 IMSR0

- Address = 0x1044_000C, Reset Value = Undefined

Name	Bit	Type	Description	Reset Value
RSVD	[31:29]	—	Reserved	0x0
TMU	[28]	R		—
nCTIIRQ[1]	[27]	R		—
PMUIRQ[1]	[26]	R		—
PARITYFAILSCU[1]	[25]	R		—
PARITYFAIL1	[24]	R		—
RSVD	[23:21]	—	Reserved	0x0
TMU	[20]	R		—
nCTIIRQ[0]	[19]	R		—
PMUIRQ[0]	[18]	R		—
PARITYFAILSCU[0]	[17]	R		—
PARITYFAIL0	[16]	R		—
G3_IRQ	[15]	R		—
G2_IRQ	[14]	R		—
G1_IRQ	[13]	R		—
G0_IRQ	[12]	R		—
RSVD	[11]	R		—
RSVD	[10]	R		—
RSVD	[9]	R		—
RSVD	[8]	R		—
RSVD	[7]	R		—
RSVD	[6]	R		—
RSVD	[5]	R		—
RSVD	[4]	R		—
RSVD	[3]	R		—
RSVD	[2]	R		—
RSVD	[1]	R		—
RSVD	[0]	R		—

7.4.1.5 IESR1

- Address = 0x1044_0010, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
SYSMMU_PCIE[1]	[31]	RW	Set the corresponding interrupt enable bit to "1". If the interrupt enable bit is set, the interrupt request will be served.	0
SYSMMU_MFC_M1[1]	[30]	RW		0
SYSMMU_MFC_M0[1]	[29]	RW		0
SYSMMU_TV_M0[1]	[28]	RW	Write) 0 = The current setting is unchanged. 1 = Set the interrupt enable bit to "1".	0
SYSMMU_LCD1_M1[1]	[27]	RW		0
SYSMMU_LCD0_M0[1]	[26]	RW	Read) The current interrupt enable bit. 0 = Masked. 1 = Enabled.	0
SYSMMU_MDMA1[1]	[25]	RW		0
SYSMMU_ROTATOR [1]	[24]	RW		0
SYSMMU_2D[1]	[23]	RW	Set the corresponding interrupt enable bit to "1". If the interrupt enable bit is set, the interrupt request will be served.	0
SYSMMU_JPEG[1]	[22]	RW		0
SYSMMU_FIMC3[1]	[21]	RW		0
SYSMMU_FIMC2[1]	[20]	RW	Write) 0 = The current setting is unchanged. 1 = Set the interrupt enable bit to "1".	0
SYSMMU_FIMC1[1]	[19]	RW		0
SYSMMU_FIMC0[1]	[18]	RW	Read) The current interrupt enable bit. 0 = Masked. 1 = Enabled.	0
SYSMMU_SSS[1]	[17]	RW		0
SYSMMU_MDMA0[1]	[16]	RW		0
SYSMMU_PCIE[0]	[15]	RW	Set the corresponding interrupt enable bit to "1". If the interrupt enable bit is set, the interrupt request will be served.	0
SYSMMU_MFC_M1[0]	[14]	RW		0
SYSMMU_MFC_M0[0]	[13]	RW		0
SYSMMU_TV_M0[0]	[12]	RW	Write) 0 = The current setting is unchanged. 1 = Set the interrupt enable bit to "1".	0
SYSMMU_LCD1_M1[0]	[11]	RW		0
SYSMMU_LCD0_M0[0]	[10]	RW	Read) The current interrupt enable bit. 0 = Masked. 1 = Enabled.	0
SYSMMU_MDMA1[0]	[9]	RW		0
SYSMMU_ROTATOR [0]	[8]	RW		0
SYSMMU_2D[0]	[7]	RW	Set the corresponding interrupt enable bit to "1". If the interrupt enable bit is set, the interrupt request will be served.	0
SYSMMU_JPEG[0]	[6]	RW		0
SYSMMU_FIMC3[0]	[5]	RW		0
SYSMMU_FIMC2[0]	[4]	RW	Write) 0 = The current setting is unchanged. 1 = Set the interrupt enable bit to "1".	0
SYSMMU_FIMC1[0]	[3]	RW		0
SYSMMU_FIMC0[0]	[2]	RW	Read) The current interrupt enable bit. 0 = Masked. 1 = Enabled.	0
SYSMMU_SSS[0]	[1]	RW		0
SYSMMU_MDMA0[0]	[0]	RW		0

7.4.1.6 IECR1

- Address = 0x1044_0014, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
SYSMMU_PCIE[1]	[31]	RW		0
SYSMMU_MFC_M1[1]	[30]	RW	Clear the corresponding interrupt enable bit to "0". If the interrupt enable bit is cleared, the interrupt will be masked.	0
SYSMMU_MFC_M0[1]	[29]	RW		0
SYSMMU_TV_M0[1]	[28]	RW	Write) 1 = Clear the interrupt enable bit to "0". 0 = The current setting is unchanged.	0
SYSMMU_LCD1_M1[1]	[27]	RW	Read) The current interrupt enable bit. 1 = Enabled. 0 = Masked.	0
SYSMMU_LCD0_M0[1]	[26]	RW		0
SYSMMU_MDMA1[1]	[25]	RW		0
SYSMMU_ROTATOR [1]	[24]	RW		0
SYSMMU_2D[1]	[23]	RW	Clear the corresponding interrupt enable bit to "0". If the interrupt enable bit is cleared, the interrupt will be masked.	0
SYSMMU_JPEG[1]	[22]	RW	Write) 0 = The current setting is unchanged. 1 = Clear the interrupt enable bit to "0".	0
SYSMMU_FIMC3[1]	[21]	RW	Read) The current interrupt enable bit. 0 = Masked. 1 = Enabled.	0
SYSMMU_FIMC2[1]	[20]	RW		0
SYSMMU_FIMC1[1]	[19]	RW		0
SYSMMU_FIMC0[1]	[18]	RW		0
SYSMMU_SSS[1]	[17]	RW		0
SYSMMU_MDMA0[1]	[16]	RW		0
SYSMMU_PCIE[0]	[15]	RW	Clear the corresponding interrupt enable bit to "0". If the interrupt enable bit is cleared, the interrupt will be masked.	0
SYSMMU_MFC_M1[0]	[14]	RW	Write) 0 = The current setting is unchanged. 1 = Clear the interrupt enable bit to "0".	0
SYSMMU_MFC_M0[0]	[13]	RW	Read) The current interrupt enable bit. 0 = Masked. 1 = Enabled.	0
SYSMMU_TV_M0[0]	[12]	RW		0
SYSMMU_LCD1_M1[0]	[11]	RW		0
SYSMMU_LCD0_M0[0]	[10]	RW		0
SYSMMU_MDMA1[0]	[9]	RW		0
SYSMMU_ROTATOR [0]	[8]	RW		0
SYSMMU_2D[0]	[7]	RW	Clear the corresponding interrupt enable bit to "0". If the interrupt enable bit is cleared, the interrupt will be masked.	0
SYSMMU_JPEG[0]	[6]	RW	Write) 0 = The current setting is unchanged. 1 = Clear the interrupt enable bit to "0".	0
SYSMMU_FIMC3[0]	[5]	RW	Read) The current interrupt enable bit. 0 = Masked. 1 = Enabled.	0
SYSMMU_FIMC2[0]	[4]	RW		0
SYSMMU_FIMC1[0]	[3]	RW		0
SYSMMU_FIMC0[0]	[2]	RW		0
SYSMMU_SSS[0]	[1]	RW		0
SYSMMU_MDMA0[0]	[0]	RW		0

7.4.1.7 ISTR1

- Address = 0x1044_0018, Reset Value = Undefined

Name	Bit	Type	Description	Reset Value
SYSMMU_PCIE[1]	[31]	R		–
SYSMMU_MFC_M1[1]	[30]	R		–
SYSMMU_MFC_M0[1]	[29]	R		–
SYSMMU_TV_M0[1]	[28]	R		–
SYSMMU_LCD1_M1[1]	[27]	R		–
SYSMMU_LCD0_M0[1]	[26]	R		–
SYSMMU_MDMA1[1]	[25]	R		–
SYSMMU_ROTATOR [1]	[24]	R		–
SYSMMU_2D[1]	[23]	R		–
SYSMMU_JPEG[1]	[22]	R		–
SYSMMU_FIMC3[1]	[21]	R		–
SYSMMU_FIMC2[1]	[20]	R		–
SYSMMU_FIMC1[1]	[19]	R		–
SYSMMU_FIMC0[1]	[18]	R		–
SYSMMU_SSS[1]	[17]	R		–
SYSMMU_MDMA0[1]	[16]	R		–
SYSMMU_PCIE[0]	[15]	R		–
SYSMMU_MFC_M1[0]	[14]	R		–
SYSMMU_MFC_M0[0]	[13]	R		–
SYSMMU_TV_M0[0]	[12]	R		–
SYSMMU_LCD1_M1[0]	[11]	R		–
SYSMMU_LCD0_M0[0]	[10]	R		–
SYSMMU_MDMA1[0]	[9]	R		–
SYSMMU_ROTATOR [0]	[8]	R		–
SYSMMU_2D[0]	[7]	R		–
SYSMMU_JPEG[0]	[6]	R		–
SYSMMU_FIMC3[0]	[5]	R		–
SYSMMU_FIMC2[0]	[4]	R		–
SYSMMU_FIMC1[0]	[3]	R		–
SYSMMU_FIMC0[0]	[2]	R		–
SYSMMU_SSS[0]	[1]	R		–
SYSMMU_MDMA0[0]	[0]	R		–

7.4.1.8 IMSR1

- Address = 0x1044_001C, Reset Value = Undefined

Name	Bit	Type	Description	Reset Value
SYSMMU_PCIE[1]	[31]	R		–
SYSMMU_MFC_M1[1]	[30]	R		–
SYSMMU_MFC_M0[1]	[29]	R		–
SYSMMU_TV_M0[1]	[28]	R		–
SYSMMU_LCD1_M1[1]	[27]	R		–
SYSMMU_LCD0_M0[1]	[26]	R		–
SYSMMU_MDMA1[1]	[25]	R		–
SYSMMU_ROTATOR [1]	[24]	R		–
SYSMMU_2D[1]	[23]	R		–
SYSMMU_JPEG[1]	[22]	R		–
SYSMMU_FIMC3[1]	[21]	R		–
SYSMMU_FIMC2[1]	[20]	R		–
SYSMMU_FIMC1[1]	[19]	R		–
SYSMMU_FIMC0[1]	[18]	R		–
SYSMMU_SSS[1]	[17]	R		–
SYSMMU_MDMA0[1]	[16]	R		–
SYSMMU_PCIE[0]	[15]	R		–
SYSMMU_MFC_M1[0]	[14]	R		–
SYSMMU_MFC_M0[0]	[13]	R		–
SYSMMU_TV_M0[0]	[12]	R		–
SYSMMU_LCD1_M1[0]	[11]	R		–
SYSMMU_LCD0_M0[0]	[10]	R		–
SYSMMU_MDMA1[0]	[9]	R		–
SYSMMU_ROTATOR [0]	[8]	R		–
SYSMMU_2D[0]	[7]	R		–
SYSMMU_JPEG[0]	[6]	R		–
SYSMMU_FIMC3[0]	[5]	R		–
SYSMMU_FIMC2[0]	[4]	R		–
SYSMMU_FIMC1[0]	[3]	R		–
SYSMMU_FIMC0[0]	[2]	R		–
SYSMMU_SSS[0]	[1]	R		–
SYSMMU_MDMA0[0]	[0]	R		–

7.4.1.9 IESR2

- Address = 0x1044_0020, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
LCD1[3]	[31]	RW	Set the corresponding interrupt enable bit to "1". If the interrupt enable bit is set, the interrupt request will be served.	0
LCD1[2]	[30]	RW		0
LCD1[1]	[29]	RW		0
LCD1[0]	[28]	RW	Write) 0 = The current setting is unchanged. 1 = Set the interrupt enable bit to "1".	0
LCD0[3]	[27]	RW	Read) The current interrupt enable bit.	0
LCD0[2]	[26]	RW	0 = Masked. 1 = Enabled.	0
LCD0[1]	[25]	RW		0
LCD0[0]	[24]	RW		0
DMC1_PPC_PEREV_M	[23]	RW	Set the corresponding interrupt enable bit to "1". If the interrupt enable bit is set, the interrupt request will be served.	0
DMC1_PPC_PEREV_A	[22]	RW		0
DMC0_PPC_PEREV_M	[21]	RW		0
DMC0_PPC_PEREV_A	[20]	RW	Write) 0 = The current setting is unchanged. 1 = Set the interrupt enable bit to "1".	0
MIU	[19]	RW	Read) The current interrupt enable bit.	0
L2CCINTR	[18]	RW	0 = Masked. 1 = Enabled.	0
RP_TIMER	[17]	RW		0
GPIO_AUDIO	[16]	RW		0
RSVD	[15]	-	Reserved	0
PPMU_MFC_M1	[14]	RW	Set the corresponding interrupt enable bit to "1". If the interrupt enable bit is set, the interrupt request will be served.	0
PPMU_MFC_M0	[13]	RW		0
PPMU_3D	[12]	RW		0
PPMU_TV_M0	[11]	RW	Write) 0 = The current setting is unchanged. 1 = Set the interrupt enable bit to "1".	0
PPMU_FILE_D_M0	[10]	RW	Read) The current interrupt enable bit.	0
PPMU_LCD1	[9]	RW	0 = Masked. 1 = Enabled.	0
PPMU_LCD0	[8]	RW		0
PPMU_IMAGE_M0	[7]	RW	Set the corresponding interrupt enable bit to "1". If the interrupt enable bit is set, the interrupt request will be served.	0
PPMU_CAMIF_M0	[6]	RW		0
PPMU_D_RIGHT_M0	[5]	RW		0
PPMU_D_LEFT_M0	[4]	RW	Write) 0 = The current setting is unchanged. 1 = Set the interrupt enable bit to "1".	0
PPMU_ACP0_M0	[3]	RW	Read) The current interrupt enable bit.	0
PPMU_CORE_D_S0	[2]	RW	0 = Masked. 1 = Enabled.	0
PPMU_CORE_D_M1	[1]	RW		0
PPMU_CORE_D_M0	[0]	RW		0

7.4.1.10 IECR2

- Address = 0x1044_0024, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
LCD1[3]	[31]	RW	Clear the corresponding interrupt enable bit to "0". If the interrupt enable bit is cleared, the interrupt will be masked.	0
LCD1[2]	[30]	RW		0
LCD1[1]	[29]	RW		0
LCD1[0]	[28]	RW	Write) 0 = The current setting is unchanged. 1 = Clear the interrupt enable bit to "0".	0
LCD0[3]	[27]	RW		0
LCD0[2]	[26]	RW	Read) The current interrupt enable bit. 0 = Masked. 1 = Enabled.	0
LCD0[1]	[25]	RW		0
LCD0[0]	[24]	RW		0
DMC1_PPC_PEREV_M	[23]	RW	Clear the corresponding interrupt enable bit to "0". If the interrupt enable bit is cleared, the interrupt will be masked.	0
DMC1_PPC_PEREV_A	[22]	RW		0
DMC0_PPC_PEREV_M	[21]	RW		0
DMC0_PPC_PEREV_A	[20]	RW	Write) 0 = The current setting is unchanged. 1 = Clear the interrupt enable bit to "0".	0
MIU	[19]	RW		0
L2CCINTR	[18]	RW	Read) The current interrupt enable bit. 0 = Masked. 1 = Enabled.	0
RP_TIMER	[17]	RW		0
GPIO_AUDIO	[16]	RW		0
RSVD	[15]	-	Reserved	0
PPMU_MFC_M1	[14]	RW	Clear the corresponding interrupt enable bit to '0'. If the interrupt enable bit is cleared, the interrupt will be masked.	0
PPMU_MFC_M0	[13]	RW		0
PPMU_3D	[12]	RW		0
PPMU_TV_M0	[11]	RW	Write) 1 = Clear the interrupt enable bit to '0'. 0 = The current setting is unchanged.	0
PPMU_FILE_D_M0	[10]	RW	Read) The current interrupt enable bit. 1 = Enabled. 0 = Masked.	0
PPMU_LCD1	[9]	RW		0
PPMU_LCD0	[8]	RW		0
PPMU_IMAGE_M0	[7]	RW	Clear the corresponding interrupt enable bit to "0". If the interrupt enable bit is cleared, the interrupt will be masked.	0
PPMU_CAMIF_M0	[6]	RW		0
PPMU_D_RIGHT_M0	[5]	RW	Write) 0 = The current setting is unchanged. 1 = Clear the interrupt enable bit to "0".	0
PPMU_D_LEFT_M0	[4]	RW		0
PPMU_ACP0_M0	[3]	RW		0
PPMU_CORE_D_S0	[2]	RW	Read) The current interrupt enable bit. 0 = Masked. 1 = Enabled.	0
PPMU_CORE_D_M1	[1]	RW		0
PPMU_CORE_D_M0	[0]	RW		0

7.4.1.11 ISTR2

- Address = 0x1044_0028, Reset Value = Undefined

Name	Bit	Type	Description	Reset Value
LCD1[3]	[31]	R	Interrupt pending status. The corresponding interrupt enable bit does not affect this pending status. 0 = The interrupt is not pending. 1 = The interrupt is pending.	–
LCD1[2]	[30]	R		–
LCD1[1]	[29]	R		–
LCD1[0]	[28]	R		–
LCD0[3]	[27]	R		–
LCD0[2]	[26]	R		–
LCD0[1]	[25]	R		–
LCD0[0]	[24]	R		–
DMC1_PPC_PEREV_M	[23]	R	Interrupt pending status. The corresponding interrupt enable bit does not affect this pending status. 0 = The interrupt is not pending. 1 = The interrupt is pending.	–
DMC1_PPC_PEREV_A	[22]	R		–
DMC0_PPC_PEREV_M	[21]	R		–
DMC0_PPC_PEREV_A	[20]	R		–
MIU	[19]	R		–
L2CCINTR	[18]	R		–
RP_TIMER	[17]	R		–
GPIO_AUDIO	[16]	R		–
RSVD	[15]	–	Reserved	0
PPMU_MFC_M1	[14]	R	Interrupt pending status. The corresponding interrupt enable bit does not affect this pending status. 0 = The interrupt is not pending. 1 = The interrupt is pending.	–
PPMU_MFC_M0	[13]	R		–
PPMU_3D	[12]	R		–
PPMU_TV_M0	[11]	R		–
PPMU_FILE_D_M0	[10]	R		–
PPMU_LCD1	[9]	R		–
PPMU_LCD0	[8]	R		–
PPMU_IMAGE_M0	[7]	R		–
PPMU_CAMIF_M0	[6]	R	Interrupt pending status. The corresponding interrupt enable bit does not affect this pending status. 0 = The interrupt is not pending. 1 = The interrupt is pending.	–
PPMU_D_RIGHT_M0	[5]	R		–
PPMU_D_LEFT_M0	[4]	R		–
PPMU_ACP0_M0	[3]	R		–
PPMU_CORE_D_S0	[2]	R		–
PPMU_CORE_D_M1	[1]	R		–
PPMU_CORE_D_M0	[0]	R		–

7.4.1.12 IMSR2

- Address = 0x1044_002C, Reset Value = Undefined

Name	Bit	Type	Description	Reset Value
LCD1[3]	[31]	R		–
LCD1[2]	[30]	R		–
LCD1[1]	[29]	R		–
LCD1[0]	[28]	R		–
LCD0[3]	[27]	R		–
LCD0[2]	[26]	R		–
LCD0[1]	[25]	R		–
LCD0[0]	[24]	R		–
DMC1_PPC_PEREV_M	[23]	R		–
DMC1_PPC_PEREV_A	[22]	R		–
DMC0_PPC_PEREV_M	[21]	R		–
DMC0_PPC_PEREV_A	[20]	R		–
MIU	[19]	R		–
L2CCINTR	[18]	R		–
RP_TIMER	[17]	R		–
GPIO_AUDIO	[16]	R		–
RSVD	[15]	–	Reserved	0
PPMU_MFC_M1	[14]	R		–
PPMU_MFC_M0	[13]	R		–
PPMU_3D	[12]	R		–
PPMU_TV_M0	[11]	R		–
PPMU_FILE_D_M0	[10]	R	0 = The interrupt is not pending. 1 = The interrupt is pending.	–
PPMU_LCD1	[9]	R		–
PPMU_LCD0	[8]	R		–
PPMU_IMAGE_M0	[7]	R		–
PPMU_CAMIF_M0	[6]	R		–
PPMU_D_RIGHT_M0	[5]	R		–
PPMU_D_LEFT_M0	[4]	R		–
PPMU_ACP0_M0	[3]	R		–
PPMU_CORE_D_S0	[2]	R	0 = The interrupt is not pending. 1 = The interrupt is pending.	–
PPMU_CORE_D_M1	[1]	R		–
PPMU_CORE_D_M0	[0]	R		–

7.4.1.13 IESR3

- Address = 0x1044_0030, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
DECERRINTR	[31]	RW	Set the corresponding interrupt enable bit to "1". If the interrupt enable bit is set, the interrupt request will be served.	0
SLVERRINTR	[30]	RW		0
ERRRDINTR	[29]	RW		0
ERRRTINTR	[28]	RW	Write) 0 = The current setting is unchanged. 1 = Set the interrupt enable bit to "1".	0
ERRWDINTR	[27]	RW		0
ERRWTINTR	[26]	RW	Read) The current interrupt enable bit. 0 = Masked. 1 = Enabled.	0
ECNTRINTR	[25]	RW		0
SCUEVABORT	[24]	RW		0
RSVD	[23]	-	Reserved	0
RSVD	[22]	-	Reserved for CPU1	0
RSVD	[21:16]	-	Reserved for G3D	0x00
RSVD	[15:14]	-	Reserved	00
RSVD	[13]	-	Reserved for CPU0	0
RSVD	[12:8]	-	Reserved for G3D	0x00
G3_IRQ	[7]	RW	Set the corresponding interrupt enable bit to "1". If the interrupt enable bit is set, the interrupt request will be served.	0
G2_IRQ	[6]	RW		0
G1_IRQ	[5]	RW		0
G0_IRQ	[4]	RW	Write) 0 = The current setting is unchanged. 1 = Set the interrupt enable bit to "1".	0
LCD1[3]	[3]	RW		0
LCD1[2]	[2]	RW	Read) The current interrupt enable bit.	0
LCD1[1]	[1]	RW	0 = Masked. 1 = Enabled.	0
LCD1[0]	[0]	RW		0

7.4.1.14 IECR3

- Address = 0x1044_0034, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
DECERRINTR	[31]	RW		0
SLVERRINTR	[30]	RW		0
ERRRDINTR	[29]	RW		0
ERRRTINTR	[28]	RW	Write) 0 = The current setting is unchanged. 1 = Clear the interrupt enable bit to "0".	0
ERRWDINTR	[27]	RW		0
ERRWTINTR	[26]	RW	Read) The current interrupt enable bit. 0 = Masked. 1 = Enabled.	0
ECNTRINTR	[25]	RW		0
SCUEVABORT	[24]	RW		0
RSVD	[23]	-	Reserved	0
RSVD	[22]	-	Reserved for CPU1	0
RSVD	[21:16]	-	Reserved for G3D	0x00
RSVD	[15:14]	-	Reserved	00
RSVD	[13]	-	Reserved for CPU0	0
RSVD	[12:8]	-	Reserved for G3D	0x00
G3_IRQ	[7]	RW		0
G2_IRQ	[6]	RW	Clear the corresponding interrupt enable bit to "0". If the interrupt enable bit is cleared, the interrupt will be masked.	0
G1_IRQ	[5]	RW		0
G0_IRQ	[4]	RW	Write) 0 = The current setting is unchanged. 1 = Clear the interrupt enable bit to "0".	0
LCD1[3]	[3]	RW		0
LCD1[2]	[2]	RW	Read) The current interrupt enable bit.	0
LCD1[1]	[1]	RW	0 = Masked. 1 = Enabled.	0
LCD1[0]	[0]	RW		0

7.4.1.15 ISTR3

- Address = 0x1044_0038, Reset Value = Undefined

Name	Bit	Type	Description	Reset Value
DECERRINTR	[31]	R		–
SLVERRINTR	[30]	R		–
ERRRDINTR	[29]	R		–
ERRRTINTR	[28]	R	Interrupt pending status. The corresponding interrupt enable bit does not affect this pending status.	–
ERRWDINTR	[27]	R	0 = The interrupt is not pending. 1 = The interrupt is pending.	–
ERRWTINTR	[26]	R		–
ECNTRINTR	[25]	R		–
SCUEVABORT	[24]	R		–
RSVD	[23]	–	Reserved	0
RSVD	[22]	–	Reserved for CPU1	–
RSVD	[21:16]	–	Reserved for G3D	–
RSVD	[15:14]	–	Reserved	00
RSVD	[13]	–	Reserved for CPU0	–
RSVD	[12:8]	–	Reserved for G3D	–
G3_IRQ	[7]	R		–
G2_IRQ	[6]	R		–
G1_IRQ	[5]	R		–
G0_IRQ	[4]	R	Interrupt pending status. The corresponding interrupt enable bit does not affect this pending status.	–
LCD1[3]	[3]	R	0 = The interrupt is not pending. 1 = The interrupt is pending.	–
LCD1[2]	[2]	R		–
LCD1[1]	[1]	R		–
LCD1[0]	[0]	R		–

7.4.1.16 IMSR3

- Address = 0x1044_003C, Reset Value = Undefined

Name	Bit	Type	Description	Reset Value
DECERRINTR	[31]	R	Masked interrupt pending status. If the corresponding interrupt enable bit is "0", the IMSR bit is read out as "0". 0 = The interrupt is not pending. 1 = The interrupt is pending.	—
SLVERRINTR	[30]	R		—
ERRRDINTR	[29]	R		—
ERRRTINTR	[28]	R		—
ERRWDINTR	[27]	R		—
ERRWTINTR	[26]	R		—
ECNTRINTR	[25]	R		—
SCUEVABORT	[24]	R		—
RSVD	[23]	—	Reserved	0
RSVD	[22]	—	Reserved for CPU1	—
RSVD	[21:16]	—	Reserved for G3D	—
RSVD	[15:14]	—	Reserved	00
RSVD	[13]	—	Reserved for CPU0	—
RSVD	[12:8]	—	Reserved for G3D	—
G3_IRQ	[7]	R	Masked interrupt pending status. If the corresponding interrupt enable bit is "0", the IMSR bit is read out as "0". 0 = The interrupt is not pending. 1 = The interrupt is pending.	—
G2_IRQ	[6]	R		—
G1_IRQ	[5]	R		—
G0_IRQ	[4]	R		—
LCD1[3]	[3]	R		—
LCD1[2]	[2]	R		—
LCD1[1]	[1]	R		—
LCD1[0]	[0]	R		—

7.4.1.17 CIPSR0

- Address = 0x1044_0100, Reset Value = Undefined

Name	Bit	Type	Description	Reset Value
INTG31 – INTG16	[31:16]	–	Reserved	–
INTG15	[15]	R	Combined interrupt pending status. 1 = The combined interrupt is pending, which means the corresponding interrupt request to the GIC is asserted. 0 = The combined interrupt is not pending.	–
INTG14	[14]	–	Reserved	–
INTG13	[13]	–	Reserved	–
INTG12	[12]	R	Combined interrupt pending status. 0 = The combined interrupt is not pending. 1 = The combined interrupt is pending, which means the corresponding interrupt request to the GIC is asserted.	–
INTG11	[11]	R		–
INTG10	[10]	R		–
INTG9	[9]	R		–
INTG8	[8]	R		–
INTG7	[7]	R		–
INTG6	[6]	R		–
INTG5	[5]	R		–
INTG4	[4]	R		–
INTG3	[3]	R		–
INTG2	[2]	R		–
INTG1	[1]	R		–
INTG0	[0]	R		–

8 DMA Controller

8.1 Overview

Exynos4210 supports two Direct Memory Access (DMA) tops: one for Memory-to-Memory (M2M) transfer (DMA_mem) and another for Peripheral-to-memory transfer and vice-versa (DMA_peri). The M2M DMA top (DMA_mem) consists of one PL330 and some logics. Peri DMA top (DMA_peri) consists of two PL330s (DMA0 and DMA1) and dma_map. DMA_mem resides in LCD0 functional block. DMA_peri resides in the FSYS functional block (Refer CMU manual for detailed description of Clock). The boot_manager_ns bit for DMA_mem and DMA_peri is tied to 1. For more details, please refer to the PL330 TRM, "AMBA DMA Controller DMA-330 technical reference manua revision r1p0" from ARM®.

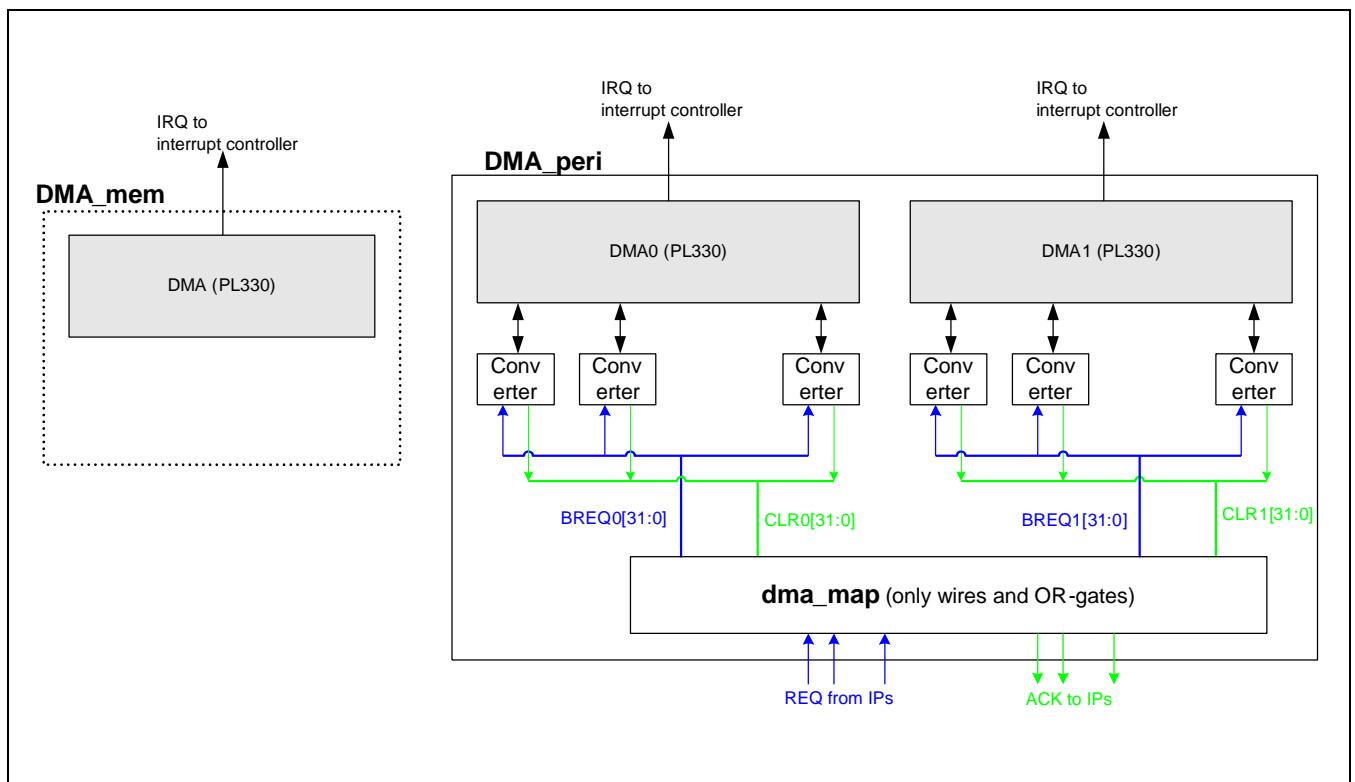


Figure 8-1 Two DMA Tops

8.2 Features

The key features of DMA Controller are listed below as reference for DMA and for writing DMA assembly code.

Key Features	DMA_mem	DMA_peri
Supports Data Size	Up to double word (64-bit)	Up to word (32-bit)
Supports Burst Size	Up to 16 burst	Up to 8 burst
Supports Channel	8 channels at the same time	16 channels at the same time

Although each DMA module has 32 interrupt sources, only one interrupt is sent to Interrupt Controller for each DMA.

Table 8-1 DMA Request Mapping Table

Module	No.	
Peri DMA1	31	Reserved
	30	Reserved
	29	Reserved
	28	Reserved
	27	SPDIF
	26	Siimbus0AUX_TX
	25	Siimbus0AUX_RX
	24	Siimbus5_RX
	23	Siimbus5_RX
	22	Siimbus3_TX
	21	Siimbus3_RX
	20	Slimbus1_TX
	19	Slimbus1_RX
	18	UART3_TX
	17	UART3_RX
	16	UART1_TX
	15	UART1_RX
	14	UART0_TX
	13	UART0_RX
	12	I2S1_TX
	11	I2S1_RX
	10	I2S0_TX
	9	I2S0_RX
	8	I2S0S_TX
	7	SPI1_TX

Module	No.	
Peri DMA0	6	SPI1_RX
	5	MSM_REQ3
	4	MSM_REQ1
	3	PCM1_TX
	2	PCM1_RX
	1	PCM0_TX
	0	PCM0_RX
	31	Reserved
	30	Reserved
	29	AC_PCMout
	28	AC_PCMin
	27	AC_MICin
	26	SlimBUS4_TX
	25	SlimBUS4_RX
	24	SlimBUS2_TX
	23	SlimBUS2_RX
	22	SlimBUS0_TX
	21	SlimBUS0_RX
	20	UART4_TX
	19	UART4_RX
	18	UART2_TX
	17	UART2_RX
	16	UART0_TX
	15	UART0_RX
	14	I2S2_TX
	13	I2S2_RX
	12	I2S0_TX
	11	I2S0_RX
	10	I2S0S_TX
	9	SPI2_TX
	8	SPI2_RX
	7	SPI0_TX
	6	SPI0_RX
	5	MSM_REQ2
	4	MSM_REQ0
	3	PCM2_TX
	2	PCM2_RX

Module	No.	
	1	PCM0_TX
	0	PCM0_RX
DMA_mem		

Caution: When PDMA0 or PDMA1 are enabled, then you have to check CLKGATE status

8.3 Functional Description

8.3.1 Instructions

Please refer to the PL330 TRM, "AMBA DMA Controller DMA-330 technical reference manual revision r1p0" from ARM®.

8.3.1.1 For the Security Scheme,

DMA_mem and DMA_Periph run only in non-secure mode.

8.3.1.2 Summary

1. You can configure the DMAC with up to eight DMA channels for 1-DMA, with each channel being capable of supporting a single concurrent thread of DMA operation. In addition, there is a single DMA manager thread to initialize the DMA channel thread.
2. Channel thread
 - A. Each channel thread can operate the DMA. You must write assembly code accordingly. If you need a number of independent DMA channels, you must write a number of assembly codes for each channel.
 - B. Assemble them, link them into one file, and load this file into memory.

8.4 Register Description

Most Special Function Registers (SFRs) are read-only. The main role of SFR is to check the PL330 status. There are many SFRs for PL330. In this section, only Exynos4210-specific SFRs are explained. For more information, please refer to Chapter 3, "please refer to a document, AMBA DMA Controller DMA-330 technical reference manual".

8.4.1 Register Map Summary for MDMA

- Base Address: 0x1284_0000(s-APB), 0x1285_0000(ns-APB))

Register	Offset	Description	Reset Value
DS	0x0000	Specifies the DMA Status Register. For more information, refer to page 3-11 of "PL330 TRM".	0x00000200
DPC	0x0004	Specifies the DMA Program Counter Register. For more information, refer to page 3-12 of "PL330 TRM".	0x0
RSVD	0x0008 to 0x001C	Reserved	Undefined
INTEN	0x0020	Specifies the Interrupt Enable Register. For more information, refer to page 3-13 of "PL330 TRM".	0x0
ES	0x0024	Specifies the Event Status Register. For more information, refer to page 3-13 of "PL330 TRM".	0x0
INTSTATUS	0x0028	Specifies the Interrupt Status Register. For more information, refer to page 3-14 of "PL330 TRM".	0x0
INTCLR	0x002C	Specifies the Interrupt Clear Register. For more information, refer to page 3-15 of "PL330 TRM".	0x0
FSM	0x0030	Specifies the Fault Status DMA Manager Register. For more information, refer to page 3-16 of "PL330 TRM".	0x0
FSC	0x0034	Specifies the Fault Status DMA Channel Register. For more information, refer to page 3-16 of "PL330 TRM".	0x0
FTM	0x0038	Specifies the Fault Type DMA Manager Register. For more information, refer to page 3-17 of "PL330 TRM".	0x0
RSVD	0x003C	Reserved	Undefined
FTC0	0x0040	Specifies the Fault type for DMA Channel 0.	0x0
FTC1	0x0044	Specifies the Fault type for DMA Channel 1.	0x0
FTC2	0x0048	Specifies the Fault type for DMA Channel 2.	0x0
FTC3	0x004C	Specifies the Fault type for DMA Channel 3.	0x0
FTC4	0x0050	Specifies the Fault type for DMA Channel 4.	0x0
FTC5	0x0054	Specifies the Fault type for DMA Channel 5.	0x0
FTC6	0x0058	Specifies the Fault type for DMA Channel 6.	0x0
FTC7	0x005C	Specifies the Fault type for DMA Channel 7.	0x0
RSVD	0x0060 to 0x00FC	Reserved	Undefined
CS0	0x0100	Specifies the Channel Status for DMA Channel 0.	0x0
CS1	0x0108	Specifies the Channel Status for DMA Channel 1.	0x0

Register	Offset	Description	Reset Value
CS2	0x0110	Specifies the Channel Status for DMA Channel 2.	0x0
CS3	0x0118	Specifies the Channel Status for DMA Channel 3.	0x0
CS4	0x0120	Specifies the Channel Status for DMA Channel 4.	0x0
CS5	0x0128	Specifies the Channel Status for DMA Channel 5.	0x0
CS6	0x0130	Specifies the Channel Status for DMA Channel 6.	0x0
CS7	0x0138	Specifies the Channel Status for DMA Channel 7.	0x0
CPC0	0x0104	Specifies the Channel PC for DMA Channel 0.	0x0
CPC1	0x010C	Specifies the Channel PC for DMA Channel 1.	0x0
CPC2	0x0114	Specifies the Channel PC for DMA Channel 2.	0x0
CPC3	0x011C	Specifies the Channel PC for DMA Channel 3.	0x0
CPC4	0x0124	Specifies the Channel PC for DMA Channel 4.	0x0
CPC5	0x012C	Specifies the Channel PC for DMA Channel 5.	0x0
CPC6	0x0134	Specifies the Channel PC for DMA Channel 6.	0x0
CPC7	0x013C	Specifies the Channel PC for DMA Channel 7.	0x0
RSVD	0x0140 to 0x03FC	Reserved	Undefined
SA_0	0x0400	Specifies the Source Address for DMA Channel 0.	0x0
SA_1	0x0420	Specifies the Source Address for DMA Channel 1.	0x0
SA_2	0x0440	Specifies the Source Address for DMA Channel 2.	0x0
SA_3	0x0460	Specifies the Source Address for DMA Channel 3.	0x0
SA_4	0x0480	Specifies the Source Address for DMA Channel 4.	0x0
SA_5	0x04A0	Specifies the Source Address for DMA Channel 5.	0x0
SA_6	0x04C0	Specifies the Source Address for DMA Channel 6.	0x0
SA_7	0x04E0	Specifies the Source Address for DMA Channel 7.	0x0
DA_0	0x0404	Specifies the Destination Address for DMA Channel 0.	0x0
DA_1	0x0424	Specifies the Destination Address for DMA Channel 1.	0x0
DA_2	0x0444	Specifies the Destination Address for DMA Channel 2.	0x0
DA_3	0x0464	Specifies the Destination Address for DMA Channel 3.	0x0
DA_4	0x0484	Specifies the Destination Address for DMA Channel 4.	0x0
DA_5	0x04A4	Specifies the Destination Address for DMA Channel 5.	0x0
DA_6	0x04C4	Specifies the Destination Address for DMA Channel 6.	0x0
DA_7	0x04E4	Specifies the Destination Address for DMA Channel 7.	0x0
CC_0	0x0408	Specifies the Channel Control for DMA Channel 0.	0x00800200
CC_1	0x0428	Specifies the Channel Control for DMA Channel 1.	0x00800200
CC_2	0x0448	Specifies the Channel Control for DMA Channel 2.	0x00800200
CC_3	0x0468	Specifies the Channel Control for DMA Channel 3.	0x00800200
CC_4	0x0488	Specifies the Channel Control for DMA Channel 4.	0x00800200

Register	Offset	Description	Reset Value
CC_5	0x04A8	Specifies the Channel Control for DMA Channel 5.	0x00800200
CC_6	0x04C8	Specifies the Channel Control for DMA Channel 6.	0x00800200
CC_7	0x04E8	Specifies the Channel Control for DMA Channel 7.	0x00800200
LC0_0	0x040C	Specifies the Loop Counter 0 for DMA Channel 0.	0x0
LC0_1	0x042C	Specifies the Loop Counter 0 for DMA Channel 1.	0x0
LC0_2	0x044C	Specifies the Loop Counter 0 for DMA Channel 2.	0x0
LC0_3	0x046C	Specifies the Loop Counter 0 for DMA Channel 3.	0x0
LC0_4	0x048C	Specifies the Loop Counter 0 for DMA Channel 4.	0x0
LC0_5	0x04AC	Specifies the Loop Counter 0 for DMA Channel 5.	0x0
LC0_6	0x04CC	Specifies the Loop Counter 0 for DMA Channel 6.	0x0
LC0_7	0x04EC	Specifies the Loop Counter 0 for DMA Channel 7.	0x0
LC1_0	0x0410	Specifies the Loop Counter 1 for DMA Channel 0.	0x0
LC1_1	0x0430	Specifies the Loop Counter 1 for DMA Channel 1.	0x0
LC1_2	0x0450	Specifies the Loop Counter 1 for DMA Channel 2.	0x0
LC1_3	0x0470	Specifies the Loop Counter 1 for DMA Channel 3.	0x0
LC1_4	0x0490	Specifies the Loop Counter 1 for DMA Channel 4.	0x0
LC1_5	0x04B0	Specifies the Loop Counter 1 for DMA Channel 5.	0x0
LC1_6	0x04D0	Specifies the Loop Counter 1 for DMA Channel 6.	0x0
LC1_7	0x04F0	Specifies the Loop Counter 1 for DMA Channel 7.	0x0
RSVD	0x0414 to 0x041C	Reserved	Undefined
RSVD	0x0434 to 0x043C	Reserved	Undefined
RSVD	0x0454 to 0x045C	Reserved	Undefined
RSVD	0x0474 to 0x047C	Reserved	Undefined
RSVD	0x0494 to 0x049C	Reserved	Undefined
RSVD	0x04B4 to 0x04BC	Reserved	Undefined
RSVD	0x04D4 to 0x04DC	Reserved	Undefined
RSVD	0x04F4 to 0x0CFC	Reserved	Undefined
DBGSTATUS	0x0D00	Specifies the Debug Status Register. For more information, refer to page 3-30 of "PL330 TRM".	0x0
DBGCMD	0x0D04	Specifies the Debug Command Register. For more information, refer to page 3-31 of "PL330 TRM".	Undefined

Register	Offset	Description	Reset Value
DBGINST0	0x0D08	Specifies the Debug Instruction-0 Register. For more information, refer to page 3-32 of "PL330 TRM".	Undefined
DBGINST1	0x0D0C	Specifies the Debug Instruction-1 Register. For more information, refer to page 3-33 of "PL330 TRM".	Undefined
CR0	0x0E00	Specifies the Configuration Register 0. For more information, refer to page 3-33 of "PL330 TRM".	0x003E_0075
CR1	0x0E04	Specifies the Configuration Register 1. For more information, refer to page 3-35 of "PL330 TRM".	0x0000_0075
CR2	0x0E08	Specifies the Configuration Register 2. For more information, refer to page 3-36 of "PL330 TRM".	0x0
CR3	0x0E0C	Specifies the Configuration Register 3. For more information, refer to page 3-36 of "PL330 TRM".	0xFFFF_FFFF
CR4	0x0E10	Specifies the Configuration Register 4. For more information, refer to page 3-37 of "PL330 TRM".	0x0000_0001
CRDn	0x0E14	Specifies the Configuration Register Dn. For more information, refer to page 3-38 of "PL330 TRM".	0x03F7_3733
periph_id_n	0x0FE0 to 0x0FEC	Specifies the Peripheral Identification Registers 0-3. For more information, refer to page 3-41 of "PL330 TRM".	Configuration-dependent
pcell_id_n	0x0FF0 to 0x0FFC	Specifies the PrimeCell Identification Registers 0-3. For more information, refer to page 3-43 of "PL330 TRM".	Configuration-dependent

8.4.2 Register Map Summary for PDMA0/PDMA1

- Base Address: 0x1268_0000, 0x1269_0000

Register	Offset	Description	Reset Value
DS	0x0000	Specifies the DMA Status Register. For more information, refer to page 3-12 of "PL330 TRM".	0x00000200
DPC	0x0004	Specifies the DMA Program Counter Register. For more information, refer to page 3-12 of "PL330 TRM".	0x0
RSVD	0x0008 to 0x001C	Reserved	Undefined
INTEN	0x0020	Specifies the Interrupt Enable Register. For more information, refer to page 3-13 of "PL330 TRM".	0x0
ES	0x0024	Specifies the Event Status Register. For more information, refer to page 3-13 of "PL330 TRM".	0x0
INTSTATUS	0x0028	Specifies the Interrupt Status Register. For more information, refer to page 3-14 of "PL330 TRM".	0x0
INTCLR	0x002C	Specifies the Interrupt Clear Register. For more information, refer to page 3-15 of "PL330 TRM".	0x0
FSM	0x0030	Specifies the Fault Status DMA Manager Register. For more information, refer to page 3-16 of "PL330 TRM".	0x0
FSC	0x0034	Specifies the Fault Status DMA Channel Register. For more information, refer to page 3-16 of "PL330 TRM".	0x0
FTM	0x0038	Specifies the Fault Type DMA Manager Register. For more information, refer to page 3-17 of "PL330 TRM".	0x0
RSVD	0x003C	Reserved	Undefined
FTC0	0x0040	Specifies the Fault Type for DMA Channel 0.	0x0
FTC1	0x0044	Specifies the Fault Type for DMA Channel 1.	0x0
FTC2	0x0048	Specifies the Fault Type for DMA Channel 2.	0x0
FTC3	0x004C	Specifies the Fault Type for DMA Channel 3.	0x0
FTC4	0x0050	Specifies the Fault Type for DMA Channel 4.	0x0
FTC5	0x0054	Specifies the Fault Type for DMA Channel 5.	0x0
FTC6	0x0058	Specifies the Fault Type for DMA Channel 6.	0x0
FTC7	0x005C	Specifies the Fault Type for DMA Channel 7.	0x0
RSVD	0x0060 to 0x00FC	Reserved	Undefined
CS0	0x0100	Specifies the Channel Status for DMA Channel 0.	0x0
CS1	0x0108	Specifies the Channel Status for DMA Channel 1.	0x0
CS2	0x0110	Specifies the Channel Status for DMA Channel 2.	0x0
CS3	0x0118	Specifies the Channel Status for DMA Channel 3.	0x0
CS4	0x0120	Specifies the Channel Status for DMA Channel 4.	0x0
CS5	0x0128	Specifies the Channel Status for DMA Channel 5.	0x0

Register	Offset	Description	Reset Value
CS6	0x0130	Specifies the Channel Status for DMA Channel 6.	0x0
CS7	0x0138	Specifies the Channel Status for DMA Channel 7.	0x0
CPC0	0x0104	Specifies the Channel PC for DMA Channel 0.	0x0
CPC1	0x010C	Specifies the Channel PC for DMA Channel 1.	0x0
CPC2	0x0114	Specifies the Channel PC for DMA Channel 2.	0x0
CPC3	0x011C	Specifies the Channel PC for DMA Channel 3.	0x0
CPC4	0x0124	Specifies the Channel PC for DMA Channel 4.	0x0
CPC5	0x012C	Specifies the Channel PC for DMA Channel 5.	0x0
CPC6	0x0134	Specifies the Channel PC for DMA Channel 6.	0x0
CPC7	0x013C	Specifies the Channel PC for DMA Channel 7.	0x0
RSVD	0x0140 to 0x03FC	Reserved	Undefined
SA_0	0x0400	Specifies the Source Address for DMA Channel 0.	0x0
SA_1	0x0420	Specifies the Source Address for DMA Channel 1.	0x0
SA_2	0x0440	Specifies the Source Address for DMA Channel 2.	0x0
SA_3	0x0460	Specifies the Source Address for DMA Channel 3.	0x0
SA_4	0x0480	Specifies the Source Address for DMA Channel 4.	0x0
SA_5	0x04A0	Specifies the Source Address for DMA Channel 5.	0x0
SA_6	0x04C0	Specifies the Source Address for DMA Channel 6.	0x0
SA_7	0x04E0	Specifies the Source Address for DMA Channel 7.	0x0
DA_0	0x0404	Specifies the Destination Address for DMA Channel 0.	0x0
DA_1	0x0424	Specifies the Destination Address for DMA Channel 1.	0x0
DA_2	0x0444	Specifies the Destination Address for DMA Channel 2.	0x0
DA_3	0x0464	Specifies the Destination Address for DMA Channel 3.	0x0
DA_4	0x0484	Specifies the Destination Address for DMA Channel 4.	0x0
DA_5	0x04A4	Specifies the Destination Address for DMA Channel 5.	0x0
DA_6	0x04C4	Specifies the Destination Address for DMA Channel 6.	0x0
DA_7	0x04E4	Specifies the Destination Address for DMA Channel 7.	0x0
CC_0	0x0408	Specifies the Channel Control for DMA Channel 0. dst_burst_size & src_burst_size don't support 16 bytes per beat.	0x0
CC_1	0x0428	Specifies the Channel Control for DMA Channel 1. dst_burst_size & src_burst_size don't support 16 bytes per beat.	0x0
CC_2	0x0448	Specifies the Channel Control for DMA Channel 2. dst_burst_size & src_burst_size don't support 16 bytes per beat.	0x0
CC_3	0x0468	Specifies the Channel Control for DMA Channel 3. dst_burst_size & src_burst_size don't support 16 bytes per beat.	0x0
CC_4	0x0488	Specifies the Channel Control for DMA Channel 4. dst_burst_size & src_burst_size don't support 16 bytes per beat.	0x0

Register	Offset	Description	Reset Value
CC_5	0x04A8	Specifies the Channel Control for DMA Channel 5. dst_burst_size & src_burst_size don't support 16 bytes per beat.	0x0
CC_6	0x04C8	Specifies the Channel Control for DMA Channel 6. dst_burst_size & src_burst_size don't support 16 bytes per beat.	0x0
CC_7	0x04E8	Specifies the Channel Control for DMA Channel 7. dst_burst_size & src_burst_size don't support 16 bytes per beat.	0x0
LC0_0	0x040C	Specifies the Loop Counter 0 for DMA Channel 0.	0x0
LC0_1	0x042C	Specifies the Loop Counter 0 for DMA Channel 1.	0x0
LC0_2	0x044C	Specifies the Loop Counter 0 for DMA Channel 2.	0x0
LC0_3	0x046C	Specifies the Loop Counter 0 for DMA Channel 3.	0x0
LC0_4	0x048C	Specifies the Loop Counter 0 for DMA Channel 4.	0x0
LC0_5	0x04AC	Specifies the Loop Counter 0 for DMA Channel 5.	0x0
LC0_6	0x04CC	Specifies the Loop Counter 0 for DMA Channel 6.	0x0
LC0_7	0x04EC	Specifies the Loop Counter 0 for DMA Channel 7.	0x0
LC1_0	0x0410	Specifies the Loop Counter 1 for DMA Channel 0.	0x0
LC1_1	0x0430	Specifies the Loop Counter 1 for DMA Channel 1.	0x0
LC1_2	0x0450	Specifies the Loop Counter 1 for DMA Channel 2.	0x0
LC1_3	0x0470	Specifies the Loop Counter 1 for DMA Channel 3.	0x0
LC1_4	0x0490	Specifies the Loop Counter 1 for DMA Channel 4.	0x0
LC1_5	0x04B0	Specifies the Loop Counter 1 for DMA Channel 5.	0x0
LC1_6	0x04D0	Specifies the Loop Counter 1 for DMA Channel 6.	0x0
LC1_7	0x04F0	Specifies the Loop Counter 1 for DMA Channel 7.	0x0
RSVD	0x0414 to 0x041C	Reserved	Undefined
RSVD	0x0434 to 0x043C	Reserved	Undefined
RSVD	0x0454 to 0x045C	Reserved	Undefined
RSVD	0x0474 to 0x047C	Reserved	Undefined
RSVD	0x0494 to 0x049C	Reserved	Undefined
RSVD	0x04B4 to 0x04BC	Reserved	Undefined
RSVD	0x04D4 to 0x04DC	Reserved	Undefined
RSVD	0x04F4 to 0x0CFC	Reserved	Undefined
DBGSTATUS	0x0D00	Specifies the Debug Status Register on page 3-30 of "TRM".	0x0

Register	Offset	Description	Reset Value
DBGCMD	0x0D04	Specifies the Debug Command Register. For more information, refer to page 3-31 of "PL330 TRM".	Undefined
DBGINST0	0x0D08	Specifies the Debug Instruction-0 Register. For more information, refer to page 3-32 of "PL330 TRM".	Undefined
DBGINST1	0x0D0C	Specifies the Debug Instruction-1 Register. For more information, refer to page 3-33 of "PL330 TRM".	Undefined
CR0	0x0E00	Specifies the Configuration Register 0. For more information, refer to page 3-33 of "PL330 TRM".	0x003F_F075
CR1	0x0E04	Specifies the Configuration Register 1. For more information, refer to page 3-35 of "PL330 TRM".	0x0000_0074
CR2	0x0E08	Specifies the Configuration Register 2. For more information, refer to page 3-36 of "PL330 TRM".	0x0000_0000
CR3	0x0E0C	Specifies the Configuration Register 3. For more information, refer to page 3-36 of "PL330 TRM".	0xFFFF_FFFF
CR4	0x0E10	Specifies the Configuration Register 4. For more information, refer to page 3-37 of "PL330 TRM".	0xFFFF_FFFF
CRDn	0x0E14	Specifies the Configuration Register Dn. For more information, refer to page 3-38 of "PL330 TRM".	0x01F7_3732
periph_id_n	0x0FE0 to 0x0FEC	Specifies the Peripheral Identification Registers 0-3. For more information, refer to page 3-41 of "PL330 TRM".	Configuration-dependent
pcell_id_n	0x0FF0 to 0x0FFC	Specifies the PrimeCell Identification Registers 0-3. For more information refer to page 3-43 of "PL330 TRM".	Configuration-dependent

NOTE: The SFR description shows only the restricted and fixed part of some SFR. PL330 TRM shows detailed information of other parts and other SFRs.

9 SROM Controller

9.1 Overview

Exynos4210 SROM Controller (SROMC) supports external 8/16-bit NOR Flash/ PROM/ SRAM memory

9.2 Features

- Supports SRAM, various ROMs and NOR flash memory
- Supports only 8 or 16-bit data bus
- Address space: Up to 128 KB per Bank
- Supports 4 banks.
- Fixed memory bank start address
- External wait to extend the bus cycle
- Supports byte and half-word access for external memory

9.3 Functional Description

9.3.1 Block Diagram

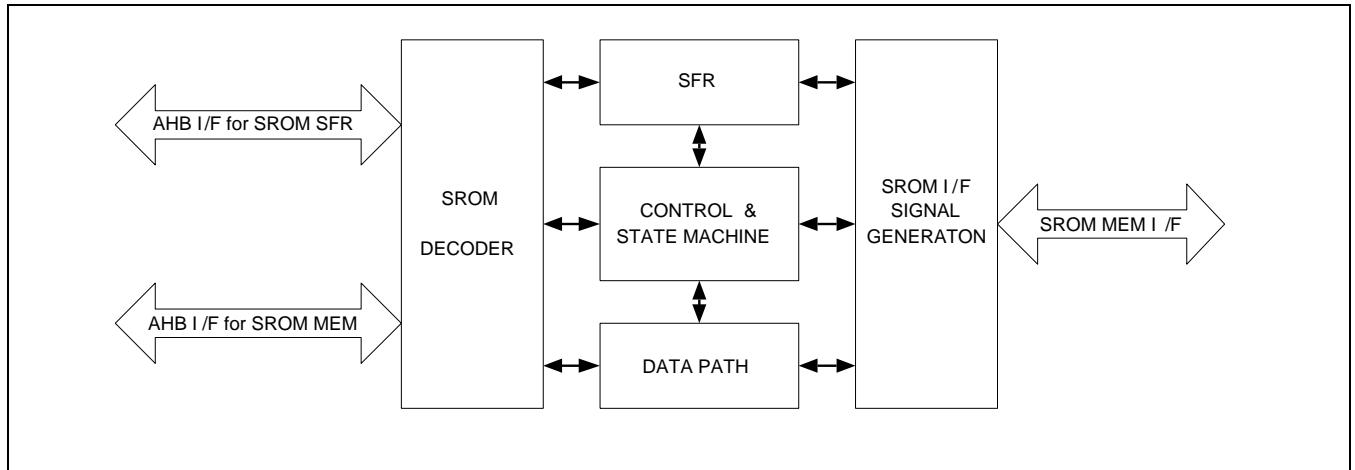


Figure 9-1 Block Diagram of SROM Controller Introduction

9.3.2 Interface Port Description

SROM Controller supports SROM interface for Bank0 to Bank3.

Signal	I/O	Description	Pad	Type
SROM_CSn[3:0]	Output	Bank selection signal	Xm0CSn_x	muxed
EBI_ADDR[15:0]	Output	SROM address bus	Xm0ADDR_x	muxed
EBI_OEn	Output	SROM output enable	Xm0OEn	muxed
EBI_WEn	Output	SROM write enable	Xm0WEn	muxed
EBI_BEn[1:0]	Output	SROM byte write enable/byte enable	Xm0BEn_x	muxed
EBI_DATA[15:0]	In/Out	SROM data bus	Xm0DATA_x	muxed
SROM_WAITn	Input	SROM wait input	Xm0WAITn	muxed

9.3.3 nWAIT Pin Operation

If the WAIT signal corresponding to each memory bank is enabled, the external nWAIT pin should prolong the duration of nOE while the memory bank is active. nWAIT is checked from tacc-1. nOE will be deasserted at the next clock after sampling nWAIT is high. The nWE signal has the same relation with nOE signal.

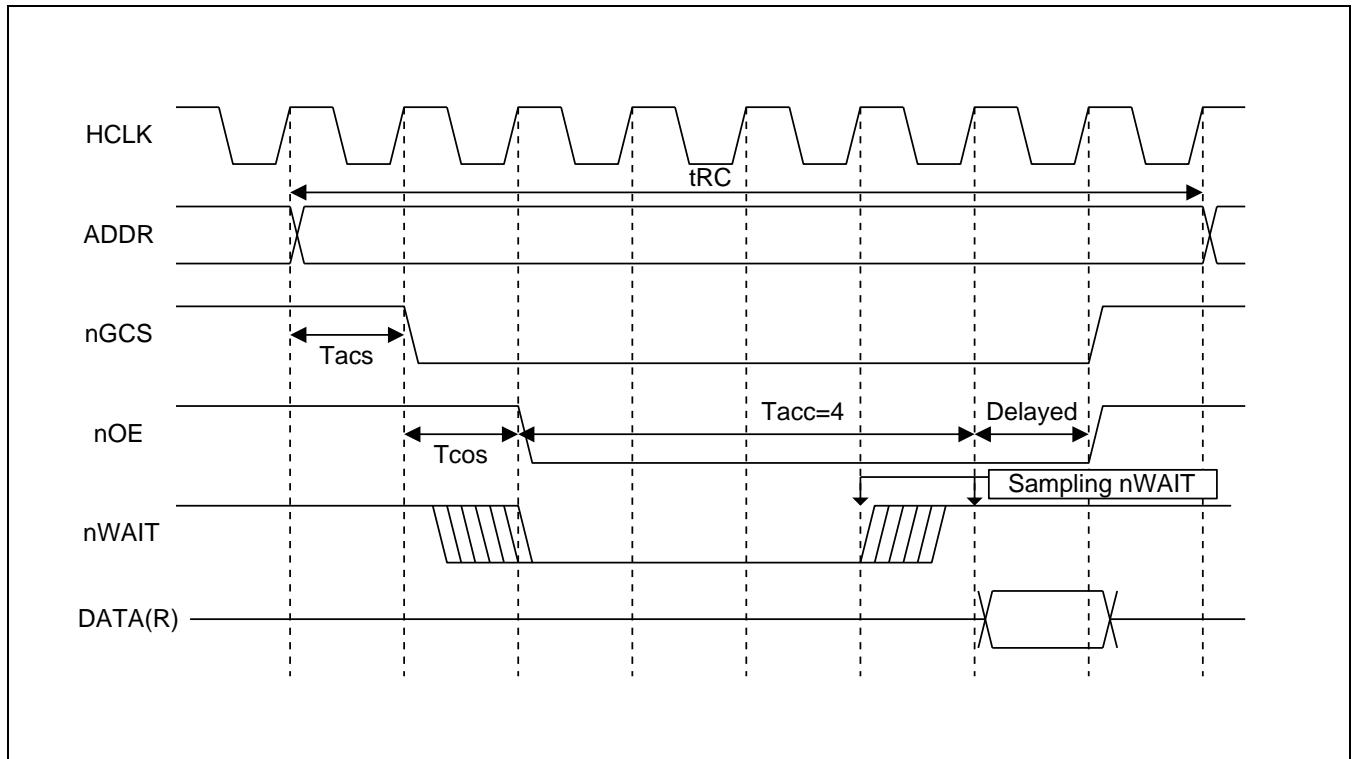


Figure 9-2 SROM Controller nWAIT Timing Diagram

9.3.4 Programmable Access Cycle

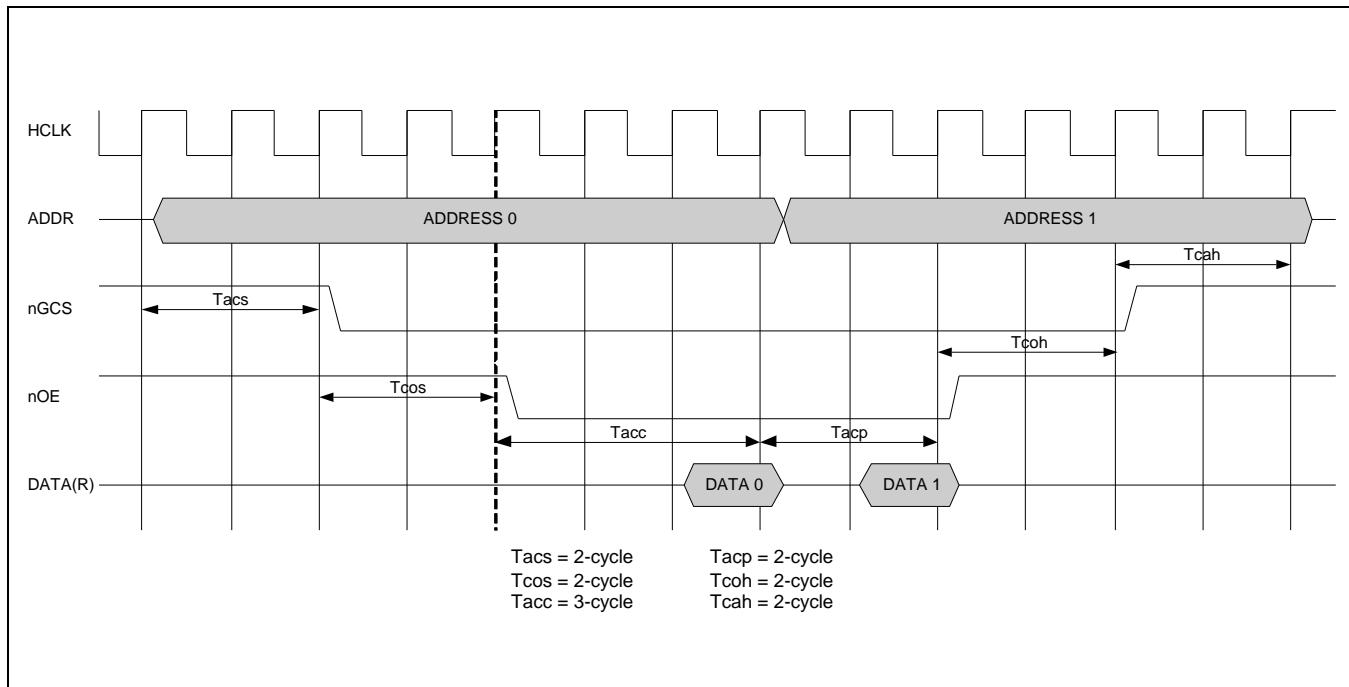


Figure 9-3 SROM Controller Read Timing Diagram

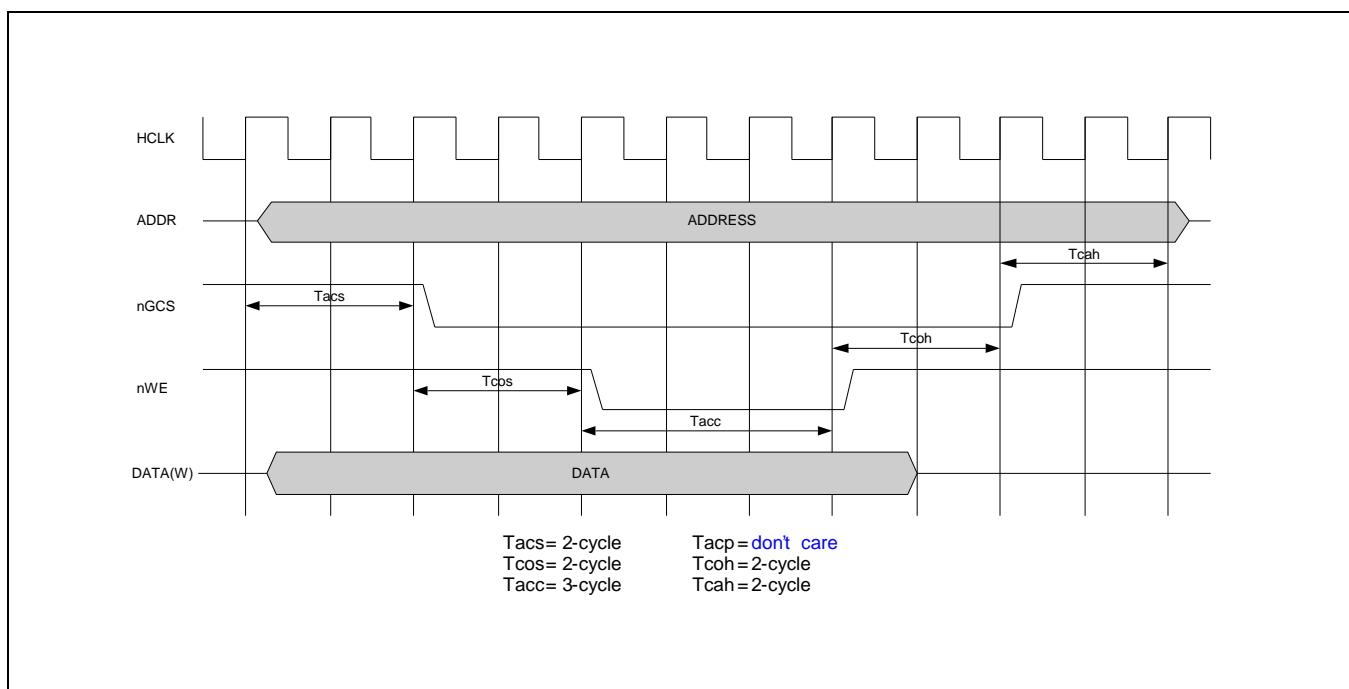


Figure 9-4 SROM Controller Write Timing Diagram

9.4 Register Description

9.4.1 Register Map Summary

- Base Address: 0x1257_0000

Register	Offset	Description	Reset Value
SROM_BW	0x0000	Specifies the SROM Bus width & wait control	0x0000_0009
SROM_BC0	0x0004	Specifies the SROM Bank 0 control register	0x000F_0000
SROM_BC1	0x0008	Specifies the SROM Bank 1 control register	0x000F_0000
SROM_BC2	0x000C	Specifies the SROM Bank 2 control register	0x000F_0000
SROM_BC3	0x0010	Specifies the SROM Bank 3 control register	0x000F_0000

9.4.1.1 SROM_BW

- Address = Base Address + 0x0000, Reset Value = 0x0000_0009

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	-	Reserved	0
ByteEnable3	[15]	RW	nWBE/ nBE (for UB/LB) control for Memory Bank3 0 = Not using UB/LB (XrnWBE[1:0] is dedicated nWBE[1:0]) 1 = Using UB/LB (XrnWBE[1:0] is dedicated nBE[1:0])	0
WaitEnable3	[14]	RW	Wait enable control for Memory Bank3 0 = Disables WAIT 1 = Enables WAIT	0
AddrMode3	[13]	RW	Select SROM ADDR Base for Memory Bank3 0 = SROM_ADDR is Half-word base address. (SROM_ADDR[15:0] ← HADDR[16:1]) 1 = SROM_ADDR is byte base address (SROM_ADDR[15:0] ← HADDR[15:0]) NOTE: When DataWidth3 is "0", SROM_ADDR is byte base address. (Ignored this bit.)	0
DataWidth3	[12]	RW	Data bus width control for Memory Bank3 0 = 8-bit 1 = 16-bit	0
ByteEnable2	[11]	RW	nWBE/ nBE (for UB/LB) control for Memory Bank2 0 = Not using UB/LB (XrnWBE[1:0] is dedicated nWBE[1:0]) 1 = Using UB/LB (XrnWBE[1:0] is dedicated nBE[1:0])	0
WaitEnable2	[10]	RW	Wait enable control for Memory Bank2 0 = Disables WAIT 1 = Enables WAIT	0
AddrMode2	[9]	RW	Select SROM ADDR Base for Memory Bank2 0 = SROM_ADDR is Half-word base address. (SROM_ADDR[15:0] ← HADDR[16:1]) 1 = SROM_ADDR is byte base address (SROM_ADDR[15:0] ← HADDR[15:0]) NOTE: When DataWidth2 is "0", SROM_ADDR is byte base address. (Ignored this bit.)	0
DataWidth2	[8]	RW	Data bus width control for Memory Bank2 0 = 8-bit 1 = 16-bit	0
ByteEnable1	[7]	RW	nWBE/ nBE (for UB/LB) control for Memory Bank1 0 = Not using UB/LB (XrnWBE[1:0] is dedicated nWBE[1:0]) 1 = Using UB/LB (XrnWBE[1:0] is dedicated nBE[1:0])	0
WaitEnable1	[6]	RW	Wait enable control for Memory Bank1 0 = Disables WAIT 1 = Enables WAIT	0
AddrMode1	[5]	RW	Select SROM ADDR Base for Memory Bank1 0 = SROM_ADDR is Half-word base address.	0

Name	Bit	Type	Description	Reset Value
			(SROM_ADDR[15:0] ← HADDR[16:1]) 1 = SROM_ADDR is byte base address (SROM_ADDR[15:0] ← HADDR[15:0]) NOTE: When DataWidth1 is "0", SROM_ADDR is byte base address. (Ignored this bit.)	
DataWidth1	[4]	RW	Data bus width control for Memory Bank1 0 = 8-bit 1 = 16-bit	0
ByteEnable0	[3]	RW	nWBE/ nBE (for UB/LB) control for Memory Bank0 0 = Not using UB/LB (XrnWBE[1:0] is dedicated nWBE[1:0]) 1 = Using UB/LB (XrnWBE[1:0] is dedicated nBE[1:0])	1
WaitEnable0	[2]	RW	Wait enable control for Memory Bank0 0 = Disables WAIT 1 = Enables WAIT	0
AddrMode0	[1]	RW	Select SROM ADDR Base for Memory Bank0 0 = SROM_ADDR is Half-word base address. (SROM_ADDR[15:0] ← HADDR[16:1]) 1 = SROM_ADDR is byte base address (SROM_ADDR[15:0] ← HADDR[15:0]) NOTE: When DataWidth0 is "0", SROM_ADDR is byte base address. (Ignored this bit.)	0
DataWidth0	[0]	RW	Data bus width control for Memory Bank0 0 = 8-bit 1 = 16-bit	1

9.4.1.2 SROM_BCn (n = 0 to 3)

- Address = Base Address + 0x0004, Reset Value = 0x000F_0000
- Address = Base Address + 0x0008, Reset Value = 0x000F_0000
- Address = Base Address + 0x000C, Reset Value = 0x000F_0000
- Address = Base Address + 0x0010, Reset Value = 0x000F_0000

Name	Bit	Type	Description	Reset Value
Tacs	[31:28]	RW	Address set-up before nGCS 0000 = 0 Clock 0001 = 1 Clocks 0010 = 2 Clocks 0011 = 3 Clocks 1100 = 12 Clocks 1101 = 13 Clocks 1110 = 14 Clocks 1111 = 15 Clocks NOTE: More 1 – 2 cycles according to bus i/f status	0000
Tcos	[27:24]	RW	Chip selection set-up before nOE 0000 = 0 Clock 0001 = 1 Clocks 0010 = 2 Clocks 0011 = 3 Clocks 1100 = 12 Clocks 1101 = 13 Clocks 1110 = 14 Clocks 1111 = 15 Clocks	0000
RSVD	[23:21]	-	Reserved	000
Tacc	[20:16]	RW	Access cycle 00000 = 1 Clock 00001 = 2 Clocks 00001 = 3 Clocks 00010 = 4 Clocks 11100 = 29 Clocks 11101 = 30 Clocks 11110 = 31 Clocks 11111 = 32 Clocks	01111
Tcoh	[15:12]	RW	Chip selection hold on nOE 0000 = 0 Clock 0001 = 1 Clocks 0010 = 2 Clocks 0011 = 3 Clocks 1100 = 12 Clocks 1101 = 13 Clocks	0000

Name	Bit	Type	Description	Reset Value
			1110 = 14 Clocks 1111 = 15 Clocks	
Tcah	[11:8]	RW	Address holding time after nGCSn 0000 = 0 Clock 0001 = 1 Clocks 0010 = 2 Clocks 0011 = 3 Clocks 1100 = 12 Clocks 1101 = 13 Clocks 1110 = 14 Clocks 1111 = 15 Clocks NOTE: More 1 – 2 cycles according to bus i/f status	0000
Tacp	[7:4]	RW	Page mode access cycle@Page mode 0000 = 0 Clock 0001 = 1 Clocks 0010 = 2 Clocks 0011 = 3 Clocks 1100 = 12 Clocks 1101 = 13 Clocks 1110 = 14 Clocks 1111 = 15 Clocks	0000
RSVD	[3:2]	-	Reserved	-
PMC	[1:0]	RW	Page mode configuration 00 = Normal (1 Data) 01 = 4 Data 10 = Reserved 11 = Reserved	00

10 NAND Flash Controller

10.1 Overview

Owing to the recent increase in the prices of NOR flash memory and the moderately priced DRAM and NAND flash, customers prefer to execute boot code on NAND flash and execute the main code on DRAM.

The boot code in Exynos4210 can be executed on external NAND flash. It will copy NAND flash data to DRAM. To validate the NAND flash data, Exynos4210 comprises of hardware Error Correction Code (ECC). After the NAND flash content is copied to DRAM, main program will be executed on DRAM. NAND flash controller uses ACLK_133 clock, which is from clock controller (FSYS).

10.2 Features

The key features of NAND flash controller include:

- Auto boot: The boot code is transferred to internal SRAM during reset. After the transfer, the boot code will be executed on the SRAM.
- NAND flash memory interface: Supports 512 Bytes, 2 KB, 4 KB, and 8 KB pages.
- Software mode: You can directly access NAND flash memory, for example, this feature can be used in read/erase/program NAND flash memory.
- Interface: Supports 8-bit NAND flash memory interface bus.
- Generates, detects, and indicates hardware ECC (Software correction).
- Supports both SLC and MLC NAND flash memories.
- ECC: Supports 1-/4-/8-/12-/16-bit ECC
- SFR interface: Supports byte/half word/word access to Data and ECC data registers, and Word access to other registers.

10.3 Functional Description

10.3.1 Block Diagram

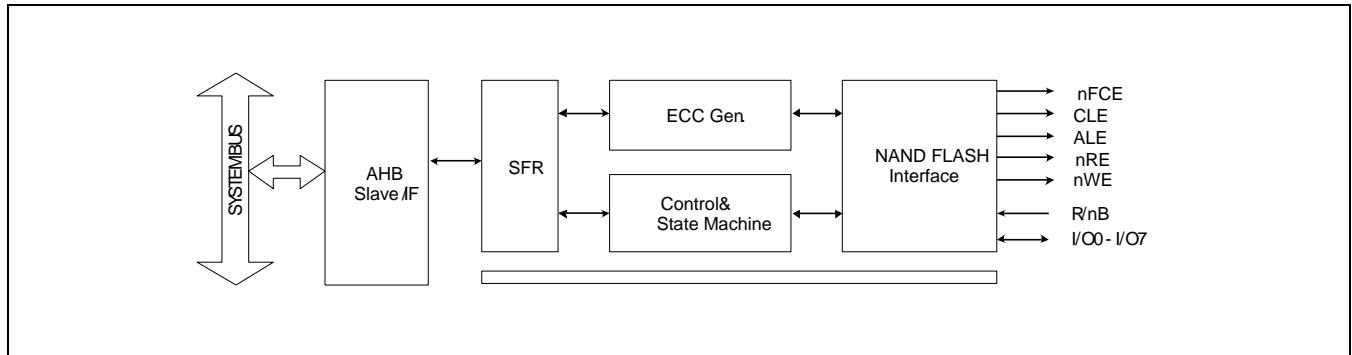


Figure 10-1 NAND Flash Controller Block Diagram

10.3.2 Interface Port Description

Signal	I/O	Description	Pad	Type
EBI_DATA[7:0]	Input/Output	Address/Data Bus	Xm0DATA[7:0]	muxed
NF_RnB[3:0]	Input	Ready and Busy	Xm0FRnB[3:0]	muxed
NF_CLE	Output	Command Latch Enable	Xm0FCLE	muxed
NF_ALE	Output	Address Latch Enable	Xm0FALE	muxed
NF_CSn[3:0]	Output	Chip Enable	Xm0CSn[3:0]	muxed
EBI_OEn	Output	Read Enable	Xm0OEn	muxed
EBI_Own	Output	Write Enable	Xm0WEn	muxed

10.3.3 NAND Flash Memory Timing

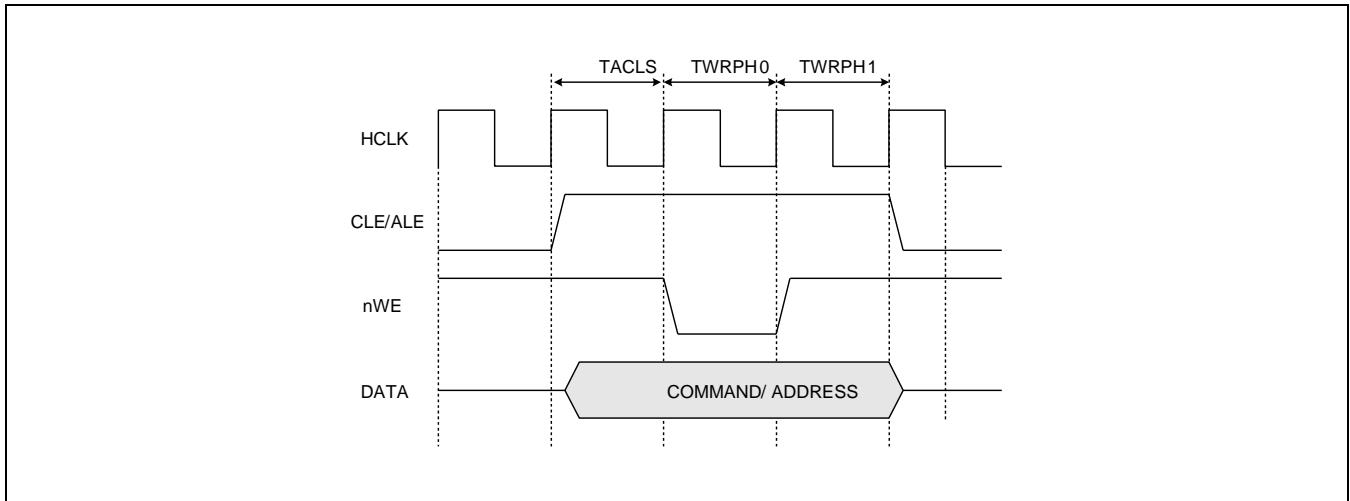


Figure 10-2 CLE and ALE Timing (TACLS = 1, TWRPH0 = 0, TWRPH1 = 0)

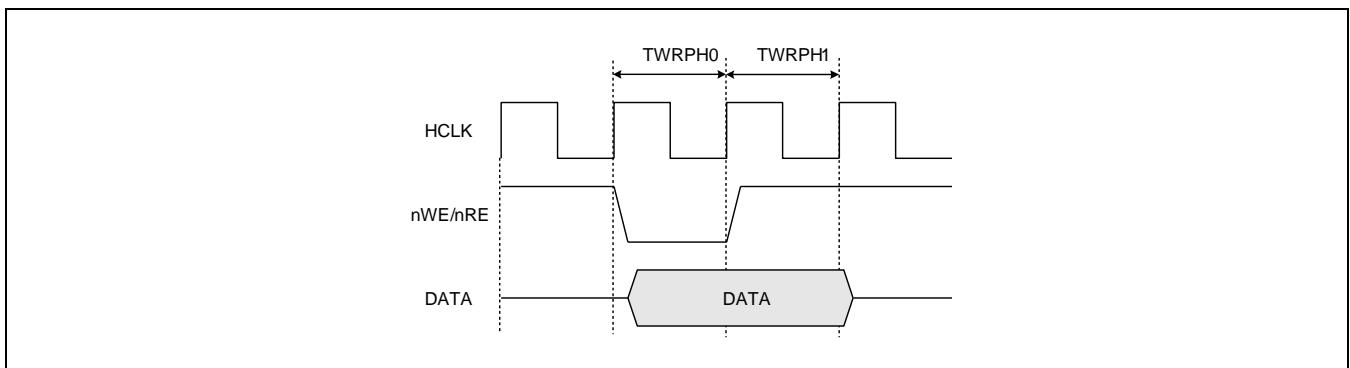


Figure 10-3 nWE and nRE Timing (TWRPH0 = 0, TWRPH1 = 0)

10.3.4 Software Mode

Exynos4210 supports only software mode access. Use this mode to access NAND flash memory. The NAND flash controller supports direct access to interface with the NAND flash memory.

- Writing to the command register (NFCMMD) specifies the NAND Flash Memory command cycle
- Writing to the address register (NFADDR) specifies the NAND Flash Memory address cycle
- Writing to the data register (NFDATA) specifies write data to the NAND Flash Memory (Write cycle)
- Reading from the data register (NFDATA) specifies read data from the NAND Flash Memory (Read cycle)
- Reading main ECC registers (NFMECCD0/NFMECCD1) and Spare ECC registers (NFSECCD) specify read data from the NAND Flash Memory

NOTE: In the software mode, use polling or interrupt to check the RnB status input pin.

10.3.4.1 Data Register Configuration

10.3.4.1.1 8-bit NAND Flash Memory Interface

- A. Word Access

Register	Endian	Bit [31:24]	Bit [23:16]	Bit [15:8]	Bit [7:0]
NFDATA	Little	4th I/O[7:0]	3rd I/O[7:0]	2nd I/O[7:0]	1st I/O[7:0]

- B. Half-word Access

Register	Endian	Bit [31:24]	Bit [23:16]	Bit [15:8]	Bit [7:0]
NFDATA	Little	Invalid value	Invalid value	2nd I/O[7:0]	1st I/O[7:0]

- C. Byte Access

Register	Endian	Bit [31:24]	Bit [23:16]	Bit [15:8]	Bit [7:0]
NFDATA	Little	Invalid value	Invalid value	Invalid value	1st I/O[7:0]

10.3.4.2 1-/4-/8-/12-/16-bit ECC

NAND flash controller supports 1-/4-/8-/12-/16-bit ECC.

For 1-bit ECC, NAND flash controller comprises ECC modules for main and spare (Meta) data. Main data ECC module generates ECC parity code for 2048 bytes (Maximum) data/message length, whereas spare (Meta) data ECC module generates ECC Parity code for 32 bytes (Maximum).

For 4-bit ECC, NAND flash controller comprises of an ECC module. It generates 512 or 24 bytes of ECC parity code. Set MsgLength (NFCONF[25]) to select 512 or 24 bytes message length.

For 8-/ 12-/ 16-bit ECC, NAND flash controller comprises ECC modules for each ECC. You can select data/message length for main and spare (Meta) data length. Usually, the length of main data is 512 bytes, and the length of spare (Meta) data depends on user application.

Since these ECC modules support variable length of main and spare (meta) data, you must set the ECC parity conversion codes to handle free page (For more information on ECC parity conversion codes, refer to [10.3.4.11 ECC Parity Conversion Code Guide for 8/12/16-bit ECC](#)). Free page specifies an erased page. The value of erased page is '0xff'. Therefore, set the ECC parity conversion codes to generate '0xff' ECC parity codes for all '0xff' data. This allows ECC module to detect errors on a free page.

ECC parity codes are described as follows:

- 28-bit ECC Parity Code = 22-bit Line parity + 6-bit Column Parity
- 10-bit ECC Parity Code = 4-bit Line parity + 6bit Column Parity

Each 1-/4-/8-/12-/16-bit ECC module guarantees up to 1-/4-/8-/12-/16-bit errors, respectively. If the errors cross the number of guaranteed errors, the result cannot be guaranteed.

[10.3.4.3 2048 Byte 1-bit ECC Parity Code Assignment Table](#)" and [10.3.4.4 32 Byte 1-bit ECC Parity Code Assignment Table](#) show 1-bit ECC parity code assignment.

10.3.4.3 2048 Byte 1-bit ECC Parity Code Assignment Table

	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0
MECCn_0	– P64	– P64'	– P32	– P32'	– P16	– P16'	– P8	– P8'
MECCn_1	– P1024	– P1024'	– P512	– P512'	– P256	– P256'	– P128	– P128'
MECCn_2	– P4	– P4'	– P2	– P2'	– P1	– P1'	– P2048	– P2048'
MECCn_3	1	1	1	1	– P8192	– P8192'	– P4096	– P4096'

10.3.4.4 32 Byte 1-bit ECC Parity Code Assignment Table

	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0
SECCn_0	– P2	– P2'	– P1	– P1'	– P16	– P16'	– P8	– P8'
SECCn_1	– P128	– P128`	– P64	– P64`	– P32	– P32`	– P4	– P4'

10.3.4.5 1-bit ECC Module Features

The ECC Lock (MainECClock and SpareECClock) bit of the control register generates the 1-bit ECC. If ECClock is low, the hardware ECC modules generate the ECC codes.

- 1-bit ECC Register Configuration

The following table shows the configuration of 1-bit ECC value read from spare area of external NAND flash memory. The format of ECC read from memory is important to compare the ECC parity code generated by the hardware modules.

NOTE: 4-bit/8-bit/12-bit/16-bit ECC decoding scheme is different compared to 1-bit ECC.

- NAND Flash Memory Interface

Register	Bit [31:24]	Bit [23:16]	Bit [15:8]	Bit [7:0]
NFMECCD0	Not used	2 nd ECC	Not used	1 st ECC
NFMECCD1	Not used	4 th ECC	Not used	3 rd ECC
NFSECCD	Not used	2 nd ECC	Not used	1 st ECC

10.3.4.6 1-bit ECC Programming Guide

1. To use SLC ECC in software mode, reset the ECCType to "0" (enable SLC ECC). ECC module generates ECC parity code for all read / write data when MainECClock(NFCON[7]) and SpareECClock(NFCON[6]) are unlocked ("0"). You must reset ECC value. To reset ECC value write the InitMECC(NFCONT[5]) and InitSECC(NFCON[4]) bit as "1" and clear the MainECClock(NFCONT[7]) bit to "0" (Unlock) before reading or writing data.
MainECClock(NFCONT[7]) and SpareECClock(NFCONT[6]) bits control whether ECC Parity code is generated or not.
2. The ECC module generates ECC parity code on register NFMECC0/1 whenever data is read or written.
3. After you complete reading or writing one page (Not including spare area data), set the MainECClock bit to "1" (Lock). ECC Parity code is locked and the value of the ECC status register does not change.
4. To generate spare area ECC parity code, Clear SpareECClock(NFCONT[6]) bit as "0" (Unlock).
5. The spare area ECC module generates ECC parity code on register NFSECC whenever data is read or written.
6. After you complete reading or writing spare area, set the SpareECClock bit to "1" (Lock). ECC Parity code is locked and the value of the ECC status register will not be changed.
7. From now on, you can use these values to record to the spare area or check the bit error.
8. For example, to check the bit error of main data area on page read operation, you must move the ECC parity codes (Is stored to spare area) to NFMECCD0 and NFMECCD1 after ECC codes for main data area is generated. From this point, the NFECCERR0 and NFECCERR1 have the valid error status values.

NOTE: NFSECCD is for ECC in the spare area (Usually, the user will write the ECC value generated from main data area to spare area, of which the value will be the same as NFMECC0/1) which is generated from the main data area.

10.3.4.7 4-bit ECC Programming Guide (ENCODING)

1. To use 4-bit ECC in software mode, set the MsgLength to 0 (512 byte message length) and the ECCType to "10" (enable 4-bit ECC). ECC module generates ECC parity code for 512 byte write data. To reset ECC value write the InitMECC(NFCONT[5]) bit as "1" and clear the MainECClock(NFCONT[7]) bit to "0" (Unlock) before writing data. MainECClock(NFCONT[7]) bit controls whether ECC Parity code is generated or not.
2. The 4-bit ECC module generates ECC parity code internally whenever data is written.
3. After you finish writing 512 byte data (not including spare area data), the parity codes are automatically updated to NFMECC0 and NFMECC1 registers. If you use 512 byte NAND Flash memory, you can program these values to spare area. However, if you use NAND Flash memory more than 512 byte page, you cannot program immediately. In this case, you have to copy these parity codes to other memory like DRAM. After writing all main data, you can write the copied ECC values to spare area.
The parity codes have self-correctable information including parity code itself.
4. To generate spare area ECC parity code, set the MsgLength to 1 (24 byte message length) and the ECCType to "10"(enable 4bit ECC). ECC module generates ECC parity code for 24 byte write data. To reset ECC value write the InitMECC(NFCONT[5]) bit as "1" and clear the MainECClock(NFCONT[7]) bit to '0'(Unlock) before writing data. MainECClock(NFCONT[7]) bit controls whether ECC Parity code is generated or not.
5. Whenever data is written, the 4-bit ECC module generates ECC parity code internally.
6. When you finish writing 24 byte meta or extra data, the parity codes are automatically updated to NFMECC0 and NFMECC1 registers. You can program these parity codes to spare area.
The parity codes have self-correctable information including parity code itself.

10.3.4.8 4-bit ECC Programming Guide (DECODING)

1. To use 4-bit ECC in software mode, set the MsgLength to 0 (512 byte message length) and the ECCType to "10" (enable 4-bit ECC). ECC module generates ECC parity code for 512 byte read data. Therefore, to reset ECC value write the InitMECC (NFCONT[5]) bit as "1" and clear the MainECClock(NFCONT[7]) bit to '0'(Unlock) before reading data.
MainECClock(NFCONT[7]) bit controls whether ECC Parity code is generated or not.
2. Whenever data is read, the 4-bit ECC module generates ECC parity code internally.
3. After you complete reading 512 byte (Not including spare area data), you must read parity codes. MLC ECC module needs parity codes to detect whether error bits have occurred or not. Therefore, you must read ECC parity code immediately after reading 512 byte. After ECC parity code is read, 4-bit ECC engine starts searching for error internally. 4-bit ECC error searching engine needs minimum of 155 cycles to find any error. During this time, you can continue reading main data from external NAND Flash memory.
Use ECCDecDone(NFSTAT[6]) to check whether ECC decoding is completed or not.
4. When ECCDecDone(NFSTAT[6]) is set ("1"), NFECCERR0 indicates whether error bit exists or not. If any error exists, refer NFECCERR0/1 and NFMLCBITPT registers to fix.
5. If you have more main data to read, go back to step 1.
6. To check meta data error, set the MsgLength to 1 (24 byte message length) and the ECCType to "1" (Enable 4-bit ECC). ECC module generates ECC parity code for 512 byte read data. Therefore, you must reset ECC value by writing the InitMECC(NFCONT[5]) bit as "1" and clear the MainECClock(NFCONT[7]) bit to "0" (Unlock) before reading data.
MainECClock(NFCONT[7]) bit controls whether ECC Parity code is generated or not.
7. Whenever data is read, the 4-bit ECC module generates ECC parity code internally.
8. After you complete reading 512 byte (Not include spare area data), you must read parity codes. 4-bit ECC module needs parity codes to detect whether error bits have occurred or not. Therefore, ensure to read ECC parity codes immediately after reading 512 byte. After ECC parity code is read, 4-bit ECC engine starts searching for error internally. 4-bit ECC error searching engine needs minimum of 155 cycles to find any error. During this time, you can continue reading main data from external NAND Flash memory. Use ECCDecDone(NFSTAT[6]) to check whether ECC decoding is completed or not.
9. When ECCDecDone(NFSTAT[6]) is set ("1"), NFECCERR0 indicates whether error bit exists or not. If any error exists, you can fix it by referring to NFECCERR0/1 and NFMLCBITPT registers.

10.3.4.9 8-bit/12-bit/16-Bit ECC Programming Guide (ENCODING)

1. To use 8/12/16-bit ECC in software mode, set the MsgLength(NFECCCONF[25:16]) to 511(512 byte message length) and the ECCType to "001/100/101" (Enable 8/12/16-bit ECC, respectively). ECC module generates ECC parity code for 512 byte write data. Therefore, reset ECC value by writing the InitMECC (NFECCCONT[2]) bit as "1" before writing data, and clear the MainECClock(NFCONT[7]) bit to '0'(Unlock) before writing data.
2. Whenever data is written, the corresponding 8/12/16-bit ECC module generates ECC parity code internally.
3. After you finish writing 512 byte data (This does not include spare area data), the parity codes are automatically updated to the NFECCPRG0 – NFECCPRGECC6 registers. If you use a NAND Flash memory having 512 byte page, you can program these values to spare area. However, if you use a NAND Flash memory more than 512 byte page, you cannot program immediately. In this case, you must copy these ECC parity codes to other memory like DRAM. After writing all main data, you can write the copied ECC values to spare area.

The parity codes have self-correctable information including parity code itself.

Table below shows the ECC parity size:

ECC type	Size of ECC Parity Codes
8-bit ECC	13 byte
12-bit ECC	20 byte
16-bit ECC	26 byte

4. To generate spare area ECC parity code for meta data, the steps are same (from 1 – 3), except setting the MsgLenght(NFECCCONF[25:16]) to the size that you prefer. When you set InitMECC(NFECCCONT[2]), all ECC parity codes generated for main data are cleared. Therefore, you should copy the ECC parity codes for main data.

NOTE: You should set the ECC parity conversion codes to check free page error. For more information about, refer to [10.3.4.11 ECC Parity Conversion Code Guide for 8/12/16-bit ECC](#).

10.3.4.10 8/12/16-bit ECC Programming Guide (DECODING)

1. To use 8/12/16-bit ECC in software mode, set the MsgLength(NFECCCONF[25:16] to 511 (512 byte message length) and the ECCType to "001/100/101" (enable 8/12/16-bit ECC, respectively). ECC module generates ECC parity code for 512 byte read data. Therefore, you must reset ECC value by writing the InitMECC (NFECCCONT[2]) bit as "1", and clear the MainECCLock(NFCONT[7]) bit to "0" (Unlock) before read data.
2. Whenever data is read, the 8/12/16-bit ECC module generates ECC parity code internally.
3. After you complete reading 512 byte (Not including spare area data), ensure to read the corresponding parity codes. ECC module needs parity codes to detect whether error bits have occurred or not. Therefore, you have to read ECC parity code immediately after reading 512 byte. After ECC parity code is read, the 8/12/16-bit ECC engine searches for error internally. 8/12/16-bit ECC search engine needs minimum of 155 cycles to find any errors. DecodeDone(NFECCSTAT[24]) can be used to check whether ECC decoding is completed or not.
4. When DecodeDone(NFECCSTAT[24]) is set ("1"), ECCError(NFECCSECSTAT[4:0]) indicates whether error bit exists or not. If any error exists, you can fix it by referencing NFECCERL0~NFECCERL7 and NFECCERP0 ~ NFECCERP3 registers.
5. If you have additional main data to read, continue the steps 1 to 4.
6. To check spare area data (Meta data) error, the sequences are same (Steps 1 to 4), except setting the MsgLength (NFECCCONF[25:16]) to the size that you want.

NOTE: You should set the ECC parity conversion codes to check free page error. For more information, refer to [10.3.4.11 ECC Parity Conversion Code Guide for 8/12/16-bit ECC](#).

10.3.4.11 ECC Parity Conversion Code Guide for 8/12/16-bit ECC

The ECC parity conversion codes are there to fix errors, which occur when reading a free page. Free page means the page erased. The 8/12/16-bit ECC modules support variable message size for meta data stored in spare area. Generally, the size of main data (Sector) is 512 byte and user should set the corresponding ECC parity conversion codes as shown in Table below.

ECC type	ECC Parity Conversion Codes
8-bit ECC	Here, 13 byte ECC parity conversion codes
12-bit ECC	Here, 20 byte ECC parity conversion codes
16-bit ECC	Here, 26 byte ECC parity conversion codes

The message size for meta data stored spare area can be different depending on user's needs. Hence, you can change the size of meta data by changing MsgLength(NFECCCONF[25:16]) and change ECC parity conversion codes.

Steps to know ECC parity conversion codes according to the size of message length:

1. Clear all ECC parity conversion registers (NFECCCONNECC0 – NFECCCONNECC6) as all zero.
2. Set all registers for page program.
3. Reset InitMECC(NFECCCONT[2] bit as "1".
4. Write '0xff' data as much as the size of meta data.
5. After you write data as MsgLength (NFECCCONF[25:16]), the EncodeDone(NFECCSTAT[25]) is set as "1" and generates the corresponding ECC parity codes.
6. Set ECC parity conversion registers as inverted values of ECC parity codes generated. For testing if these ECC parity conversion codes work well, repeat step 3 to 5. After you set ECC parity conversion codes, if the generated ECC parity code are all "0xff", then it is working correctly.

Constraints to support free page function:

1. Free page check is for only data area (512 byte)
2. If there is an error during reading a page erased (Free page), then free page engine indicates that the page is not free page.
3. To detect error(s) on free page, user should set corresponding conversion codes.

10.3.4.12 Lock Scheme for Data Protection

NFCON provides a lock scheme to protect data stored in external NAND Flash memories from malicious program.

For this scheme, the NFSBLK and NFEBLK registers are used to provide access control methods; only the memory area between NFSBLK and NFEBLK is erasable and programmable, but the read access is available to whole memory area.

This lock scheme is only available when you enable LockTight(NFCONT[17]) and LOCK(NFCONT[16]).

1. Unlock mode

In unlock mode, user can access whole NAND memory; there are no constraints to access memory.

2. Soft lock mode

In soft lock mode, you can access NAND block area between NFSBLK and NFEBLK.

When you try to program or erase the locked area, an illegal access error will occur
(NFSTAT[5] bit will be set).

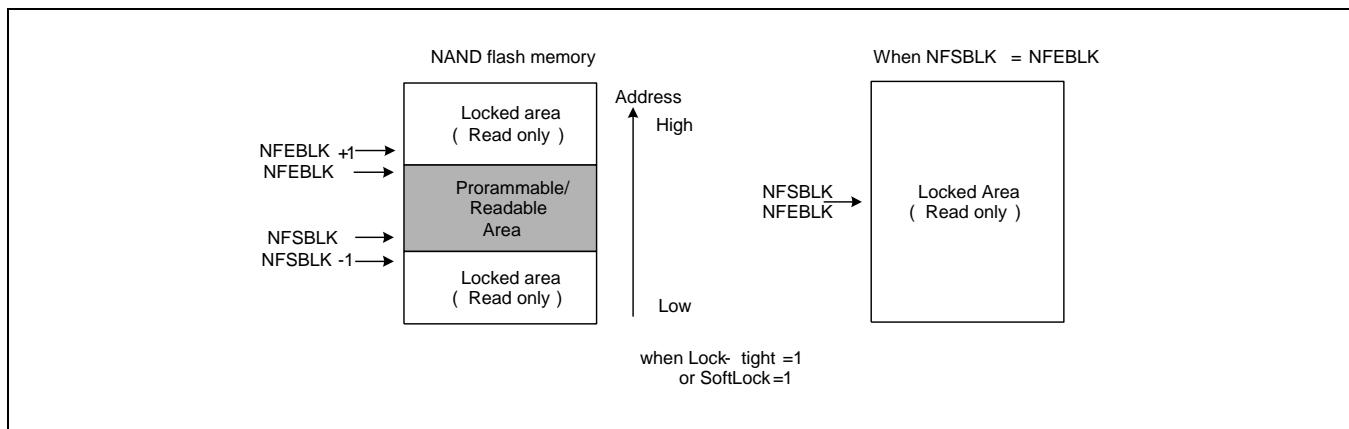
3. Lock-tight mode

In lock-tight mode, you can access NAND block area between NFSBLK and NFEBLK as soft lock mode. The differences is that you cannot change NFSBLK and NFEBLK registers, and also LockTight(NFCONT[17]) bits.

When you try to program or erase the locked area, an illegal access error will occur(NFSTAT[5] bit will be set).

The LockTight(NFCONT[17]) bit is only cleared when reset or wake up from sleep mode (It is impossible to clear it by software).

The accessibility of NAND area is illustrated in the figure below.



NOTE: If the address of NFSBLK and NFEBLK are same, then the erase and program to all NAND memory are not allowed.

10.3.5 Programming Constraints

NFCON has a constraint to access an external NAND memory. NFCON accesses NAND memory through EBI (External Bus Interface) which uses two different clocks source. The constraint occurs because EBI operates using OneNAND external interface clock and EBI interface between NFCON and EBI is handled as asynchronous interface so that a few clock latencies consume for bus handshaking. The clock of NFCON should be set lower than EBI internal operation clock. Please refer to the EBI and CMU (Clock Management Unit) manual.

10.4 Register Description

10.4.1 Register Map Summary

- Base Address: 0x0CE0_0000

Register	Offset	Description	Reset Value
1/ 4-bit ECC Register			
NFCONF	0x0000	Configuration Register	0x0000_1000
NFCONT	0x0004	Control Register	0x00C1_00C6
NFCMMD	0x0008	Command Register	0x0000_0000
NFADDR	0x000C	Address Register	0x0000_0000
NFDATA	0x0010	Data Register	0x0000_0000
NFMECCD0	0x0014	1 st and 2 nd Main ECC Data Register	0x0000_0000
NFMECCD1	0x0018	3 rd and 4 th Main ECC Data Register	0x0000_0000
NFSECCD	0x001C	Spare ECC Read Register	0xFFFF_FFFF
NFSBLK	0x0020	Programmable Start Block Address Register	0x0000_0000
NFEBLK	0x0024	Programmable End Block Address Register	0x0000_0000
NFSTAT	0x0028	NAND Status Register	0xF080_0F0D
NFECCERR0	0x002C	ECC Error Status0 Register	0x0003_FFF2
NFECCERR1	0x0030	ECC Error Status1 Register	0x0000_0000
NFMECC0	0x0034	Generated ECC Status0 Register	0xFFFF_FFFF
NFMECC1	0x0038	Generated ECC Status1 Register	0xFFFF_FFFF
NFSECC	0x003C	Generated Spare Area ECC Status Register	0xFFFF_FFFF
NFMLCBITPT	0x0040	4-bit ECC Error Bit Pattern Register	0x0000_0000

- Base Address: 0x0CE2_0000

Register	Offset	Description	Reset Value
8/ 12/ 16-bit ECC Register			
NFECCCONF	0x0000	ECC Configuration Register	0x0000_0000
NFECCCONT	0x0020	ECC Control Register	0x0000_0000
NFECCSTAT	0x0030	ECC Status Register	0x0000_0000
NFECCSECSTAT	0x0040	ECC Sector Status Register	0x0000_0000
NFECCPRGECC0	0x0090	ECC Parity Code0 Register for Page program	0x0000_0000
NFECCPRGECC1	0x0094	ECC Parity Code1 Register for Page Program	0x0000_0000
NFECCPRGECC2	0x0098	ECC parity code2 register for page program	0x0000_0000
NFECCPRGECC3	0x009C	ECC parity code3 register for page program	0x0000_0000
NFECCPRGECC4	0x00A0	ECC parity code4 register for page program	0x0000_0000
NFECCPRGECC5	0x00A4	ECC parity code5 register for page program	0x0000_0000
NFECCPRGECC6	0x00A8	ECC parity code6 register for page program	0x0000_0000
NFECCERL0	0x00C0	ECC error byte location0 register	0x0000_0000
NFECCERL1	0x00C4	ECC error byte location1 register	0x0000_0000
NFECCERL2	0x00C8	ECC error byte location2 register	0x0000_0000
NFECCERL3	0x00CC	ECC error byte location3 register	0x0000_0000
NFECCERL4	0x00D0	ECC error byte location4 register	0x0000_0000
NFECCERL5	0x00D4	ECC error byte location5 register	0x0000_0000
NFECCERL6	0x00D8	ECC error byte location6 register	0x0000_0000
NFECCERL7	0x00DC	ECC error byte location7 register	0x0000_0000
NFECCERP0	0x00F0	ECC error bit pattern0 register	0x0000_0000
NFECCERP1	0x00F4	ECC error bit pattern1 register	0x0000_0000
NFECCERP2	0x00F8	ECC error bit pattern2 register	0x0000_0000
NFECCERP3	0x00FC	ECC error bit pattern3 register	0x0000_0000
NFECCONECC0	0x0110	ECC parity conversion code0 register	0x0000_0000
NFECCONECC1	0x0114	ECC parity conversion code1 register	0x0000_0000
NFECCONECC2	0x0118	ECC parity conversion code2 register	0x0000_0000
NFECCONECC3	0x011C	ECC parity conversion code3 register	0x0000_0000
NFECCONECC4	0x0120	ECC parity conversion code4 register	0x0000_0000
NFECCONECC5	0x0124	ECC parity conversion code5 register	0x0000_0000
NFECCONECC6	0x0128	ECC parity conversion code6 register	0x0000_0000

10.4.2 Nand Flash Interface and 1/4-bit ECC Registers

10.4.2.1 NFCONF

- Address = 0x0CE0_0000, Reset Value = 0x0000_1000

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	–	Reserved	0
MsgLength	[25]	RW	0 = 512 byte Message Length 1 = 24 byte Message Length	0
ECCType0	[24:23]	RW	This bit indicates the kind of ECC to use. 00 = 1-bit ECC 10 = 4-bit ECC 01 = 11 = Disable 1-bit and 4-bit ECC	0
RSVD	[22:16]	–	Reserved	0000000
TACLS	[15:12]	RW	CLE and ALE duration setting value (0 – 15) Duration = HCLK × TACLS	0x1
TWRPH0	[11:8]	RW	TWRPH0 duration setting value (0 – 15) Duration = HCLK × (TWRPH0 + 1) NOTE: You should add additional cycles about 10ns for page read because of additional signal delay on PCB pattern.	0x0
TWRPH1	[7:4]	RW	TWRPH1 duration setting value (0 – 15) Duration = HCLK × (TWRPH1 + 1)	0x0
PageSize	[3:2]	RW	This bit indicates the page size of NAND Flash Memory. 00 = 2048 Byte 01 = 512 Byte 10 = 4096 Byte 11 = 2048 Byte NOTE: This bit is only for 1-bit ECC. The message length of 1-bit ECC is determined using this, not MsgLength (NFCONF[25] field. NFCON doesn't care about the actual page size of external NAND; the page size is handled by software.	0
AddrCycle	[1]	RW	This bit indicates the number of Address cycle of NAND Flash memory. When Page Size is 512 Bytes, 0 = 3 address cycle 1 = 4 address cycle When page size is 2K or 4K, 0 = 4 address cycle 1 = 5 address cycle NOTE: It is only used for Lock scheme, please refer to the chapter "3.11 Lock Scheme For Data Protecting".	0
RSVD	[0]	–	Reserved	0

10.4.2.2 NFCONT

- Address = 0x0CE0_0004, Reset Value = 0x00C1_00C6

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	—	Reserved	0
Reg_nCE3	[23]	RW	NAND Flash Memory nRCS[3] signal control. 0 = Force nRCS[3] to low (Enable chip select) 1 = Force nRCS[3] to High (Disable chip select)	1
Reg_nCE2	[22]	RW	NAND Flash Memory nRCS[2] signal control. 0 = Force nRCS[2] to low (Enable chip select) 1 = Force nRCS[2] to High (Disable chip select)	1
RSVD	[21:19]	—	Reserved	0
MLCEccDirection	[18]	RW	4-bit, ECC encoding/decoding control. 0 = Decoding 4-bit ECC, It is used for page read 1 = Encoding 4-bit ECC, It is be used for page program	0
LockTight	[17]	RW	Lock-tight configuration. 0 = Disable lock-tight 1 = Enable lock-tight If this bit is set to 1, you cannot clear this bit. For more information, refer to 10.3.4.12 Lock Scheme for Data Protection .	0
LOCK	[16]	RW	Soft Lock configuration. 0 = Disable lock 1 = Enable lock Software can modify soft lock area any time. For more information, refer to 10.3.4.12 Lock Scheme for Data Protection .	1
RSVD	[15:14]	—	Reserved	00
EnbMLCEncInt	[13]	RW	4-bit ECC encoding completion interrupt control. 0 = Disable interrupt 1 = Enable interrupt	0
EnbMLCDecInt	[12]	RW	4-bit ECC decoding completion interrupt control. 0 = Disable interrupt 1 = Enable interrupt	0
RSVD	[11]	—	Reserved	0
EnbIllegalAccINT	[10]	RW	Illegal access interrupt control. 0 = Disable interrupt 1 = Enable interrupt Illegal access interrupt occurs when CPU tries to program or erase locking area (the area setting in NFSBLK (0xB0E0_0020) to NFEBLK (0xB0E0_0024)-1).	0
EnbRnBINT	[9]	RW	RnB status input signal transition interrupt control. 0 = Disable RnB interrupt 1 = Enable RnB interrupt	0

Name	Bit	Type	Description	Reset Value
RnB_TransMode	[8]	RW	RnB transition detection configuration. 0 = Detect rising edge 1 = Detect falling edge	0
MECCLock	[7]	RW	Lock Main area ECC generation. 0 = Unlock Main area ECC 1 = Lock Main area ECC Main area ECC status register is NFMECC0/NFMECC1(0xB0E0_0034/0xB0E0_0038),	1
SECCLock	[6]	RW	Lock Spare area ECC generation. 0 = Unlock Spare ECC 1 = Lock Spare ECC Spare area ECC status register is NFSECC(0xB0E0_003C)	1
InitMECC	[5]	RW	1 = Initialize main area ECC decoder/encoder (write-only)	0
InitSECC	[4]	RW	1 = Initialize spare area ECC decoder/encoder (write-only)	0
HW_nCE	[3]	RW	Reserved (HW_nCE)	0
Reg_nCE1	[2]	RW	NAND Flash Memory nRCS[1] signal control.	1
Reg_nCE0	[1]	RW	NAND Flash Memory nRCS[0] signal control 0 = Force nRCS[0] to low (Enable chip select) 1 = Force nRCS[0] to High (Disable chip select) Note: The setting all nCE[3:0] zero can not be allowed. Only one nCE can be asserted to enable external NAND flash memory. The lower bit has more priority when user set all nCE[3:0] zeros.	1
MODE	[0]	RW	NAND Flash controller operating mode. 0 = Disable NAND Flash Controller. 1 = Enable NAND Flash Controller.	0

10.4.2.3 NFCMMD

- Address = 0x0CE0_0008, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x000000
REG_CMMMD	[7:0]	RW	NAND Flash memory command value.	0x00

10.4.2.4 NFADDR

- Address = 0x0CE0_000C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	-	Reserved	0x000000
REG_ADDR	[7:0]	RW	NAND Flash memory address value	0x00

10.4.2.5 NFDATA

- Address = 0x0CE0_0010, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
NFDATA	[31:0]	RW	NAND Flash read/ program data value for I/O. NOTE: For more information, refer to 10.3.4.1 Data Register Configuration .	0x00000000

10.4.2.6 NFMECCD

- Address = 0x0CE0_0014, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	-	Reserved	0x00
ECCData1 (ECC1)	[23:16]	RW	2nd ECC. NOTE: In software mode, read this register when you need to read 2nd ECC value from NAND Flash memory	0x00
RSVD	[15:8]	-	Reserved	0x00
ECCData0 (ECC0)	[7:0]	RW	1st ECC. NOTE: In software mode, read this register when you need to read 1st ECC value from NAND flash memory. This register has the same read function as NFDATA.	0x00

NOTE: Only word access is allowed.

10.4.2.7 NFMECCD1

- Address = 0x0CE0_0018, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	—	Reserved	0x00
ECCData3 (ECC3)	[23:16]	RW	4th ECC. NOTE: In software mode, read this register when you need to read 4th ECC value from NAND Flash memory	0x00
RSVD	[15:8]	—	Reserved	0x00
ECCData2 (ECC2)	[7:0]	RW	3rd ECC. NOTE: In software mode, read this register when you need to read 3rd ECC value from NAND Flash memory. This register has the same read function as NFDATA.	0x00

10.4.2.8 NFSECCD

- Address = 0x0CE0_001C, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	—	Reserved	0x00
SECCData1	[23:16]	RW	2nd ECC. NOTE: In software mode, read this register when you need to read 2nd ECC value from NAND Flash memory	0xFF
RSVD	[15:8]	—	Reserved	0x00
SECCData0	[7:0]	RW	1st ECC. NOTE: In software mode, read this register when you need to read 1st ECC value from NAND Flash memory. This register has the same read function as NFDATA.	0xFF

NOTE: Only word access is allowed.

10.4.2.9 NFSBLK

- Address = 0x0CE0_0020, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	—	Reserved	0x00
SBLK_ADDR2	[23:16]	RW	The 3rd block address of the block erase operation.	0x00
SBLK_ADDR1	[15:8]	RW	The 2nd block address of the block erase operation.	0x00
SBLK_ADDR0	[7:0]	RW	The 1st block address of the block erase operation. (Only bit [7:5] are valid).	0x00

NOTE: Advance Flash's block Address start from 3-address cycle. So block address register only needs 3 bytes. For more information about lock scheme, refer to [10.3.4.12 Lock Scheme for Data Protection](#).

10.4.2.10 NFEBLK

- Address = 0x0CE0_0024, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	—	Reserved	0x00
Eblk_ADDR2	[23:16]	RW	The 3 rd block address of the block erase operation.	0x00
Eblk_ADDR1	[15:8]	RW	The 2 nd block address of the block erase operation.	0x00
Eblk_ADDR0	[7:0]	RW	The 1 st block address of the block erase operation. (Only bit [7:5] are valid).	0x00

NOTE: Advance Flash's block Address start from 3-address cycle. So block address register only needs 3-bytes.

For more information about lock scheme, refer to [10.3.4.12 Lock Scheme for Data Protection](#).

10.4.2.11 NFSTAT

- Address = 0x0CE0_0028, Reset Value = 0xF080_0F0D

Name	Bit	Type	Description	Reset Value
Flash_RnB_GRP	[31:28]	RW	The status of RnB[3:0] input pin. 0 = NAND Flash memory busy 1 = NAND Flash memory ready to operate	0xF
RnB_TransDetect_GRP	[27:24]	RW	When RnB[3:0] low to high transition occurs, this bit is set and an interrupt is issued if RnB_TransDetect_GRP is enabled. To clear this, write "1". 0 = RnB transition is not detected 1 = RnB transition is detected Transition configuration is set in RnB_TransMode(NFCONT[8]).	—
RSVD	[23:12]	—	Reserved	0x800
Flash_nCE[3:0] (Read-only)	[11:8]	RW	The status of nCE[3:0] output pin.	0xF
MLCEncodeDone	[7]	RW	When 4-bit ECC encoding is finished, this bit is set and an interrupt is issued if MLCEncodeDone is enabled. The NFMLCECC0 and NFMLCECC1 have valid values. To clear this, write "1". 1 = 4-bit ECC encoding is completed	0
MLCDecodeDone	[6]	RW	When 4-bit ECC decoding is finished, this bit is set and an interrupt is issued if MLCDecodeDone is enabled. The NFMLCBITPT, NFMLCL0, and NFMLCEL1 have valid values. To clear this, write "1". 1 = 4-bit ECC decoding is completed	0
IllegalAccess	[5]	RW	Once Soft Lock or Lock-tight is enabled and any illegal access (program, erase) to the memory takes place, then this bit is set. 0 = Illegal access is not detected 1 = Illegal access is detected	0

Name	Bit	Type	Description	Reset Value
			To clear this value, write 1 to this bit.	
RnB_TransDetect	[4]	RW	When RnB[0] low to high transition occurs, this bit is set and an interrupt is issued if RnB_TransDetect is enabled. To clear this, write "1". 0 = RnB transition is not detected 1 = RnB transition is detected Transition configuration is set in RnB_TransMode(NFCONT[8]).	0
Flash_nCE[1] (Read-only)	[3]	RW	The status of nCE[1] output pin	1
Flash_nCE[0] (Read-only)	[2]	RW	The status of nCE[0] output pin	1
RSVD	[1]	-	Reserved	0
Flash_RnB (Read-only)	[0]	RW	The status of RnB[0] input pin. 0 = NAND Flash memory busy 1 = NAND Flash memory ready to operate	1

10.4.2.12 NFECCERR0

- Address = 0x0CE0_002C, Reset Value = 0x0003_FFF2

When ECC Type is 1-bit ECC

Name	Bit	Type	Description	Reset Value
RSVD	[31:25]	–	Reserved	0x00
ECCSDataAddr	[24:21]	R	In spare area, Indicates which number data is error	0x0
ECCSBitAddr	[20:18]	R	In spare area, Indicates which bit is error	000
ECCDataAddr	[17:7]	R	In main data area, Indicates which number data is error	0x7FF
ECCBitAddr	[6:4]	R	In main data area, Indicates which bit is error	111
ECCSprErrNo	[3:2]	R	Indicates whether spare area bit fail error occurred 00 = No Error 01 = 1-bit error (Correctable) 10 = Multiple error 11 = ECC area error	00
ECCMainErrNo	[1:0]	R	Indicates whether main data area bit fail error occurred 00 = No Error 01 = 1-bit error (Correctable) 10 = Multiple error 11 = ECC area error	10

NOTE: The above values are valid only when both ECC register and ECC status register have valid value.

When ECC Type is 4-bit ECC

Name	Bit	Type	Description	Reset Value
MLCECCBusy	[31]	R	Indicates the 4-bit ECC decoding engine is searching whether a error exists or not 0 = Idle, 1 = Busy	0
MLCECCReady	[30]	R	ECC Ready bit	1
MLCFreePage	[29]	R	Indicates the page data read from NAND flash has all "FF" value.	0
MLCECCError	[28:26]	R	4-bit ECC decoding result 000 = No error 001 = 1-bit error 010 = 2-bit error 011 = 3-bit error 100 = 4-bit error 101 = Uncorrectable 11x = reserved	000
MLCErrLocation2	[25:16]	R	Error byte location of 2nd bit error	0x000
RSVD	[15:10]	–	Reserved	0x00
MLCErrLocation1	[9:0]	R	Error byte location of 1st bit error	0x000

NOTE: These values are updated when ECCDecodeDone (NFSTAT[6]) is set ("1").

10.4.2.13 NFECCERR1

- Address = 0x0CE0_0030, Reset Value = 0x0000_0000

When ECC Type is 4-bit ECC

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	–	Reserved	0x00
MLCERRLocation4	[25:16]	R	Error byte location of 4th bit error.	0x00
RSVD	[15:10]	–	Reserved	0x00
MLCERRLocation3	[9:0]	R	Error byte location of 3rd bit error.	0x000

NOTE: These values are updated when ECCDecodeDone (NFSTAT[6]) is set ("1").

10.4.2.14 NFMECC0

- Address = 0x0CE0_0034, Reset Value = 0xFFFF_FFFF

When ECC Type is 1-bit ECC

Name	Bit	Type	Description	Reset Value
MECC3	[31:24]	R	ECC3 for data	0xFF
MECC2	[23:16]	R	ECC2 for data	0xFF
MECC1	[15:8]	R	ECC1 for data	0xFF
MECC0	[7:0]	R	ECC0 for data	0xFF

NOTE: The NAND flash controller generate NFMECC0/1 when read or write main area data while the MainECCLock(NFCONT[7]) bit is "0"(Unlock).

When ECC Type is 4-bit ECC

Name	Bit	Type	Description	Reset Value
4th Parity	[31:24]	R	4 th Check Parity generated from main area (512-byte)	0x00
3rd Parity	[23:16]	R	3 rd Check Parity generated from main area (512-byte)	0x00
2nd Parity	[15:8]	R	2 nd Check Parity generated from main area (512-byte)	0x00
1st Parity	[7:0]	R	1 st Check Parity generated from main area (512-byte)	0x00

NOTE: The NAND flash controller generate these ECC parity codes when write main area data while the MainECCLock(NFCON[7]) bit is "0"(unlock).

10.4.2.15 NFMECC1

- Address = 0x0CE0_0038, Reset Value = 0xFFFF_FFFF

When ECC Type is 4-bit ECC

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	—	Reserved	0x00
7th Parity	[23:16]	R	7 th Check Parity generated from main area (512 byte)	0x00
6th Parity	[15:8]	R	6 th Check Parity generated from main area (512 byte)	0x00
5th Parity	[7:0]	R	5 th Check Parity generated from main area (512 byte)	0x00

NOTE: The NAND flash controller generate these ECC parity codes when write main area data while the MainECCLock(NFCON[7]) bit is "0"(unlock).

10.4.2.16 NFSECC

- Address = 0x0CE0_003C, Reset Value = 0xFFFF_FFFF

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	-	Reserved	0xFFFF
SECC1	[15:8]	R	Spare area ECC1 Status	0xFF
SECC0	[7:0]	R	Spare area ECC0 Status	0xFF

NOTE: The NAND flash controller generate NFSECC when read or write spare area data while the SpareECClock(NFCONT[6]) bit is "0"(Unlock).

10.4.2.17 NFMLCBITPT

- Address = 0x0CE0_0040, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
4th Error bit pattern	[31:24]	R	4 th Error bit pattern	0x00
3rd Error bit pattern	[23:16]	R	3 rd Error bit pattern	0x00
2nd Error bit pattern	[15:8]	R	2 nd Error bit pattern	0x00
1st Error bit pattern	[7:0]	R	1st Error bit pattern	0x00

10.4.3 ECC Registers for 8, 12 and 16-bit ECC

10.4.3.1 NFECCCONF

- Address = 0x0CE2_0000, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31]	—	Reserved	0
RSVD	[28]	—	Reserved	0
MsgLength	[25:16]	RW	The ECC message size. For 512 byte message, you should set 511.	
RSVD	[15:4]	—	Reserved	0
ECCType	[3:0]	RW	These bits indicate what kind of ECC is used. 000 = Disable 8/ 12/ 16-bit ECC 001 = Reserved 010 = Reserved 011 = 8-bit ECC/512B 100 = 12-bit ECC 101 = 16-bit ECC/512B 110 = Reserved 111 = Reserved	0x0

10.4.3.2 NFECCCONT

- Address = 0x0CE2_0020, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	—	Reserved	0x00
EnbMLCEnclnt	[25]	RW	MLC ECC encoding completion interrupt control 0 = Disable interrupt 1 = Enable interrupt	0
EnbMLCDecInt	[24]	RW	MLC ECC decoding completion interrupt control 0 = Disable interrupt 1 = Enable interrupt	0
EccDirection	[16]	RW	MLC ECC encoding/decoding control 0 = Decoding, used for page read 1 = Encoding, used for page program	0
RSVD	[15:3]	—	Reserved	0x0
InitMECC	[2]	RW	1 = Initialize main area ECC decoder/encoder (write-only)	0
RSVD	[1]	—	Reserved	0
ResetECC	[0]	RW	1 = Reset ECC logic. (Write-only)	0

10.4.3.3 NFECCSTAT

- Address = 0x0CE2_0030, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
ECCBusy	[31]		Indicates the 8-bit ECC decoding engine is searching whether a error exists or not 0 = Idle 1 = Busy	0
RSVD	[30]	-	Reserved	1
EncodeDone	[25]		When MLC ECC encoding is finished, this value set and issue interrupt if EncodeDone is enabled. The NFMILCECC0 and NFMILCECC1 have valid values. To clear this, write "1". 1 = MLC ECC encoding is completed	0
DecodeDone	[24]		When MLC ECC decoding is finished, this value set and issue interrupt if DecodeDone is enabled. The NFMILCBITPT, NFMLCL0 and NFMLCEL1 have valid values. To clear this, write "1". 1 = MLC ECC decoding is completed	0
RSVD	[23:9]	-	Reserved	0x0000
FreePageStat	[8]		It indicates whether the sector is free page or not.	0
RSVD	[7:0]	-	Reserved	0x00

10.4.3.4 NFECCSECSTAT

- Address = 0x0CE2_0040, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
ValdErrorStat	[31:8]	R	Each bit indicates which ERL and ERP is valid or not.	0x0000_0000
ECCErrorNo	[4:0]	R	ECC decoding result when page read 00000 = No error 00001 = 1-bit error 00010 = 2-bit error 00011 = 3-bit error 01110 = 14-bit error 01111 = 15-bit error 10000 = 16-bit error NOTE: If 8-bit ECC is used, the valid number of error is until 8. If the number exceeds the supported error number, it means that uncorrectable error occurs.	0x00

10.4.3.5 NFECCPRGECCn

- Address = 0x0CE2_0090, Reset Value = 0x0000_0000
- Address = 0x0CE2_0094, Reset Value = 0x0000_0000
- Address = 0x0CE2_0098, Reset Value = 0x0000_0000
- Address = 0x0CE2_009C, Reset Value = 0x0000_0000
- Address = 0x0CE2_00A0, Reset Value = 0x0000_0000
- Address = 0x0CE2_00A4, Reset Value = 0x0000_0000
- Address = 0x0CE2_00A8, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
4 th Parity	[31:24]	R	4 th Check Parity for page program from main area	0x00
3 rd Parity	[23:16]	R	3 rd Check Parity for page program from main area	0x00
2 nd Parity	[15:8]	R	2 nd Check Parity for page program from main area	0x00
1 st Parity	[7:0]	R	1 st Check Parity for page program from main area	0x00
8 th Parity	[31:24]	R	8 th Check Parity generated from main area	0x00
7 th Parity	[23:16]	R	7 th Check Parity generated from main area	0x00
6 th Parity	[15:8]	R	6 th Check Parity generated from main area	0x00
5 th Parity	[7:0]	R	5 th Check Parity generated from main area	0x00
12 th Parity	[31:24]	R	12 th Check Parity generated from main area	0x00
11 th Parity	[23:16]	R	11 th Check Parity generated from main area	0x00
10 th Parity	[15:8]	R	10 th Check Parity generated from main area	0x00
9 th Parity	[7:0]	R	9 th Check Parity generated from main area	0x00
16 th Parity	[31:24]	R	16 th Check Parity generated from main area	0x00
15 th Parity	[23:16]	R	15 th Check Parity generated from main area	0x00
14 th Parity	[15:8]	R	14 th Check Parity generated from main area	0x00
13 th Parity	[7:0]	R	13 th Check Parity generated from main area	0x00
20 th Parity	[31:24]	R	20 th Check Parity generated from main area	0x00
19 th Parity	[23:16]	R	19 th Check Parity generated from main area	0x00
18 th Parity	[15:8]	R	18 th Check Parity generated from main area	0x00
17 th Parity	[7:0]	R	17 th Check Parity generated from main area	0x00
24 th Parity	[31:24]	R	24 th Check Parity generated from main area	0x00
23 rd Parity	[23:16]	R	23 rd Check Parity generated from main area	0x00
22 th Parity	[15:8]	R	22 th Check Parity generated from main area	0x00
21 th Parity	[7:0]	R	21 th Check Parity generated from main area	0x00
RSVD	[31:16]	—	Reserved	—
26 th Parity	[15:8]	R	26 th Check Parity generated from main area	0x00
25 th Parity	[7:0]	R	25 th Check Parity generated from main area	0x00

NOTE: The NAND flash controller generate these ECC parity codes when write main area data while the MainECCLock (NFCON[7]) bit is "0"(unlock).

10.4.3.6 NFECCERLn

- Address = 0x0CE2_00C0, Reset Value = 0x0000_0000
- Address = 0x0CE2_00C4, Reset Value = 0x0000_0000
- Address = 0x0CE2_00C8, Reset Value = 0x0000_0000
- Address = 0x0CE2_00CC, Reset Value = 0x0000_0000
- Address = 0x0CE2_00D0, Reset Value = 0x0000_0000
- Address = 0x0CE2_00D4, Reset Value = 0x0000_0000
- Address = 0x0CE2_00D8, Reset Value = 0x0000_0000
- Address = 0x0CE2_00DC, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	–	Reserved	0x0
ErrByteLoc2	[25:16]	R	Error byte location of 2 nd bit error	0x000
RSVD	[15:10]	–	Reserved	0x0
ErrByteLoc1	[9:0]	R	Error byte location of 1 st bit error	0x000
RSVD	[31:26]	–	Reserved	0x0
ErrByteLoc4	[25:16]	R	Error byte location of 4 th bit error	0x000
RSVD	[15:10]	–	Reserved	0x0
ErrByteLoc3	[9:0]	R	Error byte location of 3 rd bit error	0x000
RSVD	[31:26]	–	Reserved	0x0
ErrByteLoc6	[25:16]	R	Error byte location of 6 th bit error	0x000
RSVD	[15:10]	–	Reserved	0x0
ErrByteLoc5	[9:0]	R	Error byte location of 5 th bit error	0x000
RSVD	[31:26]	–	Reserved	0x0
ErrByteLoc8	[25:16]	R	Error byte location of 8 th bit error	0x000
RSVD	[15:10]	–	Reserved	0x0
ErrByteLoc7	[9:0]	R	Error byte location of 7 th bit error	0x000
RSVD	[31:26]	–	Reserved	0x0
ErrByteLoc10	[25:16]	R	Error byte location of 10 th bit error	0x000
RSVD	[15:10]	–	Reserved	0x0
ErrByteLoc9	[9:0]	R	Error byte location of 9 th bit error	0x000
RSVD	[31:26]	–	Reserved	0x0
ErrByteLoc12	[25:16]	R	Error byte location of 12 th bit error	0x000
RSVD	[15:10]	–	Reserved	0x0
ErrByteLoc11	[9:0]	R	Error byte location of 11 th bit error	0x000
RSVD	[31:26]	–	Reserved	0x0
ErrByteLoc14	[25:16]	R	Error byte location of 14 th bit error	0x000
RSVD	[15:10]	–	Reserved	0x0

Name	Bit	Type	Description	Reset Value
ErrByteLoc13	[9:0]	R	Error byte location of 13 th bit error	0x000
RSVD	[31:26]	—	Reserved	0x0
ErrByteLoc16	[25:16]	R	Error byte location of 16 th bit error	0x000
RSVD	[15:10]	—	Reserved	0x0
ErrByteLoc15	[9:0]	R	Error byte location of 15 th bit error	0x000

NOTE: These values are updated when DecodeDone (NFECCSTAT[24]) is set ("1").

10.4.3.7 NFECCERPn

- Address = 0x0CE2_00F0, Reset Value = 0x0000_0000
- Address = 0x0CE2_00F4, Reset Value = 0x0000_0000
- Address = 0x0CE2_00F8, Reset Value = 0x0000_0000
- Address = 0x0CE2_00FC, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
4 th ErrBitPattern	[31:24]	R	4 th Error bit pattern	0x00
3 rd ErrBitPattern	[23:16]	R	3 rd Error bit pattern	0x00
2 nd ErrBitPattern	[15:8]	R	2 nd Error bit pattern	0x00
1 st ErrBitPattern	[7:0]	R	1 st Error bit pattern	0x00
8 th ErrBitPattern	[31:24]	R	8 th Error bit pattern	0x00
7 th ErrBitPattern	[23:16]	R	7 th Error bit pattern	0x00
6 th ErrBitPattern	[15:8]	R	6 th Error bit pattern	0x00
5 th ErrBitPattern	[7:0]	R	5 th Error bit pattern	0x00
12 th ErrBitPattern	[31:24]	R	12 th Error bit pattern	0x00
11 th ErrBitPattern	[23:16]	R	11 th Error bit pattern	0x00
10 th ErrBitPattern	[15:8]	R	10 th Error bit pattern	0x00
9 th ErrBitPattern	[7:0]	R	9 th Error bit pattern	0x00
16 th ErrBitPattern	[31:24]	R	16 th Error bit pattern	0x00
15 th Error bit pattern	[23:16]	R	15 th Error bit pattern	0x00
14 th ErrBitPattern	[15:8]	R	14 th Error bit pattern	0x00
13 th ErrBitPattern	[7:0]	R	13 th Error bit pattern	0x00

NOTE: These values are updated when DecodeDone (NFECCSTAT[25]) is set ("1").

10.4.3.8 NFECCCONn

- Address = 0x0CE2_0110, Reset Value = 0x0000_0000
- Address = 0x0CE2_0114, Reset Value = 0x0000_0000
- Address = 0x0CE2_0118, Reset Value = 0x0000_0000
- Address = 0x0CE2_011C, Reset Value = 0x0000_0000
- Address = 0x0CE2_0120, Reset Value = 0x0000_0000
- Address = 0x0CE2_0124, Reset Value = 0x0000_0000
- Address = 0x0CE2_0128, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
4 th Conversion Code	[31:24]	RW	4 th ECC parity conversion code.	0x00
3 rd Conversion Code	[23:16]	RW	3 rd ECC parity conversion code.	0x00
2 nd Conversion Code	[15:8]	RW	2 nd ECC parity conversion code.	0x00
1 st Conversion Code	[7:0]	RW	1 st ECC parity conversion code.	0x00
8 th Conversion Code	[31:24]	RW	8 th ECC parity conversion code.	0x00
7 th Conversion Code	[23:16]	RW	7 th ECC parity conversion code.	0x00
6 th Conversion Code	[15:8]	RW	6 th ECC parity conversion code.	0x00
5 th Conversion Code	[7:0]	RW	5 th ECC parity conversion code.	0x00
12 th Conversion Code	[31:24]	RW	12 th ECC parity conversion code.	0x00
11 th Conversion Code	[23:16]	RW	11 th ECC parity conversion code.	0x00
10 th Conversion Code	[15:8]	RW	10 th ECC parity conversion code.	0x00
9 th Conversion Code	[7:0]	RW	9 th ECC parity conversion code.	0x00
16 th Conversion Code	[31:24]	RW	16 th ECC parity conversion code.	0x00
15 th Conversion Code	[23:16]	RW	15 th ECC parity conversion code.	0x00
14 th Conversion Code	[15:8]	RW	14 th ECC parity conversion code.	0x00
13 th Conversion Code	[7:0]	RW	13 th ECC parity conversion code.	0x00
20 th Conversion Code	[31:24]	RW	20 th ECC parity conversion code.	0x00
19 th Conversion Code	[23:16]	RW	19 th ECC parity conversion code.	0x00
18 th Conversion Code	[15:8]	RW	18 th ECC parity conversion code.	0x00
17 th Conversion Code	[7:0]	RW	17 th ECC parity conversion code.	0x00
24 th Conversion Code	[31:24]	RW	24 th ECC parity conversion code.	0x00
23 th Conversion Code	[23:16]	RW	23 th ECC parity conversion code.	0x00
22 th Conversion Code	[15:8]	RW	22 th ECC parity conversion code.	0x00
21 th Conversion Code	[7:0]	RW	21 th ECC parity conversion code.	0x00
RSVD	[31:16]	—	Reserved	0x0000
26 th Conversion Code	[15:8]	RW	26 th ECC parity conversion code.	0x00
25 th Conversion Code	[7:0]	RW	25 th ECC parity conversion code.	0x00

NOTE: For more information about ECC parity conversion codes, refer to [*10.3.4.11 ECC Parity Conversion Code Guide for 8/12/16-bit ECC.*](#)

11

PWM Timer

11.1 Overview

The Exynos4210 has five 32-bit Pulse Width Modulation (PWM) timers. These timers generate internal interrupts for the ARM subsystem. In addition, Timers 0, 1, 2 and 3 include a PWM function, which drives an external I/O signal. The PWM in timer 0 has an optional dead-zone generator capability to support a large current device. Timer 4 is internal timers without output pins.

The Timers use the ACLK_100 clock as source clock, which is from clock controller (PERIL). Timers 0 and 1 share a programmable 8-bit prescaler that provides the first level of division for the clock source. Timers 2, 3, and 4 share a different 8-bit prescaler. Each timer has its own private clock-divider that provides a second level of clock division (prescaler divided by 2, 4, 8, or 16).

Each timer has its own 32-bit down-counter which is driven by the timer clock. The down-counter is initially loaded from the Timer Count Buffer register (TCNTBn). If the down-counter reaches zero, the timer interrupt request is generated to inform the CPU that the timer operation is complete. If the timer down-counter reaches zero, the value of corresponding TCNTBn automatically reloads into the down-counter to start a next cycle. However, if the timer stops, for example, by clearing the timer enable bit of TCONn during the timer running mode, the value of TCNTBn is not reloaded into the counter.

The PWM function uses the value of the TCMPBn register. The timer control logic changes the output level if down-counter value matches the value of the compare register in timer control logic. Therefore, the compare register determines the turn-on time (or turn-off time) of a PWM output.

The TCNTBn and TCMPBn registers are double buffered to allow the timer parameters to be updated in the middle of a cycle. The new values do not take effect until the current timer cycle completes.

The [Figure 11-1](#) shows a simple example of a PWM cycle.

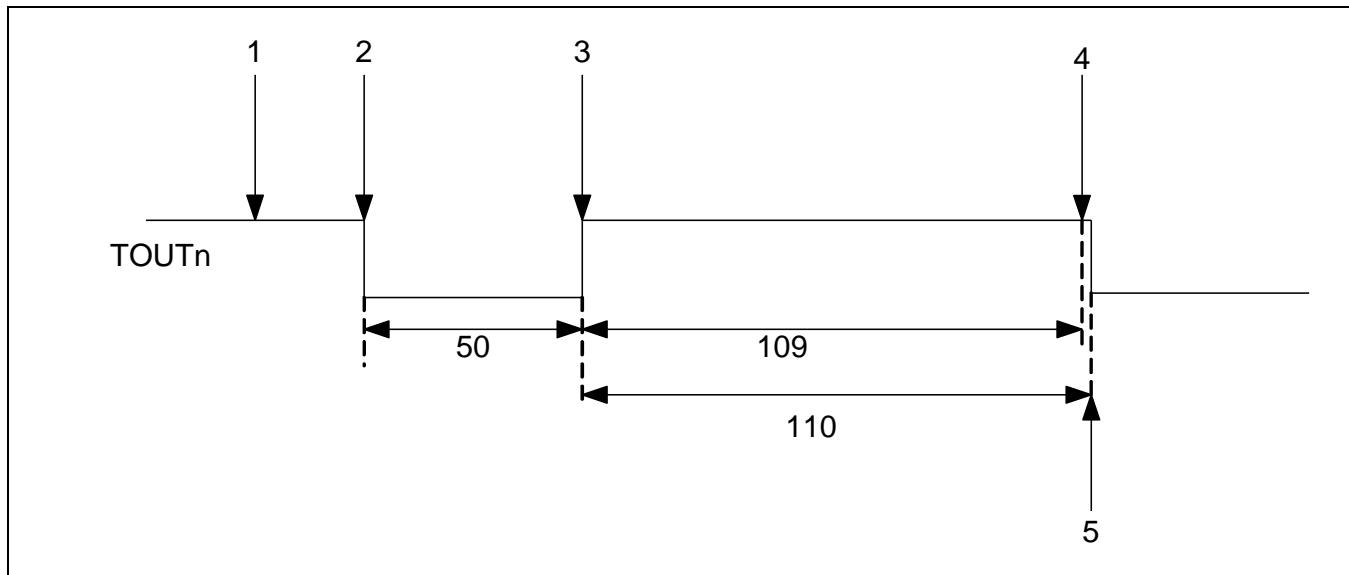


Figure 11-1 Simple Example of a PWM Cycle

Steps in PWM cycle:

- Initialize the TCNTBn register with 159(50+109) and TCMPBn with 109.
- Start Timer: Set the start bit and manually update this bit to off.
The TCNTBn value of 159 is loaded into the down-counter, and then the output TOUTn is set to low.
- If down-counter counts down the value from TCNTBn to value in the TCMPBn register 109, the output changes from low to high
- If the down-counter reaches 0, it generates an interrupt request.
- The down-counter automatically reloads TCNTBn. This restarts the cycle.

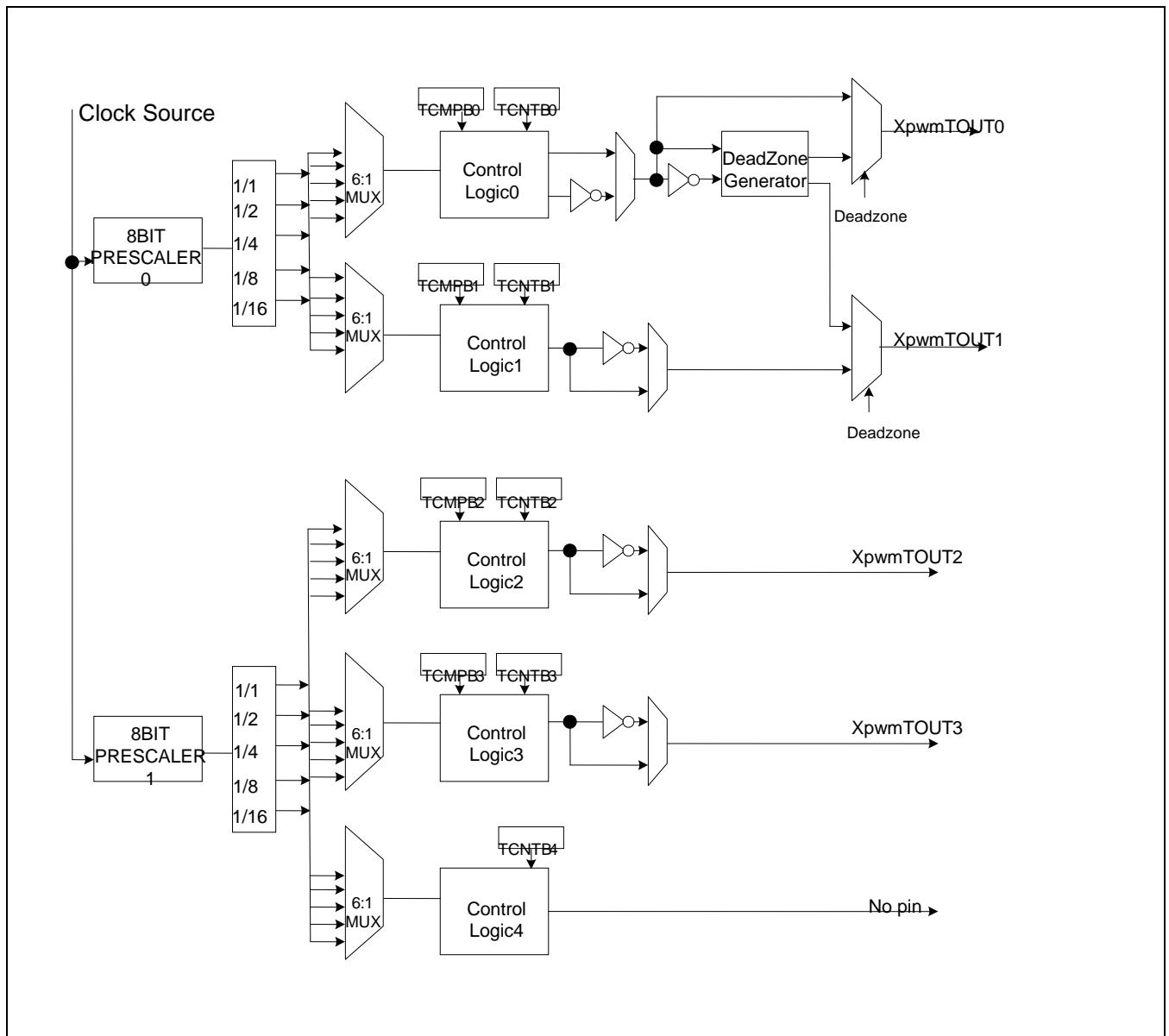


Figure 11-2 PWM TIMER Clock Tree Diagram

The [Figure 11-2](#) shows the clock generation scheme for individual PWM Channels.

Each timer can generate level interrupts.

11.2 Features

The Features supported by the PWM include:

- Five 32-bit Timers.
- Two 8-bit Clock Prescalers providing first level of division for the clock source, and five Clock Dividers and Multiplexers providing second level of division for the Prescaler clock
- Programmable Clock Select Logic for individual PWM Channels.
- Four Independent PWM Channels with Programmable Duty Control and Polarity.
- Static Configuration: PWM is stopped
- Dynamic Configuration: PWM is running.
- Auto-Reload and One-Shot Pulse Mode.
- One external input to start the PWM.
- Dead Zone Generator on two PWM Outputs.
- Level Interrupt Generation.

The PWM has two operation modes, namely, Auto-Reload and One-Shot Pulse:

- Auto-Reload Mode
In this mode, continuous PWM pulses are generated based on programmed duty cycle and polarity.
- One-Shot Pulse Mode
In this mode, only one PWM pulse is generated based on programmed duty cycle and polarity.

To control the functionality of PWM, 18 special function registers are provided. The PWM is a programmable output, dual clock input AMBA slave module and connects to the Advanced Peripheral Bus (APB). These 18 special function registers within PWM are accessed via APB transactions.

11.3 Functional Description

11.3.1 Interface Port Description

Signal	I/O	Description	Pad	Type
TOUT_0	Output	PWMTIMER TOUT[0]	XpwmTOUT[0]	muxed
TOUT_1	Output	PWMTIMER TOUT[1]	XpwmTOUT[1]	muxed
TOUT_2	Output	PWMTIMER TOUT[2]	XpwmTOUT[2]	muxed
TOUT_3	Output	PWMTIMER TOUT[3]	XpwmTOUT[3]	muxed

NOTE: Type field indicates whether pads are dedicated to the signal or pads are connected to the multiplexed signals.

11.3.2 Prescaler and Divider

An 8-bit prescaler and 3-bit divider generates the following output frequencies.

Table 11-1 Minimum and Maximum Resolution Based on Prescaler and Clock Divider Values

4-bit Divider Settings	Minimum Resolution (Prescaler Value = 1)	Maximum Resolution (Prescaler Value = 255)	Maximum Interval (TCNTBn = 4294967295)
1/1 (clk source = 66 MHz)	0.030us(33.0 MHz)	3.879us (257.8 kHz)	16659.27s
1/2 (clk source = 66 MHz)	0.061us (16.5 MHz)	7.758us (128.9 kHz)	33318.53s
1/4 (clk source = 66 MHz)	0.121us (8.25 MHz)	15.515us (64.5 kHz)	66637.07s
1/8 (clk source = 66 MHz)	0.242us (4.13 MHz)	31.03us (32.2 kHz)	133274.14s
1/16 (clk source = 66 MHz)	0.485us (2.06 MHz)	62.061us (16.1 kHz)	266548.27s

11.3.3 Basic Timer Operation

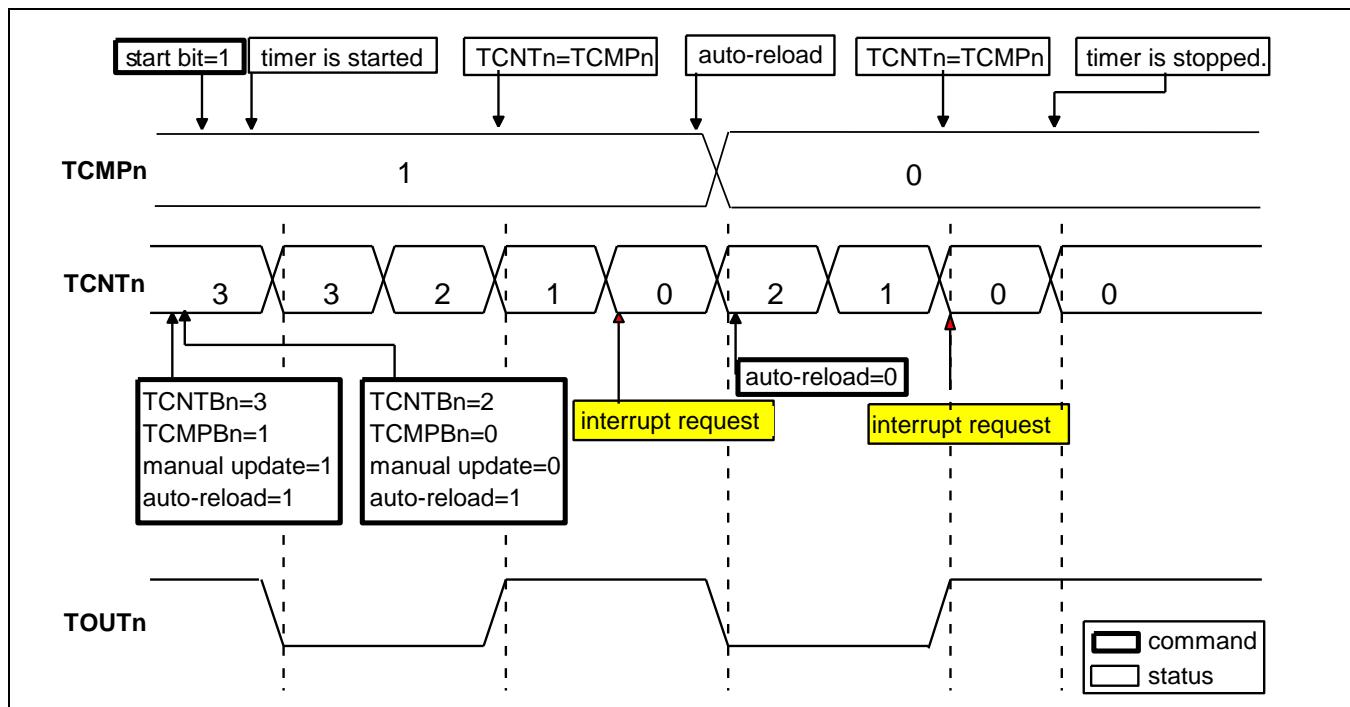


Figure 11-3 Timer Operations

The timer (except the timer channel 4) comprises of four registers, namely, TCNTBn, TCNTn, TCMPBn and TCMPn. If the timer reaches 0, then TCNTBn and TCMPBn registers are loaded into TCNTn and TCMPn. If TCNTn reaches 0, then the interrupt request occurs if the interrupt is enabled (TCNTn and TCMPn are the names of the internal registers. The TCNTn register is read from the TCNTOn register).

To generate interrupt at intervals 3cycle of XpwmTOUTn, set TCNTBn, TCMPBn and TCON register as shown in [Figure 11-3](#).

Steps to generate interrupt:

1. Set TCNTBn=3 and TCMPBn=1.
2. Set auto-reload=1 and manual update=1.
If manual update bit is 1, then TCNTBn and TCMPBn values are loaded to TCNTn and TCMPn.
3. Set TCNTBn=2 and TCMPBn=0 for the next operation. 4. Set auto-reload=1 and manual update=0.
If you set manual update=1 at this time, TCNTn is changed to 2 and TCMP is changed to 0.
Therefore, interrupt is generated at interval two-cycle instead of three-cycle.
You must set auto-reload=1 automatically for the next operation.
5. Set start = 1 for starting the operation. Then TCNTn is down counting.
If TCNTn is 0, interrupt is generated and if auto-reload is enable, TCNTn is loaded 2 (TCNTBn value) and TCMPn is loaded 0 (TCMPn value).
6. TCNTn is down counting before it stops.

11.3.4 Auto-reload and Double Buffering

The PWM Timers includes a double buffering feature, which changes the reload value for the next timer operation without stopping the current timer operation.

The timer value is written into TCNTBn (Timer Count Buffer register) and the current counter value of the timer is read from TCNTOn (Timer Count Observation register). If TCNTBn is read, the read value does not reflect the current state of the counter but the reload value for the next timer duration.

Auto-reload is the operation copies the TCNTBn into TCNTn, if TCNTn reaches 0. The value written to TCNTBn, is loaded to TCNTn if the TCNTn reaches to 0 and auto-reload is enabled. If the TCNTn is 0 and the auto-reload bit is 0, then TCNTn does not operate further.

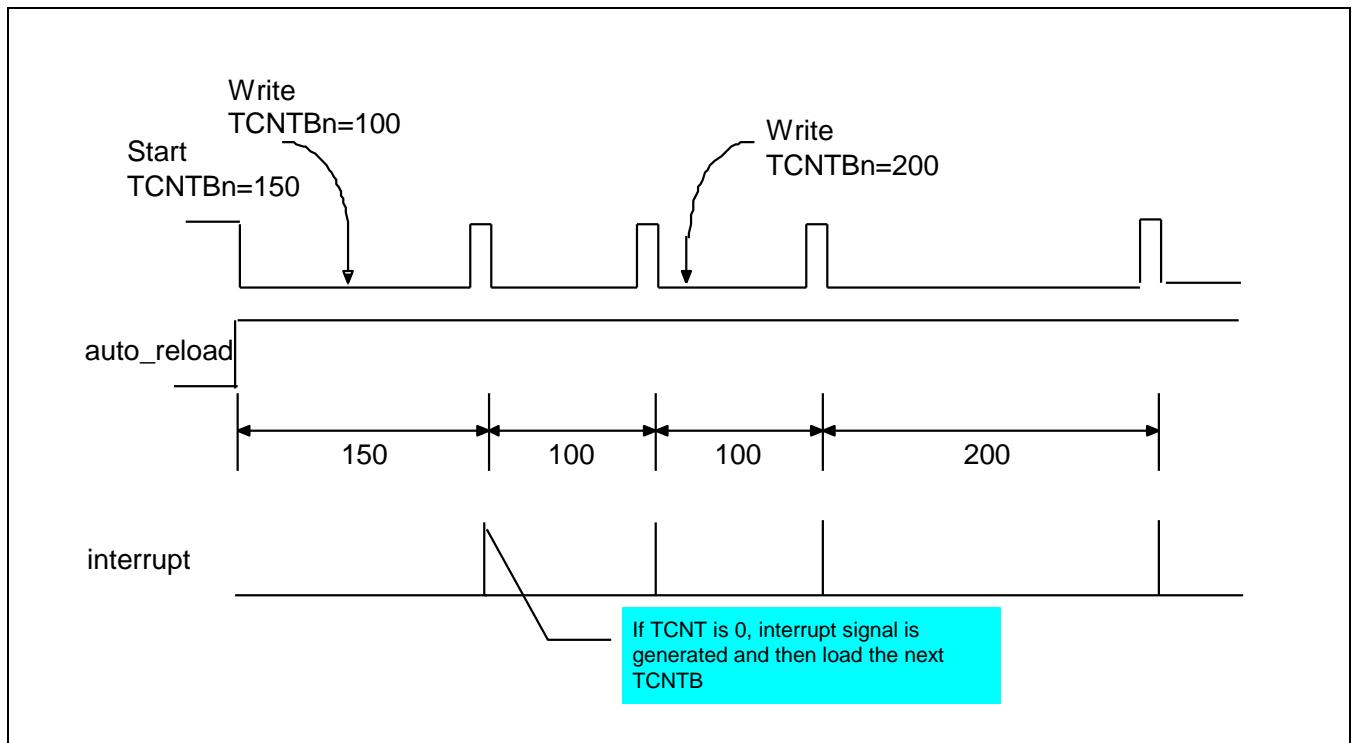


Figure 11-4 Example of Double Buffering Feature

11.3.5 Timer Operation Example

Example of timer operation is shown in [Figure 11-5](#).

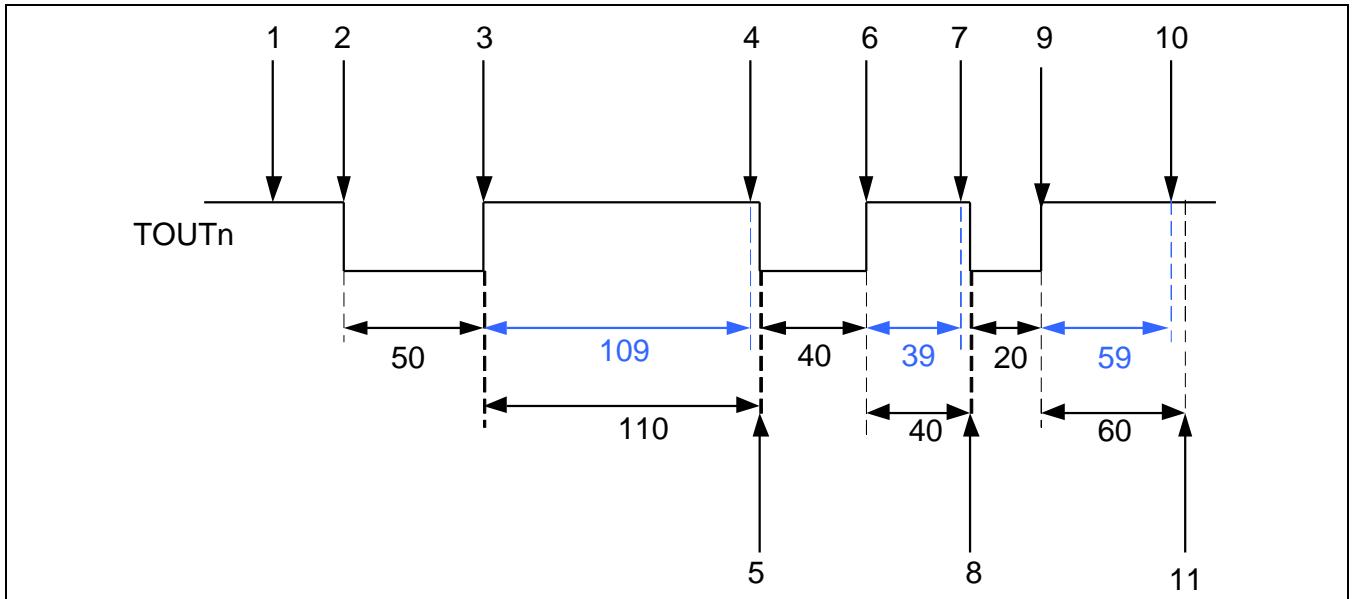


Figure 11-5 Example of a Timer Operation

Steps in timer operation:

1. Enable the auto-reload feature. Set the TCNTBn as 159(50+109) and TCMPBn as 109. Set the manual update bit on and set the manual update bit off. Set the inverter on/ off bit. The manual update bit sets TCNTn and TCMPn to the value of TCNTBn and TCMPBn.
2. Set TCNTBn and TCMPBn as 79(40+39) and 39.
3. Start Timer: Set the start bit in TCON
4. If TCNTn and TCMPn have the same value, the logic level of TOUTn is changed from low to high
5. When TCNTn reaches 0, it generates interrupt request.
6. TCNTn and TCMPn are automatically reloaded with TCNTBn and TCMPBn as (79(40+39)) and 39. In the Interrupt Service Routine (ISR), the TCNTBn and TCMPBn are set as 79(20+59) and 59.
7. If TCNTn and TCMPn have the same value, the logic level of TOUTn is changed from low to high
8. When TCNTn reaches to 0, it generates interrupt request.
9. TCNTn and TCMPn are automatically reloaded with TCNTBn, TCMPBn as (79(20+59)) and 59. The, auto-reload and interrupt request are disabled to stop the timer in the ISR.
10. If TCNTn and TCMPn have the same value, the logic level of TOUTn is changed from low to high
11. Even if TCNTn reaches to 0, no interrupt request is generated.
12. TCNTn is not reloaded and the timer is stopped because auto-reload is disabled.

11.3.6 Initialize Timer (Setting Manual-up Data and Inverter)

User must define the starting value of the TCNT n , because an auto-reload operation of the timer occurs when the down counter reaches to 0. In this case, the starting value must be loaded by manual update bit. The sequence to start a timer is as follows:

1. Write the initial value into TCNTB n and TCMPB n .
2. Set the manual update bit and clear only manual update bit of the corresponding timer.
- NOTE:** It is recommended to set the inverter on/off bit (whether inverter is used or not).
3. Set the start bit of the corresponding timer to start the timer

11.3.7 PWM (Pulse Width Modulation)

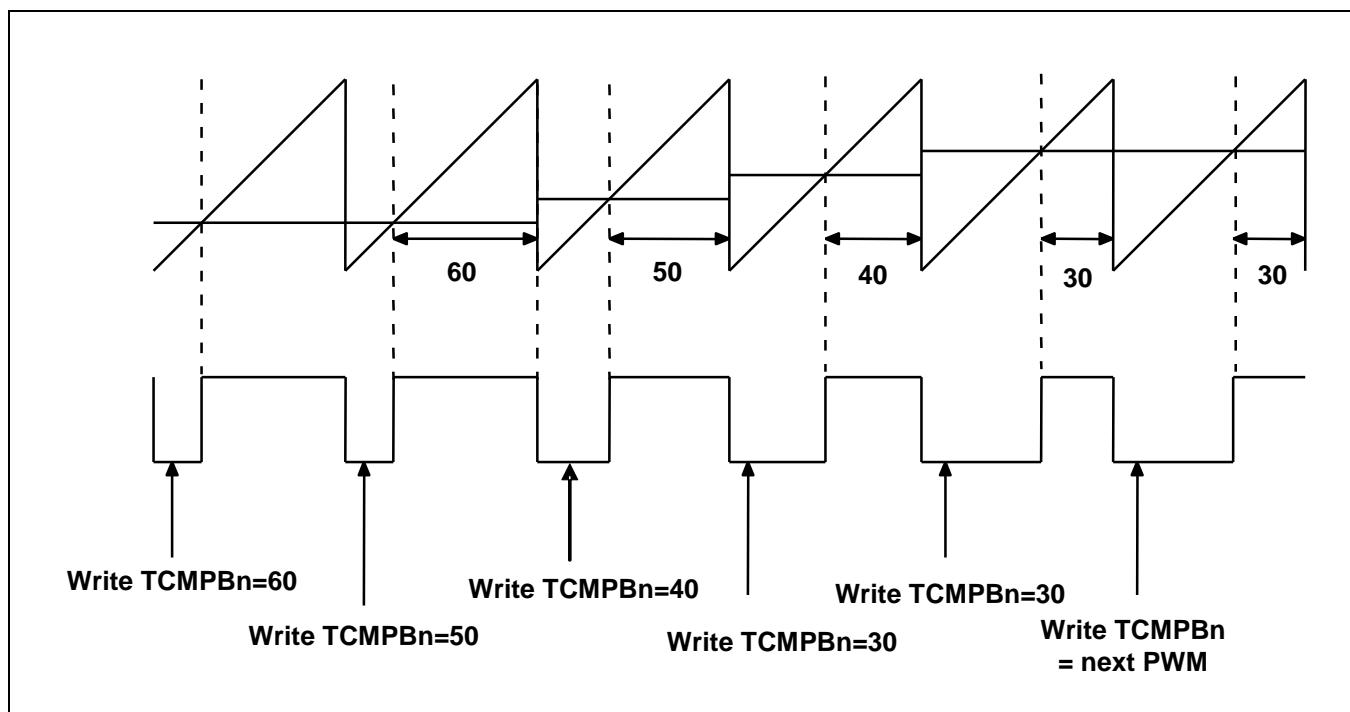


Figure 11-6 Example of PWM

Use TCMPB n to implement the PWM feature. PWM frequency is determined by TCNTB n . A PWM value is determined by TCMPB n as shown in the [Figure 11-6](#).

For a higher PWM value, decrease the TCMPB n value. For a lower PWM value, increase the TCMPB n value. If the output inverter is enabled, the increment/ decrement can be opposite.

Due to the double buffering feature, TCMPB n , for a next PWM cycle is written by ISR at any point of current PWM cycle.

11.3.8 Output Level Control

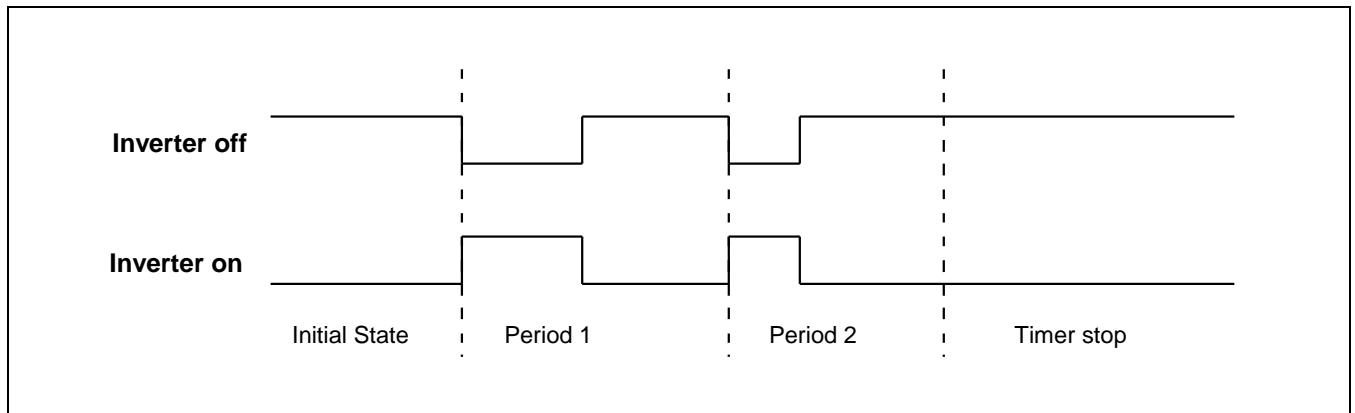


Figure 11-7 Inverter On/Off

Steps to maintain TOUT as high or low (Assume that inverter is off).

1. Turn off the auto-reload bit. Then, TOUTn goes to high level and the timer is stopped after TCNTn reaches to 0. This method is recommended.
2. Stop the timer by clearing the timer start/ stop bit to 0. If TCNTn <= TCMPn, the output level is high. If TCNTn >TCMPn, the output level is low
3. TOUTn is inverted by the inverter on/ off bit in TCON. The inverter removes the additional circuit to adjust the output level.

11.3.9 Dead Zone Generator

This feature inserts the time gap between a turn-off and turn-on of two different switching devices. This time gap prohibits the two switching device turning on simultaneously even for a very short time.

TOUT_0 specifies the PWM output. nTOUT_0 specifies the inversion of the TOUT_0. If the dead-zone is enabled, the output wave-form of TOUT_0 and nTOUT_0 is TOUT0_DZ and nTOUT0_DZ. TOUT0_DZ and nTOUT0_DZ cannot be turned on simultaneously by the dead zone interval. For functional correctness, the dead zone length must be set smaller than compare counter value.

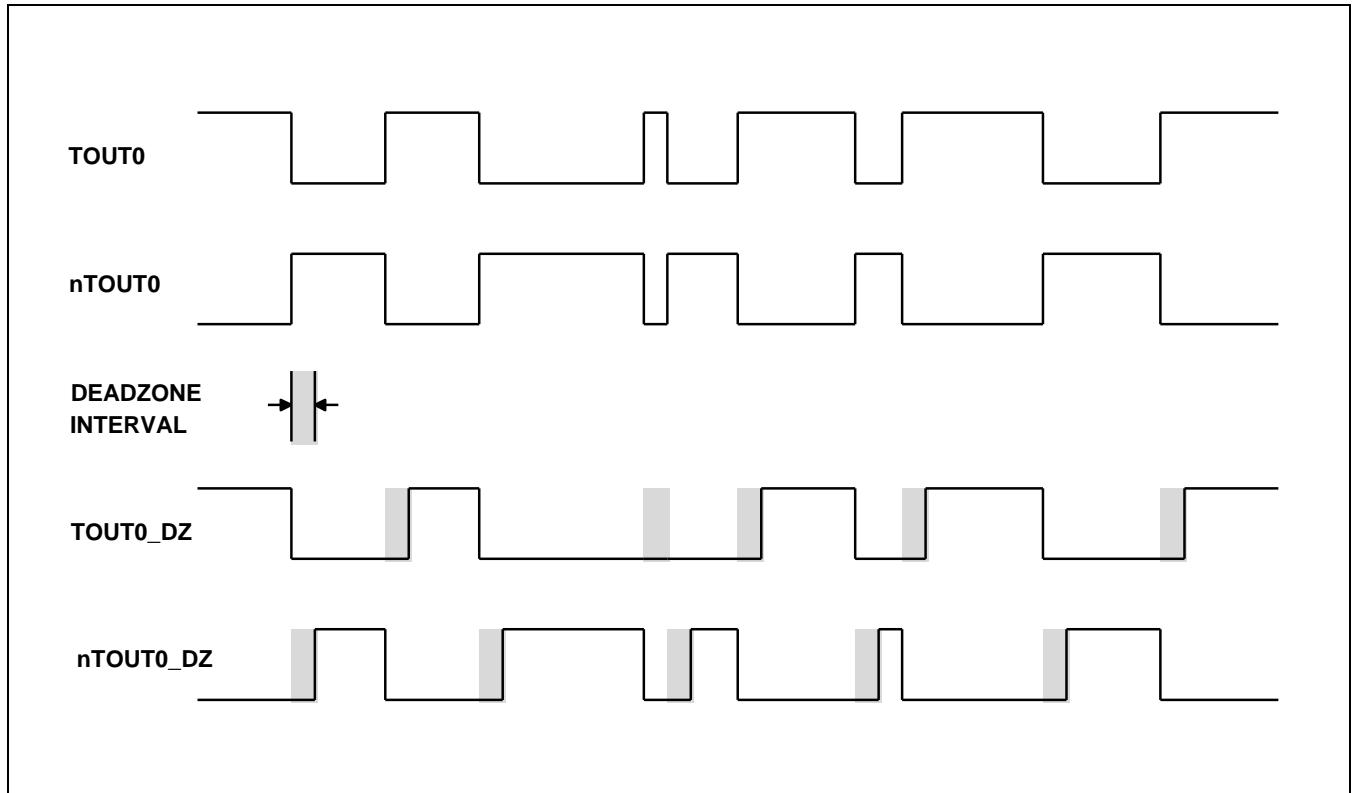


Figure 11-8 Waveform when a Deadzone Feature is Enabled

11.4 Register Description

11.4.1 Register Map Summary

- Base Address: 0x139D_0000

Register	Offset	Description	Reset Value
TCFG0	0x0000	Specifies the timer configuration register 0 that configures the two 8-bit prescaler and deadzone length	0x0000_0101
TCFG1	0x0004	Specifies the timer configuration register 1 that controls 5 mux select bit	0x0000_0000
TCON	0x0008	Specifies the timer control register	0x0000_0000
TCNTB0	0x000C	Specifies the timer 0 count buffer register	0x0000_0000
TCMPB0	0x0010	Specifies the timer 0 compare buffer register	0x0000_0000
TCNTO0	0x0014	Specifies the timer 0 count observation register	0x0000_0000
TCNTB1	0x0018	Specifies the timer 1 count buffer register	0x0000_0000
TCMPB1	0x001C	Specifies the timer 1 compare buffer register	0x0000_0000
TCNTO1	0x0020	Specifies the timer 1 count observation register	0x0000_0000
TCNTB2	0x0024	Specifies the timer 2 count buffer register	0x0000_0000
TCMPB2	0x0028	Specifies the timer 2 compare buffer register	0x0000_0000
TCNTO2	0x002C	Specifies the timer 2 count observation register	0x0000_0000
TCNTB3	0x0030	Specifies the timer 3 count buffer register	0x0000_0000
TCMPB3	0x0034	Specifies the timer 3 compare buffer register	0x0000_0000
TCNTO3	0x0038	Specifies the timer 3 count observation register	0x0000_0000
TCNTB4	0x003C	Specifies the timer 4 count buffer register	0x0000_0000
TCNTO4	0x0040	Specifies the timer 4 count observation register	0x0000_0000
TINT_CSTAT	0x0044	Specifies the timer interrupt control and status register	0x0000_0000

11.4.1.1 TCFG0

- Address = 0x139D_0000, Reset Value = 0x0000_0101

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	-	Reserved Bits	0x00
Dead zone length	[23:16]	RW	Dead zone length	0x00
Prescaler 1	[15:8]	RW	Prescaler 1 value for Timer 2, 3 and 4	0x01
Prescaler 0	[7:0]	RW	Prescaler 0 value for timer 0 and 1	0x01

Timer Input Clock Frequency = Clock Source / ({prescaler value + 1}) / {divider value}

{prescaler value} = 1 to 255

{divider value} = 1, 2, 4, 8, 16, TCLK

Dead zone length = 0 to 254

NOTE: If deadzone Length is set as "n", Real Dead Zone Length is "n+1" (n = 0 to 254).

11.4.1.2 TCFG1

- Address = 0x139D_0004, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	-	Reserved Bits	0x00
Divider MUX4	[19:16]	RW	Selects Mux input for PWM Timer 4 0000 = 1/1 0001 = 1/2 0010 = 1/4 0011 = 1/8 0100 = 1/16	0x00
Divider MUX3	[15:12]	RW	Selects Mux input for PWM Timer 3 0000 = 1/1 0001 = 1/2 0010 = 1/4 0011 = 1/8 0100 = 1/16	0x00
Divider MUX2	[11:8]	RW	Selects Mux input for PWM Timer 2 0000 = 1/1 0001 = 1/2 0010 = 1/4 0011 = 1/8 0100 = 1/16	0x00
Divider MUX1	[7:4]	RW	Selects Mux input for PWM Timer 1 0000 = 1/1 0001 = 1/2 0010 = 1/4 0011 = 1/8 0100 = 1/16	0x00
Divider MUX0	[3:0]	RW	Selects Mux input for PWM Timer 0 0000 = 1/1 0001 = 1/2 0010 = 1/4 0011 = 1/8 0100 = 1/16	0x00

11.4.1.3 TCON

- Address = 0x139D_0008, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:23]	-	Reserved Bits	0x000
Timer 4 Auto Reload on/off	[22]	RW	0 = One-Shot 1 = Interval Mode(Auto-Reload)	0x0
Timer 4 Manual Update	[21]	RW	0 = No Operation 1 = Update TCNTB4	0x0
Timer 4 Start/Stop	[20]	RW	0 = Stop 1 = Start Timer 4	0x0
Timer 3 Auto Reload on/off	[19]	RW	0 = One-Shot 1 = Interval Mode(Auto-Reload)	0x0
Timer 3 Output Inverter on/off	[18]	RW	0 = Inverter Off 1 = TOUT_3 Inverter-On	0x0
Timer 3 Manual Update	[17]	RW	0 = No Operation 1 = Update TCNTB3	0x0
Timer 3 Start/Stop	[16]	RW	0 = Stop 1 = Start Timer 3	0x0
Timer 2 Auto Reload on/off	[15]	RW	0 = One-Shot 1 = Interval Mode(Auto-Reload)	0x0
Timer 2 Output Inverter on/off	[14]	RW	0 = Inverter Off 1 = TOUT_2 Inverter-On	0x0
Timer 2 Manual Update	[13]	RW	0 = No Operation 1 = Update TCNTB2,TCMPB2	0x0
Timer 2 Start/Stop	[12]	RW	0 = Stop 1 = Start Timer 2	0x0
Timer 1 Auto Reload on/off	[11]	RW	0 = One-Shot 1 = Interval Mode(Auto-Reload)	0x0
Timer 1 Output Inverter on/off	[10]	RW	0 = Inverter Off 1 = TOUT_1 Inverter-On	0x0
Timer 1 Manual Update	[9]	RW	0 = No Operation 1 = Update TCNTB1,TCMPB1	0x0
Timer 1 Start/Stop	[8]	RW	0 = Stop 1 = Start Timer 1	0x0
Reserved	[7:5]	RW	Reserved Bits	0x0
Dead Zone Enable/Disable	[4]	RW	Dead Zone Generator Enable/Disable	0x0
Timer 0 Auto Reload on/off	[3]	RW	0 = One-Shot 1 = Interval Mode(Auto-Reload)	0x0
Timer 0 Output Inverter on/off	[2]	RW	0 = Inverter Off 1 = TOUT_0 Inverter-On	0x0
Timer 0 Manual Update	[1]	RW	0 = No Operation 1 = Update TCNTB0,TCMPB0	0x0

Name	Bit	Type	Description	Reset Value
Timer 0 Start/Stop	[0]	RW	0 = Stop 1 = Start Timer 0	0x0

11.4.1.4 TCNTB0

- Address = 0x139D_000C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
Timer 0 Count Buffer	[31:0]	RW	Timer 0 Count Buffer Register	0x0000_0000

11.4.1.5 TCMPB0

- Address = 0x139D_0010, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
Timer 0 Compare Buffer	[31:0]	RW	Timer 0 Compare Buffer Register	0x0000_0000

11.4.1.6 TCNTO0

- Address = 0x139D_0014, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
Timer 0 Count Observation	[31:0]	R	Timer 0 Count Observation Register	0x0000_0000

11.4.1.7 TCNTB1

- Address = 0x139D_0018, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
Timer 1 Count Buffer	[31:0]	RW	Timer 1 Count Buffer Register	0x0000_0000

11.4.1.8 TCMPB1

- Address = 0x139D_001C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
Timer 1 Compare Buffer	[31:0]	RW	Timer 1 Compare Buffer Register	0x0000_0000

11.4.1.9 TCNTO1

- Address = 0x139D_0020, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
Timer 1 Count Observation	[31:0]	R	Timer 1 Count Observation Register	0x0000_0000

11.4.1.10 TCNTB2

- Address = 0x139D_0024, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
Timer 2 Count Buffer	[31:0]	RW	Timer 2 Count Buffer Register	0x0000_0000

11.4.1.11 TCMPB2

- Address = 0x139D_0028, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
Timer 2 Compare Buffer	[31:0]	RW	Timer 2 Compare Buffer Register	0x0000_0000

11.4.1.12 TCNTO2

- Address = 0x139D_002C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
Timer 2 Count Observation	[31:0]	R	Timer 2 Count Observation Register	0x0000_0000

11.4.1.13 TCNTB3

- Address = 0x139D_0030, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
Timer 3 Count Buffer	[31:0]	RW	Timer 3 Count Buffer Register	0x0000_0000

11.4.1.14 TCMPB3

- Address = 0x139D_0034, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
Timer 3 Compare Buffer	[31:0]	RW	Timer 3 Compare Buffer Register	0x0000_0000

11.4.1.15 TCNTO3

- Address = 0x139D_0038, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
Timer 3 Count Observation	[31:0]	R	Timer 3 Count Observation Register	0x0000_0000

11.4.1.16 TCNTB4

- Address = 0x139D_003C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
Timer 4 Count Buffer	[31:0]	RW	Timer 4 Count Buffer Register	0x0000_0000

11.4.1.17 TCNTO4

- Address = 0x139D_0040, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
Timer 4 Count Observation	[31:0]	R	Timer 4 Count Observation Register	0x0000_0000

11.4.1.18 TINT_CSTAT

- Address = 0x139D_0044, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:10]	-	Reserved Bits	0x00000
Timer 4 Interrupt Status	[9]	RW	Timer 4 Interrupt Status Bit. Clears by writing "1" on this bit.	0x0
Timer 3 Interrupt Status	[8]	RW	Timer 3 Interrupt Status Bit. Clears by writing "1" on this bit.	0x0
Timer 2 Interrupt Status	[7]	RW	Timer 2 Interrupt Status Bit. Clears by writing "1" on this bit.	0x0
Timer 1 Interrupt Status	[6]	RW	Timer 1 Interrupt Status Bit. Clears by writing "1" on this bit.	0x0
Timer 0 Interrupt Status	[5]	RW	Timer 0 Interrupt Status Bit. Clears by writing "1" on this bit.	0x0
Timer 4 interrupt Enable	[4]	RW	Enables Timer 4 Interrupt. 0 = Disabled 1 = Enabled	0x0
Timer 3 interrupt Enable	[3]	RW	Enables Timer 3 Interrupt. 0 = Disables 1 = Enables	0x0
Timer 2 interrupt Enable	[2]	RW	Enables Timer 2 Interrupt. 0 = Disables 1 = Enables	0x0
Timer 1 interrupt Enable	[1]	RW	Enables Timer 1 Interrupt. 0 = Disables 1 = Enables	0x0
Timer 0 interrupt Enable	[0]	RW	Enables Timer 0 Interrupt. 0 = Disabled 1 = Enables	0x0

12 Watchdog Timer

12.1 Overview

The Watchdog Timer (WDT) in Exynos4210 is a timing device that resumes the controller operation after malfunctioning due to noise and system errors. WDT can be used as a normal 16-bit interval timer to request interrupt service. The WDT generates the reset signal.

The difference between WDT and PWM timer is that WDT generates the reset signal.

12.2 Features

- Supports Normal interval timer mode with interrupt request
- Activates Internal reset signal if the timer count value reaches 0 (Time-out).
- Supports Level-triggered Interrupt mechanism

12.3 Functional Description

12.3.1 Watchdog Timer Operation

[Figure 12-1](#) shows the functional block diagram of the watchdog timer. The watchdog timer uses ACLK_100 as its source clock, which is from clock controller(PERIR). The ACLK_100 frequency is prescaled to generate the corresponding watchdog timer clock, and the resulting frequency is divided again.

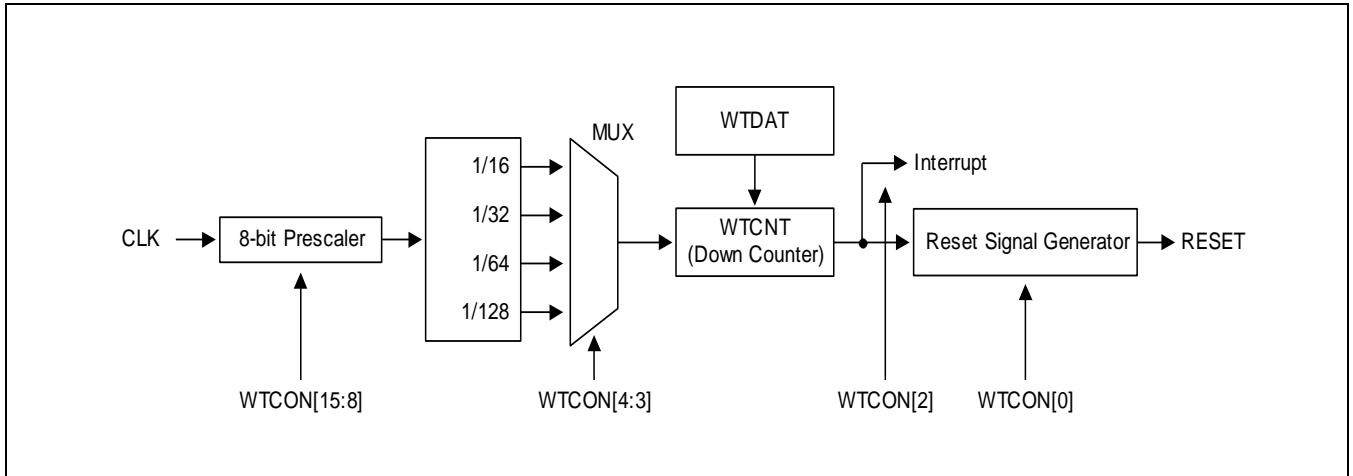


Figure 12-1 Watchdog Timer Block Diagram

The prescaler value and frequency division factor are specified in the watchdog timer control (WTCON) register. Valid prescaler values range from 0 to 2^8 -1. The frequency division factor can be selected as, 16, 32, 64, or 128.

Use the following equation to calculate the watchdog timer clock frequency and the duration of each timer clock cycle:

- $t_{\text{watchdog}} = 1/(\text{CLK}/(\text{Prescaler value} + 1)/\text{Division_factor})$

12.3.2 WTDAT and WTCNT

Once the watchdog timer is enabled, the value of watchdog timer data (WTDAT) register cannot be automatically reloaded into the timer counter (WTCNT). Therefore, an initial value must be written to the watchdog timer count (WTCNT) register, before the watchdog timer starts.

12.3.3 WDT Start

To start WDT, set WTCON[0] and WTCON[5] as 1.

12.3.4 Consideration of Debugging Environment

The watchdog timer must not operate if the Exynos4210 is in debug mode using Embedded ICE.

The watchdog timer determines from the CPU core signal (DBGACK signal) whether it is currently in the debug mode. Once the DBGACK signal is asserted, the reset output of the watchdog timer is not activated as the watchdog timer expires.

12.4 Register Description

12.4.1 Register Map Summary

- Base Address: 0x1006_000

Register	Offset	Description	Reset Value
WTCON	0x0000	Watchdog timer control register	0x0000_8021
WTDAT	0x0004	Watchdog timer data register	0x0000_8000
WTCNT	0x0008	Watchdog timer count register	0x0000_8000
WTCLRINT	0x000C	Watchdog timer interrupt clear register	Undefined

12.4.1.1 WTCON (Watchdog Timer Control Register)

- Address = 0x1006_0000, Reset Value = 0x0000_8021

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	-	Reserved	0
Prescaler value	[15:8]	RW	Prescaler value. The valid range is from 0 to $(2^8 - 1)$.	0x80
RSVD	[7:6]	-	Reserved. These two bits must be 00 in normal operation.	00
Watchdog timer	[5]	RW	Enables or disables Watchdog timer bit. 0 = Disables 1 = Enables	1
Clock select	[4:3]	RW	Determines the clock division factor. 00 = 16 01 = 32 10 = 64 11 = 128	00
Interrupt generation	[2]	RW	Enables or disables interrupt bit. 0 = Disables 1 = Enables	0
RSVD	[1]	-	Reserved. This bit must be 0 in normal operation.	0
Reset enable/disable	[0]	RW	Enables or disables Watchdog timer output bit for reset signal. 0 = Disables the reset function of the watchdog timer. 1 = Asserts reset signal of the Exynos4210 at watchdog time-out	1

The WTCON register allows you to enable/ disable the watchdog timer, select the clock signal from four different sources, enable/ disable interrupts, and enable/ disable the watchdog timer output.

The Watchdog timer is used to restart the Exynos4210 to recover from mal-function; if controller restart is not desired, the Watchdog timer should be disabled.

If you want to use the normal timer provided by the Watchdog timer, enable the interrupt and disable the Watchdog timer.

12.4.1.2 WTDAT (Watchdog Timer Data Register)

- Address = 0x1006_0004, Reset Value = 0x0000_8000

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	-	Reserved	0
Count reload value	[15:0]	RW	Watchdog timer count value for reload.	0x8000

The WTDAT register specifies the time-out duration. The content of WTDAT cannot be automatically loaded into the timer counter at initial watchdog timer operation. However, using 0x8000 (initial value) drives the first time-out. In this case, the value of WTDAT is automatically reloaded into WTCNT.

12.4.1.3 WTCNT (Watchdog Timer Count Register)

- Address = 0x1006_0008, Reset Value = 0x0000_8000

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	-	Reserved	0
Count value	[15:0]	RW	The current count value of the watchdog timer	0x8000

The WTCNT register contains the current count values for the watchdog timer during normal operation. Note that the content of the WTDAT register cannot be automatically loaded into the timer count register if the watchdog timer is enabled initially, therefore the WTCNT register must be set to an initial value before enabling it.

12.4.1.4 WTCLRINT (Watchdog Timer Interrupt Clear Register)

- Address = 0x1006_000C, Reset Value = Undefined

Name	Bit	Type	Description	Reset Value
Interrupt clear	[31:0]	RW	Write any value to clear the interrupt	-

The WTCLRINT register is used to clear the interrupt. Interrupt service routine is responsible to clear the relevant interrupt after the interrupt service is complete. Writing any values on this register clears the interrupt. Reading this register is not allowed.

13 Universal Asynchronous Receiver and Transmitter (UART)

13.1 Overview

The Universal Asynchronous Receiver and Transmitter (UART) in Exynos4210 provide four independent channels with asynchronous and serial input/output ports for general purpose (ch0~3), and one dedicated channel for communication with GPS (ch4). All the ports operate in an interrupt-based or a DMA-based mode. The UART generates an interrupt or a DMA request to transfer data to and from the CPU and the UART. The UART supports bit rates up to 3Mbps. Each UART channel contains two FIFOs to receive and transmit data: 256 bytes in ch0, 64 bytes in ch1 and ch4, and 16 bytes in ch2 and ch3.

UART includes programmable baud rates, infrared (IR) transmitter/receiver, one or two stop bit insertion, 5-bit, 6-bit, 7-bit, or 8-bit data width and parity checking.

Each UART contains a baud-rate generator, a transmitter, a receiver and a control unit, as shown in [Figure 13-1](#). The baud-rate generator uses SCLK_UART. The transmitter and the receiver contain FIFOs and data shifters. The data to be transmitted is written to Tx FIFO, and copied to the transmit shifter. The data is then shifted out by the transmit data pin (TxDn). The received data is shifted from the receive data pin (RxDn), and copied to Rx FIFO from the shifter.

13.2 Features

- RxD0, TxD0, RxD1, TxD1, RxD2, TxD2, RxD3 and TxD3 with DMA-based or interrupt-based operation
- UART Ch 0, 1, 2 and 3 with IrDA 1.0
- UART Ch 0 with 256 byte FIFO, Ch 1 and 4 with 64 byte FIFO, Ch 2 and 3 with 16 byte FIFO
- UART Ch 0, 1, 2 with nRTS0, nCTS0, nRTS1, nCTS1, nCTS2 and nRTS2 for Auto Flow Control
- UART Ch 4 is dedicated for communication with GPS and it supports Auto Flow Control
- Supports handshake transmit/receive.

13.3 Functional Description

13.3.1 Block Diagram

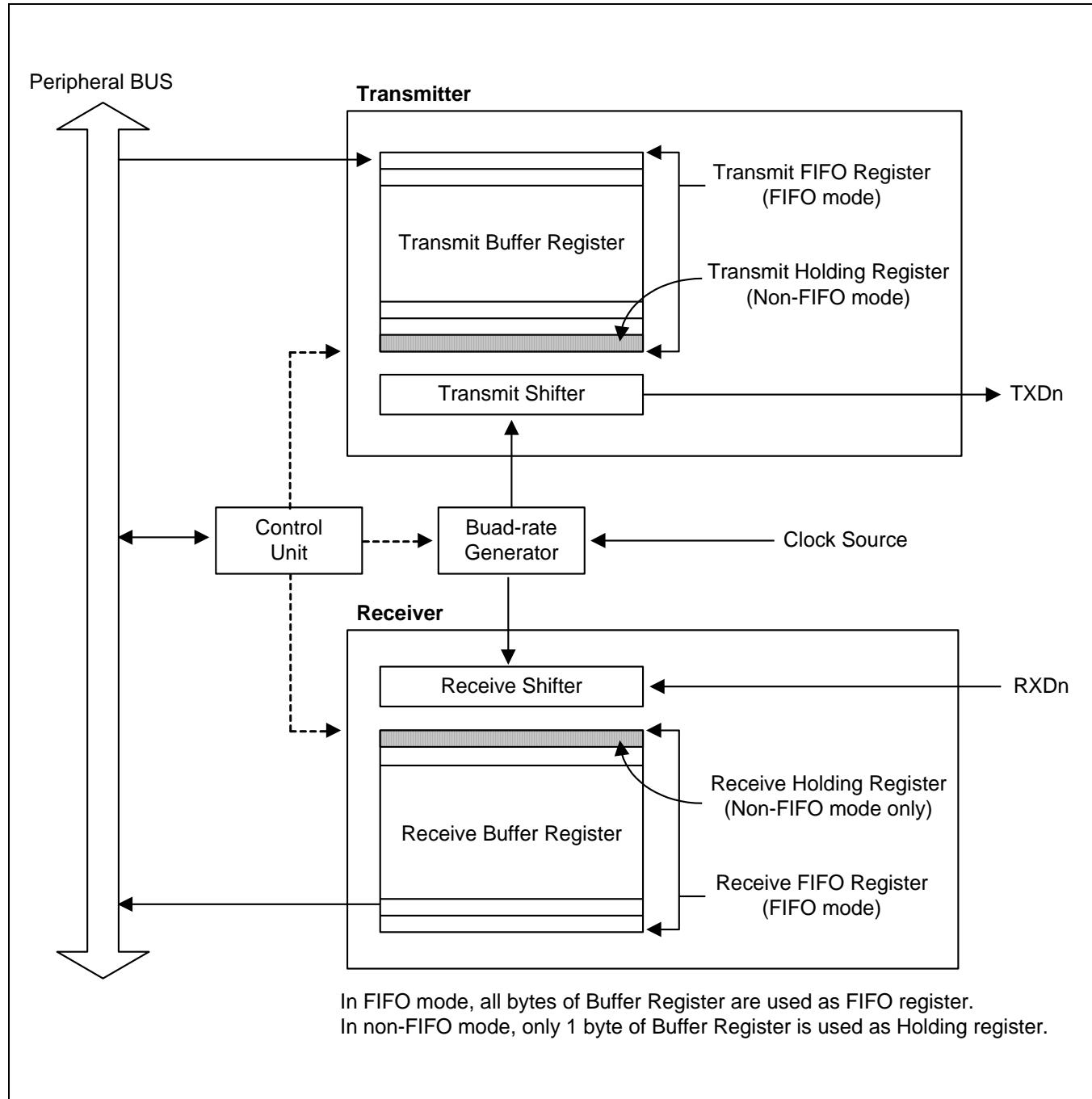


Figure 13-1 Block Diagram of UART

13.3.2 Interface Port Description

Signal	I/O	Description	Pad	Type
UART_0_RXD	Input	Receives Data for UART0	XuRXD_0	muxed
UART_0_TXD	Output	Transmits Data for UART0	XuTXD_0	muxed
UART_0_CTSn	Input	Clears to Send(active low) for UART0	XuCTSn_0	muxed
UART_0_RTn	Output	Requests to Send(active low) for UART0	XuRTSn_0	muxed
UART_1_RXD	Input	Receives Data for UART1	XuRXD_1	muxed
UART_1_TXD	Output	Transmits Data for UART1	XuTXD_1	muxed
UART_1_CTSn	Input	Clears to Send(active low) for UART1	XuCTSn_1	muxed
UART_1_RTn	Output	Requests to Send(active low) for UART1	XuRTSn_1	muxed
UART_2_RXD	Input	Receives Data for UART2	XuRXD_2	muxed
UART_2_TXD	Output	Transmits Data for UART2	XuTXD_2	muxed
UART_2_CTSn	Input	Clears to Send(active low) for UART2	XuCTSn_2	muxed
UART_2_RTn	Output	Requests to Send(active low) for UART2	XuRTSn_2	muxed
UART_3_RXD	Input	Receives Data for UART3	XuRXD_3	muxed
UART_3_TXD	Output	Transmits Data for UART3	XuTXD_3	muxed

NOTE:

1. Type field indicates whether pads are dedicated to the signal or pads are connected to the multiplexed signals. UART external pads are shared with IrDA. In order to use these pads, GPIO must be set before the start of UART. Please refer to Chapter 6 GPIO for exact settings.
2. UART4 has no IO ports. It is intended to communicate with the internal GPS module.

13.3.3 Data Transmission

The data frame for transmission is programmable. It consists of a start bit, five to eight data bits, an optional parity bit, and one to two stop bits, specified by the line control register (ULCONn). The transmitter can also produce a break condition that forces the serial output to logic 0 state for one frame transmission time. This block transmits the break signals after the present transmission word is transmitted completely. After the break signal transmission, the transmitter continuously transmits data to Tx FIFO (Tx holding register, in case of Non-FIFO mode).

13.3.4 Data Reception

Similar to data transmission, the data frame for reception is also programmable. It consists of a start bit, five to eight data bits, an optional parity bit, and one to two stop bits in the line control register (ULCONn). The receiver detects overrun error, parity error, frame error and break condition, each of which sets an error flag.

- Overrun error indicates that new data has overwritten the old data before the old data was read.
- Parity error indicates that the receiver has detected an unexpected parity condition.
- Frame error indicates that the received data does not have a valid stop bit.
- Break condition indicates that the RxDn input is held in the logic 0 state for more than one frame transmission time.

Receive time-out condition occurs if no data is received during the word time specified in UCON (this interval follows the setting of Word Length bit) and the Rx FIFO is not empty in the FIFO mode.

13.3.5 Auto Flow Control (AFC)

The UART0 and UART1 in Exynos4210 support auto flow control (AFC) using nRTS and nCTS signals. UART2 supports auto flow control if TxD3 and RxD3 are set as nRTS2 and nCTS2 by GPA1CON(GPIO SFR). In this case, it can be connected to external UARTs. To connect UART to a Modem, disable the AFC bit in UMCONn register and control the signal of nRTS using software. The UART4 supports auto flow control, but it is dedicated for communication with GPS.

In AFC, the nRTS signal depends on the condition of the receiver, whereas the nCTS signals control the operation of transmitter. The UART's transmitter transfers the data to FIFO if nCTS signals are activated (in AFC, nCTS signals means that other UART's FIFO is ready to receive data). Before UART receives data, the nRTS signals must be activated if its receive FIFO has more than 2-byte as spare. The nRTS signals must be inactivated if its receive FIFO has less than 1-byte as spare (in AFC, the nRTS signals means that its own receive FIFO is ready to receive data).

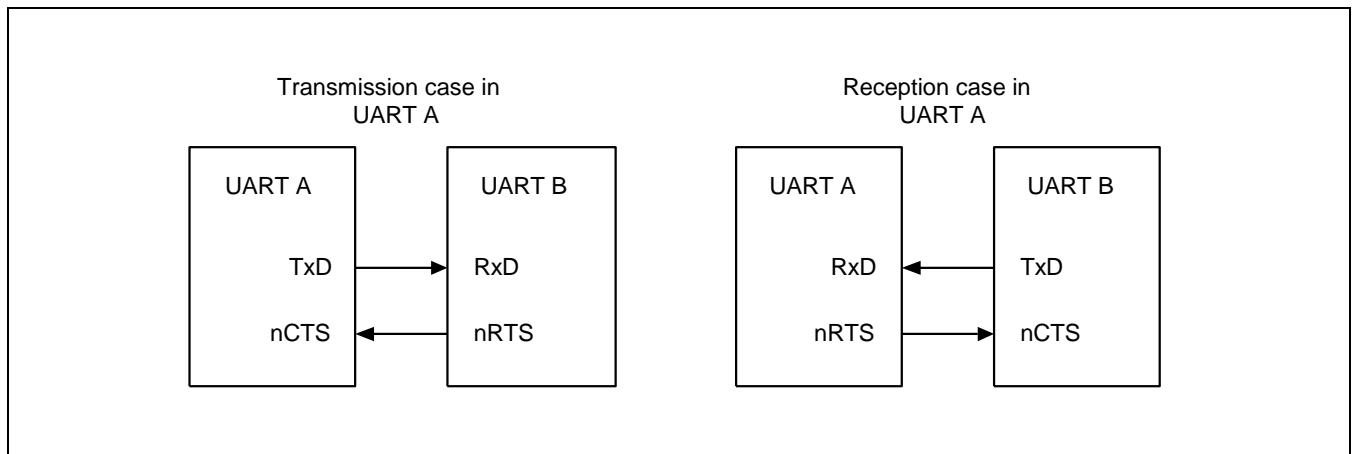


Figure 13-2 UART AFC Interface

13.3.6 Example of Non Auto-Flow Control (Controlling nRTS and nCTS by Software)

13.3.6.1 Rx Operation with FIFO

1. Select the transmit mode (Interrupt or DMA mode).
2. Check the value of Rx FIFO count in UFSTATn register. If the value is less than 16, you must set the value of UMCONn[0] to "1" (activate nRTS). However, if the value equal to or larger than 16, you must set the value to "0" (inactivate nRTS).
3. Repeat the Step 2.

13.3.6.2 Tx Operation with FIFO

1. Select the transmit mode (Interrupt or DMA mode).
2. Check the value of UMSTATn[0]. If the value is "1" (activate nCTS), you must write data to Tx FIFO register.
3. Repeat the Step 2.

13.3.7 Tx/Rx FIFO Trigger Level and DMA Burst Size in DMA Mode

If Tx/Rx data reaches the Tx/Rx FIFO trigger level of UFCONn register in DMA mode, the DMA transaction starts. A single DMA transaction transfers a data whose size is specified as the DMA burst size of UCONn register, and the DMA transactions are repeated until Tx/Rx FIFO count is less than the DMA burst size. Thus, DMA burst size should be less than or equal to Tx/Rx FIFO trigger level. In general, it is recommended to ensure that Tx/Rx FIFO trigger level and DMA burst size matches.

NOTE: Type field indicates whether pads are dedicated to the signal or pads are connected to the multiplexed signals. UART external pads are shared with IrDA. In order to use these pads, GPIO must be set before the start of UART. Please refer to Chapter 4 GPIO for exact settings.

13.3.8 RS-232C Interface

To connect the UART to modem interface (instead of null modem), nRTS, nCTS, nDSR, nDTR, DCD and nRI signals are required. You can control these signals with general I/O ports using software because the AFC does not support the RS-232C interface.

13.3.9 Interrupt/DMA Request Generation

Each UART in Exynos4210 comprises of seven status (Tx/Rx/Error) signals, namely, Overrun error, Parity error, Frame error, Break, Receive buffer data ready, Transmit buffer empty, and Transmit shifter empty. These conditions are indicated by the corresponding UART status register (UTRSTATn/UERSTATn).

The Overrun Error, Parity Error, Frame Error and Break Condition specify the receive error status. If receive-error-status-interrupt-enable bit is set to 1 in the control register (UCONn), the receive error status generates receive-error-status-interrupt. If a receive-error-status-interrupt-request is detected, you can identify the source of interrupt by reading the value of UERSTATn.

If the receiver transfers data of the receive shifter to the receive FIFO register in FIFO mode, and the number of received data is greater than or equal to the Rx FIFO Trigger Level, Rx interrupt is generated if Receive mode in control register (UCONn) is set to 1 (Interrupt request or polling mode).

In Non-FIFO mode, transferring the data of receive shifter to receive holding register causes Rx interrupt in the Interrupt request and polling modes.

If the transmitter transfers data from its transmit FIFO register to transmit shifter and the number of data left in transmit FIFO is less than or equal to the Tx FIFO Trigger Level, Tx interrupt is generated (provided Transmit mode in control register is selected as Interrupt request or polling mode). In Non-FIFO mode, transferring the data from transmit holding register to transmit shifter causes Tx interrupt in the Interrupt request and polling mode.

Remember that the Tx interrupt is always requested if the number of data in the transmit FIFO is smaller than the trigger level. This means that an interrupt is requested as soon as you enable the Tx interrupt, unless you fill the Tx buffer. It is recommended to fill the Tx buffer first and then enable the Tx interrupt.

The interrupt controllers of Exynos4210 are of the level-triggered type. You must set the interrupt type as ‘Level’ if you program the UART control registers.

If Receive and Transmit modes in control register are selected as DMA request mode, then DMA request occurs instead of Rx or Tx interrupt in the above situation.

Table 13-1 Interrupts in Connection with FIFO

Type	FIFO Mode	Non-FIFO Mode
Rx interrupt	Generated if Rx FIFO count is greater than or equal to the trigger level of received FIFO. Generated if the number of data in FIFO does not reaches Rx FIFO trigger Level and does not receive any data during the specified word time (receive time out). This interval is configurable and affected by the setting of Word Length bit.	Generated by receive holding register whenever receive buffer becomes full.
Tx interrupt	Generated if Tx FIFO count is less than or equal to the trigger level of transmit FIFO (Tx FIFO trigger Level).	Generated by transmit holding register whenever transmit buffer becomes empty.
Error interrupt	Generated if frame error, parity error, or break signal are detected. Generated if UART receives new data when Rx FIFO is full (overrun error).	Generated by all errors. However if another error occurs at the same time, only one interrupt is generated.

13.3.10 UART Error Status FIFO

UART contains the error status FIFO besides the Rx FIFO register. The error status FIFO indicates which data, among FIFO registers is received with an error. An error interrupt is issued only if the data containing an error, is ready to read out. To clear the error status FIFO, URXHn with an error and UERSTATn must be read out.

For example, it is assumed that the UART Rx FIFO receives A, B, C, D, and E characters sequentially and the frame error occurs while receiving 'B' and the parity error occurs while receiving 'D'.

The actual UART receive error does not generate any error interrupt, since the character, which was received with an error was not read. The error interrupt occurs if the character is read out.

Time	Sequence Flow	Error Interrupt	Note
#0	If no character is read out	–	–
#1	A, B, C, D, and E is received	–	–
#2	After A is read out	Frame error (in B) interrupt occurs.	The "B" has to be read out.
#3	After B is read out	–	–
#4	After C is read out	Parity error (in D) interrupt occurs.	The "D" has to be read out.
#5	After D is read out	–	–
#6	After E is read out	–	–

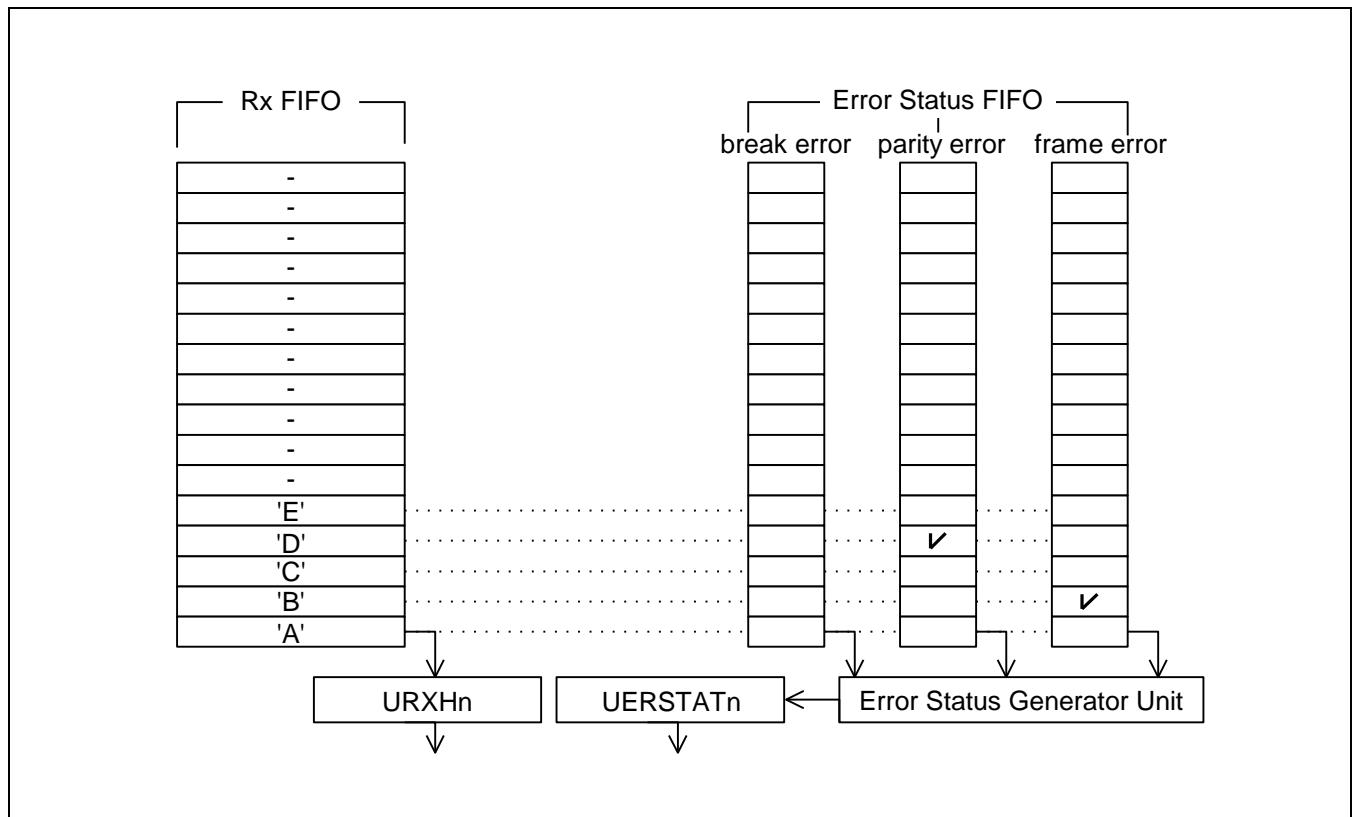


Figure 13-3 UART Receives the Five Characters Including Two Errors

13.3.10.1 Infra-Red (IR) Mode

The Exynos4210 UART block supports both infra-red (IR) transmission and reception. It is selected by setting the Infra-red-mode bit in the UART line control register (ULCONn). [Figure 13-4](#) illustrates how to implement the IR mode.

In IR transmit mode, the transmit pulse comes out at the rate of 3/16, that is, normal serial transmit rate (if the transmit data bit is 0). In IR receive mode, however, the receiver must detect the 3/16 pulsed period to recognize a 0 value (Refer to the frame timing diagrams shown in [Figure 13-5](#) and [Figure 13-7](#)).

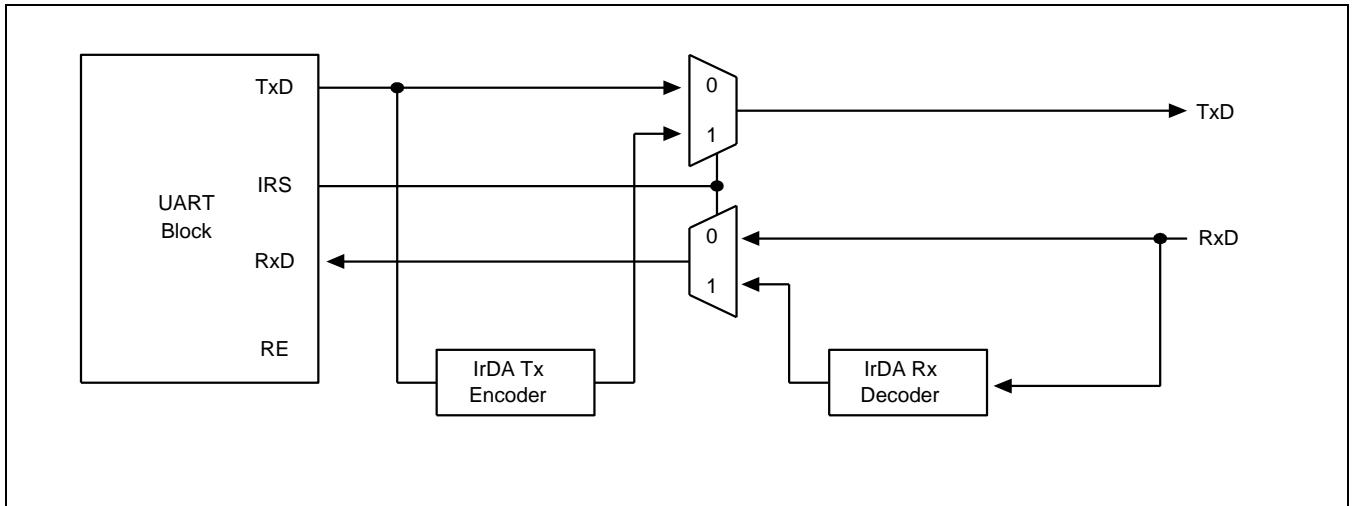


Figure 13-4 IrDA Function Block Diagram

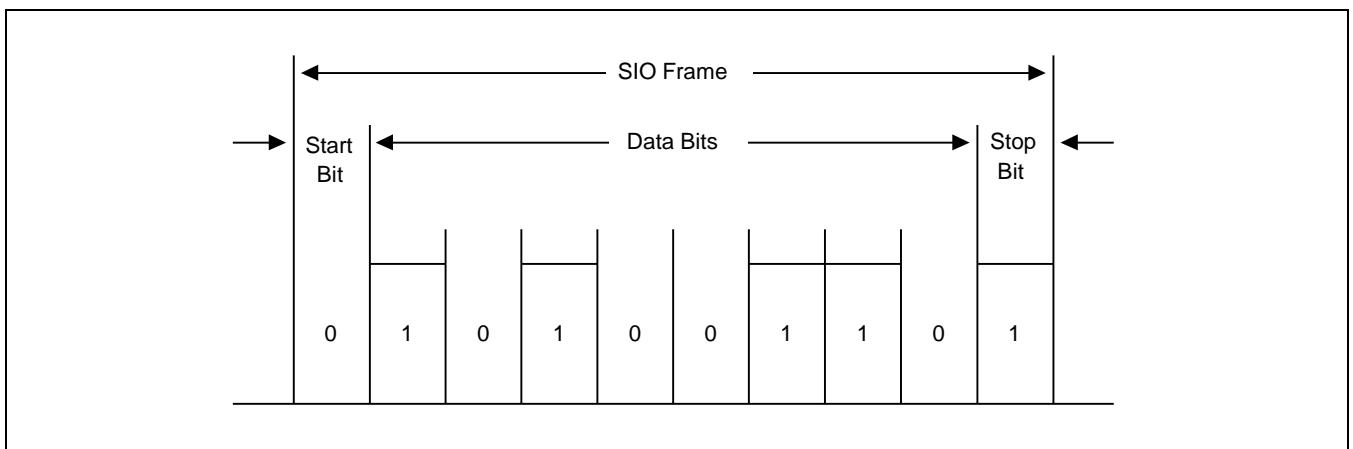


Figure 13-5 Serial I/O Frame Timing Diagram (Normal UART)

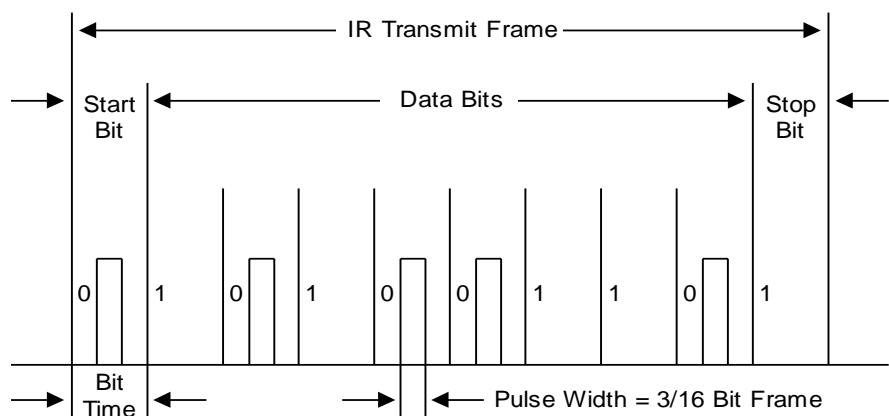


Figure 13-6 Infra-Red Transmit Mode Frame Timing Diagram

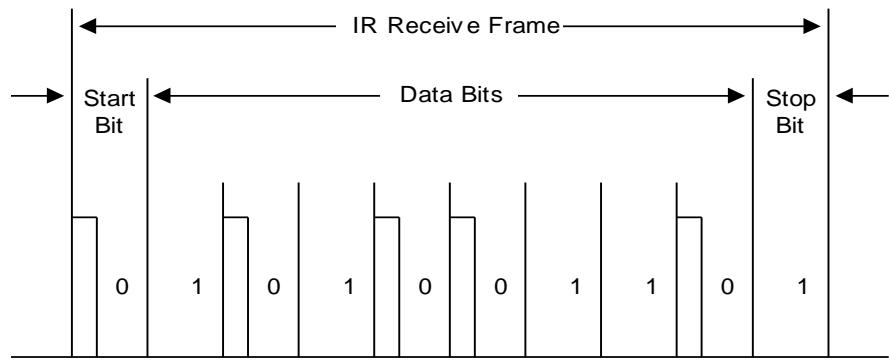


Figure 13-7 Infra-Red Receive Mode Frame Timing Diagram

13.3.11 UART Input Clock Description

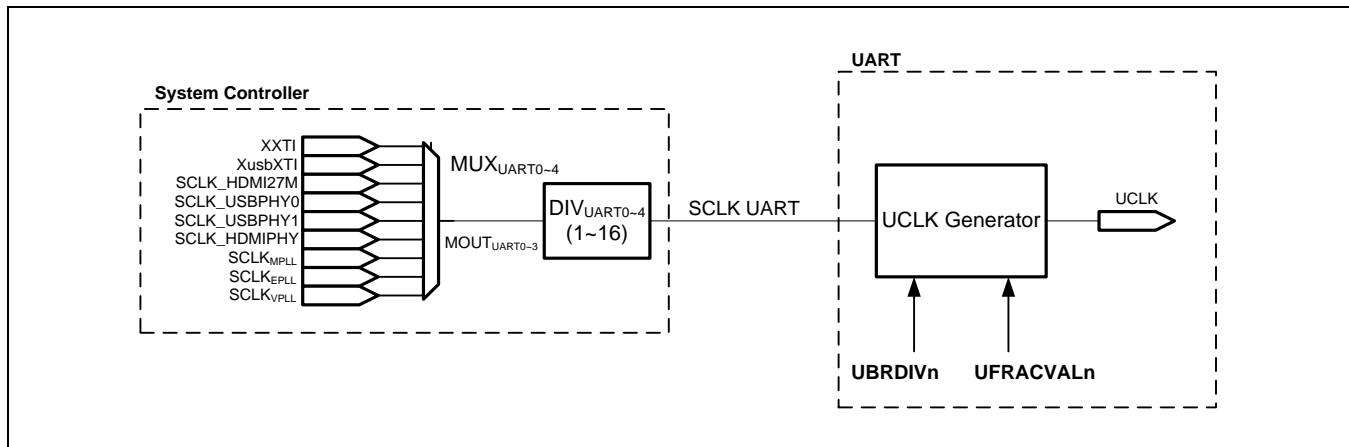


Figure 13-8 Input Clock Diagram for UART

Exynos4210 provides UART with a variety of clocks. As described in the [Figure 13-8](#), the UART uses SCLK_UART clock, which is from clock controller. You can also select SCLK_UART from various clock sources. To select SCLK_UART, please refer to Chapter 5 "Clock Management Units" (CMUs).

13.4 Register Description

13.4.1 Register Map Summary

- Base Address: 0x1380_0000, 0x1381_0000, 0x1382_0000, 0x1383_0000, 0x1384_0000

Register	Offset	Description	Reset Value
ULCONn	0x0000	Specifies line control	0x0000_0000
UCONn	0x0004	Specifies control	0x0000_3000
UFCONn	0x0008	Specifies FIFO control	0x0000_0000
UMCONn	0x000C	Specifies modem control	0x0000_0000
UTRSTATn	0x0010	Specifies Tx/Rx status	0x0000_0006
UERSTATn	0x0014	Specifies Rx error status	0x0000_0000
UFSTATn	0x0018	Specifies FIFO status	0x0000_0000
UMSTATn	0x001C	Specifies modem status	0x0000_0000
UTXHn	0x0020	Specifies transmit buffer	Undefined
URXHn	0x0024	Specifies receive buffer	0x0000_0000
UBRDIVn	0x0028	Specifies baud rate divisor	0x0000_0000
UFRACVALn	0x002C	Specifies divisor fractional value	0x0000_0000
UINTPn	0x0030	Specifies interrupt pending	0x0000_0000
UINTSPn	0x0034	Specifies interrupt source pending	0x0000_0000
UINTMn	0x0038	Specifies interrupt mask	0x0000_0000

13.4.1.1 ULCOn (n = 0 to 4)

- Address = Base Address + 0x0000, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:7]	—	Reserved	0
Infrared Mode	[6]	RW	Determines whether to use the Infrared mode. 0 = Normal mode operation 1 = Infrared Tx/Rx mode	0
Parity Mode	[5:3]	RW	Specifies the type of parity generation to be performed and checking during UART transmit and receive operation. 0xx = No parity 100 = Odd parity 101 = Even parity 110 = Parity forced/ checked as 1 111 = Parity forced/ checked as 0	000
Number of Stop Bit	[2]	RW	Specifies how many stop bits are used to signal end-of-frame signal. 0 = One stop bit per frame 1 = Two stop bit per frame	0
Word Length	[1:0]	RW	Indicates the number of data bits to be transmitted or received per frame. 00 = 5-bit 01 = 6-bit 10 = 7-bit 11 = 8-bit	00

13.4.1.2 UCONn (n = 0 to 4)

- Address = Base Address + 0x0004, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value								
RSVD	[31:23]	-	Reserved	0								
Tx DMA Burst Size	[22:20]	RW	<p>Tx DMA Burst Size It is the data transfer size of one DMA transaction which is triggered by a TX DMA request. The DMA program must be programmed to transfer the same data size as this value for a single TX DMA request.</p> <table> <tr><td>000 = 1 byte (Single)</td><td>001 = 4 bytes</td></tr> <tr><td>010 = 8 bytes</td><td>011 = Reserved</td></tr> <tr><td>100 = Reserved</td><td>101 = Reserved</td></tr> <tr><td>110 = Reserved</td><td>111 = Reserved</td></tr> </table>	000 = 1 byte (Single)	001 = 4 bytes	010 = 8 bytes	011 = Reserved	100 = Reserved	101 = Reserved	110 = Reserved	111 = Reserved	000
000 = 1 byte (Single)	001 = 4 bytes											
010 = 8 bytes	011 = Reserved											
100 = Reserved	101 = Reserved											
110 = Reserved	111 = Reserved											
RSVD	[19]	-	Reserved	0								
Rx DMA Burst Size	[18:16]	RW	<p>Rx DMA Burst Size It is the data transfer size of one DMA transaction which is triggered by a RX DMA request. The DMA program must be programmed to transfer the same data size as this value for a single RX DMA request.</p> <table> <tr><td>000 = 1 byte (Single)</td><td>001 = 4 bytes</td></tr> <tr><td>010 = 8 bytes</td><td>011 = 16 bytes</td></tr> <tr><td>100 = Reserved</td><td>101 = Reserved</td></tr> <tr><td>110 = Reserved</td><td>111 = Reserved</td></tr> </table>	000 = 1 byte (Single)	001 = 4 bytes	010 = 8 bytes	011 = 16 bytes	100 = Reserved	101 = Reserved	110 = Reserved	111 = Reserved	000
000 = 1 byte (Single)	001 = 4 bytes											
010 = 8 bytes	011 = 16 bytes											
100 = Reserved	101 = Reserved											
110 = Reserved	111 = Reserved											
Rx Timeout Interrupt Interval	[15:12]	RW	<p>Rx Timeout Interrupt Interval Rx interrupt occurs if no data is received during the word time specified in this field. The default value is 3-word time and the word time is affected by the 1 frame bit length which is configurable by ULCONn.</p>	0x3								
RX Time-out with empty RX FIFO ⁽⁴⁾	[11]	R/W	<p>It enables RX time-out feature when RX FIFO counter is 0. This bit is valid only when UCONn[7] is 1. 0 = RX time-out feature is disabled when RX FIFO is empty. 1 = RX time-out feature is enabled when RX FIFO is empty.</p>	0								
RX Time-out DMA suspend enable	[10]	R/W	<p>It enables that RX DMA FSM suspends when RX Time-out occurs. 0 = Disable suspending RX DMA FSM 1 = Enable suspending RX DMA FSM</p>	0								
Tx Interrupt Type	[9]	RW	<p>Interrupt request type. ⁽²⁾ 0 = Pulse (Interrupt is requested when the Tx buffer is empty in the Non-FIFO mode or when it reaches Tx FIFO Trigger Level in the FIFO mode.) 1 = Level (Interrupt is requested when Tx buffer is empty in the Non-FIFO mode or when it reaches Tx FIFO Trigger Level in the FIFO mode.)</p>	0								
Rx Interrupt Type	[8]	RW	<p>Interrupt request type. ⁽²⁾ 0 = Pulse (Interrupt is requested when instant Rx buffer receives data in the Non-FIFO mode or when it reaches Rx FIFO Trigger Level in the FIFO mode.) 1 = Level (Interrupt is requested when Rx buffer is receiving data in</p>	0								

Name	Bit	Type	Description	Reset Value
			the Non-FIFO mode or when it reaches Rx FIFO Trigger Level in the FIFO mode.)	
Rx Time Out Enable	[7]	RW	Enables/ Disables Rx time-out interrupts if UART FIFO is enabled. The interrupt is a receive interrupt. 0 = Disables 1 = Enables	0
Rx Error Status Interrupt Enable	[6]	RW	Enables the UART to generate an interrupt upon an exception, such as a break, frame error, parity error, or overrun error during a receive operation. 0 = Does not generate receive error status interrupt. 1 = Generates receive error status interrupt.	0
Loop-back Mode	[5]	RW	Setting loop-back bit to 1 trigger the UART to enter the loop-back mode. This mode is provided for test purposes only. 0 = Normal operation 1 = Loop-back mode	0
Send Break Signal	[4]	RW	Setting this bit trigger the UART to send a break during 1 frame time. This bit is automatically cleared after sending the break signal. 0 = Normal transmit 1 = Sends the break signal	0
Transmit Mode	[3:2]	RW	Determines which function is able to write Tx data to the UART transmit buffer. 00 = Disables 01 = Interrupt request or polling mode 10 = DMA mode 11 = Reserved	00
Receive Mode	[1:0]	RW	Determines which function is able to read data from UART receive buffer. 00 = Disables 01 = Interrupt request or polling mode 10 = DMA mode 11 = Reserved	00

NOTE:

1. DIV_VAL = UBRDIVn + UFRACVAL/16. Refer to [13.4.1.11](#) and [13.4.1.12](#).
2. Exynos4210 use a level-triggered interrupt controller. Therefore, these bits must be set to 1 for every transfer.
3. If the UART does not reach the FIFO trigger level and does not receive data during the time specified at the 'Rx Timeout Interrupt Interval' field in DMA receive mode with FIFO, the Rx interrupt is generated (receive time out). You must check the FIFO status and read out the rest.
4. Both UCONn[11] and UCONn[7] should be set to 1 if you want to enable RX time-out feature when RX FIFO counter is 0.

13.4.1.3 UFCONn (n = 0 to 4)

- Address = Base Address + 0x0008, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value																														
RSVD	[31:11]	-	Reserved	0																														
Tx FIFO Trigger Level	[10:8]	RW	<p>Determines the trigger level of Tx FIFO. If data count of Tx FIFO is less than or equal to the trigger level, Tx interrupt occurs.</p> <table> <tr><td>[Channel 0]</td><td></td></tr> <tr><td>000 = 0 byte</td><td>001 = 32 bytes</td></tr> <tr><td>010 = 64 bytes</td><td>011 = 96 bytes</td></tr> <tr><td>100 = 128 bytes</td><td>101 = 160 bytes</td></tr> <tr><td>110 = 192 bytes</td><td>111 = 224 bytes</td></tr> <tr><td>[Channel 1, 4]</td><td></td></tr> <tr><td>000 = 0 byte</td><td>001 = 8 bytes</td></tr> <tr><td>010 = 16 bytes</td><td>011 = 24 bytes</td></tr> <tr><td>100 = 32 bytes</td><td>101 = 40 bytes</td></tr> <tr><td>110 = 48 bytes</td><td>111 = 56 bytes</td></tr> <tr><td>[Channel 2, 3]</td><td></td></tr> <tr><td>000 = 0 byte</td><td>001 = 2 bytes</td></tr> <tr><td>010 = 4 bytes</td><td>011 = 6 bytes</td></tr> <tr><td>100 = 8 bytes</td><td>101 = 10 bytes</td></tr> <tr><td>110 = 12 bytes</td><td>111 = 14 bytes</td></tr> </table>	[Channel 0]		000 = 0 byte	001 = 32 bytes	010 = 64 bytes	011 = 96 bytes	100 = 128 bytes	101 = 160 bytes	110 = 192 bytes	111 = 224 bytes	[Channel 1, 4]		000 = 0 byte	001 = 8 bytes	010 = 16 bytes	011 = 24 bytes	100 = 32 bytes	101 = 40 bytes	110 = 48 bytes	111 = 56 bytes	[Channel 2, 3]		000 = 0 byte	001 = 2 bytes	010 = 4 bytes	011 = 6 bytes	100 = 8 bytes	101 = 10 bytes	110 = 12 bytes	111 = 14 bytes	000
[Channel 0]																																		
000 = 0 byte	001 = 32 bytes																																	
010 = 64 bytes	011 = 96 bytes																																	
100 = 128 bytes	101 = 160 bytes																																	
110 = 192 bytes	111 = 224 bytes																																	
[Channel 1, 4]																																		
000 = 0 byte	001 = 8 bytes																																	
010 = 16 bytes	011 = 24 bytes																																	
100 = 32 bytes	101 = 40 bytes																																	
110 = 48 bytes	111 = 56 bytes																																	
[Channel 2, 3]																																		
000 = 0 byte	001 = 2 bytes																																	
010 = 4 bytes	011 = 6 bytes																																	
100 = 8 bytes	101 = 10 bytes																																	
110 = 12 bytes	111 = 14 bytes																																	
RSVD	[7]	-	Reserved	0																														
Rx FIFO Trigger Level	[6:4]	RW	<p>Determines the trigger level of Rx FIFO. If data count of Rx FIFO is more than or equal to the trigger level, Rx interrupt occurs.</p> <table> <tr><td>[Channel 0]</td><td></td></tr> <tr><td>000 = 32 byte</td><td>001 = 64 bytes</td></tr> <tr><td>010 = 96 bytes</td><td>011 = 128 bytes</td></tr> <tr><td>100 = 160 bytes</td><td>101 = 192 bytes</td></tr> <tr><td>110 = 224 bytes</td><td>111 = 256 bytes</td></tr> <tr><td>[Channel 1, 4]</td><td></td></tr> <tr><td>000 = 8 byte</td><td>001 = 16 bytes</td></tr> <tr><td>010 = 24 bytes</td><td>011 = 32 bytes</td></tr> <tr><td>100 = 40 bytes</td><td>101 = 48 bytes</td></tr> <tr><td>110 = 56 bytes</td><td>111 = 64 bytes</td></tr> <tr><td>[Channel 2, 3]</td><td></td></tr> <tr><td>000 = 2 byte</td><td>001 = 4 bytes</td></tr> <tr><td>010 = 6 bytes</td><td>011 = 8 bytes</td></tr> <tr><td>100 = 10 bytes</td><td>101 = 12 bytes</td></tr> <tr><td>110 = 14 bytes</td><td>111 = 16 bytes</td></tr> </table>	[Channel 0]		000 = 32 byte	001 = 64 bytes	010 = 96 bytes	011 = 128 bytes	100 = 160 bytes	101 = 192 bytes	110 = 224 bytes	111 = 256 bytes	[Channel 1, 4]		000 = 8 byte	001 = 16 bytes	010 = 24 bytes	011 = 32 bytes	100 = 40 bytes	101 = 48 bytes	110 = 56 bytes	111 = 64 bytes	[Channel 2, 3]		000 = 2 byte	001 = 4 bytes	010 = 6 bytes	011 = 8 bytes	100 = 10 bytes	101 = 12 bytes	110 = 14 bytes	111 = 16 bytes	000
[Channel 0]																																		
000 = 32 byte	001 = 64 bytes																																	
010 = 96 bytes	011 = 128 bytes																																	
100 = 160 bytes	101 = 192 bytes																																	
110 = 224 bytes	111 = 256 bytes																																	
[Channel 1, 4]																																		
000 = 8 byte	001 = 16 bytes																																	
010 = 24 bytes	011 = 32 bytes																																	
100 = 40 bytes	101 = 48 bytes																																	
110 = 56 bytes	111 = 64 bytes																																	
[Channel 2, 3]																																		
000 = 2 byte	001 = 4 bytes																																	
010 = 6 bytes	011 = 8 bytes																																	
100 = 10 bytes	101 = 12 bytes																																	
110 = 14 bytes	111 = 16 bytes																																	
RSVD	[3]	-	Reserved	0																														
Tx FIFO Reset	[2]	RW	Auto-clears after resetting FIFO 0 = Normal 1 = Tx FIFO reset	0																														
Rx FIFO	[1]	RW	Auto-clears after resetting FIFO	0																														

Name	Bit	Type	Description	Reset Value
Reset			0 = Normal 1 = Rx FIFO reset	
FIFO Enable	[0]	RW	0 = Disables 1 = Enables	0

NOTE: If the UART does not reach the FIFO trigger level and does not receive data during the specified word time in DMA receive mode with FIFO, the Rx interrupt will be generated (receive time out). You must check the FIFO status and read out the rest.

13.4.1.4 UMCONn (n = 0, 1, 2, 4)

- Address = Base Address + 0x000C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value																								
RSVD	[31:8]	-	Reserved	0																								
RTS trigger Level	[7:5]	RW	<p>Determines the trigger level of Rx FIFO to control nRTS signal. If AFC bit is enabled and Rx FIFO have bytes that are greater than or equal to the trigger level, nRTS signal is deactivated.</p> <p>[Channel 0]</p> <table> <tr><td>000 = 255 bytes</td><td>001 = 224 bytes</td></tr> <tr><td>010 = 192 bytes</td><td>011 = 160 bytes</td></tr> <tr><td>100 = 128 bytes</td><td>101 = 96 bytes</td></tr> <tr><td>110 = 64 bytes</td><td>111 = 32 bytes</td></tr> </table> <p>[Channel 1, 4]</p> <table> <tr><td>000 = 63 bytes</td><td>001 = 56 bytes</td></tr> <tr><td>010 = 48 bytes</td><td>011 = 40 bytes</td></tr> <tr><td>100 = 32 bytes</td><td>101 = 24 bytes</td></tr> <tr><td>110 = 16 bytes</td><td>111 = 8 bytes</td></tr> </table> <p>[Channel 2]</p> <table> <tr><td>000 = 15 bytes</td><td>001 = 14 bytes</td></tr> <tr><td>010 = 12 bytes</td><td>011 = 10 bytes</td></tr> <tr><td>100 = 8 bytes</td><td>101 = 6 bytes</td></tr> <tr><td>110 = 4 bytes</td><td>111 = 2 bytes</td></tr> </table>	000 = 255 bytes	001 = 224 bytes	010 = 192 bytes	011 = 160 bytes	100 = 128 bytes	101 = 96 bytes	110 = 64 bytes	111 = 32 bytes	000 = 63 bytes	001 = 56 bytes	010 = 48 bytes	011 = 40 bytes	100 = 32 bytes	101 = 24 bytes	110 = 16 bytes	111 = 8 bytes	000 = 15 bytes	001 = 14 bytes	010 = 12 bytes	011 = 10 bytes	100 = 8 bytes	101 = 6 bytes	110 = 4 bytes	111 = 2 bytes	000
000 = 255 bytes	001 = 224 bytes																											
010 = 192 bytes	011 = 160 bytes																											
100 = 128 bytes	101 = 96 bytes																											
110 = 64 bytes	111 = 32 bytes																											
000 = 63 bytes	001 = 56 bytes																											
010 = 48 bytes	011 = 40 bytes																											
100 = 32 bytes	101 = 24 bytes																											
110 = 16 bytes	111 = 8 bytes																											
000 = 15 bytes	001 = 14 bytes																											
010 = 12 bytes	011 = 10 bytes																											
100 = 8 bytes	101 = 6 bytes																											
110 = 4 bytes	111 = 2 bytes																											
Auto Flow Control (AFC)	[4]	RW	0 = Disables 1 = Enables	0																								
Modem Interrupt Enable	[3]	RW	0 = Disables 1 = Enables	0																								
RSVD	[2:1]	RW	These bits must be 0	00																								
Request to Send	[0]	RW	<p>If AFC bit is enabled, this value will be ignored. In this case the Exynos4210 controls nRTS signals automatically.</p> <p>If AFC bit is disabled, the software must control nRTS signal.</p> <p>0 = "H" level (Inactivate nRTS) 1 = "L" level (Activate nRTS)</p>	0																								

NOTE:

- UART 2 supports AFC function, if nRxD3 and nTxD3 are set as nRTS2 and nCTS2 by GPA1CON.
UART 3 does not support AFC function, because the Exynos4210 has no nRTS3 and nCTS3.
- In AFC mode, RX FIFO trigger level should be set to lower than RTS trigger level, because transmitter stops data transfer when it gets deactivated nRST signal.

13.4.1.5 UTRSTATn (n = 0 to 4)

- Address = Base Address + 0x0010, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	-	Reserved	0
RX FIFO count in RX time-out status	[23:16]	R	RX FIFO counter capture value when RX time-out occurs	0x00
TX DMA FSM State	[15:12]	R	Current State of TX DMA FSM 0x0 = IDLE 0x1 = Burst Request 0x2 = Burst Acknowledgement 0x3 = Burst Next (intermediate state for next request) 0x4 = Single Request 0x5 = Single Acknowledgement 0x6 = Single Next (intermediate state for next request) 0x7 = Last Burst Request 0x8 = Last Burst Acknowledgement 0x9 = Last Single Request 0x10 = Last Single Acknowledgement	0x0
RX DMA FSM State	[11:8]	R	Current State of RX DMA FSM 0x0 = IDLE 0x1 = Burst Request 0x2 = Burst Acknowledgement 0x3 = Burst Next (intermediate state for next request) 0x4 = Single Request 0x5 = Single Acknowledgement 0x6 = Single Next (intermediate state for next request) 0x7 = Last Burst Request 0x8 = Last Burst Acknowledgement 0x9 = Last Single Request 0x10 = Last Single Acknowledgement	0x0
RSVD	[7:4]	-	Reserved	0
RX Time-out status/Clear ¹	[3]	R/W	RX Time-out status when read. 0 = RX Time out did not occur 1 = RX Time out. Clear RX Time-out status when write. 0 = No operation 1 = Clears RX Time-out status NOTE: If UCONn[10] is set to 1, writing 1 to this bit resumes Rx DMA FSM which was suspended when Rx time-out occurred.	0
Transmitter empty	[2]	R	This bit is automatically set to 1 if the transmit buffer has no valid data to transmit, and the transmit shift is empty.	1

Name	Bit	Type	Description	Reset Value
			0 = Not empty 1 = Transmitter (which includes transmit buffer and shifter) empty	
Transmit buffer empty	[1]	R	This bit is automatically set to 1 if transmit buffer is empty. 0 = Buffer is not empty 1 = Buffer is empty (In Non-FIFO mode, Interrupt or DMA is requested. In FIFO mode, Interrupt or DMA is requested, if Tx FIFO Trigger Level is set to 00 (Empty)) If UART uses FIFO, check Tx FIFO Count bits and Tx FIFO Full bit in UFSTAT instead of this bit.	1
Receive buffer data ready	[0]	R	This bit is automatically set to 1 if receive buffer contains valid data, received over the RXDn port. 0 = Buffer is empty 1 = Buffer has a received data (In Non-FIFO mode, Interrupt or DMA is requested) If UART uses the FIFO, check Rx FIFO Count bits and Rx FIFO Full bit in UFSTAT instead of this bit.	0

13.4.1.6 UERSTATn (n = 0 to 4)

- Address = Base Address + 0x0014, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	—	Reserved	0
Break Detect	[3]	R	This bit is automatically set to 1 to indicate that a break signal has been received. 0 = No break signal is received 1 = Break signal is received (Interrupt is requested.)	0
Frame Error	[2]	R	This bit is automatically set to 1 if a frame error occurs during the receive operation. 0 = No frame error occurs during the receive operation 1 = Frame error occurs (Interrupt is requested.) during the receive operation	0
Parity Error	[1]	R	This bit is automatically set to 1 if a parity error occurs during the receive operation. 0 = No parity error occurs during receive the receive operation 1 = Parity error occurs (Interrupt is requested.) the receive operation	0
Overrun Error	[0]	R	This bit is automatically set to 1 automatically if an overrun error occurs during the receive operation. 0 = No overrun error occurs during the receive operation 1 = Overrun error occurs (Interrupt is requested.) during the receive operation	0

NOTE: These bits (UERSATn[3:0]) are automatically cleared to 0 if UART error status is read.

13.4.1.7 UFSTATn (n = 0 to 4)

- Address = Base Address + 0x0018, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:25]	—	Reserved	0
Tx FIFO Full	[24]	R	This bit is automatically set to 1 if the transmitted FIFO is full during transmit operation 0 = Not full 1 = Full	0
Tx FIFO Count	[23:16]	R	Number of data in Tx FIFO NOTE: This field is set to 0 if Tx FIFO is full.	0
RSVD	[15:10]	—	Reserved	0
Rx FIFO Error	[9]	R	This bit is set to 1 if Rx FIFO contains invalid data which results from frame error, parity error, or break signal.	0
Rx FIFO Full	[8]	R	This bit is automatically set to 1 if the received FIFO is full during receive operation 0 = Not full 1 = Full	0
Rx FIFO Count	[7:0]	R	Number of data in Rx FIFO NOTE: This field is set to 0 if Rx FIFO is full.	0

13.4.1.8 UMSTAT_n ($n = 0, 1, 2, 4$)

- Address = Base Address + 0x001C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	-	Reserved	0
Delta CTS	[4]	R	This bit indicates that the nCTS input to the Exynos4210 has changed its state since the last time it was read by CPU. (Refer Figure 13-9) 0 = Has not changed 1 = Has changed NOTE: In UMSTAT4, reset value of this bit is undefined. It depends on the GPIO configuration of GPS.	0
RSVD	[3:1]	-	Reserved	-
Clear to Send	[0]	R	0 = CTS signal is not activated (nCTS pin is high) 1 = CTS signal is activated (nCTS pin is low) NOTE: In UMSTAT4, reset value of this bit is undefined. It depends on the GPIO configuration of GPS.	0

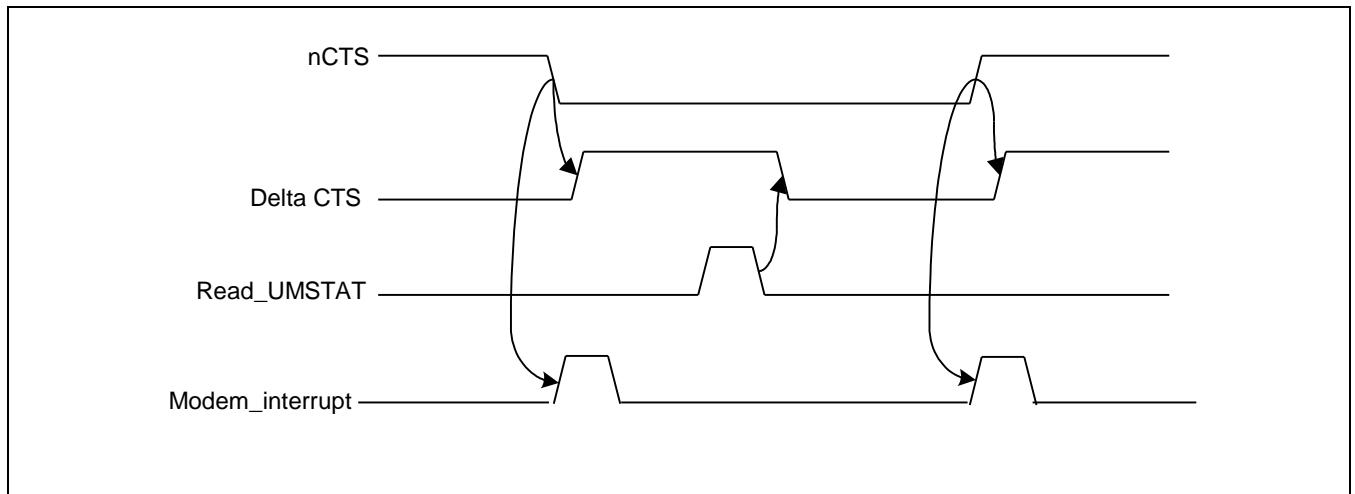


Figure 13-9 nCTS and Delta CTS Timing Diagram

13.4.1.9 UTXHn (n = 0 to 4)

- Address = Base Address + 0x0020, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	–	Reserved	-
UTXHn	[7:0]	W	Transmit data for UARTn	-

13.4.1.10 URXHn (n = 0 to 4)

- Address = Base Address + 0x0024, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:8]	–	Reserved	0
URXHn	[7:0]	R	Receive data for UARTn	0x00

NOTE: If an overrun error occurs, the URXHn must be read. If not, the next received data makes an overrun error, even though the overrun bit of UERSTATn had been cleared.

13.4.1.11 UBRDIV n (n = 0 to 4)

- Address = Base Address + 0x0028, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	–	Reserved	0
UBRDIVn	[15:0]	RW	Baud rate division value NOTE: UBRDIV should be greater than 0.	0x0000

13.4.1.12 UFRACVALn (n = 0 to 4)

- Address = Base Address + 0x002C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	–	Reserved	0
UFRACVALn	[3:0]	RW	Determine the fractional part of baud rate divisor.	0x0

1. UART Baud Rate Configuration

The value stored in the baud rate divisor (UBRDIVn) and divisor fractional value (UFRACVALn) is used to determine the serial Tx/Rx clock rate (baud rate) as follows:

$$\text{DIV_VAL} = \text{UBRDIVn} + \text{UFRACVALn}/16$$

Or

$$\text{DIV_VAL} = (\text{SCLK_UART} / (\text{bps} \times 16)) - 1$$

Where, the divisor should be from 1 to (2¹⁶-1).

Using UFRACVALn, you can generate the baud rate more accurately.

For example, if the baud-rate is 115200 bps and SCLK_UART is 40 MHz, UBRDIVn and UFRACVALn are:

$$\text{DIV_VAL} = (40000000 / (115200 \times 16)) - 1$$

$$= 21.7 - 1$$

$$= 20.7$$

UBRDIVn = 20 (integer part of DIV_VAL)

UFRACVALn/16 = 0.7

so, UFRACVALn = 11

2. Baud Rate Error Tolerance

UART Frame error should be less than 1.87% (3/160)

$$t_{UPCLK} = (\text{UBRDIVn} + 1 + \text{UFRACVAL} / 16) \times 16 \times 1\text{Frame} / \text{SCLK_UART}$$

tUPCLK: Real UART Clock

tEXTUARTCLK = 1Frame / baud-rate

tEXTUARTCLK: Ideal UART Clock

$$\text{UART error} = (t_{UPCLK} - t_{EXTUARTCLK}) / t_{EXTUARTCLK} \times 100\%$$

* 1Frame = start bit + data bit + parity bit + stop bit.

3. UART Clock and PCLK Relation

There is a constraint on the ratio of clock frequencies for PCLK to UARTCLK.

The frequency of UARTCLK must be no more than 5.5/3 times faster than the frequency of PCLK:

$$\text{FUARTCLK} \leq 5.5/3 \times \text{FPCLK}$$

$$\text{FUARTCLK} = \text{baudrate} \times 16$$

This allows sufficient time to write the received data to the receive FIFO

13.4.1.13 UINTPn (n = 0 to 4)

- Address = Base Address + 0x0030, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	—	Reserved	0
MODEM	[3]	RW	Generates Modem interrupt.	0
TXD	[2]	RW	Generates Transmit interrupt.	0
ERROR	[1]	RW	Generates Error interrupt.	0
RXD	[0]	RW	Generates Receive interrupt.	0

Interrupt pending contains the information of the interrupts that are generated

If one of above 4 bits is logical high ('1'), each UART channel generates interrupt.

This must be cleared in the interrupt service routine after clearing interrupt pending in Interrupt Controller (INTC). Clear specific bits of UINTP by writing 1's to the bits that you want to clear.

13.4.1.14 UINTSPn (n = 0 to 4)

- Address = Base Address + 0x0034, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	—	Reserved	0
MODEM	[3]	RW	Generates Modem interrupt.	0
TXD	[2]	RW	Generates Transmit interrupt.	0
ERROR	[1]	RW	Generates Error interrupt.	0
RXD	[0]	RW	Generates Receive interrupt.	0

Interrupt Source Pending contains the information of the interrupt that are generated regardless of the value of Interrupt Mask.

13.4.1.15 UINTM_n (n = 0 to 4)

- Address = Base Address + 0x0038, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	-	Reserved	0
MODEM	[3]	RW	Mask Modem interrupt.	0
TXD	[2]	RW	Mask Transmit interrupt.	0
ERROR	[1]	RW	Mask Error interrupt.	0
RXD	[0]	RW	Mask Receive interrupt.	0

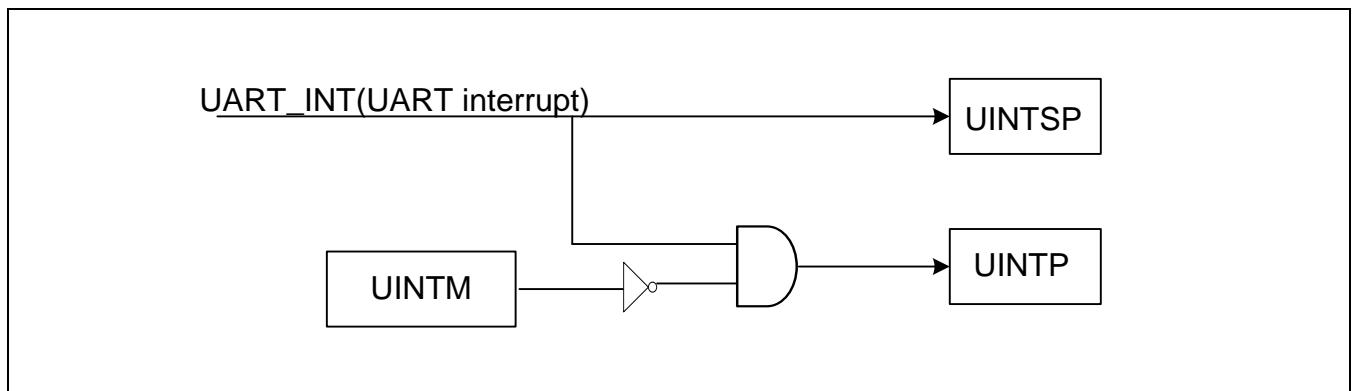


Figure 13-10 Block diagram of UINTSP, UINTP and UINTM

Interrupt mask contains the information about which interrupt source is masked. If a specific bit is set to 1, interrupt request signal to the Interrupt Controller is not generated even though corresponding interrupt is generated.

NOTE: Even in such a case, the corresponding bit of UINTSP_n is set to 1). If the mask bit is 0, the interrupt requests are serviced from the corresponding interrupt source.

14 Inter-Integrated Circuit (IIC) Bus Interface

14.1 Overview

The Exynos4210 RISC microprocessor supports four multi-master I²C bus serial interfaces. To carry information between bus masters and peripheral devices connected to the I²C bus, a dedicated Serial Data Line (SDA) and an Serial Clock Line (SCL) is used. Both SDA and SCL lines are bi-directional.

In multi-master I²C-bus mode, multiple Exynos4210 RISC microprocessors receive or transmit serial data to or from slave devices. The master Exynos4210 initiates and terminates a data transfer over the I²C bus. The I²C bus in the Exynos4210 uses a standard bus arbitration procedure.

To control multi-master I²C-bus operations, values must be written to the following registers:

- Multi-master I²C-bus control register- I2CCON
- Multi-master I²C-bus control/status register- I2CSTAT
- Multi-master I²C-bus Tx/Rx data shift register- I2CDS
- Multi-master I²C-bus address register- I2CADD

If the I²C-bus is free, both SDA and SCL lines should be both at High level. A High-to-Low transition of SDA initiates a Start condition. A Low-to-High transition of SDA initiates a Stop condition while SCL remains steady at High Level.

The master device always generates Start and Stop conditions. First 7-bit address value in the data byte that is transferred via SDA line after the Start condition has been initiated, can determine the slave device which the bus master device has selected. The 8th bit determines the direction of the transfer (read or write).

Every data byte put onto the SDA line should be eight bits in total. There is no limit to send or receive bytes during the bus transfer operation. Data is always sent from most-significant bit (MSB) first, and every byte should be immediately followed by acknowledge (ACK) bit.

14.2 Features

- 9 channels Multi-Master, Slave I²C BUS interfaces
(8 channels for general purpose, 1 channel for HDMI dedicated)
- 7-bit addressing mode
- Serial, 8-bit oriented, and bidirectional data transfer
- Supports up to 100kbit/s in the Standard mode
- Supports up to 400kbit/s in the Fast mode.
- Supports master transmit, master receive, slave transmit and slave receive operation
- Supports interrupt or polling events.

14.3 Functional Description

14.3.1 Block Diagram

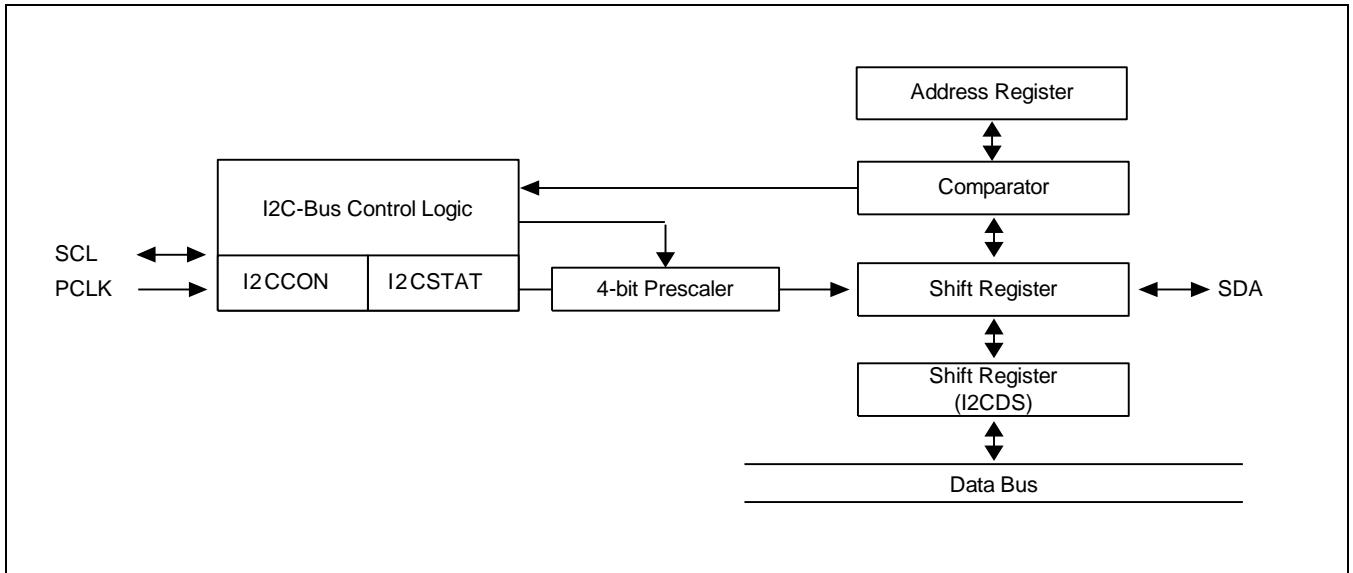


Figure 14-1 I²C-Bus Block Diagram

14.3.2 Interface Port Description

Signal	I/O	Description	Pad	Type
I2C_0_SCL	Input/Output	I ² C-Bus Interface0 Serial Clock Line	Xi2c0SCL	muxed
I2C_0_SDA	Input/Output	I ² C-Bus Interface0 Serial Data Line	Xi2c0SDA	muxed
I2C_1_SCL	Input/Output	I ² C-BUS Interface1 Serial Clock Line	Xi2c1SCL	muxed
I2C_1_SDA	Input/Output	I ² C-BUS Interface1 Serial Data Line	Xi2c1SDA	muxed
I2C_2_SCL	Input/Output	I ² C-BUS Interface2 Serial Clock Line	XuRTSn_1	muxed
I2C_2_SDA	Input/Output	I ² C-BUS Interface2 Serial Data Line	XuCTSn_1	muxed
I2C_3_SCL	Input/Output	I ² C-BUS Interface3 Serial Clock Line	XuRTSn_2	muxed
I2C_3_SDA	Input/Output	I ² C-BUS Interface3 Serial Data Line	XuCTSn_2	muxed
I2C_4_SCL	Input/Output	I ² C-BUS Interface4 Serial Clock Line	XspiMOSI_0	muxed
I2C_4_SDA	Input/Output	I ² C-BUS Interface4 Serial Data Line	XspiMISO_0	muxed
I2C_5_SCL	Input/Output	I ² C-BUS Interface5 Serial Clock Line	XspiMOSI_1	muxed
I2C_5_SDA	Input/Output	I ² C-BUS Interface5 Serial Data Line	XspiMISO_1	muxed
I2C_6_SCL	Input/Output	I ² C-BUS Interface6 Serial Clock Line	Xi2s2SDO	muxed
I2C_6_SDA	Input/Output	I ² C-BUS Interface6 Serial Data Line	Xi2s2SDI	muxed
I2C_7_SCL	Input/Output	I ² C-BUS Interface7 Serial Clock Line	XpwmTOUT_3	muxed
I2C_7_SDA	Input/Output	I ² C-BUS Interface7 Serial Data Line	XpwmTOUT_2	muxed

NOTE: I²C-BUS Interface for HDMI PHY is internally connected

14.3.3 IIC-Bus Interface Operation

The Exynos4210 I²C-bus interface has four operation modes, namely:

- Master Transmitter Mode
- Master Receive Mode
- Slave Transmitter Mode
- Slave Receive Mode

14.3.4 Start and Stop Conditions

If the I²C-bus interface is inactive, it is usually in Slave mode. In other words, the interface should be in Slave mode before detecting a Start condition on the SDA line (a Start condition is initiated with a High-to-Low transition of the SDA line while the clock signal of SCL is High). If the interface state is changed to Master mode, SDA line initiates data transfer and generates SCL signal.

A Start condition transfers one-byte serial data via SDA line, and a Stop condition terminates the data transfer. A Stop condition is a Low-to-High transition of the SDA line while SCL is High. The master generates Start and Stop conditions. The I²C-bus gets busy if a Start condition is generated. On the other hand, a Stop condition frees the I²C-bus.

If a master initiates a Start condition, it should send a slave address to notify the slave device. One byte of address field consists of a 7-bit address and a 1-bit transfer direction indicator (that shows write or read).

If bit 8 is 0, it indicates a write operation (Transmit Operation); if bit 8 is 1, it indicates a request for data read (Receive Operation).

The master transmits Stop condition to complete the transfer operation. If the master wants to continue the data transmission to the bus, it should generate another Start condition as well as a slave address. In this way, the read-write operation is performed in various formats.

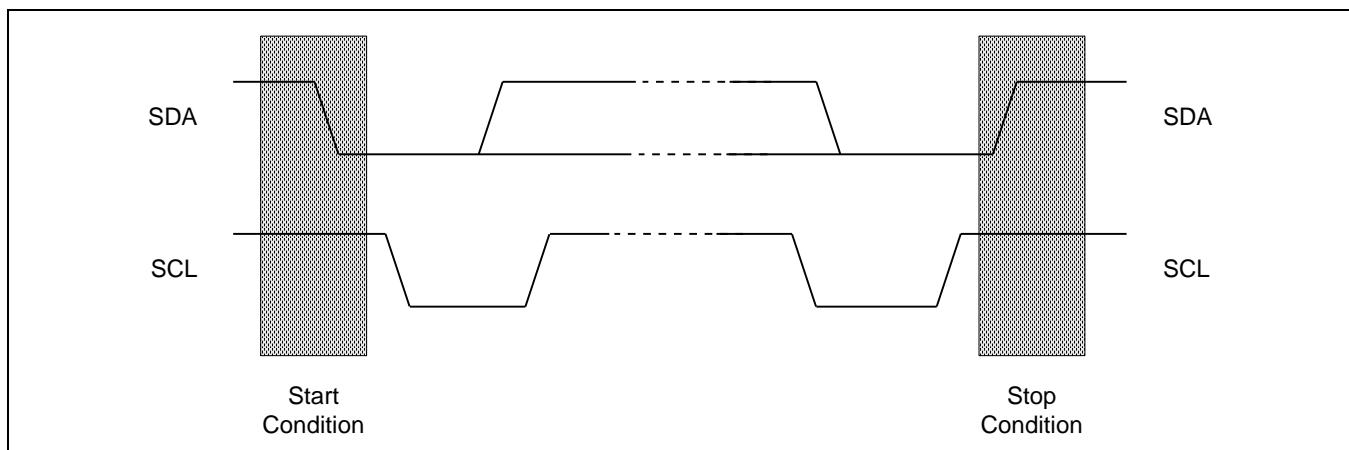


Figure 14-2 Start and Stop Condition

14.3.5 Data Transfer Format

Every byte placed on the SDA line should be eight bits in length. There is no limit to transmit bytes per transfer. The first byte following a Start condition should have the address field. If the I²C-bus is operating in Master mode, master transmits the address field. Each byte should be followed by an acknowledgement (ACK) bit. The MSB bit of the serial data and addresses are sent first.

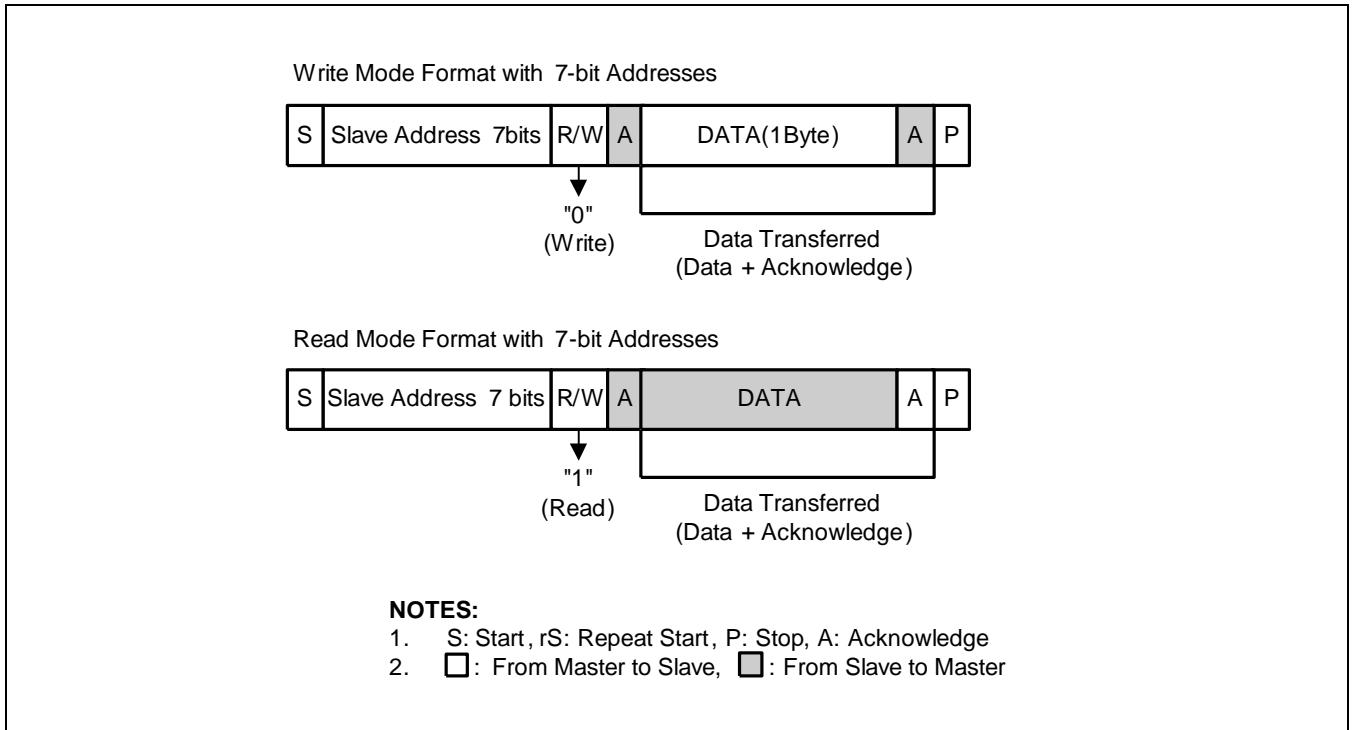


Figure 14-3 I²C-Bus Interface Data Format

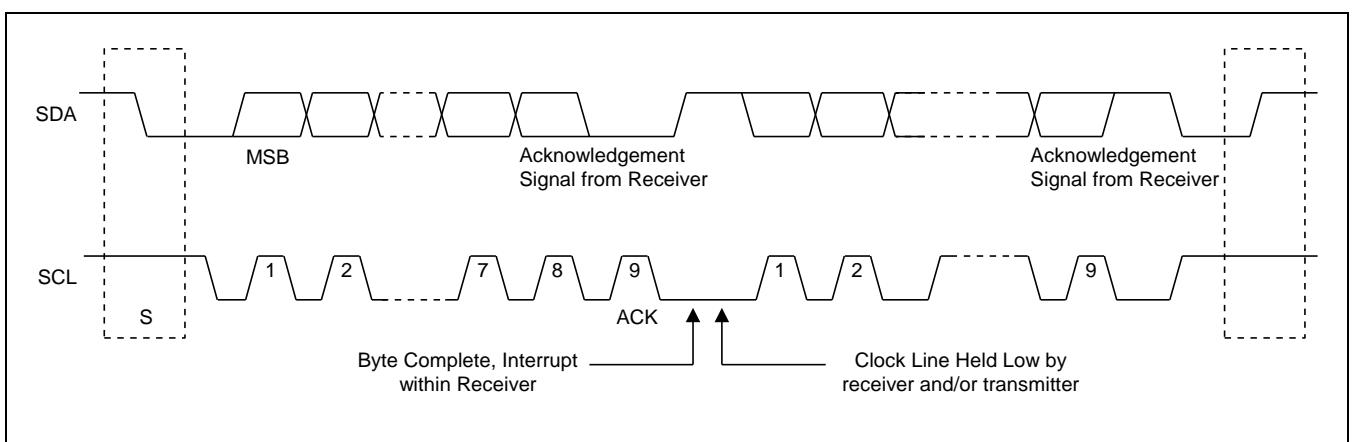


Figure 14-4 Data Transfer on the I²C-Bus

14.3.6 ACK Signal Transmission

To complete a one-byte transfer operation, the receiver sends an ACK bit to the transmitter. The ACK pulse occurs at the ninth clock of the SCL line. Eight clocks are required for the one-byte data transfer. The master generates clock pulse required to transmit the ACK bit.

The transmitter sets the SDA line to High to release the SDA line if the ACK clock pulse is received. The receiver drives the SDA line Low during the ACK clock pulse so that the SDA keeps Low during the High period of the ninth SCL pulse. The software (I2CSTAT) enables or disables ACK bit transmit function. However, the ACK pulse on the ninth clock of SCL is required to complete the one-byte data transfer operation.

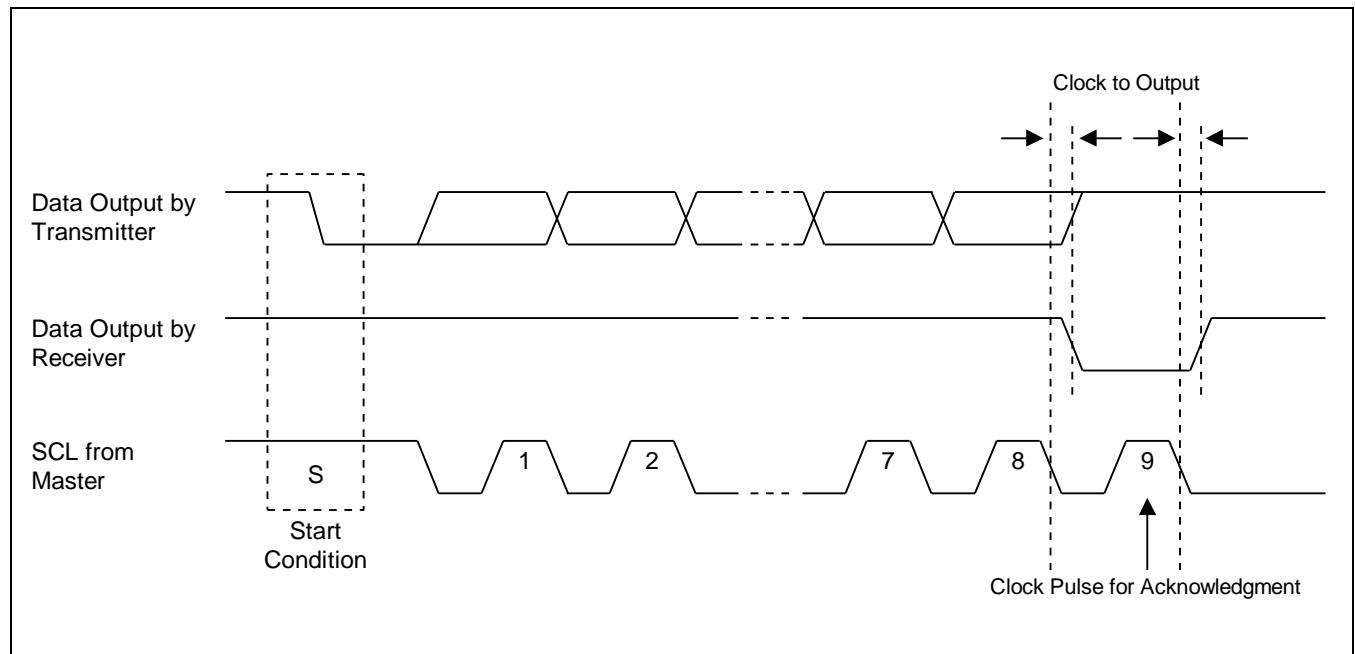


Figure 14-5 Acknowledge on the I2C-Bus

14.3.7 Read-Write Operation

If data is transmitted in Transmitter mode, the I²C-bus interface waits until I²C-bus Data Shift (I2CDS) register receives the new data. Before the new data is written to the register, the SCL line is held low. The line is only released after the data has been written. Exynos4210 holds the interrupt to identify the completion of current data transfer. After the CPU receives the interrupt request, it writes new data to the I2CDS register again.

If data is received in Receive mode, the I²C-bus interface waits until I2CDS register is read. Before the new data is read out, the SCL line is held low. The line is only released after the data has been read. Exynos4210 holds the interrupt to identify the completion of new data reception. After the CPU receives the interrupt request, it reads the data from the I2CDS register.

14.3.8 Bus Arbitration Procedures

Arbitration takes place on the SDA line to prevent the contention on the bus between two masters. If a master with a SDA High level detects other master with a SDA active Low level, it does not initiate a data transfer because the current level on the bus does not correspond to its own. The arbitration procedure extends until the SDA line turns high.

If the masters lower the SDA line simultaneously, each master evaluates whether the mastership is allocated itself or not. For the purpose of evaluation each master detects the address bits. While each master generates the slave address, it detects the address bit on the SDA line because the SDA line is likely to get Low rather than high.

Assume that one master generates a Low as first address bit, while the other master is maintaining High. In this case, both masters detect Low on the bus because the Low status is superior to the High status in power. If this happens, Low (as the first bit of address) generating master gets the mastership while High (as the first bit of address) generating master withdraws the mastership. If both masters generate Low as the first bit of address, there is arbitration for the second address bit again. This arbitration continues to the end of last address bit.

14.3.9 Abort Conditions

If a slave receiver cannot acknowledge the confirmation of the slave address, it holds the level of the SDA line High. In this case, the master generates a Stop condition and cancels the transfer.

If a master receiver is involved in the aborted transfer, it signals the end of slave transmit operation by canceling the generation of an ACK after the last data byte received from the slave. The slave transmitter releases the SDA to allow a master to generate a Stop condition.

14.3.10 Configuring I²C-Bus

To control the frequency of the serial clock (SCL), the 4-bit prescaler value is programmed in the I2CCON register. The I²C-bus interface address is stored in the I²C-bus address (I2CADD) register (By default, the I²C-bus interface address has an unknown value).

14.3.11 Flowcharts of Operations in Each Mode

Following steps must be executed before any I²C Tx/Rx operations:

1. If required, write own slave address on I2CADD register.
2. Set I2CCON register.
 - a) Enable interrupt
 - b) Define SCL period
3. Set I2CSTAT to enable Serial Output

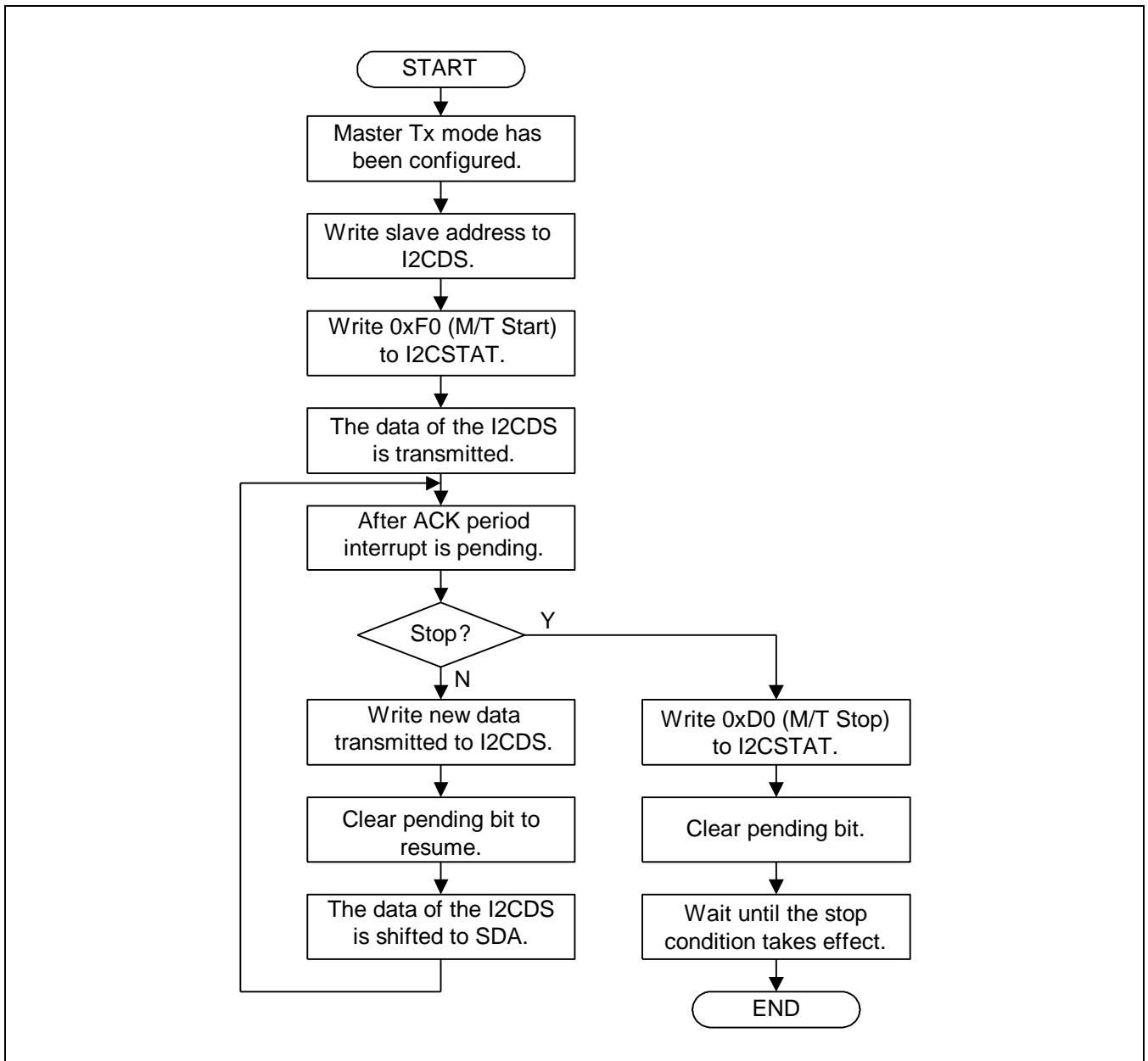


Figure 14-6 Operations for Master/Transmitter Mode

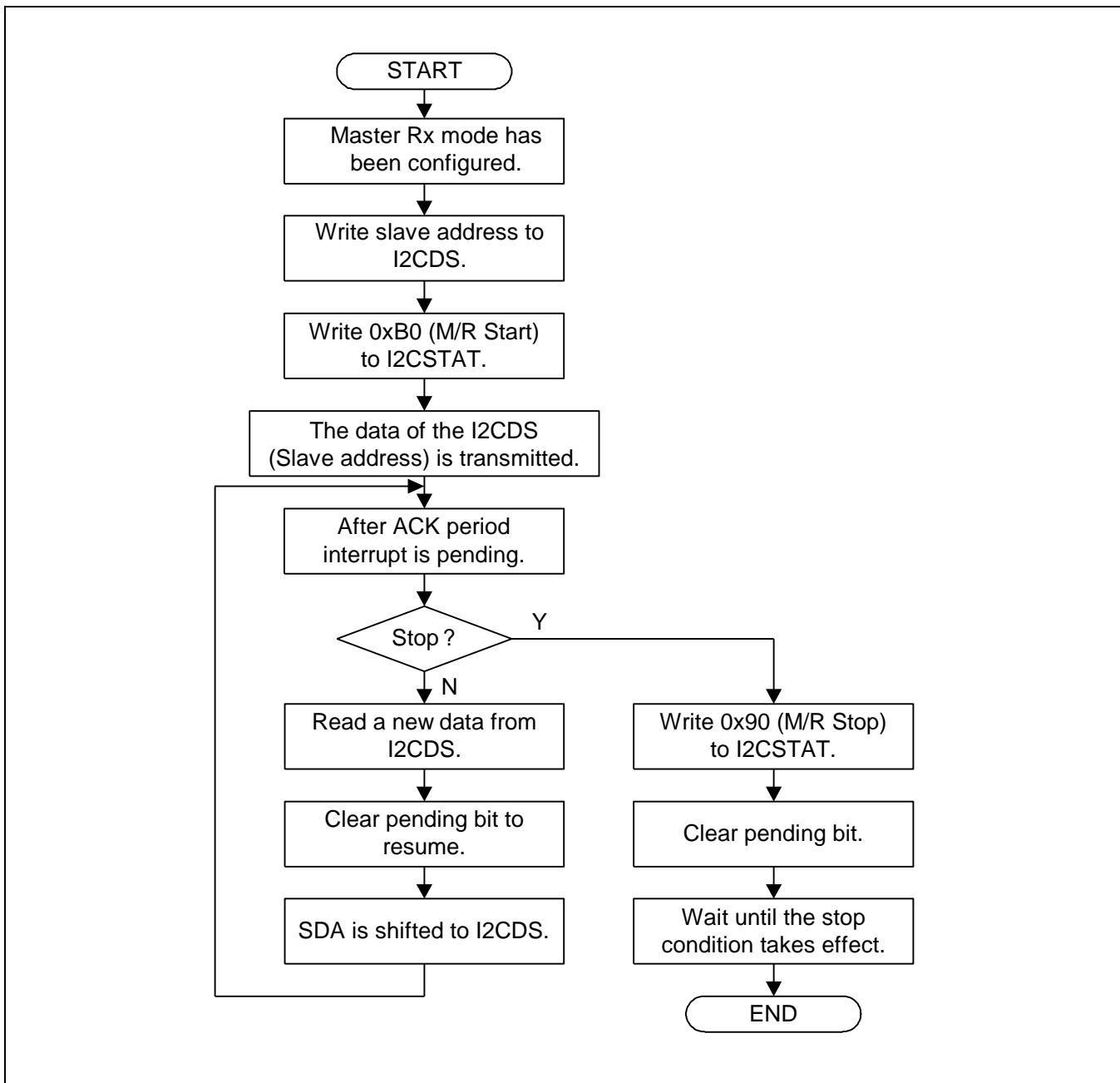


Figure 14-7 Operations for Master/ Receiver Mode

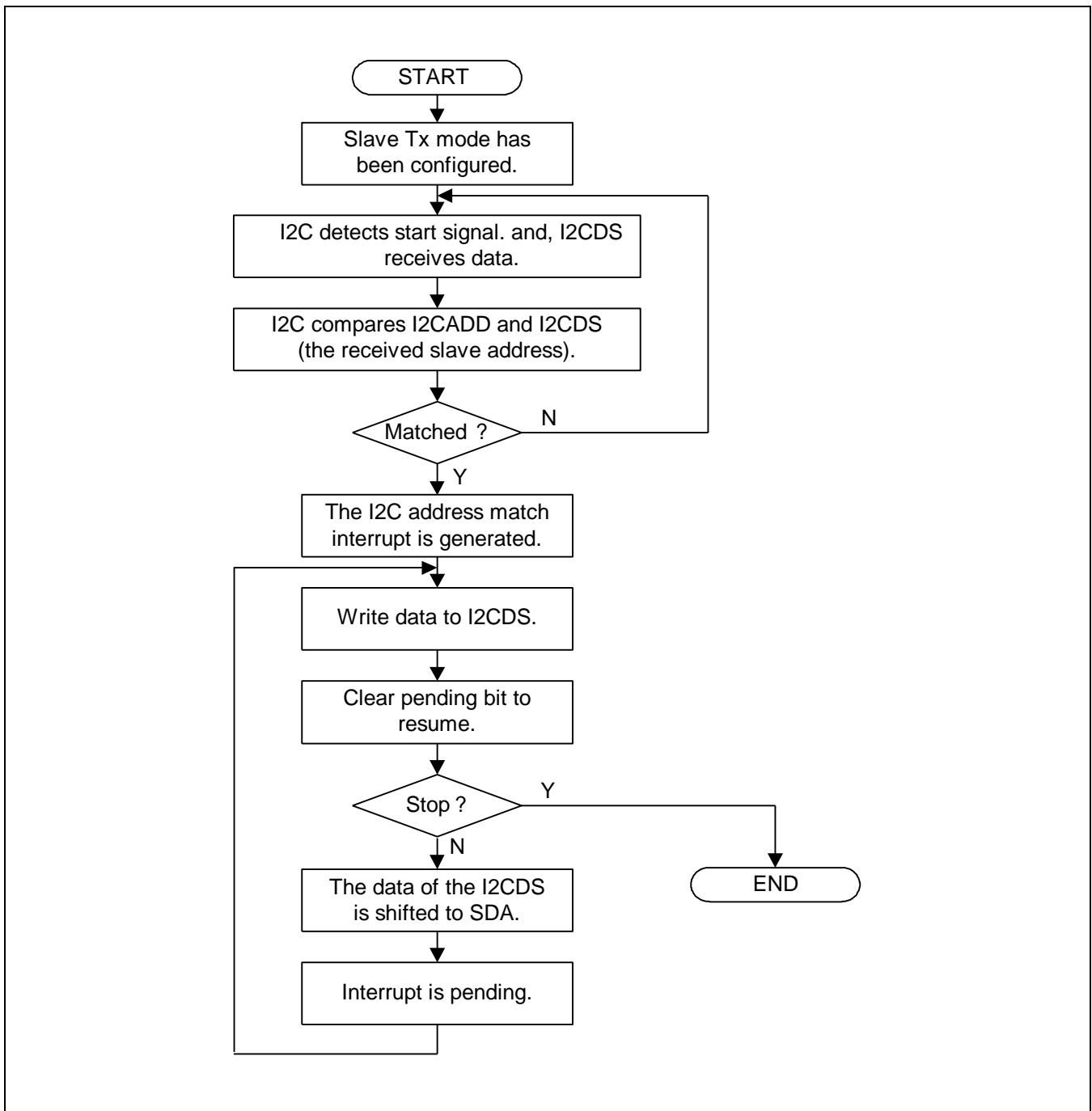


Figure 14-8 Operations for Slave/ Transmitter Mode

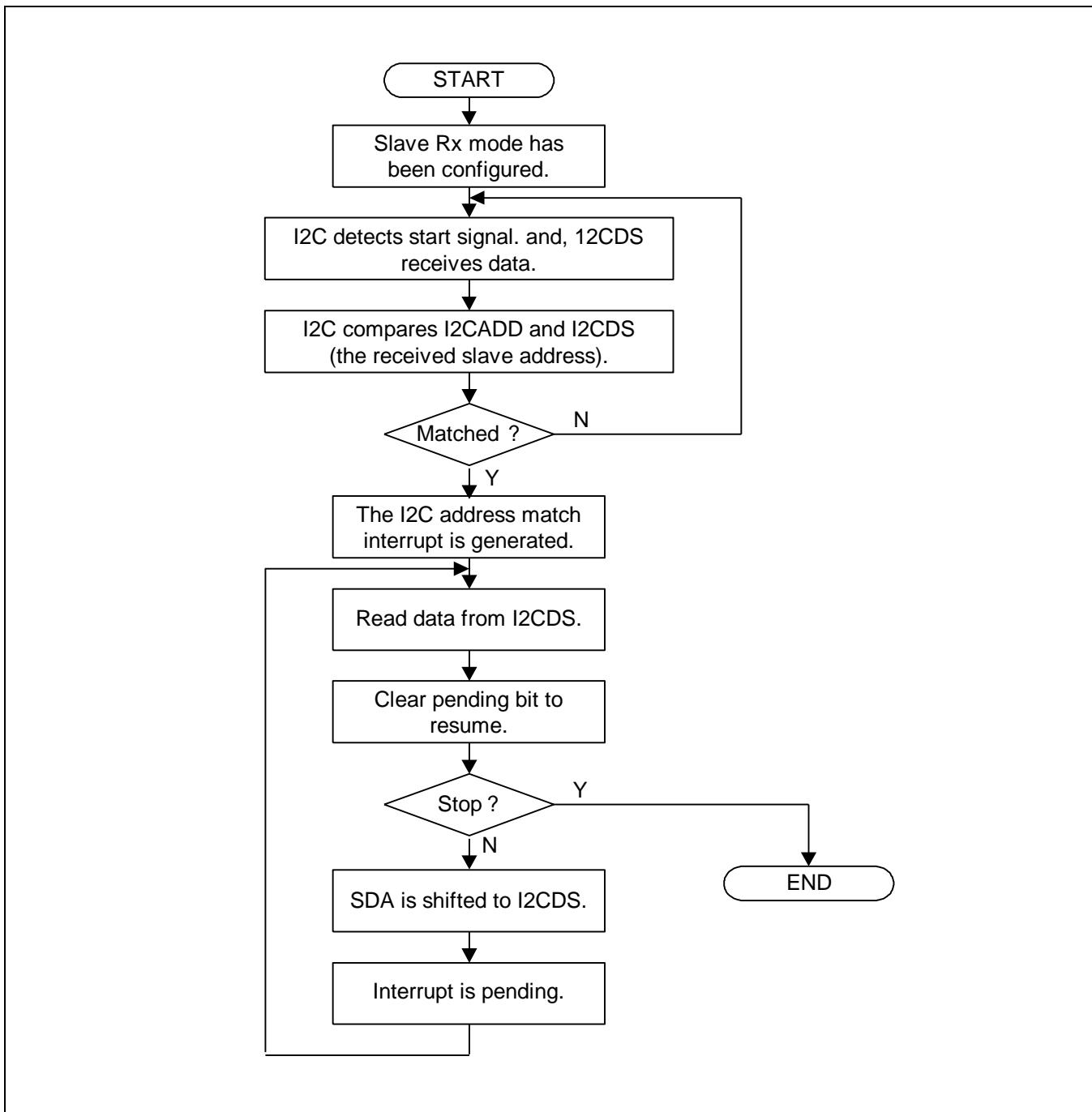


Figure 14-9 Operations for Slave/Receiver Mode

14.4 Register Description

14.4.1 Register Map Summary

- Base Address: 0x1386_0000, 0x1387_0000, 0x1388_0000, 0x1389_0000, 0x138A_0000, 0x138B_0000, 0x138C_0000, 0x138D_0000, 0x138E_0000

Register	Offset	Description	Reset Value
I2CCONn	0x0000	Specifies the I ² C-Bus Interface0 control register	0x0X
I2CSTATn	0x0004	Specifies the I ² C-Bus Interface0 control/status register	0x00
I2CADDn	0x0008	Specifies the I ² C-Bus Interface0 address register	0xXX
I2CDSn	0x000C	Specifies the I ² C-Bus Interface0 transmit/receive data shift register	0xXX
I2CLCn	0x0010	Specifies the I ² C-Bus Interface0 multi-master line control register	0x00

14.4.1.1 I2CCONn (n = 0 to 7)

- Address = Base Address + 0x0000, Reset Value = 0x0X

Name	Bit	Type	Description	Reset Value
Acknowledge generation ⁽¹⁾	[7]	RW	I ² C-bus acknowledge enable bit. 0 = Disables 1 = Enables In Tx mode, the I2CSDA is free in the ACK time. In Rx mode, the I2CSDA is L in the ACK time.	0
Tx clock source selection	[6]	RW	Source clock of I2C-bus transmit clock prescaler selection bit. 0 = I2CCLK = fPCLK /16 1 = I2CCLK = fPCLK /512	0
Tx/Rx Interrupt ⁽⁵⁾	[5]	RW	I ² C-Bus Tx/Rx interrupt enable/ disable bit. 0 = Disables 1 = Enables	0
Interrupt pending flag ^{(2) (3)}	[4]	RW	I ² C-bus Tx/Rx interrupt pending flag. This bit cannot be written to 1. If this bit is read as 1, the I2CSCL is tied to L and the I ² C is stopped. To resume the operation, clear this bit as 0. 0 = 1) No interrupt is pending (If read). 2) Clear pending condition and Resume the operation (If write). 1 = 1) Interrupt is pending (If read) 2) N/A (If write)	0
Transmit clock value ⁽⁴⁾	[3:0]	RW	I ² C-Bus transmit clock prescaler. I ² C-Bus transmit clock frequency is determined by this 4-bit prescaler value, according to the following formula: $\text{Tx clock} = \text{I2CCLK}/(\text{I2CCON}[3:0]+1).$	—

NOTE:

- Interfacing with EEPROM, the ACK generation may be disabled before reading the last data to generate the STOP condition in Rx mode.
- An I²C-bus interrupt occurs if 1) if a 1-byte transmit or receive operation is complete. In other words, ack period is finished. 2) A general call or a slave address match occurs, 3) Bus arbitration fails.
- To adjust the setup time of SDA before SCL rising edge, I2CDS has to be written before clearing the I²C interrupt pending bit.
- I2CCLK is determined by I2CCON[6].
Tx clock can vary by SCL transition time. If I2CCON[6]=0, I2CCON[3:0]=0x0 or 0x1 is not available.
- If the I2CCON[5]=0, I2CCON[4] does not operate correctly.
Therefore, It is recommended to set I2CCON[5]=1, even if you do not use the I²C interrupt.

14.4.1.2 I2CSTATn (n = 0 to 7)

- Address = Base Address + 0x0004, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
Mode selection	[7:6]	RW	I ² C-bus master/ slave Tx/Rx mode select bits. 00 = Slave receive mode 01 = Slave transmit mode 10 = Master receive mode 11 = Master transmit mode	00
Busy signal status / START STOP condition	[5]	RW	I ² C-Bus busy signal status bit. 0 = (read) Not busy (If read) write) STOP signal generation 1 = (read) Busy (If read) write) START signal generation. The data in I2CDS is transferred automatically just after the start signal.	0
Serial output	[4]	RW	I ² C-bus data output enable/ disable bit. 0 = Disables Rx/Tx, 1 = Enables Rx/Tx	0
Arbitration status flag	[3]	RW	I ² C-bus arbitration procedure status flag bit. 0 = Bus arbitration successful 1 = Bus arbitration failed during serial I/O	0
Address-as-slave status flag	[2]	RW	I ² C-bus address-as-slave status flag bit. 0 = Cleared when START/STOP condition was detected 1 = Received slave address matches the address value in the I2CADD	0
Address zero status flag	[1]	RW	I ² C-bus address zero status flag bit. 0 = Cleared if START/ STOP condition is detected 1 = Received slave address is 00000000b.	0
Last-received bit status flag	[0]	RW	I ² C-bus last-received bit status flag bit. 0 = Last-received bit is 0 (ACK was received). 1 = Last-received bit is 1 (ACK was not received).	0

14.4.1.3 I2CADD_n (n = 0 to 7)

- Address = Base Address + 0x0008, Reset Value = 0xXX

Name	Bit	Type	Description	Reset Value
Slave address	[7:0]	RW	7-bit slave address, latched from the I ² C-bus. If serial output enable = 0 in the I2CSTAT, I2CADD is write-enabled. The I2CADD value is read any time, regardless of the current serial output enable bit (I2CSTAT) setting. Slave address: [7:1] Not mapped : [0]	–

14.4.1.4 I2CDS_n (n = 0 to 7)

- Address = Base Address + 0x000C, Reset Value = 0xXX

Name	Bit	Type	Description	Reset Value
Data shift	[7:0]	RW	8-bit data shift register for I ² C-bus Tx/Rx operation. If serial output enable = 1 in the I2CSTAT, I2CDS is write-enabled. The I2CDS value is read any time, regardless of the current serial output enable bit (I2CSTAT) setting.	–

14.4.1.5 I2CLCn (n = 0 to 7)

- Address = Base Address + 0x0010, Reset Value = 0x00

Name	Bit	Type	Description	Reset Value
Filter enable	[2]		I ² C-bus filter enable bit. If SDA port is operating as input, this bit should be High. This filter prevents error caused by glitch between two PCLK clock. 0 = Disables Filter 1 = Enables Filter	0
SDA output delay	[1:0]		I ² C-Bus SDA line delay length selection bits. SDA line is delayed as following clock time(PCLK) 00 = 0 clocks 01 = 5 clocks 10 = 10 clocks 11 = 15 clocks	00

15 Serial Peripheral Interface (SPI)

15.1 Overview

The Serial Peripheral Interface (SPI) in Exynos4210 transfers serial data using various peripherals. SPI includes two 8, 16, 32-bit shift registers to transmit and receive data. During an SPI transfer, data is simultaneously transmitted (shifted out serially) and received (shifted in serially). SPI supports the protocols for National Semiconductor Microwire and Motorola Serial Peripheral Interface.

15.2 Features

The key features of SPI include:

- Full duplex
- 8/16/32-bit shift register for TX/RX
- Supports 8-bit/16-bit/32-bit bus interface
- Supports the Motorola SPI protocol and National Semiconductor Microwire
- Two independent 32 bits wide transmit and receive FIFOs: depth 64 in port 0 and depth 16 in port 1 and 2
- Master-mode and Slave-mode
- Receive-without-transmit operation
- Tx/Rx maximum frequency at up to 50 MHz

15.3 Functional Description

15.3.1 Interface Port Description

The following table lists the external signals between the SPI and external device. The unused SPI ports are used as General Purpose I/O ports. Refer to “General Purpose I/O” chapter for more information.

Signal	I/O	Description	Pad	Type
SPI_0_CLK SPI_1_CLK SPI_2_CLK	In/Out	XspiCLK is the serial clock used to control time of data transfer. Out, when used as master In, when used as slave	XspiCLK_0 XspiCLK_1 Xi2s2CDCLK	muxed
SPI_0_nSS SPI_1_nSS SPI_2_nSS	In/Out	Slave selection signal. All data TX/RX sequences are executed if XspiCS is low. Out, when used as master In, when used as slave	XspiCSn_0 XspiCSn_1 Xi2s2LRCK	muxed
SPI_0_MISO SPI_1_MISO SPI_2_MISO	In/Out	This port is the input port in Master mode. Input mode is used to get data from slave output port. Data are transmitted to master through this port in slave mode. Out, when used as slave In, when used as master	XspiMISO_0 XspiMISO_1 Xi2s2SDI	muxed
SPI_0_MOSI SPI_1_MOSI SPI_2_MOSI	In/Out	This port is the output port in Master mode. This port is used to transfer data from master output port. Data are received from master through this port in slave mode. Out, when used as master In, when used as slave	XspiMOSI_0 XspiMOSI_1 Xi2s2SDO	muxed

NOTE: Type field indicates whether pads are dedicated to the signal or pads are connected to the multiplexed signals.

15.3.1.1 PAD Driving Strength

PAD driving strength of SPI is controlled by setting drive strength control register in GPIO. SPI related SFR is GPBDRV (for SPI port 0, 1) and GPCDRV (for SPI port 2).

15.3.2 Operation of Serial Peripheral Interface

The SPI transfers 1-bit serial data between Exynos4210 and external device. The SPI in Exynos4210 supports the CPU or DMA to transmit or receive FIFOs separately and to transfer data in both directions simultaneously. SPI has two 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.

CPU (or DMA) must write data on the register SPI_TX_DATA, to write data in FIFO. Data on the register are automatically moved to Tx FIFOs. To read data from Rx FIFOs, CPU (or DMA) must access the register SPI_RX_DATA and data are automatically sent to the SPI_RX_DATA register.

SPI operating frequency can be controlled by CMU registers. Refer to "CMU" chapter for more information.

15.3.2.1 Operation Mode

SPI has two modes, namely, master and slave mode. In master mode, SPICLK is generated and transmitted to external device. XspiCS#, which is the signal to select slave, indicates data valid when XspiCS# is low level. XspiCS# must be set low before packets are transmitted or received.

15.3.2.2 FIFO Access

The SPI supports CPU access and DMA access to FIFOs. Data size of CPU access and DMA access to FIFOs are selected either from 8-bit, 16-bit, or 32-bit data. If 8-bit data size is selected, valid bits are from 0 bit to 7 bit. User can define the trigger threshold to raise interrupt to CPU. The trigger level of each FIFO in port 0 is set by 4 bytes step from 0 byte to 252 bytes, and that of each FIFO in port 1 is set by 1 byte step from 0 byte to 63 bytes. TxDMAOn or RxDMAOn bit of SPI_MODE_CFG register must be set to use DMA access. DMA access supports only single transfer and 4-burst transfer. In TX FIFO, DMA request signal is high until TX FIFO is full. In RX FIFO, DMA request signal is high if FIFO is not empty.

15.3.2.3 Trailing Bytes in the Rx FIFO

If the number of samples in Rx FIFO is less than the threshold value in INT mode or DMA 4 burst mode and no additional data is received, the remaining bytes are called trailing bytes. To remove these bytes in RX FIFO, internal timer and interrupt signal are used. The value of internal timer is set up to 1024 clocks based on APB BUS clock. When timer value is zero, interrupt signal occurs and CPU can remove trailing bytes in FIFO.

15.3.2.4 Packet Number Control

SPI controls the number of packets to be received in master mode. Set SFR (PACKET_CNT_REG) to receive any number of packets. SPI stops generating SPICLK if the number of packets is the same as PACKET_CNT_REG. The size of one packet depends on channel width. (One packet is one byte if channel width is configured as byte, and one packet is four bytes if channel width is configured as word). It is mandatory to follow software or hardware reset before this function is reloaded. (Software reset can clear all registers except special function registers, but hardware reset clears all registers).

15.3.2.5 Chip Select Control

Chip select XspiCS# is active low signal. In other words, a chip is selected when XspiCS# input is 0.

XspiCS# can be controlled automatically or manually.

When you use manual control mode, AUTO_N_MANUAL must be cleared (Default value is 0). XspiCS# level is controlled by NSSOUT bit.

When you use auto control mode, AUTO_N_MANUAL must be set as 1. XspiCS toggled between packet and packet automatically. Inactive period of XspiCS is controlled by NCS_TIME_COUNT. NSSOUT is not available at this time.

15.3.2.6 High Speed Operation as Slave

Exynos4210 SPI supports Tx/Rx operations upto 50 MHz, but there is a limitation. When Exynos4210 SPI works as a slave, it consumes large delay over than 15ns in worst operating condition. Such a large delay can cause setup violation at SPI master device. To overcome the problem, Exynos4210 SPI provides fast slave Tx mode by setting 1 to HIGH_SPEED bit of CH_CFG register. In that mode, MISO output delay is reduced by half cycle, so that the SPI master device has more setup margin.

However, the fast slave Tx mode can be used only when CPHA = 0.

15.3.2.7 FeedBack Clock Selection

Under SPI protocol spec, SPI master should capture the input data launched by slave (MISO) with its internal SPICLK. If SPI runs at high operating frequency such as 50MHz, it is difficult to capture the MISO input because the required arrival time of MISO, which is an half cycle period in Exynos4210, is shorter than the arrival time of MISO that consists of SPICLK output delay of SPI master, MISO output delay of SPI slave, and MISO input delay of SPI master. To overcome the problem, Exynos4210 SPI provides 3 feedback clocks that are phase-delayed clock of internal SPICLK.

A selection of feedback clock depends on MISO output delay of SPI slave. To capture MISO data correctly, it is selected the feedback clock that satisfies the following constraint.

$$t_{SPIMIS}(s) < t_{period}/2 - t_{SPISOD}$$

* $t_{SPIMIS}(s)$: MISO input setup time of SPI master on a given feedback clock selection 's'

* t_{SPISOD} : MISO output delay of SPI slave

* t_{period} : SPICLK cycle period

If multiple feedback clocks meet the constraint, the feedback clock with smallest phase delay should be selected. It is because a feedback clock with large phase delay may capture data of next cycle.

For example of Exynos4210 SPI CH1 with master configuration of 50MHz operating frequency, 1.8V external voltage and 15pF load, 270 degree phase-delayed feedback clock should be used if the MISO output delay of SPI slave is assumed as 11ns ($t_{SPIMIS}(s) < 10\text{ns} - 11\text{ns} = -1\text{ns}$).

If the operating clock frequency is 33MHz and other conditions are the same as the previous example, it is better to use 180 degree phase-delayed feedback clock ($t_{SPIMIS}(s) < 15\text{ns} - 11\text{ns} = 4\text{ns}$).

15.3.2.8 SPI Transfer Format

The Exynos4210 supports four different formats for data transfer. [Figure 15-1](#) describes four waveforms for SPICLK.

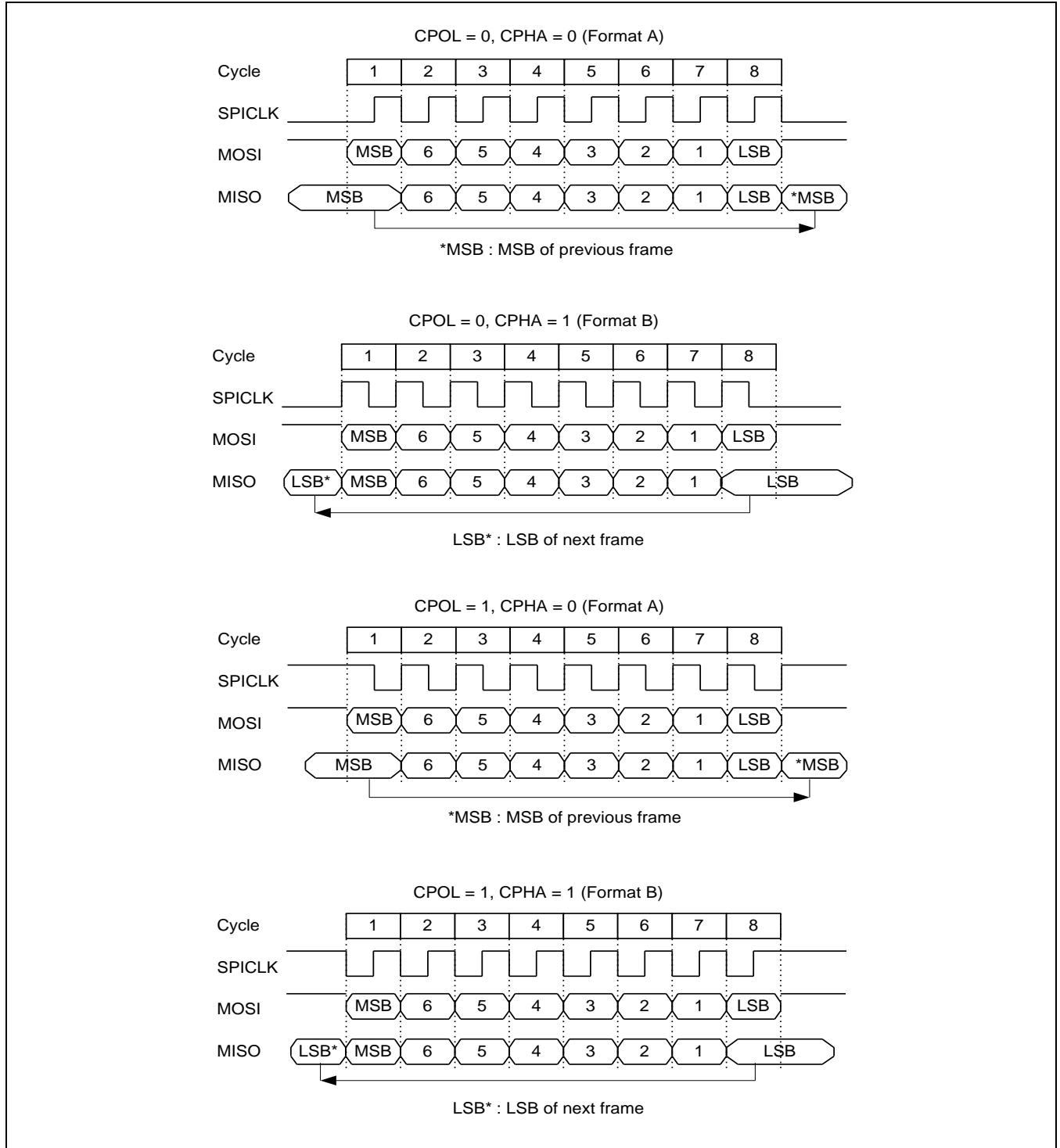


Figure 15-1 SPI Transfer Format

15.3.3 Setting Sequence of Special Function Register

Steps to set Special Function Register (nCS manual mode):

1. Set Transfer Type. (CPOL & CPHA set)
2. Set Feedback Clock Selection register.
3. Set SPI MODE_CFG register.
4. Set SPI INT_EN register.
5. Set PACKET_CNT_REG register if necessary.
6. Set Tx or Rx Channel on.
7. Set nSSout low to start Tx or Rx operation.
 - a. Set nSSout Bit to low, then start TX data writing.
 - b. If auto chip selection bit is set, nSSout is controlled automatically.

15.3.4 SPI Input Clock Description

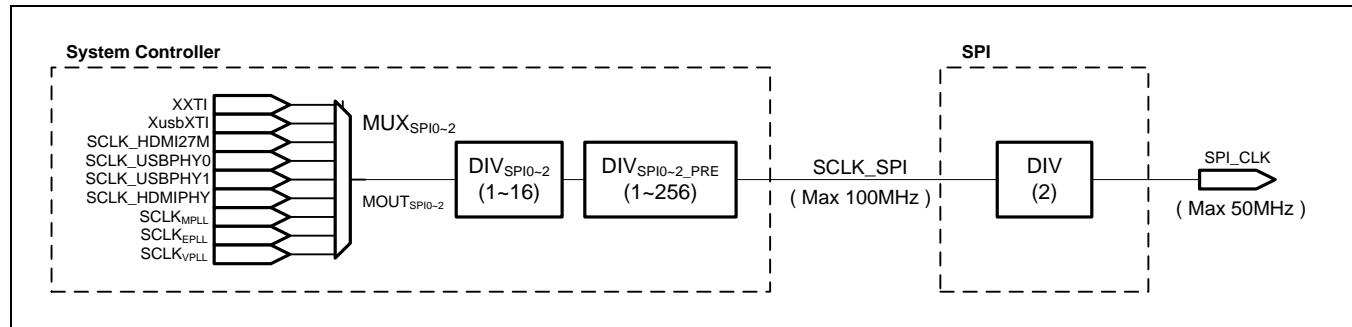


Figure 15-2 Input Clock Diagram for SPI

Exynos4210 provides SPI with a variety of clocks. As described in the [Figure 15-2](#), the SPI uses SCLK_SPI0, 1, 2 clock, which is from clock controller. You can also select SCLK_SPI0, 1, 2 from various clock sources. To select SCLK_SPI0, 1, 2, please refer to Chapter 7 Clock Controller.

NOTE: SPI has an internal 2x clock divider. SCLK_SPI should be configured to have a double of the SPI operating clock frequency.

15.4 Register Description

15.4.1 Register Map Summary

- Base Address: 0x1392_0000, 0x1393_0000, 0x1394_0000

Register	Offset	Description	Reset Value
CH_CFGn	0x0000	Specifies SPI Configuration	0x0
MODE_CFGn	0x0008	Specifies FIFO Control	0x0
CS_REGn	0x000C	Specifies Slave Selection Control	0x1
SPI_INT_ENn	0x0010	Specifies Interrupt Enable	0x0
SPI_STATUSn	0x0014	Specifies SPI Status	0x0
SPI_TX_DATA _n	0x0018	Specifies Tx Data	0x0
SPI_RX_DATA _n	0x001C	Specifies Rx Data	0x0
PACKET_CNT_REGn	0x0020	Specifies Packet Count	0x0
PENDING_CLR_REGn	0x0024	Specifies Interrupt Pending Clear	0x0
SWAP_CFGn	0x0028	Specifies Swap Configuration	0x0
FB_CLK_SELn	0x002C	Specifies Feedback Clock Selection	0x0

15.4.2 Special Function Register

15.4.2.1 CH_CFGn (n = 0 to 2)

- Address = Base Address + 0x0000, Reset Value = 0x0

Name	Bit	Type	Description	Reset Value
HIGH_SPEED_EN	[6]	RW	Slave TX output time control bit. If this bit is enabled, slave TX output time is reduced as much as half period of SPICLKout period. This bit is valid only in CPHA 0. 0 = Disables 1 = Enables	0
SW_RST	[5]	RW	Software reset. The following registers and bits are cleared by this bit. Rx/Tx FIFO Data, SPI_STATUS Once reset, this bit must be clear manually. 0 = Inactive 1 = Active	0
SLAVE	[4]	RW	Whether SPI Port is Master or Slave 0 = Master 1 = Slave	0
CPOL	[3]	RW	Determines whether active high or active low clock 0 = Active High 1 = Active Low	0
CPHA	[2]	RW	Select one of the two fundamentally different transfer format 0 = Format A 1 = Format B	0
RX_CH_ON	[1]	RW	SPI Rx Channel On 0 = Channel Off 1 = Channel On	0
TX_CH_ON	[0]	RW	SPI Tx Channel On 0 = Channel Off 1 = Channel On	0

NOTE: SPI controller should reset when

1. Reconfiguration SPI registers.
2. Error interrupt occurred.

15.4.2.2 MODE_CFGn (n = 0 to 2)

- Address = Base Address + 0x0008, Reset Value = 0x0

Name	Bit	Type	Description	Reset Value
CH_WIDTH	[30:29]	RW	00 = Byte 01 = Halfword 10 = Word 11 = Reserved	0
TRAILING_CNT	[28:19]	RW	Count value from writing the last data in RX FIFO to flush trailing bytes in FIFO	0
BUS_WIDTH	[18:17]	RW	00 = Byte 01 = Halfword 10 = Word 11 = Reserved	0
RX_RDY_LVL	[16:11]	RW	Rx FIFO trigger level in INT mode. Port 0: Trigger level (bytes) = 4 x N Port 1, 2: Trigger level (bytes) = N (N = value of RX_RDY_LVL field)	0
TX_RDY_LVL	[10:5]	RW	Tx FIFO trigger level in INT mode. Port 0: Trigger level (bytes) = 4 x N Port 1, 2: Trigger level (bytes) = N (N = value of TX_RDY_LVL field)	0
RSVD	[4:3]	-	Reserved	-
RX_DMA_SW	[2]	RW	Rx DMA mode enable/disable 0 = Disables DMA Mode 1 = Enables DMA Mode	0
TX_DMA_SW	[1]	RW	Tx DMA mode enable/disable 0 = Disables DMA Mode 1 = Enables DMA Mode	0
DMA_TYPE	[0]	RW	DMA transfer type, single or 4 busts. 0 = Single 1 = 4 burst DMA transfer size must be set as the same size in SPI DMA.	0

NOTE:

- CH_WIDTH is shift-register width.
- BUS_WIDTH is SPI FIFO width, transfer data size should be aligned at BUS_WIDTH.
- CH_WIDTH must be smaller than BUS_WIDTH or same.

15.4.2.3 CS_REGn (n = 0 to 2)

- Address = Base Address + 0x000C, Reset Value = 0x1

Name	Bit	Type	Description	Reset Value
NCS_TIME_COUNT	[9:4]	RW	NSSOUT inactive time = ((nCS_time_count + 3) / 2) × SPICLKout	0
RSVD	[3:2]	—	Reserved	—
AUTO_N_MANUAL	[1]	RW	Chip select toggle manual or auto selection 0 = Manual 1 = Auto	0
NSSOUT	[0]	RW	Slave selection signal (manual only) 0 = Active 1 = Inactive	1

If AUTO_N_MANUAL is set, NSSOUT is controlled by SPI controller and data transfer is not performed continuously.

- Unit data size depends on CH_WIDTH.

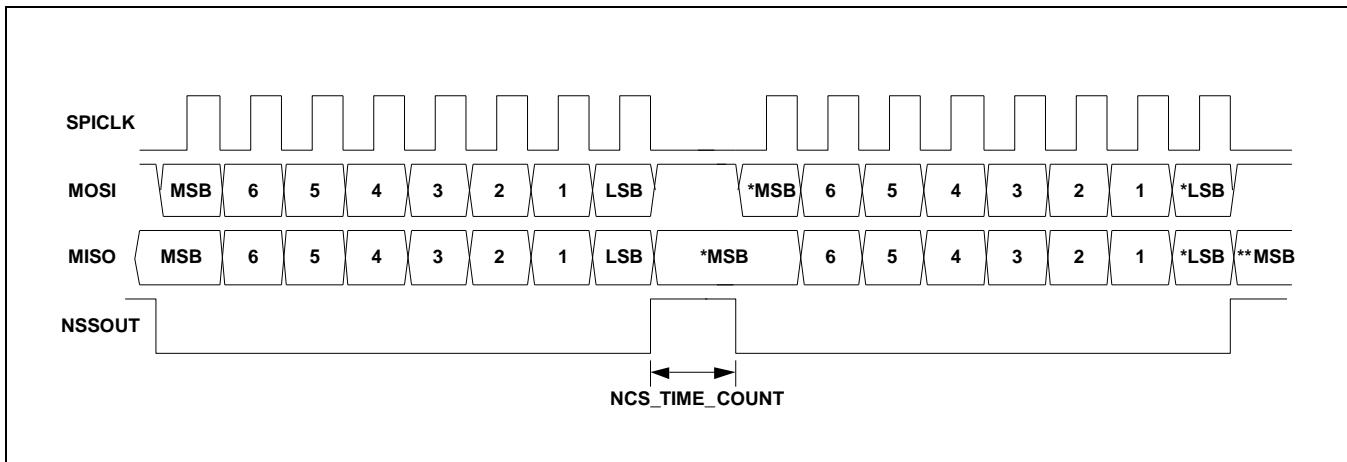


Figure 15-3 Auto Chip Select Mode Waveform (CPOL = 0, CPHA = 0, CH_WIDTH = Byte)

15.4.2.4 SPI_INT_ENn (n = 0 to 2)

- Address = Base Address + 0x0010, Reset Value = 0x0

Name	Bit	Type	Description	Reset Value
INT_EN_TRAILING	[6]	RW	Interrupt Enable for trailing count to be 0 0 = Disables 1 = Enables	0
INT_EN_RX_OVERRUN	[5]	RW	Interrupt Enable for RxOverrun 0 = Disables 1 = Enables	0
INT_EN_RX_UNDERUN	[4]	RW	Interrupt Enable for RxUnderrun 0 = Disables 1 = Enables	0
INT_EN_TX_OVERRUN	[3]	RW	Interrupt Enable for TxOverrun 0 = Disables 1 = Enables	0
INT_EN_TX_UNDERUN	[2]	RW	Interrupt Enable for TxUnderrun. In slave mode, this bit must be clear first after turning on slave TX path. 0 = Disables 1 = Enables	0
INT_EN_RX_FIFO_RDY	[1]	RW	Interrupt Enable for RxFifoRdy (INT mode) 0 = Disables 1 = Enables	0
INT_EN_TX_FIFO_RDY	[0]	RW	Interrupt Enable for TxFifoRdy (INT mode) 0 = Disables 1 = Enables	0

15.4.2.5 SPI_STATUSn (n = 0 to 2)

- Address = Base Address + 0x0014, Reset Value = 0x0

Name	Bit	Type	Description	Reset Value
TX_DONE	[25]	R	Indication of transfer done in Shift register(master mode only) 0 = All case except below case 1 = If Tx FIFO and shift register are empty after transmission start	0
TRAILING_BYTE	[24]	R	Indication that trailing count is 0	0
RX_FIFO_LVL	[23:15]	R	Data level in Rx FIFO 0 to 256 bytes in port 0 0 to 64 bytes in port 1, 2	0
TX_FIFO_LVL	[14:6]	R	Data level in Tx FIFO 0 to 256 bytes in port 0 0 to 64 bytes in port 1, 2	0
RX_OVERRUN	[5]	R	Rx FIFO overrun error 0 = No Error 1 = Overrun Error	0
RX_UNDERRUN	[4]	R	Rx FIFO underrun error 0 = No Error, 1 = Underrun Error	0
TX_OVERRUN	[3]	R	Tx FIFO overrun error 0 = No Error 1 = Overrun Error	0
TX_UNDERRUN	[2]	R	Tx FIFO underrun error 0 = No Error 1 = Underrun Error NOTE: Tx FIFO underrun error is occurred if TX FIFO is empty in slave mode.	0
RX_FIFO_RDY	[1]	R	0 = Data in FIFO less than trigger level 1 = Data in FIFO more than trigger level	0
TX_FIFO_RDY	[0]	R	0 = Data in FIFO more than trigger level 1 = Data in FIFO less than trigger level	0

15.4.2.6 SPI_TX_DATA_n (n = 0 to 2)

- Address = Base Address + 0x0018, Reset Value = 0x0

Name	Bit	Type	Description	Reset Value
TX_DATA	[31:0]	W	This field contains the data to be transmitted over the SPI channel.	0

15.4.2.7 SPI_RX_DATA_n (n = 0 to 2)

- Address = Base Address + 0x001C, Reset Value = 0x0

Name	Bit	Type	Description	Reset Value
RX_DATA	[31:0]	R	This field contains the data to be received over the SPI channel.	0

15.4.2.8 PACKET_CNT_REG_n (n = 0 to 2)

- Address = Base Address + 0x0020, Reset Value = 0x0

Name	Bit	Type	Description	Reset Value
PACKET_CNT_EN	[16]	RW	Enable bit for packet count 0 = Disable 1 = Enable	0
COUNT_VALUE	[15:0]	RW	Packet count value	0

15.4.2.9 PENDING_CLR_REGn (n = 0 to 2)

- Address = Base Address + 0x0024, Reset Value = 0x0

Name	Bit	Type	Description	Reset Value
TX_UNDERRUN_CLR	[4]	RW	TX underrun pending clear bit 0 = Non-Clear 1 = Clear	0
TX_OVERRUN_CLR	[3]	RW	TX overrun pending clear bit 0 = Non-Clear 1 = Clear	0
RX_UNDERRUN_CLR	[2]	RW	RX underrun pending clear bit 0 = Non-clear 1 = Clear	0
RX_OVERRUN_CLR	[1]	RW	RX overrun pending clear bit 0 = Non-Clear 1 = Clear	0
TRAILING_CLR	[0]	RW	Trailing pending clear bit 0 = Non-Clear 1 = Clear	0

NOTE: After error interrupt pending clear, SPI controller should be reset.

Error interrupt list: Tx underrun, Tx overrun, Rx underrun, Rx overrun

15.4.2.10 SWAP_CFGn (n = 0 to 2)

- Address = Base Address + 0x0028, Reset Value = 0x0

Name	Bit	Type	Description	Reset Value
RX_HWORD_SWAP	[7]	RW	0 = Off 1 = Swap	0
RX_BYTE_SWAP	[6]	RW	0 = Off 1 = Swap	0
RX_BIT_SWAP	[5]	RW	0 = Off 1 = Swap	0
RX_SWAP_EN	[4]	RW	Swap enable 0 = Normal 1 = Swap	0
TX_HWORD_SWAP	[3]	RW	0 = Off 1 = Swap	0
TX_BYTE_SWAP	[2]	RW	0 = Off 1 = Swap	0
TX_BIT_SWAP	[1]	RW	0 = Off 1 = Swap	0
TX_SWAP_EN	[0]	RW	Swap enable 0 = Normal 1 = Swap	0

NOTE: Data size must be larger than swap size.

15.4.2.11 FB_CLK_SEL_n (n = 0 to 2)

- Address = Base Address + 0x002C, Reset Value = 0x0

Name	Bit	Type	Description	Reset Value
FB_CLK_SEL	[1:0]	RW	<p>In master mode, SPI uses a clock which is feedback from the SPICLK. The feedback clock is intended to capture safely the slave TX signal which can be lagged if slave device is very far. There are four kinds of feedback clocks which experience different path delays. This register selects which one is to be used.</p> <p>Note that this register value is meaningless when SPI operates in slave mode.</p> <p>00 = SPICLK bypass (do not use feedback clock) 01 = A feedback clock with 90 degree phase lagging 10 = A feedback clock with 180 degree phase lagging 11 = A feedback clock with 270 degree phase lagging 90 degree phase lagging means 5ns delay in 50 MHz operating frequency.</p>	0x0

16 Display Controller

16.1 Overview

The Display controller consists of logic for transferring image data from a local bus of the camera interface controller or a video buffer located in system memory to an external LCD driver interface. The LCD driver interface supports three kinds of interface, namely, RGB-interface, indirect-i80 interface, and YUV interface for writeback. The display controller uses up to five overlay image windows that support various color formats, 256 level alpha blending, color key, x-y position control, soft scrolling, and variable window size, among others.

The display controller supports various color formats such as RGB (1 bpp to 24 bpp) and YCbCr 4:4:4 (only local bus). It is programmed to support the different requirements on screen related to the number of horizontal and vertical pixels, data line width for the data interface, interface timing, and refresh rate.

The display controller transfers the video data and generates the necessary control signals, such as, RGB_VSYNC, RGB_HSYNC, RGB_VCLK, RGB_VDEN and SYS_CS0, SYS_CS1, SYS_WE. In addition to generating control signals, the display controller contains data ports for video data (RGB_VD[23:0], and SYS_VD), as shown in [Figure 16-1](#).

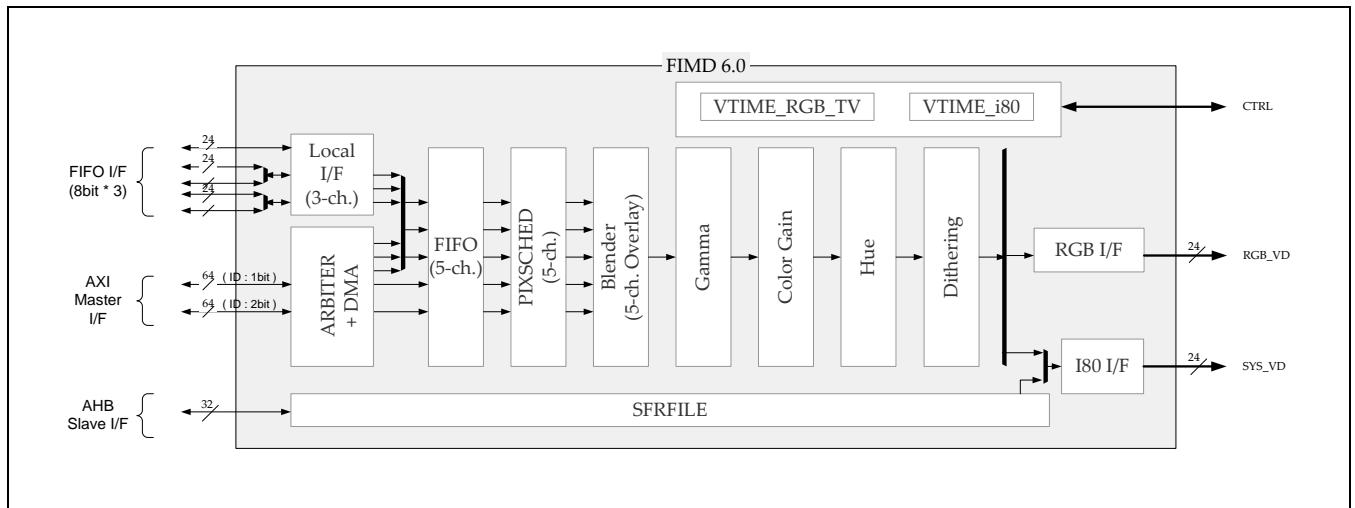


Figure 16-1 Block Diagram of Display Controller

16.2 Features

There are 2 same display controllers (Display Controller0 and Display Controller 1).

The key features of the display controller include:

Video Output Interface	RGB Interface (24-bit Parallel/8-bit Serial) Indirect i80 interface WriteBack interface
Dual Output Mode	Supports i80 and WriteBack Supports RGB and WriteBack
PIP (OSD) function	Supports 8-bpp (bit per pixel) palletized color Supports 16-bpp non-palletized color Supports unpacked 18-bpp non-palletized color Supports unpacked 24-bpp non-palletized color
	Supports X,Y indexed position
	Supports 8-bit Alpha blending (Plane/Pixel)
CSC (Internal)	RGB to YCbCr (4:2:2)
Source format	Window 0 Supports 1, 2, 4, or 8-bpp (bit per pixel) palletized color Supports 16, 18, or 24-bpp non-palletized color Supports RGB (8:8:8) local input from Local Bus (CAMIF0)
	Window 1 Supports 1, 2, 4, or 8-bpp (bit per pixel) palletized color Supports 16, 18, or 24-bpp non-palletized color Supports RGB (8:8:8) local input from Local Bus (CAMIF 1)
	Window 2 Supports 1, 2, 4, or 8-bpp (bit per pixel) palletized color Supports 16, 18, or 24-bpp non-palletized color Supports RGB (8:8:8) local input from Local Bus (CAMIF 2 or CAMIF3)
	Window 3/4 Supports 1, 2, 4, or 8-bpp (bit per pixel) palletized color Supports 16, 18, or 24-bpp non-palletized color
Configurable Burst Length	Programmable 4/8/16 Burst DMA
Palette	Window 0/1/2/3/4 Supports 256 x 32 bits palette memory (5ea: One palette memory for each window)
Soft Scrolling	Horizontal = 1 Byte resolution Vertical = 1 pixel resolution
Virtual Screen	Virtual image can have up to 16 MB image size. Each window can have its own virtual area.

Transparent Overlay	Supports Transparent Overlay
Color Key (Chroma Key)	Supports color key function Supports simultaneously color key and blending function
Partial Display	Supports LCD partial display function through i80 interface
Image Enhancement	Supports Gamma control
	Supports Hue control
	Supports color gain control
Video Clock Source	SCLK_FIMD0 for display controller 0 (from CMU module) SCLK_FIMD1 for display controller 1 (from CMU module)
Maximum VCLK in RGB Interface	Display Controller 0: 80 MHz Display Controller 1: 50 MHz

16.3 Functional Description

16.3.1 Brief Description of the Sub-Block

The display controller consists of a VSFR, VDMA, VPRCS, VTIME, and video clock generator.

To configure the display controller, the VSFR has 121 programmable register sets, one gamma LUT register set (64 registers), one i80 command register set (12 registers), and five 256×32 palette memories.

VDMA is a dedicated display DMA that transfers video data in frame memory to VPRCS. By using this special DMA, you can display video data on screen without CPU intervention.

VPRCS receives video data from VDMA and sends it to display device (LCD) through data ports (RGB_VD, or SYS_VD), after changing the video data into a suitable data format. For example, 8-bit per pixel mode (8 bpp mode) or 16-bit per pixel mode (16 bpp mode).

VTIME consists of programmable logic to support the variable requirement of interface timing and rates commonly found in different LCD drivers. The VTIME block generates RGB_VSYNC, RGB_HSYNC, RGB_VCLK, RGB_VDEN, VEN_VSYNC, VEN_HSYNC, VEN_FIELD, VEN_HREF and SYS_CS0, SYS_CS1, SYS_WE, and so on.

Using the display controller data, you can select one of the above data paths by setting LCDBLKC_CFG Register (0x1001_0210). For more information, refer to the "System Others" manual.

16.3.2 Data Flow

FIFO is in the VDMA. If FIFO is empty or partially empty, the VDMA requests data fetching from frame memory based on burst memory transfer mode. The data transfer rate determines the size of FIFO.

The display controller contains five FIFOs (Three local FIFOs and two DMA FIFOs), since it needs to support the overlay window display mode. Use one FIFO for one screen display mode.

VPRCS fetches data from FIFO. It contains the following functions for final image data: blending, image enhancing, and scheduling. It also supports the overlay function. This can overlay any image up to five window images, whose smaller or same size can be blended with the main window image having programmable alpha blending or color (chroma) key function.

[Figure 16-2](#) shows the data flow from system bus to output buffer.

VDMA has five DMA channels (Ch0 – Ch4) and three local input interfaces (CAMIF0, CAMIF1, and (CAMIF2 or CAMIF3)). The Color Space Conversion (CSC) block changes Hue (YCbCr, local input only) data to RGB data for blending operation. Also, the alpha values written in SFR determine the level of blending. Data from output buffer appears in the Video Data Port.

NOTE: The performance of the all these local input interfaces is limited by the scale ratio of the input and output image resolution (TBD).

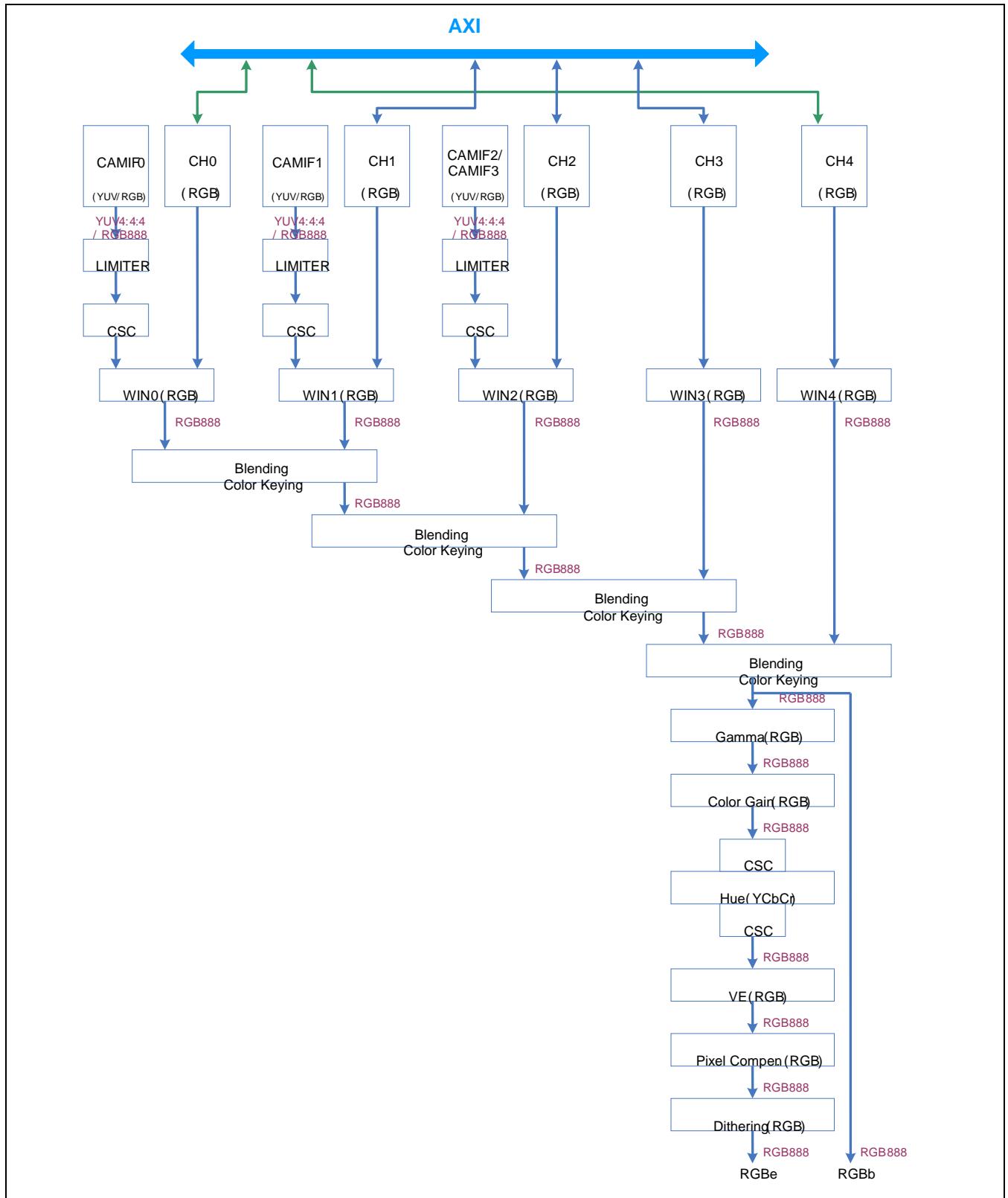


Figure 16-2 Block Diagram of the Data Flow

16.3.2.1 Interface

The display controller supports three types of interfaces:

- The first type is the conventional RGB interface, which uses RGB data, vertical/ horizontal sync, data valid signal, and data sync clock.
- The second type is the indirect i80 Interface, which uses address, data, chip select, read/ write control, and register/ status indicating signal. The LCD driver using i80 Interface contains a frame buffer and can self-refresh, so the display controller updates one still image by writing only one time to the LCD.
- The third type is FIFO interface with CAMIFx selected FIMDxWB_DEST Bit Field on CAMERA_CONTROL Register in System Register for writeback.

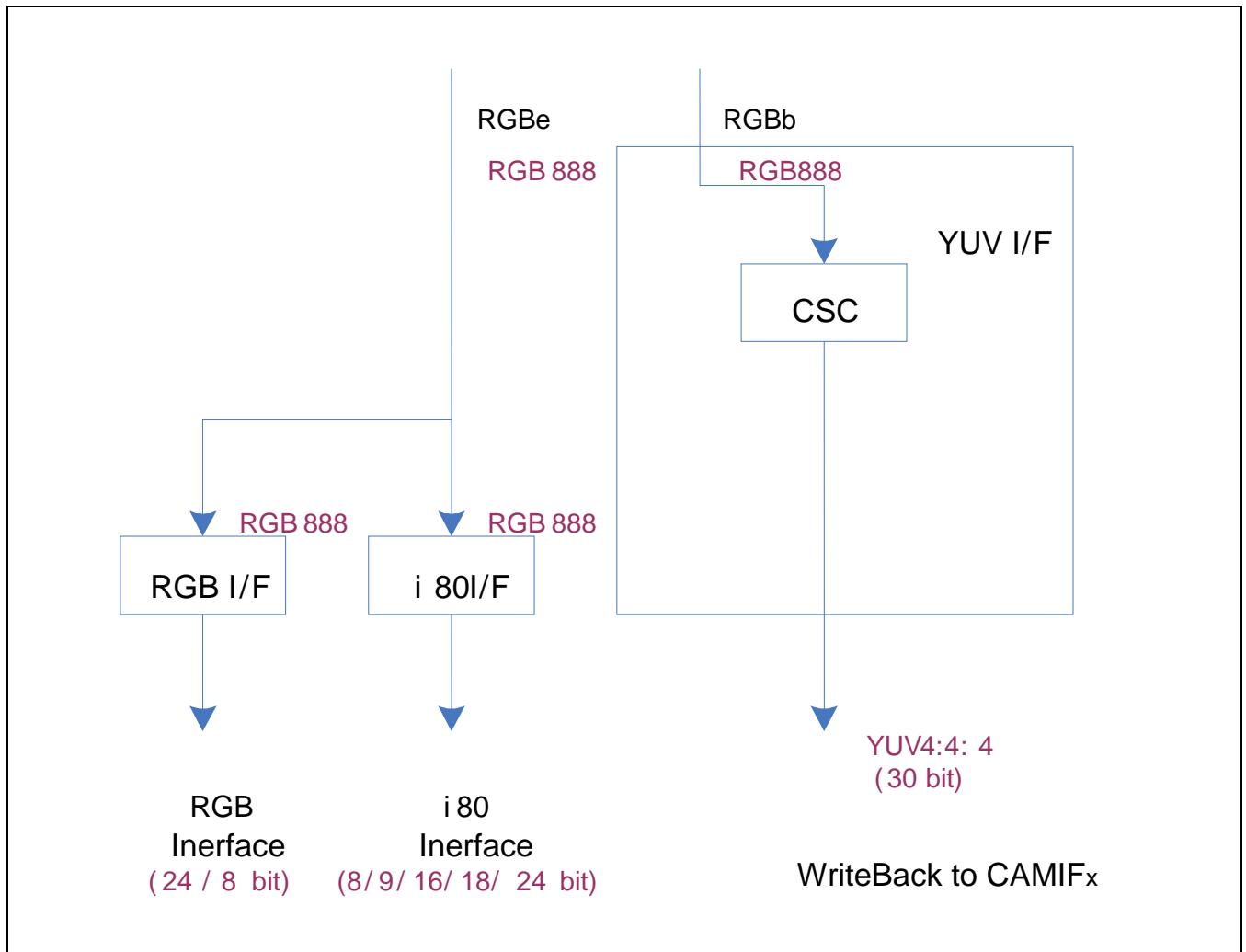


Figure 16-3 Block Diagram of the Interface

16.3.3 Overview of the Color Data

16.3.3.1 RGB Data Format

The display controller requests the specified memory format of frame buffer. The table below shows some examples of each display mode.

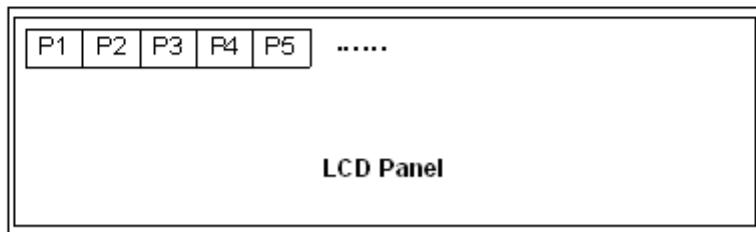
16.3.3.2 25BPP Display (A888)

(BSWP=0, HWSWP=0, WSWP=0)

	D[63:57]	D[56]	D[55:32]	D[31:25]	D[24]	D[23:0]
000H	Dummy Bit	AEN	P1	Dummy Bit	AEN	P2
008H	Dummy Bit	AEN	P3	Dummy Bit	AEN	P4
010H	Dummy Bit	AEN	P5	Dummy Bit	AEN	P6
...						

(BSWP=0, HWSWP=0, WSWP=0)

	D[63:57]	D[56]	D[55:32]	D[31:25]	D[24]	D[23:0]
000H	Dummy Bit	AEN	P1	Dummy Bit	AEN	P2
008H	Dummy Bit	AEN	P3	Dummy Bit	AEN	P4
010H	Dummy Bit	AEN	P5	Dummy Bit	AEN	P6
...						



NOTE:

1. AEN: Specifies the transparency value selection bit.
AEN" 0 = Selects ALPHA0.
AEN: 1 = Selects ALPHA1.
If per-pixel blending is set, then this pixel blends with alpha value selected by AEN.
Alpha value is selected by SFR as ALPHA0_R, ALPHA0_G, ALPHA0_B, ALPHA1_R, ALPHA1_G, and ALPHA1_B.
For more information, refer to the section on "SFR".
2. D[23:16] = Red data, D[15:8] = Green data, and D[7:0] = Blue data.

16.3.3.2.1 32BPP (8888) Mode

Pixel data contains Alpha value.

(BYSWP=0 , HWSWP=0 , WSWP=0)

	D[63:56]	D[55:32]	D[31:24]	D[23:0]
000H	ALPHA value	P1	ALPHA value	P2
008H	ALPHA value	P3	ALPHA value	P4
010H	ALPHA value	P5	ALPHA value	P6
...				

(BYSWP=0 , HWSWP=0 , WSWP=1)

	D[63:56]	D[55:32]	D[31:24]	D[23:0]
000H	ALPHA value	P2	ALPHA value	P1
008H	ALPHA value	P4	ALPHA value	P3
010H	ALPHA value	P6	ALPHA value	P5
...				

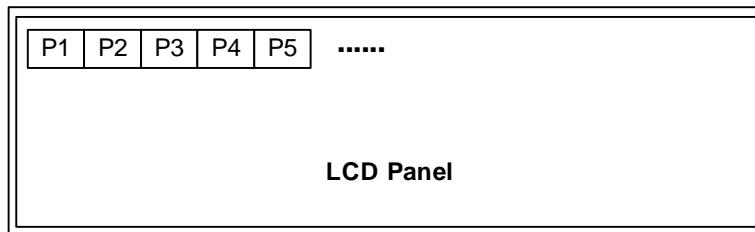
16.3.3.2.2 24BPP Display (A887)

(BSWP=0, HWSWP=0, WSWP=0)

	D[63:56]	D[55]	D[54:32]	D[31:24]	D[23]	D[22:0]
000H	Dummy Bit	AEN	P1	Dummy Bit	AEN	P2
008H	Dummy Bit	AEN	P3	Dummy Bit	AEN	P4
010H	Dummy Bit	AEN	P5	Dummy Bit	AEN	P6
...						

(BSWP=0, HWSWP=0, WSWP=1)

	D[63:56]	D[55]	D[54:32]	D[31:24]	D[23]	D[22:0]
000H	Dummy Bit	AEN	P2	Dummy Bit	AEN	P1
008H	Dummy Bit	AEN	P4	Dummy Bit	AEN	P3
010H	Dummy Bit	AEN	P6	Dummy Bit	AEN	P5
...						



NOTE:

1. AEN: Specifies the transparency value selection bit.

AEN: 0 = Selects ALPHA0.

AEN: 1 = Selects ALPHA1.

If per-pixel blending is set, then this pixel blends with alpha value selected by AEN.

Alpha value is selected by SFR as ALPHA0_R, ALPHA0_G, ALPHA0_B, ALPHA1_R, ALPHA1_G, and ALPHA1_B.

For more information, refer to the section on "SFR".

2. D[22:15] = Red data, D[14:7] = Green data, and D[6:0] = Blue data.

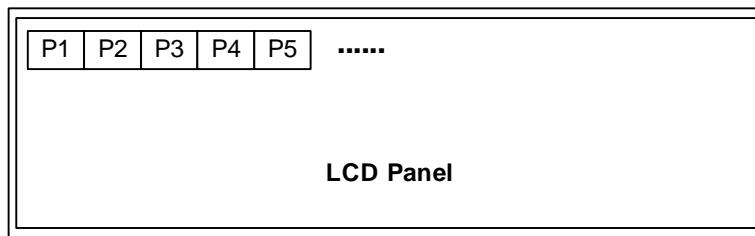
16.3.3.2.3 24BPP Display (888)

(BSWP=0, HWSWP=0, WSWP=0)

	D[63:56]	D[55:32]	D[31:24]	D[23:0]
000H	Dummy Bit	P1	Dummy Bit	P2
008H	Dummy Bit	P3	Dummy Bit	P4
010H	Dummy Bit	P5	Dummy Bit	P6
...				

(BSWP=0, HWSWP=0, WSWP=1)

	D[63:56]	D[55:32]	D[31:24]	D[23:0]
000H	Dummy Bit	P2	Dummy Bit	P1
008H	Dummy Bit	P4	Dummy Bit	P3
010H	Dummy Bit	P6	Dummy Bit	P5
...				



NOTE: D[23:16] = Red data, D[15:8] = Green data, and D[7:0] = Blue data.

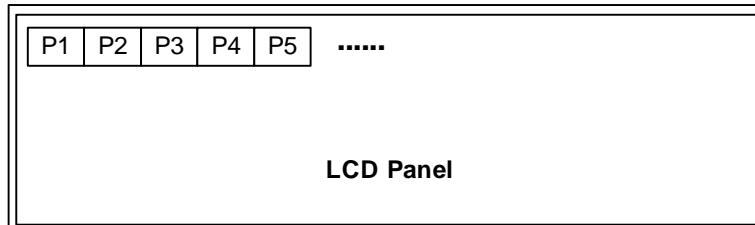
16.3.3.2.4 19BPP Display (A666)

(BSWP=0, HWSWP=0, WSWP=0)

	D[63:51]	D[50]	D[49:32]	D[31:19]	D[18]	D[17:0]
000H	Dummy Bit	AEN	P1	Dummy Bit	AEN	P2
008H	Dummy Bit	AEN	P3	Dummy Bit	AEN	P4
010H	Dummy Bit	AEN	P5	Dummy Bit	AEN	P6
...						

(BSWP=0, HWSWP=0, WSWP=1)

	D[63:51]	D[50]	D[49:32]	D[31:19]	D[18]	D[17:0]
000H	Dummy Bit	AEN	P2	Dummy Bit	AEN	P1
008H	Dummy Bit	AEN	P4	Dummy Bit	AEN	P3
010H	Dummy Bit	AEN	P6	Dummy Bit	AEN	P5
...						



NOTE:

1. AEN: Specifies the transparency value selection bit.

AEN: 0 = Selects ALPHA0.

AEN: 1 = Selects ALPHA1.

If per-pixel blending is set, then this pixel blends with alpha value selected by AEN.

Alpha value is selected by SFR as ALPHA0_R, ALPHA0_G, ALPHA0_B, ALPHA1_R, ALPHA1_G, and ALPHA1_B.

For more information, refer to the section on "SFR".

2. D[17:12] = Red data, D[11:6] = Green data, and D[5:0] = Blue data.

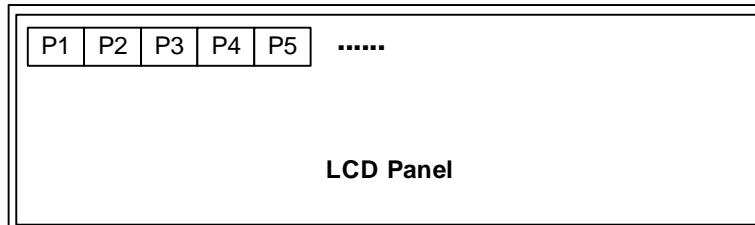
16.3.3.2.5 18BPP Display (666)

(BSWP=0, HWSWP=0, WSWP=0)

	D[63:50]	D[49:32]	D[31:18]	D[17:0]
000H	Dummy Bit	P1	Dummy Bit	P2
008H	Dummy Bit	P3	Dummy Bit	P4
010H	Dummy Bit	P5	Dummy Bit	P6
...				

(BSWP=0, HWSWP=0, WSWP=1)

	D[63:50]	D[49:32]	D[31:18]	D[17:0]
000H	Dummy Bit	P2	Dummy Bit	P1
008H	Dummy Bit	P4	Dummy Bit	P3
010H	Dummy Bit	P6	Dummy Bit	P5
...				



NOTE: D[17:12] = Red data, D[11:6] = Green data, and D[5:0] = Blue data.

16.3.3.2.6 16BPP Display (A555)

(BSWP=0, HWSWP=0, WSWP=0)

	D[63]	D[62:48]	D[47]	D[46:32]	D[31]	D[30:16]	D[15]	D[14:0]
000H	AEN1	P1	AEN2	P2	AEN3	P3	AEN4	P4
004H	AEN5	P5	AEN6	P6	AEN7	P7	AEN8	P8
008H	AEN9	P9	AEN10	P10	AEN11	P11	AEN12	P12
...								

(BSWP=0, HWSWP=0, WSWP=1)

	D[63]	D[62:48]	D[47]	D[46:32]	D[31]	D[30:16]	D[15]	D[14:0]
000H	AEN3	P3	AEN4	P4	AEN1	P1	AEN2	P2
004H	AEN7	P7	AEN8	P8	AEN5	P5	AEN6	P6
008H	AEN11	P11	AEN12	P12	AEN9	P9	AEN10	P10
...								

(BSWP=0, HWSWP=1, WSWP=0)

	D[63]	D[62:48]	D[47]	D[46:32]	D[31]	D[30:16]	D[15]	D[14:0]
000H	AEN4	P4	AEN3	P3	AEN2	P2	AEN1	P1
004H	AEN8	P8	AEN7	P7	AEN6	P6	AEN5	P5
008H	AEN12	P12	AEN11	P11	AEN10	P10	AEN9	P9
...								

NOTE:

1. AEN: Specifies the transparency value selection bit.

AEN: 0 = Selects ALPHA0.

AEN: 1 = Selects ALPHA1.

If per-pixel blending is set, then this pixel blends with alpha value selected by AEN.

Alpha value is selected by SFR as ALPHA0_R, ALPHA0_G, ALPHA0_B, ALPHA1_R, ALPHA1_G, and ALPHA1_B.

For more information, refer to the section on "SFR".

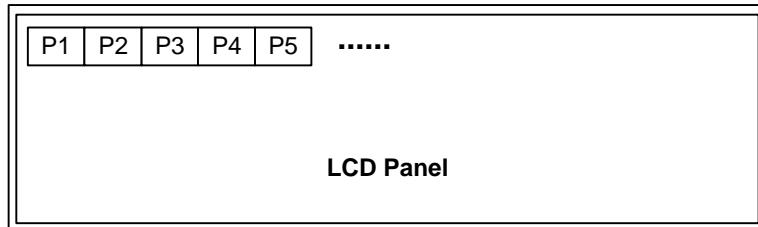
2. D[14:10] = Red data, D[9:5] = Green data, and D[4:0] = Blue data.

16.3.3.2.7 16BPP Display (1555)

(BSWP=0 , HWSWP=0 , WSWP=0)				
	D[63:48]	D[47:32]	D[31:16]	D[15:0]
000H	P1	P2	P3	P4
008H	P5	P6	P7	P8
010H	P9	P10	P11	P12
...				

(BSWP=0 , HWSWP=0 , WSWP=1)				
	D[63:48]	D[47:32]	D[31:16]	D[15:0]
000H	P3	P4	P1	P2
008H	P7	P8	P5	P6
010H	P11	P12	P9	P10
...				

(BSWP=0 , HWSWP=1 , WSWP=0)				
	D[63:48]	D[47:32]	D[31:16]	D[15:0]
000H	P4	P3	P2	P1
008H	P8	P7	P6	P5
010H	P12	P11	P10	P9
...				



NOTE: {D[14:10], D[15]} = Red data, {D[9:5], D15} = Green data, and {D[4:0], D[15]} = Blue data.

16.3.3.2.8 6BPP Display (565)

(BSWP=0, HWSWP=0, WSWP=0)

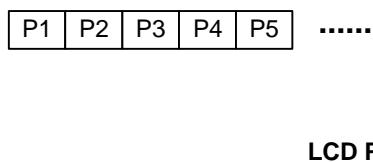
	D[63:48]	D[47:32]	D[31:16]	D[15:0]
000H	P1	P2	P3	P4
008H	P5	P6	P7	P8
010H	P9	P10	P11	P12
...				

(BSWP=0, HWSWP=0, WSWP=1)

	D[63:48]	D[47:32]	D[31:16]	D[15:0]
000H	P3	P4	P1	P2
008H	P7	P8	P5	P6
010H	P11	P12	P9	P10
...				

(BSWP=0, HWSWP=1, WSWP=0)

	D[63:48]	D[47:32]	D[31:16]	D[15:0]
000H	P4	P3	P2	P1
008H	P8	P7	P6	P5
010H	P12	P11	P10	P9
...				



NOTE: D[15:10] = Red data, D[10:5] = Green data, and D[4:0] = Blue data.

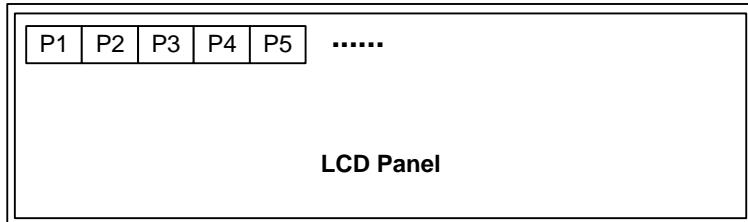
16.3.3.2.9 13BPP Display (A444)

(BYSWP=0, HWSWP=0, WSWP=0)

	D[63:61]	D[60]	D[59:48]	D[47:45]	D[44]	D[43:32]	D[31:29]	D[28]	D[27:16]	D[15:13]	D[12]	D[11:0]
000H	Dummy	AEN1	P1	Dummy	AEN2	P2	Dummy	AEN3	P3	Dummy	AEN4	P4
004H	Dummy	AEN5	P5	Dummy	AEN6	P6	Dummy	AEN7	P7	Dummy	AEN8	P8
008H	Dummy	AEN9	P9	Dummy	AEN10	P10	Dummy	AEN11	P11	Dummy	AEN12	P12
...												

(BYSWP=0, HWSWP=1, WSWP=0)

	D[63:61]	D[60]	D[59:48]	D[47:45]	D[44]	D[43:32]	D[31:29]	D[28]	D[27:16]	D[15:13]	D[12]	D[11:0]
000H	Dummy	AEN4	P4	Dummy	AEN3	P3	Dummy	AEN2	P2	Dummy	AEN1	P1
004H	Dummy	AEN8	P8	Dummy	AEN7	P7	Dummy	AEN6	P6	Dummy	AEN5	P5
008H	Dummy	AEN12	P12	Dummy	AEN11	P11	Dummy	AEN10	P10	Dummy	AEN9	P9
...												


NOTE:

1. AEN: Specifies the transparency value selection bit.
AEN: 0 = Selects ALPHA0.
AEN: 1 = Selects ALPHA1.
If per-pixel blending is set, then this pixel blends with alpha value selected by AEN.
2. D[11:8] = Red data, D[7:4] = Green data, and D[3:0] = Blue data.
3. 16BPP (4444) mode. (For more information, refer to the section on "SFR") Data has Alpha value.

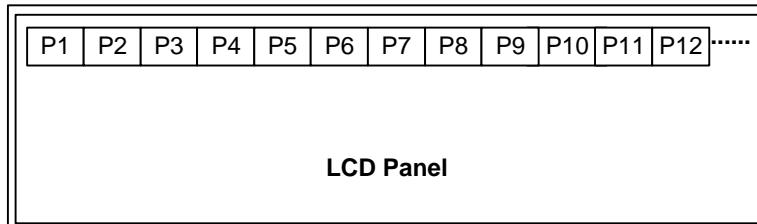
(BYSWP=0, HWSWP=0, WSWP=0)

	D[63:60]	D[59:48]	D[47:44]	D[43:32]	D[31:28]	D[27:16]	D[15:12]	D[11:0]
000H	ALPHA1	P1	ALPHA2	P2	ALPHA3	P3	ALPHA4	P4
004H	ALPHA6	P5	ALPHA6	P6	ALPHA7	P7	ALPHA8	P8
008H	ALPHA9	P9	ALPHA10	P10	ALPHA11	P11	ALPHA12	P12
...								

16.3.3.2.10 8BPP Display (A232)

(BYSWP=0, HWSWP=0, WSWP=0)																
	D[63]	D[62:56]	D[55]	D[54:48]	D[47]	D[46:40]	D[39]	D[38:32]	D[31]	D[30:24]	D[23]	D[22:16]	D[15]	D[14:8]	D[7]	D[6:0]
000H	AEN	P1	AEN	P2	AEN	P3	AEN	P4	AEN	P5	AEN	P6	AEN	P7	AEN	P8
008H	AEN	P9	AEN	P10	AEN	P11	AEN	P12	AEN	P13	AEN	P14	AEN	P15	AEN	P16
010H	AEN	P17	AEN	P18	AEN	P19	AEN	P20	AEN	P21	AEN	P22	AEN	P23	AEN	P24
...																

(BYSWP=1, HWSWP=0, WSWP=0)																
	D[63]	D[62:56]	D[55]	D[54:48]	D[47]	D[46:40]	D[39]	D[38:32]	D[31]	D[30:24]	D[23]	D[22:16]	D[15]	D[14:8]	D[7]	D[6:0]
000H	AEN	P8	AEN	P7	AEN	P6	AEN	P5	AEN	P4	AEN	P3	AEN	P2	AEN	P1
008H	AEN	P16	AEN	P15	AEN	P14	AEN	P13	AEN	P12	AEN	P11	AEN	P10	AEN	P9
010H	AEN	P24	AEN	P23	AEN	P22	AEN	P21	AEN	P20	AEN	P19	AEN	P18	AEN	P17
...																



NOTE:

1. AEN: Specifies the transparency value selection bit.

AEN: 0 = Selects ALPHA0.

AEN: 1 = Selects ALPHA1.

If per-pixel blending is set, then this pixel blends alpha value selected by AEN.

Alpha value is selected by SFR as ALPHA0_R, ALPHA0_G, ALPHA0_B, ALPHA1_R, ALPHA1_G, and ALPHA1_B.

For more information, refer to the section on "SFR".

2. D [6:5] = Red data, D [4:2] = Green data, and D [1:0] = Blue data.

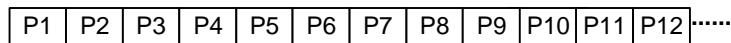
16.3.3.2.11 8BPP Display (Palette)

(BYSWP=0, HWSWP=0, WSWP=0)

	D[63:56]	D[55:48]	D[47:40]	D[39:32]	D[31:24]	D[23:16]	D[15:8]	D[7:0]
000H	P1	P2	P3	P4	P5	P6	P7	P8
008H	P9	P10	P11	P12	P13	P14	P15	P16
010H	P17	P18	P19	P20	P21	P22	P23	P24
...								

(BYSWP=1, HWSWP=0, WSWP=0)

	D[63:56]	D[55:48]	D[47:40]	D[39:32]	D[31:24]	D[23:16]	D[15:8]	D[7:0]
000H	P8	P7	P6	P5	P4	P3	P2	P1
008H	P16	P15	P14	P13	P12	P11	P10	P9
010H	P24	P23	P22	P21	P20	P19	P18	P17
...								



LCD Panel

NOTE: AEN: Specifies the transparency value selection bit (with WPALCON: Palette output format).

AEN: 0 = Selects ALPHA0.

AEN: 1 = Selects ALPHA1.

If per-pixel blending is set, then this pixel blends with alpha value selected by AEN.

Alpha value is selected by SFR as ALPHA0_R, ALPHA0_G, ALPHA0_B, ALPHA1_R, ALPHA1_G, and ALPHA1_B.

For more information, refer to the section on "SFR".

16.3.3.2.12 4BPP Display (Palette)

(BYSWP=0, HWSWP=0, WSWP=0)								
	D[63:60]	D[59:56]	D[55:52]	D[51:48]	D[47:44]	D[43:40]	D[39:36]	D[35:32]
000H	P1	P2	P3	P4	P5	P6	P7	P8
008H	P17	P18	P19	P20	P21	P22	P23	P24
...								

	D[31:28]	D[27:24]	D[23:20]	D[19:16]	D[15:12]	D[11:8]	D[7:4]	D[3:0]
000H	P9	P10	P11	P12	P13	P14	P15	P16
008H	P25	P26	P27	P28	P29	P30	P31	P32
...								

(BYSWP=1, HWSWP=0, WSWP=0)								
	D[63:60]	D[59:56]	D[55:52]	D[51:48]	D[47:44]	D[43:40]	D[39:36]	D[35:32]
000H	P15	P16	P13	P14	P11	P12	P9	P10
008H	P31	P32	P29	P30	P27	P28	P25	P26
...								

	D[31:28]	D[27:24]	D[23:20]	D[19:16]	D[15:12]	D[11:8]	D[7:4]	D[3:0]
000H	P7	P8	P5	P6	P3	P4	P1	P2
008H	P23	P24	P21	P22	P19	P20	P17	P18
...								

NOTE: AEN: Specifies the transparency value selection bit (with WPALCON: Palette output format)

AEN: 0 = Selects ALPHA0.

AEN: 1 = Selects ALPHA1.

If per-pixel blending is set, then this pixel blends with alpha value selected by AEN.

Alpha value is selected by SFR as ALPHA0_R, ALPHA0_G, ALPHA0_B, ALPHA1_R, ALPHA1_G, and ALPHA1_B.

For more information, refer to the section on "SFR".

16.3.3.2.13 2BPP Display (Palette)

(BYSWP=0, HWSWP=0, WSWP=0)								
	D[63:62]	D[61:60]	D[59:58]	D[57:56]	D[55:54]	D[53:52]	D[51:50]	D[49:48]
000H	P1	P2	P3	P4	P5	P6	P7	P8
008H	P33	P34	P35	P36	P37	P38	P39	P40
...								
	D[47:46]	D[45:44]	D[43:42]	D[41:40]	D[39:38]	D[37:36]	D[35:34]	D[33:32]
000H	P9	P10	P11	P12	P13	P14	P15	P16
008H	P41	P42	P43	P44	P45	P46	P47	P48
...								
	D[31:30]	D[29:28]	D[27:26]	D[25:24]	D[23:22]	D[21:20]	D[19:18]	D[17:16]
000H	P17	P18	P19	P20	P21	P22	P23	P24
008H	P49	P50	P51	P52	P53	P54	P55	P56
...								
	D[15:14]	D[13:12]	D[11:10]	D[9:8]	D[7:6]	D[5:4]	D[3:2]	D[1:0]
000H	P25	P26	P27	P28	P29	P30	P31	P32
008H	P57	P58	P59	P60	P61	P62	P63	P64
...								

NOTE: AEN: Specifies the transparency value selection bit (with WPALCON: Palette output format).

AEN: 0 = Selects ALPHA0.

AEN: 1 = Selects ALPHA1.

If per-pixel blending is set, then this pixel blends with alpha value selected by AEN.

Alpha value is selected by SFR as ALPHA0_R, ALPHA0_G, ALPHA0_B, ALPHA1_R, ALPHA1_G, and ALPHA1_B.

For more information, refer to the section on "SFR".

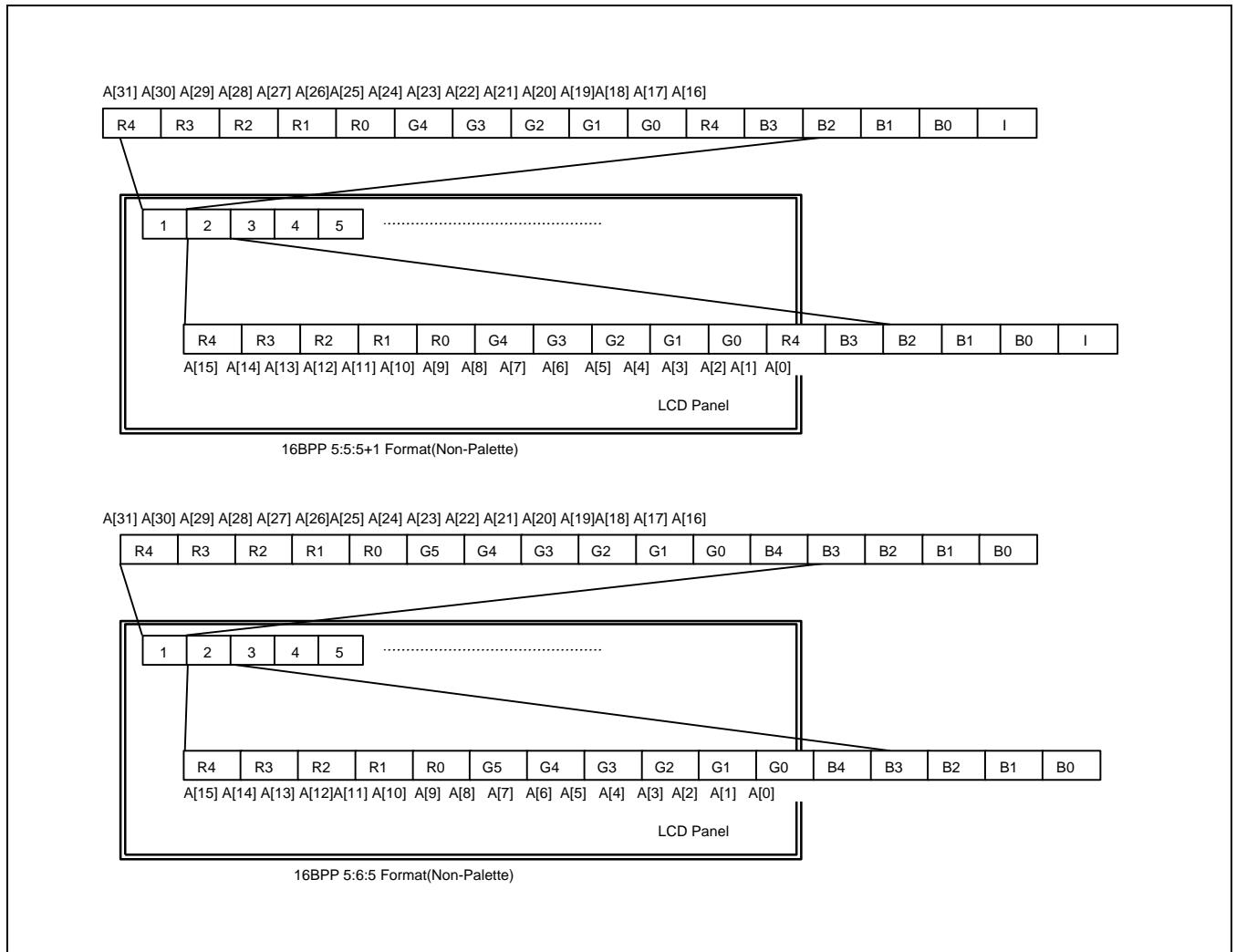


Figure 16-4 16BPP (5:6:5) Display Types

16.3.4 Palette Usage

16.3.4.1 Palette Configuration and Format Control

The display controller supports 256-color palette to select color mapping. You can select up to 256 colors from 32-bit colors using these formats.

256 color palette consists of 256 (depth) × 32-bit SPSRAM. Palette supports 8:8:8, 6:6:6, 5:6:5 (R: G: B), and other formats.

For Example:

See A:5:5:5 format, write palette, as shown in [Table 16-2](#).

Connect VD pin to TFT LCD panel (R(5)=VD[23:19], G(5)=VD[15:11], and B(5)=VD[7:3]). AEN bit controls the blending function, enable or disable. Finally, set WPALCON (W1PAL, case window0) register to 0'b101. The 32-bit (8:8:8:8) format has an alpha value directly, without using alpha value register (ALPHA_0/1).

Table 16-1 32BPP (8:8:8:8) Palette Data Format

INDEX/ Bit Pos.	3 1	3 0	2 9	2 8	2 7	2 6	2 5	2 4	2 3	2 2	2 1	2 0	1 9	1 8	1 7	1 6	1 5	1 4	1 3	1 2	1 1	1 0	9	8	7	6	5	4	3	2	1	0
00h									R 7	R 6	R 5	R 4	R 3	R 2	R 1	R 0	G 7	G 6	G 5	G 4	G 3	G 2	G 1	G 0	B 7	B 6	B 5	B 4	B 3	B 2	B 1	B 0
01h									R 7	R 6	R 5	R 4	R 3	R 2	R 1	R 0	G 7	G 6	G 5	G 4	G 3	G 2	G 1	G 0	B 7	B 6	B 5	B 4	B 3	B 2	B 1	B 0
.....		
FFh									R 7	R 6	R 5	R 4	R 3	R 2	R 1	R 0	G 7	G 6	G 5	G 4	G 3	G 2	G 1	G 0	B 7	B 6	B 5	B 4	B 3	B 2	B 1	B 0
Number of VD	-	-	-	-	-	-	-	-	2 3	2 2	2 1	2 0	1 9	1 8	1 7	1 6	1 5	1 4	1 3	1 2	1 1	1 0	9	8	7	6	5	4	3	2	1	0

Table 16-2 25BPP (A: 8:8:8) Palette Data Format

INDEX/ Bit Pos.	3 1	3 0	2 9	2 8	2 7	2 6	2 5	2 4	2 3	2 2	2 1	2 0	1 9	1 8	1 7	1 6	1 5	1 4	1 3	1 2	1 1	1 0	9	8	7	6	5	4	3	2	1	0
00 h	-	-	-	-	-	-	-	A E N	R 7	R 6	R 5	R 4	R 3	R 2	R 1	R 0	G 7	G 6	G 5	G 4	G 3	G 2	G 1	G 0	B 7	B 6	B 5	B 4	B 3	B 2	B 1	B 0
01h	-	-	-	-	-	-	-	A E N	R 7	R 6	R 5	R 4	R 3	R 2	R 1	R 0	G 7	G 6	G 5	G 4	G 3	G 2	G 1	G 0	B 7	B 6	B 5	B 4	B 3	B 2	B 1	B 0
.....		
FFh	-	-	-	-	-	-	-	A E N	R 7	R 6	R 5	R 4	R 3	R 2	R 1	R 0	G 7	G 6	G 5	G 4	G 3	G 2	G 1	G 0	B 7	B 6	B 5	B 4	B 3	B 2	B 1	B 0
Number of VD	-	-	-	-	-	-	-	-	2 3	2 2	2 1	2 0	1 9	1 8	1 7	1 6	1 5	1 4	1 3	1 2	1 1	1 0	9	8	7	6	5	4	3	2	1	0

Table 16-3 19BPP (A: 6:6:6) Palette Data Format

INDEX/ Bit Pos.	3 1	3 0	2 9	2 8	2 7	2 6	2 5	2 4	2 3	2 2	2 1	2 0	1 9	1 8	1 7	1 6	1 5	1 4	1 3	1 2	1 1	1 0	9	8	7	6	5	4	3	2	1	0	
00 h	-	-	-	-	-	-	-	-	-	-	-	-	-	A E N	R 5	R 4	R 3	R 2	R 1	R 0	G 5	G 4	G 3	G 2	G 1	G 0	B 5	B 4	B 3	B 2	B 1	B 0	
01 h	-	-	-	-	-	-	-	-	-	-	-	-	-	A E N	R 5	R 4	R 3	R 2	R 1	R 0	G 5	G 4	G 3	G 2	G 1	G 0	B 5	B 4	B 3	B 2	B 1	B 0	
.....			
FFh	-	-	-	-	-	-	-	-	-	-	-	-	-	A E N	R 5	R 4	R 3	R 2	R 1	R 0	G 5	G 4	G 3	G 2	G 1	G 0	B 5	B 4	B 3	B 2	B 1	B 0	
Number of VD	-	-	-	-	-	-	-	-	-	-	-	-	-	2 3	2 2	2 1	2 0	1 9	1 8	1 5	1 4	1 3	1 2	1 1	1 0	7	6	5	4	3	2	1	0

Table 16-4 16BPP (A: 5:5:5) Palette Data Format

INDEX / Bit Pos .	3 1	3 0	2 9	2 8	2 7	2 6	2 5	2 4	2 3	2 2	2 1	2 0	1 9	1 8	1 7	1 6	1 5	1 4	1 3	1 2	1 1	1 0	9	8	7	6	5	4	3	2	1	0		
00 h	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A E N	R 4	R 3	R 2	R 1	R 0	G 4	G 3	G 2	G 1	G 0	B 4	B 3	B 2	B 1	B 0				
01 h	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A E N	R 4	R 3	R 2	R 1	R 0	G 4	G 3	G 2	G 1	G 0	B 4	B 3	B 2	B 1	B 0				
.....				
FFh	-	-	-	-	-	-	-	-	-	-	-	-	-	-	A E N	R 4	R 3	R 2	R 1	R 0	G 4	G 3	G 2	G 1	G 0	B 4	B 3	B 2	B 1	B 0				
Number of VD	-	-	-	-	-	-	-	-	-	-	-	-	-	-	2 3	2 2	2 1	2 0	1 9	1 8	1 5	1 4	1 3	1 2	1 1	1 0	7	6	5	4	3	2	1	0

16.3.4.2 Palette Read/Write

You should not access palette memory when the Vertical Status (VSTATUS) register has an ACTIVE status. VSTATUS must be checked to do Read/Write operation on the palette.

16.3.5 Window Blending

16.3.5.1 Overview of Window Blending

The main function of the VPRCS module is window blending. Display controller comprises five window layers (win0 to win4).

Example of Application:

The system uses

win0 as OS window, full TV screen window, and so on.
win1 as small (next channel) TV screen with win2 as menu.
win3 as caption.
win 4 as channel information.
win3 and win4 have color limitation while using color index with Color LUT. This feature enhances the system performance by reducing the data rate of total system.

Example of Total Five Windows:

win 0 (base): Local/ (YCbCr, RGB without palette)
win 1 (Overlay1): RGB with palette
win 2 (Overlay2): RGB with palette
win 3 (Caption): RGB (1/2/4) with 16-level Color LUT
win 4 (Cursor): RGB (1/2) with 4-level Color LUT

Overlay Priority

- Win 4 > Win 3 > Win 2 > Win 1 > Win 0

Color Key

The register value to Color Key register must be set by 24-bit RGB format.

Blending Equation

<Data blending>

$\text{Win01(R,G,B)} = \text{Win0(R,G,B)} \times b1 + \text{Win1(R,G,B)} \times a1$
 $\text{Win012(R/G/B)} = \text{Win01(R/G/B)} \times b2 + \text{Win2(R/G/B)} \times a2$
 $\text{Win0123(R/G/B)} = \text{Win012(R/G/B)} \times b3 + \text{Win3(R/G/B)} \times a3$
 $\text{WinOut(R/G/B)} = \text{Win0123(R/G/B)} \times b4 + \text{Win4(R/G/B)} \times a4$

, where,

Win0(R) = Window 0's Red data,
 Win0(G) = Window 0's Green data,
 Win0(B) = Window 0's Blue data,
 Win1(R) = Window 1's Red data,

...

$b1$ = Background's Data blending equation1 factor,
 $a1$ = Foreground's Data blending equation1 factor,
 $b2$ = Background's Data blending equation2 factor,
 $a2$ = Foreground's Data blending equation2 factor,

<Alpha value blending>

$\text{AR(G,B)01} = \text{AR(G,B)0} \times q1 + \text{AR(G,B)1} \times p1$
 $\text{AR(G,B)012} = \text{AR(G,B)01} \times q2 + \text{AR(G,B)2} \times p2$
 $\text{AR(G,B)0123} = \text{AR(G,B)012} \times q3 + \text{AR(G,B)3} \times p3$

, where,

AR0 = Window 0's Red blending factor,
 AG0 = Window 0's Green blending factor,
 AB0 = Window 0's Blue blending factor,
 AR1 = Window 1's Red blending factor, ...

AR01 = Window01's Red blending factor (alpha value blending between AR0 and AR1),
 AG01 = Window01's Green blending factor (alpha value blending between AG0 and AG1),
 AB01 = Window01's Blue blending factor (alpha value blending between AB0 and AB1),
 AR012 = Window012's Red blending factor (alpha value blending between AR01 and AR2),

...

$q1$ = Background's Alpha value blending equation1 factor,
 $p1$ = Foreground's Alpha value blending equation1 factor,
 $q2$ = Background's Alpha value blending equation2 factor,
 $p2$ = Foreground's Alpha value blending equation2 factor, ...

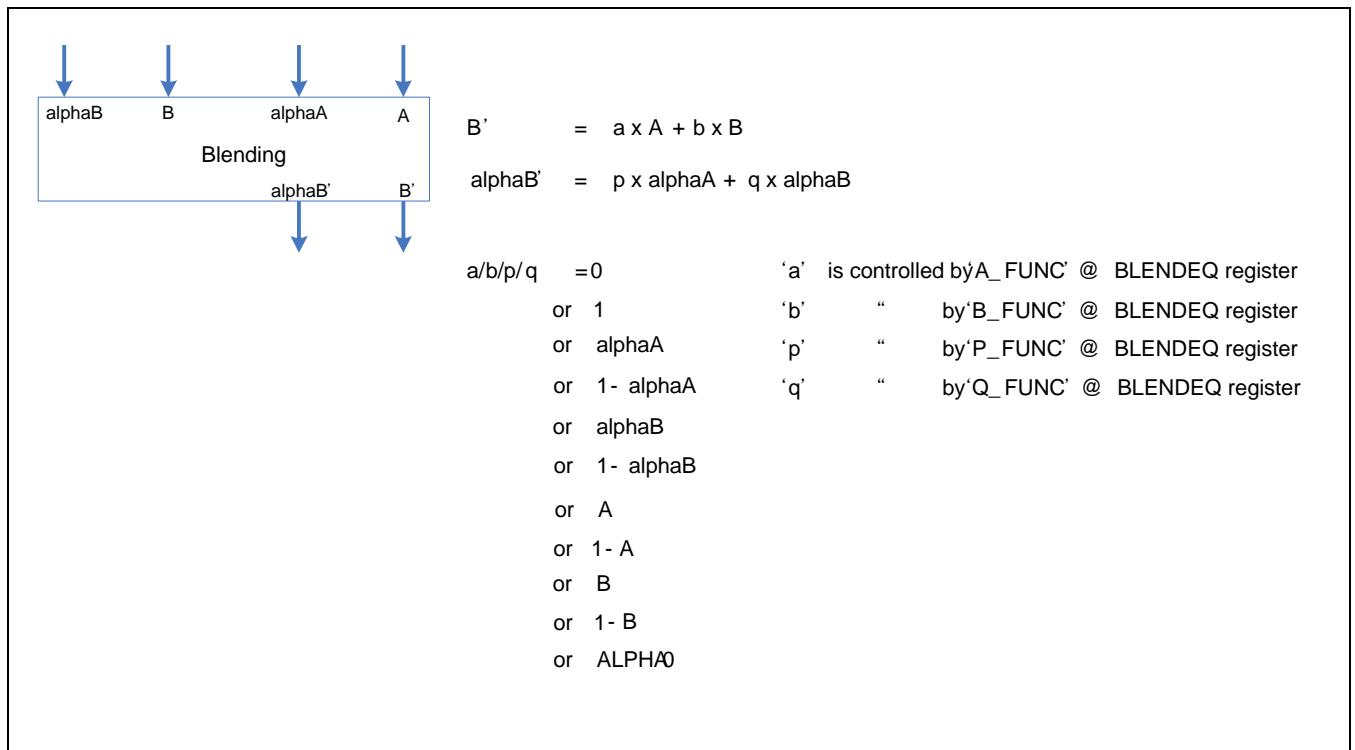


Figure 16-5 Blending Equation

< Default blending equation >
Data blending
 $B' = B \times (1-\alpha A) + A \times \alpha A$
Alpha value blending
 $\alpha B' = 0 (= \alpha B \times 0 + \alpha A \times 0)$

16.3.5.2 Blending Diagram

The display controller can blend five layers for one pixel at the same time. ALPHA0_R, ALPHA0_G, ALPHA0_B, ALPHA1_R, ALPHA1_G, and ALPHA1_B registers control the alpha value (Blending factor), which are implemented for each window layer and color (R, G, B).

The example below shows the R (Red) output using ALPHA_R value of each window.

All windows have two kinds of alpha blending value:

- Alpha value for transparency enable (AEN value ==1)
- Alpha value for transparency disable (AEN value == 0)

If WINEN_F and BLD_PIX are enabled and ALPHA_SEL is disabled, then AR is chosen using the following equation:

- $AR = (\text{Pixel}(R)\text{'s AEN value} == 1'b1) ? \text{Reg } (\text{ALPHA1_R}): \text{Reg } (\text{ALPHA0_R});$
- $AG = (\text{Pixel}(G)\text{'s AEN value} == 1'b1) ? \text{Reg } (\text{ALPHA1_G}): \text{Reg } (\text{ALPHA0_G});$
- $AB = (\text{Pixel}(B)\text{'s AEN value} == 1'b1) ? \text{Reg } (\text{ALPHA1_B}): \text{Reg } (\text{ALPHA0_B});$
(where, BLD_PIX == 1, ALPHA_SEL == 0)

If WINEN_F is enabled and BLD_PIX is disabled, then AR is controlled by ALPHA_SEL ALPHA0. AEN bit information is not used anymore.

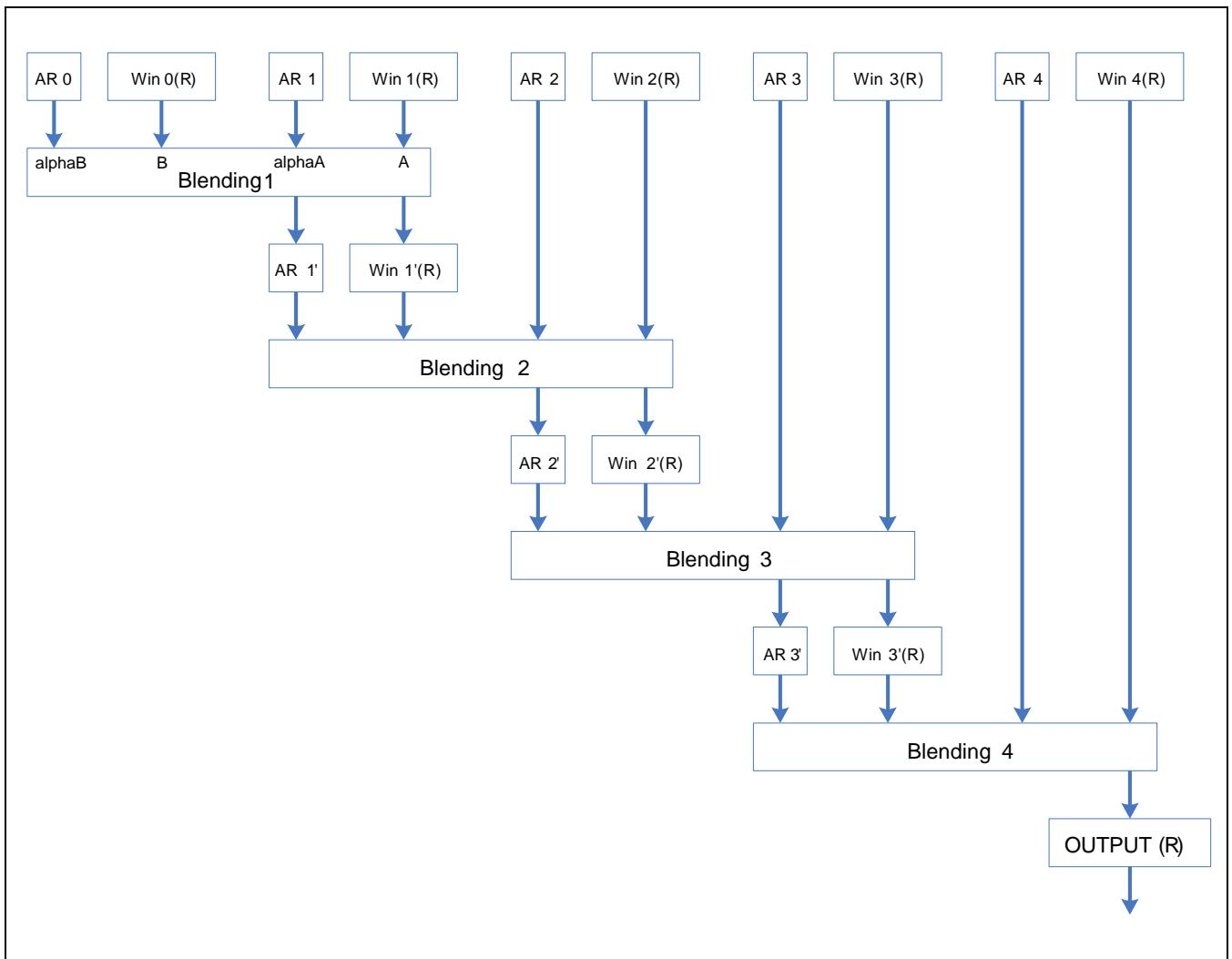


Figure 16-6 Blending Diagram

Example:

Window n's blending factor decision ($n = 0, 1, 2, 3, 4$). For more information, refer to the section on "SFR".

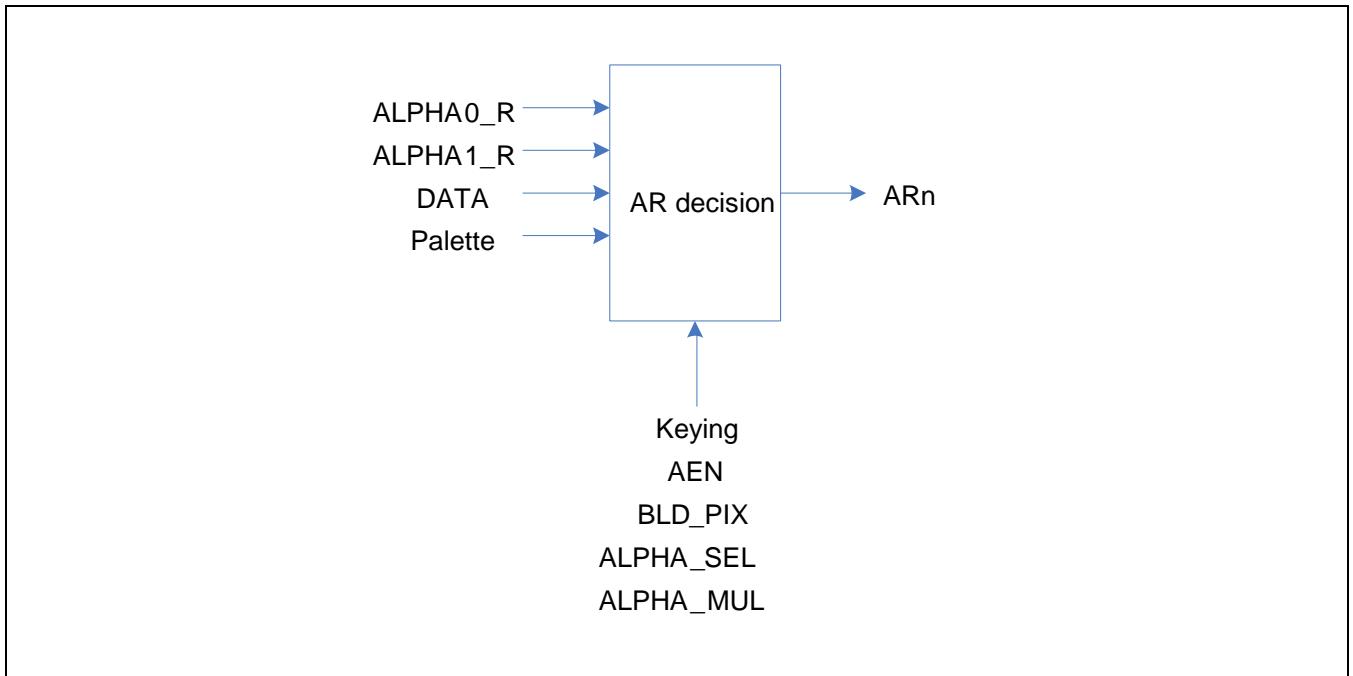


Figure 16-7 Blending Factor Decision

NOTE: If DATA[15:12] (BPPMODE_F = b'1110, ARGB4444 format) is used to blend, alpha value is {DATA[15:12], DATA [15:12]} (4-bit → 8-bit expanding).

16.3.5.3 Color-Key Function

The Color-Key function in display controller supports various effects for image mapping. For special functionality, the Color-Key register that specifies the color image of OSD layer is substituted by the background image--either as cursor image or preview image of the camera.

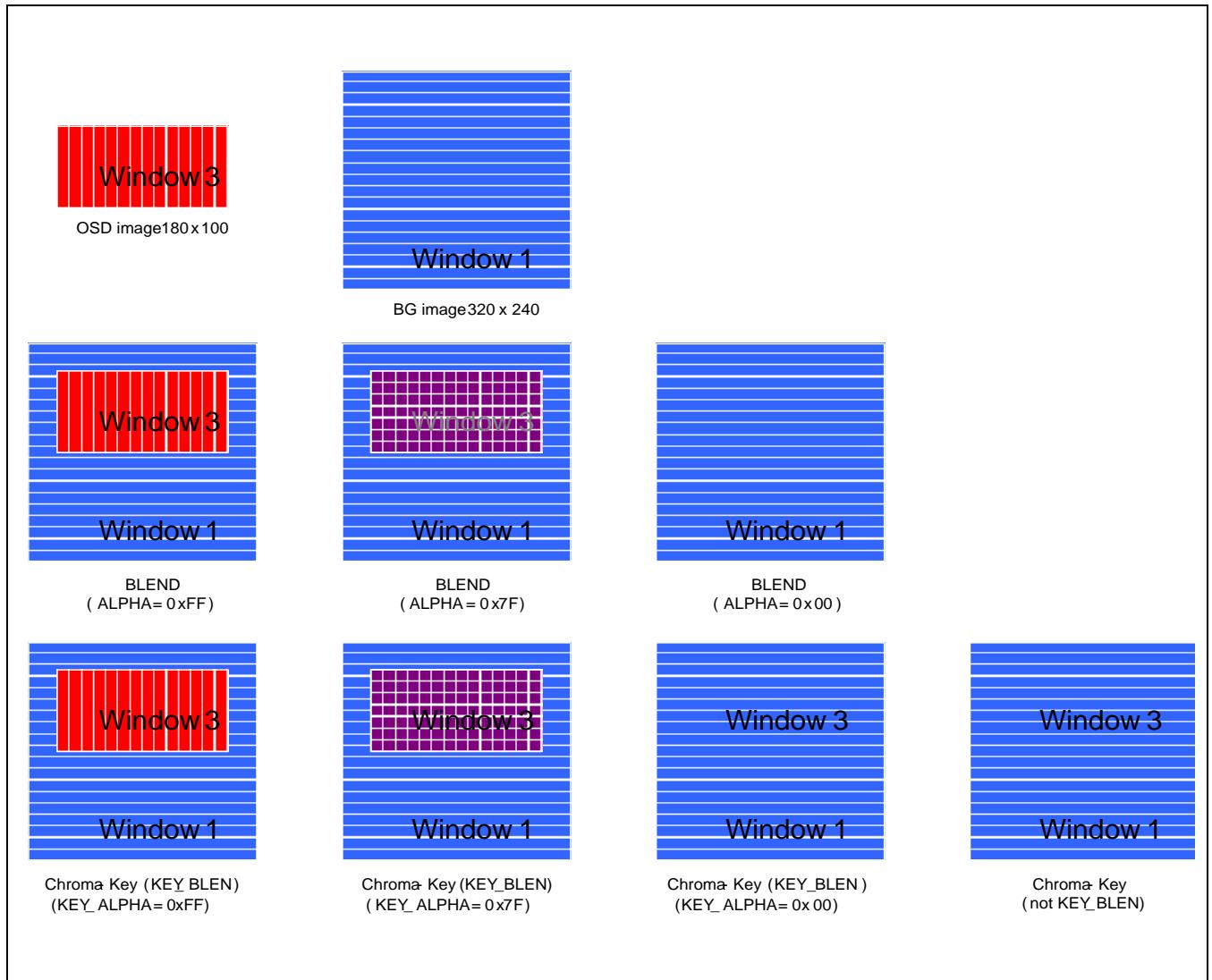


Figure 16-8 Color-Key Function Configurations

16.3.5.4 Blending and Color-Key Function

The display controller supports simultaneous blending function--with two transparency factors and Color-Key function in the same window.

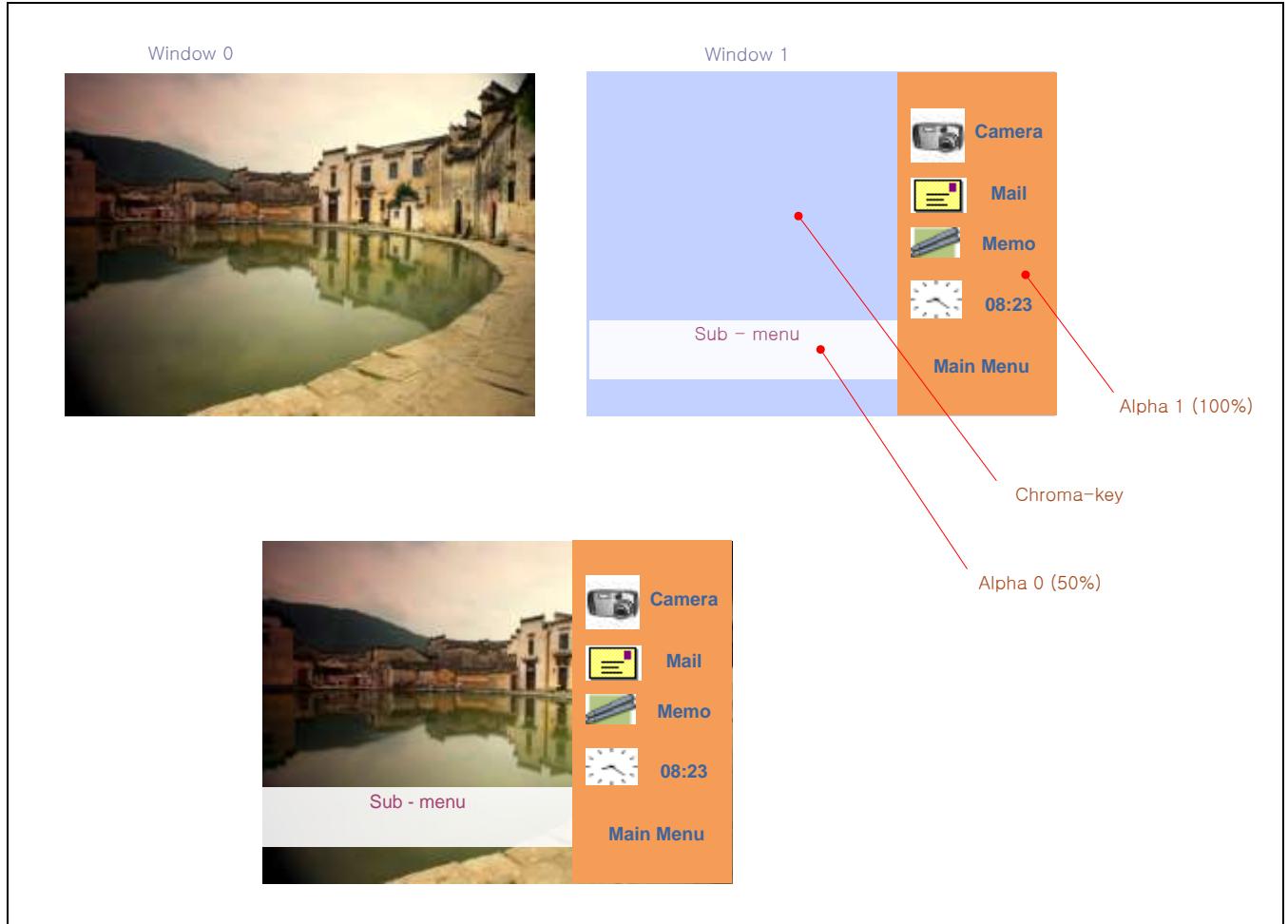


Figure 16-9 Blending and Color-Key Function

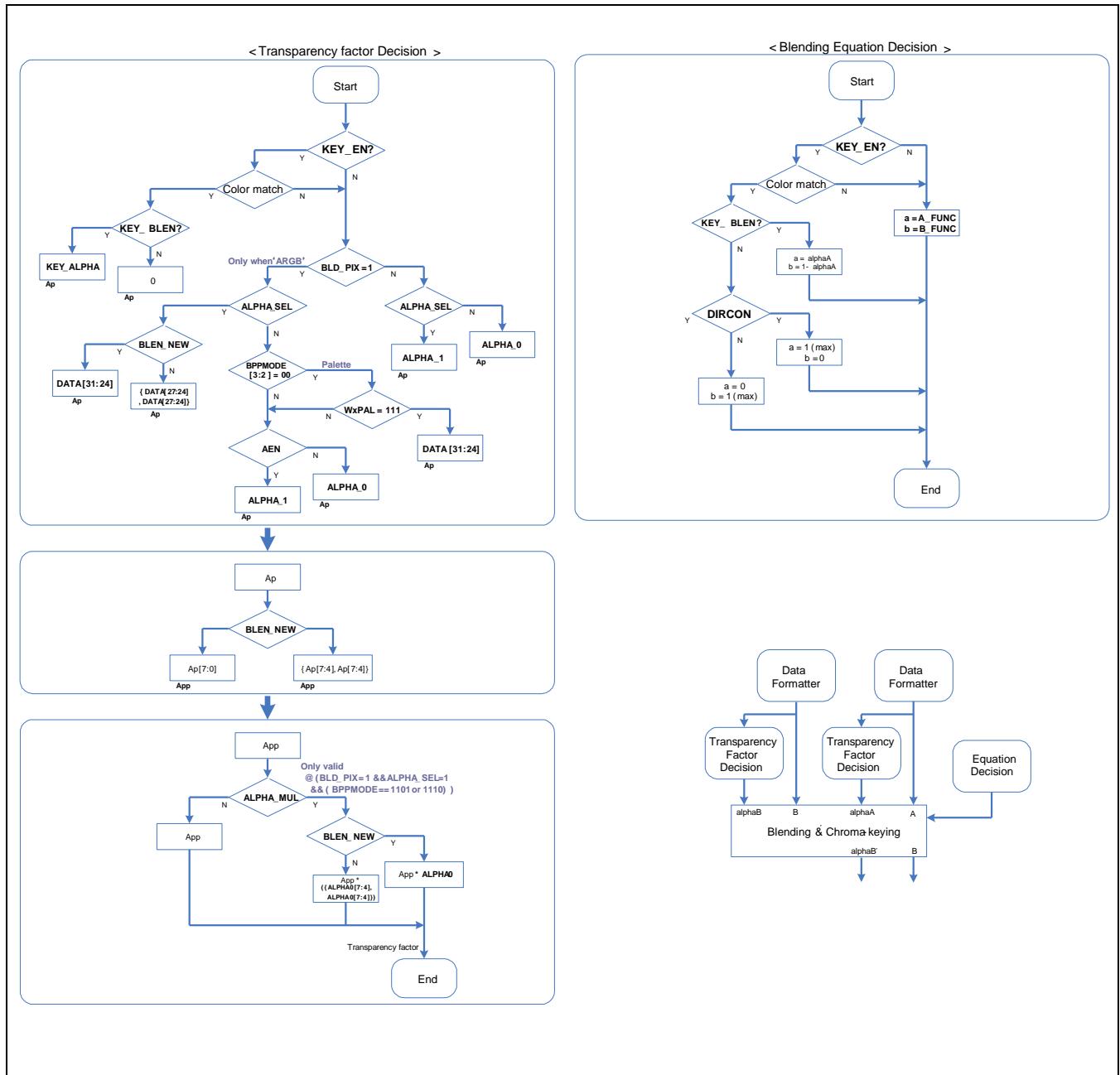


Figure 16-10 Blending Decision Diagram

16.3.6 Image Enhancement

16.3.6.1 Overview of Image Enhancement

One of the main functions of the VPRCS module is Image Enhancement. The display controller supports Gamma, Hue, Color Gain, and Pixel Compensation Control functions.

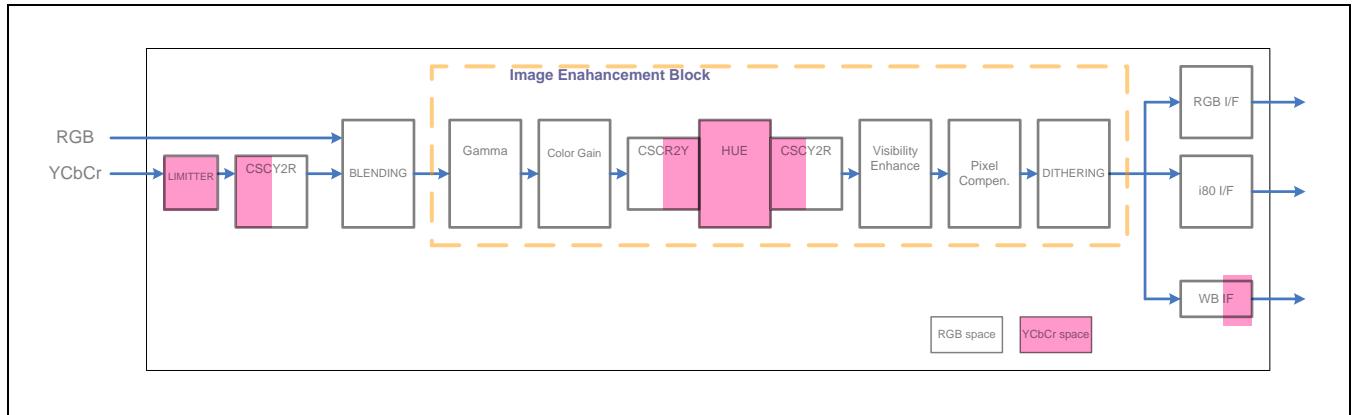


Figure 16-11 Image Enhancement Flow

16.3.7 VTIME Controller Operation

VTIME comprises of two blocks, namely:

- VTIME_RGB_TV for RGB timing control
- VTIME_I80 for indirect i80 interface timing control

16.3.7.1 RGB Interface Controller

VTIME generates control signals such as RGB_VSYNC, RGB_HSYNC, RGB_VDEN, and RGB_VCLK signal for the RGB interface. These control signals are used while configuring the VIDTC0N0/1/2 registers in the VSFR register.

Based on the programmable configurations of display control registers in the VSFR, the VTIME module generates programmable control signals that support different types of display devices.

The RGB_VSYNC signal causes the LCD line pointer to begin at the top of display. The configuration of both HOZVAL field and LINEVAL registers control pulse generation of RGB_VSYNC and RGB_HSYNC. Based on the following equations, the size of the LCD panel determines HOZVAL and LINEVAL:

- HOZVAL = (Horizontal display size) – 1
- LINEVAL = (Vertical display size) – 1

The CLKVAL field in VIDCON0 register controls the rate of RGB_VCLK signal. $\text{RGB_VCLK} (\text{Hz}) = \text{SCLK_FIMDx} / (\text{CLKVAL} + 1)$, where $\text{CLKVAL} \geq 1$

where, SCLK_FIMDx ($x = 0, 1$)

Table 16-5 [Table 16-5](#) defines the relationship of RGB_VCLK and CLKVAL. The minimum value of CLKVAL is 1.

- $\text{RGB_VCLK} (\text{Hz}) = \text{SCLK_FIMDx} / (\text{CLKVAL} + 1)$, where $\text{CLKVAL} \geq 1$

where, SCLK_FIMDx ($x = 0, 1$)

**Table 16-5 Relation 16 bpp Between VCLK and CLKVAL
(TFT, Frequency of Video Clock Source = 60 MHz)**

CLKVAL	60 MHz/X	VCLK
2	60 MHz/3	20.0 MHz
3	60 MHz/4	15.0 MHz
:	:	:
63	60 MHz/64	937.5 kHz

VSYNC, VBPD, VFPD, HSYNC, HBPD, HFpd, HOZVAL, and LINEVAL configure RGB_HSYNC and RGB_VSYNC signal. For more information, refer to $\text{RGB_VCLK (Hz)} = \text{SCLK_FIMDx} / (\text{CLKVAL} + 1)$, where $\text{CLKVAL} \geq 1$

where, SCLK_FIMDx ($x = 0, 1$)

Table 16-5 [Table 16-5](#).

The frame rate is RGB_VSYNC signal frequency. The frame rate is related to the field of VSYNC, VBPD, VFPD, LINEVAL, HSYNC, HBPD, HFpd, HOZVAL, and CLKVAL registers. Most LCD drivers need their own adequate frame rate.

To calculate frame rate, use the following equation:

$$\begin{aligned}\text{Frame Rate} = 1 / & \{ \{ (\text{VSPW}+1) + (\text{VBPD}+1) + (\text{LINEVAL} + 1) + (\text{VFPD}+1) \} \times \{ (\text{HSPW}+1) + (\text{HBPD} + 1) \\ & + (\text{HFpd}+1) + (\text{HOZVAL} + 1) \} \times \{ (\text{CLKVAL}+1) / (\text{Frequency of Clock source}) \} \}\end{aligned}$$

16.3.7.2 I80 Interface Controller

VTIME_I80 controls display controller for CPU style LDI.

It has the following functions:

- Generates I80 Interface Control Signals
- CPU style LDI Command Control
- Timing Control for VDMA and VDPRCS

16.3.7.3 Output Control Signal Generation

VTIME_I80 generates SYS_CS0, SYS_CS1, SYS_WE, and SYS_RS control signals (For Timing Diagram, refer to [RGB_VCLK](#) (Hz) = $SCLK_FIMDx$ / (CLKVAL+1), where CLKVAL >= 1

where, SCLK_FIMDx (x = 0, 1)

[Table](#) 16-5.

Their timing parameters, LCD_CS_SETUP, LCD_WR_SETUP, LCD_WR_ACT, and LCD_WR_HOLD are set through I80IFCONA0 and I80IFCONA1 SFRs.

16.3.7.4 Partial Display Control

Although partial display is the main feature of CPU style LDI, VTIME_I80 does not support this function in hardware logic.

This function is implemented by SFR setting (LINEVAL, HOZVAL, OSD_LeftTopX_F, OSD_LeftTopY_F, OSD_RightBotX_F, OSD_RightBotY_F, PAGEWIDTH, and OFFSIZE).

16.3.7.5 LDI Command Control

LDI receives both command and data. Command specifies an index for selecting the SFR in LDI. In control signal for command and data, only SYS_RS signal has a special function. Usually, SYS_RS has a polarity of '1' for issuing command and vice versa.

Display controller has two kinds of command control:

- Auto command
- Normal command

Auto command is issued automatically, that is, without software control and at a pre-defined rate (rate = 2, 4, 6, ... 30). If the rate is equal to 4, it implies that auto commands are send to LDI at the end of every 4 image frames. The software control issues Normal command.

16.3.8 Setting of Commands

16.3.8.1 Auto Command

If 0x1 (index), 0x32, 0x2 (index), 0x8f, 0x4 (index), or 0x99 are required to be sent to LDI at every 10 frames, the following steps are recommended:

- LDI_CMD0 ← 0x1, LDI_CMD1 ← 0x32, LDI_CMD2 ← 0x2,
- LDI_CMD3 ← 0x8f, LDI_CMD4 ← 0x4, LDI_CMD5 ← 0x99
- CMD0_EN ← 0x2, CMD1_EN ← 0x2, CMD2_EN ← 0x2,
- CMD3_EN ← 0x2, CMD4_EN ← 0x2, CMD5_EN ← 0x2
- CMD0_RS ← 0x1, CMD1_RS ← 0x0, CMD2_RS ← 0x1,
- CMD3_RS ← 0x0, CMD4_RS ← 0x1, CMD5_RS ← 0x0
- AUTO_CMD_RATE ← 0x5

NOTE:

1. For checking RS polarity, refer to your LDI specification.
2. It is not required to pack LDI_CMD from LDI_CMD0 to LDI_CMD11 contiguously. For example, it is only possible to use LDI_CMD0, LDI_CMD3, and LDI_CMD11.
3. Maximum 12 auto commands are available.

16.3.8.2 Normal Command

To execute Normal command, follow these steps:

- Put commands into LDI_CMD0 – 11 (maximum 12 commands).
- Set CMDx_EN in LDI_CMDCON0 to enable normal command × (For example, if you want to enable command 4, you have to set CMD4_EN to 0x01).
- Set NORMAL_CMD_ST in I80IFCONB0/1.

The display controller has the following characteristics for command operations:

- Auto/Normal/Auto and Normal command mode is possible for each of the 12 commands.
- Sends 12 maximum commands between frames in its normal operation (Normal operation means ENVID = 1 and video data is displayed in LCD panel).
- Issues commands in the order of CMD0 → CMD1 → CMD2 → CMD3 → ... → CMD10 → CMD11.
- Skips disabled commands (CMDx_EN = 0x0).
 - Sends over 12 commands (Possible in Normal command and system initialization).
 - Set 12 LDI_CMDx, CMDx_EN, and CMDx_RS.
 - Set NORMAL_CMD_ST.
 - Read NORMAL_CMD_ST with polling. If 0, go to NORMAL_CMD_ST setting.

1.2.7.2.1 Command Setting Example

- CMD0_EN = 2'b10, CMD1_EN = 2'b11, CMD2_EN = 2'b01, CMD3_EN = 2'b11, and CMD4_EN = 2'b01
(Auto Command: CMD0, CMD1, CMD3, Normal Command: CMD1, CMD2, CMD3, and CMD4)
- AUTO_COMMAND_RATE = 4'b0010 (per 4 frames)
- CMD0_RS = 1, CMD1_RS = 1, CMD2_RS = 0, CMD3_RS = 1, and CMD4_RS = 0.
- RSPOL = 0

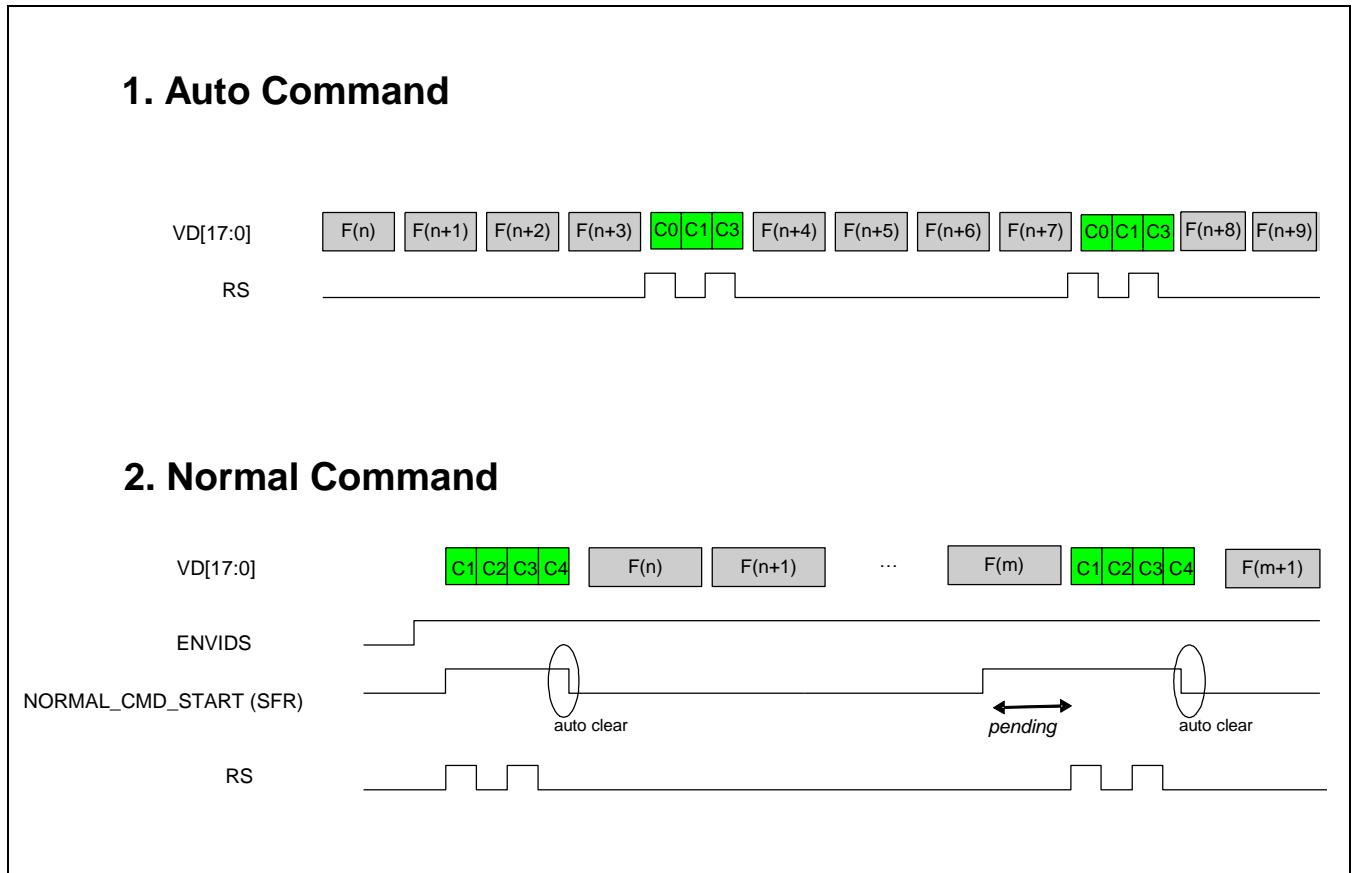


Figure 16-12 Sending Command

1.2.7.2.2 Indirect I80 Interface Trigger

VTIME_I80 starts its operation when a software trigger occurs. There are two kinds of triggers. However, software trigger is generated by setting TRGCON SFR.

16.3.8.3 Interrupt

Completion of one frame generates Frame Done Interrupt.

16.3.8.3.1 Indirect I80 Interface Output Mode

The following table shows the output mode of Indirect i80 interface based on mode@VIDCON0.

Table 16-6 i80 Output Mode

VIDCON0 Register	Value	BPP	Bus Width	Split	DATA	Command
DSI_EN	1	24	24	X	{R[7:0],G[7:0],B[7:0]}	CMD [23:0]
L0/1_DATA	000	16	16	X	{R[7:3],G[7:2],B[7:3]}	CMD [15:0]
	001	18	16	O (1st) (2nd)	{R[7:2],G[7:2],B[7:4]} { 14'b0,B[3:2]}	CMD [15:0] -
	010	18	9	O (1st) (2nd)	{ R[7:2],G[7:5]} { G[4:2],B[7:2]}	CMD [17:9] CMD [8:0]
	011	24	16	O (1st) (2nd)	{ R[7:0],G[7:0]} { B[7:0], 8'b0}	- -
	100	18	18	X	{ R[7:2],G[7:2],B[7:2]}	CMD [17:0]
	101	16	8	O (1st) (2nd)	{ R[7:3],G[7:5]} { G[4:2],B[7:3]}	CMD [15:8] CMD [7:0]

16.3.9 Virtual Display

The display controller supports hardware horizontal or vertical scrolling. If the screen scrolls, change the fields of LCDBASEU and LCDBASEL (refer to [Figure 16-13](#)), but not the values of PAGEWIDTH and OFFSIZE. The size of video buffer in which the image is stored should be larger than the LCD panel screen size.

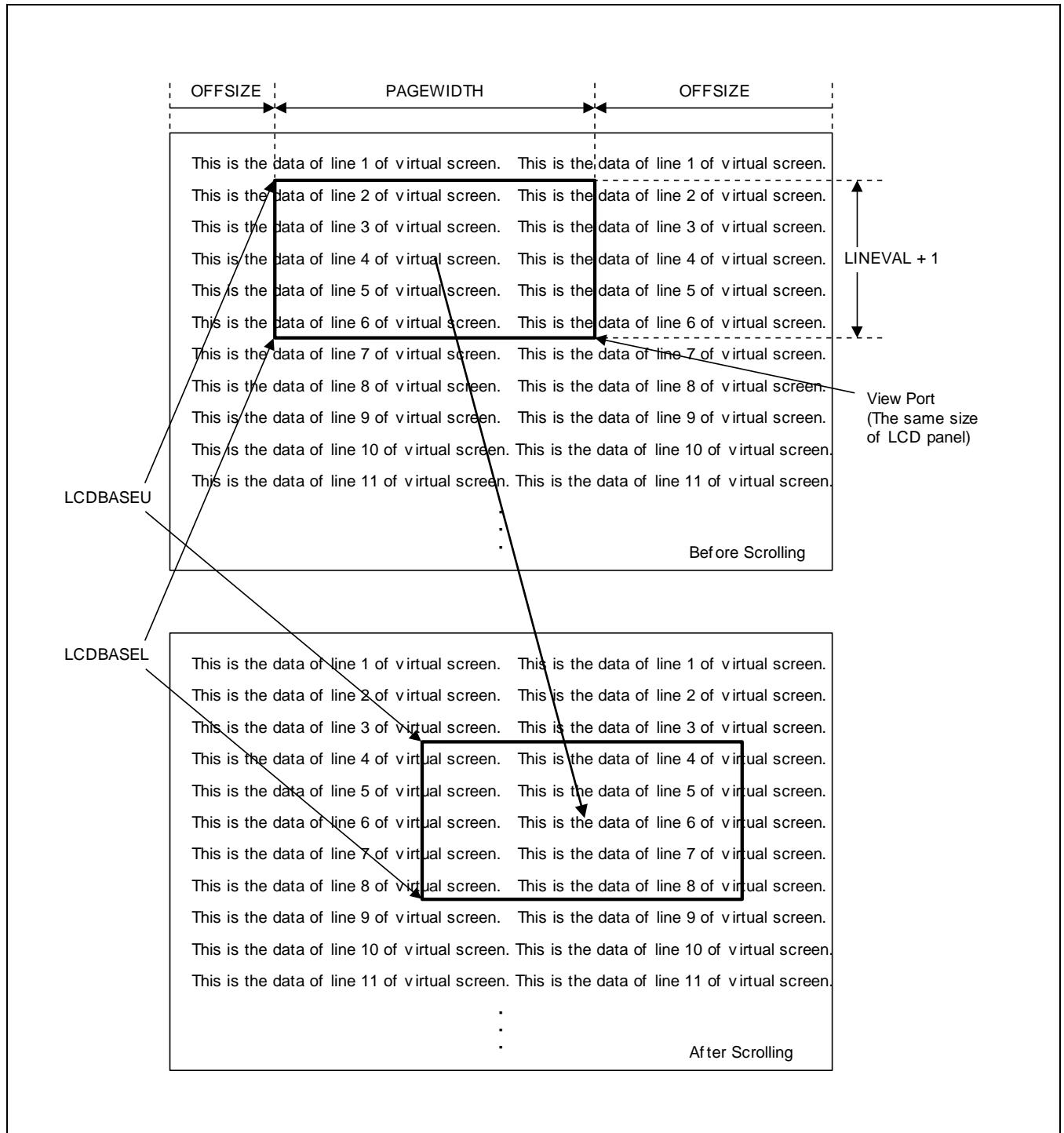


Figure 16-13 Example of Scrolling in Virtual Display

16.3.10 RGB Interface Spec

16.3.10.1 Signals

Signal	In/ Out	Description	Display Controller 0		Display Controller 1	
			Pads	GPIO Control	Pads	GPIO Control
LCD_HSYN_C	O	Horizontal Sync. Signal	XvHSYNC	GPF0CON[0]	XmdmRn	GPE0CON[2]
LCD_VSYN_C	O	Vertical Sync. Signal	XvVSYNC	GPF0CON[1]	XmdmRQn	GPE0CON[1]
LCD_VCLK	O	LCD Video Clock	XvVCLK	GPF0CON[3]	XmdmWE	GPE0CON[0]
LCD_VDEN	O	Data Enable	XvVDEN	GPF0CON[2]	XmdmADV	GPE0CON[4]
LCD_VD[0]	O	RGB data output	XvVD_0	GPF0CON[4]	XmdmADDR[0]	GPE1CON[0]
LCD_VD[1]	O	RGB data output	XvVD_1	GPF0CON[5]	XmdmADDR[1]	GPE1CON[1]
LCD_VD[2]	O	RGB data output	XvVD_2	GPF0CON[6]	XmdmADDR[2]	GPE1CON[2]
LCD_VD[3]	O	RGB data output	XvVD_3	GPF0CON[7]	XmdmADDR[3]	GPE1CON[3]
LCD_VD[4]	O	RGB data output	XvVD_4	GPF1CON[0]	XmdmADDR[4]	GPE1CON[4]
LCD_VD[5]	O	RGB data output	XvVD_5	GPF1CON[1]	XmdmADDR[5]	GPE1CON[5]
LCD_VD[6]	O	RGB data output	XvVD_6	GPF1CON[2]	XmdmADDR[6]	GPE1CON[6]
LCD_VD[7]	O	RGB data output	XvVD_7	GPF1CON[3]	XmdmADDR[7]	GPE1CON[7]
LCD_VD[8]	O	RGB data output	XvVD_8	GPF1CON[4]	XmdmADDR[8]	GPE2CON[0]
LCD_VD[9]	O	RGB data output	XvVD_9	GPF1CON[5]	XmdmADDR[9]	GPE2CON[1]
LCD_VD[10]	O	RGB data output	XvVD_10	GPF1CON[6]	XmdmADDR[10]	GPE2CON[2]
LCD_VD[11]	O	RGB data output	XvVD_11	GPF1CON[7]	XmdmADDR[11]	GPE2CON[3]
LCD_VD[12]	O	RGB data output	XvVD_12	GPF2CON[0]	XmdmADDR[12]	GPE2CON[4]
LCD_VD[13]	O	RGB data output	XvVD_13	GPF2CON[1]	XmdmADDR[13]	GPE2CON[5]
LCD_VD[14]	O	RGB data output	XvVD_14	GPF2CON[2]	XmdmDATA[0]	GPE3CON[0]
LCD_VD[15]	O	RGB data output	XvVD_15	GPF2CON[3]	XmdmDATA[1]	GPE3CON[1]
LCD_VD[16]	O	RGB data output	XvVD_16	GPF2CON[4]	XmdmDATA[2]	GPE3CON[2]
LCD_VD[17]	O	RGB data output	XvVD_17	GPF2CON[5]	XmdmDATA[3]	GPE3CON[3]
LCD_VD[18]	O	RGB data output	XvVD_18	GPF2CON[6]	XmdmDATA[4]	GPE3CON[4]
LCD_VD[19]	O	RGB data output	XvVD_19	GPF2CON[7]	XmdmDATA[5]	GPE3CON[5]
LCD_VD[20]	O	RGB data output	XvVD_20	GPF3CON[0]	XmdmDATA[6]	GPE3CON[6]
LCD_VD[21]	O	RGB data output	XvVD_21	GPF3CON[1]	XmdmDATA[7]	GPE3CON[7]
LCD_VD[22]	O	RGB data output	XvVD_22	GPF3CON[2]	XmdmDATA[8]	GPE4CON[0]
LCD_VD[23]	O	RGB data output	XvVD_23	GPF3CON[3]	XmdmDATA[9]	GPE4CON[1]

When RGB Interface is used, VT_LBLKx Bit Fields in LCDBLKC_CFG (0x1001_0210) Register Should be set RGB Interface out (2'b00), even though DSI Video Mode used.

16.3.10.2 LCD RGB Interface Timing

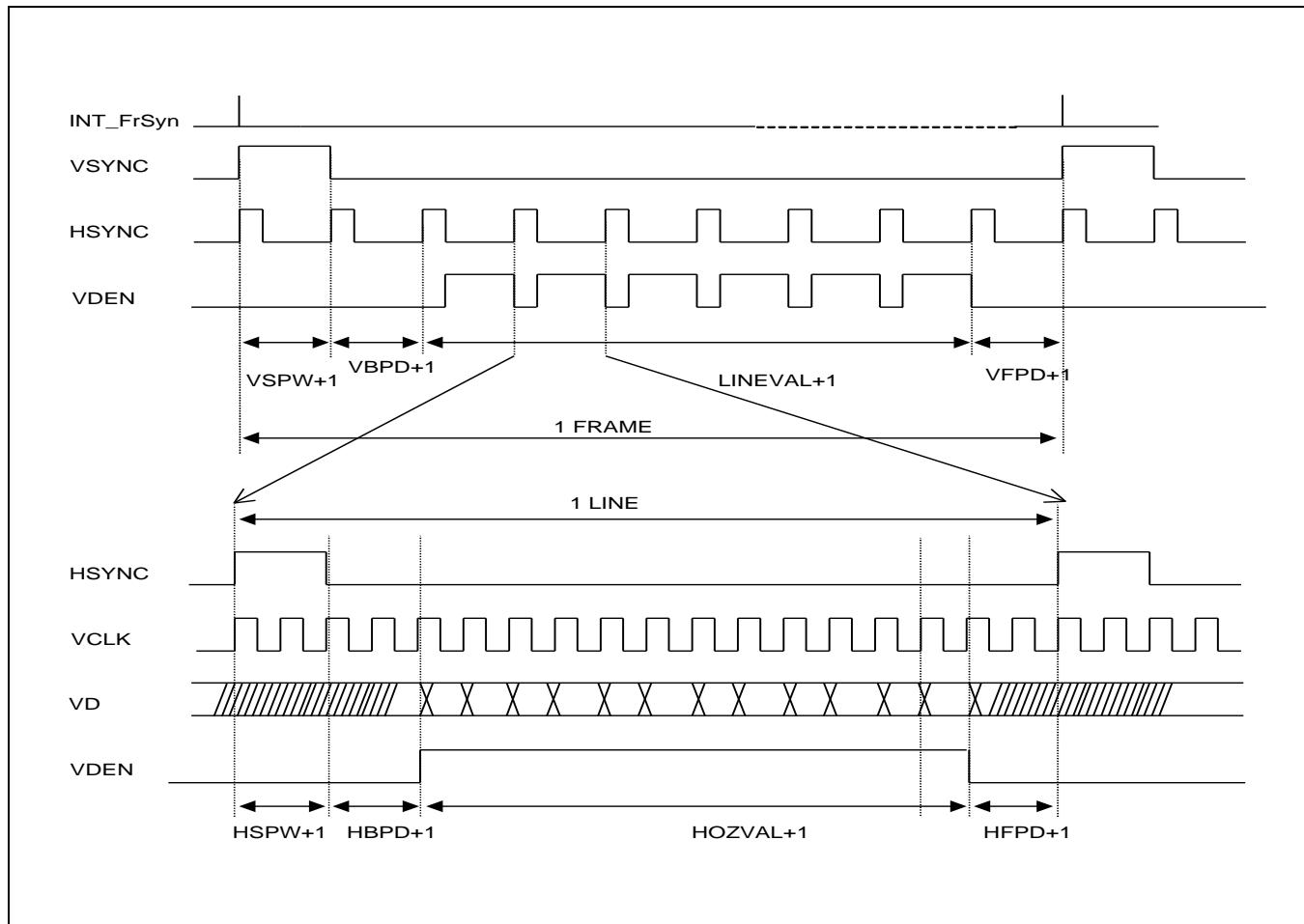


Figure 16-14 LCD RGB Interface Timing

16.3.10.3 Parallel Output

16.3.10.3.1 General 24-bit Output (RGBPSEL = 0, RGB_SKIP_EN = 0)

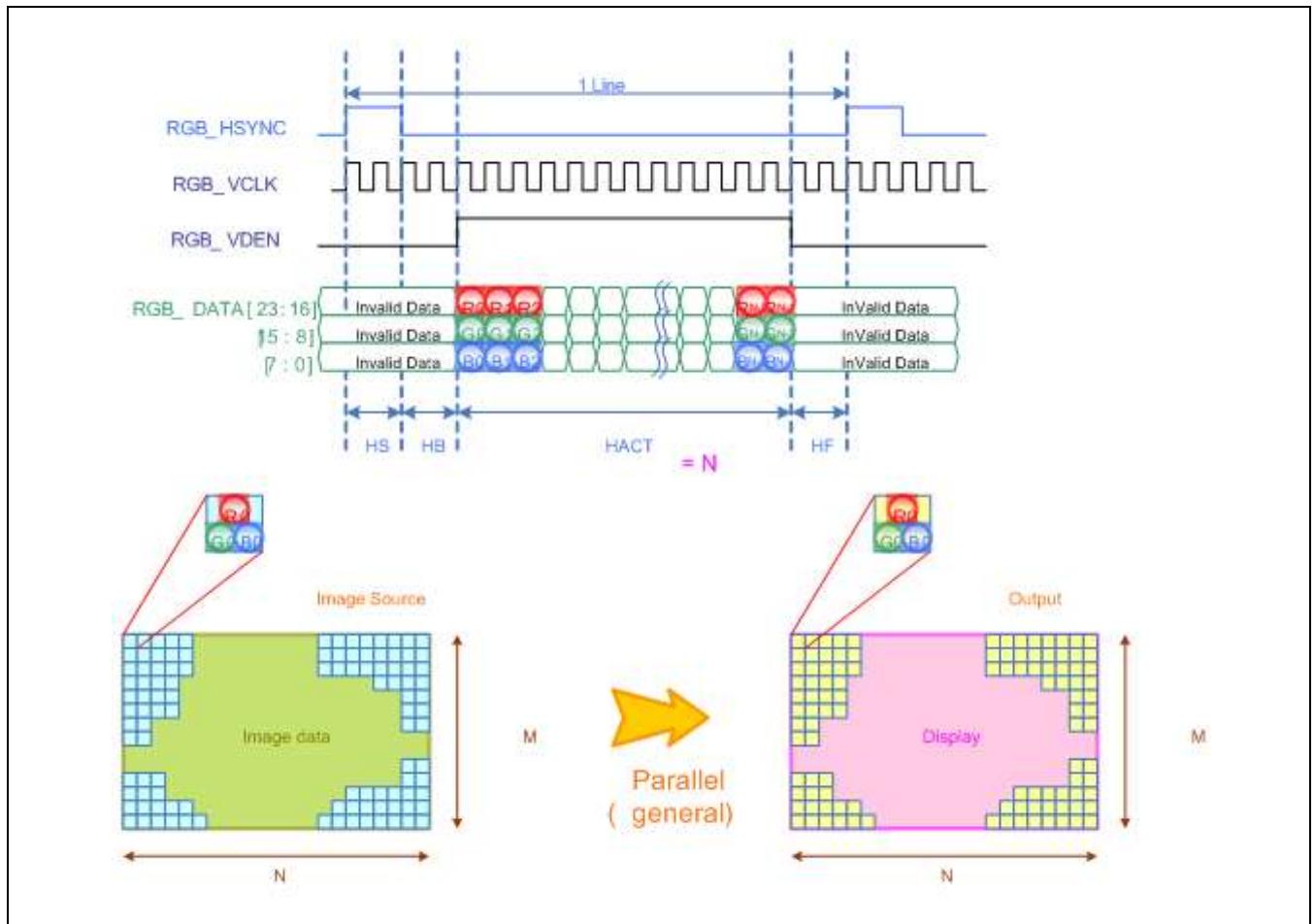


Figure 16-15 LCD RGB Interface Timing (RGB Parallel)

16.3.10.4 Serial 8-bit Output

16.3.10.4.1 General 8-bit Output (RGBSPSEL = 1, RGB_DUMMY_EN = 0)

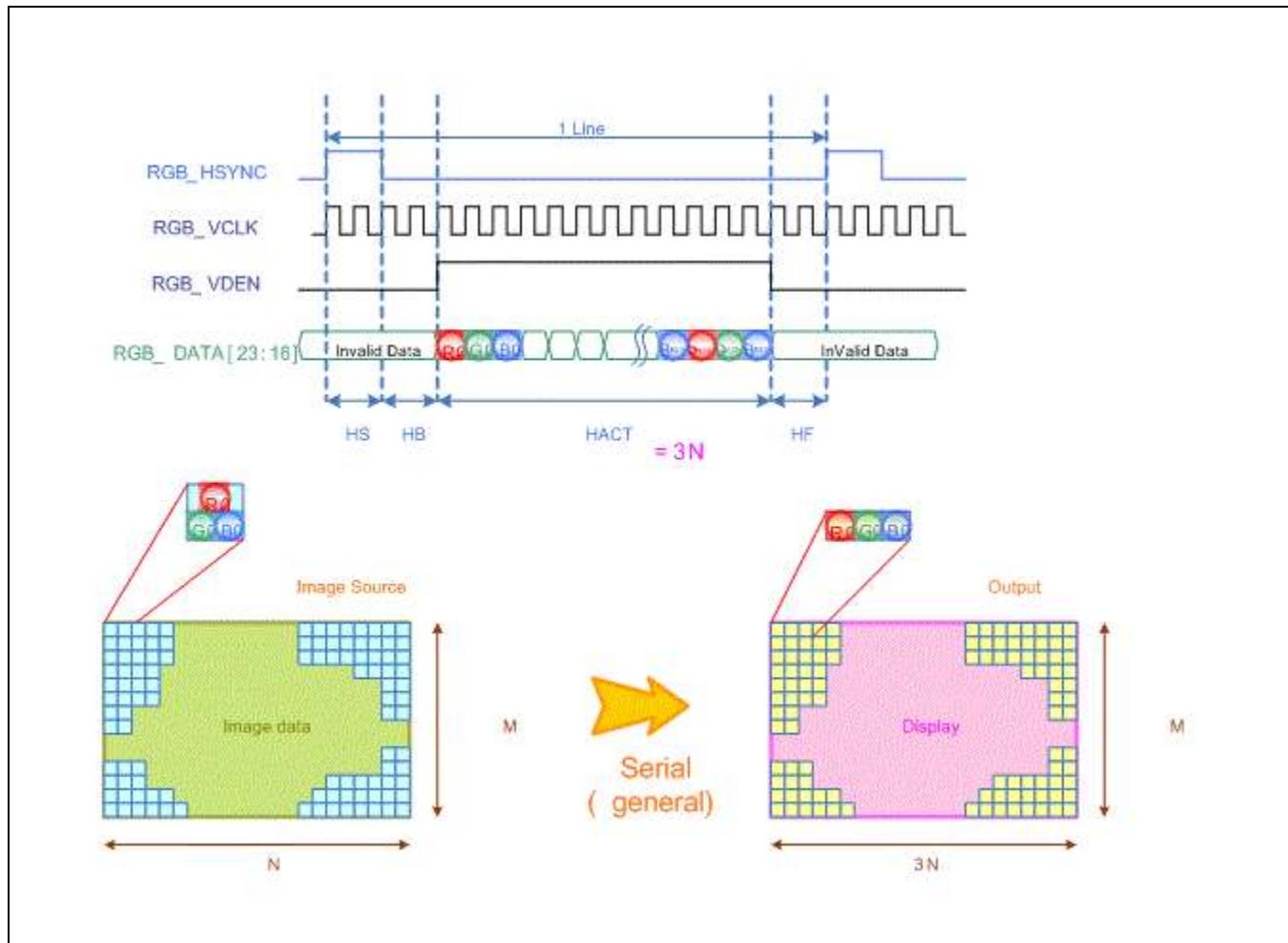


Figure 16-16 LCD RGB Interface Timing (RGB Serial, Dummy Disable)

16.3.10.5 Output Configuration Structure

16.3.10.5.1 Color Order Control

"RGB_ORDER_O" controls odd line color structure. On the other hand, "RGB_ORDER_E" @VIDCON2 controls even line color structure.

RGB_ORDERO/E)	Output width	
	24 bit	8 bit
000	[23 - 0]	R → G → B → 1st → 2nd → 3rd
001	G → B → R → 1st → 2nd → 3rd	R → G → B → 1st → 2nd → 3rd
010	B → R → G → 1st → 2nd → 3rd	B → R → G → 1st → 2nd → 3rd
100	R → B → G → 1st → 2nd → 3rd	B → R → G → 1st → 2nd → 3rd
101	R → G → B → 1st → 2nd → 3rd	R → B → G → 1st → 2nd → 3rd
110	G → R → B → 1st → 2nd → 3rd	G → B → R → 1st → 2nd → 3rd

Figure 16-17 LCD RGB Output Order

16.3.11 LCD Indirect i80 System Interface

16.3.11.1 Signals

Signal	In/ Out	Description	Display Controller 0		Display Controller 1	
			Pad	GPIO Control	Pad	GPIO Control
SYS_VD[0]	I/O	Data bit 0	XvVD_0	GPF0CON[4]	XmdmADDR[0]	GPE1CON[0]
SYS_VD[1]	I/O	Data bit 1	XvVD_1	GPF0CON[5]	XmdmADDR[1]	GPE1CON[1]
SYS_VD[2]	I/O	Data bit 2	XvVD_2	GPF0CON[6]	XmdmADDR[2]	GPE1CON[2]
SYS_VD[3]	I/O	Data bit 3	XvVD_3	GPF1CON[7]	XmdmADDR[3]	GPE1CON[3]
SYS_VD[4]	I/O	Data bit 4	XvVD_4	GPF1CON[0]	XmdmADDR[4]	GPE1CON[4]
SYS_VD[5]	I/O	Data bit 5	XvVD_5	GPF1CON[1]	XmdmADDR[5]	GPE1CON[5]
SYS_VD[6]	I/O	Data bit 6	XvVD_6	GPF1CON[2]	XmdmADDR[6]	GPE1CON[6]
SYS_VD[7]	I/O	Data bit 7	XvVD_7	GPF1CON[3]	XmdmADDR[7]	GPE1CON[7]
SYS_VD[8]	I/O	Data bit 8	XvVD_8	GPF1CON[4]	XmdmADDR[8]	GPE2CON[0]
SYS_VD[9]	I/O	Data bit 9	XvVD_9	GPF1CON[5]	XmdmADDR[9]	GPE2CON[1]
SYS_VD[10]	I/O	Data bit 10	XvVD_10	GPF1CON[6]	XmdmADDR[10]	GPE2CON[2]
SYS_VD[11]	I/O	Data bit 11	XvVD_11	GPF1CON[7]	XmdmADDR[11]	GPE2CON[3]
SYS_VD[12]	I/O	Data bit 12	XvVD_12	GPF2CON[0]	XmdmADDR[12]	GPE2CON[4]
SYS_VD[13]	I/O	Data bit 13	XvVD_13	GPF2CON[1]	XmdmADDR[13]	GPE2CON[5]
SYS_VD[14]	I/O	Data bit 14	XvVD_14	GPF2CON[2]	XmdmDATA[0]	GPE3CON[0]
SYS_VD[15]	I/O	Data bit 15	XvVD_15	GPF2CON[3]	XmdmDATA[1]	GPE3CON[1]
SYS_VD[16]	I/O	Data bit 16	XvVD_16	GPF2CON[4]	XmdmDATA[2]	GPE3CON[2]
SYS_VD[17]	I/O	Data bit 17	XvVD_17	GPF2CON[5]	XmdmDATA[3]	GPE3CON[3]
SYS_CS0	O	Chip select for LCD0	XvHSYNC	GPF0CON[0]	XmdmRn	GPE0CON[2]
SYS_CS1	O	Chip select for LCD1	XvVSYNC	GPF0CON[1]	XmdmRQn	GPE0CON[1]
SYS_WE	O	Write enable	XvVCLK	GPF0CON[3]	XmdmWEn	GPE0CON[0]
SYS_OE	O	Output enable	XvSYS_OE	GPF3CON[5]	XmdmDATA[11]	GPE4CON[3]
SYS_RS/ SYS_ADD[0]	O	Address Output [0]	XvVDEN	GPF0CON[2]	XmdmADVN	GPE0CON[4]
SYS_ST/ SYS_ADD[1]	O	Address Output [1]	-	Internal Connection	-	Internal Connection

NOTE:

1. SYS_ST/SYS_ADD[1] is valid in DSI Mode (VIDCON0 [30] = 1)
 SYS_ADD [1] = SYS_ST: 0 when VDOUT is from Frame
 SYS_ADD [1] = SYS_ST: 1 when VDOUT is from Command
2. When RGB Interface is used, VT_LBLKx Bit Fields in LCDBLKC_CFG (0x1001_0210) Register Should be set i80 Interface out (2'b01) even though DSI Command Mode used.

16.3.11.2 Indirect i80 System Interface WRITE Cycle Timing

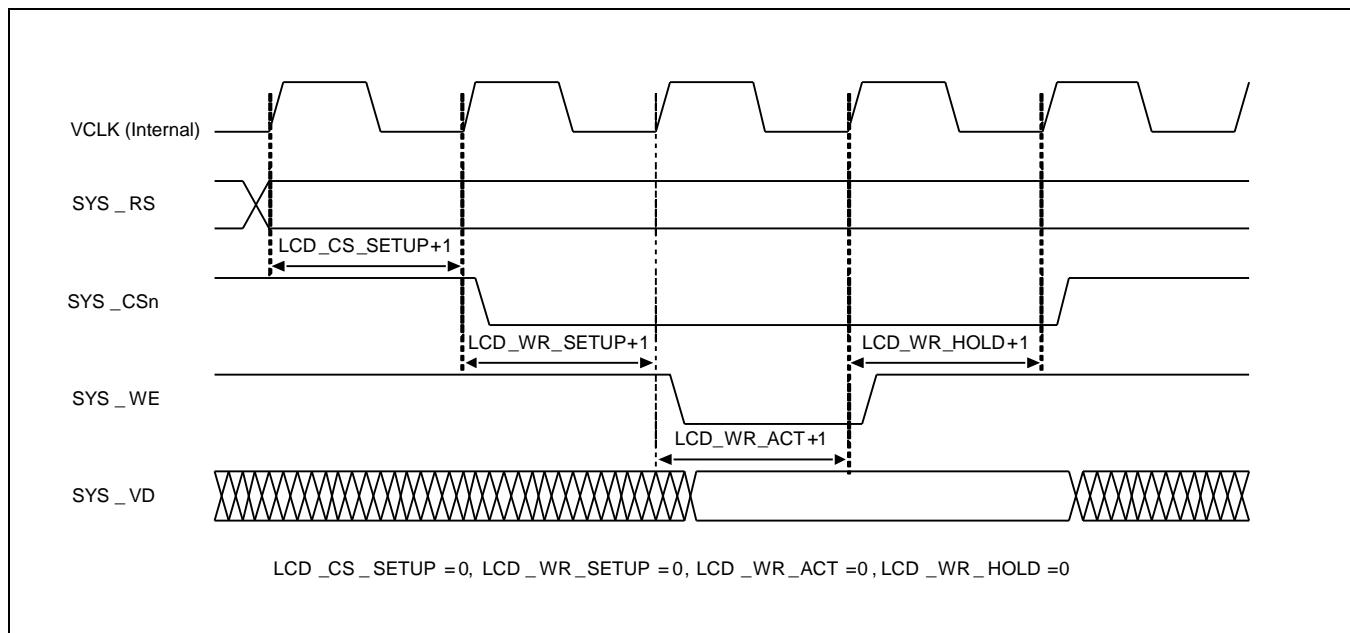


Figure 16-18 Indirect i80 System Interface WRITE Cycle Timing

Table 16-7 Timing Reference Code (XY Definition)

	Parallel RGB			Serial RGB	
	24 bpp (888)	18 bpp (666)	16 bpp (565)	24 bpp (888)	18 bpp (666)
VD[23]	R[7]	R[5]	R[4]	D[7]	D[5]
VD[22]	R[6]	R[4]	R[3]	D[6]	D[4]
VD[21]	R[5]	R[3]	R[2]	D[5]	D[3]
VD[20]	R[4]	R[2]	R[1]	D[4]	D[2]
VD[19]	R[3]	R[1]	R[0]	D[3]	D[1]
VD[18]	R[2]	R[0]	–	D[2]	D[0]
VD[17]	R[1]	–	–	D[1]	–
VD[16]	R[0]	–	–	D[0]	–
VD[15]	G[7]	G[5]	G[5]	–	–
VD[14]	G[6]	G[4]	G[4]	–	–
VD[13]	G[5]	G[3]	G[3]	–	–
VD[12]	G[4]	G[2]	G[2]	–	–
VD[11]	G[3]	G[1]	G[1]	–	–
VD[10]	G[2]	G[0]	G[0]	–	–
VD[9]	G[1]	–	–	–	–
VD[8]	G[0]	–	–	–	–
VD[7]	B[7]	B[5]	B[4]	–	–
VD[6]	B[6]	B[4]	B[3]	–	–
VD[5]	B[5]	B[3]	B[2]	–	–
VD[4]	B[4]	B[2]	B[1]	–	–
VD[3]	B[3]	B[1]	B[0]	–	–
VD[2]	B[2]	B[0]	–	–	–
VD[1]	B[1]	–	–	–	–
VD[0]	B[0]	–	–	–	–

16.4 I/O Description

Signal	In/Out	Description	PAD for Display Controller0	PAD for Display Controller1	Type (1)
RGB Interface					
LCD_HSYNC	Out	Horizontal Sync. Signal	XvHsync	XmdmRn	Muxed
LCD_VSYNC	Out	Vertical Sync. Signal	XvVsync	XmdmRQn	Muxed
LCD_VCLK	Out	LCD Video Clock	XvVclk	XmdmWEn	Muxed
LCD_VDEN	Out	Data Enable	XvVden	XmdmADVn	Muxed
LCD_VD[23:0]	Out	RGB data output	XvVD_23 to XvVD_0	XmdmADDR0 to XmdmADDR13, XmdmDATA0 to 9	Muxed
CPU Interface (i80)					
SYS_VD[17:0]	In/Out	Data to/from Display Controller from/to Display Module	XvVD_17 to XvVD_0	XmdmADDR0 to XmdmADDR13, XmdmDATA0 to 3	Muxed
SYS_CS0	Out	Chip select for LCD0	XvHsync	XmdmRn	Muxed
SYS_CS1	Out	Chip select for LCD1	XvVsync	XmdmRQn	Muxed
SYS_WE	Out	Write enable	XvVclk	XmdmWEn	Muxed
SYS_OE	Out	Output enable	XvSYS_OE	XmdmDATA11	Muxed
SYS_RS/SYS_ADD[0]	Out	Address Output SYS_ADD[0] is Register/State select	XvVden	XmdmADVn	Muxed
Etc.					
VSYNC_LDI	Out	VSYNC signal for Vsync interface (2)	XvVsync_LDI	XmdmDATA10	Muxed
LCD_FRM	Out	Frame Sync Signal for general use (3)	XpwmTout_0	XmdmCSn	Muxed

NOTE:

1. Type field indicates whether pads are dedicated to signal or pads are connected to multiplexed signals.
2. This signal is controlled by VSYNCEN register.
3. This signal is controlled by FRMEN control register.

16.5 Register Description

16.5.1 Overview

Use the following registers to configure display controller:

1. VIDCON0: Configures video output format and displays enable/disable.
2. VIDCON1: Specifies RGB I/F control signal.
3. VIDCON2: Specifies output data format control.
4. VIDCON3: Specifies image enhancement control.
5. I80IFCONx: Specifies CPU interface control signal.
6. VIDTCONx: Configures video output timing and determines the size of display.
7. WINCONx: Specifies each window feature setting.
8. VIDOSDxA, VIDOSDxB: Specifies window position setting.
9. VIDOSDxC,D: Specifies OSD size setting.
10. VIDWxALPHA0/1: Specifies alpha value setting.
11. BLENDEQx: Specifies blending equation setting.
12. VIDWxxADDx: Specifies source image address setting.
13. WxKEYCONx: Specifies color key setting register.
14. WxKEYALPHA: Specifies color key alpha value setting.
15. WINxMAP: Specifies window color control.
16. GAMMALUT_xx: Specifies gamma value setting.
17. COLORGAINCON: Specifies color gain value setting.
18. HUExx: Specifies Hue coefficient and offset value setting.
19. WPALCON: Specifies palette control register.
20. WxRTQOSCON: Specifies RTQoS control register.
21. WxPDATAx: Specifies Window Palette Data of each Index.
22. SHDOWCON: Specifies Shadow control register.
23. WxRTQOSCON: Specifies QoS control register.

16.5.2 Register Map Summary

- Base Address: 0x11C0_0000 (Display Controller 0)
- Base Address: 0x1200_0000 (Display Controller 1)

Register	Offset	Description	Reset Value
Control Registers			
VIDCON0	0x0000	Specifies video control 0 register.	0x0000_0000
VIDCON1	0x0004	Specifies video control 1 register.	0x0000_0000
VIDCON2	0x0008	Specifies video control 2 register.	0x0000_0000
VIDCON3	0x000C	Specifies video control 3 register.	0x0000_0000
VIDTCOM0	0x0010	Specifies video time control 0 register.	0x0000_0000
VIDTCOM1	0x0014	Specifies video time control 1 register.	0x0000_0000
VIDTCOM2	0x0018	Specifies video time control 2 register.	0x0000_0000
VIDTCOM3	0x001C	Specifies video time control 3 register.	0x0000_0000
WINCON0	0x0020	Specifies window control 0 register.	0x0000_0000
WINCON1	0x0024	Specifies window control 1 register.	0x0000_0000
WINCON2	0x0028	Specifies window control 2 register.	0x0000_0000
WINCON3	0x002C	Specifies window control 3 register.	0x0000_0000
WINCON4	0x0030	Specifies window control 4 register.	0x0000_0000
SHADOWCON	0x0034	Specifies window shadow control register.	0x0000_0000
WINCHMAP2	0x003C	Specifies window, channel mapping control register.	0x7D51_7D51
VIDOSD0A	0x0040	Specifies video window 0's position control register.	0x0000_0000
VIDOSD0B	0x0044	Specifies video window 0's position control register.	0x0000_0000
VIDOSD0C	0x0048	Specifies video window 0's size control register.	0x0000_0000
VIDOSD1A	0x0050	Specifies video window 1's position control register.	0x0000_0000
VIDOSD1B	0x0054	Specifies video window 1's position control register	0x0000_0000
VIDOSD1C	0x0058	Specifies video window 1's alpha control register.	0x0000_0000
VIDOSD1D	0x005C	Specifies video window 1's size control register.	0x0000_0000
VIDOSD2A	0x0060	Specifies video window 2's position control register.	0x0000_0000
VIDOSD2B	0x0064	Specifies video window 2's position control register.	0x0000_0000
VIDOSD2C	0x0068	Specifies video window 2's alpha control register.	0x0000_0000
VIDOSD2D	0x006C	Specifies video window 2's size control register.	0x0000_0000
VIDOSD3A	0x0070	Specifies video window 3's position control register.	0x0000_0000
VIDOSD3B	0x0074	Specifies video window 3's position control register.	0x0000_0000
VIDOSD3C	0x0078	Specifies video window 3's alpha control register.	0x0000_0000
VIDOSD4A	0x0080	Specifies video window 4's position control register.	0x0000_0000
VIDOSD4B	0x0084	Specifies video window 4's position control register.	0x0000_0000
VIDOSD4C	0x0088	Specifies video window 4's alpha control register.	0x0000_0000

Register	Offset	Description	Reset Value
VIDW00ADD0B0	0x00A0	Specifies window 0's buffer start address register, buffer 0.	0x0000_0000
VIDW00ADD0B1	0x00A4	Specifies window 0's buffer start address register, buffer 1.	0x0000_0000
VIDW00ADD0B2	0x20A0	Specifies window 0's buffer start address register, buffer 2.	0x0000_0000
VIDW01ADD0B0	0x00A8	Specifies window 1's buffer start address register, buffer 0.	0x0000_0000
VIDW01ADD0B1	0x00AC	Specifies Window 1's buffer start address register, buffer 1.	0x0000_0000
VIDW01ADD0B2	0x20A8	Specifies Window 1's buffer start address register, buffer 2.	0x0000_0000
VIDW02ADD0B0	0x00B0	Specifies Window 2's buffer start address register, buffer 0.	0x0000_0000
VIDW02ADD0B1	0x00B4	Specifies Window 2's buffer start address register, buffer 1.	0x0000_0000
VIDW02ADD0B2	0x20B0	Specifies window 2's buffer start address register, buffer 2.	0x0000_0000
VIDW03ADD0B0	0x00B8	Specifies window 3's buffer start address register, buffer 0.	0x0000_0000
VIDW03ADD0B1	0x00BC	Specifies window 3's buffer start address register, buffer 1.	0x0000_0000
VIDW03ADD0B2	0x20B8	Specifies window 3's buffer start address register, buffer 2.	0x0000_0000
VIDW04ADD0B0	0x00C0	Specifies window 4's buffer start address register, buffer 0.	0x0000_0000
VIDW04ADD0B1	0x00C4	Specifies window 4's buffer start address register, buffer 1.	0x0000_0000
VIDW04ADD0B2	0x20C0	Specifies window 4's buffer start address register, buffer 2.	0x0000_0000
VIDW00ADD1B0	0x00D0	Specifies window 0's buffer end address register, buffer 0.	0x0000_0000
VIDW00ADD1B1	0x00D4	Specifies window 0's buffer end address register, buffer 1.	0x0000_0000
VIDW00ADD1B2	0x20D0	Specifies window 0's buffer end address register, buffer 2.	0x0000_0000
VIDW01ADD1B0	0x00D8	Specifies window 1's buffer end address register, buffer 0.	0x0000_0000
VIDW01ADD1B1	0x00DC	Specifies window 1's buffer end address register, buffer 1.	0x0000_0000
VIDW01ADD1B2	0x20D8	Specifies window 1's buffer end address register, buffer 2.	0x0000_0000
VIDW02ADD1B0	0x00E0	Specifies window 2's buffer end address register, buffer 0.	0x0000_0000
VIDW02ADD1B1	0x00E4	Specifies window 2's buffer end address register, buffer 1.	0x0000_0000
VIDW02ADD1B2	0x20E0	Specifies window 2's buffer end address register, buffer 2.	0x0000_0000
VIDW03ADD1B0	0x00E8	Specifies window 3's buffer end address register, buffer 0.	0x0000_0000
VIDW03ADD1B1	0x00EC	Specifies window 3's buffer end address register, buffer 1.	0x0000_0000
VIDW03ADD1B2	0x20E8	Specifies window 3's buffer end address register, buffer 2.	0x0000_0000
VIDW04ADD1B0	0x00F0	Specifies window 4's buffer end address register, buffer 0.	0x0000_0000
VIDW04ADD1B1	0x00F4	Specifies window 4's buffer end address register, buffer 1.	0x0000_0000
VIDW04ADD1B2	0x20F0	Specifies window 4's buffer end address register, buffer 2.	0x0000_0000
VIDW00ADD2	0x0100	Specifies window 0's buffer size register.	0x0000_0000
VIDW01ADD2	0x0104	Specifies window 1's buffer size register.	0x0000_0000
VIDW02ADD2	0x0108	Specifies window 2's buffer size register.	0x0000_0000
VIDW03ADD2	0x010C	Specifies window 3's buffer size register.	0x0000_0000

Register	Offset	Description	Reset Value
VIDW04ADD2	0x0110	Specifies window 4's buffer size register.	0x0000_0000
VIDINTCON0	0x0130	Specifies video interrupt control register.	0x0000_0000
VIDINTCON1	0x0134	Specifies video interrupt pending register.	0x0000_0000
W1KEYCON0	0x0140	Specifies color key control register.	0x0000_0000
W1KEYCON1	0x0144	Specifies color key value (transparent value) register.	0x0000_0000
W2KEYCON0	0x0148	Specifies color key control register.	0x0000_0000
W2KEYCON1	0x014C	Specifies color key value (transparent value) register.	0x0000_0000
W3KEYCON0	0x0150	Specifies color key control register.	0x0000_0000
W3KEYCON1	0x0154	Specifies color key value (transparent value) register.	0x0000_0000
W4KEYCON0	0x0158	Specifies color key control register.	0x0000_0000
W4KEYCON1	0x015C	Specifies color key value (transparent value) register.	0x0000_0000
W1KEYALPHA	0x0160	Specifies color key alpha value register.	0x0000_0000
W2KEYALPHA	0x0164	Specifies color key alpha value register.	0x0000_0000
W3KEYALPHA	0x0168	Specifies color key alpha value register.	0x0000_0000
W4KEYALPHA	0x016C	Specifies color key alpha value register.	0x0000_0000
DITHMODE	0x0170	Specifies dithering mode register.	0x0000_0000
WIN0MAP	0x0180	Specifies window 0's color control.	0x0000_0000
WIN1MAP	0x0184	Specifies window 1's color control.	0x0000_0000
WIN2MAP	0x0188	Specifies window 2's color control.	0x0000_0000
WIN3MAP	0x018C	Specifies window 3's color control.	0x0000_0000
WIN4MAP	0x0190	Specifies window 4's color control.	0x0000_0000
WPALCON_H	0x019C	Specifies window palette control register.	0x0000_0000
WPALCON_L	0x01A0	Specifies window palette control register.	0x0000_0000
TRIGCON	0x01A4	Specifies i80/ RGB trigger control register.	0x0000_0000
I80IFCONA0	0x01B0	Specifies i80 interface control 0 for main LDI.	0x0000_0000
I80IFCONA1	0x01B4	Specifies i80 interface control 0 for sub LDI.	0x0000_0000
I80IFCONB0	0x01B8	Specifies i80 interface control 1 for main LDI.	0x0000_0000
I80IFCONB1	0x01BC	Specifies i80 interface control 1 for sub LDI.	0x0000_0000
COLORGAINCON	0x01C0	Specifies color gain control register.	0x1004_0100
LDI_CMDCON0	0x01D0	Specifies i80 interface LDI command control 0.	0x0000_0000
LDI_CMDCON1	0x01D4	Specifies i80 interface LDI command control 1.	0x0000_0000
SIFCCON0	0x01E0	Specifies LCD i80 system interface command control 0.	0x0000_0000
SIFCCON1	0x01E4	Specifies LCD i80 system interface command control 1.	0x0000_0000
SIFCCON2	0x01E8	Specifies LCD i80 system interface command control 2.	0x????_????
HUECOEF_CR_1	0x01EC	Specifies Hue coefficient control register.	0x0100_0100
HUECOEF_CR_2	0x01F0	Specifies Hue coefficient control register.	0x0000_0000
HUECOEF_CR_3	0x01F4	Specifies Hue coefficient control register.	0x0000_0000

Register	Offset	Description	Reset Value
HUECOEF_CR_4	0x01F8	Specifies Hue coefficient control register.	0x0100_0100
HUECOEF_CB_1	0x01FC	Specifies Hue coefficient control register.	0x0100_0100
HUECOEF_CB_2	0x0200	Specifies Hue coefficient control register.	0x0000_0000
HUECOEF_CB_3	0x0204	Specifies Hue coefficient control register.	0x0000_0000
HUECOEF_CB_4	0x0208	Specifies Hue coefficient control register.	0x0100_0100
HUEOFFSET	0x020C	Specifies Hue offset control register.	0x0180_0080
VIDW0ALPHA0	0x021C	Specifies window 0's alpha value 0 register.	0x0000_0000
VIDW0ALPHA1	0x0220	Specifies window 0's alpha value 1 register.	0x0000_0000
VIDW1ALPHA0	0x0224	Specifies window 1's alpha value 0 register.	0x0000_0000
VIDW1ALPHA1	0x0228	Specifies window 1's alpha value 1 register.	0x0000_0000
VIDW2ALPHA0	0x022C	Specifies window 2's alpha value 0 register.	0x0000_0000
VIDW2ALPHA1	0x0230	Specifies window 2's alpha value 1 register.	0x0000_0000
VIDW3ALPHA0	0x0234	Specifies window 3's alpha value 0 register.	0x0000_0000
VIDW3ALPHA1	0x0238	Specifies window 3's alpha value 1 register.	0x0000_0000
VIDW4ALPHA0	0x023C	Specifies window 4's alpha value 0 register.	0x0000_0000
VIDW4ALPHA1	0x0240	Specifies window 4's alpha value 1 register.	0x0000_0000
BLENDEQ1	0x0244	Specifies window 1's blending equation control register.	0x0000_00c2
BLENDEQ2	0x0248	Specifies window 2's blending equation control register.	0x0000_00c2
BLENDEQ3	0x024C	Specifies window 3's blending equation control register.	0x0000_00c2
BLENDEQ4	0x0250	Specifies window 4's blending equation control register.	0x0000_00c2
BLENDCON	0x0260	Specifies blending control register.	0x0000_0000
W0RTQOSCON	0x0264	Specifies window 0's RTQOS control register.	0x0000_0000
W1RTQOSCON	0x0268	Specifies window 1's RTQOS control register.	0x0000_0000
W2RTQOSCON	0x026C	Specifies window 2's RTQOS control register.	0x0000_0000
W3RTQOSCON	0x0270	Specifies window 3's RTQOS control register.	0x0000_0000
W4RTQOSCON	0x0274	Specifies window 4's RTQOS control register.	0x0000_0000
LDI_CMD0	0x0280	Specifies i80 interface LDI command 0.	0x0000_0000
LDI_CMD1	0x0284	Specifies i80 interface LDI command 1.	0x0000_0000
LDI_CMD2	0x0288	Specifies i80 interface LDI command 2.	0x0000_0000
LDI_CMD3	0x028C	Specifies i80 interface LDI command 3.	0x0000_0000
LDI_CMD4	0x0290	Specifies i80 interface LDI command 4.	0x0000_0000
LDI_CMD5	0x0294	Specifies i80 interface LDI command 5.	0x0000_0000
LDI_CMD6	0x0298	Specifies i80 interface LDI command 6.	0x0000_0000
LDI_CMD7	0x029C	Specifies i80 interface LDI command 7.	0x0000_0000
LDI_CMD8	0x02A0	Specifies i80 interface LDI command 8.	0x0000_0000
LDI_CMD9	0x02A4	Specifies i80 interface LDI command 9.	0x0000_0000
LDI_CMD10	0x02A8	Specifies i80 interface LDI command 10.	0x0000_0000

Register	Offset	Description	Reset Value
LDI_CMD11	0x02AC	Specifies i80 interface LDI command 11.	0x0000_0000
Gamma LUT Data for 64 step mode			
GAMMALUT_01_00	0x037C	Specifies Gamma LUT data of the index 0, 1.	0x0010_0000
GAMMALUT_03_02	0x0380	Specifies Gamma LUT data of the index 2, 3.	0x0030_0020
GAMMALUT_05_04	0x0384	Specifies Gamma LUT data of the index 4, 5.	0x0050_0040
GAMMALUT_07_06	0x0388	Specifies Gamma LUT data of the index 6, 7.	0x0070_0060
GAMMALUT_09_08	0x038C	Specifies Gamma LUT data of the index 8, 9.	0x0090_0080
GAMMALUT_11_10	0x0390	Specifies Gamma LUT data of the index 10, 11.	0x00B0_00A0
GAMMALUT_13_12	0x0394	Specifies Gamma LUT data of the index 12, 13.	0x00D0_00C0
GAMMALUT_15_14	0x0398	Specifies Gamma LUT data of the index 14, 15.	0x00F0_00E0
GAMMALUT_17_16	0x039C	Specifies Gamma LUT data of the index 16, 17.	0x0110_0100
GAMMALUT_19_18	0x03A0	Specifies Gamma LUT data of the index 18, 19.	0x0130_0120
GAMMALUT_21_20	0x03A4	Specifies Gamma LUT data of the index 20, 21.	0x0150_0140
GAMMALUT_23_22	0x03A8	Specifies Gamma LUT data of the index 22, 23.	0x0170_0160
GAMMALUT_25_24	0x03AC	Specifies Gamma LUT data of the index 24, 25.	0x0190_0180
GAMMALUT_27_26	0x03B0	Specifies Gamma LUT data of the index 26, 27.	0x01B0_01A0
GAMMALUT_29_28	0x03B4	Specifies Gamma LUT data of the index 28, 29.	0x01F0_01C0
GAMMALUT_31_30	0x03B8	Specifies Gamma LUT data of the index 30, 31.	0x01F0_01E0
GAMMALUT_33_32	0x03BC	Specifies Gamma LUT data of the index 32, 33.	0x0210_0200
GAMMALUT_35_34	0x03C0	Specifies Gamma LUT data of the index 34, 35.	0x0230_0220
GAMMALUT_37_36	0x03C4	Specifies Gamma LUT data of the index 36, 37.	0x0250_0240
GAMMALUT_39_38	0x03C8	Specifies Gamma LUT data of the index 38, 39.	0x0270_0260
GAMMALUT_41_40	0x03CC	Specifies Gamma LUT data of the index 40, 41.	0x0290_0280
GAMMALUT_43_42	0x03D0	Specifies Gamma LUT data of the index 42, 43.	0x02B0_02A0
GAMMALUT_45_44	0x03D4	Specifies Gamma LUT data of the index 44, 45.	0x02D0_02C0
GAMMALUT_47_46	0x03D8	Specifies Gamma LUT data of the index 46, 47.	0x02F0_02E0
GAMMALUT_49_48	0x03DC	Specifies Gamma LUT data of the index 48, 49.	0x0310_0300
GAMMALUT_51_50	0x03E0	Specifies Gamma LUT data of the index 50, 51.	0x0330_0320
GAMMALUT_53_52	0x03E4	Specifies Gamma LUT data of the index 52, 53.	0x0350_0340
GAMMALUT_55_54	0x03E8	Specifies Gamma LUT data of the index 54, 55.	0x0370_0360
GAMMALUT_57_56	0x03EC	Specifies Gamma LUT data of the index 56, 57.	0x0390_0380
GAMMALUT_59_58	0x03F0	Specifies Gamma LUT data of the index 58, 59.	0x03B0_03A0
GAMMALUT_61_60	0x03F4	Specifies Gamma LUT data of the index 60, 61.	0x03D0_03C0
GAMMALUT_63_62	0x03F8	Specifies Gamma LUT data of the index 62, 63.	0x03F0_03E0
GAMMALUT_xx_64	0x03FC	Specifies Gamma LUT data of the index 64.	0x0000_0400
Gamma LUT Data for 16 step mode			
GAMMALUT_R_1_0	0x037C	Specifies the Gamma RED LUT data of the index 0, 1.	0X0010_0000

Register	Offset	Description	Reset Value
GAMMALUT_R_3_2	0x0380	Specifies the Gamma RED data of the index 2, 3.	0X0030_0020
GAMMALUT_R_5_4	0x0384	Specifies the Gamma RED data of the index 4, 5.	0X0050_0040
GAMMALUT_R_7_6	0x0388	Specifies the Gamma RED data of the index 6, 7.	0X0070_0060
GAMMALUT_R_9_8	0x038C	Specifies the Gamma RED data of the index 8, 9.	0X0090_0080
GAMMALUT_R_11_10	0x0390	Specifies the Gamma RED data of the index 10, 11.	0X00B0_00A0
GAMMALUT_R_13_12	0x0394	Specifies the Gamma RED data of the index 12, 13.	0X00D0_00C0
GAMMALUT_R_15_14	0x0398	Specifies the Gamma RED data of the index 14, 15.	0X00F0_00E0
GAMMALUT_R_16	0x039C	Specifies the Gamma RED data of the index 16.	0X0110_0100
GAMMALUT_R_1_0	0x03A0	Specifies the Gamma GREEN LUT data of the index 0, 1.	0X0130_0120
GAMMALUT_R_3_2	0x03A4	Specifies the Gamma GREEN data of the index 2, 3.	0X0150_0140
GAMMALUT_R_5_4	0x03A8	Specifies the Gamma GREEN data of the index 4, 5.	0X0170_0160
GAMMALUT_R_7_6	0x03AC	Specifies the Gamma GREEN data of the index 6, 7.	0X0190_0180
GAMMALUT_R_9_8	0x03B0	Specifies the Gamma GREEN data of the index 8, 9.	0X01B0_01A0
GAMMALUT_R_11_10	0x03B4	Specifies the Gamma GREEN data of the index 10, 11.	0X01D0_01C0
GAMMALUT_R_13_12	0x03B8	Specifies the Gamma GREEN data of the index 12, 13.	0X01F0_01E0
GAMMALUT_R_15_14	0x03BC	Specifies the Gamma GREEN data of the index 14, 15.	0X0210_0200
GAMMALUT_R_16	0x03C0	Specifies the Gamma GREEN data of the index 16.	0X0230_0220
GAMMALUT_R_1_0	0x03C4	Specifies the Gamma BLUE data of the index 0, 1.	0X0250_0240
GAMMALUT_R_3_2	0x03C8	Specifies the Gamma BLUE data of the index 2, 3.	0X0270_0260
GAMMALUT_R_5_4	0x03CC	Specifies the Gamma BLUE data of the index 4, 5.	0X0290_0280
GAMMALUT_R_7_6	0x03D0	Specifies the Gamma BLUE data of the index 6, 7.	0X02B0_02A0
GAMMALUT_R_9_8	0x03D4	Specifies the Gamma BLUE data of the index 8, 9.	0X02D0_02C0
GAMMALUT_R_11_10	0x03D8	Specifies the Gamma BLUE data of the index 10, 11.	0X02F0_02E0
GAMMALUT_R_13_12	0x03DC	Specifies the Gamma BLUE data of the index 12, 13.	0X0310_0300
GAMMALUT_R_15_14	0x03E0	Specifies the Gamma BLUE data of the index 14, 15.	0X0330_0320
GAMMALUT_R_16	0x03E4	Specifies the Gamma BLUE data of the index 16	0X0350_0340
RSVD	0x03E8	Not used	0X0370_0360
RSVD	0x03EC	Not used	0X0390_0380
RSVD	0x03F0	Not used	0X03B0_03A0

Register	Offset	Description	Reset Value
RSVD	0x03F4	Not used	0X03D0_03C0
RSVD	0x03F8	Not used	0X03F0_03E0
RSVD	0x03FC	Not used	0X0000_0400
Shadow Windows Control			
SHD_VIDW00ADD0	0x40A0	Specifies window 0's buffer start address register (shadow).	0x0000_0000
SHD_VIDW01ADD0	0x40A8	Specifies window 1's buffer start address register (shadow).	0x0000_0000
SHD_VIDW02ADD0	0x40B0	Specifies window 2's buffer start address register (shadow).	0x0000_0000
SHD_VIDW03ADD0	0x40B8	Specifies window 3's buffer start address register (shadow).	0x0000_0000
SHD_VIDW04ADD0	0x40C0	Specifies window 4's buffer start address register (shadow).	0x0000_0000
SHD_VIDW00ADD1	0x40D0	Specifies window 0's buffer end address register (shadow)	0x0000_0000
SHD_VIDW01ADD1	0x40D8	Specifies window 1's buffer end address register (shadow)	0x0000_0000
SHD_VIDW02ADD1	0x40E0	Specifies window 2's buffer end address register (shadow).	0x0000_0000
SHD_VIDW03ADD1	0x40E8	Specifies window 3's buffer end address register (shadow).	0x0000_0000
SHD_VIDW04ADD1	0x40F0	Specifies window 4's buffer end address register (shadow).	0x0000_0000
SHD_VIDW00ADD2	0x4100	Specifies window 0's buffer size register (shadow).	0x0000_0000
SHD_VIDW01ADD2	0x4104	Specifies window 1's buffer size register (shadow).	0x0000_0000
SHD_VIDW02ADD2	0x4108	Specifies window 2's buffer size register (shadow).	0x0000_0000
SHD_VIDW03ADD2	0x410C	Specifies window 3's buffer size register (shadow).	0x0000_0000
SHD_VIDW04ADD2	0x4110	Specifies window 4's buffer size register (shadow).	0x0000_0000

16.5.3 Palette Memory

- Base Address: 0x11C0_0000, 0x1200_0000

	Start Address	End Address	Description	Reset Value
Win0 PalRam	0x2400 (0x0400)	0x27FC (0x07FC)	Specifies 0 to 255 entry palette data.	Undefined
Win1 PalRam	0x2800 (0x0800)	0x2BFC (0x0BFC)	Specifies 0 to 255 entry palette data.	Undefined
Win2 PalRam	0x2C00	0x2FFC	Specifies 0 to 255 entry palette data.	Undefined
Win3 PalRam	0x3000	0x33FC	Specifies 0 to 255 entry palette data.	Undefined
Win4 PalRam	0x3400	0x37FC	Specifies 0 to 255 entry palette data.	Undefined

16.5.4 Control Registers

16.5.4.1 VIDCON0

- Address = Base Address + 0x0000, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31]	-	Reserved (should be 0).	0
DSI_EN	[30]	RW	Enables MIPI DSI. 0 = Disables 1 = Enables (i80 24bit data interface, SYS_ADD[1])	0
RSVD	[29]	-	Reserved (should be 0)	0
VIDOUT	[28:26]	RW	Determines the output format of Video Controller. 000 = RGB interface 001 = Reserved 010 = Indirect I80 interface for LDI0 011 = Indirect I80 interface for LDI1 100 = WB interface and RGB interface 101 = Reserved 110 = WB Interface and i80 interface for LDI0 111 = WB Interface and i80 interface for LDI1	000
L1_DATA16	[25:23]	RW	Selects output data format mode of indirect i80 interface (LDI1). (VIDOUT[1:0] == 2'b11) 000 = 16-bit mode (16 bpp) 001 = 16 + 2-bit mode (18 bpp) 010 = 9 + 9-bit mode (18 bpp) 011 = 16 + 8-bit mode (24 bpp) 100 = 18-bit mode (18bpp) 101 = 8 + 8-bit mode (16bpp)	000
L0_DATA16	[22:20]	RW	Selects output data format mode of indirect i80 interface (LDI0). (VIDOUT[1:0] == 2'b10) 000 = 16-bit mode (16 bpp) 001 = 16 + 2-bit mode (18 bpp) 010 = 9 + 9-bit mode (18 bpp) 011 = 16 + 8-bit mode (24 bpp) 100 = 18-bit mode (18bpp) 101 = 8 + 8-bit mode (16bpp)	000
RSVD	[19]	-	Reserved (should be 0).	0
RGSPSEL	[18]	RW	Selects display mode (VIDOUT[1:0] == 2'b00). 0 = RGB parallel format 1 = RGB serial format Selects the display mode (VIDOUT[1:0] != 2'b00). 0 = RGB parallel format	0
PNRMODE	[17]	RW	Controls inverting RGB_ORDER (@VIDCON3). 0 = Normal: RGBORDER[2] @VIDCON3 1 = Invert: ~RGBORDER[2] @VIDCON3	00

Name	Bit	Type	Description	Reset Value
			NOTE: This bit is used for the previous version of FIMD. You do not have to use this bit if you use RGB_ORDER@VIDCON3 register.	
CLKVALUP	[16]	RW	Selects CLKVAL_F update timing control. 0 = Always 1 = Start of a frame (only once per frame)	0
RSVD	[15:14]	-	Reserved.	0
CLKVAL_F	[13:6]	RW	Determines the rates of VCLK and CLKVAL[7:0]. VCLK = FIMDxSCLK/(CLKVAL+1), where CLKVAL >= 1 NOTE: The maximum frequency of VCLK is 80 MHz. (80 MHz for Display Controller 0, 50 MHz for Display Controller 1)	0
VCLKFREE	[5]	RW	Controls VCLK Free Run (Only valid at RGB IF mode). 0 = Normal mode (controls using ENVID) 1 = Free-run mode	0
RSVD	[4:2]	-	Should be 0.	0x0
ENVID	[1]	RW	Enables/disables video output and logic immediately. 0 = Disables the video output and display control signal. 1 = Enables the video output and display control signal.	0
ENVID_F	[0]	RW	Enables/disables video output and logic at current frame end. 0 = Disables the video output and display control signal. 1 = Enables the video output and display control signal. If this bit is set to "on" and "off", then "H" is read and video controller is enabled until the end of current frame. (NOTE)	0

NOTE: Display On: ENVID and ENVID_F are set to "1".

Direct Off: ENVID and ENVID_F are set to "0" simultaneously.

Per Frame Off: ENVID_F is set to "0" and ENVID is set to "1".

Caution: 1. If VIDCON0 is set for Per Frame Off in interlace mode, the value of INTERLACE_F should be set to "0" in the same time.

2. If display controller is off using direct off, it is impossible to turn on the display controller without reset.

16.5.4.2 VIDCON1

- Address = Base Address + 0x0004, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
LINECNT (read only)	[26:16]	RW	Provides the status of the line counter (read only). Up count from 0 to LINEVAL.	0
FSTATUS	[15]	RW	Specifies the Field Status (read only). 0 = ODD Field 1 = EVEN Field	0
VSTATUS	[14:13]	RW	Specifies the Vertical Status (read only). 00 = VSYNC 01 = BACK Porch 10 = ACTIVE 11 = FRONT Porch	0
RSVD	[12:11]	-	Reserved	0
FIXVCLK	[10:9]	RW	Specifies the VCLK hold scheme at data under-flow. 00 = VCLK hold 01 = VCLK running 11 = VCLK running and VDEN disable	0
RSVD	[8]	-	Reserved	0
IVCLK	[7]	RW	Controls the polarity of the VCLK active edge. 0 = Video data is fetched at VCLK falling edge 1 = Video data is fetched at VCLK rising edge	0
IHSYNC	[6]	RW	Specifies the HSYNC pulse polarity. 0 = Normal 1 = Inverted	0
IVSYNC	[5]	RW	Specifies the VSYNC pulse polarity. 0 = Normal 1 = Inverted	0
IVDEN	[4]	RW	Specifies the VDEN signal polarity. 0 = Normal 1 = Inverted	0
RSVD	[3:0]	RW	Reserved	0x0

16.5.4.3 VIDCON2

- Address = Base Address + 0x0008, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:28]	-	Reserved	0
RGB_SKIP_EN	[27]	RW	Enables the RGB skip mode. (only where RGBPSEL == 1'b0). 0 = Disables 1 = Enables	0
RSVD	[26]	-	Reserved	0
RGB_DUMMY_LOC	[25]	RW	Controls RGB dummy insertion location (only where RGBPSEL == 1'b1 and RGB_SKIP_EN == 1'b1) 0 = Last (4th) position 1 = 1st position	0
RGB_DUMMY_EN	[24]	RW	Enables RGB dummy insertion mode (only where RGBPSEL == 1'b1) 0 = Disables 1 = Enables	0
RSVD	[23:22]	-	Reserved (should be 0)	0
RGB_ORDER_E	[21:19]	RW	Controls RGB interface output order. (Even line, line # 2, 4, 6, 8.) where, RGBPSEL== 1'b0 000 = RGB 001 = GBR 010 = BRG 100 = BGR 101 = RBG 110 = GRB where, (RGBPSEL == 1'b1) or (RGBPSEL == 1'b0 and RGB_SKIP_EN = 1'b1) 000 = R→G→B 001 = G→B→R 010 = B→R→G 100 = B→G→R 101 = R→B→G 110 = G→R→B Note: PNR0[0] @VIDCON0 should be 0, when you use RGB_ORDER_O[2:0] @VIDCON3 register.	0
RGB_ORDER_O	[18:16]	RW	Controls RGB interface output order (Odd Line, line #1, 3, 5, 7.) where, RGBPSEL== 1'b0 000 = RGB 001 = GBR 010 = BRG 100 = BGR 101 = RBG	0

Name	Bit	Type	Description	Reset Value
			<p>110 = GRB where, (RGBSPSEL == 1'b1) or (RGBSPSEL == 1'b0 and RGB_SKIP_EN = 1'b1)</p> <p>000 = R→G→B 001 = G→B→R 010 = B→R→G 100 = B→G→R 101 = R→B→G 110 = G→R→B</p> <p>NOTE: PNR0[0] @VIDCON0 should be 0, when you use RGB_ORDER_E[2:0]@VIDCON3 register.</p>	
RSVD	[15:14]	–	<p>Reserved. NOTE: This bit should be 1.</p>	0
TVFORMATSEL	[13:12]	RW	<p>Specifies the output format of YUV data. 00 = Reserved 01 = YUV422 1x = YUV444</p>	0
RSVD	[11:9]	–	Reserved	0
OrgYCbCr	[8]	RW	<p>Specifies the order of YUV data. 0 = Y - CbCr 1 = CbCr - Y</p>	0
YUVOrd	[7]	RW	<p>Specifies the order of Chroma data. 0 = Cb - Cr 1 = Cr - Cb</p>	0
RSVD	[6:5]	–	Reserved	0
WB_FRAME_SKIP	[4:0]	RW	<p>Controls the WB frame skip rate. The maximum rate is up to 1:30 [only where (VIDOUT[2:0] == 3'b001 or 3'b100 TV encoder interface), (INTERLACE_F==1'b0) and (TV422 or TVRGB output)].</p> <p>00000 = No skip (1: 1) 00001 = Skip rate (1: 2) 00010 = Skip rate (1: 3) ... 11101 = Skip rate (1: 30) 1111x = Reserved</p>	0

16.5.4.4 VIDCON3

- Address = Base Address + 0x000C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:21]	–	Reserved (should be 0)	0
RSVD	[20:19]	–	Reserved	0
CG_ON	[18]	RW	Enables Control Color Gain. 0 = Disables (bypass) 1 = Enables	0
RSVD	[17]	–	Reserved	0
GM_ON	[16]	RW	Enables Control Gamma. 0 = Disables (bypass) 1 = Enables	0
GM_MODE	[15]	RW	Gamma mode selection 0 = Apply 64 step identical value to all R, G, B data 1 = Apply 16 step independent value to each R, G, B data	0
HUE_CSC_F_Narrow	[14]	RW	Controls HUE CSC_F Narrow/ Wide. 0 = Wide 1 = Narrow	0
HUE_CSC_F_EQ709	[13]	RW	Controls HUE_CSC_F parameter. 0 = Eq. 601 1 = Eq.709	0
HUE_CSC_F_ON	[12]	RW	Enables HUE_CSC_F. 0 = Disables 1 = Enables (when HUE_ON == 1'b1)	0
RSVD	[11]	–	Reserved	0
HUE_CSC_B_Narrow	[10]	RW	Controls HUE CSC_B Narrow/ Wide. 0 = Wide 1 = Narrow	0
HUE_CSC_B_EQ709	[9]	RW	Controls HUE_CSC_B parameter. 0 = Eq. 601 1 = Eq.709	0
HUE_CSC_B_ON	[8]	RW	Enables HUE_CSC_B. 0 = Disables 1 = Enables (when HUE_ON == 1'b1)	0
HUE_ON	[7]	RW	Enables Control Hue. 0 = Disables (bypass) 1 = Enables	0
RSVD	[6:2]	–	Reserved (should be 0)	0
PC_DIR	[1]	RW	Controls Pixel Compensation direction. 0 = + 0.5 (pos.) 1 = - 0.5 (neg.)	0
PC_ON	[3:0]	RW	Enables Pixel Compensation.	0x0

Name	Bit	Type	Description	Reset Value
			0 = Disable 1 = Enable NOTE: TV output data is compensated by PC_ON == 1'b1.	

16.5.4.5 VIDTCON0

- Address = Base Address + 0x0010, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
VBPDE	[31:24]	RW	Vertical back porch specifies the number of inactive lines at the start of a frame after vertical synchronization period. (Only for even field of YVU interface)	0x00
VBPD	[23:16]	RW	Vertical back porch specifies the number of inactive lines at the start of a frame after vertical synchronization period.	0x00
VFPD	[15:8]	RW	Vertical front porch specifies the number of inactive lines at the end of a frame before vertical synchronization period.	0x00
VSPW	[7:0]	RW	Vertical sync pulse width determines the high-level width of VSYNC pulse by counting the number of inactive lines.	0x00

16.5.4.6 VIDTCON1

- Address = Base Address + 0x0014, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
VFPDE	[31:24]	RW	Vertical front porch specifies the number of inactive lines at the end of a frame before vertical synchronization period. (Only for the even field of YVU interface).	0
HBDP	[23:16]	RW	Horizontal back porch specifies the number of VCLK periods between the falling edge of HSYNC and start of active data.	0x00
HFPD	[15:8]	RW	Horizontal front porch specifies the number of VCLK periods between the end of active data and rising edge of HSYNC.	0x00
HSPW	[7:0]	RW	Horizontal sync pulse width determines the high-level width of HSYNC pulse by counting the number of VCLK.	0x00

16.5.4.7 VIDTCON2

- Address = Base Address + 0x0018, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
LINEVAL	[21:11]	RW	Determines the vertical size of display. In the Interlace mode, (LINEVAL + 1) should be even.	0
HOZVAL	[10:0]	RW	Determines the horizontal size of display.	0

NOTE: HOZVAL = (Horizontal display size) – 1 and LINEVAL = (Vertical display size) – 1.

16.5.4.8 VIDTCON3

- Address = Base Address + 0x001C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
VSYNCEN	[31]	RW	Enables VSYNC Signal Output. 0 = Disables 1 = Enables VBPD(VFPD, VSPW) + 1 < LINEVAL (when VSYNCEN =1)	0
RSVD	[30]	–	Reserved (should be 0).	0
FRMEN	[29]	RW	Enables the FRM signal output. 0 = Disables 1 = Enables	0
INVFRM	[28]	RW	Controls the polarity of FRM pulse. 0 = Active HIGH 1 = Active LOW	0
FRMVFRATE	[27:24]	RW	Controls the FRM issue rate (Maximum rate up to 1:16)	0x00
RSVD	[23:16]	RW	Reserved	0x00
FRMVFPD	[15:8]	RW	Specifies the number of line between data active and FRM signal.	0x00
FRMVSPW	[7:0]	RW	Specifies the number of line of FRM signal width. (FRMVFPD + 1) + (FRMVSPW + 1) < LINEVAL + 1 (in RGB) ??	0x00

16.5.4.9 WINCON0

- Address = Base Address + 0x0020, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
BUFSTATUS_H	[31]	RW	Specifies the Buffer Status (read only). NOTE: BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L} 00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2	0
BUFSEL_H	[30]	RW	Selects the Buffer set. NOTE: BUFSEL = {BUFSEL_H, BUFSEL_L} 00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2 (only available where BUF_MODE == 1'b1)	0
LIMIT_ON	[29]	RW	Enables CSC source limiter (for clamping xvYCC source). 0 = Disables 1 = Enables (when local SRC data has xvYCC color space, InRGB=1)	0
EQ709	[28]	RW	Controls CSC parameter. 0 = Eq. 601 1 = Eq. 709 (when local SRC data has HD (709) color gamut)	0
nWide/Narrow	[27:26]	RW	Chooses the color space conversion equation from YCbCr to RGB according to input value range (2'00 for YCbCr Wide range and 2'11 for YCbCr Narrow range) <ul style="list-style-type: none"> Wide Range: Y/Cb/Cr: 255-0 Narrow Range: Y: 235-16, Cb/Cr: 240-16 	00
TRGSTATUS	[25]	RW	Specifies the Trigger Status (read only). 0 = No trigger is issued 1 = Trigger is issued	0
RSVD	[24:23]	-	Reserved	00
ENLOCAL_F	[22]	RW	Selects the Data access method. 0 = Dedicated DMA 1 = Local Path	0
BUFSTATUS_L	[21]	RW	Specifies the Buffer Status (read only). NOTE: BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L}	0
BUFSEL_L	[20]	RW	Selects the Buffer set. NOTE: BUFSEL = {BUFSEL_H, BUFSEL_L}	0
BUFAUTOEN	[19]	RW	Specifies the Double Buffer Auto control bit. 0 = Fixed by BUFSEL 1 = Auto Changed by Trigger Input	0
BITSWP_F	[18]	RW	Specifies the Bit swap control bit. 0 = Swap Disable 1 = Swap Enable	0

Name	Bit	Type	Description	Reset Value
			NOTE: It should be 0 when ENLOCAL is 1.	
BYTSPW_F	[17]	RW	Specifies the Byte swaps control bit. 0 = Swap Disable 1 = Swap Enable NOTE: It should be 0 when ENLOCAL is 1.	0
HAWSPW_F	[16]	RW	Specifies the Half-Word swap control bit. 0 = Swap Disable 1 = Swap Enable NOTE: It should be 0 when ENLOCAL is 1.	0
WSWP_F	[15]	RW	Specifies the Word swap control bit. 0 = Swap Disable 1 = Swap Enable NOTE: It should be 0 when ENLOCAL is 1.	0
BUF_MODE	[14]	RW	Selects the auto-buffering mode. 0 = Double 1 = Triple	0
InRGB	[13]	RW	Specifies the input color space of source image (only for 'ENLOCAL' enable). 0 = RGB 1 = YCbCr	0
RSVD	[12:11]	-	Reserved (Should be 0)	0
BURSTLEN	[10:9]	RW	Selects the DMA Burst Maximum Length. 00 = 16 word-burst 01 = 8 word-burst 10 = 4 word-burst	0
RSVD	[8:7]	-	Reserved (Should be 0)	0
BLD_PIX_F	[6]	RW	Selects the blending category (In case of window0, this is only required for deciding window 0's blending factor.) 0 = Per plane blending 1 = Per pixel blending	0
BPPMODE_F	[5:2]	RW	Selects the Bits Per Pixel (BPP) mode for Window image. 0000 = 1 bpp 0001 = 2 bpp 0010 = 4 bpp 0011 = 8 bpp (palletized) 0100 = 8 bpp (non-palletized, A: 1-R:2-G:3-B:2) 0101 = 16 bpp (non-palletized, R:5-G:6-B:5) 0110 = 16 bpp (non-palletized, A:1-R:5-G:5-B:5) 0111 = 16 bpp (non-palletized, I :1-R:5-G:5-B:5) 1000 = Unpacked 18 bpp (non-palletized, R:6-G:6-B:6) 1001 = Unpacked 18 bpp (non-palletized, A:1-R:6-G:6-B:5) 1010 = Unpacked 19 bpp (non-palletized, A:1-R:6-G:6-B:6) 1011 = Unpacked 24 bpp (non-palletized R:8-G:8-B:8) 1100 = Unpacked 24 bpp	0

Name	Bit	Type	Description	Reset Value
			<p>(non-palletized A:1-R:8-G:8-B:7) 1101 = Unpacked 25 bpp (non-palletized A:1-R:8-G:8-B:8) (1) 1110 = Unpacked 13 bpp (non-palletized A:1-R:4-G:4-B:4) (2) 1111 = Unpacked 15 bpp (non-palletized R:5-G:5-B:5)</p> <p>NOTE</p> <p>1. 1101 = Supports unpacked 32 bpp (non-palletized A:8-R:8-G:8-B:8) for per pixel blending.</p> <p>2. 1110 = Supports 16 bpp (non-palletized A:4-R:4-G:4-B:4) for per pixel blending. (16 level blending)</p>	
ALPHA_SEL_F	[1]	RW	<p>Selects the Alpha value. When per plane blending case (BLD_PIX ==0): 0 = Using ALPHA0_R/G/B values 1 = Using ALPHA1_R/G/B values</p> <p>When per pixel blending (BLD_PIX ==1) 0 = Selected by AEN (A value) 1 = Using DATA[31:24] data in word boundary (only when BPPMODE_F = 4'b1101) DATA[31:28], [15:12] data in word boundary (only when BPPMODE_F = 4'b1110)</p>	0
ENWIN_F	[0]	RW	Enables/disables video output and logic immediately. 0 = Disables the video output and video control signal. 1 = Enables the video output and video control signal.	0

16.5.4.10 WINCON1

- Address = Base Address + 0x0024, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
BUFSTATUS_H	[31]	RW	<p>Specifies the Buffer Status (read only). NOTE: BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L}</p> <p>00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2</p>	0
BUFSEL_H	[30]	RW	<p>Select the Buffer set. NOTE: BUFSEL = {BUFSEL_H, BUFSEL_L}</p> <p>00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2 (only available when BUF_MODE == 1'b1)</p>	0
LIMIT_ON	[29]	RW	<p>Enables Control CSC source limiter (for clamping xvYCC source). 0 = Disables 1 = Enables (when local SRC data has xvYCC color space, InRGB=1)</p>	0
EQ709	[28]	RW	<p>Controls the CSC parameter. 0 = Eq.601 1 = Eq.709 (when local SRC data has HD(709) color gamut)</p>	0
nWide/Narrow	[27:26]	RW	<p>Chooses the color space conversion equation from YCbCr to RGB based on input value range (2'00 for YCbCr Wide range and 2'11 for YCbCr Narrow range).</p> <ul style="list-style-type: none"> Wide Range: Y/Cb/Cr: 255-0 Narrow Range: Y: 235-16, Cb/Cr: 240-16 	00
TRGSTATUS	[25]	RW	<p>Specifies Window 0 Software Trigger Update Status (read only). 0 = Update 1 = Not Update</p> <p>If the Software Trigger in window 1 occurs, this bit is automatically set to '1'. This value is cleared only after updating the shadow register sets.</p>	0
RSVD	[24:23]	-	Reserved. (should be 0)	0
ENLOCAL_F	[22]	RW	<p>Selects the Data access method. 0 = Dedicated DMA 1 = Local Path</p>	0
BUFSTATUS_L	[21]	RW	<p>Specifies the Buffer Status (read only). NOTE: BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L}</p>	0
BUFSEL_L	[20]	RW	<p>Selects the Buffer set. NOTE: BUFSEL = {BUFSEL_H, BUFSEL_L}</p>	0
BUFAUTOEN	[19]	RW	<p>Specifies the Double Buffer Auto control bit. 0 = Fixed by BUFSEL</p>	0

Name	Bit	Type	Description	Reset Value
			1 = Auto changed by Trigger Input	
BITSWP_F	[18]	RW	Specifies the Bit swap control bit. 0 = Swap Disable 1 = Swap Enable NOTE: It should be 0 when ENLOCAL is 1.	0
BYTSPW_F	[17]	RW	Specifies the Byte swaps control bit. 0 = Swap Disable 1 = Swap Enable NOTE: It should be 0 when ENLOCAL is 1.	0
HAWSPW_F	[16]	RW	Specifies the Half-Word swap control bit. 0 = Swap Disable 1 = Swap Enable NOTE: It should be 0 when ENLOCAL is 1.	0
WSWP_F	[15]	RW	Specifies the Word swap control bit. 0 = Swap Disable 1 = Swap Enable NOTE: It should be 0 when ENLOCAL is 1.	0
BUF_MODE	[14]	RW	Selects the auto-buffering mode. 0 = Double 1 = Triple	0
InRGB	[13]	RW	Indicates the input color space of source image (only for "EnLcal" enable). 0 = RGB 1 = YCbCr	0
RSVD	[12:11]	-	Reserved (should be 0)	0
BURSTLEN	[10:9]	RW	Specifies the DMA's Burst Maximum Length selection. 00 = 16 word - burst 01 = 8 word - burst 10 = 4 word - burst	0
RSVD	[8]	-	Reserved (should be 0)	0
ALPHA_MUL_F	[7]	RW	Specifies the Multiplied Alpha value mode. 0 = Disables 1 = Enables multiplied mode When ALPHA_MUL is 1, set BLD_PIX =1, ALPHA_SEL=1, and BPPMODE_F[5:2] = 4'b1101 or 4'b1110. NOTE: Alpha value = alpha_pixel (from data) × ALPHA0_R/G/B	0
BLD_PIX_F	[6]	RW	Selects the blending category. 0 = Per plane blending 1 = Per pixel blending	0
BPPMODE_F	[5:2]	RW	Selects the Bits Per Pixel (BPP) mode in Window image. 0000 = 1 bpp 0001 = 2 bpp	0

Name	Bit	Type	Description	Reset Value
			<p>0010 = 4 bpp 0011 = 8 bpp (palletized) 0100 = 8 bpp (non-palletized, A: 1-R:2-G:3-B:2) 0101 = 16 bpp (non-palletized, R:5-G:6-B:5) 0110 = 16 bpp (non-palletized, A:1-R:5-G:5-B:5) 0111 = 16 bpp (non-palletized, I :1-R:5-G:5-B:5) 1000 = Unpacked 18 bpp (non-palletized, R:6-G:6-B:6) 1001 = Unpacked 18 bpp (non-palletized, A:1-R:6-G:6-B:5) 1010 = Unpacked 19 bpp (non-palletized, A:1-R:6-G:6-B:6) 1011 = Unpacked 24 bpp (non-palletized R:8-G:8-B:8) 1100 = Unpacked 24 bpp (non-palletized A:1-R:8-G:8-B:7) 1101 = Unpacked 25 bpp (non-palletized A:1-R:8-G:8-B:8) ⁽¹⁾ 1110 = Unpacked 13 bpp (non-palletized A:1-R:4-G:4-B:4) ⁽²⁾ 1111 = Unpacked 15 bpp (non-palletized R:5-G:5-B:5)</p> <p>NOTE</p> <p>1. 1101 = supports unpacked 32 bpp (non-palletized A:8-R:8-G:8-B:8) for per pixel blending.</p> <p>2. 1110 = supports 16 bpp (non-palletized A:4-R:4-G:4-B:4) for per pixel blending. (16 level blending)</p>	
ALPHA_SEL_F	[1]	RW	<p>Selects the Alpha value. When Per plane blending case (BLD_PIX == 0) 0 = Using ALPHA0_R/G/B values 1 = Using ALPHA1_R/G/B values</p> <p>When Per pixel blending (BLD_PIX == 1) 0 = Selected by AEN (A value) 1 = Using DATA[31:24] data in word boundary (only when BPPMODE_F = 4'b1101) DATA[31:28], [15:12] data in word boundary (only when BPPMODE_F = 4'b1110)</p>	0
ENWIN_F	[0]	RW	<p>Enables/disables video output and logic immediately. 0 = Disables the video output and video control signal. 1 = Enables the video output and video control signal.</p>	0

16.5.4.11 WINCON2

- Address = Base Address + 0x0028, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
BUFSTATUS_H	[31]	RW	Specifies the Buffer Status (read only). NOTE: BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L} 00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2	0
BUFSEL_H	[30]	RW	Selects the Buffer set. NOTE: BUFSEL = {BUFSEL_H, BUFSEL_L} 00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2 (only available when BUF_MODE == 1'b1)	0
LIMIT_ON	[29]	RW	Enables CSC source limiter (for clamping xvYCC source). 0 = Disables 1 = Enables (when local SRC data has xvYCC color space, InRGB=1)	0
EQ709	[28]	RW	Controls CSC parameter. 0 = Eq.601 1 = Eq.709 (when local SRC data has HD (709) color gamut)	0
nWide/Narrow	[27:26]	RW	Chooses color space conversion equation from YCbCr to RGB based on the input value range (2'00 for YCbCr Wide range and 2'11 for YCbCr Narrow range). <ul style="list-style-type: none"> Wide Range: Y/Cb/Cr: 255-0 Narrow Range: Y: 235-16, Cb/Cr: 240-16 	00
RSVD	[25:24]	-	Reserved	00
LOCALSEL_F	[23]	RW	Selects the local path source. 0 = CAMIF2 1 = CAMIF3	0
ENLOCAL_F	[22]	RW	Selects the Data access method. 0 = Dedicated DMA 1 = Local Path	0
BUFSTATUS_L	[21]	RW	Specifies the Buffer Status (read only). NOTE: BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L}	
BUFSEL_L	[20]	RW	Selects the Buffer set. NOTE: BUFSEL = {BUFSEL_H, BUFSEL_L}	0
BUFAUTOEN	[19]	RW	Specifies the Double Buffer Auto control bit. 0 = Fixed by BUFSEL 1 = Auto changed by Trigger Input	0
BITSWP_F	[18]	RW	Specifies the Bit swap control bit. 0 = Swap Disable 1 = Swap Enable	0

Name	Bit	Type	Description	Reset Value
			NOTE: It should be 0 when ENLOCAL is 1.	
BYTSPW_F	[17]	RW	Specifies the Byte swaps control bit. 0 = Swap Disable 1 = Swap Enable NOTE: It should be 0 when ENLOCAL is 1.	0
HAWSPW_F	[16]	RW	Specifies the Half-Word swap control bit. 0 = Swap Disable 1 = Swap Enable NOTE: It should be 0 when ENLOCAL is 1.	0
WSWP_F	[15]	RW	Specifies the Word swap control bit. 0 = Swap Disable 1 = Swap Enable NOTE: It should be 0 when ENLOCAL is 1.	0
BUF_MODE	[14]	RW	Selects the auto-buffering mode. 0 = Double 1 = Triple	0
InRGB	[13]	RW	Specifies the input color space of source image (only for "EnLcal" enable). 0 = RGB 1 = YCbCr	0
RSVD	[12:11]	-	Reserved (should be 0).	0
BURSTLEN	[10:9]	RW	Selects the DMA's Burst Maximum Length. 00 = 16 word-burst 01 = 8 word-burst 10 = 4 word-burst	0
RSVD	[8]	-	Reserved (should be 0).	0
ALPHA_MUL_F	[7]	RW	Specifies the Multiplied Alpha value mode. 0 = Disables 1 = Enables multiplied mode When ALPHA_MUL is 1, set BLD_PIX =1, ALPHA_SEL=1, and BPPMODE_F[5:2] = 4'b1101 or 4'b1110. NOTE: Alpha value = alpha_pixel (from data) × ALPHA0_R/G/B	0
BLD_PIX_F	[6]	RW	Selects the blending category. 0 = Per plane blending 1 = Per pixel blending	0
BPPMODE_F	[5:2]	RW	Selects the Bits Per Pixel (BPP) mode in Window image. 0000 = 1 bpp 0001 = 2 bpp 0010 = 4 bpp 0011 = 8 bpp (palletized) 0100 = 8 bpp (non-palletized, A: 1-R:2-G:3-B:2) 0101 = 16 bpp (non-palletized, R:5-G:6-B:5) 0110 = 16 bpp (non-palletized, A:1-R:5-G:5-B:5)	0

Name	Bit	Type	Description	Reset Value
			<p>0111 = 16 bpp (non-palletized, I :1-R:5-G:5-B:5) 1000 = Unpacked 18 bpp (non-palletized, R:6-G:6-B:6) 1001 = Unpacked 18 bpp (non-palletized, A:1-R:6-G:6-B:5) 1010 = Unpacked 19 bpp (non-palletized, A:1-R:6-G:6-B:6) 1011 = Unpacked 24 bpp (non-palletized R:8-G:8-B:8) 1100 = Unpacked 24 bpp (non-palletized A:1-R:8-G:8-B:7) 1101 = Unpacked 25 bpp (non-palletized A:1-R:8-G:8-B:8) ⁽¹⁾ 1110 = Unpacked 13 bpp (non-palletized A:1-R:4-G:4-B:4) ⁽²⁾ 1111 = Unpacked 15 bpp (non-palletized R:5-G:5-B:5)</p> <p>NOTE:</p> <p>1. 1101 = Supports unpacked 32 bpp (non-palletized A:8-R:8-G:8-B:8) for per pixel blending.</p> <p>2. 1110 = Supports 16 bpp (non-palletized A:4-R:4-G:4-B:4) for per pixel blending. (16 level blending)</p>	
ALPHA_SEL_F	[1]	RW	<p>Selects the Alpha value. When Per plane blending case BLD_PIX ==0: 0 = Using ALPHA0_R/G/B values 1 = Using ALPHA1_R/G/B values</p> <p>When Per pixel blending BLD_PIX ==1: 0 = Selected by AEN (A value) 1 = Using DATA[31:24] data in word boundary (only when BPPMODE_F = 4'b1101) DATA[31:28], [15:12] data in word boundary (only when BPPMODE_F = 4'b1110)</p>	0
ENWIN_F	[0]	RW	<p>Enables/disables the video output and logic immediately. 0 = Disables the video output and video control signal. 1 = Enables the video output and video control signal.</p>	0

16.5.4.12 WINCON3

- Address = Base Address + 0x002C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
BUFSTATUS_H	[31]	RW	Specifies the Buffer Status (read only). NOTE: BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L} 00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2	-
BUFSEL_H	[30]	RW	Selects the Buffer set NOTE: BUFSEL = {BUFSEL_H, BUFSEL_L} 00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2 (only available where BUF_MODE == 1'b1)	-
RSVD	[29:26]	-	Reserved (should be 0).	-
TRIGSTATUS	[25]	RW	Specifies the Trigger Status (read only) 0 = No trigger is issued 1 = Trigger is issued	-
RSVD	[24:22]	-	Reserved (should be 0).	-
BUFSTATUS_L	[21]	RW	Specifies the Buffer Status (read only). NOTE: BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L}	-
BUFSEL_L	[20]	RW	Selects the Buffer set. NOTE: BUFSEL = {BUFSEL_H, BUFSEL_L}	-
BUFAUTOEN	[19]	RW	Specifies the Double Buffer Auto control bit. 0 = Fixed by BUFSEL 1 = Auto changed by Trigger Input	-
BITSWP_F	[18]	RW	Specifies the Bit swap control bit. 0 = Swap Disable 1 = Swap Enable	0
BYTSWP_F	[17]	RW	Specifies the Byte swaps control bit. 0 = Swap Disable 1 = Swap Enable	0
HAWSWP_F	[16]	RW	Specifies the Half-Word swap control bit. 0 = Swap Disable 1 = Swap Enable	0
WSWP_F	[15]	RW	Specifies the Word swap control bit. 0 = Swap Disable 1 = Swap Enable	
BUF_MODE	[14]	RW	Selects the auto-buffering mode. 0 = Double 1 = Triple	0
RSVD	[13:11]	-	Reserved (should be 0).	0
BURSTLEN	[10:9]	RW	Selects the DMA Burst Maximum Length.	0

Name	Bit	Type	Description	Reset Value
			00 = 16 word- burst 01 = 8 word- burst 10 = 4 word- burst	
RSVD	[8]	-	Reserved (should be 0).	0
ALPHA_MUL_F	[7]	RW	Specifies the Multiplied Alpha value mode. 0 = Disables 1 = Enables multiplied mode When ALPHA_MUL is 1, set BLD_PIX =1, ALPHA_SEL=1, and BPPMODE_F[5:2] = 4'b1101 or 4'b1110. NOTE: Alpha value = alpha_pixel (from data) × ALPHA0_R/G/B	0
BLD_PIX_F	[6]	RW	Selects the blending category. 0 = Per plane blending 1 = Per pixel blending	
BPPMODE_F	[5:2]	RW	Selects the Bits Per Pixel (BPP) mode in Window image. 0000 = 1 bpp 0001 = 2 bpp 0010 = 4 bpp 0011 = 8 bpp (palletized) 0100 = 8 bpp (non-palletized, A: 1-R:2-G:3-B:2) 0101 = 16 bpp (non-palletized, R:5-G:6-B:5) 0110 = 16 bpp (non-palletized, A:1-R:5-G:5-B:5) 0111 = 16 bpp (non-palletized, I :1-R:5-G:5-B:5) 1000 = Unpacked 18 bpp (non-palletized, R:6-G:6-B:6) 1001 = Unpacked 18 bpp (non-palletized, A:1-R:6-G:6-B:5) 1010 = Unpacked 19 bpp (non-palletized, A:1-R:6-G:6-B:6) 1011 = Unpacked 24 bpp (non-palletized R:8-G:8-B:8) 1100 = Unpacked 24 bpp (non-palletized A:1-R:8-G:8-B:7) 1101 = Unpacked 25 bpp (non-palletized A:1-R:8-G:8-B:8) ⁽¹⁾ 1110 = Unpacked 13 bpp (non-palletized A:1-R:4-G:4-B:4) ⁽²⁾ 1111 = Unpacked 15 bpp (non-palletized R:5-G:5-B:5) NOTE: 1. 1101 = Supports unpacked 32 bpp (non-palletized A:8-R:8-G:8-B:8) for per pixel blending. 2. 1110 = Supports 16 bpp (non-palletized A:4-R:4-G:4-B:4) for per pixel blending. (16 level blending)	0
ALPHA_SEL_F	[1]	RW	Selects the Alpha value. When Per plane blending case BLD_PIX ==0: 0 = Using ALPHA0_R/G/B values 1 = Using ALPHA1_R/G/B values When Per pixel blending BLD_PIX ==1:	0

Name	Bit	Type	Description	Reset Value
			0 = Selected by AEN (A value) 1 = Using DATA[31:24] data in word boundary (only when BPPMODE_F = 4'b1101) DATA[31:28], [15:12] data in word boundary (only when BPPMODE_F = 4'b1110)	
ENWIN_F	[0]	RW	Enables/disables video output and logic immediately. 0 = Disables the video output and video control signal. 1 = Enables the video output and video control signal.	0

16.5.4.13 WINCON4

- Address = Base Address + 0x0030, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
BUFSTATUS_H	[31]	RW	Specifies the Buffer Status (read only). NOTE: BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L} 00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2	0
BUFSEL_H	[30]	RW	Selects the Buffer set. NOTE: BUFSEL = {BUFSEL_H, BUFSEL_L} 00 = Buffer set 0 01 = Buffer set 1 10 = Buffer set 2 (only available where BUF_MODE == 1'b1)	0
RSVD	[29:26]	-	Reserved (should be 0).	0
TRIGSTATUS	[25]	RW	Specifies the Trigger Status (read only). 0 = No trigger is issued 1 = Trigger is issued	0
RSVD	[24:22]	-	Reserved (should be 00).	0
BUFSTATUS_L	[21]	RW	Specifies the Buffer Status (read only). NOTE: BUFSTATUS = {BUFSTATUS_H, BUFSTATUS_L}	0
BUFSEL_L	[20]	RW	Selects the Buffer set. NOTE: BUFSEL = {BUFSEL_H, BUFSEL_L}	0
BUFAUTOEN	[19]	RW	Specifies the Double Buffer Auto control bit. 0 = Fixed by BUFSEL 1 = Auto changed by Trigger Input	0
BITSWP_F	[18]	RW	Specifies the Bit swap control bit. 0 = Swap Disable 1 = Swap Enable	0
BYTSWP_F	[17]	RW	Specifies the Byte swap control bit. 0 = Swap Disable 1 = Swap Enable	0
HAWSWP_F	[16]	RW	Specifies the Half-Word swap control bit. 0 = Swap Disable 1 = Swap Enable	0
WSWP_F	[15]	RW	Specifies the Word swap control bit. 0 = Swap Disable 1 = Swap Enable	0
BUF_MODE	[14]	RW	Selects the auto-buffering mode. 0 = Double 1 = Triple	0
RSVD	[13:11]	-	Reserved (should be 0).	0
BURSTLEN	[10:9]	RW	Selects the DMA Burst Maximum Length.	0

Name	Bit	Type	Description	Reset Value
			When Per pixel blending BLD_PIX ==1: 0 = Selected by AEN (A value) 1 = Using DATA[31:24] data in word boundary (only when BPPMODE_F = 4'b1101) DATA[31:28], [15:12] data in word boundary (only when BPPMODE_F = 4'b1110)	
ENWIN_F	[0]	RW	Enables/disables video output and logic immediately. 0 = Disables the video output and video control signal. 1 = Enables the video output and video control signal.	0

16.5.4.14 SHADOWCON

- Address = Base Address + 0x0034, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:15]	-	Reserved (should be 0).	0
W4_SHADOW _PROTECT	[14]	RW	Protects to update window 4's shadow register (xxx_F). 0 = Updates shadow register per frame 1 = Protects to update (updates shadow register at next frame after 'SHADOW_PROTECT' turns to be 1'b0)	0
W3_SHADOW _PROTECT	[13]	RW	Protects to update window 3's shadow register (xxx_F) 0 = Updates shadow register per frame 1 = Protects to update (updates shadow register at next frame after 'SHADOW_PROTECT' turns to be 1'b0)	0
W2_SHADOW _PROTECT	[12]	RW	Protects to update window 2's shadow register (xxx_F) 0 = Updates shadow register per frame 1 = Protects to update (updates shadow register at next frame after 'SHADOW_PROTECT' turns to be 1'b0)	0
W1_SHADOW _PROTECT	[11]	RW	Protects to update window 1's shadow register (xxx_F) 0 = Updates shadow register per frame 1 = Protects to update (update shadow register at next frame after 'SHADOW_PROTECT' turns to be 1'b0)	0
W0_SHADOW _PROTECT	[10]	RW	Protects to update window 0's shadow register (xxx_F) 0 = Updates shadow register per frame 1 = Protects to update (update shadow register at next frame after 'SHADOW_PROTECT' turns to be 1'b0)	0
RSVD	[9:8]	-	Reserved	0
C2_ENLOCAL_F	7	RW	Enables Channel 2 Local Path. 0 = Disables 1 = Enables	0
C1_ENLOCAL_F	6	RW	Enables Channel 1 Local Path. 0 = Disables 1 = Enables	0
C0_ENLOCAL_F	5	RW	Enables Channel 0 Local Path. 0 = Disables 1 = Enables	0
C4_EN_F	4	RW	Enables Channel 4. 0 = Disables 1 = Enables	0
C3_EN_F	3	RW	Enables Channel 3. 0 = Disables 1 = Enables	0
C2_EN_F	2	RW	Enables Channel 2. 0 = Disables 1 = Enables	0
C1_EN_F	1	RW	Enables Channel 1.	0

Name	Bit	Type	Description	Reset Value
			0 = Disables 1 = Enables	
C0_EN_F	0	RW	Enables Channel 0. 0 = Disables 1 = Enables	0

16.5.4.15 VIDOSD0A

- Address = Base Address + 0x0040, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
OSD_LeftTopX_F	[21:11]	RW	Specifies the horizontal screen coordinate for left top pixel of OSD image.	0
OSD_LeftTopY_F	[10:0]	RW	Specifies the vertical screen coordinate for left top pixel of OSD image (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate must be even.)	0

16.5.4.16 VIDOSD0B

- Address = Base Address + 0x0044, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
OSD_RightBotX_F	[21:11]	RW	Specifies the horizontal screen coordinate for right bottom pixel of OSD image.	0
OSD_RightBotY_F	[10:0]	RW	Specifies the vertical screen coordinate for right bottom pixel of OSD image. (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate must be odd value.)	0

NOTE: Registers must have word boundary X position. Therefore, 24 bpp mode must have X position by 1 pixel. (For example, X = 0, 1, 2, 3....)

16 bpp mode must have X position by 2 pixel. (For example, X = 0, 2, 4, 6....)

8 bpp mode must have X position by 4 pixel. (For example, X = 0, 4, 8, 12....)

16.5.4.17 VIDOSD0C

- Address = Base Address + 0x0048, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[25:24]	—	Reserved (should be 0)	0
OSDSIZE	[23:0]	RW	Specifies the Window Size For example, Height × Width (Number of Word)	0

16.5.4.18 VIDOSD0C

- Address = Base Address + 0x0050, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
OSD_LeftTopX_F	[21:11]	RW	Specifies the Horizontal screen coordinate for left top pixel of OSD image.	0
OSD_LeftTopY_F	[10:0]	RW	Specifies the Vertical screen coordinate for left top pixel of OSD image. (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate must be even.)	0

16.5.4.19 VIDOSD1B

- Address = Base Address + 0x0054, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
OSD_RightBotX_F	[21:11]	RW	Specifies the horizontal screen coordinate for right bottom pixel of OSD image.	0
OSD_RightBotY_F	[10:0]	RW	Specifies the vertical screen coordinate for right bottom pixel of OSD image. (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate must be odd value.)	0

NOTE: Registers must have word boundary X position.

Therefore, 24 bpp mode must have X position by 1 pixel. (For example, X = 0, 1, 2, 3....)

16 bpp mode must have X position by 2 pixel. (For example, X = 0, 2, 4, 6....)

8 bpp mode must have X position by 4 pixel. (For example, X = 0, 4, 8, 12....)

16.5.4.20 VIDOSD1C

- Address = Base Address + 0x0058, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[24]	–	Reserved	0
ALPHA0_R_H_F	[23:20]	RW	Specifies Red Alpha upper value (case AEN == 0)	0
ALPHA0_G_H_F	[19:16]	RW	Specifies Green Alpha upper value (case AEN == 0)	0
ALPHA0_B_H_F	[15:12]	RW	Specifies Blue Alpha upper value (case AEN == 0)	0
ALPHA1_R_H_F	[11:8]	RW	Specifies Red Alpha upper value (case AEN == 1)	0
ALPHA1_G_H_F	[7:4]	RW	Specifies Green Alpha upper value (case AEN == 1)	0
ALPHA1_B_H_F	[3:0]	RW	Specifies Blue Alpha upper value (case AEN == 1)	0

NOTE: Refer to VIDW1ALPHA0,1 register

16.5.4.21 VIDOSD1D

- Address = Base Address + 0x005C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[25:24]	–	Reserved (should be 0)	0
OSDSIZE	[23:0]	RW	Specifies Window Size. For example, Height × Width(Number of Word)	0

16.5.4.22 VIDOSD2A

- Address = Base Address + 0x0060, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
OSD_LeftTopX_F	[21:11]	RW	Specifies the horizontal screen coordinate for left top pixel of OSD image.	0
OSD_LeftTopY_F	[10:0]	RW	Specifies the vertical screen coordinate for left top pixel of OSD image. (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate must be even value.)	0

16.5.4.23 VIDOSD2B

- Address = Base Address + 0x0064, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
OSD_RightBotX_F	[21:11]	RW	Specifies the horizontal screen coordinate for right bottom pixel of OSD image.	0
OSD_RightBotY_F	[10:0]	RW	Specifies the vertical screen coordinate for right bottom pixel of OSD image. (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate must be odd value.)	0

NOTE: Registers must have word boundary X position.

Therefore, 24 bpp mode must have X position by 1 pixel. (For example, X = 0, 1, 2, 3....)

16 bpp mode must have X position by 2 pixel. (For example, X = 0, 2, 4, 6....)

8 bpp mode must have X position by 4 pixel. (For example, X = 0, 4, 8, 12....)

16.5.4.24 VIDOSD2C

- Address = Base Address + 0x0068, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[24]	-	Reserved	0
ALPHA0_R_H_F	[23:20]	RW	Specifies the Red Alpha upper value (case AEN == 0).	0
ALPHA0_G_H_F	[19:16]	RW	Specifies the Green Alpha upper value (case AEN == 0).	0
ALPHA0_B_H_F	[15:12]	RW	Specifies the Blue Alpha upper value (case AEN == 0).	0
ALPHA1_R_H_F	[11:8]	RW	Specifies the Red Alpha upper value (case AEN == 1).	0
ALPHA1_G_H_F	[7:4]	RW	Specifies the Green Alpha upper value (case AEN == 1).	0
ALPHA1_B_H_F	[3:0]	RW	Specifies the Blue Alpha upper value (case AEN == 1).	0

NOTE: For more information, refer to VIDW2ALPHA0, 1 register.

16.5.4.25 VIDOSD2D

- Address = Base Address + 0x006C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[25:24]	-	Reserved (should be 0)	0
OSDSIZE	[23:0]	RW	Specifies the Window Size For example, Height × Width(Number of Word)	0

16.5.4.26 VIDOSD3A

- Address = Base Address + 0x0070, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
OSD_LeftTopX_F	[21:11]	RW	Specifies the Horizontal screen coordinate for left top pixel of OSD image	0
OSD_LeftTopY_F	[10:0]	RW	Specifies the Vertical screen coordinate for left top pixel of OSD image. (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate must be even value.)	0

16.5.4.27 VIDOSD3B

- Address = Base Address + 0x0074, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
OSD_RightBotX_F	[21:11]	RW	Specifies the Horizontal screen coordinate for right bottom pixel of OSD image.	0
OSD_RightBotY_F	[10:0]	RW	Specifies the Vertical screen coordinate for right bottom pixel of OSD image. (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate must be odd value.)	0

NOTE: Registers must have word boundary X position.

Therefore, 24 bpp mode must have X position by 1 pixel. (For example, X = 0, 1, 2, 3....)

16 bpp mode must have X position by 2 pixel. (For example, X = 0, 2, 4, 6....)

8 bpp mode must have X position by 4 pixel. (For example, X = 0, 4, 8, 12....)

16.5.4.28 VIDOSD3C

- Address = Base Address + 0x0078, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[24]	–	Reserved	0
ALPHA0_R_H_F	[23:20]	RW	Specifies the Red Alpha upper value (case AEN == 0).	0
ALPHA0_G_H_F	[19:16]	RW	Specifies the Green Alpha upper value (case AEN == 0).	0
ALPHA0_B_H_F	[15:12]	RW	Specifies the Blue Alpha upper value (case AEN == 0).	0
ALPHA1_R_H_F	[11:8]	RW	Specifies the Red Alpha upper value (case AEN == 1).	0
ALPHA1_G_H_F	[7:4]	RW	Specifies the Green Alpha upper value (case AEN == 1).	0
ALPHA1_B_H_F	[3:0]	RW	Specifies the Blue Alpha upper value (case AEN == 1).	0

NOTE: For more information, refer to VIDW3ALPHA0, 1 register.

16.5.4.29 VIDOSD4A

- Address = Base Address + 0x0080, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
OSD_LeftTopX_F	[21:11]	RW	Specifies the Horizontal screen coordinate for left top pixel of OSD image.	0
OSD_LeftTopY_F	[10:0]	RW	Specifies the Vertical screen coordinate for left top pixel of OSD image. (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate MUST be even value.)	0

16.5.4.30 VIDOSD4B

- Address = Base Address + 0x0084, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
OSD_RightBotX_F	[21:11]	RW	Specifies the Horizontal screen coordinate for right bottom pixel of OSD image.	0
OSD_RightBotY_F	[10:0]	RW	Specifies the Vertical screen coordinate for right bottom pixel of OSD image. (For interlace TV output, this value must be set to half of the original screen y coordinate. The original screen y coordinate MUST be odd value.)	0

NOTE: Registers must have word boundary X position.

Therefore, 24 bpp mode must have X position by 1 pixel. (For example, X = 0, 1, 2, 3....)

16 bpp mode must have X position by 2 pixel. (For example, X = 0, 2, 4, 6....)

8 bpp mode must have X position by 4 pixel. (For example, X = 0, 4, 8, 12....)

16.5.4.31 VIDOSD4C

- Address = Base Address + 0x0088, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[24]	-	Reserved	0
ALPHA0_R_H_F	[23:20]	RW	Specifies the Red Alpha upper value (case AEN == 0).	0
ALPHA0_G_H_F	[19:16]	RW	Specifies the Green Alpha upper value (case AEN == 0).	0
ALPHA0_B_H_F	[15:12]	RW	Specifies the Blue Alpha upper value (case AEN == 0).	0
ALPHA1_R_H_F	[11:8]	RW	Specifies the Red Alpha upper value (case AEN == 1).	0
ALPHA1_G_H_F	[7:4]	RW	Specifies the Green Alpha upper value (case AEN == 1).	0
ALPHA1_B_H_F	[3:0]	RW	Specifies the Blue Alpha upper value (case AEN == 1).	0

NOTE: For more information, refer to VIDW4ALPHA0, 1 register.

16.5.4.32 VIDW0nADD0Bn (n = 0 to 4)

- Address = Base Address + 0x00A0, Reset Value = 0x0000_0000 (VIDW00ADD0B0)
- Address = Base Address + 0x00A4, Reset Value = 0x0000_0000 (VIDW00ADD0B1)
- Address = Base Address + 0x20A0, Reset Value = 0x0000_0000 (VIDW00ADD0B2)
- Address = Base Address + 0x00A8, Reset Value = 0x0000_0000 (VIDW01ADD0B0)
- Address = Base Address + 0x00AC, Reset Value = 0x0000_0000 (VIDW01ADD0B1)
- Address = Base Address + 0x20A8, Reset Value = 0x0000_0000 (VIDW01ADD0B2)
- Address = Base Address + 0x00B0, Reset Value = 0x0000_0000 (VIDW02ADD0B0)
- Address = Base Address + 0x00B4, Reset Value = 0x0000_0000 (VIDW02ADD0B1)
- Address = Base Address + 0x20B0, Reset Value = 0x0000_0000 (VIDW02ADD0B2)
- Address = Base Address + 0x00B8, Reset Value = 0x0000_0000 (VIDW03ADD0B0)
- Address = Base Address + 0x00BC, Reset Value = 0x0000_0000 (VIDW03ADD0B1)
- Address = Base Address + 0x20B8, Reset Value = 0x0000_0000 (VIDW03ADD0B2)
- Address = Base Address + 0x00C0, Reset Value = 0x0000_0000 (VIDW04ADD0B0)
- Address = Base Address + 0x00C4, Reset Value = 0x0000_0000 (VIDW04ADD0B1)
- Address = Base Address + 0x20C0, Reset Value = 0x0000_0000 (VIDW04ADD0B2)

Name	Bit	Type	Description	Reset Value
VBASEU_F	[31:0]	RW	Specifies A[31:0] of the start address for Video frame buffer.	0

16.5.4.33 VIDW0nADD1Bn (n = 0 to 4)

- Address = Base Address + 0x00D0, Reset Value = 0x0000_0000 (VIDW00ADD1B0)
- Address = Base Address + 0x00D4, Reset Value = 0x0000_0000 (VIDW00ADD1B1)
- Address = Base Address + 0x20D0, Reset Value = 0x0000_0000 (VIDW00ADD1B2)
- Address = Base Address + 0x00D8, Reset Value = 0x0000_0000 (VIDW01ADD1B0)
- Address = Base Address + 0x00DC, Reset Value = 0x0000_0000 (VIDW01ADD1B1)
- Address = Base Address + 0x20D8, Reset Value = 0x0000_0000 (VIDW01ADD1B2)
- Address = Base Address + 0x00E0, Reset Value = 0x0000_0000 (VIDW02ADD1B0)
- Address = Base Address + 0x00E4, Reset Value = 0x0000_0000 (VIDW02ADD1B1)
- Address = Base Address + 0x20E0, Reset Value = 0x0000_0000 (VIDW02ADD1B2)
- Address = Base Address + 0x00E8, Reset Value = 0x0000_0000 (VIDW03ADD1B0)
- Address = Base Address + 0x00EC, Reset Value = 0x0000_0000 (VIDW03ADD1B1)
- Address = Base Address + 0x20E8, Reset Value = 0x0000_0000 (VIDW03ADD1B2)
- Address = Base Address + 0x00F0, Reset Value = 0x0000_0000 (VIDW04ADD1B0)
- Address = Base Address + 0x00F4, Reset Value = 0x0000_0000 (VIDW04ADD1B1)
- Address = Base Address + 0x20F0, Reset Value = 0x0000_0000 (VIDW04ADD1B2)

Name	Bit	Type	Description	Reset Value
VBASEL_F	[31:0]	RW	Specifies A[31:0] of the end address for Video frame buffer. VBASEL = VBASEU + (PAGEWIDTH+OFFSIZE) × (LINEVAL+1)	0x0

16.5.4.34 VIDW0nADD2 (n = 0 to 4)

- Address = Base Address + 0x0100, Reset Value = 0x0000_0000 (VIDW00ADD2)
- Address = Base Address + 0x0104, Reset Value = 0x0000_0000 (VIDW01ADD2)
- Address = Base Address + 0x0108, Reset Value = 0x0000_0000 (VIDW02ADD2)
- Address = Base Address + 0x010C, Reset Value = 0x0000_0000 (VIDW03ADD2)
- Address = Base Address + 0x0110, Reset Value = 0x0000_0000 (VIDW04ADD2)

Name	Bit	Type	Description	Reset Value
OFFSIZE_F	[25:13]	RW	Specifies the Virtual screen offset size (number of byte). This value defines the difference between address of last byte displayed on the previous Video line and address of first byte to be displayed in the new Video line. OFFSIZE_F must have value that is multiple of 4-byte size or 0.	0
PAGEWIDTH_F	[12:0]	RW	Specifies the Virtual screen page width (number of byte). This value defines the width of view port in the frame. PAGEWIDTH must have bigger value than the burst size and the size must be aligned word boundary.	0

NOTE: "PAGEWIDTH + OFFSET" should be aligned double-word aligned (8-byte).

16.5.4.35 VIDINTCON0

- Address = Base Address + 0x0130, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	-	Reserved	0
FIFOINTERVAL	[25:20]	RW	Controls the interval of the FIFO interrupt.	0
SYSMAINCON	[19]	RW	Sends complete interrupt enable bit to Main LCD 0 = Disables Interrupt 1 = Enables Interrupt NOTE: This bit is meaningful if both INTEN and I80IFDONE are high.	0
SYSSUBCON	[18]	RW	Sends complete interrupt enable bit to Sub LCD 0 = Disables Interrupt. 1 = Enables Interrupt. NOTE: This bit is meaningful if both INTEN and I80IFDONE are high.	0
I80IFDONE	[17]	RW	Enables the I80 Interface Interrupt (only for I80 Interface mode). 0 = Disables Interrupt. 1 = Enables Interrupt. NOTE: This bit is meaningful if INTEN is high.	0
FRAMESEL0	[16:15]	RW	Specifies the Video Frame Interrupt 0 at start of: 00 = BACK Porch 01 = VSYNC 10 = ACTIVE 11 = FRONT Porch	0
FRAMESEL1	[14:13]	RW	Specifies the Video Frame Interrupt 1 at start of: 00 = None 01 = BACK Porch 10 = VSYNC 11 = FRONT Porch	0
INTFRMEN	[12]	RW	Specifies the Video Frame Interrupt Enable Control Bit. 0 = Disables Video Frame Interrupt 1 = Enables Video Frame Interrupt NOTE: This bit is meaningful when INTEN is high.	0
FIFOSEL	[11:5]	RW	Specifies the FIFO Interrupt control bit. Each bit has a special significance: [11] Window 4 control (0 = disables, 1 = enables) [10] Window 3 control (0 = disables, 1 = enables) [9] Window 2 control (0 = disables, 1 = enables) [8] Reserved [7] Reserved [6] Window 1 control (0 = disables, 1 = enables) [5] Window 0 control (0 = disables, 1 = enables) NOTE: This bit is meaningful if both INTEN and INTFIFOEN are high	0

Name	Bit	Type	Description	Reset Value
FIFOLEVEL	[4:2]	RW	Selects the Video FIFO Interrupt Level. 011 = 0% (empty) 100 = 100% (full)	0
INTFIFOEN	[1]	RW	Specifies the Video FIFO Interrupt Enable Control Bit. 0 = Disables Video FIFO Level Interrupt 1 = Enables Video FIFO Level Interrupt NOTE: This bit is meaningful if INTEN is high.	0
INTEN	[0]	RW	Specifies the Video Interrupt Enable Control Bit. 0 = Disables Video Interrupt 1 = Enables Video Interrupt	0

NOTE:

1. If video frame interrupt occurs, you can select maximum two points by setting FRAMESEL0 and FRAMESEL1. For example, in case of FRAMESEL0=00 and FRAMESEL1=11, video frame interrupt is triggered both at the start of back porch and front porch.
2. Interrupt controller has three interrupt sources related to display controller, namely, LCD[0], LCD[1] and LCD[2]. (For more information, refer to Chapter 6, "Interrupt Controller"). LCD[0] specifies FIFO Level interrupt, LCD[1] specifies video frame sync interrupt and LCD[2] specifies i80 done interface interrupt.

16.5.4.36 VIDINTCON1

- Address = Base Address + 0x0134, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:5]	–	Reserved	0
RSVD	[4:3]	–	Reserved (should be 0).	0
INTI80PEND	[2]	RW	Specifies the i80 Done interrupt. Writes “1” to clear this bit. 0 = Interrupt has not been requested 1 = I80 Done status has asserted the interrupt request	0
INTFRMPEND	[1]	RW	Specifies the Frame sync interrupt. Writes “1” to clear this bit. 0 = Interrupt has not been requested 1 = Frame sync status has asserted the interrupt request	0
INTFIFOOPEND	[0]	RW	Specifies the FIFO Level interrupt. Writes “1” to clear this bit. 0 = Interrupt has not been requested 1 = FIFO empty status has asserted the interrupt request	0

16.5.4.37 W1KEYCON0

- Address = Base Address + 0x0140, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
KEYBLEN_F	[26]	RW	Enables blending. 0 = Disables blending 1 = Enables blending using original Alpha for non-key area and KEY_ALPHA for key area	0
KEYEN_F	[25]	RW	Enables/Disables Color Key (Chroma key). 0 = Disables color key 1 = Enables color key	0
DIRCON_F	[24]	RW	Controls color key (Chroma key) direction. 0 = If the pixel value matches foreground image with COLVAL, the pixel from background image is displayed (only in OSD area) 1 = If the pixel value matches background image with COLVAL, the pixel from foreground image is displayed (only in OSD area)	0
COMPKEY_F	[23:0]	RW	Each bit corresponds to COLVAL[23:0]. If some position bit is set, then it disables the position bit of COLVAL.	0

NOTE: Set BLD_PIX = 1, ALPHA_SEL =0, A_FUNC = 0x2, and B_FUNC = 0x3 to enable alpha blending using color key.

16.5.4.38 W1KEYCON1

- Address = Base Address + 0x0144, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
COLVAL_F	[23:0]	RW	Specifies the color key value for transparent pixel effect.	0

16.5.4.39 W2KEYCON0

- Address = Base Address + 0x0148, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
KEYBLEN_F	[26]	RW	Enables blending. 0 = Disables blending 1 = Enables blending using original Alpha for non-key area and KEY_ALPHA for key area	0
KEYEN_F	[25]	RW	Enables color key (Chroma key). 0 = Disables color key 1 = Enables color key	0
DIRCON_F	[24]	RW	Controls color key (Chroma key) direction. 0 = If the pixel value matches foreground image with COLVAL, the pixel from background image is displayed (only in OSD area) 1 = If the pixel value matches background image with COLVAL, the pixel from foreground image is displayed (only in OSD area)	0
COMPKEY_F	[23:0]	RW	Each bit corresponds to COLVAL [23:0]. If some position bit is set, then it disables the position bit of COLVAL.	0

NOTE: Set BLD_PIX = 1, ALPHA_SEL =0, A_FUNC = 0x2, and B_FUNC = 0x3 to enable alpha blending using color key.

16.5.4.40 W2KEYCON1

- Address = Base Address + 0x014C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
COLVAL_F	[23:0]	RW	Specifies the color key value for transparent pixel effect.	0

16.5.4.41 W3KEYCON0

- Address = Base Address + 0x0150, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
KEYBLEN_F	[26]	RW	Enables blending. 0 = Disables blending 1 = Enables blending using original Alpha for non-key area and KEY_ALPHA for key area	0
KEYEN_F	[25]	RW	Enables Color Key (Chroma key). 0 = Disables color key 1 = Enables color key	0
DIRCON_F	[24]	RW	Controls Color key (Chroma key) direction. 0 = If the pixel value matches foreground image with COLVAL, the pixel from background image is displayed (only in OSD area) 1 = If the pixel value matches background image with COLVAL, the pixel from foreground image is displayed (only in OSD area)	0
COMPKEY_F	[23:0]	RW	Each bit corresponds to COLVAL [23:0]. If some position bit is set, then it disables the position bit of COLVAL.	0

NOTE: Set BLD_PIX = 1, ALPHA_SEL =0, A_FUNC = 0x2, and B_FUNC = 0x3 to enable alpha blending using color key.

16.5.4.42 W3KEYCON1

- Address = Base Address + 0x0154, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
COLVAL_F	[23:0]	RW	Specifies the color key value for transparent pixel effect.	0

16.5.4.43 W4KEYCON0

- Address = Base Address + 0x0158, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
KEYBLEN_F	[26]	RW	Enables blending. 0 = Disables blending 1 = Enables blending using original Alpha for non-key area and KEY_ALPHA for key area	0
KEYEN_F	[25]	RW	Enables color Key (Chroma key). 0 = Disables color key 1 = Enables color key	0
DIRCON_F	[24]	RW	Controls color key (Chroma key) direction. 0 = If the pixel value matches foreground image with COLVAL, the pixel from background image is displayed (only in OSD area) 1 = If the pixel value matches background image with COLVAL, the pixel from foreground image is displayed (only in OSD area)	0
COMPKEY_F	[23:0]	RW	Each bit corresponds to COLVAL [23:0]. If some position bit is set, then it disables the COLVAL position bit.	0

NOTE: Set BLD_PIX = 1, ALPHA_SEL =0, A_FUNC = 0x2, and B_FUNC = 0x3 to enable alpha blending using color key.

16.5.4.44 W4KEYCON1

- Address = Base Address + 0x015C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
COLVAL_F	[23:0]	RW	Specifies the color key value for transparent pixel effect.	0

NOTE: Both COLVAL and COMPKEY use 24-bit color data in all bpp modes.

@ bpp 24 mode: 24-bit color value is valid.

- A. COLVAL
 - Red: COLVAL[23:17]
 - Green: COLVAL[15: 8]
 - Blue: COLVAL[7:0]
- B. COMPKEY
 - Red: COMPKEY[23:17]
 - Green: COMPKEY[15: 8]
 - Blue: COMPKEY[7:0]

@ BPP16 (5:6:5) mode: 16-bit color value is valid.

- A. COLVAL
 - Red: COLVAL[23:19]
 - Green: COLVAL[15: 10]
 - Blue: COLVAL[7:3]
- B. COMPKEY
 - Red: COMPKEY[23:19]
 - Green: COMPKEY[15: 10]
 - Blue: COMPKEY[7:3]
 - COMPKEY[18:16] must be 0x7.
 - COMPKEY[9: 8] must be 0x3.
 - COMPKEY[2:0] must be 0x7.

NOTE: COMPKEY register must be set properly for each bpp mode.

16.5.4.45 W1KEYALPHA

- Address = Base Address + 0x0160, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:14]	–	Reserved.	0
KEYALPHA_R_F	[23:0]	RW	Specifies the Key alpha R value.	0
KEYALPHA_G_F	[15:8]	RW	Specifies the Key alpha G value.	0
KEYALPHA_B_F	[7:0]	RW	Specifies the Key alpha B value.	0

16.5.4.46 W2KEYALPHA

- Address = Base Address + 0x0164, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:14]	–	Reserved.	0
KEYALPHA_R_F	[23:0]	RW	Specifies the Key alpha R value.	0
KEYALPHA_G_F	[15:8]	RW	Specifies the Key alpha G value.	0
KEYALPHA_B_F	[7:0]	RW	Specifies the Key alpha B value.	0

16.5.4.47 W3KEYALPHA

- Address = Base Address + 0x0168, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:14]	–	Reserved.	0
KEYALPHA_R_F	[23:0]	RW	Specifies the Key alpha R value.	0
KEYALPHA_G_F	[15:8]	RW	Specifies the Key alpha G value.	0
KEYALPHA_B_F	[7:0]	RW	Specifies the Key alpha B value.	0

16.5.4.48 W4KEYALPHA

- Address = Base Address + 0x016C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:14]	—	Reserved	0
KEYALPHA_R_F	[23:0]	RW	Specifies the Key alpha R value.	0
KEYALPHA_G_F	[15:8]	RW	Specifies the Key alpha G value.	0
KEYALPHA_B_F	[7:0]	RW	Specifies the Key alpha B value.	0

16.5.4.49 DITHMODE

- Address = Base Address + 0x0170, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[7]	RW	Not used for normal access (Writing non-zero values to these registers results in abnormal behavior.)	0
RDithPos	[6:5]	RW	Controls Red Dither bit. 00 = 8-bit 01 = 6-bit 10 = 5-bit	0
GDithPos	[4:3]	RW	Controls Green Dither bit. 00 = 8-bit 01 = 6-bit 10 = 5-bit	0
BDithPos	[2:1]	RW	Controls Blue Dither bit. 00 = 8-bit 01 = 6-bit 10 = 5-bit	0
DITHEN_F	[0]	RW	Enables Dithering bit. 0 = Disables dithering 1 = Enables dithering	0

16.5.4.50 WIN0MAP

- Address = Base Address + 0x0180, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
MAPCOLEN_F	[24]	RW	Specifies window's color mapping control bit. If this bit is enabled, then Video DMA stops and MAPCOLOR appears on background image instead of original image. 0 = Disables 1 = Enables	0
MAPCOLOR	[23:0]	RW	Specifies the color value.	0

16.5.4.51 WIN1MAP

- Address = Base Address + 0x0184, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
MAPCOLEN_F	[24]	RW	Specifies the window's color mapping control bit. If this bit is enabled, then Video DMA stops and MAPCOLOR appears on background image instead of original image. 0 = Disables 1 = Enables	0
MAPCOLOR	[23:0]	RW	Specifies the color value.	0

16.5.4.52 WIN2MAP

- Address = Base Address + 0x0188, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
MAPCOLEN_F	[24]	RW	Specifies the window's color mapping control bit. If this bit is enabled, then Video DMA stops and MAPCOLOR appears on background image instead of original image. 0 = Disables 1 = Enables	0
MAPCOLOR	[23:0]	RW	Specifies the color value.	0

16.5.4.53 WIN3MAP

- Address = Base Address +0x018C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
MAPCOLEN_F	[24]	RW	Specifies the window's color mapping control bit. If this bit is enabled, then Video DMA stops and MAPCOLOR appears on background image instead of original image. 0 = Disables 1 = Enables	0
MAPCOLOR	[23:0]	RW	Specifies the color value.	0

16.5.4.54 WIN4MAP

- Address = Base Address + 0x0190, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
MAPCOLEN_F	[24]	RW	Specifies the window's color mapping control bit. If this bit is enabled, then Video DMA stops and MAPCOLOR appears on background image instead of original image. 0 = Disables 1 = Enables	0
MAPCOLOR	[23:0]	RW	Specifies the color value.	0

16.5.4.55 WPALCON_H

- Address = Base Address + 0x019C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:19]	–	Reserved	0
W4PAL_H	[18:17]	RW	W4PAL[2:1]	0
RSVD	[16:15]	RW	Reserved	0
W3PAL_H	[14:13]	RW	W3PAL[2:1]	0
RSVD	[12:11]	RW	Reserved	0
W2PAL_H	[10: 9]	RW	W2PAL[2:1]	0
RSVD	[8: 0]	RW	Reserved	0

16.5.4.56 WPALCON_L

- Address = Base Address + 0x01A0, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:23]	-	Reserved	0
PALUPDATEEN	[9]	RW	0 = Normal Mode 1 = Enable (Palette Update)	0
W4PAL_L	[8]	RW	W4PAL[0]	0
W3PAL_L	[7]	RW	W3PAL[0]	0
W2PAL_L	[6]	RW	W2PAL[0]	0
W1PAL_L	[5:3]	RW	W1PAL[2:0]	0
W0PAL_L	[2:0]	RW	W0PAL[2:0]	0

NOTE: WPALCON = {WPALCON_H,WPALCON_L}

Name	Description	Reset Value
PALUPDATEEN	0 = Normal Mode 1 = Enable (Palette Update)	0
W4PAL[3:0]	Specifies the size of palette data format of Window 4. 000 = 16-bit (5:6:5) 001 = 16-bit (A:5:5:5) 010 = 18-bit (6:6:6) 011 = 18-bit (A:6:6:5) 100 = 19-bit (A:6:6:6) 101 = 24-bit (8:8:8) 110 = 25-bit (A:8:8:8) 111 = 32-bit (8:8:8:8) (A: 8-bit)	0
W3PAL[2:0]	Specifies the size of palette data format of Window 3. 000 = 16-bit (5:6:5) 001 = 16-bit (A:5:5:5) 010 = 18-bit (6:6:6) 011 = 18-bit (A:6:6:5) 100 = 19-bit (A:6:6:6) 101 = 24-bit (8:8:8) 110 = 25-bit (A:8:8:8) 111 = 32-bit (8:8:8:8) (A: 8-bit)	0
W2PAL[2:0]	Specifies the size of palette data format of Window 2. 000 = 16-bit (5:6:5) 001 = 16-bit (A:5:5:5) 010 = 18-bit (6:6:6) 011 = 18-bit (A:6:6:5) 100 = 19-bit (A:6:6:6) 101 = 24-bit (8:8:8)	0

Name	Description	Reset Value
	110 = 25-bit (A:8:8:8) 111 = 32-bit (8:8:8:8) (A: 8-bit)	
W1PAL[2:0]	Specifies the size of palette data format of Window 1. 000 = 25-bit (A:8:8:8) 001 = 24-bit (8:8:8) 010 = 19-bit (A:6:6:6) 011 = 18-bit (A:6:6:5) 100 = 18-bit (6:6:6) 101 = 16-bit (A:5:5:5) 110 = 16-bit (5:6:5) 111 = 32-bit (8:8:8:8) (A: 8-bit)	0
W0PAL[2:0]	Specifies the size of palette data format of Window 0. 000 = 25-bit (A:8:8:8) 001 = 24-bit (8:8:8) 010 = 19-bit (A:6:6:6) 011 = 18-bit (A:6:6:5) 100 = 18-bit (6:6:6) 101 = 16-bit (A:5:5:5) 110 = 16-bit (5:6:5) 111 = 32-bit (8:8:8:8) (A: 8-bit)	0

NOTE: The bit map for W0/W1 is different from W2/W3/W4.

16.5.4.57 TRIGCON

- Address = Base Address + 0x01A4, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:27]	—	Reserved	0
SWTRGCM _D _W4BUF	[26]	RW	Specifies Window 4 double buffer trigger. 1 = Enables Software Trigger Command (write only) *Only when TRGMODE_W4BUF is "1"	0
TRGMODE _W4BUF	[25]	RW	Specifies Window 4 double buffer trigger. 0 = Disables Trigger 1 = Enables Trigger	0
RSVD	[24:22]	—	Reserved	0
SWTRGCM _D _W3BUF	[21]	RW	Specifies Window 3 double buffer trigger. 1 = Enables Software Trigger Command (write only) *Only when TRGMODE_W3BUF is "1"	0
TRGMODE _W3BUF	[20]	RW	Specifies Window 3 double buffer trigger. 0 = Disables Trigger 1 = Enables Trigger	0
RSVD	[19:17]	—	Reserved	0
SWTRGCM _D _W2BUF	[16]	RW	Specifies Window 2 double buffer trigger. 1 = Enables Software Trigger Command (write only) *Only when TRGMODE_W2BUF is "1"	0
TRGMODE _W2BUF	[15]	RW	Specifies Window 2 double buffer trigger. 0 = Disables Trigger 1 = Enables Trigger	0
RSVD	[14:12]	—	Reserved	0
SWTRGCM _D _W1BUF	[11]	RW	Specifies Window 1 double buffer trigger. 1 = Enables Software Trigger Command (write only) *Only when TRGMODE_W1BUF is "1"	0
TRGMODE _W1BUF	[10]	RW	Specifies Window 1 double buffer trigger. 0 = Disables Trigger 1 = Enables Trigger	0
RSVD	[9:7]	—	Reserved	0
SWTRGCM _D _W0BUF	[6]	RW	Specifies Window 0 double buffer trigger. 1 = Enables Software Trigger Command (write only) *Only when TRGMODE_W0BUF is "1"	0
TRGMODE_W0 BUF	[5]	RW	Specifies Window 0 double buffer trigger. 0 = Disables Trigger 1 = Enables Trigger	0

Name	Bit	Type	Description	Reset Value
RSVD	[4:3]	-	Reserved	0
SWFRSTATUS_I80	[2]	RW	Specifies Frame Done Status (read only; I80 start trigger) 0 = Not Requested 1 = Requested *Clear Condition: Read or New Frame Start *Only when TRGMODE is "1"	0
SWTRGCMDS_I80	[1]	RW	Enables I80 start trigger. 1 = Software Triggering Command (write only) *Only when TRGMODE is "1"	0
TRGMODE_I80	[0]	RW	Enables I80 start trigger. 0 = Disables i80 Software Trigger 1 = Enables i80 Software Trigger	0

NOTE: Two continuous software trigger inputs generated in some video clocks (VCLK) are recognized as one.

16.5.4.58 I80IFCONAn (n = 0 to 1)

- Address = Base Address + 0x01B0, Reset Value = 0x0000_0000
- Address = Base Address + 0x01B4, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[22:20]	–	Reserved	0
LCD_CS_SETUP	[19:16]	RW	Specifies the numbers of clock cycles for the active period of address signal enable to chip select enable.	0
LCD_WR_SETUP	[15:12]	RW	Specifies the numbers of clock cycles for the active period of CS signal enable to write signal enable.	0
LCD_WR_ACT	[11:8]	RW	Specifies the numbers of clock cycles for the active period of chip select enable.	0
LCD_WR_HOLD	[7:4]	RW	Specifies the numbers of clock cycles for the active period of chip select disable to write signal disable.	0
RSVD	[3]	–	Reserved	
RSPOL	[2]	RW	Specifies the polarity of RS Signal 0 = Low 1 = High	0
RSVD	[1]	–	Reserved	0
I80IFEN	[0]	RW	Controls the LCD I80 interface. 0 = Disables 1 = Enables	0

16.5.4.59 I80IFCONB_n (n = 0 to 1)

- Address = Base Address + 0x01B8, Reset Value = 0x0000_0000
- Address = Base Address + 0x01BC, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[11:10]	–	Reserved	0
NORMAL_CMD_ST	[9]	RW	1 = Normal Command Start * Auto clears after sending out one set of commands	0
RSVD	[8:7]	–	Reserved	
FRAME_SKIP	[6:5]	RW	Specifies the I80 Interface Output Frame Decimation Factor. 00 = 1 (No Skip) 01 = 2 10 = 3	00
RSVD	[4]	–	Reserved	0
AUTO_CMD_RATE	[3:0]	RW	0000 = Disables auto command (If you don't use any auto-command, then you should set AUTO_CMD_RATE as "0000"). 0001 = per 2 Frames 0010 = per 4 Frames 0011 = per 6 Frames ... 1111 = per 30 Frames	0000

16.5.4.60 COLORGAINCON

- Address = Base Address + 0x01C0, Reset Value = 0x1004_0100

Name	Bit	Type	Description	Reset Value
RSVD	[31:30]	-	Reserved	0
CG_RGAIN	[29:20]	RW	<p>Specifies the color gain value of R data (maximum 4, 8-bit resolution).</p> <p>0h000 = 0 0h001 = 0.00390625 (1/256) 0h002 = 0.0078125 (2/256) ... 0h0FF = 0.99609375 (255/256) 0h100 = 1.0 ... 0x3FF = 3.99609375 (max)</p>	0x100
CG_GGAIN	[19:10]	RW	<p>Specifies the color gain value of G data (maximum 4, 8-bit resolution).</p> <p>0h000 = 0 0h001 = 0.00390625 (1/256) 0h002 = 0.0078125 (2/256) ... 0h0FF = 0.99609375 (255/256) 0h100 = 1.0 ... 0x3FF = 3.99609375 (max)</p>	0x100
CG_BGAIN	[9:0]	RW	<p>Specifies the color gain value of B data (maximum 4, 8-bit resolution).</p> <p>0h000 = 0 0h001 = 0.00390625 (1/256) 0h002 = 0.0078125 (2/256) ... 0h0FF = 0.99609375 (255/256) 0h100 = 1.0 ... 0x3FF = 3.99609375 (max)</p>	0x100

16.5.4.61 LDI_CMDCON0

- Address = Base Address + 0x01D0, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:24]	-	Reserved	-
CMD11_EN	[23:22]	RW	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00
CMD10_EN	[21:20]	RW	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00
CMD9_EN	[19:18]	RW	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00
CMD8_EN	[17:16]	RW	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00
CMD7_EN	[15:14]	RW	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00
CMD6_EN	[13:12]	RW	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00
CMD5_EN	[11:10]	RW	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00
CMD4_EN	[9:8]	RW	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00
CMD3_EN	[7:6]	RW	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00
CMD2_EN	[5:4]	RW	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Normal and Auto Command Enable	00
CMD1_EN	[3:2]	RW	00 = Disables 01 = Enables Normal Command	00

Name	Bit	Type	Description	Reset Value
			10 = Enables Auto Command 11 = Enables Normal and Auto Command	
CMD0_EN	[1:0]	RW	00 = Disables 01 = Enables Normal Command 10 = Enables Auto Command 11 = Enables Normal and Auto Command	00

16.5.4.62 LDI_CMDCON1

- Address = Base Address + 0x01D4, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:10]	–	Reserved	0
CMD11_RS	[11]	RW	Controls Command 11 RS	0
CMD10_RS	[10]	RW	Controls Command 10 RS	0
CMD9_RS	[9]	RW	Controls Command 9 RS	0
CMD8_RS	[8]	RW	Controls Command 8 RS	0
CMD7_RS	[7]	RW	Controls Command 7 RS	0
CMD6_RS	[6]	RW	Controls Command 6 RS	0
CMD5_RS	[5]	RW	Controls Command 5 RS	0
CMD4_RS	[4]	RW	Controls Command 4 RS	0
CMD3_RS	[3]	RW	Controls Command 3 RS	0
CMD2_RS	[2]	RW	Controls Command 2 RS	0
CMD1_RS	[1]	RW	Controls Command 1 RS	0
CMD0_RS	[0]	RW	Controls Command 0 RS	0

16.5.4.63 SIFCCON0

- Address = Base Address + 0x01E0, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[7]	-	Reserved (should be 0)	0
SYS_ST_CON	[6]	RW	Controls LCD i80 System Interface ST Signal. 0 = Low 1 = High	0
SYS_RS_CON	[5]	RW	Controls LCD i80 System Interface RS Signal. 0 = Low 1 = High	0
SYS_nCS0_CON	[4]	RW	Controls LCD i80 System Interface nCS0 (main) Signal. 0 = Disables (High) 1 = Enables (Low)	0
SYS_nCS1_CON	[3]	RW	Controls LCD i80 System Interface nCS1 (sub) Signal. 0 = Disables (High) 1 = Enables (Low)	0
SYS_nOE_CON	[2]	RW	Controls LCD i80 System Interface nOE Signal. 0 = Disables (High) 1 = Enables (Low)	0
SYS_nWE_CON	[1]	RW	Controls LCD i80 System Interface nWE Signal. 0 = Disables (High) 1 = Enables (Low)	0
SCOMEN	[0]	RW	Enables LCD i80 System Interface Command Mode. 0 = Disables (Normal Mode) 1= Enables (Manual Command Mode)	

16.5.4.64 SIFCCON1

- Address = Base Address + 0x01E4, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
SYS_WDATA	[23:0]	RW	Controls the LCD i80 System Interface Write Data.	0

16.5.4.65 SIFCCON2

- Address = Base Address + 0x01E8, Reset Value = 0x????_????

Name	Bit	Type	Description	Reset Value
SYS_RDATA	[23:0]	R	Controls the LCD i80 System Interface Read Data.	0

16.5.4.66 HUECOEF_CR_n (n = 1 to 4)

- Address = Base Address + 0x01EC, Reset Value = 0x0100_0100
- Address = Base Address + 0x01F0, Reset Value = 0x0000_0000
- Address = Base Address + 0x01F4, Reset Value = 0x0000_0000
- Address = Base Address + 0x01F8, Reset Value = 0x0100_0100

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	-	Reserved	0
CRG0_x	[25:16]	RW	<p>Specifies the Hue matrix coefficient 00 (when "cb + ln_offset" is positive). (Signed) 0h000 = 0 0h001 = 0.00390625 (1/256) 0h002 = 0.0078125 (2/256) ... 0h0FF = 0.99609375 (255/256) 0h100 = 1.0 (256/256) 0h300 = - 1.0 (- 256/256) 0h301 = - 0.99609375 (- 255/256) ... 0h3FF = - 0.00390625 (- 1/256) 0h101 to 2FF = Reserved (don't use)</p>	0x100
RSVD	[15:10]	-	Reserved	0
CRG1_x	[9:0]	RW	<p>Specifies the Hue matrix coefficient 00 (when "cb + ln_offset" is negative). (Signed) 0h000 = 0 0h001 = 0.00390625 (1/256) 0h002 = 0.0078125 (2/256) ... 0h0FF = 0.99609375 (255/256) 0h100 = 1.0 (256/256) 0h300 = - 1.0 (- 256/256) 0h301 = - 0.99609375 (- 255/256) ... 0h3FF = - 0.00390625 (- 1/256) 0h101 to 2FF = Reserved (don't use)</p>	0x100

16.5.4.67 HUECOEF_CB_n (n = 1 to 4)

- Address = Base Address + 0x01FC, Reset Value = 0x0100_0100
- Address = Base Address + 0x0200, Reset Value = 0x0000_0000
- Address = Base Address + 0x0204, Reset Value = 0x0000_0000
- Address = Base Address + 0x0208, Reset Value = 0x0100_0100

Name	Bit	Type	Description	Reset Value
RSVD	[31:26]	-	Reserved	0
CBG0_x	[25:16]	RW	<p>Specifies the Hue matrix coefficient 00 (when "cb + ln_offset" is positive). (Signed) 0h000 = 0 0h001 = 0.00390625 (1/256) 0h002 = 0.0078125 (2/256) ... 0h0FF = 0.99609375 (255/256) 0h100 = 1.0 (256/256) 0h300 = - 1.0 (- 256/256) 0h301 = - 0.99609375 (- 255/256) ... 0h3FF = - 0.00390625 (- 1/256) 0h101 to 2FF = Reserved (don't use)</p>	0x100
RSVD	[15:10]	-	Reserved	0
CBG1_x	[9:0]	RW	<p>Specifies the Hue matrix coefficient 00 (when "cb + ln_offset" is negative). (Signed) 0h000 = 0 0h001 = 0.00390625 (1/256) 0h002 = 0.0078125 (2/256) ... 0h0FF = 0.99609375 (255/256) 0h100 = 1.0 (256/256) 0h300 = - 1.0 (- 256/256) 0h301 = - 0.99609375 (- 255/256) ... 0h3FF = - 0.00390625 (- 1/256) 0h101 to 2FF = Reserved (don't use)</p>	0x100

16.5.4.68 HUEOFFSET

- Address = Base Address + 0x020C, Reset Value = 0x0108_0080

Name	Bit	Type	Description	Reset Value
RSVD	[31:25]	-	Reserved	0
OFFSET_IN	[24:16]	RW	Specifies the Hue matrix input offset (signed). 0h000 = + 0 0h001 = + 1 0h002 = + 2 ... 0hOFF = + 255 0h100 = - 256 ... 0x1FF = -1	0x180 (-128)
RSVD	[15:9]	-	Reserved	0
OFFSET_OUT	[8:0]	RW	Specifies the Hue matrix output offset (signed). 0h000 = + 0 0h001 = + 1 0h002 = + 2 ... 0hOFF = + 255 0h100 = - 256 ... 0x1FF = -1	0x080 (+128)

NOTE: Generally, HUE_OFFSET_IN = - 128 and HUE_OFFSET_OUT = + 128

Hue Equation:

$Cb<hue> = CBG0 \cdot (Cb + OFFSET_IN) + CBG1 \cdot (Cr + OFFSET_IN) + OFFSET_OUT$
$Cr<hue> = CRG0 \cdot (Cb + OFFSET_IN) + CRG1 \cdot (Cr + OFFSET_IN) + OFFSET_OUT$

Coefficient Decision:

$CBG0 = (Cb - 128) \geq 0 ? CBG0_P : CBG0_N$
$CBG1 = (Cr - 128) \geq 0 ? CBG1_P : CBG1_N$
$CRG0 = (Cb - 128) \geq 0 ? CRG0_P : CRG0_N$
$CRG1 = (Cr - 128) \geq 0 ? CRG1_P : CRG1_N$

16.5.4.69 VIDW0ALPHA0

- Address = Base Address + 0x021C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[24]	–	Reserved	0
ALPHA0_R_F	[23:16]	RW	Specifies the Red Alpha value (case AEN == 0).	0
ALPHA0_G_F	[15:8]	RW	Specifies the Green Alpha value (case AEN == 0).	0
ALPHA0_B_F	[7:0]	RW	Specifies the Blue Alpha value (case AEN == 0).	0

16.5.4.70 VIDW0ALPHA1

- Address = Base Address + 0x0220, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[24]	–	Reserved	0
ALPHA1_R_F	[23:16]	RW	Specifies the Red Alpha value (case AEN == 1).	0
ALPHA1_G_F	[15:8]	RW	Specifies the Green Alpha value (case AEN == 1).	0
ALPHA1_B_F	[7:0]	RW	Specifies the Blue Alpha value (case AEN == 1).	0

16.5.4.71 VIDW1ALPHA0

- Address = Base Address + 0x0224, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[24]	–	Reserved	0
RSVD	[23:20]	–	Reserved	0
ALPHA0_R_L_F	[19:16]	RW	Specifies the Red Alpha lower value (case AEN == 0).	0
RSVD	[15:12]	–	Reserved	0
ALPHA0_G_L_F	[11:8]	RW	Specifies the Green Alpha lower value (case AEN == 0).	0
RSVD	[7:4]	–	Reserved	0
ALPHA0_B_L_F	[3:0]	RW	Specifies the Blue Alpha lower value (case AEN == 0).	0

NOTE: ALPHA0_R (G, B)[7:4] = ALPHA0_R (G,B)_H[3:0]@VIDOSD1C
 ALPHA0_R (G,B)[3:0] = ALPHA0_R (G,B)_L[3:0]@VIDW1ALPHA0

16.5.4.72 VIDW0ALPHA1

- Address = Base Address + 0x0228, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[24]	–	Reserved	0
RSVD	[23:20]	–	Reserved	0
ALPHA1_R_L_F	[19:16]	RW	Specifies the Red Alpha lower value (case AEN == 1).	0
RSVD	[15:12]	–	Reserved	0
ALPHA1_G_L_F	[11: 8]	RW	Specifies the Green Alpha lower value (case AEN == 1).	0
RSVD	[7: 4]	–	Reserved	0
ALPHA1_B_L_F	[3: 0]	RW	Specifies the Blue Alpha lower value (case AEN == 1).	0

NOTE: ALPHA1_R(G,B) [7:4] = ALPHA1_R(G,B)_H[3:0]@VIDOSD1C

ALPHA1_R(G,B) [3:0] = ALPHA1_R(G,B)_L[3:0]@VIDW1ALPHA1

16.5.4.73 VIDW0ALPHA0

- Address = Base Address + 0x022C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[24]	–	Reserved	0
RSVD	[23:20]	–	Reserved	0
ALPHA0_R_L_F	[19:16]	RW	Specifies the Red Alpha lower value (case AEN == 0).	0
RSVD	[15:12]	–	Reserved	0
ALPHA0_G_L_F	[11: 8]	RW	Specifies the Green Alpha lower value (case AEN == 0).	0
RSVD	[7: 4]	–	Reserved	0
ALPHA0_B_L_F	[3: 0]	RW	Specifies the Blue Alpha lower value (case AEN == 0).	0

NOTE: ALPHA0_R (G, B)[7:4] = ALPHA0_R (G, B)_H[3:0]@VIDOSD2C

ALPHA0_R (G, B)[3:0] = ALPHA0_R (G, B)_L[3:0]@VIDW2ALPHA0

16.5.4.74 VIDW2ALPHA1

- Address = Base Address + 0x0230, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[24]	-	Reserved	0
RSVD	[23:20]	-	Reserved	0
ALPHA1_R_L_F	[19:16]	RW	Specifies the Red Alpha lower value (case AEN == 1).	0
RSVD	[15:12]	-	Reserved	0
ALPHA1_G_L_F	[11: 8]	RW	Specifies the Green Alpha lower value (case AEN == 1).	0
RSVD	[7: 4]	-	Reserved	0
ALPHA1_B_L_F	[3: 0]	RW	Specifies the Blue Alpha lower value (case AEN == 1).	0

NOTE: ALPHA1_R (G, B)[7:4] = ALPHA1_R (G, B)_H[3:0]@VIDOSD2C
 ALPHA1_R (G, B)[3:0] = ALPHA1_R (G, B)_L[3:0]@VIDW2ALPHA1

16.5.4.75 VIDW0ALPHA0

- Address = Base Address + 0x0234, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[24]	-	Reserved	0
RSVD	[23:20]	-	Reserved	0
ALPHA0_R_L_F	[19:16]	RW	Specifies the Red Alpha lower value (case AEN == 0).	0
RSVD	[15:12]	-	Reserved	0
ALPHA0_G_L_F	[11: 8]	RW	Specifies the Green Alpha lower value (case AEN == 0).	0
RSVD	[7: 4]	-	Reserved	0
ALPHA0_B_L_F	[3: 0]	RW	Specifies the Blue Alpha lower value (case AEN == 0).	0

NOTE: ALPHA0_R(G,B)[7:4] = ALPHA0_R(G,B)_H[3:0]@VIDOSD3C
 ALPHA0_R(G,B)[3:0] = ALPHA0_R(G,B)_L[3:0]@VIDW3ALPHA0

16.5.4.76 VIDW3ALPHA1

- Address = Base Address + 0x0238, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[24]	–	Reserved	0
RSVD	[23:16]	–	Reserved	0
ALPHA1_R_L_F	[19:16]	RW	Specifies the Red Alpha lower value (case AEN == 1).	0
RSVD	[15:12]	–	Reserved	0
ALPHA1_G_L_F	[11: 8]	RW	Specifies the Green Alpha lower value (case AEN == 1).	0
RSVD	[7: 4]	–	Reserved	0
ALPHA1_B_L_F	[3: 0]	RW	Specifies the Blue Alpha lower value (case AEN == 1).	0

NOTE: ALPHA1_R(G,B)[7:4] = ALPHA1_R(G,B)_H[3:0]@VIDOSD3C

ALPHA1_R(G,B)[3:0] = ALPHA1_R(G,B)_L[3:0]@VIDW3ALPHA1

16.5.4.77 VIDW4ALPHA0

- Address = Base Address + 0x023C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[24]	–	Reserved	0
RSVD	[23:20]	–	Reserved	0
ALPHA0_R_L_F	[19:16]	RW	Specifies the Red Alpha lower value (case AEN == 0).	0
RSVD	[15:12]	–	Reserved	0
ALPHA0_G_L_F	[11: 8]	RW	Specifies the Green Alpha lower value (case AEN == 0).	0
RSVD	[7: 4]	–	Reserved	0
ALPHA0_B_L_F	[3: 0]	RW	Specifies the Blue Alpha lower value (case AEN == 0).	0

NOTE: ALPHA0_R(G,B)[7:4] = ALPHA0_R(G,B)_H[3:0]@VIDOSD4C

ALPHA0_R(G,B)[3:0] = ALPHA0_R(G,B)_L[3:0]@VIDW4ALPHA0

16.5.4.78 VIDW0ALPHA1

- Address = Base Address + 0x0240, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[24]	-	Reserved	0
RSVD	[23:20]	-	Reserved	0
ALPHA1_R_L_F	[19:16]	RW	Specifies the Red Alpha lower value (case AEN == 1).	0
RSVD	[15:12]	-	Reserved	0
ALPHA1_G_L_F	[11: 8]	RW	Specifies the Green Alpha lower value (case AEN == 1).	0
RSVD	[7: 4]	-	Reserved	0
ALPHA1_B_L_F	[3: 0]	RW	Specifies the Blue Alpha lower value (case AEN == 1).	0

NOTE: ALPHA1_R(G,B)[7:4] = ALPHA1_R(G,B)_H[3:0]@VIDOSD4C

ALPHA1_R(G,B)[3:0] = ALPHA1_R(G,B)_L[3:0]@VIDW4ALPHA1

16.5.4.79 BLENDEQ1

- Address = Base Address + 0x0244, Reset Value = 0x0000_00C2

Name	Bit	Type	Description	Reset Value
RSVD	[31:22]	-	Reserved	0x000
Q_FUNC_F	[21:18]	RW	Specifies the constant used in alphaB (alpha value of *background) 0000 = 0 (zero) 0001 = 1 (maximum) 0010 = **alphaA (alpha value of *foreground) 0011 = 1 – alphaA 0100 = alphaB 0101 = 1 – alphaB 0110 = ALPHA0 0111 = Reserved 100x = Reserved 1010 = A (foreground color data) 1011 = 1 – A 1100 = B (background color data) 1101 = 1 – B 111x = Reserved	0x0
RSVD	[17:16]	-	Reserved	00
P_FUNC_F	[15:12]	RW	Specifies the constant used in alpha. Same as above (see COEF_Q).	0x0
RSVD	[11:10]	-	Reserved	00
B_FUNC_F	[9:6]	RW	Specifies the constant used in B. Same as above (see COEF_Q).	0x3
RSVD	[5:4]	-	Reserved	00
A_FUNC_F	[3:0]	RW	Specifies the constant used in A. Same as above (see COEF_Q).	0x2

For more information, refer to [Figure 16-5 Blending Equation](#).

background = Window 0, foreground = Window 1 (in Blend Equation 1)

alphaA and alphaB are decided by BPPMODE_F, BLD_PIX, ALPHA_SEL @WINCONx, and WxPAL @WPALCON.

16.5.4.80 BLENDEQ2

- Address = Base Address + 0x0248, Reset Value = 0x0000_00C2

Name	Bit	Type	Description	Reset Value
RSVD	[31:22]	—	Reserved	0x000
Q_FUNC_F	[21:18]	RW	Specifies the constant used in alphaB (alpha value of *background). 0000 = 0 (zero) 0001 = 1 (maximum) 0010 = **alphaA (alpha value of *foreground) 0011 = 1 – alphaA 0100 = alphaB 0101 = 1 – alphaB 0110 = ALPHA0 0111 = Reserved 100x = Reserved 1010 = A (foreground color data) 1011 = 1 – A 1100 = B (background color data) 1101 = 1 – B 111x = Reserved	0x0
RSVD	[17:16]	—	Reserved	00
P_FUNC_F	[15:12]	RW	Specifies the constant used in alpha. Same as above (see COEF_Q)	0x0
RSVD	[11:10]	—	Reserved	00
B_FUNC_F	[9:6]	RW	Specifies the constant used in B. Same as above (see COEF_Q)	0x3
RSVD	[5:4]	—	Reserved	00
A_FUNC_F	[3:0]	RW	Specifies the constant used in A. Same as above (see COEF_Q)	0x2

For more information, refer to [Figure 16-5 Blending Equation](#).

background = Window 01, foreground = Window 2 (in Blend Equation 2)

alphaA and alphaB are decided by BPPMODE_F, BLD_PIX, ALPHA_SEL @WINCONx, and WxPAL @WPALCON.

16.5.4.81 BLENDEQ3

- Address = Base Address + 0x024C, Reset Value = 0x0000_00C2

Name	Bit	Type	Description	Reset Value
RSVD	[31:22]	—	Reserved	0x000
Q_FUNC_F	[21:18]	RW	Specifies the constant used in alphaB (alpha value of *background) 0000 = 0 (zero) 0001 = 1 (maximum) 0010 = **alphaA (alpha value of *foreground) 0011 = 1 – alphaA 0100 = alphaB 0101 = 1 – alphaB 0110 = ALPHA0 0111 = Reserved 100x = Reserved 1010 = A (foreground color data) 1011 = 1 – A 1100 = B (background color data) 1101 = 1 – B 111x = Reserved	0x0
RSVD	[17:16]	—	Reserved	00
P_FUNC_F	[15:12]	RW	Specifies the constant used in alpha. Same as above (see COEF_Q).	0x0
RSVD	[11:10]	—	Reserved	00
B_FUNC_F	[9:6]	RW	Specifies the constant used in B. Same as above (see COEF_Q).	0x3
RSVD	[5:4]	—	Reserved	00
A_FUNC_F	[3:0]	RW	Specifies the constant used in A. Same as above (see COEF_Q).	0x2

For more information, refer to [Figure 16-5 Blending Equation](#).

background = Window 012, foreground = Window 3 (in Blend Equation 3)

alphaA and alphaB are decided by BPPMODE_F, BLD_PIX, ALPHA_SEL @WINCONx, and WxPAL @WPALCON.

16.5.4.82 BLENDEQ4

- Address = Base Address + 0x0250, Reset Value = 0x0000_00C2

Name	Bit	Type	Description	Reset Value
RSVD	[31:22]	-	Reserved	0x000
Q_FUNC_F	[21:18]	RW	Specifies the constant used in alphaB (alpha value of *background) 0000 = 0 (zero) 0001 = 1 (maximum) 0010 = **alphaA (alpha value of *foreground) 0011 = 1 – alphaA 0100 = alphaB 0101 = 1 – alphaB 0110 = ALPHA0 0111 = Reserved 100x = Reserved 1010 = A (foreground color data) 1011 = 1 – A 1100 = B (background color data) 1101 = 1 – B 111x = Reserved	0x0
RSVD	[17:16]	-	Reserved	00
P_FUNC_F	[15:12]	RW	Specifies the constant used in alpha. Same as above (see COEF_Q).	0x0
RSVD	[11:10]	-	Reserved	00
B_FUNC_F	[9:6]	RW	Specifies the constant used in B. Same as above (see COEF_Q).	0x3
RSVD	[5:4]	-	Reserved	00
A_FUNC_F	[3:0]	RW	Specifies the constant used in A. Same as above (see COEF_Q).	0x2

For more information, refer to [Figure 16-5 Blending Equation](#).

background = Window 0123, foreground = Window 4 (in Blend Equation 4)

alphaA and alphaB are decided by BPPMODE_F, BLD_PIX, ALPHA_SEL @WINCONx, and WxPAL @WPALCON.

16.5.4.83 BLENDCON

- Address = Base Address + 0x0260, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:1]	-	Reserved	0x000
BLEND_NEW	[0]	RW	Specifies the Alpha value width. 0 = 4-bit alpha value 1 = 8-bit alpha value	0x0

16.5.4.84 WnRTQOSCON (n = 0 to 4)

- Address = Base Address + 0x0264, Reset Value = 0x0000_0000
- Address = Base Address + 0x0268, Reset Value = 0x0000_0000
- Address = Base Address + 0x026C, Reset Value = 0x0000_0000
- Address = Base Address + 0x0270, Reset Value = 0x0000_0000
- Address = Base Address + 0x0274, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:12]	-	Reserved (should be 0)	0
FIFOLEVEL	[11:4]	RW	Specifies the real-time QoS FIFO level. If FIFO depth is less than FIFOLEVEL[7:0], then RTQoS output is 1.	0
RSVD	[3:2]	-	Reserved (should be 0)	0
QOS_GATE_DIS	[1]	RW	Disables the RTQoS output signal gate. 0 = Gated 1 = Not gated	0
RSVD	[0]	-	Reserved (should be 0).	0

16.5.4.85 I80IFCONn (n = 0 to 11)

- Address = Base Address + 0x0280, Reset Value = 0x0000_0000
- Address = Base Address + 0x0284, Reset Value = 0x0000_0000
- Address = Base Address + 0x0288, Reset Value = 0x0000_0000
- Address = Base Address + 0x028C, Reset Value = 0x0000_0000
- Address = Base Address + 0x0290, Reset Value = 0x0000_0000
- Address = Base Address + 0x0294, Reset Value = 0x0000_0000
- Address = Base Address + 0x0298, Reset Value = 0x0000_0000
- Address = Base Address + 0x029C, Reset Value = 0x0000_0000
- Address = Base Address + 0x02A0, Reset Value = 0x0000_0000
- Address = Base Address + 0x02A4, Reset Value = 0x0000_0000
- Address = Base Address + 0x02A8, Reset Value = 0x0000_0000
- Address = Base Address + 0x02AC, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
LDI_CMD	[23:0]	RW	Specifies the LDI command.	0

16.5.5 Palette Ram

16.5.5.1 Win0 Palette Ram Access Address (not SFR)

- Address = Base Address + 0x2400,0x0400, Reset Value = 0x0000_0000
- Address = Base Address + 0x2404,0x0404, Reset Value = 0x0000_0000
- Address = Base Address + 0x27FC,0x07FC, Reset Value = 0x0000_0000

Register	Address	Type	Description	Reset Value
00	0x0_2400 (0x0_0400)	RW	Specifies the Window 0 Palette entry 0 address.	Undefined
01	0x0_2404 (0x0_0404)	RW	Specifies the Window 0 Palette entry 1 address.	Undefined
-	-	-	-	-
FF	0x0_27FC (0x0_07FC)	RW	Specifies the Window 0 Palette entry 255 address.	Undefined

16.5.5.2 Win1 Palette Ram Access Address (not SFR)

- Address = Base Address + 0x2800,0x0800, Reset Value = 0x0000_0000
- Address = Base Address + 0x2804,0x0804, Reset Value = 0x0000_0000
- Address = Base Address + 0x2BFC,0x0BFC, Reset Value = 0x0000_0000

Register	Address	Type	Description	Reset Value
00	0x0_2800 (0x0_0800)	RW	Specifies the Window 1 Palette entry 0 address.	Undefined
01	0x0_2804 (0x0_0804)	RW	Specifies the Window 1 Palette entry 1 address.	Undefined
-	-	-	-	-
FF	0x0_2BFC (0x0_0BFC)	RW	Specifies the Window 1 Palette entry 255 address.	Undefined

16.5.5.3 Win2 Palette Ram Access Address (not SFR)

- Address = Base Address + 0x2C00,0x0C00, Reset Value = 0x0000_0000
- Address = Base Address + 0x2C04,0x0C04, Reset Value = 0x0000_0000
- Address = Base Address + 0x2FFC,0x0FFC, Reset Value = 0x0000_0000

Register	Address	Type	Description	Reset Value
00	0x0_2C00	RW	Specifies the Window 2 Palette entry 0 address.	Undefined
01	0x0_2C04	RW	Specifies the Window 2 Palette entry 1 address.	Undefined
–	–	–	–	–
FF	0x0_2FFC	RW	Specifies the Window 2 Palette entry 255 address.	Undefined

16.5.5.4 Win3 Palette Ram Access Address (not SFR)

- Address = Base Address + 0x3000, Reset Value = 0x0000_0000
- Address = Base Address + 0x3004, Reset Value = 0x0000_0000
- Address = Base Address + 0x33FC, Reset Value = 0x0000_0000

Register	Address	Type	Description	Reset Value
00	0x0_3000	RW	Specifies the Window 3 Palette entry 0 address.	Undefined
01	0x0_3004	RW	Specifies the Window 3 Palette entry 1 address.	Undefined
–	–	–	–	–
FF	0x0_33FC	RW	Specifies the Window 3 Palette entry 255 address.	Undefined

16.5.5.5 Win4 Palette Ram Access Address (not SFR)

- Address = Base Address + 0x3400, Reset Value = 0x0000_0000
- Address = Base Address + 0x3404, Reset Value = 0x0000_0000
- Address = Base Address + 0x37FC, Reset Value = 0x0000_0000

Register	Address	Type	Description	Reset Value
00	0x0_3400	RW	Specifies the Window 4 Palette entry 0 address.	Undefined
01	0x0_3404	RW	Specifies the Window 4 Palette entry 1 address.	Undefined
–	–	–	–	–
FF	0x0_37FC	RW	Specifies the Window 4 Palette entry 255 address.	Undefined

16.5.6 Gamma Lookup Table

16.5.6.1 Gamma LUT Data for 64 Step Mode

Name	Bit	Type	Description	Reset Value
GM_LUT_x	[26:18]	RW	Specifies the Gamma LUT value register of index x.	Undefined
GM_LUT_y	[10: 2]	RW	Specifies the Gamma LUT value register of index y.	Undefined

16.5.6.2 Gamma LUT Data for 16 Step Mode

Name	Bit	Type	Description	Reset Value
GM_LUT_x	[26:18]	RW	Specifies the Gamma LUT value register of index x.	Undefined
GM_LUT_y	[10: 2]	RW	Specifies the Gamma LUT value register of index y.	Undefined

16.5.7 Shadow Windows Control

16.5.7.1 SHD_VIDW0nADD0 (n = 0 to 4)

- Address = Base Address + 0x40A0, Reset Value = 0x0000_0000
- Address = Base Address + 0x40A8, Reset Value = 0x0000_0000
- Address = Base Address + 0x40B0, Reset Value = 0x0000_0000
- Address = Base Address + 0x40B8, Reset Value = 0x0000_0000
- Address = Base Address + 0x40C0, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
VBASEU_F	[31:0]	R	Specifies A[31:0] of the start address for video frame buffer (Shadow).	0

16.5.7.2 SHD_VIDW0nADD1 (n = 0 to 4)

- Address = Base Address + 0x40D0, Reset Value = 0x0000_0000
- Address = Base Address + 0x40D8, Reset Value = 0x0000_0000
- Address = Base Address + 0x40E0, Reset Value = 0x0000_0000
- Address = Base Address + 0x40E8, Reset Value = 0x0000_0000
- Address = Base Address + 0x40F0, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
VBASEL_F	[31:0]	R	Specifies A[31:0] of the end address for video buffer (Shadow).	0x0

16.5.7.3 SHD_VIDW0nADD2 (n = 0 to 4)

- Address = Base Address + 0x4100, Reset Value = 0x0000_0000
- Address = Base Address + 0x4104, Reset Value = 0x0000_0000
- Address = Base Address + 0x4108, Reset Value = 0x0000_0000
- Address = Base Address + 0x410C, Reset Value = 0x0000_0000
- Address = Base Address + 0x4110, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
OFFSIZE_F	[25:13]	R	Specifies the Virtual screen offset size that is the number of byte (Shadow).	0
PAGEWIDTH_F	[12:0]	R	Specifies the Virtual screen page width (number of byte). This value defines the width of view port in the frame (Shadow).	0

17 ADC & Touch Screen Interface

This chapter describes the functions and usage of ADC and Touch Screen interface.

17.1 Overview

The 10-bit or 12-bit CMOS Analog to Digital Converter (ADC) comprises of 10-channel analog inputs. It converts the analog input signal into 10-bit or 12-bit binary digital codes at a maximum conversion rate of 1MSPS with 5 MHz A/D converter clock. A/D converter operates with on-chip sample-and-hold function. ADC supports low power mode. Clock source is the ACLK_100 (PCLK).

Touch Screen Interface can control input pads (XP, XM, YP, and YM) to obtain X/Y-position on the external touch screen device. Touch Screen Interface contains three main blocks, namely, touch screen pads control logic, ADC interface logic and interrupt generation logic. There are two set of touch screen interfaces, which share one ADC.

17.2 Features

The ADC & Touch Screen interface includes the following features:

- Resolution: 10-bit/12-bit (optional)
- Differential Nonlinearity Error: ± 1.0 LSB (Max.)
- Integral Nonlinearity Error: ± 4.0 LSB (Max.)
- Maximum Conversion Rate: 1 MSPS
- Low Power Consumption
- Power Supply Voltage: 3.3 V
- Analog Input Range: 0 to 3.3 V
- On-chip sample-and-hold function
- Normal Conversion Mode
- Separate X/Y position conversion Mode
- Auto (Sequential) X/Y Position Conversion Mode
- Waiting for Interrupt Mode
- IDLE, DIDLE, STOP and DSTOP mode wakeup source
- Two touch screen interfaces

17.3 Functional Description

17.3.1 Block Diagram

[Figure 17-1](#) is the functional block diagram of A/D converter and Touch Screen Interface.

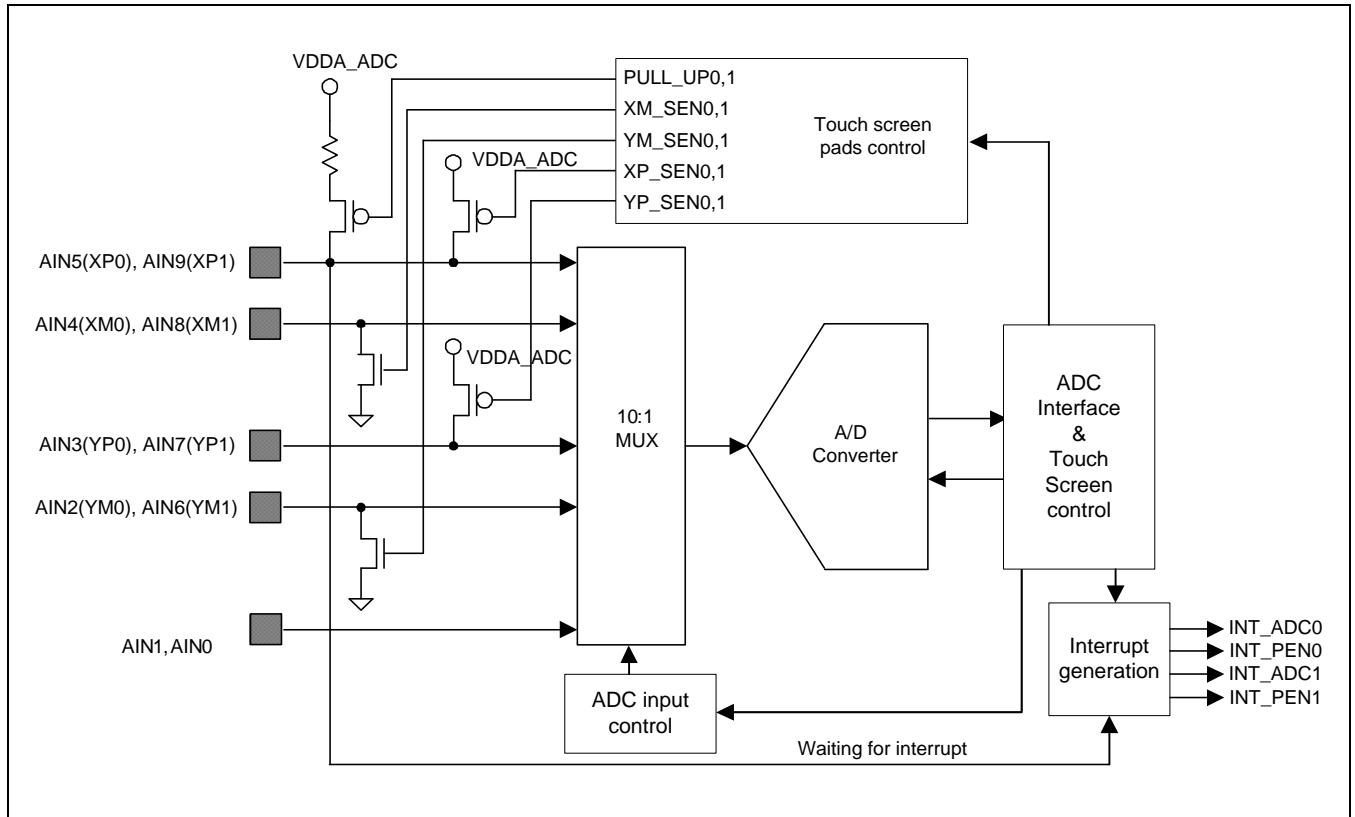


Figure 17-1 ADC and Touch Screen Interface Functional Block Diagram

NOTE: When Touch Screen device is not used, XM, XP, YM or YP can be connected to analog Input Signal for Normal ADC conversion.

17.3.2 Interface Port Description

Signal	I/O	Description	Pad	Type
AIN[9]	Input	ADC Channel[9] Analog input	XadcAIN_9	Analog
AIN[8]	Input	ADC Channel[8] Analog input	XadcAIN_8	Analog
AIN[7]	Input	ADC Channel[7] Analog input	XadcAIN_7	Analog
AIN[6]	Input	ADC Channel[6] Analog input	XadcAIN_6	Analog
AIN[5]	Input	ADC Channel[5] Analog input	XadcAIN_5	Analog
AIN[4]	Input	ADC Channel[4] Analog input	XadcAIN_4	Analog
AIN[3]	Input	ADC Channel[3] Analog input	XadcAIN_3	Analog
AIN[2]	Input	ADC Channel[2] Analog input	XadcAIN_2	Analog
AIN[1]	Input	ADC Channel[1] Analog input	XadcAIN_1	Analog
AIN[0]	Input	ADC Channel[0] Analog input	XadcAIN_0	Analog

17.3.3 A/D Conversion Time

When the PCLK frequency is 100 MHz and the prescaler value is 99, total 12-bit conversion time is as follows.

- A/D converter freq. = $100 \text{ MHz} / (19 + 1) = 5 \text{ MHz}$
- Conversion time = $1 / (5 \text{ MHz} / 5 \text{ cycles}) = 1 / 200 \text{ kHz} = 1 \text{ us}$

NOTE: This A/D converter was designed to operate at maximum 5 MHz clock, so the conversion rate can go up to 1 MSPS.

17.3.4 Touch Screen Interface Mode

17.3.4.1 Normal Conversion Mode (AUTO_PST = 0, XY_PST = 0)

The operation of this mode is same as AIN0~AIN9's. To initialize this mode, set the TSADCCON0 (ADC control register) and TSCONn (Touch screen control register). The switches and pull-up resistor should be turned off (all switches are turned off if TSCON0 and TSCON1 are set to 0x58). The converted data can be read out from TSDATX0 (ADC conversion data X register).

NOTE: TSADCCON1 register is useless in normal conversion mode. Therefore, TSSEL bit of TSADCCON0 register should be 0. TSADCCON1 register is meaningless if TSSEL bit is 0.

17.3.4.2 Separate X/Y Position Conversion Mode (AUTO_PST = 0, XY_PST = control)

This mode consists of two states, namely, X-position measurement state and Y-position measurement state. Steps to operate X-position measurement state;

- Set "0x69" to TSCONn.
(XY_PST=1, AUTO_PST=0, PULL_UP disable, XP enable, XM enable, YP disable, YM disable)
- Start conversion by setting TSADCCONn.
- The end of X-position conversion can be notified by interrupt (INT_ADCn).
- Read out the converted data (X-position) from TSDATXn.

Steps to operate Y-position measurement state;

- Set "0x9a" to TSCONn.
(XY_PST=2, AUTO_PST=0, PULL_UP disable XP disable, XM disable, YP enable, YM enable)
- Start conversion by setting TSADCCONn.
- The end of Y-position conversion can be notified by interrupt (INT_ADCn).
- Read out the converted data (Y-position) from TSDATYn.

17.3.4.2.1 Touch Screen0 pin Conditions in X/Y Position MEASUREMENT

State	XP0	XMO	YP0	YMO
TS0: X-position measurement	VDDA_ADC	VSSA_ADC	AIN3	Hi-z
TS0: Y-position measurement	AIN5	Hi-z	VDDA_ADC	VSSA_ADC

17.3.4.2.2 Touch Screen1 pin Conditions in X/Y Position MEASUREMENT

State	XP1	XM1	YP1	YM1
TS1: X-position measurement	VDDA_ADC	VSSA_ADC	AIN7	Hi-z
TS1: Y-position measurement	AIN9	Hi-z	VDDA_ADC	VSSA_ADC

17.3.4.3 Auto (Sequential) X/Y Position Conversion Mode (AUTO_PST = 1, XY_PST = 0)

Steps to operate Auto (Sequential) X/Y Position Conversion Mode:

- Set "0x5c" to TSCONn. (XY_PST=0, AUTO_PST=1, PULL_UP disable, XP disable, XM disable, YP disable, YM disable)
- Start conversion by setting TSADCCONn.
- Touch screen controller converts X-Position and writes it to TSDATXn.
- Touch screen controller converts Y-Position and writes it to TSDATYn.
- Touch screen interface generates interrupt (INT_ADCn). In other words, INT_ADCn is occurred only once, not twice.

17.3.4.4 Waiting for Interrupt Mode (TSCONn[7:0] = 0xd3)

Touch screen controller generates an interrupt signal (INT_PENn) when the stylus pen is down or up. The value of TSCONn[7:0] should be '0xd3', that is, pull-up enable, XP disable, XM disable, YP disable and YM enable. After touch screen controller generates interrupt signal (INT_PENn), waiting for interrupt Mode must be cleared (Set 0 to XY_PST).

17.3.4.4.1 Touch Screen0 Pin Conditions in Wait for Interrupt Mode

Mode	XP0	XM0	YP0	YM0
TS0: Waiting for Interrupt Mode	VDDA_ADC (Pull-up enable)	Hi-z	Hi-z	VSSA_ADC

17.3.4.4.2 Touch Screen1 Pin Conditions in Wait for Interrupt Mode

Mode	XP1	XM1	YP1	YM1
TS1: Waiting for Interrupt Mode	VDDA_ADC (Pull-up enable)	Hi-z	Hi-z	VSSA_ADC

17.3.5 Standby Mode

Standby mode is activated when TSSEL bit is "0" and STANDBY bit is "1" in TSADCCON0 register. In this mode, A/D conversion operation is halted and TSDATXn and TSDATYn registers hold their values.

17.3.6 Two Touch Screen Interfaces

There are two set of touch screen interfaces, namely, AIN[5] to AIN[2] for touch screen 0 and AIN[9] to AIN[6] for touch screen 1. There are separate switches for XP, XM, YP and YM control and separate registers to interface with two touch screens. They share one analog digital converter, so interfacing with the two touch screens should be performed in turn. TSSEL bit of TSADCCON0 register is used to select which touch screen is connected to the ADC. Therefore, you must set "1" to TSSEL before an access to TSADCCON1. Similarly, you must set "0" to TSSEL before an access to TSADCCON0.

An access to TSADCCON1 bits is prohibited when TSSEL bit is "0", and an access to TSADCCON0 bits except TSSEL is also prohibited when TSSEL bit is "1". An access to TSSEL bit is always permitted.

17.3.6.1 Programming Notes

1. The A/D converted data can be accessed by means of interrupt or polling method. With interrupt method, the overall conversion time - from A/D converter start to converted data read - may be delayed because of the return time of interrupt service routine and data access time. With polling method, to determine the read time for TSDATXn or TSDATYn register, check the TSADCCONn[15] - end of conversion flag - bit.
2. A/D conversion can be activated in different way. After TSADCCONn[1] - A/D conversion start-by-read mode is set to 1. A/D conversion starts simultaneously when converted data is read.

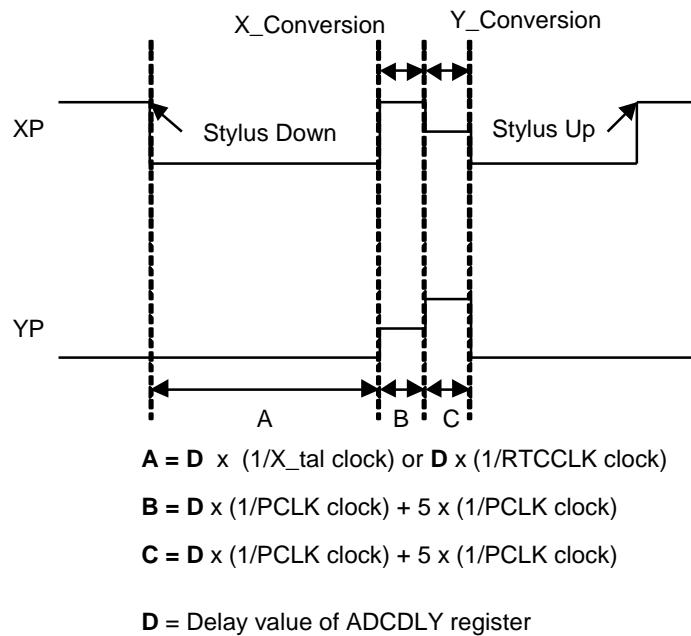


Figure 17-2 ADC and Touch Screen Operation Signal

3. If INT_PEN interrupt is used as an wakeup source in IDLE, DIDLE, STOP and DSTOP mode, XY_PST bit (TSCONn[1:0]) should be set to waiting for interrupt mode (2b'11). UD_SEN bit (TSCONn[8]) determines a stylus pen up wakeup or pen down wakeup.

17.3.7 ADC & Touch Screen Interface Input Clock Diagram

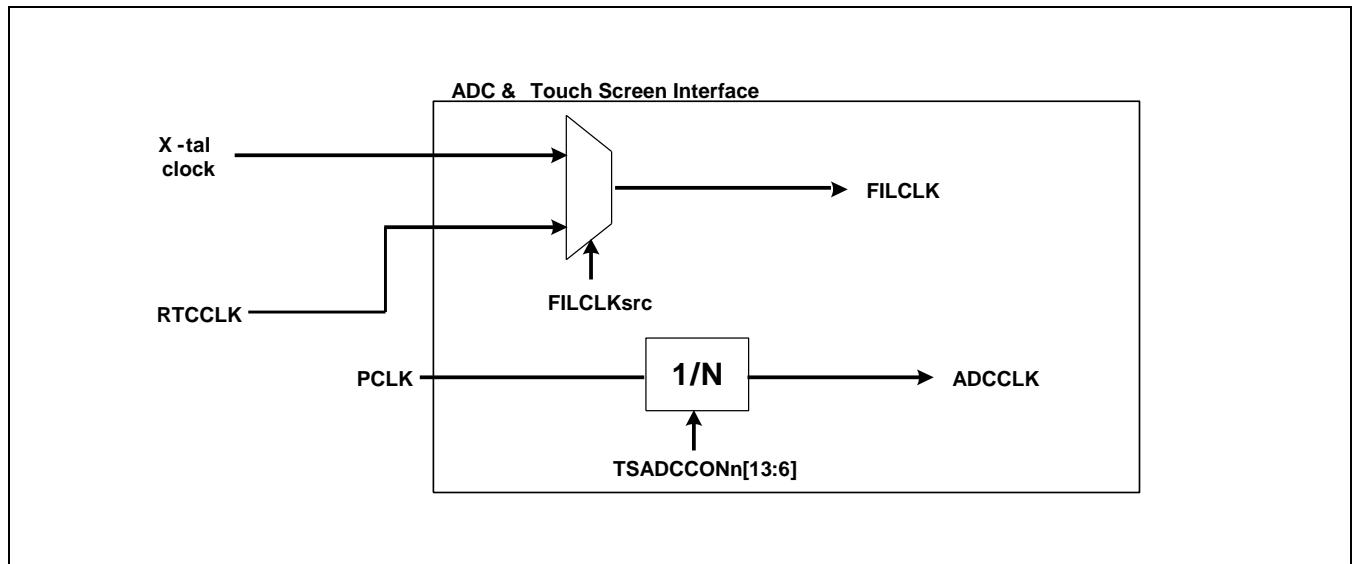


Figure 17-3 Input Clock Diagram for ADC & Touch Screen Interface

17.4 Register Description

17.4.1 Register Map Summary

- Base Address: 0x1391_0000, 0x1391_1000

Register	Offset	Description	Reset Value
TSADCCONn	0x0000	Specifies the TS0 - ADC control register	0x0000_3FC4
TSCONn	0x0004	Specifies the TS0 - Touch screen control register	0x0000_0058
TSDLYn	0x0008	Specifies the TS0 - ADC start or interval delay register	0x0000_00FF
TSDATXn	0x000C	Specifies the TS0 - ADC conversion data X register	Undefined
TSDATYn	0x0010	Specifies the TS0 - ADC Conversion Data Y Register	Undefined
TSPENSTATn	0x0014	Specifies the TS0 - Pen0 up or down status register	0x0000_0000
CLRINTADCn	0x0018	Specifies the TS0 - Clear ADC0 interrupt	Undefined
ADCMUX	0x001C	Specifies the Analog input channel selection	0x0000_0000
CLRINTPENn	0x0020	Specifies the TS0 - Clear Pen0 down/up interrupt	Undefined

17.4.1.1 TSADCCONn (n = 0 to 1)

- Address = Base Address + 0x0000, Reset Value = 0x0000_3FC4

Name	Bit	Type	Description	Reset Value
TSSEL	[17]	RW	Touch screen selection 0 = Touch screen 0 (AIN2 to AIN5) 1 = Touch screen 1 (AIN6 to AIN9) This bit exists only in TSADCCON0. NOTE: An access to TSADCCON1 bits is prohibited when TSSEL bit is 0, and an access to TSADCCON0 bits except TSSEL is prohibited when TSSEL bit is 1. An access to TSSEL bit is always permitted.	0
RES	[16]	RW	ADC output resolution selection 0 = 10-bit A/D conversion 1 = 12-bit A/D conversion	0
ECFLG	[15]	RW	End of conversion flag(Read only) 0 = A/D conversion in process 1 = End of A/D conversion	0
PRSCEN	[14]	RW	A/D converter prescaler enable 0 = Disable 1 = Enable	0
PRSCVL	[13:6]	RW	A/D converter prescaler value Data value: 19 to 255 The division factor is (N+1) when the prescaler value is N. For example, ADC frequency is 5 MHz if PCLK is 100 MHz and the prescaler value is 19. NOTE: This A/D converter is designed to operate at maximum 5 MHz clock, so the prescaler value should be set such that the resulting clock does not exceed 5 MHz.	0xFF
RSVD	[5:3]	-	Reserved	0
STANDBY	[2]	RW	Standby mode select 0 = Normal operation mode 1 = Standby mode NOTE: In standby mode, prescaler should be disabled to reduce more leakage power consumption.	1
READ_START	[1]	RW	A/D conversion start by read 0 = Disables start by read operation 1 = Enables start by read operation	0
ENABLE_START	[0]	RW	A/D conversion starts by enable. If READ_START is enabled, this value is not valid. 0 = No operation 1 = A/D conversion starts and this bit is automatically cleared after the start-up.	0

17.4.1.2 TSCONn (n = 0 to 1)

- Address = Base Address + 0x0004, Reset Value = 0x0000_0058

Name	Bit	Type	Description	Reset Value
UD_SEN	[8]	RW	Detect Pen Up or Down status. 0 = Detects pen down 1 = Detects pen up	0
YM_SEN	[7]	RW	YM to GND Switch Enable 0 = Switch disable (YM = Hi-z) 1 = Switch enable (YM = VSSA_ADC)	0
YP_SEN	[6]	RW	YP to VDD Switch Enable 0 = Switch enable (YP = VDDA_ADC) 1 = Switch disable (YP = Hi-z)	1
XM_SEN	[5]	RW	XM to GND Switch Enable 0 = Switch disable (XM = Hi-z) 1 = Switch enable (XM = VSSA_ADC)	0
XP_SEN	[4]	RW	XP to VDD Switch Enable 0 = Switch enable (XP = VDDA_ADC) 1 = Switch disable (XP = Hi-z)	1
PULL_UP	[3]	RW	Pull-up Switch Enable 0 = XP Pull-up Enable 1 = XP Pull-up Disable	1
AUTO_PST	[2]	RW	Automatic sequencing conversion of X-Position and Y-Position 0 = Normal ADC conversion. 1 = Auto Sequential measurement of X-position, Y-position.	0
XY_PST	[1:0]	RW	Manually measurement of X-Position or Y-Position. 00 = No operation mode 01 = X-position measurement 10 = Y-position measurement 11 = Waiting for Interrupt Mode	0

NOTE:

- While waiting for touch screen Interrupt, XP_SEN bit must be set to "1", namely 'XP output disable' and PULL_UP bit must be set to "0", namely "XP pull-up enable".
- AUTO_PST bit should be set "1" only in Automatic & Sequential X/Y Position conversion.
- If the touch screen is not connected or AP is in SLEEP mode, PULL_UP bit should be set "0" (Pull-up Enable).

- Touch screen0 pin conditions in X/Y position conversion.

	XP0	XM0	YP0	YM0	ADC ch. select
TS0: X Position	Vref	GND	AIN[3]	Hi-Z	YP0
TS0: Y Position	AIN[5]	Hi-Z	Vref	GND	XP0

- Touch screen1 pin conditions in X/Y position conversion.

	XP1	XM1	YP1	YM1	ADC ch. select
TS1: X Position	Vref	GND	AIN[7]	Hi-Z	YP1
TS1: Y Position	AIN[9]	Hi-Z	Vref	GND	XP1

17.4.1.3 TSDLYn (n = 0 to 1)

- Address = Base Address + 0x0008, Reset Value = 0x0000_00FF

Name	Bit	Type	Description	Reset Value
FILCLKsrc	[16]	RW	Reference clock source for delay. 0 = X-tal clock. 1 = RTC clock.	0
DELAY	[15:0]	RW	In case of ADC conversion mode (Normal, Separate, Auto conversion); ADC conversion is delayed by counting this value. Counting clock is PCLK. → ADC conversion delay value. In case of waiting for Interrupt mode: When stylus down occurs in waiting for interrupt mode, it generates interrupt signal (INT_PENn) at interval of several ms for Auto X/Y position conversion. If this interrupt occurs in STOP mode, it generates Wake-Up signal, having interval (several ms), for Exiting STOP MODE. NOTE: Do not use zero value(0x0000)	00ff

Before ADC conversion, Touch screen uses X-tal clock.

During ADC conversion PCLK is used.

17.4.1.4 TSDATXn (n = 0 to 1)

- Address = Base Address + 0x000C, Reset Value = Undefined

Name	Bit	Type	Description	Reset Value
UPDOWN	[15]	R	Up or Down state of stylus pen at Waiting for Interrupt Mode 0 = Pen down state. 1 = Pen up state.	—
AUTO_PST_VAL	[14]	R	Monitoring value of AUTO_PST field in TSCONn register. Read only. 0 = Normal ADC conversion. 1 = Sequencing measurement of X-position, Y-position.	—
XY_PST_VAL	[13:12]	R	Monitoring value of XY_PST field in TSCONn register. Read only. 00 = No operation mode 01 = X-position measurement 10 = Y-position measurement 11 = Waiting for Interrupt Mode	—
XPDATA (Normal ADC)	[11:0]	R	X-Position conversion data value (includes normal ADC conversion data value) Data value: 0x0 to 0xFFFF	—

17.4.1.5 TSDATYn (n = 0 to 1)

- Address = Base Address + 0x0010, Reset Value = Undefined

Name	Bit	Type	Description	Reset Value
UPDOWN	[15]	R	Up or Down state of stylus pen at Waiting for Interrupt Mode 0 = Pen down state. 1 = Pen up state.	—
AUTO_PST_VAL	[14]	R	Monitoring value of AUTO_PST field in TSCONn register. Read only. 0 = Normal ADC conversion. 1 = Sequencing measurement of X-position, Y-position.	—
XY_PST_VAL	[13:12]	R	Monitoring value of XY_PST field in TSCONn register. Read only. 00 = No operation mode 01 = X-position measurement 10 = Y-position measurement 11 = Waiting for Interrupt Mode	—
YPDATA	[11:0]	R	Y-Position conversion data value Data value: 0x0 to 0xFFFF	—

17.4.1.6 TSPENSTATn (n = 0 to 1)

- Address = Base Address + 0x0014, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
TSC_UP	[1]	RW	Pen up interrupt history. (after check, this bit should be cleared manually) 0 = No pen up state. 1 = Pen up interrupt has been occurred.	0
TSC_DN	[0]	RW	Pen down interrupt history. (after check, this bit should be cleared manually) 0 = No pen down state. 1 = Pen down interrupt has been occurred.	0

17.4.1.7 CLRINTADCn (n = 0 to 1)

- Address = Base Address + 0x0018, Reset Value = Undefined

Name	Bit	Type	Description	Reset Value
INTADCCLR	[0]	W	INT_ADCn interrupt clear. Cleared if any value is written.	–

These registers are used to clear the interrupts. Interrupt service routine is responsible to clear interrupts after the interrupt service is completed. Writing any values on this register will clear up the relevant interrupts asserted. When it is read, undefined value will be returned.

17.4.1.8 ADCMUX

- Address = Base Address + 0x001C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
SEL_MUX	[3:0]	RW	Analog input channel select 0000 = AIN 0 0001 = AIN 1 0010 = AIN 2 (YM0) 0011 = AIN 3 (YP0) 0100 = AIN 4 (XM0) 0101 = AIN 5 (XP0) 0110 = AIN 6 (YM1) 0111 = AIN 7 (YP1) 1000 = AIN 8 (XM1) 1001 = AIN 9 (XP1)	0

NOTE:

- When touch screen is not used, the touch screen ports (AIN2 to AIN9) can be used as analog input ports for ADC.
- SEL_MUX value is invalid when TSADC is set as 1) separate X/Y position conversion mode or 2) auto (sequential) X/Y position conversion mode.

17.4.1.9 CLRINTPENn (n = 0 to 1)

- Address = Base Address + 0x0020, Reset Value = Undefined

Name	Bit	Type	Description	Reset Value
INTPENCLR	[0]	W	INT_PENn interrupt clear. Cleared if any value is written.	-

18 Keypad Interface

18.1 Overview

The Key Pad Interface block in chip facilitates communication with external keypad devices. The ports multiplexed with GPIO ports provide up to 14 rows and 8 columns. You can use keypad interface on port0 or port1. Port 0 and port 1 has the same function. User can use any port for their GPIO connection. The column of port 0 is using alive power, so it can be used wakeup source without any setting. But the column of port 1 is using normal power, so it can be used wakeup source with GPIO setting for retention. The events of key press or key release are delivered to the CPU by an interrupt. You can select software scan or hardware scan. In software scan mode, if one of the interrupt from row lines occurs, the software must scan the column lines using the proper procedure to detect one or multiple key press or release. In hardware scan mode, if one of the key pressed, the hardware reports the pressed row and column number after it scan the column line automatically. Multiple key press support in hardware scan mode is limited to dual key with other row. Clock source is the ACLK_100 (PCLK).

It provides interrupt status register bits at the time of key pressed or key released or both cases (when two interrupt conditions are enabled). To prevent the switching noises, keypad interface comprise of internal debouncing filter.

18.2 Features

- Max 14 rows and 8 columns key matrix
- Key press or key release event
- Software scan/hardware scan
- Internal key switching noise filter
- Wakeup source

18.3 Functional Description

18.3.1 Interface Port Description

Table 18-1 Keypad Interface I/O Description

Signal	I/O	Description	Pad		Type
			Port0	Port1	
KP_ROW[13]	I	KEYPAD Interface Row[13] Data	XEINT_29 (GPX3[5])	XEINT_29 (GPX3[5])	muxed
KP_ROW[12]	I	KEYPAD Interface Row[12] Data	XEINT_28 (GPX3[4])	XEINT_28 (GPX3[4])	muxed
KP_ROW[11]	I	KEYPAD Interface Row[11] Data	XEINT_27 (GPX3[3])	XEINT_27 (GPX3[3])	muxed
KP_ROW[10]	I	KEYPAD Interface Row[10] Data	XEINT_26 (GPX3[2])	XEINT_26 (GPX3[2])	muxed
KP_ROW[9]	I	KEYPAD Interface Row[9] Data	XEINT_25 (GPX3[1])	XEINT_25 (GPX3[1])	muxed
KP_ROW[8]	I	KEYPAD Interface Row[8] Data	XEINT_24 (GPX3[0])	XEINT_24 (GPX3[0])	muxed
KP_ROW[7]	I	KEYPAD Interface Row[7] Data	XEINT_23 (GPX2[7])	XEINT_23 (GPX2[7])	muxed
KP_ROW[6]	I	KEYPAD Interface Row[6] Data	XEINT_22 (GPX2[6])	XEINT_22 (GPX2[6])	muxed
KP_ROW[5]	I	KEYPAD Interface Row[5] Data	XEINT_21 (GPX2[5])	XEINT_21 (GPX2[5])	muxed
KP_ROW[4]	I	KEYPAD Interface Row[4] Data	XEINT_20 (GPX2[4])	XEINT_20 (GPX2[4])	muxed
KP_ROW[3]	I	KEYPAD Interface Row[3] Data	XEINT_19 (GPX2[3])	XEINT_19 (GPX2[3])	muxed
KP_ROW[2]	I	KEYPAD Interface Row[2] Data	XEINT_18 (GPX2[2])	XEINT_18 (GPX2[2])	muxed
KP_ROW[1]	I	KEYPAD Interface Row[1] Data	XEINT_17 (GPX2[1])	XEINT_17 (GPX2[1])	muxed
KP_ROW[0]	I	KEYPAD Interface Row[0] Data	XEINT_16 (GPX2[0])	XEINT_16 (GPX2[0])	muxed
KP_COL[7]	O	KEYPAD Interface Column[7] Data	XEINT_15 (GPX1[7])	XGNSS_GPIO_7 (GPL2[7])	muxed
KP_COL[6]	O	KEYPAD Interface Column[6] Data	XEINT_14 (GPX1[6])	XGNSS_GPIO_6 (GPL2[6])	muxed
KP_COL[5]	O	KEYPAD Interface Column[5] Data	XEINT_13 (GPX1[5])	XGNSS_GPIO_5 (GPL2[5])	muxed
KP_COL[4]	O	KEYPAD Interface Column[4] Data	XEINT_12	XGNSS_GPIO_4	muxed

Signal	I/O	Description	Pad		Type
			Port0	Port1	
			(GPX1[4])	(GPL2[4])	
KP_COL[3]	O	KEYPAD Interface Column[3] Data	XEINT_11 (GPX1[3])	XGNSS_GPIO_3 (GPL2[3])	muxed
KP_COL[2]	O	KEYPAD Interface Column[2] Data	XEINT_10 (GPX1[2])	XGNSS_GPIO_2 (GPL2[2])	muxed
KP_COL[1]	O	KEYPAD Interface Column[1] Data	XEINT_9 (GPX1[1])	XGNSS_GPIO_1 (GPL2[1])	muxed
KP_COL[0]	O	KEYPAD Interface Column[0] Data	XEINT_8 (GPX1[0])	XGNSS_GPIO_0 (GPL2[0])	muxed

18.3.2 Key Matrix Interface External Connection Guide

An example of external key matrix interface is as shown in [Figure 18-1](#).

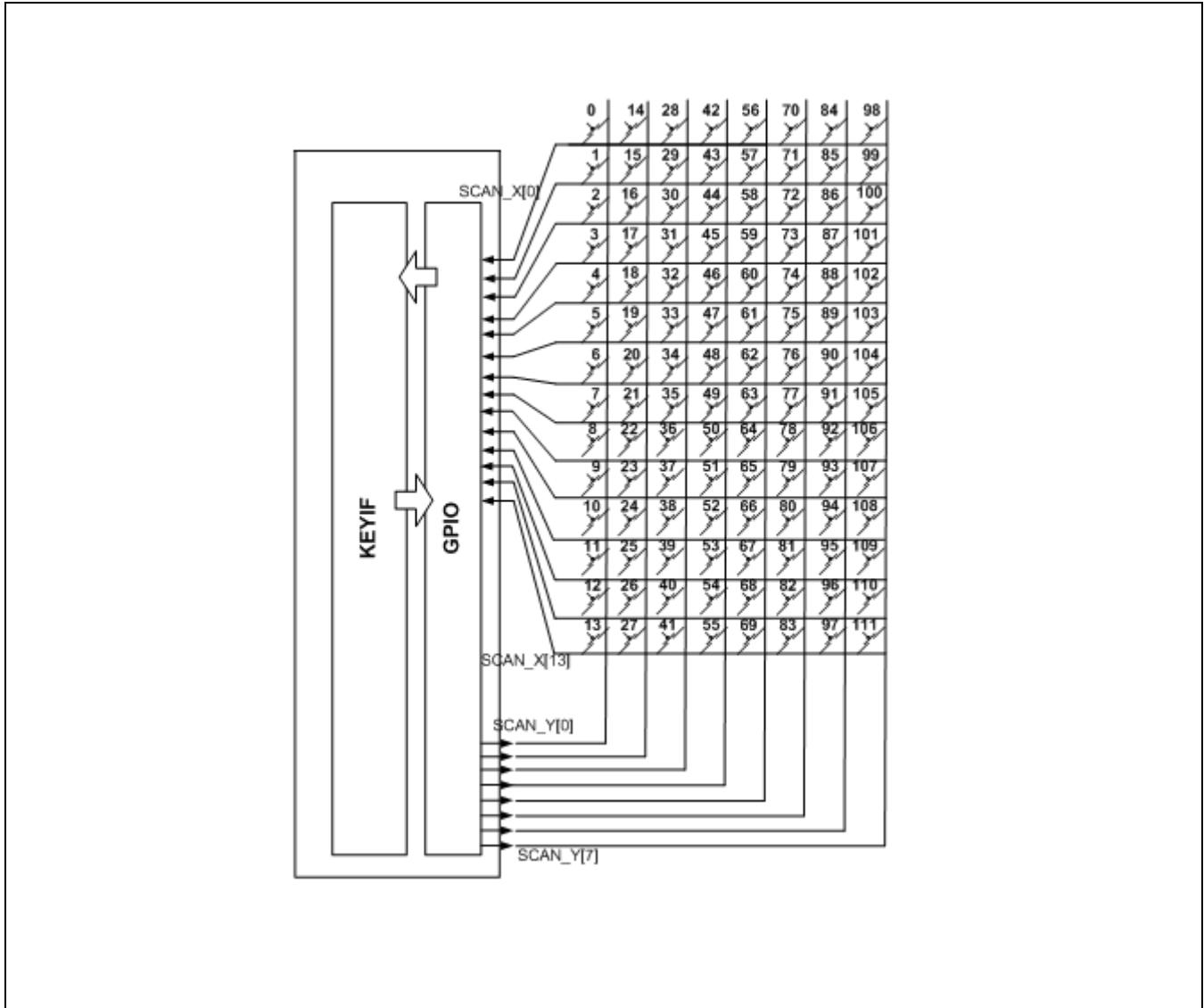


Figure 18-1 Key Matrix Interface External Connection Guide

18.3.3 Debouncing Filter

The debouncing filter is supported for keypad interrupt of any key input. The filtering width is approximately 62.5 usec ("FCLK" two-clock, when the FCLK is 32 kHz). The keypad interrupt (key pressed or key released) to the CPU in software scan mode is an ANDed signal of the all row input lines after filtering.

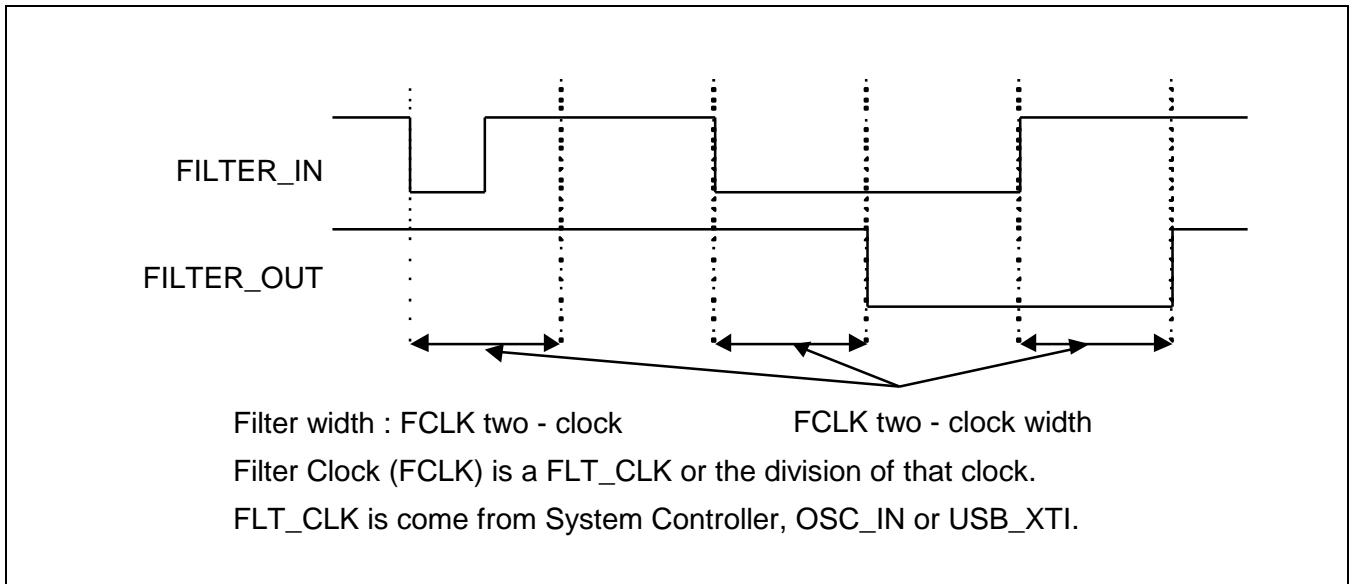


Figure 18-2 Internal Debouncing Filter Operation

18.3.4 Filter Clock

KEYPAD interface debouncing filter clock (FCLK) is divided from FLT_CLK, that is OSC_IN. User can set compare value for 10-bit up-counter (KEYIFFC). When filter enable bit (FC_EN) is HIGH, filter clock divider is ON. The frequency of FCLK is frequency of $\text{FLT_CLK}/(\text{KEYIFFC} + 1) \times 2$. On the contrary, if FC_EN is Low, filter clock divider does not divide FLT_CLK.

18.3.5 Wakeup Source

KEYPAD inputs can be used as a wakeup source. When the Key input is used for wakeup source from Audio playback, STOP, DSTOP or SLEEP mode, KEYPAD interface register setting is not required. GPIO register setting (GPX2CON, GPX3CON, GPX1CON or GPL2CON) for KEYPAD interface and SYSCON register for masking are required for wakeup.

18.3.6 Keypad Scanning Procedure for software scan

At initial state, all column lines (outputs) are low level. But column data output tri-state enable bits are all high, so, when the tri-state enable mode is not used, these bits should be written to zeros. If state is no key pressed, all row lines (inputs) are high (used pull-up pads). If any key is pressed, the corresponding row and column lines are shortened together and a low level is driven on the corresponding row line, generating a keypad interrupt. The CPU (software) outputs a LOW on one column line and Hi-Z on the others by setting KEYIFCOLEN and KEYIFCOL fields in KEYIFCOL register. Each write time, the CPU reads the value of the KEYIFROW register and detects if one key of the corresponding column line is pressed. Because the KEYIF has pull-up PAD, each KEYIFROW bits will be read as HIGH, except pressed ROW bit. When the scanning procedure ends, the pressed key (one or more) can be detected.

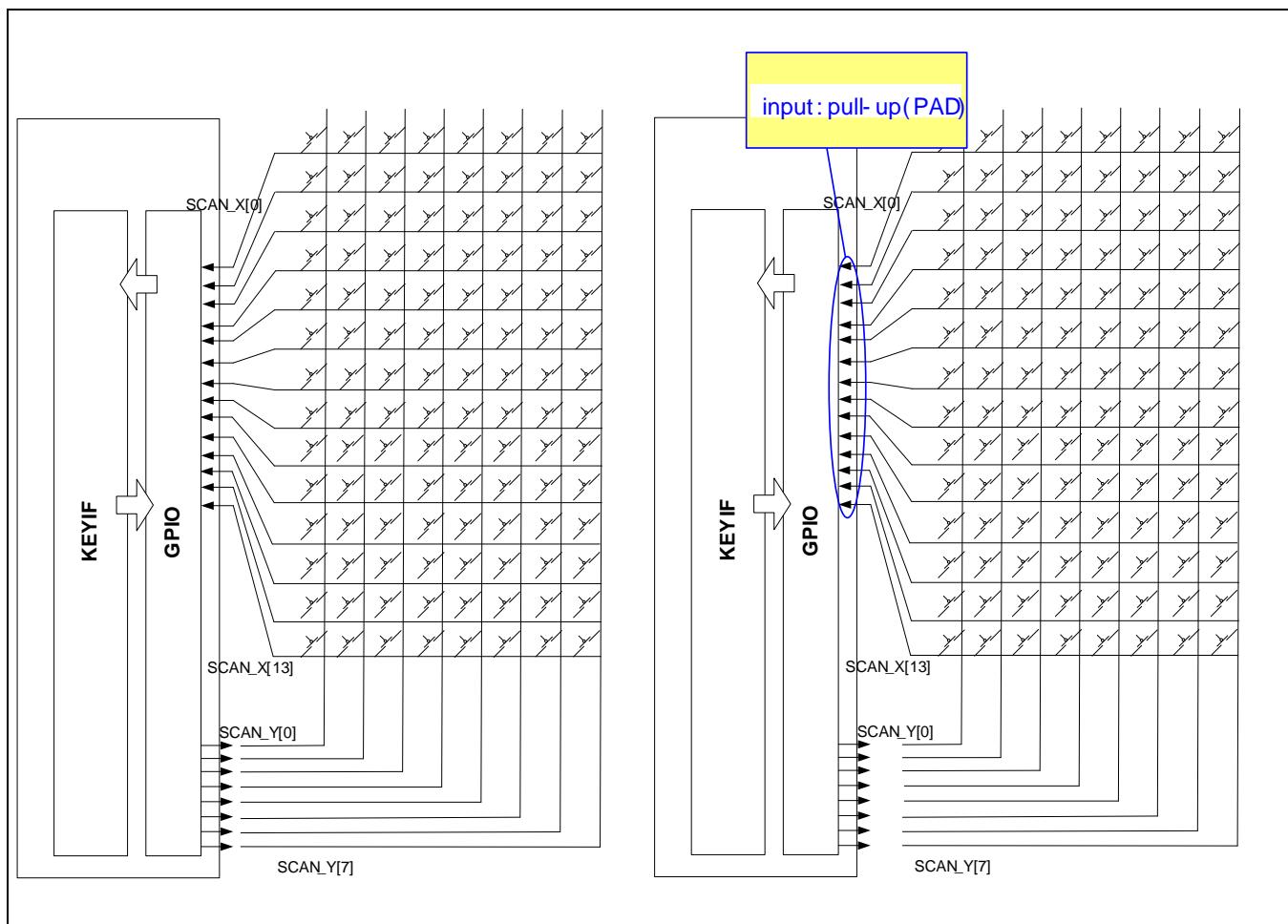


Figure 18-3 Keypad Scanning Procedure

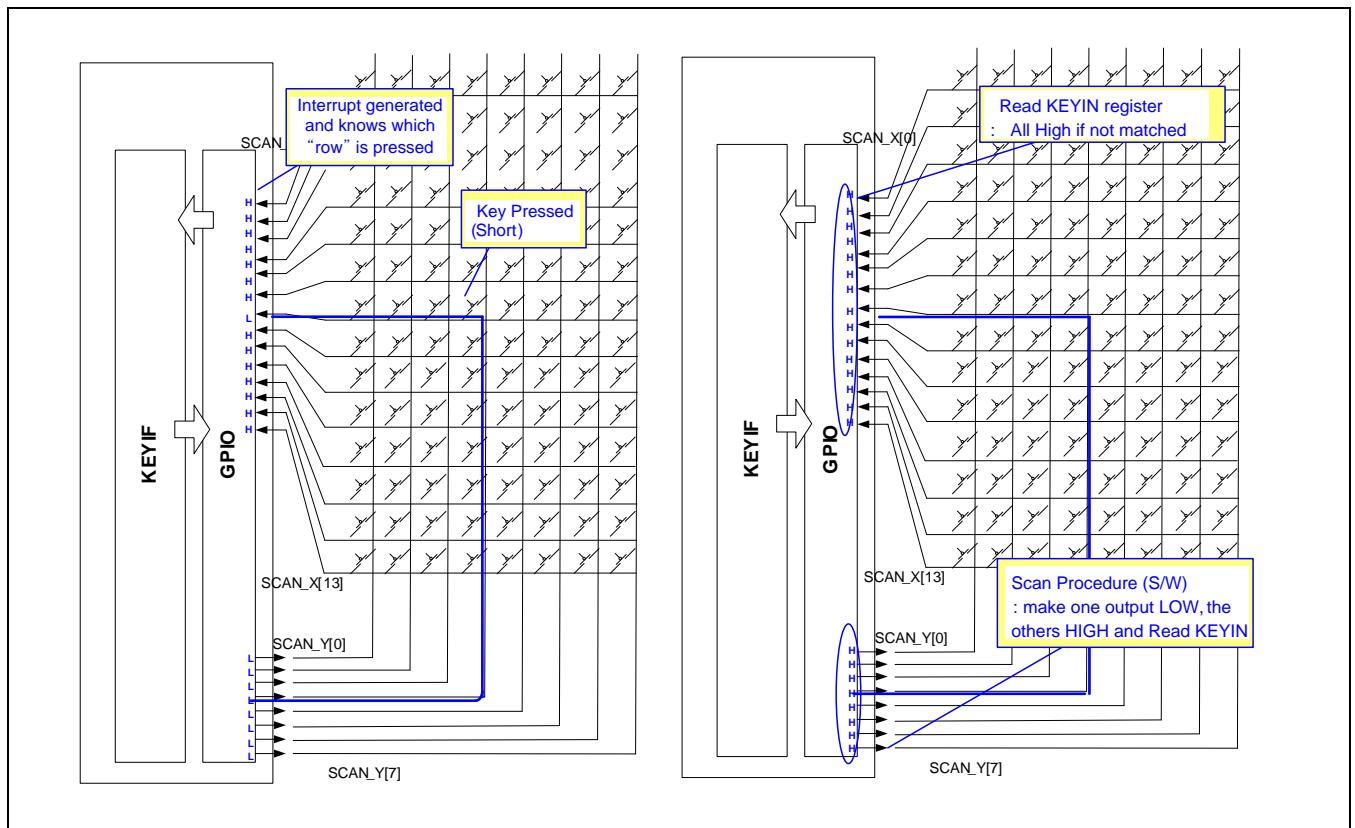


Figure 18-4 Keypad Scanning Procedure II

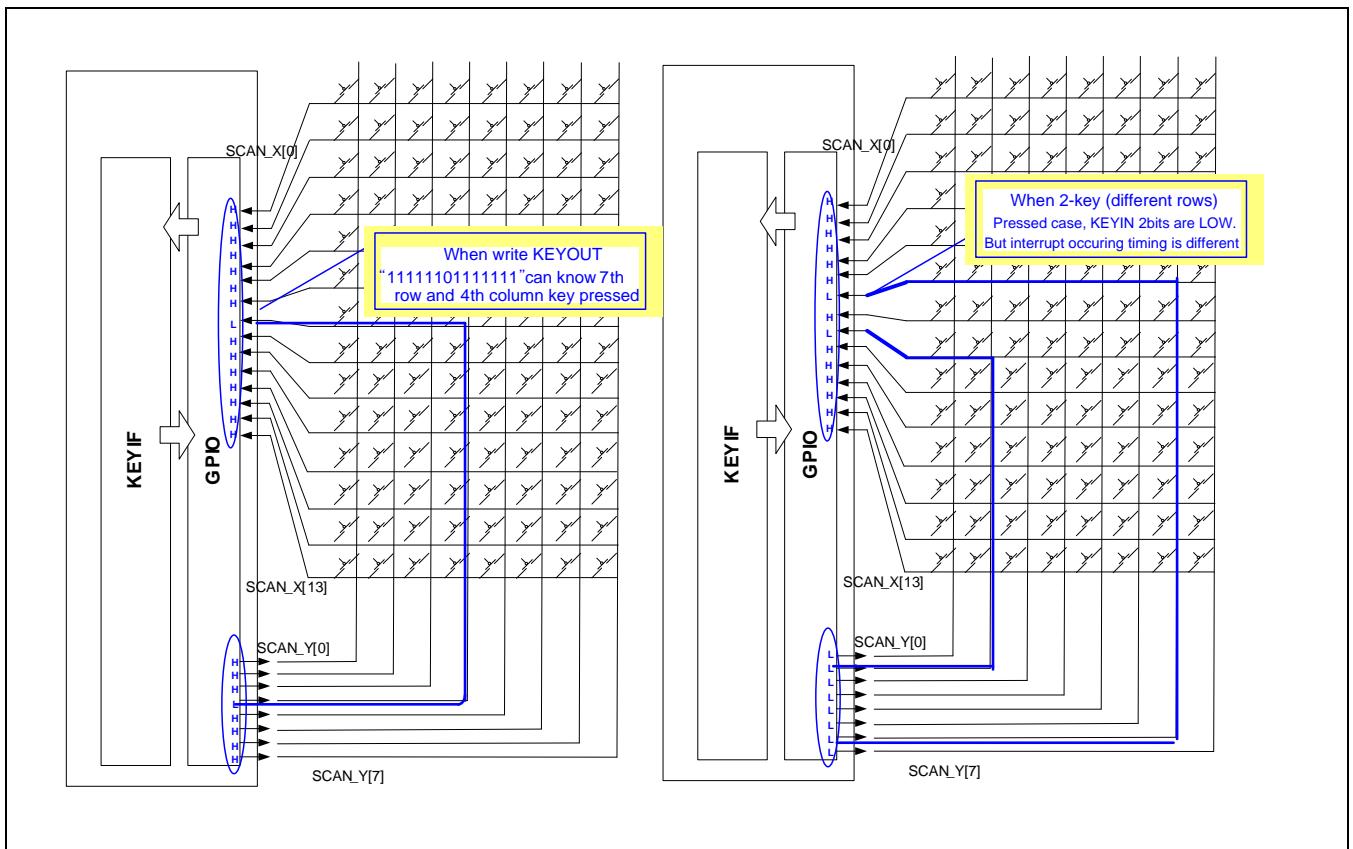


Figure 18-5 Keypad Scanning Procedure III

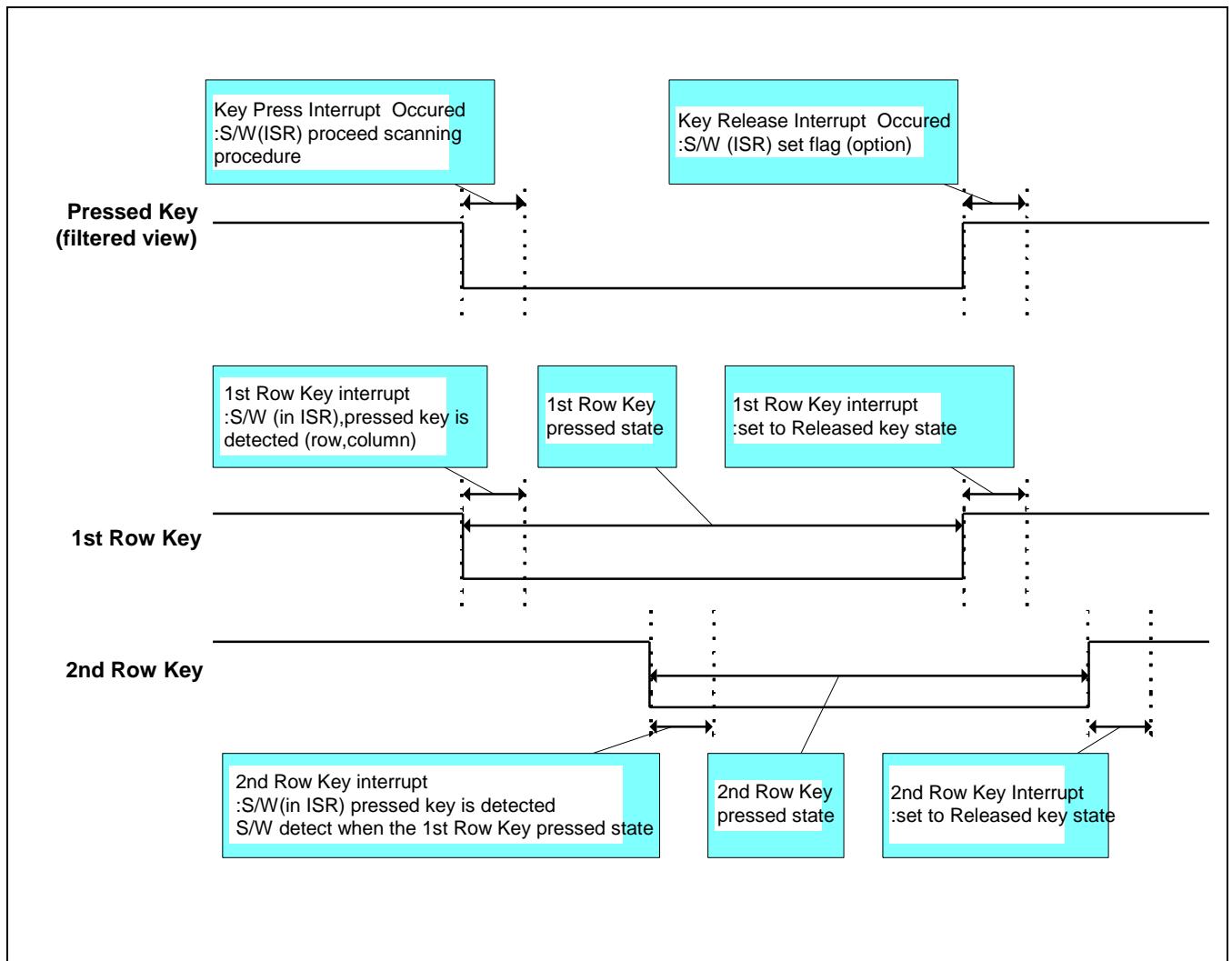


Figure 18-6 Keypad Scanning Procedure when the Two-key Pressed with Different Row

18.3.7 Keypad Scanning Procedure for Hardware scan

At initial state, it is the same as software scan. If any key is pressed, the hardware automatically scanning the corresponding row and column lines and write the information to the register. After scan and write to the register, keypad interrupt is generated. The CPU (software) can get the row and column number by accessing the KEYIFSCAN1 (first key) or KEYIFSCAN2 (second key) register. The value of KEYIFSCAN1 and KEYIFSCAN2 is valid while the key is pressed. At hardware scan mode, H_CNT value in KEYIFCON register must be set. The initial value is 0xF. Scanning hardware check the row input signal after the H_CNT cycle from driving the column.

Multiple key press support in hardware scan mode is limited to dual key with other row.

18.4 Register Description

18.4.1 Register Map Summary

- Base Address: 0x100A_0000

Register	Offset	Description	Reset Value
KEYIFCON	0x0000	Specifies the KEYPAD interface control register	0x000F_0000
KEYIFSTSCLR	0x0004	Specifies the KEYPAD interrupt for software scan status and clear register	0x0000_0000
KEYIFCOL	0x0008	Specifies the KEYPAD interface column data output register	0x0000_FF00
KEYIFROW	0x000C	Specifies the KEYPAD interface row data input register	Reflects input ports
KEYIFFC	0x0010	Specifies the KEYPAD interface debouncing filter clock division register	0x0000_0000
KEYIFSCAN1	0x0014	Specifies the KEYPAD interface output result of hardware scan for first key register	0x0000_0000
KEYIFSCAN2	0x0018	Specifies the KEYPAD interface output result of hardware scan for second key register	0x0000_0000
KEYIFHSC	0x001C	Specifies the KEYPAD interrupt for hardware scan status and clear register	0x0000_0000

18.4.1.1 KEYIFCON

- Address = Base Address + 0x0000, Reset Value = 0x000F_0000

Name	Bit	Type	Description	Reset Value
H_CNT	[31:16]	RW	Counter value for HW scan col to row interval	16'hF
RSVD	[15:10]	-	Reserved	-
HIZSCAN_EN	[9]	RW	Hi-Z mode scan enable for hardware scan 0 = normal scan (driving "low and high") 1 = Hi-Z mode scan (driving "low and Hi-Z") In Hi-Z mode GPIO internal pull-down must be disabled.	1'b0
SEL_HSCAN	[8]	RW	Select hardware scan / software scan 0 = Software scan 1 = Hardware scan	1'b0
RSVD	[7:4]	-	Reserved	-
FC_EN	[3]	RW	10-bit counter (for debouncing digital filter clock) enable 0 = Disables: No use division counter 1 = Enables: Use division counter	1'b0
DF_EN	[2]	RW	KEYPAD input port debouncing filter enable 0 = Disables 1 = Enables	1'b0
INT_R_EN	[1]	RW	KEYPAD input port rising edge (key-released) interrupt 0 = Disables 1 = Enables	1'b0
INT_F_EN	[0]	RW	KEYPAD input port falling edge (key-pressed) interrupt 0 = Disables 1 = Enables	1'b0

NOTE: Both edge interrupt is selected when both INT_F_EN and INT_R_EN are set.

18.4.1.2 KEYIFSTSCLR

- Address = Base Address+ 0x0004, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
R_INT	[29:16]	RW	<p>KEYPAD input "release" interrupt (rising edge) status(read) and clear(write)</p> <p>Read: 0 = Not occurred 1 = Released interrupt occurred</p> <p>Write: Released interrupt is cleared when write "1"</p> <p>The R_INT[13:0] indicate that each key pressed from 0 to 13 has a dedicated interrupt from R_INT[16] to R_INT[29]</p>	14'b0
P_INT	[13:0]	RW	<p>KEYPAD input "press" interrupt (falling edge) status(read) and clear(write)</p> <p>Read: 0 = Not occurred 1 = Pressed interrupt occurred</p> <p>Write: Pressed interrupt is cleared when write "1"</p> <p>The P_INT[13:0] indicate that each key released from 0 to 13 has a dedicated interrupt from P_INT[0] to P_INT[13]</p>	14'b0

NOTE: Keypad wakeup interrupt is also cleared when the write access to the KEYIFSTSCLR.

18.4.1.3 KEYIFCOL

- Address = Base Address +0x0008, Reset Value = 0x0000_FF00

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	—	Reserved	—
KEYIFCOLEN	[15:8]	RW	KEYPAD interface column data output tri-state enable register Each bit is for each KEYIFCOL bit. 0 = Output pad tri-state buffer enable (Normal output, KEY enable) 1 = Output pad Tri-state buffer disable (High-Z output, KEY disable)	8'b1111_1111
KEYIFCOL	[7:0]	RW	KEYPAD interface column data output register	8'b0

18.4.1.4 KEYIFROW

- Address : Base Address +0x000C, Reset Value = Reflects input ports

Name	Bit	Type	Description	Reset Value
RSVD	[31:16]	—	Reserved	—
KEYIFROW	[13:0]	R	KEYPAD interface row data input register (read only) This register values from input ports are not filtered data.	Reflects input ports

18.4.1.5 KEYIFFC

- Address = Base Address+ 0x0010, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:10]	—	Reserved	—
KEYIFFC	[9:0]	RW	KEYPAD interface debouncing filter clock division register. User can set compare value for 10-bit up-counter. This register value means when FC_EN bit is HIGH. $FCLK = FLT_CLK / (KEYIFFC[9:0] + 1)$ (FLT_CLK is from OSC_IN)	10'b0

18.4.1.6 KEYIFSCAN1

- Address = Base Address+ 0x0014, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:22]	—	Reserved	—
ROWSCAN1	[21:8]	R	KEYPAD interface Scan result of Row (Only pressed row has "1") Value is cleared when first key is released	14'b0
COLSCAN1	[7:0]	R	KEYPAD interface Scan result of column (Only pressed column has "1") Value is cleared when first key is released	8'b0

18.4.1.7 KEYIFSCAN2

- Address = Base Address+ 0x0018, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:22]	—	Reserved	—
ROWSCAN2	[21:8]	R	KEYPAD interface Scan result of Row (Only pressed row has "1") Value is cleared when first key is released	14'b0
COLSCAN2	[7:0]	R	KEYPAD interface Scan result of column (Only pressed column has "1") Value is cleared when first key is released	8'b0

18.4.1.8 KEYIFHSC

- Address = Base Address+ 0x001C, Reset Value = 0x0000_0000

Name	Bit	Type	Description	Reset Value
RSVD	[31:4]	-	Reserved	-
HSCAN_R2	[3]	R	KEYPAD input "release" interrupt (rising edge) status (read) and clear (write) for HW scan of second Key Read: 0 = Not occurred 1 = Released interrupt occurred Write: Released interrupt is cleared when write "1"	1'b0
HSCAN_P2	[2]	R	KEYPAD input "press" interrupt (falling edge) status (read) and clear (write) for HW scan of second Key Read: 0 = Not occurred 1 = Pressed interrupt occurred Write: Pressed interrupt is cleared when write "1"	1'b0
HSCAN_R1	[1]	R	KEYPAD input "release" interrupt (rising edge) status (read) and clear (write) for HW scan of first Key Read: 0 = Not occurred 1 = Released interrupt occurred Write: Released interrupt is cleared when write "1"	1'b0
HSCAN_P1	[0]	R	KEYPAD input "press" interrupt (falling edge) status (read) and clear (write) for HW scan of first Key Read: 0 = Not occurred 1 = Pressed interrupt occurred Write: Pressed interrupt is cleared when write "1"	1'b0