MT7621 Overview

The MT7621 SoC includes a high performance 880 MHz MIPS1004Kc CPU core and high speed USB3.0/PCIe/SDXC interfaces, which is designed to enable a multitude of high performance, cost-effective IEEE 802.11n/ac applications with a MediaTek (Ralink) WiFi client card.

Functional Block Diagram

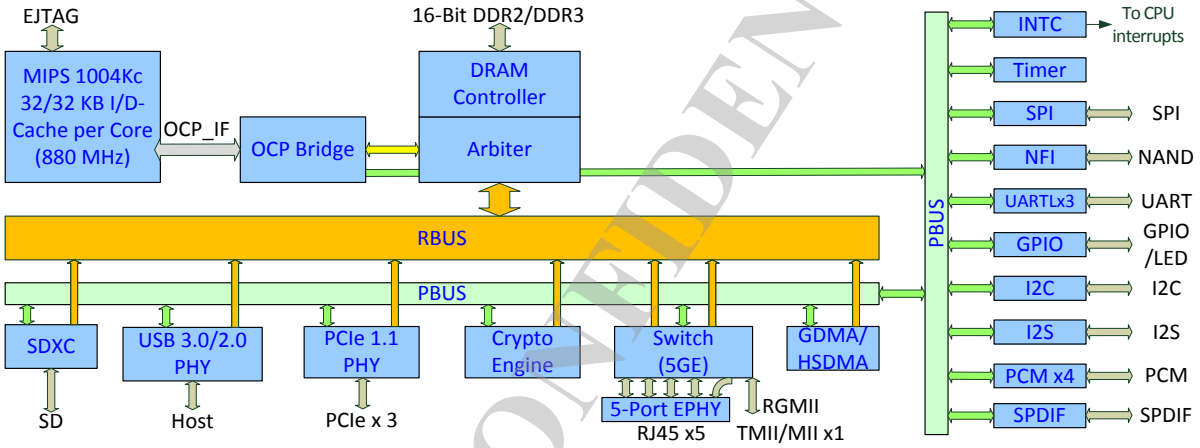


Figure 1-1 MT7621 Block Diagram

There are several masters (MIPS 1004KEc, USB, PCI Express, SDXC, FE) in the MT7621 SoC on a high performance, low latency Rbus, (Ralink Bus). In addition, the MT7621 SoC supports lower speed peripherals such as UART Lite, GPIO, NFI and SPI via a low speed peripheral bus (Pbus). The DDR2/DDR3 controller is the only bus slave on the Rbus. It includes an Advanced Memory Scheduler to arbitrate the requests from bus masters, enhancing the performance of memory access intensive tasks.

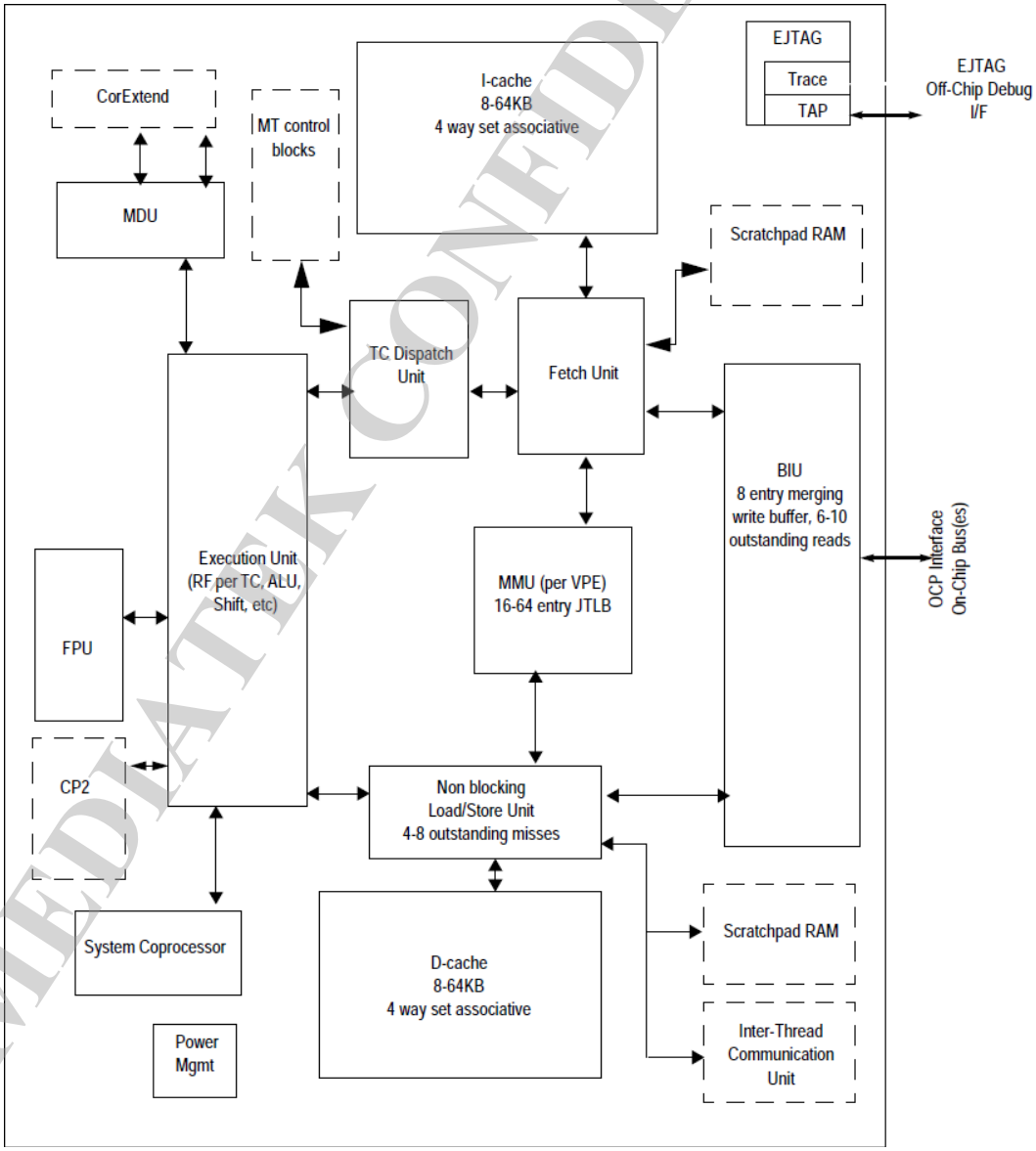
Table of Contents

1. MIPS 1004Kc Processor

1.1 Features

- 8-9-stage pipeline
- 32-bit Address Paths
- 64-bit Data Paths to Caches
- MIPS32 Enhanced Architecture (Release 2) Features
  - Standardized Instruction Set Architecture
  - Vectored interrupts and support for an external interrupt controller
  - Programmable exception vector base
  - Atomic interrupt enable/disable
  - Bit field manipulation instructions
- MIPS16e Application Specific Extension
  - 16 bit encodings of 32-bit instructions to improve code density
  - Special PC-relative instructions for efficient loading of addresses and constants
  - Data type conversion instructions (ZEB, SEB, ZEH, SEH)
  - Compact jumps (JRC, JALRC)
  - Stack frame set-up and tear down “macro” instructions (SAVE and RESTORE)
- MIPS MT Application Specific Extension (ASE)
  - Support for 2 Virtual Processing Elements (VPEs) per CORE
  - One Thread Context (TC) per VPE
- Programmable L1 Cache Sizes
  - Individually configurable instruction and data caches
  - 32KB I/D cache
  - 4-way set associative
  - Up to 9 non-blocking loads
  - Data cache supports coherent and non-coherent Write-back with write-allocation
  - 32-byte cache line size, doubleword sectored - suitable for standard single-port SRAM
  - Cache line locking support
  - Non-blocking prefetches
  - Duplicate tag array in D-cache allows coherence requests to access the cache in parallel with normal load/store traffic
- Standard Memory Management Unit
  - 32 dual-entry MIPS32-style JTLB per VPE with variable page sizes
  - JTLBs are sharable under software control
  - 4-5 entry instruction TLB
  - 8-entry data TLB
- OCP Bus Interface Unit (BIU)
  - 32b address and 64b data
  - Supports bursts of 4x64b
  - 8 entry write buffer - handles eviction data, intervention response, uncached, and uncached accelerated store data
  - Simple Byte enable mode allows easier bridging to other bus standards
  - Extensions for management of front side L2 cache
  - Intervention port supports memory coherency for use in a 1004K Coherent Processing System
- Multiply-Divide Unit
  - Maximum issue rate of one 32x32 multiply per clock
  - Early-in divide control. Minimum 11, maximum 34 clock latency on divide
- Power Control
  - No minimum frequency
  - Support for software-controlled clock divider
  - Support for extensive use of fine-grain clock gating

- EJTAG Debug Support
  - Start, stop, and single stepping control
  - Software breakpoints via the SDBBP instruction
  - Optional hardware breakpoints on virtual addresses; 0, 2, or 4 instruction and 0,1, or 2 data breakpoints per VPE
- SOC-it L2 Cache Controller
  - 7-stage pipeline. (Optional 8th stage for pipelined memory arrays.)
  - 32-bit address paths, 256-bit internal data paths
  - 8-way set associativity
  - Cache size: 256KB
  - Line Size: 32 bytes (4 doublewords)



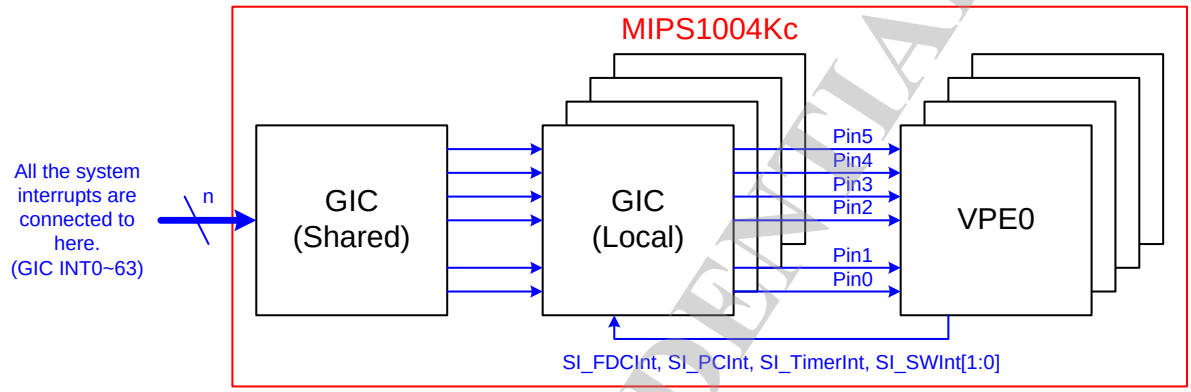
1004K CPU Block Diagram

1.2 Memory Map Summary

Start	End	Size	Description
0	1BFFFFFF	448M	DRAM Direct Map
1C000000	1DFFFFFF	32M	<<Reserved>>
1E000000	1E0000FF	256	SYSCTL
1E000100	1E0001FF	256	TIMER
1E000200	1E0002FF	256	INTCTL
1E000300	1E0003FF	256	Flash Controller (NOR/SRAM/SDRAM)
1E000400	1E0004FF	256	Rbus Matrix CTRL
1E000500	1E0005FF	256	MIPS CNT
1E000600	1E0006FF	256	GPIO
1E000700	1E0007FF	256	S/PDIF
1E000800	1E0008FF	256	DMA_CFG_ARB
1E000900	1E0009FF	256	I2C
1E000A00	1E000AFF	256	I2S
1E000B00	1E000BFF	256	SPI CSR
1E000C00	1E000CFF	256	UARTLITE 1
1E000D00	1E000DFF	256	UARTLITE 2
1E000E00	1E000EFF	256	UARTLITE 3
1E000F00	1E000FFF	256	ANACTL
1E001000	1E0017FF	2K	<<Reserved>>
1E001800	1E001FFF	2K	<<Reserved>>
1E002000	1E0027FF	2K	PCM (up to 16 channel)
1E002800	1E002FFF	2K	Generic DMA (up to 64 channel)
1E003000	1E0037FF	2K	NAND Controller *(actually 1K in Module)
1E003800	1E003FFF	2K	NAND_ECC Controller *(actually 3K in module)
1E004000	1E004FFF	4K	Crypto Engine
1E005000	1E005FFF	4K	MEM_CTRL (DDRII/DDRIII)
1E006000	1E006FFF	4K	EXT_MC_ARB
1E007000	1E007FFF	4K	HS DMA
1E008000	1E00FFFF	32K	<<Reserved>>
1E010000	1E0FFFFF	960K	<<Reserved>>
1E100000	1E10DFFF	56K	Frame Engine (FE SRAM: 0x1E108000~0x1E10DFFF)
1E10E000	1E10FFFF	8K	PCIe SRAM
1E110000	1E117FFF	32K	Ethernet GMAC
1E118000	1E11FFFF	32K	ROM
1E120000	1E12FFFF	64K	<<Reserved>>
1E130000	1E137FFF	32K	SDXC
1E138000	1E13FFFF	32K	<<Reserved>>
1E140000	1E17FFFF	256K	PCI Express
1E180000	1E1BFFFF	256K	<<Reserved>>
1E1C0000	1E1FFFFF	256K	USB Host (U2+U3)
1E200000	1E23FFFF	256K	<<Reserved>>
1E240000	1E24FFFF	64K	<<Reserved>>

1E250000	1E7FFFFF	5824K	<<Reserved>>
1E800000	1EBFFFFF	4M	PCIE Direct Access for iNIC
1EC00000	1FBBFFFF	16128K	<<Reserved>>
1FBC0000	1FBDFFFF	128	CM_GIC
1FBE0000	1FBEFFFF	64K	<<Reserved>>
1FBF0000	1FBF7FFF	32K	CM_CPC
1FBF8000	1FBFFFFF	32K	CM_GCR
1FC00000	1FFFFFFF	4M	ROM/SPI FLASH Direct Access
20000000	23FFFFFF	64M	DRAM Re-Map
24000000	5FFFFFFF	960M	<<Reserved>>
60000000	6FFFFFFF	256M	PCIE Direct Access
70000000	7FFFFFFF	256M	<<Reserved>>

1.3 Interupt Table Summary



MT7621 Interrupt architecture

	Moudle	Source Pin	Level/Edge	Map to VPE	Map to Pin (Compatibility)
GIC INT0	MIPS1004Kc	SI_CM_Err	Level	VPE0	Pin0(0x0500)
GIC INT1	MIPS1004Kc	SI_CM_PCInt	Level	VPE0	Pin0(0x0504)
GIC INT2					Pin0(0x0508)
GIC INT3	FE	fe_int_req	Level	VPE0	Pin3(0x050C)
GIC INT4	PCIE0	pcie_int_req0	Level	VPE0	Pin4(0x0510)
GIC INT5	MIPS1004Kc/AUX_STCK	SI_TimerInt/MIPS_CNTInt	Level	VPE0	Pin5(0x0514)
GIC INT6					Pin0(0x0518)
GIC INT7					Pin0(0x051C)
GIC INT8					Pin0(0x0520)
GIC INT9					Pin0(0x0524)
GIC INT10	PCM	pcm_int	Level	VPE0	Pin0(0x0528)
GIC INT11	HSDMA	hs_dma_int	Level	VPE0	Pin0(0x052C)
GIC INT12	GPIO	gpio_int	Level	VPE0	Pin0(0x0530)
GIC INT13	GDMA	gdma_int	Level	VPE0	Pin0(0x0534)
GIC INT14	NFI	nd_intreq	Level	VPE0	Pin0(0x0538)
GIC INT15	NFIECC	nfiecc_intreq	Level	VPE0	Pin0(0x053C)
GIC INT16	I2S	i2s_int	Level	VPE0	Pin0(0x0540)
GIC INT17	SPI	spi_int	Level	VPE0	Pin0(0x0544)
GIC INT18	S/PDIF	spdif_int	Level	VPE0	Pin0(0x0548)
GIC INT19	CRYPTO	crypto_int	Level	VPE0	Pin0(0x054C)
GIC INT20	SDXC	sdxc_int	Level	VPE0	Pin0(0x0550)
GIC INT21	PCTRL	r2p_int	Level	VPE0	Pin0(0x0554)
GIC INT22	USB2.0/3.0	usb_int	Level	VPE0	Pin0(0x0558)
GIC INT23	SWITCH	esw_int_in	Level	VPE0	Pin0(0x055C)
GIC INT24	PCIE1	pcie_int_req1	Level	VPE0	Pin4(0x0560)
GIC INT25	PCIE2	pcie_int_req2	Level	VPE0	Pin4(0x0564)
GIC INT26	UART-LITE	uartl1_int	Level	VPE0	Pin0(0x0568)
GIC INT27	UART-LITE	uartl2_int	Level	VPE0	Pin0(0x056C)
GIC INT28	UART-LITE	uartl3_int	Level	VPE0	Pin0(0x0570)
GIC INT29	TIMER	wdtimer_int	Level	VPE0	Pin0(0x0574)
GIC INT30	TIMER	timer0_int	Level	VPE0	Pin0(0x0578)
GIC INT31	TIMER	timer1_int	Level	VPE0	Pin0(0x057C)

	Moudle	Source Pin	Level/Edge	Map to VPE	Map to Pin (Compatibility)
GIC INT32					Pin0(0x0580)
GIC INT33					Pin0(0x0584)
GIC INT34					Pin0(0x0588)
GIC INT35					Pin0(0x058C)
GIC INT36					Pin0(0x0590)
GIC INT37					Pin0(0x0594)
GIC INT38					Pin0(0x0598)
GIC INT39					Pin0(0x059C)
GIC INT40					Pin0(0x05A0)
GIC INT41					Pin0(0x05A4)
GIC INT42					Pin0(0x05A8)
GIC INT43					Pin0(0x05AC)
GIC INT44					Pin0(0x05B0)
GIC INT45					Pin0(0x05B4)
GIC INT46					Pin0(0x05B8)
GIC INT47					Pin0(0x05BC)
GIC INT48					Pin0(0x05C0)
GIC INT49					Pin0(0x05C4)
GIC INT50					Pin0(0x05C8)
GIC INT51					Pin0(0x05CC)
GIC INT52					Pin0(0x05D0)
GIC INT53					Pin0(0x05D4)
GIC INT54					Pin0(0x05D8)
GIC INT55					Pin0(0x05DC)
GIC INT56	MIPS1004Kc	IPI0_VPE0	Edge	VPE0	Pin1(0x05E0)
GIC INT57	MIPS1004Kc	IPI0_VPE1	Edge	VPE1	Pin1(0x05E4)
GIC INT58	MIPS1004Kc	IPI0_VPE2	Edge	VPE2	Pin1(0x05E8)
GIC INT59	MIPS1004Kc	IPI0_VPE3	Edge	VPE3	Pin1(0x05EC)
GIC INT60	MIPS1004Kc	IPI1_VPE0	Edge	VPE0	Pin2(0x05F0)
GIC INT61	MIPS1004Kc	IPI1_VPE1	Edge	VPE1	Pin2(0x05F4)
GIC INT62	MIPS1004Kc	IPI1_VPE2	Edge	VPE2	Pin2(0x05F8)
GIC INT63	MIPS1004Kc	IPI1_VPE3	Edge	VPE3	Pin2(0x05F8)

PS: the empty part means reserved.

2. Registers

2.1 Nomenclature

The following nomenclature is used for register types:

RO	Read Only
WO	Write Only
RW	Read or Write
RC	Read Clear
W1C	Write One Clear
-	Reserved bit
X	Undefined binary value

2.2 System Control

2.2.1 Features

- Provides read-only chip revision registers
- Provides a window to access boot-strapping signals
- Supports memory remapping configurations
- Supports software reset to each platform building block
- Provides registers to determine GPIO and other peripheral pin muxing schemes
- Provides some power-on-reset only test registers for software programmers
- Combines miscellaneous registers (such as clock skew control, status register, memo registers, etc)

2.2.2 Block Diagram

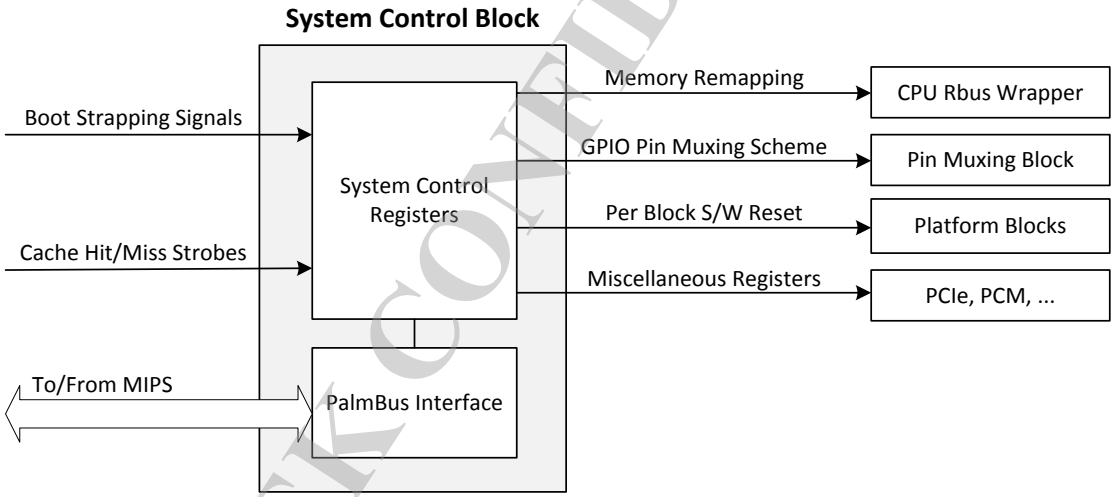


Figure 2-1 System Control Block Diagram

2.2.3 Registers

SYSCTL Changes LOG

Revision	Date	Author	Change Log
0.1	2012/7/11	James Hu	Initialization

Module name: SYSCTL Base address: (+1E000000h)

Address	Name	Width	Register Function
1E000000	CHIPID0_3	32	CHIP ID ASCII Character 0-3
1E000004	CHIPID4_7	32	CHIP ID ASCII Character 4-7
1E00000C	CHIP_REV_ID	32	Chip Revision Identification
1E000010	SYSCFG	32	System Configuration Register
1E000014	SYSCFG1	32	System Configuration Register 1
1E000018	TESTSTAT	32	Firmware Test Status
1E00001C	TESTSTAT2	32	Firmware Test Status 2
1E000020	BOOT_SRAM_BASE	32	Boot from SRAM base Address
1E000024	BOOT_RELEASE	32	Release CPU's reset to let CPU boot in boot from SRAM mode
1E00002C	CLKCFG0	32	Clock Configuration Register 0
1E000030	CLKCFG1	32	Clock Configuration Register 1
1E000034	RSTCTL	32	Reset Control Register
1E000038	RSTSTAT	32	Reset Status Register
1E00003C	MISR_GOLDEN	32	ROM BIST MISR Golden Value
1E000040	MISR_RESULT	32	ROM BIST MISR Result Value
1E000044	CUR_CLK_STS	32	Current clock status
1E000048	PAD_UART1_GPIO0_CFG	32	PAD configuration of UART1 and GPIO0 groups
1E00004C	PAD_UART3_I2C_CFG	32	PAD configuration of UART3 and I2C groups
1E000050	PAD_UART2_JTAG_CFG	32	PAD configuration of UART2 and JTAG groups
1E000054	PAD_PERST_WDT_CFG	32	PAD configuration of PICe RST and WDT RST groups
1E000058	PAD_RGMII2_MDIO_CFG	32	PAD configuration of RGMII2 and MDIO RST groups
1E00005C	PAD_SDXC_SPI_CFG	32	PAD configuration of SDXC and SPI RST groups
1E000060	GPIO_MODE	32	GPIO purpose selection
1E000068	MEMO1	32	Memory1
1E00006C	MEMO2	32	Memory2
1E000070	PAD_BOPT_ESWINT_CFG	32	PAD configuration of Bonding OPT and ESW INT groups
1E000074	PAD_RGMII1_CFG	32	PAD configuration of RGMII1 group
1E000078	CPE_ROSC_SEL0	32	
1E00007C	CPE_ROSC_SEL1	32	

1E000080	CPU_CPE_CNT0	32	CPU CPE counter 0
1E000084	CPU_CPE_CNT1	32	CPU CPE counter 1
1E000088	CPU_CFG	32	CPU configuration
1E00008C	CPU_MEM_CFG	32	CPU memory delay, power down and sleep control
1E000090	FMTR_CFG0	32	Frequency meter configuration 0
1E000094	FMTR_CNT_MAX	32	Frequency meter count maximum
1E000098	FMTR_CNT_MIN	32	Frequency meter count minimum
1E00009C	FMTR_CNT_VAL	32	Frequency meter counter value

1E000000 CHIPID0_3 CHIP ID ASCII Character 0-3 3637544D

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CHIP_ID3								CHIP_ID2							
Type	RO								RO							
Reset	0	0	1	1	0	1	1	0	0	0	1	1	0	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHIP_ID1								CHIP_ID0							
Type	RO								RO							
Reset	0	1	0	1	0	1	0	0	0	1	0	0	1	1	0	1

Bit(s)	Name	Description
31:24	CHIP_ID3	ASCII CHIP Name Identification Character 3
23:16	CHIP_ID2	ASCII CHIP Name Identification Character 2
15:8	CHIP_ID1	ASCII CHIP Name Identification Character 1
7:0	CHIP_ID0	ASCII CHIP Name Identification Character 0

1E000004 CHIPID4_7 CHIP ID ASCII Character 4-7 20203132

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CHIP_ID7								CHIP_ID6							
Type	RO								RO							
Reset	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHIP_ID5								CHIP_ID4							
Type	RO								RO							
Reset	0	0	1	1	0	0	0	1	0	0	1	1	0	0	1	0

Bit(s)	Name	Description
31:24	CHIP_ID7	ASCII CHIP Name Identification Character 3
23:16	CHIP_ID6	ASCII CHIP Name Identification Character 2
15:8	CHIP_ID5	ASCII CHIP Name Identification Character 1
7:0	CHIP_ID4	ASCII CHIP Name Identification Character 0

1E00000C CHIP_REV_ID Chip Revision Identification 00020101

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Name															DUMMY_ID	PKG_ID
Type															RO	RO
Reset															1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					VER_ID								ECO_ID			
Type					RO								RO			
Reset					0	0	0	1					0	0	0	1

Bit(s)	Name	Description
17	DUMMY_ID	**DUMMY ID** 1: Reserved 0: Reserved
16	PKG_ID	**Package ID** 1: A 0: N
11:8	VER_ID	**Chip Version ID**
3:0	ECO_ID	**Chip ECO ID**

1E000010 SYSCFG System Configuration Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TEST_CODE										BS_SHADOW[9:4]					
Type	RW										RO					
Reset	0	0	0	0	0	0	0	0			0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BS_SHADOW[3:0]						DRAM_FROM_EE	XTAL_MODE_SEL			OCP_RATIO	DRAM_TYPE	CHIP_MODE			
Type	RO						RO	RO			RO	RO	RO			
Reset	0	0	0	0			0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	TEST_CODE	**Default value is from bootstrap and can be modified by software.**
21:12	BS_SHADOW	**BS shadow register for last boot-up value** Displays a backup copy of the last bootup value
9	DRAM_FROM_EE	**DRAM configuration source** 1: Auto detection 0: from EEPROM
8:6	XTAL_MODE_SEL	**XTAL mode selection** 0: 20 MHz, Self Oscillation mode 1: 20 MHz, Single end input 2: 20 MHz, differential input 3: 40 MHz, Self Oscillation mode 4: 40 MHz, Single end input 5: 40 MHz, differential input 6: 25 MHz, Self Oscillation mode 7: 25 MHz, Single end input
5	OCP_RATIO	0: 1/3 1: 1/4
4	DRAM_TYPE	**DDR type** 1: DDR2

		0: DDR3
3:0	CHIP_MODE	**A vector to set chip function/test/debug modes in non-test/debug operation.** For more information see the Bootstrapping Pins Description in the datasheet for this chip.

1E000014 SYSCFG1 System Configuration Register 1 0000C100

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GE2_MODE		GE1_MODE					PCIE_RC_MODE				CPU_CTRL_UTIF				
Type	RW		RW					RW				RW				
Reset	1	1	0	0				1				0				

Bit(s)	Name	Description
15:14	GE2_MODE	**Gigabit Port #2 Mode** Sets the interface mode on Gigabit port 2 3: RJ-45 Mode 2: Reverse MII Mode (10/100 Mbps) 1: MII Mode (10/100 Mbps) 0: RGMII Mode (10/100/1000 Mbps)
13:12	GE1_MODE	**Gigabit Port #1 Mode** Sets the interface mode on Gigabit port 1. 0: RGMII Mode (10/100/1000 Mbps) 1: MII Mode (10/100 Mbps) 2: Reverse MII Mode (10/100 Mbps) 31: Reserved
8	PCIE_RC_MODE	**PCIe Mode** 1: Root Complex mode 0: End Point mode
4	CPU_CTRL_UTIF	**CPU control and monitor UTIF interface enable** 1: Enable 0: Disable

1E000018 TESTSTAT Firmware Test Status 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TESTSTAT[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TESTSTAT[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TESTSTAT	**Firmware Test Status register**

NOTE: This register is reset only by a power-on reset.

1E00001C TESTSTAT2 Firmware Test Status 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TESTSTAT2[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TESTSTAT2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TESTSTAT2	**Firmware Test Status Register 2** NOTE: This register is reset only by a power-on reset.

1E000020 BOOT_SRAM_BASE Boot from SRAM base Address 1E180000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BOOTSRAMBASE[31:16]															
Type	RW															
Reset	0	0	0	1	1	1	1	0	0	0	0	1	1	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BOOTSRAMBASE[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	BOOTSRAMBASE	**Boot from SRAM base address (Test mode only)** Addr_tuned = bootsram[31:0] | oc_maddr[15:0]

1E000024 BOOT_RELEASE Release CPU's reset to let CPU boot in boot from SRAM mode 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																BOOT_RELEASE
Type																RW
Reset																0

Bit(s)	Name	Description

0	BOOT_RELEASE	**Release CPU's command to access DRAM or CR.** 1: Release CPU command 0: Block CPU command

1E000028 RESERVED_CR Reserved CR1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED_CR1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED_CR1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

1E00002C CLKCFG0 Clock Configuration Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CPU_CLK_SEL		OSC_1US_DIV						MPLL_CFG_SEL	REFCLK_FDIV					PCIE_CLK_SEL	REFCLK_FFRAC[4:4]
Type	RW		RW						RW	RW					RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REFCLK_FFRAC[3:0]				REFCLK0_RATE					TRGMII_CLK_SEL		PERI_CLK_SEL				
Type	RW				RW					RW		RW				
Reset	0	0	0	0	0	0	0			0	0	0				

Bit(s)	Name	Description
31:30	CPU_CLK_SEL	**CPU clock selection.** CPU PLL is programmable. 3: XTAL clock 2: XTAL clock 1: CPU PLL 0: 500MHz
29:24	OSC_1US_DIV	**Oscillator 1 usec Divider** Sets the maximum for the reference clock counter for either a 20 MHz or 40 MHz external XTAL input. The count increments each 1usec (indicating 1 MHz), up to the maximum, before resetting to zero. This counts the frequency of an external XTAL. This count is used to output a 32 KHz frequency to the REFCLK0 pin. 0: Automatically generates a 1 usec system tick regardless of whether XTAL frequency is 20 MHz or 40 MHz. 39: Default value for an external 40 MHz XTAL. 19: Default value for an external 20 MHz XTAL. Others: Manual mode for tick generation.
23	MPLL_CFG_SEL	**MEMPLL parameter configuration selection**

		1: from CR configuration 0: follow XTAL frequncy boot strapping
22:18	REFCLK_FDIV	**Internal Clock Frequency Divider** The frequency divider used to generate the Fraction-N clock frequency. Valid values range from 1 to 31. Fraction-N clock frequency = (INT_CLK_FFRAC/INT_CLK_FDIV)*PLL_FREQ
17	PCIE_CLK_SEL	**PCIe clock selection.** 1: from GPLL 125MHz 0: from PCIe PHY
16:12	REFCLK_FFRAC	**Internal Clock Fraction-N Frequency** A parameter used in conjunction with INT_CLK_FDIV to generate the Fraction-N clock frequency. Valid values range from 0 to 31. Fraction-N clock Frequency = (INT_CLK_FFRAC/INT_CLK_FDIV)*PLL_FREQ
11:9	REFCLK0_RATE	**Output clock rate of reference Clock 0** 7: CPU clock/8 6: Reserved 5: Internal Fraction-N_CLK/2 4: Internal Fraction-N_CLK/4 3: Reserved 2: 25 MHz 1: 12.5 MHz 0: Xtal clock(20/25/40 MHz by boot strap)
6:5	TRGMII_CLK_SEL	**TRGMII Tx clock selection** 2: APLL 1: DDR PLL to DRAMC 0: 250MHz
4	PERI_CLK_SEL	**Peripheral Clock Source Select** 1: XTAL input 0: 50 MHz from EPLL

1E000030 CLKCFG1 Clock Configuration Register 1 67BFEFE0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		SHXC_CLK_EN	CRYPTO_CLK_EN			PCIE2_CLK_EN	PCIE1_CLK_EN	PCIE0_CLK_EN	ETH_CLK_EN		UART3_CLK_EN	UART2_CLK_EN	UART1_CLK_EN	SPI_CLK_EN	I2S_CLK_EN	I2C_CLK_EN
Type		RW	RW			RW	RW	RW	RW		RW	RW	RW	RW	RW	RW
Reset		1	1			1	1	1	1		1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	NAND_CLK_EN	GDMA_CLK_EN	PIO_CLK_EN		PCM_CLK_EN	MC_CLK_EN	INT_CLK_EN	TIMER_CLK_EN	SPDIFTX_CLK_EN	FE_CLK_EN	HSDMA_CLK_EN					
Type	RW	RW	RW		RW	RW	RW	RW	RW	RW	RW					
Reset	1	1	1		1	1	1	1	1	1	1					

Bit(s)	Name	Description
30	SHXC_CLK_EN	**SHXC clock control** 1: Clock Enable 0: Clock Disable
29	CRYPTO_CLK_EN	**AUX system tick counter clock control** 1: Clock Enable

		0: Clock Disable
26	PCIE2_CLK_EN	**PCIE2 clock control** 1: Clock Enable 0: Clock Disable
25	PCIE1_CLK_EN	**PCIE1 clock control** 1: Clock Enable 0: Clock Disable
24	PCIE0_CLK_EN	**PCIE0 clock control** 1: Clock Enable 0: Clock Disable
23	ETH_CLK_EN	**ETH clock control** 1: Clock Enable 0: Clock Disable
21	UART3_CLK_EN	**UART3 clock control** 1: Clock Enable 0: Clock Disable
20	UART2_CLK_EN	**UART2 clock control** 1: Clock Enable 0: Clock Disable
19	UART1_CLK_EN	**UART1 clock control** 1: Clock Enable 0: Clock Disable
18	SPI_CLK_EN	**SPI clock control** 1: Clock Enable 0: Clock Disable
17	I2S_CLK_EN	**I2S clock control** 1: Clock Enable 0: Clock Disable
16	I2C_CLK_EN	**I2C clock control** 1: Clock Enable 0: Clock Disable
15	NAND_CLK_EN	**NAND clock control** 1: Clock Enable 0: Clock Disable
14	GDMA_CLK_EN	**GDMA clock control** 1: Clock Enable 0: Clock Disable
13	PIO_CLK_EN	**PIO clock control** 1: Clock Enable 0: Clock Disable
11	PCM_CLK_EN	**PCM clock control** 1: Clock Enable 0: Clock Disable
10	MC_CLK_EN	**MC clock control** 1: Clock Enable 0: Clock Disable
9	INT_CLK_EN	**INT clock control** 1: Clock Enable 0: Clock Disable
8	TIMER_CLK_EN	**TIMER clock control** 1: Clock Enable 0: Clock Disable
7	SPDIFTX_CLK_EN	**SPDIFTX clock control** 1: Clock Enable 0: Clock Disable
6	FE_CLK_EN	**FE clock control**

		1: Clock Enable 0: Clock Disable
5	HSDMA_CLK_EN	**HSDMA clock control** 1: Clock Enable 0: Clock Disable

1E000034 RSTCTL Reset Control Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PPE_RST	SDXC_RST	CRYPTO_RST	AUX_STCK_RST		PCIE2_RST	PCIE1_RST	PCIE0_RST	ETH_RST		UART3_RST	UART2_RST	UART1_RST	SPI_RST	I2S_RST	I2C_RST
Type	RW	RW	RW	RW		RW	RW	RW	RW		RW	RW	RW	RW	RW	RW
Reset	0	0	0	0		0	0	0	0		0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	NFI_RST	GDMA_RST	PIO_RST		PCM_RST	MC_RST	INT_RST	TIMER_RST	SPDIFTX_RST	FE_RST	HSDMA_RST			MCM_RST		SYS_RST
Type	RW	RW	RW		RW	RW	RW	RW	RW	RW	RW			RW		W1C
Reset	0	0	0		0	0	0	0	0	0	0			0		0

Bit(s)	Name	Description
31	PPE_RST	**PPE reset control** 1: Reset Assert 0: Reset Deassert
30	SDXC_RST	**SHXC reset control** 1: Reset Assert 0: Reset Deassert
29	CRYPTO_RST	**Crypto engine reset control** 1: Reset Assert 0: Reset Deassert
28	AUX_STCK_RST	**AUX system tick counter clock control** 1: Reset Assert 0: Reset Deassert
26	PCIE2_RST	**PCIE2 reset control** 1: Reset Assert 0: Reset Deassert
25	PCIE1_RST	**PCIE1 reset control** 1: Reset Assert 0: Reset Deassert
24	PCIE0_RST	**PCIE0 reset control** 1: Reset Assert 0: Reset Deassert
23	ETH_RST	**ETH reset control** 1: Reset Assert 0: Reset Deassert
21	UART3_RST	**UART3 reset control** 1: Reset Assert 0: Reset Deassert
20	UART2_RST	**UART2 reset control**

		1: Reset Assert 0: Reset Deassert
19	UART1_RST	**UART1 reset control** 1: Reset Assert 0: Reset Deassert
18	SPI_RST	**SPI reset control** 1: Reset Assert 0: Reset Deassert
17	I2S_RST	**I2S reset control** 1: Reset Assert 0: Reset Deassert
16	I2C_RST	**I2C reset control** 1: Reset Assert 0: Reset Deassert
15	NFI_RST	**NFI reset control** 1: Reset Assert 0: Reset Deassert
14	GDMA_RST	**GDMA reset control** 1: Reset Assert 0: Reset Deassert
13	PIO_RST	**PIO reset control** 1: Reset Assert 0: Reset Deassert
11	PCM_RST	**PCM reset control** 1: Reset Assert 0: Reset Deassert
10	MC_RST	**MC reset control** 1: Reset Assert 0: Reset Deassert
9	INT_RST	**INT reset control** 1: Reset Assert 0: Reset Deassert
8	TIMER_RST	**TIMER reset control** 1: Reset Assert 0: Reset Deassert
7	SPDIFTX_RST	**SPDIFTX reset control** 1: Reset Assert 0: Reset Deassert
6	FE_RST	**FE reset control** 1: Reset Assert 0: Reset Deassert
5	HSDMA_RST	**HSDMA reset control** 1: Reset Assert 0: Reset Deassert
2	MCM_RST	**MCM(MT7530) reset control** 1: Reset Assert 0: Reset Deassert
0	SYS_RST	**Whole System Reset Control** 1: Whole System Reset 0: NA

1E000038 RSTSTAT Reset Status Register C0030000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Name	WDT2SYSRST_EN	WDT2RSTO_EN	WDTRSTPD													
Type	RW	RW	RW													
Reset	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name														SWSYSRST	WDRST	
Type														W1C	W1C	
Reset														0	0	

Bit(s)	Name	Description
31	WDT2SYSRST_EN	**WDT reset apply to System Reset** Enables watchdog timeout to trigger a system reset. 1: Enable 0: Disable
30	WDT2RSTO_EN	**WDT reset apply to watch dog reset pin out.** 1: Enable 0: Disable
29:16	WDTRSTPD	**Watchdog Reset Output Low Period** Controls the WDT reset output low period. For example: If the pin share mode was set correctly and WDT2RSTO_EN=1, When WDTRSTPD= 0, you can see duration of 1 usec low on the WDT reset output pin. When WDTRSTPD= 3, you can see duration of 4 usec low on the WDT reset output pin. (unit: 1 usec)
2	SWSYSRST	**Software System Reset** Indicates when software has reset the chip by writing to the RSTSYS bit in RSTCTL. NOTE: This register is reset only by a power on reset. 0: Has no effect. 1: Clears this bit.
1	WDRST	**Watchdog Reset** Indicates when the watchdog timer has reset the chip. NOTE: This register is reset only by power-on reset. 0: Has no effect. 1: Clears this bit.

1E00003C MISR_GOLDEN ROM BIST MISR Golden Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MISR_GOLDEN[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MISR_GOLDEN[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MISR_GOLDEN	**ROM BIST golden value**

1E000040 MISR_RESULT ROM BIST MISR Result Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MISR_RESULT[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MISR_RESULT[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MISR_RESULT	**ROM BIST result**

1E000044 CUR_CLK_STS Current clock status 00030A01

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name												SAME_FREQ		CUR_OCP_RATIO		
Type												RO		RO		
Reset												0		0	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				CUR_CPU_FDIV								CUR_CPU_FFRAC				
Type				RO								RO				
Reset				0	1	0	1	0				0	0	0	0	1

Bit(s)	Name	Description
20	SAME_FREQ	**SYS_CLK and DRAM_clk are same frequency.**
18:16	CUR_OCP_RATIO	**Current CPU_OCP_RATIO(SYS:CPU)** 4: 1:4 3: 1:3
12:8	CUR_CPU_FDIV	**Current divider number of CPU frequency**
4:0	CUR_CPU_FFRAC	**Current fraction number of CPU frequency.**

1E000048 PAD_UART1_GPIO0_CFG PAD configuration of UART1 and GPIO0 groups 0A180A18

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			UART1_RDSEL		UART1_TDSEL						UART1_E4_E2		UART1_PU	UART1_PD	UART1_SMT	UART1_SR
Type			RW		RW						RW		RW	RW	RW	RW
Reset			0	0	1	0	1	0			0	1	1	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			GPIO0_RDSEL		GPIO0_TDSEL						GPIO0_E4_E2		GPIO0_PU	GPIO0_PD	GPIO0_SMT	GPIO0_SR
Type			RW		RW						RW		RW	RW	RW	RW

Reset			0	0	1	0	1	0			0	1	1	0	0	0

Bit(s)	Name	Description
29:28	UART1_RDSEL	**RX duty select** RDSEL[0]: Level shifter duty high when asserted (high pulse width adjustment) RDSEL[1]: Level shifter duty low when asserted (low pulse width adjustment) 3: For 1.8V 0: For 3.3V
27:24	UART1_TDSEL	**TX duty select** TDSEL[1:0]: Output level shifter duty high when asserted (high pulse width adjustment) TDSEL[3:2]: Output level shifter duty low when asserted(low pulse width adjustment) 10: For3.3V 0: For 1.8V
21:20	UART1_E4_E2	**TX Driving Strength Control.** 3: 8mA 2: 6mA 1: 4mA 0: 2mA
19	UART1_PU	**75K pull-up resistor control.** 1: Enable 0: Disable
18	UART1_PD	**75K pull-down resistor control.** 1: Enable 0: Disable
17	UART1_SMT	**RX input buffer schmit trigger hysteresis control enable.** 1: Enable 0: Disable
16	UART1_SR	**Output Slew Rate Control.** 1: Slower slew. 0: No slew rate controlled.
13:12	GPIO0_RDSEL	**RX duty select** RDSEL[0]: Level shifter duty high when asserted (high pulse width adjustment) RDSEL[1]: Level shifter duty low when asserted (low pulse width adjustment) 3: For 1.8V 0: For 3.3V
11:8	GPIO0_TDSEL	**TX duty select** TDSEL[1:0]: Output level shifter duty high when asserted (high pulse width adjustment) TDSEL[3:2]: Output level shifter duty low when asserted(low pulse width adjustment) 10: For3.3V 0: For 1.8V
5:4	GPIO0_E4_E2	**TX Driving Strength Control.** 3: 8mA 2: 6mA 1: 4mA 0: 2mA
3	GPIO0_PU	**75K pull-up resistor control.** 1: Enable 0: Disable
2	GPIO0_PD	**75K pull-down resistor control.** 1: Enable 0: Disable
1	GPIO0_SMT	**RX input buffer schmit trigger hysteresis control enable.** 1: Enable 0: Disable
0	GPIO0_SR	**Output Slew Rate Control.** 1: Slower slew. 0: No slew rate controlled.

1E00004C **PAD_UART3_I2C_CFG** **PAD configuration of UART3 and I2C groups** **0A140A10**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			UART3_RDSEL		UART3_TDSEL						UART3_E4_E2		UART3_PU	UART3_PD	UART3_SMT	UART3_SR
Type			RW		RW						RW		RW	RW	RW	RW
Reset			0	0	1	0	1	0			0	1	0	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			I2C_RDSEL		I2C_TDSEL						I2C_E4_E2		I2C_PU	I2C_PD	I2C_SMT	I2C_SR
Type			RW		RW						RW		RW	RW	RW	RW
Reset			0	0	1	0	1	0			0	1	0	0	0	0

Bit(s)	Name	Description
29:28	UART3_RDSEL	**RX duty select** RDSEL[0]: Level shifter duty high when asserted (high pulse width adjustment) RDSEL[1]: Level shifter duty low when asserted (low pulse width adjustment) 3: For 1.8V 0: For 3.3V
27:24	UART3_TDSEL	**TX duty select** TDSEL[1:0]: Output level shifter duty high when asserted (high pulse width adjustment) TDSEL[3:2]: Output level shifter duty low when asserted(low pulse width adjustment) 10: For3.3V 0: For 1.8V
21:20	UART3_E4_E2	**TX Driving Strength Control.** 3: 8mA 2: 6mA 1: 4mA 0: 2mA
19	UART3_PU	**75K pull-up resistor control.** 1: Enable 0: Disable
18	UART3_PD	**75K pull-down resistor control.** 1: Enable 0: Disable
17	UART3_SMT	**RX input buffer schmit trigger hysteresis control enable.** 1: Enable 0: Disable
16	UART3_SR	**Output Slew Rate Control.** 1: Slower slew. 0: No slew rate controlled.
13:12	I2C_RDSEL	**RX duty select** RDSEL[0]: Level shifter duty high when asserted (high pulse width adjustment) RDSEL[1]: Level shifter duty low when asserted (low pulse width adjustment) 3: For 1.8V 0: For 3.3V
11:8	I2C_TDSEL	**TX duty select** TDSEL[1:0]: Output level shifter duty high when asserted (high pulse width adjustment) TDSEL[3:2]: Output level shifter duty low when asserted(low pulse width adjustment) 10: For3.3V 0: For 1.8V
5:4	I2C_E4_E2	**TX Driving Strength Control.** 3: 8mA

		2: 6mA 1: 4mA 0: 2mA
3	I2C_PU	**75K pull-up resistor control.** 1: Enable 0: Disable
2	I2C_PD	**75K pull-down resistor control.** 1: Enable 0: Disable
1	I2C_SMT	**RX input buffer schmit trigger hysteresis control enable.** 1: Enable 0: Disable
0	I2C_SR	**Output Slew Rate Control.** 1: Slower slew. 0: No slew rate controlled.

1E000050 PAD_UART2_JTAG_CFG PAD configuration of UART2 and JTAG groups 0A140A14

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			JTAG_RDSEL		JTAG_TDSEL						JTAG_E4_E2		JTAG_PU	JTAG_PD	JTAG_SMT	JTAG_SR
Type			RW		RW						RW		RW	RW	RW	RW
Reset			0	0	1	0	1	0			0	1	0	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			UART2_RDSEL		UART2_TDSEL						UART2_E4_E2		UART2_PU	UART2_PD	UART2_SMT	UART2_SR
Type			RW		RW						RW		RW	RW	RW	RW
Reset			0	0	1	0	1	0			0	1	0	1	0	0

Bit(s)	Name	Description
29:28	JTAG_RDSEL	**RX duty select** RDSEL[0]: Level shifter duty high when asserted (high pulse width adjustment) RDSEL[1]: Level shifter duty low when asserted (low pulse width adjustment) 3: For 1.8V 0: For 3.3V
27:24	JTAG_TDSEL	**TX duty select** TDSEL[1:0]: Output level shifter duty high when asserted (high pulse width adjustment) TDSEL[3:2]: Output level shifter duty low when asserted(low pulse width adjustment) 10: For3.3V 0: For 1.8V
21:20	JTAG_E4_E2	**TX Driving Strength Control.** 3: 8mA 2: 6mA 1: 4mA 0: 2mA
19	JTAG_PU	**75K pull-up resistor control.** 1: Enable 0: Disable
18	JTAG_PD	**75K pull-down resistor control.** 1: Enable 0: Disable
17	JTAG_SMT	**RX input buffer schmit trigger hysteresis control enable.**

		1: Enable 0: Disable
16	JTAG_SR	**Output Slew Rate Control.** 1: Slower slew. 0: No slew rate controlled.
13:12	UART2_RDSEL	**RX duty select** RDSEL[0]: Level shifter duty high when asserted (high pulse width adjustment) RDSEL[1]: Level shifter duty low when asserted (low pulse width adjustment) 3: For 1.8V 0: For 3.3V
11:8	UART2_TDSEL	**TX duty select** TDSEL[1:0]: Output level shifter duty high when asserted (high pulse width adjustment) TDSEL[3:2]: Output level shifter duty low when asserted(low pulse width adjustment) 10: For3.3V 0: For 1.8V
5:4	UART2_E4_E2	**TX Driving Strength Control.** 3: 8mA 2: 6mA 1: 4mA 0: 2mA
3	UART2_PU	**75K pull-up resistor control.** 1: Enable 0: Disable
2	UART2_PD	**75K pull-down resistor control.** 1: Enable 0: Disable
1	UART2_SMT	**RX input buffer schmit trigger hysteresis control enable.** 1: Enable 0: Disable
0	UART2_SR	**Output Slew Rate Control.** 1: Slower slew. 0: No slew rate controlled.

1E000054 PAD_PERST_WDT_CFG PAD configuration of PICe RST and WDT RST groups 0A180A18

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			PERST_RDSEL		PERST_TDSEL						PERST_E4_E2		PERST_PU	PERST_PD	PERST_SMT	PERST_SR
Type			RW		RW						RW		RW	RW	RW	RW
Reset			0	0	1	0	1	0			0	1	1	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			WDT_RDSEL		WDT_TDSEL						WDT_E4_E2		WDT_PU	WDT_PD	WDT_SMT	WDT_SR
Type			RW		RW						RW		RW	RW	RW	RW
Reset			0	0	1	0	1	0			0	1	1	0	0	0

Bit(s)	Name	Description
29:28	PERST_RDSEL	**RX duty select** RDSEL[0]: Level shifter duty high when asserted (high pulse width adjustment) RDSEL[1]: Level shifter duty low when asserted (low pulse width adjustment) 3: For 1.8V 0: For 3.3V
27:24	PERST_TDSEL	**TX duty select**

		TDSEL[1:0]: Output level shifter duty high when asserted (high pulse width adjustment) TDSEL[3:2]: Output level shifter duty low when asserted(low pulse width adjustment) 10: For3.3V 0: For 1.8V
21:20	PERST_E4_E2	**TX Driving Strength Control.** 3: 8mA 2: 6mA 1: 4mA 0: 2mA
19	PERST_PU	**75K pull-up resistor control.** 1: Enable 0: Disable
18	PERST_PD	**75K pull-down resistor control.** 1: Enable 0: Disable
17	PERST_SMT	**RX input buffer schmit trigger hysteresis control enable.** 1: Enable 0: Disable
16	PERST_SR	**Output Slew Rate Control.** 1: Slower slew. 0: No slew rate controlled.
13:12	WDT_RDSEL	**RX duty select** RDSEL[0]: Level shifter duty high when asserted (high pulse width adjustment) RDSEL[1]: Level shifter duty low when asserted (low pulse width adjustment) 3: For 1.8V 0: For 3.3V
11:8	WDT_TDSEL	**TX duty select** TDSEL[1:0]: Output level shifter duty high when asserted (high pulse width adjustment) TDSEL[3:2]: Output level shifter duty low when asserted(low pulse width adjustment) 10: For3.3V 0: For 1.8V
5:4	WDT_E4_E2	**TX Driving Strength Control.** 3: 8mA 2: 6mA 1: 4mA 0: 2mA
3	WDT_PU	**75K pull-up resistor control.** 1: Enable 0: Disable
2	WDT_PD	**75K pull-down resistor control.** 1: Enable 0: Disable
1	WDT_SMT	**RX input buffer schmit trigger hysteresis control enable.** 1: Enable 0: Disable
0	WDT_SR	**Output Slew Rate Control.** 1: Slower slew. 0: No slew rate controlled.

1E000058 PAD_RGMII2_MDIO_CFG PAD configuration of RGMII2 and MDIO RST groups 0A210A21

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			RGMII2_RDSEL		RGMII2_TDSEL						RGMII2_E4_E2		RGMII2_PU	RGMII2_PD	RGMII2_SMT	RGMII2_SR

Type			RW		RW						RW		RW	RW	RW	RW
Reset			0	0	1	0	1	0			1	0	0	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			MDIO_RDSEL		MDIO_TDSEL						MDIO_E4_E2		MDIO_PU	MDIO_PD	MDIO_SMT	MDIO_SR
Type			RW		RW						RW		RW	RW	RW	RW
Reset			0	0	1	0	1	0			1	0	0	0	0	1

Bit(s)	Name	Description
29:28	RGMII2_RDSEL	**RX duty select** RDSEL[0]: Level shifter duty high when asserted (high pulse width adjustment) RDSEL[1]: Level shifter duty low when asserted (low pulse width adjustment) 3: For 1.8V 0: For 3.3V
27:24	RGMII2_TDSEL	**TX duty select** TDSEL[1:0]: Output level shifter duty high when asserted (high pulse width adjustment) TDSEL[3:2]: Output level shifter duty low when asserted(low pulse width adjustment) 10: For3.3V 0: For 1.8V
21:20	RGMII2_E4_E2	**TX Driving Strength Control. (CID)** 3: 16mA 2: 12mA 1: 8mA 0: 4mA
19	RGMII2_PU	**75K pull-up resistor control.** 1: Enable 0: Disable
18	RGMII2_PD	**75K pull-down resistor control.** 1: Enable 0: Disable
17	RGMII2_SMT	**RX input buffer schmit trigger hysteresis control enable.** 1: Enable 0: Disable
16	RGMII2_SR	**Output Slew Rate Control. (CID)** 1: Slower slew. 0: No slew rate controlled.
13:12	MDIO_RDSEL	**RX duty select** RDSEL[0]: Level shifter duty high when asserted (high pulse width adjustment) RDSEL[1]: Level shifter duty low when asserted (low pulse width adjustment) 3: For 1.8V 0: For 3.3V
11:8	MDIO_TDSEL	**TX duty select** TDSEL[1:0]: Output level shifter duty high when asserted (high pulse width adjustment) TDSEL[3:2]: Output level shifter duty low when asserted(low pulse width adjustment) 10: For3.3V 0: For 1.8V
5:4	MDIO_E4_E2	**TX Driving Strength Control.** 3: 8mA 2: 6mA 1: 4mA 0: 2mA
3	MDIO_PU	**75K pull-up resistor control.** 1: Enable 0: Disable
2	MDIO_PD	**75K pull-down resistor control.** 1: Enable 0: Disable

1	MDIO_SMT	**RX input buffer schmit trigger hysteresis control enable.** 1: Enable 0: Disable
0	MDIO_SR	**Output Slew Rate Control.** 1: Slower slew. 0: No slew rate controlled.

1E00005C PAD_SDXC_SPI_CFG PAD configuration of SDXC and SPI RST groups 0A210A18

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			SDXC_RDSEL		SDXC_TDSEL						SDXC_E4_E2		SDXC_PU	SDXC_PD	SDXC_SMT	SDXC_SR
Type			RW		RW						RW		RW	RW	RW	RW
Reset			0	0	1	0	1	0			1	0	0	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			SPI_RDSEL		SPI_TDSEL						SPI_E4_E2		SPI_PU	SPI_PD	SPI_SMT	SPI_SR
Type			RW		RW						RW		RW	RW	RW	RW
Reset			0	0	1	0	1	0			0	1	1	0	0	0

Bit(s)	Name	Description
29:28	SDXC_RDSEL	**RX duty select** RDSEL[0]: Level shifter duty high when asserted (high pulse width adjustment) RDSEL[1]: Level shifter duty low when asserted (low pulse width adjustment) 3: For 1.8V 0: For 3.3V
27:24	SDXC_TDSEL	**TX duty select** TDSEL[1:0]: Output level shifter duty high when asserted (high pulse width adjustment) TDSEL[3:2]: Output level shifter duty low when asserted(low pulse width adjustment) 10: For3.3V 0: For 1.8V
21:20	SDXC_E4_E2	**TX Driving Strength Control.** 3: 8mA 2: 6mA 1: 4mA 0: 2mA
19	SDXC_PU	**75K pull-up resistor control.** 1: Enable 0: Disable
18	SDXC_PD	**75K pull-down resistor control.** 1: Enable 0: Disable
17	SDXC_SMT	**RX input buffer schmit trigger hysteresis control enable.** 1: Enable 0: Disable
16	SDXC_SR	**Output Slew Rate Control.** 1: Slower slew. 0: No slew rate controlled.
13:12	SPI_RDSEL	**RX duty select** RDSEL[0]: Level shifter duty high when asserted (high pulse width adjustment) RDSEL[1]: Level shifter duty low when asserted (low pulse width adjustment) 3: For 1.8V 0: For 3.3V

11:8	SPI_TDSEL	**TX duty select** TDSEL[1:0]: Output level shifter duty high when asserted (high pulse width adjustment) TDSEL[3:2]: Output level shifter duty low when asserted(low pulse width adjustment) 10: For3.3V 0: For 1.8V
5:4	SPI_E4_E2	**TX Driving Strength Control.** 3: 8mA 2: 6mA 1: 4mA 0: 2mA
3	SPI_PU	**75K pull-up resistor control.** 1: Enable 0: Disable
2	SPI_PD	**75K pull-down resistor control.** 1: Enable 0: Disable
1	SPI_SMT	**RX input buffer schmit trigger hysteresis control enable.** 1: Enable 0: Disable
0	SPI_SR	**Output Slew Rate Control.** 1: Slower slew. 0: No slew rate controlled.

1E000060 GPIO_MODE GPIO purpose selection 0004D42C

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name												ESWINT_MODE	SDXC_MODE		SPI_MODE	
Type												RW	RW		RW	
Reset												0	0	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RGMII2_MODE	RGMII1_MODE	MDIO_MODE		PERST_MODE		WDT_MODE		JTAG_MODE	UART2_MODE		UART3_MODE		I2C_MODE	UART1_MODE	
Type	RW	RW	RW		RW		RW		RW	RW		RW		RW	RW	
Reset	1	1	0	1	0	1	0	0	0	0	1	0	1	1	0	

Bit(s)	Name	Description
20	ESWINT_MODE	**Ether switch interrupt GPIO mode** 1: GPIO 0: Ether switch interrupt
19:18	SDXC_MODE	**SDXC GPIO mode** 3: NAND 2: NAND 1: GPIO 0: SDXC
17:16	SPI_MODE	**SPI GPIO mode** 3: NAND 2: NAND 1: GPIO 0: SPI
15	RGMII2_MODE	**RGMII2 GPIO mode**

		1: GPIO 0: RGMII2
14	RGMII1_MODE	**RGMII1 GPIO mode** 1: GPIO 0: RGMII1
13:12	MDIO_MODE	**MDC/MDIO GPIO mode** 3: GPIO 2: GPIO 1: GPIO 0: MDIO
11:10	PERST_MODE	**PCIe reset GPIO mode** 3: Reference clock 2: Reference clock 1: GPIO 0: PCIe reset
9:8	WDT_MODE	**Watch dog timeout GPIO mode** 3: Reference clock 2: Reference clock 1: GPIO 0: Watch dog
7	JTAG_MODE	**JTAG GPIO mode** 1: GPIO 0: JTAG
6:5	UART2_MODE	**UART2 GPIO mode** 3: GPIO 2: PCM 1: GPIO 0: UART2
4:3	UART3_MODE	**UART3 GPIO mode** 3: SPDIF 2: I2S 1: GPIO 0: UART3
2	I2C_MODE	**I2C GPIO mode** 1: GPIO 0: I2C
1	UART1_MODE	**UART1 GPIO mode** 1: GPIO 0: UART1

1E000068 MEMO1 Memory1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MEMO1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MEMO1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MEMO1	**Memory1**

1E00006C MEMO2 Memory2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MEMO2[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MEMO2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MEMO2	**Memory2**

1E000070 PAD_BOPT_ESWINT_CFG PAD configuration of Bonding OPT and ESW INT groups 0A000A00

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			BOPT_RDSEL		BOPT_TDSEL						BOPT_E4_E2		BOPT_PU	BOPT_PD	BOPT_SMT	BOPT_SR
Type			RW		RW						RW		RW	RW	RW	RW
Reset			0	0	1	0	1	0			0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			ESW_RDSEL		ESW_TDSEL						ESW_E4_E2		ESW_PU	ESW_PD	ESW_SMT	ESW_SR
Type			RW		RW						RW		RW	RW	RW	RW
Reset			0	0	1	0	1	0			0	0	0	0	0	0

Bit(s)	Name	Description
29:28	BOPT_RDSEL	**RX duty select** RDSEL[0]: Level shifter duty high when asserted (high pulse width adjustment) RDSEL[1]: Level shifter duty low when asserted (low pulse width adjustment) 3: For 1.8V 0: For 3.3V
27:24	BOPT_TDSEL	**TX duty select** TDSEL[1:0]: Output level shifter duty high when asserted (high pulse width adjustment) TDSEL[3:2]: Output level shifter duty low when asserted(low pulse width adjustment) 10: For3.3V 0: For 1.8V
21:20	BOPT_E4_E2	**TX Driving Strength Control.** 3: 8mA 2: 6mA 1: 4mA 0: 2mA
19	BOPT_PU	**75K pull-up resistor control.** 1: Enable 0: Disable
18	BOPT_PD	**75K pull-down resistor control.** 1: Enable 0: Disable
17	BOPT_SMT	**RX input buffer schmit trigger hysteresis control enable.** 1: Enable

		0: Disable
16	BOPT_SR	**Output Slew Rate Control.** 1: Slower slew. 0: No slew rate controlled.
13:12	ESW_RDSEL	**RX duty select** RDSEL[0]: Level shifter duty high when asserted (high pulse width adjustment) RDSEL[1]: Level shifter duty low when asserted (low pulse width adjustment) 3: For 1.8V 0: For 3.3V
11:8	ESW_TDSEL	**TX duty select** TDSEL[1:0]: Output level shifter duty high when asserted (high pulse width adjustment) TDSEL[3:2]: Output level shifter duty low when asserted(low pulse width adjustment) 10: For3.3V 0: For 1.8V
5:4	ESW_E4_E2	**TX Driving Strength Control.** 3: 8mA 2: 6mA 1: 4mA 0: 2mA
3	ESW_PU	**75K pull-up resistor control.** 1: Enable 0: Disable
2	ESW_PD	**75K pull-down resistor control.** 1: Enable 0: Disable
1	ESW_SMT	**RX input buffer schmit trigger hysteresis control enable.** 1: Enable 0: Disable
0	ESW_SR	**Output Slew Rate Control.** 1: Slower slew. 0: No slew rate controlled.

1E000074 PAD_RGMII1_CFG PAD configuration of RGMII1 group 00000005

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name							RGMII1_RDSEL						RGMII1_TDSEL			
Type							RW						RW			
Reset							0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RGMII1_DRVP				RGMII1_DRVN					RGMII1_RTT				RGMII1_PDB	RMII1_PD	RGMII1_SR
Type	RW				RW					RW				RW	RW	RW
Reset	0	0	0	0	0	0	0	0		0	0	0		1	0	1

Bit(s)	Name	Description
25:20	RGMII1_RDSEL	**RX duty select** RDSEL[1:0]: Input buffer duty high when asserted (high pulse width adjustment) RDSEL[3:2]: Input buffer duty low when asserted(low pulse width adjustment) RDSEL4: Level shifter duty high when asserted (high pulse width adjustment) RDSEL5: Level shifter duty low when asserted (low pulse width adjustment) DDR3: RDSEL[5:0]=[000000] DDR2: RDSEL[5:0]=[000001]

19:16	RGMII1_TDSEL	**TX duty select** TDSEL[1:0]: Output level shifter duty high when asserted (high pulse width adjustment) TDSEL[3:2]: Output level shifter duty low when asserted(low pulse width adjustment)
15:12	RGMII1_DRVP	**GDDR3/DDR2 Pull-Up Driving Strength Control.** 00000: weakest. 11111: strongest. Default(Typical): GDDR3/POD18:DRVP[3:0]=[0100] (90 Ohm) DDR2/SSTL18: DRVP[3:0]=[1110] (40 Ohm) DDR3/SSTL15: DRVP[3:0]=[1111] (40 Ohm) 3: Strongest. 0: Weakest.
11:8	RGMII1_DRVN	**GDDR3/DDR2 Pull-Down Driving Strength Control.** 00000: weakest. 11111: strongest. Default(Typical): GDDR3/POD18:DRVN[3:0]=[1110] (40 Ohm) DDR2/SSTL18: DRVN[3:0]=[1110] (40 Ohm) DDR3/SSTL15: DRVN[3:0]=[1111] (40 Ohm) 3: Strongest. 0: Weakest.
6:4	RGMII1_RTT	**GDDR3(POD18)/DDR2(SSTL18) On-Die-Termination.** *Suggest to turn-off RTT[2:0] when not in read mode for power saving. GDDR3(POD18) 1.Comply to JESD8-19(POD18) for ODT pull-up 60/120/240 ohm requirement 2.Comply to GDDR3-SDRAM requirement DDR2(SSTL18) Follow JESD79-2B EMRS(1) Programming for Address Field [A6,A2] Supported all range [A6,A2] ODT setting. GDDR3 mode: RTT[2:0]=[110], GDDR3 ODT pull-up 60ohm (or [100]) RTT[2:0]=[001], GDDR3 ODT pull-up 120ohm RTT[2:0]=[010], GDDR3 ODT pull-up 240ohm DDR2 mode: RTT[2:0]=[000], ODT disable,Default for DDR/LVTTL mode. RTT[2:0]=[001], DDR2 ODT 75ohm RTT[2:0]=[010], DDR2 ODT 150ohm RTT[2:0]=[011], DDR2 ODT 50ohm DDR3 mode: RTT[2:0]=[000], ODT disable,Default for DDR/LVTTL mode. RTT[2:0]=[001], DDR2 ODT 60ohm RTT[2:0]=[010], DDR2 ODT 120ohm RTT[2:0]=[100], DDR2 ODT 40ohm
2	RGMII1_PDB	**75K pull-down resistor control. Low activiate.** GDDR3/DDR2/DDR1 mode, PDB=1 to disable 75K pull-down resistors. 1: Disable 0: Enable
1	RMII1_PD	**GDDR3/DDR2/DDR input buffer Power Down mode.High asserted. PD=1, O=0.** At LVTTL mode, set IE=0,PD=1 At GDDR3/DDR2/DDR1 power down mode, set IE=0, PD=1 1: Enable 0: Disable
0	RGMII1_SR	**Output Slew Rate Control. High asserted.** 1: Slower slew. 0: No slew rate controlled.

1E000078 **CPE_ROSC_S** **0000000**

EL0 0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CPE_ROSC_SEL0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CPE_ROSC_SEL0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	CPE_ROSC_SEL0	**CPE ROSC cell selection bit 31 ~ 0**

1E00007C CPE_ROSC_SEL1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CPE_ROSC_SEL1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CPE_ROSC_SEL1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	CPE_ROSC_SEL1	**CPE ROSC cell selection bit 63 ~ 32**

1E000080 CPU_CPE_CNT0 CPU CPE counter 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CPU_DFD_CNT0[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CPU_DFD_CNT0[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	CPU_DFD_CNT0	**CPU DFD counter value bit 31 ~ 0**

1E000084 CPU_CPE_CNT1 CPU CPE counter 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									CPU_DFD_CNT1							

Type									RO							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7:0	CPU_DFD_CNT1	**CPU DFD counter value bit 39 ~ 32**

1E000088 CPU_CFG CPU configuration 55AA0021

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MBIST_BKGND															
Type	RW															
Reset	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name											CM_DFT_TARGET			RG_RW_SW_TGR	IT_TOITU	SI_SYNCTX_EN
Type											RW			RW	RW	RW
Reset											1	0		0	0	1

Bit(s)	Name	Description
31:16	MBIST_BKGND	**CPU MBISTA background pattern**
5:4	CM_DFT_TARGET	**CM default target** 2: IOCU 0: Memory
2	RG_RW_SW_TGR	**SW trigger DFD counter to start** 1: Enable 0: Disable
1	IT_TOITU	**CM arbitraction** 1: DRAM access priority - favor CPU cores, but RRB between two cores 0: DRAM access priority - CPU core and IOCU master port RRB
0	SI_SYNCTX_EN	**Bus support OCP SYNC command or not** 1: Enable 0: Disable

1E00008C CPU_MEM_CFG CPU memory delay, power down and sleep control 0000000A

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CPE_ROSC_EN	CPE_ROSC_SEL2			CPE_ROSC_OUT											
Type	RW	RW			RO											
Reset	0	0	0		0	0	0	0								
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CPU_RAM_PD	CPU_RAM_SLEEP											CPU_L2_DEL_SEL		CPU_L1_DEL_SEL	

Type	RW	RW											RW		RW	
Reset	0	0											1	0	1	0

Bit(s)	Name	Description
31	CPE_ROSC_EN	**CPE ROSC enable** 1: Enable 0: Disable
30:29	CPE_ROSC_SEL2	**CPE ROSC cell selection bit 65 ~ 64**
27:24	CPE_ROSC_OUT	**CPE ROSC output**
15	CPU_RAM_PD	**CPU RAM power down enable** 1: Enable 0: Disable
14	CPU_RAM_SLEEP	**CPU RAM sleep enable** 1: Enable 0: Disable
3:2	CPU_L2_DEL_SEL	**CPU L2 cache RAM delay selection**
1:0	CPU_L1_DEL_SEL	**CPU L1 cache RAM delay selection**

1E000090 FMTR_CFG0 Frequency meter configuration 0 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FMTR_CNT_LMT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FMTR_CK_SEL						FMTR_CK_DIV					FMTR_RST				FMTR_CNT_EN
Type	RW						RW					RW				RW
Reset	0	0	0	0			0	0				0				1

Bit(s)	Name	Description
31:16	FMTR_CNT_LMT	**Freq. meter counter limitation**
15:12	FMTR_CK_SEL	**Freq. meter DUT clock selection** 10: CPLL monitor clock 9: DDRPLL4 monitor clock 8: DDRPLL3 monitor clock 7: DDRPLL2 monitor clock 6: DDRPLL1 monitor clock 5: PCIe1 clock 4: PCIe clock 3: CPU clock 2: GMPLL clock 1: APLL clock 0: XTAL clock
9:8	FMTR_CK_DIV	**Freq. meter clock divider selection** 3: divided by 40 2: divided by 40 1: divided by 10 0: no divided
4	FMTR_RST	**Reset freq. meter** 1: Reset 0: Not reset

0	FMTR_CNT_EN	**Freq. meter counter enable** 1: Enable 0: Disable

1E000094 FMTR_CNT_MAX Frequency meter count maximum 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FMTR_CNT_MAX[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FMTR_CNT_MAX[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	FMTR_CNT_MAX	**Freq. meter counter maximum value**

1E000098 FMTR_CNT_MIN Frequency meter count minimum 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FMTR_CNT_MAX[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FMTR_CNT_MAX[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	FMTR_CNT_MAX	**Freq. meter counter minimum value**

1E00009C FMTR_CNT_VAL Frequency meter counter value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FMTR_CNT_VAL[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FMTR_CNT_VAL[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	FMTR_CNT_VAL	**Freq. meter counter final value**

2.3 Timer

2.3.1 Features

- Independent 1usec tick pre-scale for each timer.
- Independent interrupts for each timer.
- Two general-purpose timers and a watchdog timer. Watchdog timer resets system on time-out.
- Timer Modes
  - *Periodic*
    In periodic mode, the timer counts down to zero from the limited value. An interrupt is generated when the count is zero. After reaching zero, the limited value is reloaded into the timer and the timer counts down again. A limited value of zero disables the timer.
  - *Timeout*
    In timeout mode, the timer counts down to zero from the limited value. An interrupt is generated when the count is zero. In this mode, the ENABLE bit is reset when the timer reaches zero, stopping the counter.
  - *Watchdog*
    In watchdog mode, the timer counts down to zero from the limited value. If the load value is not reloaded or the timer is not disabled before the count is zero, the chip will be reset. When this occurs, every register in the chip is reset except the watchdog reset status bit WDRST in the RSTSTAT register in the system control block; it remains set to alert firmware of the timeout event when it re-executes its bootstrap.

2.3.2 Block Diagram

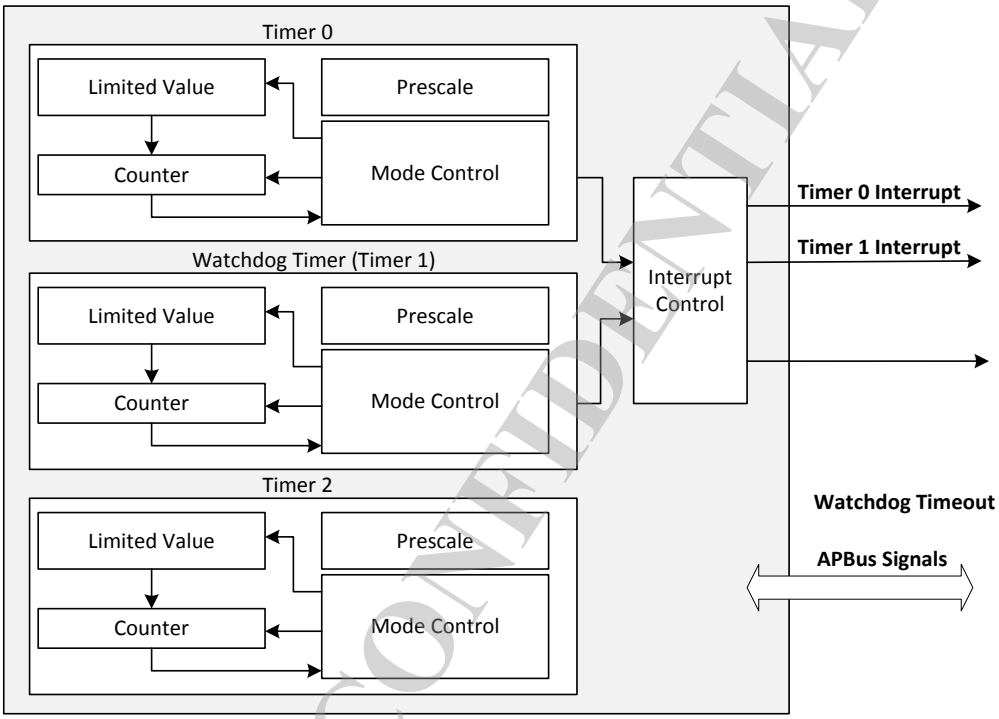


Figure 2-2 Timer Block Diagram

2.3.3 Registers

Address	Name	Width	Register Function
1E000100	TGLB_REG	32	RISC Global Control Register
1E000110	T0CTL_REG	32	RISC Timer 0 Control Register
1E000114	T0LMT_REG	32	RISC Timer 0 Limit Register
1E000118	T0_REG	32	RISC Timer 0 Register
1E000120	WDTCTL_REG	32	Watch Dog Timer Control Register
1E000124	WDTLMT_REG	32	Watch Dog Timer Limit Register
1E000128	WDT_REG	32	Watch Dog Timer Register
1E000130	T1CTL_REG	32	RISC Timer 1 Control Register
1E000134	T1LMT_REG	32	RISC Timer 1 Limit Register
1E000138	T1_REG	32	RISC Timer 1 Register

1E000100 TGLB_REG RISC Global Control Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV1[20:5]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESV1[4:0]					T1RST	WDTRST	T0RST	RESV0					T1INT	WDTINT	T0INT
Type	RW					RW	RW	RW	RW					RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:11	RESV1	**Reserved**
10	T1RST	**Timer 1 reset** 1: to reset timer 1 to T1LMT value
9	WDTRST	**Watch dog timer reset** 1: to reset watch dog timer to WDTLMT value
8	T0RST	**Timer 0 reset** 1: to reset timer 0 to T0LMT value
7:3	RESV0	**Reserved**
2	T1INT	**Timer 1 interrupt status**
1	WDTINT	**Watch dog timer interrupt status**
0	T0INT	**Timer 0 interrupt status**

1E000110 T0CTL_REG RISC Timer 0 Control Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	T0PRES															
Type	RW															

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESV2								T0EN	RESV1		T0AL	RESV0			
Type	RW								RW	RW		RW	DC			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	T0PRES	**Timer 0 count down tick pre-scale. Unit is 1u second.**
15:8	RESV2	**Reserved**
7	T0EN	**Timer 0 count down enable**
6:5	RESV1	**Reserved**
4	T0AL	**Timer 0 auto load enable** 1: Enable 0: Disable
3:0	RESV0	**Reserved**

1E000114 T0LMT_REG RISC Timer 0 Limit Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	T0LMT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV0	**Reserved**
15:0	T0LMT	**Timer 0 Limit.** When T0AL is set to 1, T0LMT will be loaded into timer 0 when timer 0 is enabled or when count down to 0.

1E000118 T0_REG RISC Timer 0 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	T0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV0	**Reserved**
15:0	T0	**RISC down-count timer 0**

1E000120 WDTCTL_REG Watch Dog Timer Control Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	WDTPRES															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESV2								WDTEN	RESV1		WDTAL	RESV0			
Type	RW								RW	RW		RW	DC			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	WDTPRES	**Watch dog timer count down tick pre-scale. Unit is 1u second.**
15:8	RESV2	**Reserved**
7	WDTEN	**Watch dog timer count down enable**
6:5	RESV1	**Reserved**
4	WDTAL	**Watch dog timer auto load enable** 1: Enable 0: Disable
3:0	RESV0	**Reserved**

1E000124 WDTLMT_REG Watch Dog Timer Limit Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	WDTLMT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV0	**Reserved**
15:0	WDTLMT	**Watch dog timer Limit.** When WDTAL is set to 1, WDTLMT will be loaded into watch dog timer when watch dog timer is enabled or when count down to 0.

1E000128 WDT_REG Watch Dog Timer Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	WDT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV0	**Reserved**
15:0	WDT	**watch dog timer.**

1E000130 T1CTL_REG RISC Timer 1 Control Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	T1PRES															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESV2								T1EN	RESV1		T1AL	RESV0			
Type	RW								RW	RW		RW	DC			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	T1PRES	**Timer 1 count down tick pre-scale. Unit is 1u second.**
15:8	RESV2	**Reserved**
7	T1EN	**Timer 1 count down enable**
6:5	RESV1	**Reserved**
4	T1AL	**Timer 1 auto load enable**
3:0	RESV0	**Reserved**

1E000134 T1LMT_REG RISC Timer 1 Limit Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	T1LMT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV0	**Reserved**
15:0	T1LMT	**Timer 1 Limit.** When T1AL is set to 1, T1LMT will be loaded into timer 1 when timer 1 is enabled or when count down to 0.

1E000138 T1_REG RISC Timer 1 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name	T1															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV0	**Reserved**
15:0	T1	**RISC down-count timer 1**

2.4 System Tick Counter

2.4.1 Registers

Address	Name	Width	Register Function
1E000500	STCK_CNT_CFG	32	**MIPS Configuration**
1E000504	CMP_CNT	32	**MIPS Compare** Sets the cutoff point for the free run counter (MIPS counter). If the free run counter equals the compare counter, then the timer circuit generates an interrupt. The interrupt remains active until the compare counter is written again.
1E000508	CNT	32	**MIPS Counter** The MIPS counter (free run counter) increases by 1 every 20 us (50 KHz). The counter continues to count until it reaches the value loaded into CMP_CNT.

1E000500 STCK_CNT_CFG MIPS Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV[29:14]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESV[13:0]														EXT_STK_EN	CNT_EN
Type	RW														RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:2	RESV	
1	EXT_STK_EN	**External System Tick Enable - Selects the system tick source.** 0: Use the MIPS internal timer interrupts. 1: Use the external timer interrupt from an external MIPS counter.
0	CNT_EN	**Counter Enable - Enable the free run counter (MIPS counter).** 0: Disable 1: Enable

1E000504 CMP_CNT MIPS Compare 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CMP_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV	
15:0	CMP_CNT	**Compare Count**

1E000508 **CNT** **MIPS Counter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV	
15:0	CNT	**MIPS Counter**

2.5 UART Lite

2.5.1 Features

- 2-pin UART
- 16550-compatible register set, except for Divisor Latch register
- 5-8 data bits
- 1-2 stop bits (1 or 2 stop bits are supported with 5 data bits)
- Even, odd, stick or no parity
- All standard baud rates up to 345600 b/s
- 16-byte receive buffer
- 16-byte transmit buffer
- Receive buffer threshold interrupt
- Transmit buffer threshold interrupt
- False start bit detection in asynchronous mode
- Internal diagnostic capabilities
- Break simulation
- Loop-back control for communications link fault isolation

2.5.2 Registers

n = 1; for uart1 only.

UARTn+0000h RX Buffer Register — UARTn_RBR

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									RBR[7:0]							
Type									RO							

RBR RX Buffer Register. Read-only register. The received data can be read by accessing this register.
Modified when LCR[7] = 0.

UARTn+0000h TX Holding Register — UARTn_THR

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									THR[7:0]							
Type									WO							

THR TX Holding Register. Write-only register. The data to be transmitted is written to this register, and then sent to the PC via serial communication.
Modified when LCR[7] = 0.

UARTn+0004h Interrupt Enable Register — UARTn_IER

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									CTSI	RTSI	XOFFI	X	EDSSI	ELSI	ETBEI	ERBFI
Type									R/W							
Reset									0							

IER By storing a '1' to a specific bit position, the interrupt associated with that bit is enabled. Otherwise, the interrupt is disabled.
IER[3:0] are modified when LCR[7] = 0.
IER[7:4] are modified when LCR[7] = 0 & EFR[4] = 1.

CTSI Masks an interrupt that is generated when a rising edge is detected on the CTS modem control line.
Note: This interrupt is only enabled when hardware flow control is enabled.
0 Unmask an interrupt that is generated when a rising edge is detected on the CTS modem control line.
1 Mask an interrupt that is generated when a rising edge is detected on the CTS modem control line.

RTSI Masks an interrupt that is generated when a rising edge is detected on the RTS modem control line.
Note: This interrupt is only enabled when hardware flow control is enabled.
0 Unmask an interrupt that is generated when a rising edge is detected on the RTS modem control line.
1 Mask an interrupt that is generated when a rising edge is detected on the RTS modem control line.

XOFFI Masks an interrupt that is generated when an XOFF character is received.
Note: This interrupt is only enabled when software flow control is enabled.
0 Unmask an interrupt that is generated when an XOFF character is received.
1 Mask an interrupt that is generated when an XOFF character is received.

EDSSI When set ("1"), an interrupt is generated if DDCD, TERI, DDSR or DCTS (MSR[4:1]) becomes set.
0 No interrupt is generated if DDCD, TERI, DDSR or DCTS (MSR[4:1]) becomes set.

1 An interrupt is generated if DDCD, TERI, DDSR or DCTS (MSR[4:1]) becomes set.

ELSI When set ("1"), an interrupt is generated if BI, FE, PE or OE (LSR[4:1]) becomes set.

0 No interrupt is generated if BI, FE, PE or OE (LSR[4:1]) becomes set.

1 An interrupt is generated if BI, FE, PE or OE (LSR[4:1]) becomes set.

ETBEI When set ("1"), an interrupt is generated if the TX Holding Register is empty or the contents of the TX FIFO

have been reduced to its Trigger Level.

0 No interrupt is generated if the TX Holding Register is empty or the contents of the TX FIFO have been reduced to its Trigger Level.

1 An interrupt is generated if the TX Holding Register is empty or the contents of the TX FIFO have been reduced to its Trigger Level

ERBFI When set ("1"), an interrupt is generated if the RX Buffer contains data.

0 No interrupt is generated if the RX Buffer contains data.

1 An interrupt is generated if the RX Buffer contains data.

UARTn+0008h Interrupt Identification Register UARTn_IIR

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									FIFOE		ID4	ID3	ID2	ID1	ID0	NINT
Type									RO							
Reset									0	0	0	0	0	0	0	1

IIR Identify if there are pending interrupts; ID4 and ID3 are presented only when EFR[4] = 1.

The following table gives the IIR[5:0] codes associated with the possible interrupts:

IIR[5:0]	Priority Level	Interrupt	Source
000001	-	No interrupt pending	
000110	1	Line Status Interrupt	BI, FE, PE or OE set in LSR
000100	2	RX Data Received	RX Data received or RX Trigger Level reached.
001100	2	RX Data Timeout	Timeout on character in RX FIFO.
000010	3	TX Holding Register Empty	TX Holding Register empty or TX FIFO Trigger Level reached.
000000	4	Modem Status change	DDCD, TERI, DDSR or DCTS set in MSR
010000	5	Software Flow Control	XOFF Character received
100000	6	Hardware Flow Control	CTS or RTS Rising Edge

Table 1 The IIR[5:0] codes associated with the possible interrupts

Line Status Interrupt: A RX Line Status Interrupt (IIR[5:0`] == 000110b) is generated if ELSI (IER[2]) is set and any of BI, FE, PE or OE (LSR[4:1]) becomes set. The interrupt is cleared by reading the Line Status Register.
RX Data Received Interrupt: A RX Received interrupt (IER[5:0] == 000100b) is generated if EFRBI (IER[0]) is set and either RX Data is placed in the RX Buffer Register or the RX Trigger Level is reached. The interrupt is cleared by reading the RX Buffer Register or the RX FIFO (if enabled).
RX Data Timeout Interrupt:
When virtual FIFO mode is disabled, RX Data Timeout Interrupt is generated if all of the following apply:
1. FIFO contains at least one character;

2. The most recent character was received longer than four character periods ago (including all start, parity and stop bits);

3. The most recent CPU read of the FIFO was longer than four character periods ago.

The timeout timer is restarted on receipt of a new byte from the RX Shift Register, or on a CPU read from the RX FIFO.
The RX Data Timeout Interrupt is enabled by setting EFRBI (IER[0]) to 1, and is cleared by reading RX FIFO.
When virtual FIFO mode is enabled, RX Data Timeout Interrupt is generated if all of the following apply:

1. FIFO is empty;

2. The most recent character was received longer than four character periods ago (including all start, parity and stop bits);

3. The most recent CPU read of the FIFO was longer than four character periods ago.

The timeout timer is restarted on receipt of a new byte from the RX Shift Register.
RX Holding Register Empty Interrupt: A TX Holding Register Empty Interrupt (IIR[5:0] = 000010b) is generated if ETRBI (IER[1]) is set and either the TX Holding Register or, if FIFOs are enabled, the TX FIFO becomes empty. The interrupt is cleared by writing to the TX Holding Register or TX FIFO if FIFO enabled.
Modem Status Change Interrupt: A Modem Status Change Interrupt (IIR[5:0] = 000000b) is generated if EDSSI (IER[3]) is set and either DDCD, TERI, DDSR or DCTS (MSR[3:0]) becomes set. The interrupt is cleared by reading the Modem Status Register.
Software Flow Control Interrupt: A Software Flow Control Interrupt (IIR[5:0] = 010000b) is generated if Software Flow Control is enabled and XOFFI (IER[5]) becomes set, indicating that an XOFF character has been received. The interrupt is cleared by reading the Interrupt Identification Register.
Hardware Flow Control Interrupt: A Hardware Flow Control Interrupt (IER[5:0] = 100000b) is generated if Hardware Flow Control is enabled and either RTSI (IER[6]) or CTSI (IER[7]) becomes set indicating that a rising edge has been detected on either the RTS/CTS Modem Control line. The interrupt is cleared by reading the Interrupt Identification Register.

UARTn+0008h FIFO Control Register UARTn_FCR

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									RFTL1	RFTL0	TFTL1	TFTL0	DMA1	CLRT	CLRR	FIFOE
Type									WO							

FCR FCR is used to control the trigger levels of the FIFOs, or flush the FIFOs.
FCR[7:6] is modified when LCR != BFh
FCR[5:4] is modified when LCR != BFh & EFR[4] = 1
FCR[4:0] is modified when LCR != BFh

FCR[7:6] RX FIFO trigger threshold
0 1
1 6
2 12
3 **RXTRIG**

FCR[5:4] TX FIFO trigger threshold
0 1
1 4

2 8

3 14 (FIFOSIZE - 2)

DMA1 This bit determines the DMA mode, which the TXRDY and RXRDY pins support. TXRDY and RXRDY act to support single-byte transfers between the UART and memory (DMA mode 0) or multiple byte transfers (DMA mode1). Note that this bit has no effect unless the FIFOE bit is set as well

0 The device operates in DMA Mode 0.

1 The device operates in DMA Mode 1.

TXRDY – mode0: Goes active (low) when the TX FIFO or the TX Holding Register is empty. Becomes inactive when a byte is written to the Transmit channel.

TXRDY – mode1: Goes active (low) when there are no characters in the TX FIFO. Becomes inactive when the TX FIFO is full.

RXRDY – mode0: Becomes active (low) when at least one character is in the RX FIFO or the RX Buffer Register is full. Becomes inactive when there are no more characters in the RX FIFO or RX Buffer register.

RXRDY – mode1: Becomes active (low) when the RX FIFO Trigger Level is reached or an RX FIFO Character Timeout occurs. Goes inactive when the RX FIFO is empty.

CLRT Clear Transmit FIFO. This bit is self-clearing.

0 Leave TX FIFO intact.

1 Clear all the bytes in the TX FIFO.

CLRR Clear Receive FIFO. This bit is self-clearing.

0 Leave RX FIFO intact.

1 Clear all the bytes in the RX FIFO.

FIFOE FIFO Enabled. This bit must be set to 1 for any of the other bits in the registers to have any effect.

0 Disable both the RX and TX FIFOs.

1 Enable both the RX and TX FIFOs.

UARTn+000Ch Line Control Register — UARTn_LCR

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									**DLAB**	**SB**	**SP**	**EPS**	**PEN**	**STB**	**WLS1**	**WLS0**
Type									R/W							
Reset									0	0	0	0	0	0	0	0

LCR Line Control Register. Determines characteristics of serial communication signals.
Modified when LCR[7] = 0.

DLAB Divisor Latch Access Bit.

0 The RX and TX Registers are read/written at Address 0 and the IER register is read/written at Address 4.

1 The Divisor Latch LS is read/written at Address 0 and the Divisor Latch MS is read/written at Address 4.

SB Set Break

0 No effect

1 SOUT signal is forced into the “0” state.

SP Stick Parity

0 No effect.

1 The Parity bit is forced into a defined state, depending on the states of EPS and PEN:
If EPS=1 & PEN=1, the Parity bit is set and checked = 0.
If EPS=0 & PEN=1, the Parity bit is set and checked = 1.

EPS Even Parity Select

0 When EPS=0, an odd number of ones is sent and checked.

1 When EPS=1, an even number of ones is sent and checked.

PEN Parity Enable

0 The Parity is neither transmitted nor checked.

1 The Parity is transmitted and checked.

STB Number of STOP bits

0 One STOP bit is always added.

1 Two STOP bits are added after each character is sent; unless the character length is 5 when 1 STOP bit is added.

WLS1, 0 Word Length Select.

0 5 bits

1 6 bits

2 7 bits

3 8 bits

UARTn+0010h Modem Control Register UARTn_MCR

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									XOFF STATUS		X	DCM_EN	OUT2	OUT1	RTS	DTR
Type									R/W							
Reset									0		0	0	0	0	0	0

MCR Modem Control Register. Control interface signals of the UART.
MCR[4:0] are modified when LCR[7] = 0,
MCR[7:6] are modified when LCR[7] = 0 & EFR[4] = 1.

XOFF Status This is a read-only bit.

0 When an XON character is received.

1 When an XOFF character is received.

DCM_EN UART DCM function enable bit

0 UART DCM is disabled.

1 UART DCM is enabled.

OUT2 Controls the state of the output NOUT2, even in loop mode.

0 NOUT2=1.

1 NOUT2=0.

OUT1 Controls the state of the output NOUT1, even in loop mode.

0 NOUT1=1.

1 NOUT1=0.

RTS Controls the state of the output NRTS, even in loop mode.

0 NRTS=1.

1 NRTS=0.

DTR Control the state of the output NDTR, even in loop mode.

0 NDTR=1.

1 NDTR=0.

UARTn+0014h Line Status Register UARTn_LSR

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									FIFOERR	TEMT	THRE	BI	FE	PE	OE	DR
Type									R/W							
Reset									0	1	1	0	0	0	0	0

LSR Line Status Register.

Modified when LCR[7] = 0.

FIFOERR RX FIFO Error Indicator.

0 No PE, FE, BI set in the RX FIFO.

1 Set to 1 when there is at least one PE, FE or BI in the RX FIFO.

TEMT TX Holding Register (or TX FIFO) and the TX Shift Register are empty.

0 Empty conditions below are not met.

1 If FIFOs are enabled, the bit is set whenever the TX FIFO and the TX Shift Register are empty. If FIFOs are disabled, the bit is set whenever TX Holding Register and TX Shift Register are empty.

THRE Indicates if there is room for TX Holding Register or TX FIFO is reduced to its Trigger Level.

0 **Reset whenever the contents of the TX FIFO are more than its Trigger Level (FIFOs are enabled), or whenever TX Holding Register is not empty(FIFOs are disabled).**

1 Set whenever the contents of the TX FIFO are reduced to its Trigger Level (FIFOs are enabled), or whenever TX Holding Register is empty and ready to accept new data (FIFOs are disabled).

BI Break Interrupt.

0 Reset by the CPU reading this register

1 If the FIFOs are disabled, this bit is set whenever the SIN is held in the 0 state for more than one transmission time (START bit + DATA bits + PARITY + STOP bits).
If the FIFOs are enabled, this error is associated with a corresponding character in the FIFO and is flagged when this byte is at the top of the FIFO. When a break occurs, only one zero character is loaded into the FIFO: the next character transfer is enabled when SIN goes into the marking state and receives the next valid start bit.

FE Framing Error.

0 Reset by the CPU reading this register

1 If the FIFOs are disabled, this bit is set if the received data did not have a valid STOP bit. If the FIFOs are enabled, the state of this bit is revealed when the byte it refers to is the next to be read.

PE Parity Error

0 Reset by the CPU reading this register

1 If the FIFOs are disabled, this bit is set if the received data did not have a valid parity bit. If the FIFOs are enabled, the state of this bit is revealed when the referred byte is the next to be read.

OE Overrun Error.

0 Reset by the CPU reading this register.

1 If the FIFOs are disabled, this bit is set if the RX Buffer was not read by the CPU before new data from the RX Shift Register overwrote the previous contents.
If the FIFOs are enabled, an overrun error occurs when the RX FIFO is full and the RX Shift Register becomes full. OE is set as soon as this happens. The character in the Shift Register is then overwritten, but not transferred to the FIFO.

DR Data Ready.

0 Cleared by the CPU reading the RX Buffer or by reading all the FIFO bytes.

1 Set by the RX Buffer becoming full or by a byte being transferred into the FIFO.

UARTn+0018h Modem Status Register UARTn_MSR

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									**DCD**	**RI**	**DSR**	**CTS**	**DDCD**	**TERI**	**DDSR**	**DCTS**
Type									R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset									Input	Input	Input	Input	0	0	0	0

Note: After a reset, D4-D7 are inputs. A modem status interrupt can be cleared by writing '0' or set by writing '1' to this register. D0-D3 can be written to.
Modified when LCR[7] = 0.

MSR Modem Status Register

DCD Data Carry Detect.

When Loop = "0", this value is the complement of the NDCD input signal.
When Loop = "1", this value is equal to the OUT2 bit in the Modem Control Register.

RI Ring Indicator.

When Loop = "0", this value is the complement of the NRI input signal.
When Loop = "1", this value is equal to the OUT1 bit in the Modem Control Register.

DSR Data Set Ready

When Loop = "0", this value is the complement of the NDSR input signal.
When Loop = "1", this value is equal to the DTR bit in the Modem Control Register.

CTS Clear To Send.

When Loop = "0", this value is the complement of the NCTS input signal.
When Loop = "1", this value is equal to the RTS bit in the Modem Control Register.

DDCD Delta Data Carry Detect.

0 The state of DCD has not changed since the Modem Status Register was last read

1 Set if the state of DCD has changed since the Modem Status Register was last read.

TERI Trailing Edge Ring Indicator

0 The NRI input does not change since this register was last read.

1 Set if the NRI input changes from "0" to "1" since this register was last read.

DDSR Delta Data Set Ready

0 Cleared if the state of DSR has not changed since this register was last read.

1 Set if the state of DSR has changed since this register was last read.

DCTS Delta Clear To Send

0 Cleared if the state of CTS has not changed since this register was last read.

1 Set if the state of CTS has changed since this register was last read.

UARTn+001Ch Scratch Register — UARTn_SCR

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									SCR[7:0]							
Type									R/W							

A general purpose read/write register. After reset, its value is un-defined.
Modified when LCR[7] = 0.

UARTn+0000h Divisor Latch (LS) — UARTn_DLL

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DLL[7:0]							
Type									R/W							
Reset									1							

UARTn+0004h Divisor Latch (MS) — UARTn_DLM

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DLL[7:0]							
Type									R/W							
Reset									0							

Note: DLL & DLM can only be updated if DLAB is set ("1").. Note too that division by 1 generates a BAUD signal that is constantly high.
Modified when LCR[7] = 1.
The table below shows the divisor needed to generate a given baud rate from CLK inputs of 13, 26 MHz and 52 MHz. The effective clock enable generated is 16 x the required baud rate.

BAUD	13MHz	26MHz	52MHz
110	7386	14773	29545
300	2708	5417	10833
1200	677	1354	2708
2400	338	677	1354
4800	169	339	677
9600	85	169	339
19200	42	85	169
38400	21	42	85
57600	14	28	56
115200	6	14	28

Table 2 Divisor needed to generate a given baud rate

UARTn+0008h Enhanced Feature Register — UARTn_EFR

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									AUTO CTS	AUTO RTS	D5	ENABLE -E	SW FLOW CONT[3:0]			
Type									R/W	R/W	R/W	R/W	R/W			
Reset									0	0	0	0	0			

*NOTE: Only when LCR=BF'h

Auto CTS Enables hardware transmission flow control
- **0** Disabled.
- **1** Enabled.

Auto RTS Enables hardware reception flow control
- **0** Disabled.
- **1** Enabled.

Enable-E Enable enhancement features.
- **0** Disabled.
- **1** Enabled.

CONT[3:0] Software flow control bits.
- **00xx** No TX Flow Control
- **10xx** Transmit XON1/XOFF1 as flow control bytes
- **01xx** Transmit XON2/XOFF2 as flow control bytes
- **11xx** Transmit XON1 & XON2 and XOFF1 & XOFF2 as flow control words
- **xx00** No RX Flow Control
- **xx10** Receive XON1/XOFF1 as flow control bytes
- **xx01** Receive XON2/XOFF2 as flow control bytes
- **xx11** Receive XON1 & XON2 and XOFF1 & XOFF2 as flow control words

UARTn+0010h XON1 UARTn_XON1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									**XON1[7:0]**							
Type									R/W							
Reset									0							

UARTn+0014h XON2 UARTn_XON2

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									**XON2[7:0]**							
Type									R/W							
Reset									0							

UARTn+0018h XOFF1 UARTn_XOFF1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									**XOFF1[7:0]**							
Type									R/W							
Reset									0							

UARTn+001Ch XOFF2 UARTn_XOFF2

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									**XOFF2[7:0]**							
Type									R/W							
Reset									0							

UARTn+0024h HIGH SPEED UART UARTn_HIGHSPEED

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															SPEED [1:0]	
Type															R/W	
Reset															0	

SPEED UART sample counter base

- **0** based on 16*baud_pulse, baud_rate = system clock frequency/16/{DLH, DLL}
- **1** based on 8*baud_pulse, baud_rate = system clock frequency/8/{DLH, DLL}
- **2** based on 4*baud_pulse, baud_rate = system clock frequency/4/{DLH, DLL}
- **3** based on sampe_count * baud_pulse, baud_rate = system clock frequency / sampe_count

When HIGHSPEED=3, the value (A * B) means ({DLM, DLL} * SAMPLE_COUNT).
When the Baudrate is more than 115200, it will be more accurate if we set HIGHSPEED=3.
The table below shows the divisor needed to generate a given baud rate from CLK inputs of 13M Hz based on different HIGHSPEED value.

BAUD	HIGHSPEED = 0	HIGHSPEED = 1	HIGHSPEED = 2	HIGHSPEED = 3
110	7386	14773	29545	7386 * 16
300	2708	7386	14773	2708 * 16
1200	677	2708	7386	677 * 16
2400	338	677	2708	338 * 16
4800	169	338	677	169 * 16
9600	85	169	338	85 * 16
19200	42	85	169	9 * 75
38400	21	42	85	13 * 26
57600	14	21	42	8 * 28
115200	7	14	21	4 * 28
230400	*	7	14	2 * 28
460800	*	*	7	1 * 28
921600	*	*	*	1 * 14

Table 3 *Divisor needed to generate a given baud rate from 13MHz based on different HIGHSPEED value*

The table below shows the divisor needed to generate a given baud rate from CLK inputs of 26 MHz based on different HIGHSPEED value.

BAUD	HIGHSPEED = 0	HIGHSPEED = 1	HIGHSPEED = 2	HIGHSPEED = 3
110	14773	29545	59091	7386 * 32
300	5417	14773	29545	2708 * 32
1200	1354	5417	14773	677 * 32
2400	677	1354	5417	338 * 32
4800	339	677	1354	169 * 32

9600	169	339	667	85 * 32
19200	85	169	339	18 * 75
38400	42	85	169	26 * 26
57600	28	42	85	16 * 28
115200	14	28	42	8 * 28
230400	7	14	28	4 * 28
460800	*	7	14	2 * 28
921600	*	*	7	1 * 28

Table 4 *Divisor needed to generate a given baud rate from 26 MHz based on different HIGHSPEED value*

The table below shows the divisor needed to generate a given baud rate from CLK inputs of 52MHz based on different HIGHSPEED value.

BAUD	HIGHSPEED = 0	HIGHSPEED = 1	HIGHSPEED = 2	HIGHSPEED = 3
110	29545	59091	118182	14773 * 32
300	10833	29545	59091	5417 * 32
1200	2708	10833	29545	1354 * 32
2400	1354	2708	10833	667 * 32
4800	677	1354	2708	339 * 32
9600	339	677	1354	169 * 32
19200	169	339	677	36 * 75
38400	85	169	339	52 * 26
57600	56	85	169	32 * 28
115200	28	56	85	16 * 28
230400	14	28	56	8 * 28
460800	7	14	28	4 * 28
921600	*	7	14	2 * 28

Table 5 Divisor needed to generate a given baud rate from 52 MHz based on different HIGHSPEED value

UARTn+0028h SAMPLE_COUNT UARTn_SAMPLE_COUNT

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									SAMPLECOUNT [7:0]							
Type									R/W							
Reset									0							

When HIGHSPEED=3, the sample_count is the threshold value for UART sample counter (sample_num).

Count from 0 to sample_count.

UARTn+002Ch SAMPLE_POINT — UARTn_SAMPLE_POINT

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									SAMPLEPOINT [7:0]							
Type									R/W							
Reset									Ffh							

When HIGHSPEED=3, UART gets the input data when sample_count=sample_num.
e.g. system clock = 13MHz, 921600 = 13000000 / 14
sample_count = 14 and sample point = 7 (sample the central point to decrease the inaccuracy)
The SAMPLE_POINT is usually (SAMPLE_COUNT/2).

UARTn+0034h Rate Fix Address — UARTn_RATEFIX_AD

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																RXTE_FIX
Type																R/W
Reset																0

rate_fix When you set "rate_fix"(34H[0]), you can transmit and receive data only if the input f16m_en is enable.

UARTn+003Ch Guard time added register — UARTn_GUARD

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												GUARD_EN	GUARD_CNT[3:0]			
Type												R/W	R/W	R/W	R/W	R/W
Reset												0	0	0	0	0

GUARD_CNT Guard interval count value. Guard interval = (1/(system clock / **div_step** / div)) * GUARD_CNT.

GUARD_EN Guard interval add enable signal.

- **0** No guard interval added.
- **1** Add guard interval after stop bit.

UARTn+0040h Escape character register — UARTn_ESCAPE_DAT

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									ESCAPE_DAT[7:0]							
Type									WO							
Reset									FFh							

ESCAPE_DAT Escape character added before software flow control data and escape character, i.e. if tx data is xon (31h), with esc_en =1, uart transmits data as esc + CEh (~xon).

UARTn+0044h Escape enable register — UARTn_ESCAPE_EN

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																ESC_EN
Type																R/W

Reset																0

ESC_EN Add escape character in transmitter and remove escape character in receiver by UART.

0 Do not deal with the escape character.

1 Add escape character in transmitter and remove escape character in receiver.

UARTn+0048h Sleep enable register UARTn_SLEEP_EN

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																SELLP_EN
Type																R/W
Reset																0

SLEEP_EN For sleep mode issue

0 Do not deal with sleep mode indicate signal

1 To activate hardware flow control or software control according to software initial setting when chip enters sleep mode. Releasing hardware flow when chip wakes up; but for software control, uart sends xon when awaken and when FIFO does not reach threshold level.

UARTn+004Ch Virtual FIFO enable register UARTn_VFIFO_EN

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																VFIFO_EN
Type																R/W
Reset																0

VFIFO_EN Virtual FIFO mechanism enable signal.

0 Disable VFIFO mode.

1 Enable VFIFO mode. When virtual mode is enabled, the flow control is based on the DMA threshold, and generates a timeout interrupt for DMA.

UARTn+0050h Rx Trigger Address UARTn_RXTRI_AD

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													RXTRIG[3:0]			
Type													R/W			
Reset													0			

RXTRIG **When {rtm,rtl}=2'b11, The Rx FIFO threshold will be Rxtrig.**

UARTn+0054h Fractional Divider LSB Address UARTn_FRACDIV_L

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									FRACDIV_L							
Type									R/W							
Reset									0	0	0	0	0	0	0	0

FRACDIV_L Add sampling count (+1) from state data7 to state data0, in order to contribute fractional divisor.

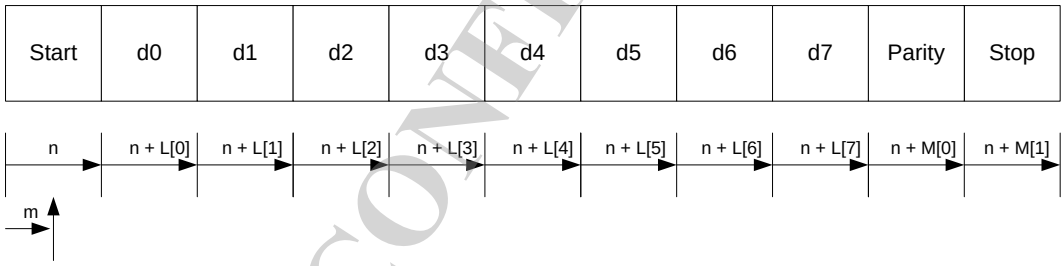
UARTn+0058h Fractional Divider MSB Address — UARTn_FRACDIV_M

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															FRACDIV_M	
Type															R/W	
Reset															0	0

FRACDIV_M Add sampling count in state stop and state parity, in order to contribute fractional divisor.

FRACDIV_L / FRACDIV_L Add one sampling period to each symbol, in order to increase the baud rate accuracy.

bit_extend register = FRACDIV_L[7:0]
FRACDIV_M[1:0]



UARTn+005Ch FIFO Control Register — UARTn_FCR_RD

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									RFTL1	RFTL0	TFTL1	TFTL0	DMA1			FIFOE
Type									RO							RO

Read out UARTn_FCR register.

UARTn+0060h TX Active Enable Address — UARTn_TX_ACTIVE_EN

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															TX_PU_EN	TX_OE_EN
Type															R/W	R/W
Reset															0	0

TX_OE_EN Enable UART_TX_OE switching function. TX_OE is to control UART_TX output enable.

TX_PU_EN Enable UART_TX_PU switching function. TX_PU is to control UART_TX pull up enable.

2.6 Programmable I/O

2.6.1 Features

- Parameterized numbers of independent inputs, outputs, and inouts
- Independent polarity controls for each pin
- Independently masked edge detect interrupt on any input transition

2.6.2 Block Diagram

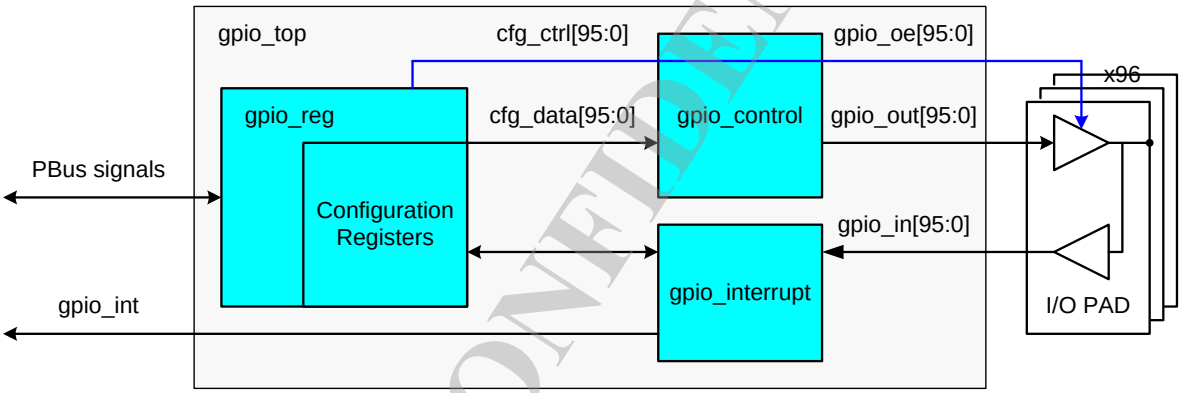


Figure 2-3 Programmable I/O Block Diagram

2.6.3 GPIO pin mapping

PAD Name	Function 0	Function 1	Function 2	Function 3	strap	pmux_group	GPIO
PAD_GPIO0	gpio (I/O)				0	gpio_psel[0]	0
PAD_RXD1	rxd1 (I)	gpio (I/O)				uartl_psel[0]	1
PAD_TXD1	txd1 (O)	gpio (I/O)			1		2
PAD_I2C_SD	i2c_sd (I/O)	gpio (I/O)				i2c_psel[0]	3
PAD_I2C_SCLK	i2c_sclk (I/O)	gpio (I/O)					4
PAD_RTS3_N	rts3_n (O)	gpio (I/O)	i2s_sdo (O)	spdif_tx (O)	2	uart3_psel[1:0]	5
PAD_CTS3_N	cts3_n (I)	gpio (I/O)	i2s_clk (I/O)	gpio (I/O)			6
PAD_TXD3	txd3 (O)	gpio (I/O)	i2s_ws (I/O)	gpio (I/O)			7
PAD_RXD3	rxd3 (I)	gpio (I/O)	i2s_sdi (I)	gpio (I/O)			8
PAD_RTS2_N	rts2_n (O)	gpio (I/O)	pcm_dtx (I/O)	gpio (I/O)	3	uart2_psel[1:0]	9
PAD_CTS2_N	cts2_n (I)	gpio (I/O)	pcm_drx (I)	gpio (I/O)			10
PAD_TXD2	txd2 (O)	gpio (I/O)	pcm_clk (O)	spdif_tx (O)	4		11
PAD_RXD2	rxd2 (I)	gpio (I/O)	pcm_fs (I/O)	gpio (I/O)			12
PAD_JTDO	jtdo (I/O)	gpio (I/O)				jtag_psel[0]	13
PAD_JTDI	jtdi (I)	gpio (I/O)					14
PAD_JTMS	jtms (I)	gpio (I/O)					15
PAD_JTCLK	jtclk (I)	gpio (I/O)					16
PAD_JTRST_N	jtrst_n (I)	gpio (I/O)					17
PAD_WDT_RST_N	wdt_rst_n (I/O)	gpio (I/O)	ref_clk0_out (O)			wdt_psel[1:0]	18
PAD_PERST_N	perst_n (O)	gpio (I/O)	ref_clk0_out (O)		5	perst_psel[1:0]	19

PAD_MDIO	mdio (I/O)	gpio (I/O)	gpio (I/O)			mdio_psel[1:0]	20
PAD_MDC	mdc (O)	gpio (I/O)	ref_clk0_out (O)		6		21
PAD_G1_TXD0	g1_txd[0] (I/O)	gpio (I/O)				rgmii1_psel[0]	49
PAD_G1_TXD1	g1_txd[1] (I/O)	gpio (I/O)					50
PAD_G1_TXD2	g1_txd[2] (I/O)	gpio (I/O)					51
PAD_G1_TXD3	g1_txd[3] (I/O)	gpio (I/O)					52
PAD_G1_TXEN	g1_txen (I/O)	gpio (I/O)					53
PAD_G1_TXC	g1_txc (I/O)	gpio (I/O)					54
PAD_G1_RXD0	g1_rxd[0] (I/O)	gpio (I/O)					55
PAD_G1_RXD1	g1_rxd[1] (I/O)	gpio (I/O)					56
PAD_G1_RXD2	g1_rxd[2] (I/O)	gpio (I/O)					57
PAD_G1_RXD3	g1_rxd[3] (I/O)	gpio (I/O)					58
PAD_G1_RXDV	g1_rxdv (I/O)	gpio (I/O)					59
PAD_G1_RXC	g1_rxc (I/O)	gpio (I/O)					60
PAD_G2_TXD0	g2_txd[0] (I/O)	gpio (I/O)				rgmii2_psel[0]	22
PAD_G2_TXD1	g2_txd[1] (I/O)	gpio (I/O)					23
PAD_G2_TXD2	g2_txd[2] (I/O)	gpio (I/O)					24
PAD_G2_TXD3	g2_txd[3] (I/O)	gpio (I/O)					25
PAD_G2_TXEN	g2_txen (I/O)	gpio (I/O)					26
PAD_G2_TXC	g2_txc (I/O)	gpio (I/O)					27
PAD_G2_RXD0	g2_rxd[0] (I/O)	gpio (I/O)					28
PAD_G2_RXD1	g2_rxd[1] (I/O)	gpio (I/O)					29
PAD_G2_RXD2	g2_rxd[2] (I/O)	gpio (I/O)					30
PAD_G2_RXD3	g2_rxd[3] (I/O)	gpio (I/O)					31
PAD_G2_RXDV	g2_rxdv (I/O)	gpio (I/O)					32
PAD_G2_RXC	g2_rxc (I/O)	gpio (I/O)					33
PAD_SPI_CS0_N	spi_cs0 (I/O)	gpio (I/O)	nd_cs_n (O)		7	spi_psel[1:0]	34
PAD_SPI_CS1_N	spi_cs1 (I/O)	gpio (I/O)	nd_we_n (O)		8		35
PAD_SPI_SCLK	spi_clk (I/O)	gpio (I/O)	nd_re_n (O)		9		36
PAD_SPI_MISO	spi_miso (I/O)	gpio (I/O)	nd_d[4] (I/O)				37
PAD_SPI_MOSI	spi_mosi (I/O)	gpio (I/O)	nd_d[5] (I/O)				38
PAD_SPI_WP_N	spi_wp (I/O)	gpio (I/O)	nd_d[6] (I/O)				39
PAD_SPI_HOLD_N	spi_hold (I/O)	gpio (I/O)	nd_d[7] (I/O)				40
PAD_SD_WP	sd_wp (I)	gpio (I/O)	nd_wp (O)			sd_psel[1:0]	41
PAD_SD_CLK	sd_clk (I/O)	gpio (I/O)	nd_rb_n (I)				42
PAD_SD_CD	sd_cd (I)	gpio (I/O)	nd_cle (O)				43
PAD_SD_CMD	sd_cmd (I/O)	gpio (I/O)	nd_ale (O)				44
PAD_SD_D0	sd_data[0] (I/O)	gpio (I/O)	nd_d[0] (I/O)				45
PAD_SD_D1	sd_data[1] (I/O)	gpio (I/O)	nd_d[1] (I/O)				46
PAD_SD_D2	sd_data[2] (I/O)	gpio (I/O)	nd_d[2] (I/O)				47
PAD_SD_D3	sd_data[3] (I/O)	gpio (I/O)	nd_d[3] (I/O)				48
PAD_ESW_INT	esw_int(I)	gpio (I/O)				esw_psel[0]	61

2.6.4 Registers

Module name: GPIO Base address: (+1E000600h)

Address	Name	Width	Register Function
1E000600	GPIO_CTRL_0	32	**GPIO0 to GPIO31 direction control register** These direction control registers are used to select the data direction of the GPIO pin. The value driven onto the GPIO pins, are controlled by the GPIO_POL_x, and GPIO_DATA_x registers.
1E000604	GPIO_CTRL_1	32	**GPIO32 to GPIO63 direction control register** These direction control registers are used to select the data direction of the GPIO pin. The value driven onto the GPIO pins, are controlled by the GPIO_POL_x, and GPIO_DATA_x registers.
1E000608	GPIO_CTRL_2	32	**GPIO64 to GPIO95 direction control register** These direction control registers are used to select the data direction of the GPIO pin. The value driven onto the GPIO pins, are controlled by the GPIO_POL_x, and GPIO_DATA_x registers.
1E000610	GPIO_POL_0	32	**GPIO0 to GPIO31 polarity control register** These polarity control registers are used to control the polarity of the data is driven on or read from the GPIO pin.
1E000614	GPIO_POL_1	32	**GPIO32 to GPIO63 polarity control register** These polarity control registers are used to control the polarity of the data is driven on or read from the GPIO pin.
1E000618	GPIO_POL_2	32	**GPIO64 to GPIO95 polarity control register** These polarity control registers are used to control the polarity of the data is driven on or read from the GPIO pin.
1E000620	GPIO_DATA_0	32	**GPIO0 to GPIO31 data register** These data registers store current GPIO data value for GPIO input mode, or output driven value for GPIO output mode. Bit position stand for correspondent GPIO pin.
1E000624	GPIO_DATA_1	32	**GPIO32 to GPIO63 data register** These data registers store current GPIO data value for GPIO input mode, or output driven value for GPIO output mode. Bit position stand for correspondent GPIO pin.
1E000628	GPIO_DATA_2	32	**GPIO64 to GPIO95 data register** These data registers store current GPIO data value for GPIO input mode, or output driven value for GPIO output mode. Bit position stand for correspondent GPIO pin.
1E000630	GPIO_DSET_0	32	**GPIO0 to GPIO31 data set register** These data set registers are used to set bits in the GPIO_DATA_x registers.
1E000634	GPIO_DSET_1	32	**GPIO32 to GPIO63 data set register** These data set registers are used to set bits in the GPIO_DATA_x registers.
1E000638	GPIO_DSET_2	32	**GPIO64 to GPIO95 data set register** These data set registers are used to set bits in the GPIO_DATA_x registers.
1E000640	GPIO_DCLR_0	32	**GPIO0 to GPIO31 data clear register** These data set registers are used to clear bits in the GPIO_DATA_x registers.
1E000644	GPIO_DCLR_1	32	**GPIO32 to GPIO63 data clear register** These data set registers are used to clear bits in the GPIO_DATA_x registers.
1E000648	GPIO_DCLR_2	32	**GPIO64 to GPIO95 data clear register** These data set registers are used to clear bits in the GPIO_DATA_x registers.
1E000650	GINT_REDGE_0	32	**GPIO0 to GPIO31 rising edge interrupt enable register**

			These registers are used to enable the condition of rising edge triggered interrupt.
1E000654	GINT_REDGE_1	32	**GPIO32 to GPIO63 rising edge interrupt enable register** These registers are used to enable the condition of rising edge triggered interrupt.
1E000658	GINT_REDGE_2	32	**GPIO64 to GPIO95 rising edge interrupt enable register** These registers are used to enable the condition of rising edge triggered interrupt.
1E000660	GINT_FEDGE_0	32	**GPIO0 to GPIO31 falling edge interrupt enable register** These registers are used to enable the condition of falling edge triggered interrupt.
1E000664	GINT_FEDGE_1	32	**GPIO32 to GPIO63 falling edge interrupt enable register** These registers are used to enable the condition for falling edge triggered interrupt.
1E000668	GINT_FEDGE_2	32	**GPIO64 to GPIO95 falling edge interrupt enable register** These registers are used to enable the condition of falling edge triggered interrupt.
1E000670	GINT_HLVL_0	32	**GPIO0 to GPIO31 high level interrupt enable register** These registers are used to enable the condition of high level triggered interrupt. The bit in this register and the corresponded bit in GINT_LLVL_0 cannot be set to 1 at the same time.
1E000674	GINT_HLVL_1	32	**GPIO32 to GPIO63 high level interrupt enable register** These registers are used to enable the condition of high level triggered interrupt. The bit in this register and the corresponded bit in GINT_LLVL_1 cannot be set to 1 at the same time.
1E000678	GINT_HLVL_2	32	**GPIO64 to GPIO95 high level interrupt enable register** These registers are used to enable the condition of high level triggered interrupt. The bit in this register and the corresponded bit in GINT_LLVL_2 cannot be set to 1 at the same time.
1E000680	GINT_LLVL_0	32	**GPIO0 to GPIO31 low level interrupt enable register** These registers are used to enable the condition of low level triggered interrupt. The bit in this register and the corresponded bit in GINT_HLVL_0 cannot be set to 1 at the same time.
1E000684	GINT_LLVL_1	32	**GPIO32 to GPIO63 low level interrupt enable register** These registers are used to enable the condition of low level triggered interrupt. The bit in this register and the corresponded bit in GINT_HLVL_1 cannot be set to 1 at the same time.
1E000688	GINT_LLVL_2	32	**GPIO64 to GPIO95 low level interrupt enable register** These registers are used to enable the condition of low level triggered interrupt. The bit in this register and the corresponded bit in GINT_HLVL_2 cannot be set to 1 at the same time.
1E000690	GINT_STAT_0	32	**GPIO0 to GPIO31 interrupt status register** These registers are used to record the GPIO current interrupt status.
1E000694	GINT_STAT_1	32	**GPIO32 to GPIO63 interrupt status register** These registers are used to record the GPIO current interrupt status.
1E000698	GINT_STAT_2	32	**GPIO64 to GPIO95 interrupt status register** These registers are used to record the GPIO current interrupt status.
1E0006A0	GINT_EDGE_0	32	**GPIO0 to GPIO31 edge status register** These registers are used to record the GPIO current interrupt's edge status. These registers are useful only in edge triggered interrupt.
1E0006A4	GINT_EDGE_1	32	**GPIO32 to GPIO63 edge status register** These registers are used to record the GPIO current interrupt's edge status. These registers are useful only in edge triggered interrupt.
1E0006A8	GINT_EDGE_2	32	**GPIO64 to GPIO95 edge status register** These registers are used to record the GPIO current interrupt's edge status. These registers are useful only in edge triggered interrupt.

1E000600 GPIO_CTRL_0 GPIO0 to GPIO31 direction control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GPIOCTRL0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIOCTRL0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GPIOCTRL0	**GPIO Pin Direction** 0: GPIO input mode 1: GPIO output mode

1E000604 GPIO_CTRL_1 GPIO32 to GPIO63 direction control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GPIOCTRL1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIOCTRL1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GPIOCTRL1	**GPIO Pin Direction** 0: GPIO input mode 1: GPIO output mode

1E000608 GPIO_CTRL_2 GPIO64 to GPIO95 direction control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GPIOCTRL2[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIOCTRL2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GPIOCTRL2	**GPIO Pin Direction** 0: GPIO input mode 1: GPIO output mode

1E000610 GPIO_POL_0 GPIO0 to GPIO31 polarity control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GPIOPOL0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name	GPIOPOL0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GPIOPOL0	**GPIO Data Polarity** 0: Data is non-inverted 1: Data is inverted

1E000614 GPIO_POL_1 GPIO32 to GPIO63 polarity control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GPIOPOL1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIOPOL1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GPIOPOL1	**GPIO Data Polarity** 0: Data is non-inverted 1: Data is inverted

1E000618 GPIO_POL_2 GPIO64 to GPIO95 polarity control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GPIOPOL2[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIOPOL2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GPIOPOL2	**GPIO Data Polarity** 0: Data is non-inverted 1: Data is inverted

1E000620 GPIO_DATA_0 GPIO0 to GPIO31 data register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GPIODATA0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIODATA0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GPIODATA0	**GPIO Data**

1E000624 GPIO_DATA_1 GPIO32 to GPIO63 data register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GPIODATA1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIODATA1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GPIODATA1	**GPIO Data**

1E000628 GPIO_DATA_2 GPIO64 to GPIO95 data register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GPIODATA2[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIODATA2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GPIODATA2	**GPIO Data**

1E000630 GPIO_DSET_0 GPIO0 to GPIO31 data set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GPIODSET0[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIODSET0[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GPIODSET0	**GPIO Data Set** 1: Set the GPIO_DATA_0 register 0: No effect

1E000634 GPIO_DSET_1 GPIO32 to GPIO63 data set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GPIODSET1[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIODSET1[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GPIODSET1	**GPIO Data Set** 1: Set the GPIO_DATA_1 register 0: No effect

1E000638 GPIO_DSET_2 GPIO64 to GPIO95 data set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GPIODSET2[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIODSET2[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GPIODSET2	**GPIO Data Set** 1: Set the GPIO_DATA_2 register 0: No effect

1E000640 GPIO_DCLR_0 GPIO0 to GPIO31 data clear register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GPIODCLR0[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIODCLR0[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GPIODCLR0	**GPIO Data Clear** 1: Clear the GPIO_DATA_0 register 0: No effect

1E000644 GPIO_DCLR_1 GPIO32 to GPIO63 data clear register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GPIODCLR1[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIODCLR1[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GPIODCLR1	**GPIO Data Clear** 1: Clear the GPIO_DATA_1 register 0: No effect

1E000648 GPIO_DCLR_2 GPIO64 to GPIO95 data clear register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GPIODCLR2[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIODCLR2[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GPIODCLR2	**GPIO Data Clear** 1: Clear the GPIO_DATA_2 register 0: No effect

1E000650 GINT_REDGE_0 GPIO0 to GPIO31 rising edge interrupt enable register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GINTREDGE0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GINTREDGE0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GINTREDGE0	**GPIO Rising Edge Interrupt Enable** 1: Enable rising edge triggered 0: Disable rising edge triggered

1E000654 GINT_REDGE_1 GPIO32 to GPIO63 rising edge interrupt enable register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GINTREDGE1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GINTREDGE1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GINTREDGE1	**GPIO Rising Edge Interrupt Enable** 1: Enable rising edge triggered 0: Disable rising edge triggered

1E000658 GINT_REDGE_2 GPIO64 to GPIO95 rising edge interrupt enable register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GINTREDGE2[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name	GINTREDGE2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GINTREDGE2	**GPIO Rising Edge Interrupt Enable** 1: Enable rising edge triggered 0: Disable rising edge triggered

1E000660 GINT_FEDGE_0 GPIO0 to GPIO31 falling edge interrupt enable register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GINTFEDGE0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GINTFEDGE0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GINTFEDGE0	**GPIO Falling Edge Interrupt Enable** 1: Enable falling edge triggered 0: Disable falling edge triggered

1E000664 GINT_FEDGE_1 GPIO32 to GPIO63 falling edge interrupt enable register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GINTFEDGE1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GINTFEDGE1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GINTFEDGE1	**GPIO Falling Edge Interrupt Enable** 1: Enable falling edge triggered 0: Disable falling edge triggered

1E000668 GINT_FEDGE_2 GPIO64 to GPIO95 falling edge interrupt enable register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GINTFEDGE2[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GINTFEDGE2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GINTFEDGE2	**GPIO Falling Edge Interrupt Enable**

1: Enable falling edge triggered
0: Disable falling edge triggered

1E000670 GINT_HLVL_0 GPIO0 to GPIO31 high level interrupt enable register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GINTHLVL0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GINTHLVL0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GINTHLVL0	**GPIO High Level Interrupt Enable** 1: Enable high level triggered 0: Disable high level triggered

1E000674 GINT_HLVL_1 GPIO32 to GPIO63 high level interrupt enable register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GINTHLVL1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GINTHLVL1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GINTHLVL1	**GPIO High Level Interrupt Enable** 1: Enable high level triggered 0: Disable high level triggered

1E000678 GINT_HLVL_2 GPIO64 to GPIO95 high level interrupt enable register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GINTHLVL2[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GINTHLVL2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GINTHLVL2	**GPIO High Level Interrupt Enable** 1: Enable high level triggered 0: Disable high level triggered

1E000680 GINT_LLVL_0 GPIO0 to GPIO31 low level interrupt enable register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GINTLLVL0[31:16]															

Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GINTLLVL0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GINTLLVL0	**GPIO Low Level Interrupt Enable** 1: Enable low level triggered 0: Disable low level triggered

1E000684 GINT_LLVL_1 GPIO32 to GPIO63 low level interrupt enable register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GINTLLVL1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GINTLLVL1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GINTLLVL1	**GPIO Low Level Interrupt Enable** 1: Enable low level triggered 0: Disable low level triggered

1E000688 GINT_LLVL_2 GPIO64 to GPIO95 low level interrupt enable register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GINTLLVL2[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GINTLLVL2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GINTLLVL2	**GPIO Low Level Interrupt Enable** 1: Enable low level triggered 0: Disable low level triggered

1E000690 GINT_STAT_0 GPIO0 to GPIO31 interrupt status register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GINTSTAT0[31:16]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GINTSTAT0[15:0]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GINTSTAT0	**GPIO Interrupt Status** 1: Interrupt is detected 0: Interrupt is not detected

1E000694 GINT_STAT_1 GPIO32 to GPIO63 interrupt status register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GINTSTAT1[31:16]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GINTSTAT1[15:0]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GINTSTAT1	**GPIO Interrupt Status** 1: Interrupt is detected 0: Interrupt is not detected

1E000698 GINT_STAT_2 GPIO64 to GPIO95 interrupt status register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GINTSTAT2[31:16]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GINTSTAT2[15:0]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GINTSTAT2	**GPIO Interrupt Status** 1: Interrupt is detected 0: Interrupt is not detected

1E0006A0 GINT_EDGE_0 GPIO0 to GPIO31 edge status register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GINTEDGE0[31:16]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GINTEDGE0[15:0]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GINTEDGE0	**GPIO Interrupt Edge Status** 1: Rising edge 0: Falling edge

1E0006A4 GINT_EDGE_1 GPIO32 to GPIO63 edge status register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GINTEDGE1[31:16]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GINTEDGE1[15:0]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GINTEDGE1	**GPIO Interrupt Edge Status** 1: Rising edge 0: Falling edge

1E0006A8 GINT_EDGE_2 GPIO64 to GPIO95 edge status register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GINTEDGE2[31:16]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GINTEDGE2[15:0]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	GINTEDGE2	**GPIO Interrupt Edge Status** 1: Rising edge 0: Falling edge

2.7 I²C Controller

2.7.1 Features

- Programmable I²C bus clock rate
- Supports the Synchronous Inter-Integrated Circuits (I²C) serial protocol
- Bi-directional data transfer
- Programmable address width up to 8 bits
- Sequential byte read or write capability
- Device address and data address can be transmitted for device, page and address selection
- Supports Standard mode and Fast mode

2.7.2 List of Registers

Address	Name	Width	Register Function
1E000908	SM0CFG0	32	SERIAL INTERFACE MASTER 0 CONFIG 0 REGISTER
1E000910	SM0DOUT	32	SERIAL INTERFACE MASTER 0 DATAOUT REGISTER
1E000914	SM0DIN	32	SERIAL INTERFACE MASTER 0 DATAIN REGISTER
1E000918	SM0ST	32	SERIAL INTERFACE MASTER 0 STATUS REGISTER
1E00091C	SM0AUTO	32	SERIAL INTERFACE MASTER 0 AUTO-MODE REGISTER
1E000920	SM0CFG1	32	SERIAL INTERFACE MASTER 0 CONFIG 1 REGISTER
1E000928	SM0CFG2	32	SERIAL INTERFACE MASTER 0 CONFIG 2 REGISTER
1E000940	SM0CTL0	32	Serial interface master 0 control 0 register
1E000944	SM0CTL1	32	Serial interface master 0 control 1 register
1E000950	SM0D0	32	Serial interface master 0 data 0 register
1E000954	SM0D1	32	Serial interface master 0 data 1 register
1E00095C	PINTEN	32	Peripheral interrupt enable register
1E000960	PINTST	32	Peripheral interrupt status register
1E000964	PINTCL	32	Peripheral interrupt clear register

1E000908 SM0CFG0 SERIAL INTERFACE MASTER 0 CONFIG 0 REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0[24:9]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV0[8:0]									SM0_DEVADDR						
Type	RO									RW						
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:7	RSV0	Reserved
6:0	SM0_DEVADDR	Device address for transmission

1E000910 SM0DOUT SERIAL INTERFACE MASTER 0 DATAOUT REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0[23:8]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV0[7:0]								SM0_DATAOUT							
Type	RO								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

31:8	RSV0	**Reserved**
7:0	SM0_DATAOUT	**Data out register for auto mode**

1E000914 SM0DIN SERIAL INTERFACE MASTER 0 DATAIN REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0[23:8]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV0[7:0]								SM0_DATAIN							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:8	RSV0	**Reserved**
7:0	SM0_DATAIN	**Data in register for auto mode**

1E000918 SM0ST SERIAL INTERFACE MASTER 0 STATUS REGISTER 00000002

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0[28:13]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV0[12:0]													SM0_RDATA_RDY	SM0_WDATA_EMPTY	SM0_BUSY
Type	RO													RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bit(s)	Name	Description
31:3	RSV0	**Reserved**
2	SM0_RDATA_RDY	**I2C read data is ready**
1	SM0_WDATA_EMPTY	**I2C data output register is empty**
0	SM0_BUSY	**State machine is busy**

1E00091C SM0AUTO SERIAL INTERFACE MASTER 0 AUTO-MODE REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0[30:15]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV0[14:0]															SM0_S

																TART_RW
Type	RO															RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:1	RSV0	**Reserved**
0	SM0_START_RW	**Written with 1 to start a read transaction, and 0 to start a write transaction. This bit is only valid at auto mode.**

1E000920 SM0CFG1 SERIAL INTERFACE MASTER 0 CONFIG 1 REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0[25:10]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV0[9:0]										SM0_BYTECNT					
Type	RO										RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:6	RSV0	**Reserved**
5:0	SM0_BYTECNT	**The value + 1 indicateds the number of data bytes for sequential reads/writes. (word address is included in data bytes)**

1E000928 SM0CFG2 SERIAL INTERFACE MASTER 0 CONFIG 2 REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0[30:15]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV0[14:0]															SM0_IS_AUTOMODE
Type	RO															RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:1	RSV0	**Reserved**
0	SM0_IS_AUTOMODE	**Set 1 to configure auto mode**

1E000940 SM0CTL0 Serial interface master 0 control 0 register 0000034C

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SM0_ODRAIN	RSV0	SM0_VSYNC_MODE		SM0_CLK_DIV											
Type	RW	RO	RW		RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SM0_DEG_CNT								RSV1	SM0_WAIT_LEVEL	SM0_DEG_EN	SM0_CS_STATUS	SM0_SCL_STATE	SM0_SDA_STATE	SM0_EN	SM0_SCL_STRECH
Type	RW								RO	RW	RW	RO	RO	RO	RW	RW
Reset	0	0	0	0	0	0	1	1	0	1	0	0	1	1	0	0

Bit(s)	Name	Description
31	SM0_ODRAIN	**Open-drain output configuration** 0: When SIF output is logic 1, the output is pulled high by outer devices. SIF output is open-drained. 1: When SIF output is logic 1, the output is pulled high by SIF master 0.
30	RSV0	**Reserved**
29:28	SM0_VSYNC_MODE	**Restrict SIF master 0 trigger within VSYNC pulse** 00: Disable 01: Allow triggered in VSYNC pulse 10: Allow triggered at VSYNC rising edge
27:16	SM0_CLK_DIV	**SIF master 0 clock divide value** This is used to set the divider to generate expected SCL.
15:8	SM0_DEG_CNT	**SIF master 0 de-glitch value** This is used to set the de-glitch number of SDA and SCL input.
7	RSV1	**Reserved**
6	SM0_WAIT_LEVEL	**SIF master 0 wait level configuration** 0: output L when SIF master 0 is in WAIT state 1: output H when SIF master 0 is in WAIT state
5	SM0_DEG_EN	**SIF master 0 de-glitch enable bit** 0: Disable SIF master de-glitch. 1: Enable SIF master de-glitch.
4	SM0_CS_STATUS	**Clock stretching status** 0: no clock stretching 1: clock stretching
3	SM0_SCL_STATE	**SCL value on the bus**
2	SM0_SDA_STATE	**SDA value on the bus**
1	SM0_EN	**SIF master 0 enable bit** 0: Disable SIF master 0. 1: Enable SIF master 0.
0	SM0_SCL_STRECH	**Clock stretching enable** 0: Not allow slaves hold SCL 1: Allow slaves hold SCL

1E000944 SM0CTL1 Serial interface master 0 control 1 register 00000080

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV3								SM0_ACK							
Type	RO								RO							

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV2					SM0_PGLEN			RSV1	SM0_MODE			RSV0			SM0_TRI
Type	RO					RW			RO	RW			RO			RW
Reset	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	RSV3	**Reserved**
23:16	SM0_ACK	**Acknowledge bits** ACK[7:0] is acknowledge of 8 bytes of data
15:11	RSV2	**Reserved**
10:8	SM0_PGLEN	**Page length** Page length of sequential read/write. The maximum is 8 bytes. Set 0 as 1 byte.
7	RSV1	**Reserved**
6:4	SM0_MODE	**SIF master mode** 001: Start 010: Write data 011: Stop 100: Read data with no ack for final byte 101: Read data with ack
3:1	RSV0	**Reserved**
0	SM0_TRI	**Trigger serial interface** 0: Read back as serial interface is idle. 1: Set 1 to trigger this serial interface. Read back as serial interface is busy.

1E000950 SM0D0 Serial interface master 0 data 0 register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SM0_DATA3								SM0_DATA2							
Type	RW								RW							
Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SM0_DATA1								SM0_DATA0							
Type	RW								RW							
Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Bit(s)	Name	Description
31:24	SM0_DATA3	**Serial interface data byte 3**
23:16	SM0_DATA2	**Serial interface data byte 2**
15:8	SM0_DATA1	**Serial interface data byte 1**
7:0	SM0_DATA0	**Serial interface data byte 0**

1E000954 SM0D1 Serial interface master 0 data 1 register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SM0_DATA7								SM0_DATA6							
Type	RW								RW							
Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name	SM0_DATA5								SM0_DATA4							
Type	RW								RW							
Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Bit(s)	Name	Description
31:24	SM0_DATA7	**Serial interface data byte 7**
23:16	SM0_DATA6	**Serial interface data byte 6**
15:8	SM0_DATA5	**Serial interface data byte 5**
7:0	SM0_DATA4	**Serial interface data byte 4**

1E00095C PINTEN Peripheral interrupt enable register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0[30:15]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV0[14:0]															SM0_INT_EN
Type	RO															RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:1	RSV0	**Reserved**
0	SM0_INT_EN	**Serial interface master 0 interrupt enable**

1E000960 PINTST Peripheral interrupt status register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0[30:15]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV0[14:0]															SM0_INT_ST
Type	RO															RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:1	RSV0	**Reserved**
0	SM0_INT_ST	**Serial interface master 0 interrupt status**

1E000964 PINTCL Peripheral interrupt clear register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Name	RSV0[30:15]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV0[14:0]															SM0_INT_CL
Type	RO															RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:1	RSV0	**Reserved**
0	SM0_INT_CL	**Serial interface master 0 interrupt clear**

2.8 NAND Flash Interface

2.8.1 Features

- ECC (BCH code) acceleration capable of 4/6/8 error correction. (with ECC engine)
- Programmable page size and spare size
- Programmable FDM data size and protected FDM data size.
- Word/byte access through APB bus.
- DMA for massive data transfer.
- Latch sensitive interrupt to indicate ready state for read, program, erase operation.
- Programmable wait states, command/address setup and hold time, read enable hold time, and write enable recovery time.

2.8.2 Registers

Address	Name	Width	Register Function
1E003000	NFI_CNFG	16	**NFI Configuration** The register controls the NFI functions. For all enable fields, Setting to be logic-1 represents enabled, while 0 represents disabled.
1E003004	NFI_PAGEFMT	16	**NFI Page Format Control Register** This register manages the page format of the device. It includes the bus width selection, the page size, the associated address format, and the spare format.
1E003008	NFI_CON	16	**NFI Operation Control Register** This is recommended to reset the state machine, data FIFO and flush the data FIFO before starting a new command
1E00300C	NFI_ACCCON	32	**NAND Flash Access Timing Control register** This is the timing access control register for the NAND FLASH interface. In order to accommodate operations for different system clock frequency ranges from 13MHz to 61.44MHz, wait states and setup/hold time margin can be configured in this register.
1E003010	NFI_INTR_EN	16	**NFI Interrupt Enable Register** This register controls the activity for the interrupt sources. These enable should be turned on only while SW expects the corresponding interrupt will occur.
1E003014	NFI_INTR	16	**NFI Interrupt Status Register** The register indicates the status of all the interrupt sources. Read this register will clear all interrupts.
1E003020	NFI_CMD	16	**NFI Command register** This is the command input register. The user should write this register to issue a command. Please refer to device datasheet for the command set. Before write the command, please check out the settings for register NFI_CON.
1E003030	NFI_ADDRNOB	16	**NFI Address Length Register** This register represents the number of bytes corresponding to current command. The each valid number of bytes ranges from 0 to 4. The address format depends on what device to be used and what commands to be applied. The NFI core is made transparent to those different situations except that the user has to define the number of bytes. The user should write the target address to the address register NFI_COLADDR and NFI_ROWADDR before programming this register.
1E003034	NFI_COLADDR	32	**NFI Column Address Register** This defines the 4 bytes of the column address field to be applied to the device. Since the device bus width is 1 byte, the NFI core arranges the order of address data to be least significant byte first. The user should put the first address byte in the field ADDR0, the second byte in the field ADDR1, and so on.
1E003038	NFI_ROWADDR	32	**NFI Row Address Register** This defines the 4 bytes of the row address field to be applied to the device. Since the device bus width is 1 byte, the NFI core arranges the order of address data to be least significant byte first. The user should put the first address byte in the field ADDR0, the second byte in the field ADDR1, and so on.
1E003040	NFI_STRDATA	16	**NFI Data Transfer Start Trigger Register** This register controls the activity for the interrupt sources.
1E003044	NFI_CNRNB	16	**NFI Check NAND Ready/Busy Register** This register controls the activity for the interrupt sources.
1E003050	NFI_DATAW	32	**NFI Write Data Buffer** This is the write port of the data FIFO. It supports word access. The least significant byte DW0 is to be programmed to the device first, then DW1, and so on.
1E003054	NFI_DATAR	32	**NFI Read Data Buffer**

			This is the read port of the data FIFO. It supports word access. The least significant byte DR0 is the first byte read from the device, then DR1, and so on.
1E003058	NFI_PIO_DIRDY	16	**PIO_mode Data Ready Register** This register indicates the data is ready for input
1E003060	NFI_STA	32	**NFI Status** This register represents the NFI core control status including command mode, address mode, data program and read mode. The user should poll this register for the end of those operations.
1E003064	NFI_FIFOSTA	16	**NFI FIFO Status** The register represents the status of the data FIFO. The FIFO top and bottom pointer of read & write will be reset when issue "command" to NAND Flash
1E003068	NFI_LOCKSTA	16	**NFI Lock Status** This register represents the lock status for each lock range. If any access_lockxx happens, the nfi core will automatic issue a reset (0xFF) command to NAND device.
1E003070	NFI_ADDRCNTR	16	**NFI Page Address Counter Register** The register represents the current read/write address with respect to initial address input. It counts in unit of byte. In page read and page program operation, the address should be the same as that in the state machine in the target device.
1E003080	**NFI_STRADDR**	32	**NFI AHB Start Address Register** The register represents the start address for DMA to access EMI. These memory from the start address is used to put read data from NAND or write data to NAND in DMA mode
1E003084	NFI_BYTELEN	16	**NFI DMA Byte Length Register** The register represents the current transfer length for DMA to access EMI.
1E003090	**NFI_CSEL**	16	**NFI device select register** The register is used to select the target device. It decides which CEB pin to be functional. This is useful while using the high-density device.
1E003094	**NFI_IOCON**	16	**NFI IO Control register** Data bus pull down when no use.
1E0030A0	**NFI_FDM0L**	32	**NFI Least FDM Data for Sector 0 Register** This register represents the Least FDM data for the sector 0. Since the device bus width is 1 byte, the NFI core arranges the order of address data to be least significant byte first. The user should put the first address byte in the field FDM0_0, the second byte in the field FDM0_1, and so on. It will be reset to 0xFF when issue NFI_Reset.
1E0030A4	**NFI_FDM0M**	32	**NFI Most FDM Data for Sector 0 Register** This register represents the Most FDM data for the sector 0. Since the device bus width is 1 byte, the NFI core arranges the order of address data to be least significant byte first. The user should put the first address byte in the field
1E003100	NFI_LOCK	16	**NFI Lock Enable Register** This register enable the lock function of NFI . These setting can only be set once after reset chip.
1E003104	NFI_LOCKCON	32	**NFI Lock Control Register** This register control the lock function of NFI . These setting can only be set once after reset chip.
1E003108	NFI_LOCKANOB	16	**NFI Address Format for Lock Register** This register represents the number of bytes corresponding to erase and program command. The each valid number of bytes ranges from 0 to 4. The address format depends on what device to be used and what commands to be applied. The NFI core will force these setting during some command operation(8X or 6X). These setting can only be set once after reset chip.
1E003110	NFI_LOCK00ADD	32	**NFI Row Start Address for Lock Set00 Register** This defines the 4 bytes of the row start address field to be locked range for the device. These setting can only be set once after reset chip.
1E003114	NFI_LOCK00FMT	32	**NFI Row Address Format for Lock Set00 Register**

			This defines the 4 bytes format of the row address field to be locked range for the device. These setting can only be set once after reset chip. The MSB unused range must be set to 0 for LOCKxxFMT.
1E003190	**NFI_FIFODATA0**	32	**NFI FIFO Content Data 0** This register represents the content data 0 of fifo.
1E003194	**NFI_FIFODATA1**	32	**NFI FIFO Content Data 1** This register represents the content data 1 of fifo.
1E003198	**NFI_FIFODATA2**	32	**NFI FIFO Content Data 2** This register represents the content data 2 of fifo.
1E00319C	**NFI_FIFODATA3**	32	**NFI FIFO Content Data 3** This register represents the content data 3 of fifo.
1E003200	**NFI_MCON**	16	**NFI LCD Monitor Control Register**
1E003204	**NFI_TOTALCNT**	32	**NFI LCD Monitor Total Cycle Count**
1E003208	**NFI_RQCNT**	32	**NFI LCD Monitor Request Cycle Count**
1E00320C	**NFI_ACCNT**	32	**NFI LCD Monitor Access Cycle Count**
1E003210	**NFI_MASTERSTA**	16	**NFI Master Status** The four indicator represents MASTER status in the BUS access. There are three channels for AHB master. The MSB(Bit 2) to LSB(bit0) repesent ECC, Auto-Correction and NFI channel respectively. Each bit represents the channel is active or inactive. 0 is inactive, 1 is active. After NFI reset, the NFI_MASTERSTA should be checked to guarantee the master is stopped. For example: MAS_XX[0] The NFI channel is in the XX status. MAS_XX[1] The Auto-Correction channel is in the XX status. MAS_XX[2] The ECC channel is in the XX status.

1E003000 NFI_CNFG NFI Configuration 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne		OP_MODE					AUTO_FMT_EN	HW_ECC_EN		BYTE_RW				DMA_BURST_EN	READ_MODE	DMA_MODE
Type		R/W					R/W	R/W		R/W				R/W	R/W	R/W
Reset		0	0	0			0	0		0				0	0	0

Bit(s)	Name	Description
14:12	OP_MODE	**The field control the operating process flow of FSM for NFI.** 000b: Idle state. 001b: Read Process. Recommend for basic read operation. 010b: Single Read Process. Recommend for read id and read status. 011b: Program Process. Recommend for basic program operation. 100b: Erase Process. Recommend for basic erase operation. 101b: Reset Process. Recommend for basic reset operation. 110b: Custom Process. Recommend for all advance operation. Others: Reserved
9	AUTO_FMT_EN	**Automatic HW ECC encode or decode enable.** If enabled, the ECC parity from HW ECC engine and FDM data from Register are written automatically to the spare area. If disable, the spare data all comes from PIO register, like DATAR, DATAW, (PIO Mode) or the memory(DMA Mode) as main area data.
8	HW_ECC_EN	**This field is used to enable encoding or decoding operation of HW ECC engine. If the bit is enabled, the data is transferring to ECC engine for encoding and decoding. The ECC Engine should be configured as nfi encoding mode, otherwise the NFI will hang.**
6	BYTE_RW	**Enable byte access. The valid bytes read from NFI_DATAR and NFI_DATAW is only DR0 and**

		DW0 if BYTE_RW is enabled.
2	DMA_BURST_EN	
1	READ_MODE	**This field is used to control the activity of read or write transfer.** 0: write operation of DMA or PIO. 1: read operation of DMA or PIO.
0	DMA_MODE	**This field is used to control the Operation mode.** 0: PIO mode. All data (include read or write) move by MCU through APB access. 1: DMA mode. All data (include read or write) move by HW automation through AHB bus.

1E003004 NFI_PAGEFMT NFI Page Format Control Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	FDM_ECC_NUM				FDM_NUM						SPARE_SIZE		DBYTE_EN		PAGE_SIZE	
Type	R/W				R/W						R/W		R/W		R/W	
Reset	0	0	0	0	0	0	0	0			0	0	0		0	0

Bit(s)	Name	Description
15:12	FDM_ECC_NUM	**The number of each FDM data for HW ECC protection. The valid number of bytes ranges are from 0 to 8.**
11:8	FDM_NUM	**The FDM data number for each spare area. The valid number of bytes are from 0 to 8.**
5:4	SPARE_SIZE	
3	DBYTE_EN	**16 bits I/O bus interface enable.**
1:0	PAGE_SIZE	**Page Size. The field specifies the size of one page for the device. Some most widely used page size are supported.** 0: The page size is 512 bytes (including 512 bytes data area and (spare_size*1) bytes spare area). 1: The page size is 2k bytes (including 2048 bytes data area and (spare_size*4) bytes spare area). 2: The page size is 4k bytes (including 4096 bytes data area and (spare_size*8) bytes spare area). 3: Reserved.

1E003008 NFI_CON NFI Operation Control Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	SEC_NUM						BWR	BRD	NOB			SRD			NFI_RST	FIFO_FLUSH
Type	R/W						R/W	R/W	W/R			WO			WO	WO
Reset	0	0	0	0			0	0	0	0	0	0			0	0

Bit(s)	Name	Description
15:12	SEC_NUM	**The field represents the sector number to be retrieved from the device or DMA Master. The valid number ranges from 1 to 8.**
9	BWR	**Burst write mode. Setting to be logic-1 enables the data burst write operation.**
8	BRD	**Burst read mode. Setting this field to be logic-1 enables the data read operation. The NFI core will issue read cycles to retrieve data from the device when the data FIFO is not full or the device is not in the busy state. The NFI core supports consecutive page reading.**
7:5	NOB	**The field represents the number of bytes to be retrieved from the device in single mode, and the number of bytes per APB transaction in both single and burst mode. If device is 16-bit IO, the read bytes number will double** 0: Read 8 bytes from the device. (16 byte for 16-bit IO) 1: Read 1 byte from the device. (2 byte for 16-bit IO)

		2: Read 2 bytes from the device. (4 byte for 16-bit IO) 3: Read 3 bytes from the device. (6 byte for 16-bit IO) 4: Read 4 bytes from the device. (8 byte for 16-bit IO) 5: Read 5 byte from the device. (10 byte for 16-bit IO) 6: Read 6 bytes from the device. (12 byte for 16-bit IO) 7: Read 7 bytes from the device. (14 byte for 16-bit IO)
4	SRD	**Setting to be logic-1 initializes the one-shot data read operation. It's mainly used for read ID and read status command, which requires no more than 4 read cycles to retrieve data from the device. It used when FIFO is empty or after reset nficore**
1	NFI_RST	**Reset the state machine, data FIFO (0x0000) and FDM data (0xffff)**
0	FIFO_FLUSH	**Flush the data FIFO.**

1E00300C NFI_ACCCON NAND Flash Access Timing Control register NA

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	POECS				PRECS						C2R					
Type	R/W				R/W						R/W					
Reset	F	F	F	F	0F	0F	0F	0F	0F	0F	3F	3F	3F	3F	3F	3F
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	W2R				WH				WST				RLT			
Type	R/W				R/W				R/W				R/W			
Reset	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F

Bit(s)	Name	Description
31:28	POECS	**The field represents the minimum required time for CS post-pulling down after the access to device.** Minimum required time = PRECS[1:0] + PRECS[2]*8 + PRECS[3]*64 (T)
27:22	PRECS	**The field represents the minimum required time for CS pre-pulling down before any access to device.** Minimum required time = PRECS[1:0] + PRECS[3:2]*8 + PRECS[5:4]*128 (T)
21:16	C2R	**The field represents the minimum required time from NCEB low to NREB low. It's in unit of 2T.** Minimum required time = C2R[5:0]*2 + 1 (T)
15:12	W2R	**The field represents the minimum required time from NWEB high to NREB low. It's in unit of 2T. So the actual time ranges from 0T to 30T in step of 2T.** Minimum required time = W2R[3:0]*2 + 1 (T)
11:8	WH	**Write-enable hold-time.** The field specifies the hold time of NALE, NCLE, NCEB signals relative to the rising edge of NWEB. This field is associated with WST to expand the write cycle time, and is associated with RLT to expand the read cycle time.
7:4	WST	**Write Wait State** The field specifies the wait states to be inserted to meet the requirement of the pulse width of the NWEB signal. 00b: No wait state. 01b: 1T wait state. 10b: 2T wait state. 11b: 3T wait state.
3:0	RLT	**Read Latency Time** The field specifies how many wait states to be inserted to meet the requirement of the read access time for the device. 00b: No wait state. 01b: 1T wait state. 10b: 2T wait state. 11b: 3T wait state.

1E003010 NFI_INTR_EN NFI Interrupt Enable Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne										AHB_DONE_EN	ACCESS_LOCK_EN	BUSY_RETURN_EN	ERASE_DONE_EN	RESET_DONE_EN	WR_DONE_EN	RD_DONE_EN
Type										R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset										0	0	0	0	0	0	0

Bit(s)	Name	Description
6	AHB_DONE_EN	**The done interrupt enable for DMA mode.**
5	ACCESS_LOCK_EN	
4	BUSY_RETURN_EN	**The busy return interrupt enable.**
3	ERASE_DONE_EN	**The erase completion interrupt enable.**
2	RESET_DONE_EN	**The reset completion interrupt enable.**
1	WR_DONE_EN	**The single page write completion interrupt enable.**
0	RD_DONE_EN	**The single page read completion interrupt enable.**

1E003014 NFI_INTR NFI Interrupt Status Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne										AHB_DONE	ACCESS_LOCK	BUSY_RETURN	ERASE_DONE	RESET_DONE	WR_DONE	RD_DONE
Type										RC	RC	RC	RC	RC	RC	RC
Reset										0	0	0	0	0	0	0

Bit(s)	Name	Description
6	AHB_DONE	**Indicates that the AHB operation is completed.**
5	ACCESS_LOCK	
4	BUSY_RETURN	**Indicates that the device state returns from busy by inspecting the R/B# pin.**
3	ERASE_DONE	**Indicates that the erase operation is completed.**
2	RESET_DONE	**Indicates that the reset operation is completed.**
1	WR_DONE	**Indicates that the write operation is completed.**
0	RD_DONE	**Indicates that the single page read operation is completed.**

1E003020 NFI_CMD NFI Command register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne									CMD							
Type									R/W							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7:0	CMD	**Command word.**

1E003030 NFI_ADDRNOB NFI Address Length Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne										ROW_ADDR_NOB				COL_ADDR_NOB		
Type										R/W				R/W		
Reset										0	0	0		0	0	0

Bit(s)	Name	Description
6:4	ROW_ADDR_NOB	**Number of bytes for the row address**
2:0	COL_ADDR_NOB	**Number of bytes for the column address**

1E003034 NFI_COLADDR NFI Column Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	COL_ADDR3								COL_ADDR2							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	COL_ADDR1								COL_ADDR0							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	COL_ADDR3	**The 3-th column address byte.**
23:16	COL_ADDR2	**The 2-th column address byte.**
15:8	COL_ADDR1	**The 1-th column address byte.**
7:0	COL_ADDR0	**The 0-th column address byte.**

1E003038 NFI_ROWADDR NFI Row Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	ROW_ADDR3								ROW_ADDR2							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	ROW_ADDR1								ROW_ADDR0							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	ROW_ADDR3	**The 3-th row address byte.**
23:16	ROW_ADDR2	**The 2-th row address byte.**
15:8	ROW_ADDR1	**The 1-th row address byte.**
7:0	ROW_ADDR0	**The 0-th row address byte.**

1E003040 NFI_STRDATA NFI Data Transfer Start Trigger Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne																STR_DATA
Type																WO
Reset																0

Bit(s)	Name	Description
0	STR_DATA	**This signal triggers the data transfer for read or write. It only takes effect as custom operation mode**

1E003044 NFI_CNRNB NFI Check NAND Ready/Busy Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne									CB2R_TIME							STR_CNRNB
Type									R/W							WO
Reset									0	0	0	0				0

Bit(s)	Name	Description
7:4	CB2R_TIME	**This time-out registers for polling the NAND busy/ready signal. The unit is 16T clock cycles. The clock rate is 61.44MHz in normal mode. It will be slow down after enable HW DCM mode.**
0	STR_CNRNB	**This signal triggers NFI to poll the status the NAND busy/ready signal after CB2R_TIME*16 cycles. This function is used to avoid the fail function of "BUSY2READY" status or "BUSY_RETURN" interrupt when NAND is operating at very low frequency(<7MHz). If NAND is operating in lower frequency, the sampling for the event, NAND busy/ready signal from low to high, may be failed and NFI will be hanged in busy state. This signal is a time-out register to check the NAND status. The results will be report to "BUSY2READY" status and "BUSY_RETURN" interrupt.**

1E003050 NFI_DATAW NFI Write Data Buffer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	DW3								DW2							
Type	WO								WO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	DW1								DW0							
Type	WO								WO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DW3	**Write data byte 3.**
23:16	DW2	**Write data byte 2.**
15:8	DW1	**Write data byte 1.**
7:0	DW0	**Write data byte 0.**

1E003054 NFI_DATAR NFI Read Data Buffer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	DR3								DR2							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	DR1								DR0							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DR3	**Read data byte 3.**
23:16	DR2	**Read data byte 2.**
15:8	DR1	**Read data byte 1.**
7:0	DR0	**Read data byte 0.**

1E003058 NFI_PIO_DIRDY PIO_mode Data Ready Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne																PIO_DI_RDY
Type																RO
Reset																0

Bit(s)	Name	Description
0	PIO_DI_RDY	**indicates the PIO mode is ready for read data in read mode and ready for write data in write mode.** 0: NFI_DATAR and NFI_DATAW should not be read or write (not ready). 1: NFI is ready for reading data in ready mode and writing data in write mode.

1E003060 NFI_STA NFI Status 00001000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne				NAND_FSM									NFI_FSM			
Type				RO									RO			
Reset				0	0	0	0	0					0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne				READ_EMPTY			BUSY2READY	BUSY				ACCESS_LOCK	DATAW	DATAR	ADDR	CMD
Type				RO			RO	RO				RO	RO	RO	RO	RO
Reset				1			0	0				0	0	0	0	0

Bit(s)	Name	Description
28:24	NAND_FSM	**The field represents the state of NAND interface FSM.** 000000b: IDLE. idle. 111000b: PRE_CS. Pre CS state. 001001b: CMD_WRST. command write set up 001010b: CMD_WR. Command write enable. 001011b: CMD_WRHD. Command write hold. 001000b: CMD_WRRDY 010001b: ADDR_WRST. Address write set up 010010b: ADDR_WR. Address write enable 010011b: ADDR_WRHD. Address write hold 010000b: ADDR_WRRDY. 011000b: CA2DEXT. Command address write extension. 100001b: DATA_RDST. Data read set up. 100010b: DATA_RD. Data read enable. 100011b: DATA_RDHD. Data read hold. 110001b: DATA_WRST. Data write set up. 110010b: DATA_WR. Data write enable. 110011b: DATA_WRHD. Data write hold. Others: Reserved
19:16	NFI_FSM	**The field represents the state of NFI internal FSM.** 0000b: idle.

		0001b: reset. Reset command to ready 0010b: read busy. 0011b: read data. 0100b: program busy 0101b: program data. Input data command to program command 1000b: erase busy. Erase command to ready 1001b: erase data. Erase command 1 to erase command 2 1111b: custom mode 1110b: custom mode for data access Others: Reserved
12	READ_EMPTY	**Empty page indication during read operation, include all data, FDM and parity for all sectors**
9	BUSY2READY	**It's read-only. This signal indicates NAND from busy to ready state and it will be reset after nfi_reset or write command/address.**
8	BUSY	**Synchronized busy signal from the NAND flash. It's read-only. This signal is sampled from NFI**
4	ACCESS_LOCK	**The access range is locked for erase or program .**
3	DATAW	**The NFI core is in data write mode.**
2	DATAR	**The NFI core is in data read mode.**
1	ADDR	**The NFI core is in address mode.**
0	CMD	**The NFI core is in command mode.**

1E003064 NFI_FIFOSTA NFI FIFO Status 4040

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	WR_FULL	WR_EMPTY		WR_REMAIN					RD_FULL	RD_EMPTY		RD_REMAIN				
Type	RO	RO		RO					RO	RO		RO				
Reset	0	1		0	0	0	0	0	0	1		0	0	0	0	0

Bit(s)	Name	Description
15	WR_FULL	**Data FIFO full in burst write mode.**
14	WR_EMPTY	**Data FIFO empty in burst write mode.**
12:8	WR_REMAIN	**Data FIFO remaining byte number in burst write mode.**
7	RD_FULL	**Data FIFO full in burst read mode.**
6	RD_EMPTY	**Data FIFO empty in burst read mode.**
4:0	RD_REMAIN	**Data FIFO remaining byte number in burst read mode.**

1E003068 NFI_LOCKSTA NFI Lock Status 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	ACCESS_LOCK15	ACCESS_LOCK14	ACCESS_LOCK13	ACCESS_LOCK12	ACCESS_LOCK11	ACCESS_LOCK10	ACCESS_LOCK09	ACCESS_LOCK08	ACCESS_LOCK07	ACCESS_LOCK06	ACCESS_LOCK05	ACCESS_LOCK04	ACCESS_LOCK03	ACCESS_LOCK02	ACCESS_LOCK01	ACCESS_LOCK00
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15	ACCESS_LOCK15	**The access command violates the locking range 15**
14	ACCESS_LOCK14	**The access command violates the locking range 14**
13	ACCESS_LOCK13	**The access command violates the locking range 13**
12	ACCESS_LOCK12	**The access command violates the locking range 12**
11	ACCESS_LOCK11	**The access command violates the locking range 11**

10	ACCESS_LOCK10	**The access command violates the locking range 10**
9	ACCESS_LOCK09	**The access command violates the locking range 9**
8	ACCESS_LOCK08	**The access command violates the locking range 8**
7	ACCESS_LOCK07	**The access command violates the locking range 7**
6	ACCESS_LOCK06	**The access command violates the locking range 6**
5	ACCESS_LOCK05	**The access command violates the locking range 5**
4	ACCESS_LOCK04	**The access command violates the locking range 4**
3	ACCESS_LOCK03	**The access command violates the locking range 3**
2	ACCESS_LOCK02	**The access command violates the locking range 2**
1	ACCESS_LOCK01	**The access command violates the locking range 1**
0	ACCESS_LOCK00	**The access command violates the locking range 0**

1E003070 NFI_ADDRCNTR NFI Page Address Counter Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	**SEC_CNTR**						**SEC_ADDR**									
Type	RO						RO									
Reset	0	0	0	0			0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:12	SEC_CNTR	**The sector count.**
9:0	SEC_ADDR	**The address count of 512 main data and spare data for each sector.**

1E003080 NFI_STRADDR NFI AHB Start Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	**STR_ADDR[31:16]**															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	**STR_ADDR[15:0]**															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	STR_ADDR	**The start address of EMI for both read or write in DMA mode.** If start address of any sector data is not 4-byte aligned, the transfer will be automatically split into byte and word transaction by NFI DMA. Non 4-byte aligned data will be transferred in single-byte transaction. Non 16-byte aligned data will be transferred in single-word transaction. 16-byte aligned data will be transferred by 4 word incrementing bust if the NFI_CNFG->DMA_BURST_EN is enabled.

1E003084 NFI_BYTELEN NFI DMA Byte Length Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	**BUS_SEC_CNTR**						**BUS_SEC_ADDR**									
Type	RO						RO									
Reset	0	0	0	0			0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

15:12	BUS_SEC_CNTR	**The sector count.**
9:0	BUS_SEC_ADDR	**The address count of 512 main data and spare data for each sector.**

1E003090 NFI_CSEL NFI device select register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne																CSEL
Type																R/W
Reset																0

Bit(s)	Name	Description
0	CSEL	**Chip select. The value defaults to 0.** 0: Device 1 is selected. 1: Device 2 is selected.

1E003094 NFI_IOCON NFI IO Control register 0006

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne									BRSTN					L2NW	L2NR	NLD_PD
Type									R/W					R/W	R/W	R/W
Reset									0	0	0	0		1	1	0

Bit(s)	Name	Description
7:4	BRSTN	**Maximum Burst Number for NAND read and writes. The unit is number of byte (8bits I/O) or double byte (16bits I/O)**
2	L2NW	**Enable 1T latency for the arbitration from LCD to NAND write operation, this is used to prevent bus contention between chip, NAND flash and LCD device.**
1	L2NR	**Enable 1T latency for the arbitration from LCD to NAND read operation, this is used to prevent bus contention between chip, NAND flash and LCD device.**
0	NLD_PD	**data bus pull down when no use.** 0: disable. 1: enable.

1E0030A0 NFI_FDM0L NFI Least FDM Data for Sector 0 Register NA

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	FDM0_3								FDM0_2							
Type	R/W								R/W							
Reset	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	FDM0_1								FDM0_0							
Type	R/W								R/W							
Reset	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff

Bit(s)	Name	Description
31:24	FDM0_3	**The 3-th FDM byte data for sector 0.**
23:16	FDM0_2	**The 2-th FDM byte data for sector 0.**
15:8	FDM0_1	**The 1-th FDM byte data for sector 0.**
7:0	FDM0_0	**The 0-th FDM byte data for sector 0.**

1E0030A4 NFI_FDM0M NFI Most FDM Data for Sector 0 Register NA

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	FDM0_7								FDM0_6							
Type	R/W								R/W							
Reset	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	FDM0_5								FDM0_4							
Type	R/W								R/W							
Reset	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff

Bit(s)	Name	Description
31:24	FDM0_7	**The 3-th FDM byte data for sector 0.**
23:16	FDM0_6	**The 2-th FDM byte data for sector 0.**
15:8	FDM0_5	**The 1-th FDM byte data for sector 0.**
7:0	FDM0_4	**The 0-th FDM byte data for sector 0.**

1E003100 NFI_LOCK NFI Lock Enable Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne																LOCK_ON
Type																R/W1
Reset																0

Bit(s)	Name	Description
0	LOCK_ON	**Enable the lock checking process for any lock set.** 0: Disable lock checking process. 1: Enable lock checking process.

1E003104 NFI_LOCKCON NFI Lock Control Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	LOCK15_CS	LOCK15_EN	LOCK14_CS	LOCK14_EN	LOCK13_CS	LOCK13_EN	LOCK12_CS	LOCK12_EN	LOCK11_CS	LOCK11_EN	LOCK10_CS	LOCK10_EN	LOCK09_CS	LOCK09_EN	LOCK08_CS	LOCK08_EN
Type	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	LOCK07_CS	LOCK07_EN	LOCK06_CS	LOCK06_EN	LOCK05_CS	LOCK05_EN	LOCK04_CS	LOCK04_EN	LOCK03_CS	LOCK03_EN	LOCK02_CS	LOCK02_EN	LOCK01_CS	LOCK01_EN	LOCK00_CS	LOCK00_EN
Type	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	LOCK15_CS	**Indicate the lock checking process of lock set.n for CS0 or CS1** 0: Lock range check of set 15 for CS0. 1: Lock range check of set 15 for CS1.
30	LOCK15_EN	**Enable the lock checking process of lock set 15. Before it takes effect, the LOCK_ON must be**

		turned on. 0: Disable Lock Range check for set 15. 1: Enable Lock Range check for set 15.
29	LOCK14_CS	**Indicate the lock checking process of lock set.n for CS0 or CS1** 0: Lock range check of set 14 for CS0. 1: Lock range check of set 14 for CS1.
28	LOCK14_EN	**Enable the lock checking process of lock set 14. Before it takes effect, the LOCK_ON must be turned on.** 0: Disable Lock Range check for set 14. 1: Enable Lock Range check for set 14.
27	LOCK13_CS	**Indicate the lock checking process of lock set.n for CS0 or CS1** 0: Lock range check of set 13 for CS0. 1: Lock range check of set 13 for CS1.
26	LOCK13_EN	**Enable the lock checking process of lock set 13. Before it takes effect, the LOCK_ON must be turned on.** 0: Disable Lock Range check for set 13. 1: Enable Lock Range check for set 13.
25	LOCK12_CS	**Indicate the lock checking process of lock set.n for CS0 or CS1** 0: Lock range check of set 12 for CS0. 1: Lock range check of set 12 for CS1.
24	LOCK12_EN	**Enable the lock checking process of lock set 12. Before it takes effect, the LOCK_ON must be turned on.** 0: Disable Lock Range check for set 12. 1: Enable Lock Range check for set 12.
23	LOCK11_CS	**Indicate the lock checking process of lock set.n for CS0 or CS1** 0: Lock range check of set 11 for CS0. 1: Lock range check of set 11 for CS1.
22	LOCK11_EN	**Enable the lock checking process of lock set 11. Before it takes effect, the LOCK_ON must be turned on.** 0: Disable Lock Range check for set 11. 1: Enable Lock Range check for set 11.
21	LOCK10_CS	**Indicate the lock checking process of lock set.n for CS0 or CS1** 0: Lock range check of set 10 for CS0. 1: Lock range check of set 10 for CS1.
20	LOCK10_EN	**Enable the lock checking process of lock set 10. Before it takes effect, the LOCK_ON must be turned on.** 0: Disable Lock Range check for set 10. 1: Enable Lock Range check for set 10.
19	LOCK09_CS	**Indicate the lock checking process of lock set.n for CS0 or CS1** 0: Lock range check of set 9 for CS0. 1: Lock range check of set 9 for CS1.
18	LOCK09_EN	**Enable the lock checking process of lock set 9. Before it takes effect, the LOCK_ON must be turned on.** 0: Disable Lock Range check for set 9. 1: Enable Lock Range check for set 9.
17	LOCK08_CS	**Indicate the lock checking process of lock set.n for CS0 or CS1** 0: Lock range check of set 8 for CS0. 1: Lock range check of set 8 for CS1.
16	LOCK08_EN	**Enable the lock checking process of lock set 8. Before it takes effect, the LOCK_ON must be turned on.** 0: Disable Lock Range check for set 8. 1: Enable Lock Range check for set 8.
15	LOCK07_CS	**Indicate the lock checking process of lock set.n for CS0 or CS1** 0: Lock range check of set 7 for CS0. 1: Lock range check of set 7 for CS1.
14	LOCK07_EN	**Enable the lock checking process of lock set 7. Before it takes effect, the LOCK_ON must be turned on.**

		0: Disable Lock Range check for set 7. 1: Enable Lock Range check for set 7.
13	LOCK06_CS	**Indicate the lock checking process of lock set.n for CS0 or CS1** 0: Lock range check of set 6 for CS0. 1: Lock range check of set 6 for CS1.
12	LOCK06_EN	**Enable the lock checking process of lock set 6. Before it takes effect, the LOCK_ON must be turned on.** 0: Disable Lock Range check for set 6. 1: Enable Lock Range check for set 6.
11	LOCK05_CS	**Indicate the lock checking process of lock set.n for CS0 or CS1** 0: Lock range check of set 5 for CS0. 1: Lock range check of set 5 for CS1.
10	LOCK05_EN	**Enable the lock checking process of lock set 5. Before it takes effect, the LOCK_ON must be turned on.** 0: Disable Lock Range check for set 5. 1: Enable Lock Range check for set 5.
9	LOCK04_CS	**Indicate the lock checking process of lock set.n for CS0 or CS1** 0: Lock range check of set 4 for CS0. 1: Lock range check of set 4 for CS1.
8	LOCK04_EN	**Enable the lock checking process of lock set 4. Before it takes effect, the LOCK_ON must be turned on.** 0: Disable Lock Range check for set 4. 1: Enable Lock Range check for set 4.
7	LOCK03_CS	**Indicate the lock checking process of lock set.n for CS0 or CS1** 0: Lock range check of set 3 for CS0. 1: Lock range check of set 3 for CS1.
6	LOCK03_EN	**Enable the lock checking process of lock set 3. Before it takes effect, the LOCK_ON must be turned on.** 0: Disable Lock Range check for set 3. 1: Enable Lock Range check for set 3.
5	LOCK02_CS	**Indicate the lock checking process of lock set.n for CS0 or CS1** 0: Lock range check of set 2 for CS0. 1: Lock range check of set 2 for CS1.
4	LOCK02_EN	**Enable the lock checking process of lock set 2. Before it takes effect, the LOCK_ON must be turned on.** 0: Disable Lock Range check for set 2. 1: Enable Lock Range check for set 2.
3	LOCK01_CS	**Indicate the lock checking process of lock set.n for CS0 or CS1** 0: Lock range check of set 1 for CS0. 1: Lock range check of set 1 for CS1.
2	LOCK01_EN	**Enable the lock checking process of lock set 1. Before it takes effect, the LOCK_ON must be turned on.** 0: Disable Lock Range check for set 1. 1: Enable Lock Range check for set 1.
1	LOCK00_CS	**Indicate the lock checking process of lock set.n for CS0 or CS1** 0: Lock range check of set 0 for CS0. 1: Lock range check of set 0 for CS1.
0	LOCK00_EN	**Enable the lock checking process of lock set 0. Before it takes effect, the LOCK_ON must be turned on.** 0: Disable Lock Range check for set 0. 1: Enable Lock Range check for set 0.

1E003108 NFI_LOCKANOB NFI Address Format for Lock Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne		PROG_RADD_NOB				PROG_CADD_NOB				ERASE_RADD_NOB				ERASE_CADD_NOB		

Type		R/W1				R/W1				R/W1				R/W1		
Reset		0	0	0		0	0	0		0	0	0		0	0	0

Bit(s)	Name	Description
14:12	PROG_RADD_NOB	**Number of bytes for the row address for program operation (command is 8'h8X)**
10:8	PROG_CADD_NOB	**Number of bytes for the column address for program operation (command is 8'h8X)**
6:4	ERASE_RADD_NOB	**Number of bytes for the row address for erase operation (command is 8'h6X)**
2:0	ERASE_CADD_NOB	**Number of bytes for the column address for erase operation (command is 8'h6X)**

1E003110 NFI_LOCK00ADD NFI Row Start Address for Lock Set00 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	LOCK00_ROW3								LOCK00_ROW2							
Type	R/W1								R/W1							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	LOCK00_ROW1								LOCK00_ROW0							
Type	R/W1								R/W1							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	LOCK00_ROW3	**The 3-th row start address byte to be locked for lock set 0.**
23:16	LOCK00_ROW2	**The 2-th row start address byte to be locked for lock set 0.**
15:8	LOCK00_ROW1	**The 1-th row start address byte to be locked for lock set 0.**
7:0	LOCK00_ROW0	**The 0-th row start address byte to be locked for lock set 0.**

1E003114 NFI_LOCK00FMT NFI Row Address Format for Lock Set00 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	LOCK00_FMT3								LOCK00_FMT2							
Type	R/W1								R/W1							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	LOCK00_FMT1								LOCK00_FMT0							
Type	R/W1								R/W1							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	LOCK00_FMT3	**The 3-th row address format byte to be locked for lock set 0.**
23:16	LOCK00_FMT2	**The 2-th row address format byte to be locked for lock set 0.**
15:8	LOCK00_FMT1	**The 1-th row address format byte to be locked for lock set 0.**
7:0	LOCK00_FMT0	**The 0-th row address format byte to be locked for lock set 0.**

1E003190 NFI_FIFODATA0 NFI FIFO Content Data 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	FIFO_DATA0[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	FIFO_DATA0[15:0]															

Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	FIFO_DATA0	

1E003194 NFI_FIFODATA1 NFI FIFO Content Data 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	FIFO_DATA1[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	FIFO_DATA1[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	FIFO_DATA1	

1E003198 NFI_FIFODATA2 NFI FIFO Content Data 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	FIFO_DATA2[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	FIFO_DATA2[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	FIFO_DATA2	

1E00319C NFI_FIFODATA3 NFI FIFO Content Data 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	FIFO_DATA3[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	FIFO_DATA3[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	FIFO_DATA3	

1E003200 NFI_MCON NFI LCD Monitor Control Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Mne															BMCLR	BMSTR
Type															WO	R/W
Reset															0	0

Bit(s)	Name	Description
1	BMCLR	**Clear NFI-LCD bandwidth monitor register counter**
0	BMSTR	**Enable NFI-LCD bandwidth monitor** 0: disable. 1: enable.

1E003204 NFI_TOTALCNT NFI LCD Monitor Total Cycle Count 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	NFI_TOTALCNT[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	NFI_TOTALCNT[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	NFI_TOTALCNT	**The total clock cycle count during enabling NFI-LCD bandwidth monitor**

1E003208 NFI_RQCNT NFI LCD Monitor Request Cycle Count 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	NFI_RQCNT[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	NFI_RQCNT[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	NFI_RQCNT	**The request clock cycle count during enabling NFI-LCD bandwidth monitor**

1E00320C NFI_ACCNT NFI LCD Monitor Access Cycle Count 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	NFI_ACCNT[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	NFI_ACCNT[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	NFI_ACCNT	**The access clock cycle count during enabling NFI-LCD bandwidth monitor**

1E003210 NFI_MASTERSTA NFI Master Status 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne					MAS_ADDR			MAS_RD			MAS_WR			MAS_RDDLY		
Type					RO			RO			RO			RO		
Reset					0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
11:9	MAS_ADDR	**MAS_is in the Address phase of AHB protocol. In this phase, Bus gots the address data from Master.** 000b: There is no MAS in the Address phase of AHB protocol. 001b: NFI is in the Address pahse of AHB protocol. 010b: Auto-Correction is in the Address pahse of AHB protocol. 100b: ECC is in the Address pahse of AHB protocol.
8:6	MAS_RD	**MAS_is in the Read DATA phase of AHB protocol. In this phase, Bus returns the read data.**
5:3	MAS_WR	**MAS_is in the Write DATA phase of AHB protocol. In this phase, Bus receives the write data.**
2:0	MAS_RDDLY	**MAS is in the Read DATA delay phase of AHB protocol. In this phase, NFI and ECC got the read back data**

2.8.3 Programming Guide

This section lists the program sequences for the NAND flash operations.

NAND Device Reset

Programming Sequence	Memo
*NFI_INTR_EN = 0x4;	// enable reset complete interrupt
*NFI_CMD = 0xff;	Reset command
*NFI_CNRNB = 0xf1	
Wait for reset complete interrupt	

NFI reset (General)

Programming Sequence	Memo
*NFI_CON = 0x3	The NFI reset to reset all register and force NFI master be early terminated
while (*NFI_MASTERSTA != 0) ;	Wait for master finish the last transaction
*NFI_CON = 0x3	The second NFI reset is to ensure any status register affected by NFI master is reset to normal status

Read ID

Programming Sequence	Memo
NFI Configuration	
*NFI_CON = 0x3;	//Reset NFI before any command. (NFI_state back to IDLE)
*NFI_CNFG = 0x2042;	//Single word PIO read. (set 0x2000 for single byte PIO read).
Write Command and Address to NAND Device	
*NFI_CMD = 0x90;	(NFI_state from IDLE state jump to READDATA state) (Issue command when SW write this APB address)
*NFI_COLADDR = col_addr ;	//column address
*NFI_ROWADDR = row_addr ;	//row address
*NFI_ADDRNOB = addres_byte_num;	//column and row number of bytes.

	(Issue address when the SW write this APB address).
while (*NFI_STA & 0xF != 0) ;	//wait for writing the address to NAND
Trigger the start to read register and NFI start to read status or ID from NAND Device	
*NFI_CON = 0x0090	//set number of read command of single read. If this is x8 NF device, this means read 4*(byte) data. If this is x16 NF device, this means read 4 * (2byte) data.
Read Data from NFI by using PIO Mode	
for (int i = 0 ; i < number of byte ; i++) {	//PIO mode to read out read id.
while (*NFI_PIO_RDY == 0);	//if the pio_rdy is not 1, keep polling.
pio_rdy = *NFI_PIO_RDY;	//if the pio_rdy is equal to 1, data is available for read out.
read_id[i] = *NFI_DATAR;	//The read out data from NFI_DATAR can be byte or word
}	//It depends on the setting of BYTE_RW in NFI_CNFG

Read Status

Programming Sequence	Memo
Configuration	
*NFI_CON = 0x3;	//Reset NFI before any command. (NFI_state back to IDLE)
*NFI_CNFG = 0x2042;	//Single word PIO read. (set 0x2000 for single byte PIO read).
Write Command to NAND Device	
*NFI_CMD = 0x70;	(NFI_state from IDLE state jump to READDATA state) (Issue command when SW write this APB address)
Start to read status or ID from NAND Device	
*NFI_CON = 0x0090	//set number of read command of single read. If this is x8 NF device, this means read 4*(byte) data. If this is x16 NF device, this means read 4 * (2byte) data.
Read Data by using PIO Mode	
for (int i = 0 ; i < number of byte ; i++) {	//PIO mode to read out read id.
while (*NFI_PIO_RDY == 0);	//if the pio_rdy is not 1, keep polling.
pio_rdy = *NFI_PIO_RDY;	//if the pio_rdy is equal to 1, data is available for read out.
read_status[i] = *NFI_DATAR;	//The read out data from NFI_DATAR can be byte or word
}	//It depends on the setting of BYTE_RW in NFI_CNFG

Block Erase

Programming Sequence	Memo
Configuration	
*NFI_CON = 0x3;	//Reset NFI before any command. (NFI state back to IDLE)
*NFI_INTR_EN = 0x8 ;	//Enable erase complete interrupt
*NFI_CNFG = 0x4000;	//erase operation. (NFI_state from IDLE state jump to ERASEDATA state)
Write Command and Address to NAND Device	
*NFI_CMD = 0x60;	// erase first command. (Issue command when SW write this APB address)
*NFI_COLADDR = col_addr ;	//column address
*NFI_ROWADDR = row_addr ;	//row address
*NFI_ADDRNOB = addres_byte_num;	//column and row number of bytes. (Issue address when the SW write this APB address).
while (*NFI_STA & 0xF != 0) ;	//wait for writing the address to NAND

*NFI_CMD = 0xD0;	//erase second command. (NFI state from ERASEDATA state jump to ERASEBUSY state)
*NFI_CNRNB = 0xf1	
After 2nd Command 0xD0, Waiting for Erase Done Interrupt	
Wait for interrupt ……	//After Nand flash from busy to ready will issue the IRQ. (NFI_state back to IDLE)

Page Program (Using DMA Mode)

Configure	**Memo**
ECC Engine Configuration	
if (hw_ecc_en) {	//if hw_ecc_en is needed, set ECC configuration. (reference NFIECC Functional spec)
while (*NFIECC_ENCIDLE == 0);	//Polling IDLE signal until Encoder is available.

*NFIECC_ENCCNFG = 0x10400010;	//Configure Encoder parameter in NFI mode.
	//The setting must be referred to NFIECC document
	//The encode size depends on the FDMECC setting
	//0x10400010 means ENC_MS = 520 (512+8), the setting is used for hwecc_en = 1, FDM_ECC_NUM = 8

*NFIECC_DECCON = 0x0 ;	//make sure Decoder is close.
*NFIECC_ENCCON = 0x1 ;	//enable Encoder.
}	
NFI Configuration	
*NFI_CON = 0x3;	//Reset NFI before any command. (NFI state back to IDLE)
*NFI_PAGEFMT = 0x8800;	//Set device format according to NF device and fdm number.
*NFI_ACCCON = 0x41_0112;	//Set device access control according to NF device.
if (custom_mode) {	
*NFI_CNFG = (0x6001 | auto_fmt_en | hw_ecc_en | nfi_dma_burst)	
} else {	
*NFI_CNFG = (0x3001 | auto_fmt_en | hw_ecc_en | nfi_dma_burst)	//Setting NFI configuration according your usage.
}	
*NFI_CON = 0x4000;	//Set the length of Burst write
if (auto_fmt_en) {	
*NFI_FDMxx = FDM_value	//set FDM value
}	
Write Command and Address to NAND Device	
*NFI_CMD = 0x80;	// write first command. (Issue command when SW write this APB address) NFI_state from IDLE state jumps to PROGDATA state.
*NFI_COLADDR = col_addr ;	//column address
*NFI_ROWADDR = row_addr ;	//row address
*NFI_ADDRNOB = addres_byte_num;	//column and row number of bytes. (Issue address when the SW write this APB address).
while (*NFI_STA & 0xF != 0) ;	//wait for writing the address to NAND
*NFI_STRADDR = 0x1000_0000	//Write Data start address
Setting Interrupt,	
*NFI_INTR_EN = 0x0042;	//Set ahb_done and wr_done interrupt. Ahb_done interrupt

	must be set after setting the length of burst write in NFI_CON and before burst write strobe.
Trigger the BWR register and NFI start to write the data into NAND device	
*NFI_CON = *NFI_CON | 0x0200;	//Set Burst write strobe.
if (custom_mode) *NFI_STRDATA = 1;	//strobe to transfer data
wait for ahb done interrupt	//After Nand flash finishing transferring data from ahb bus into NFI FIFO, AHB_DONE interrupt will be issued
while (*NFI_ADDRCNTR != expected_nfi_tsf_num);	//polling when nf_tsf_num is not equal to expted transfer data number. This action is to guarantee all the data in NFI FIFO has been written into NAND device.
*NFI_CMD = 0x10;	// write second command. (Issue command when SW write this APB address) NFI_state from PROGDATA jumps to PROGBUSY.
*NFI_CNRNB = 0xf1	
Waiting for wr_done interrupt	//After Nand flash from busy to ready will issue ready_return IRQ, also the write_done IRQ. NFI state from PROGBUSY jumps to IDLE.

Page Program (Using PIO Mode)

Configure	**Memo**
ECC Engine Configuration	
if (hw_ecc_en) {	//if hw_ecc_en is needed, set ECC configuration. (reference NFIECC Functional spec)
while (*NFIECC_ENCIDLE == 0);	//Polling IDLE signal until Encoder is available.

*NFIECC_ENCCNFG = 0x10400010;	//Configure Encoder parameter in NFI mode.
	//The setting must be referred to NFIECC document
	//The encode size depends on the FDMECC setting

*NFIECC_DECCON = 0x0 ;	//make sure Decoder is close.
*NFIECC_ENCCON = 0x1 ;	//enable Encoder.
}	
NFI Configuration	
*NFI_CON = 0x3;	//Reset NFI before any command. (NFI state back to IDLE)
*NFI_PAGEFMT = 0x8800;	//Set device format according to NF device and fdm number.
*NFI_ACCCON = 0x41_0112;	//Set device access control according to NF device.
if (custom_mode) {	
*NFI_CNFG = (0x6000 | auto_fmt_en | hw_ecc_en)	
} else {	
*NFI_CNFG = (0x3000 | auto_fmt_en | hw_ecc_en)	//Setting NFI configuration according your usage.
}	
*NFI_CON = 0x4000;	//Set the length of Burst write
if (custom_mode) *NFI_STRDATA = 1;	//strobe to transfer data
if (auto_fmt_en) {	
*NFI_FDMxx = FDM_value	//set FDM value
}	
Write Command and Address to NAND Device	
*NFI_CMD = 0x80;	// write first command.

	(Issue command when SW write this APB address) NFI_state from IDLE state jumps to PROGDATA state.
*NFI_COLADDR = col_addr ;	//column address
*NFI_ROWADDR = row_addr ;	//row address
*NFI_ADDRNOB = addres_byte_num;	//column and row number of bytes. (Issue address when the SW write this APB address).
while (*NFI_STA & 0xF != 0) ;	//wait for writing the address to NAND
*NFI_STRADDR = 0x1000_0000	//Write Data start address
Setting Interrupt,	
*NFI_INTR_EN = 0x0042;	//Set ahb_done and wr_done interrupt. Ahb_done interrupt must be set after setting the length of burst write in NFI_CON and before burst write strobe.
Trigger the BWR register and NFI start to write the data into NAND device	
*NFI_CON = *NFI_CON | 0x0200;	//Set Burst write strobe.
Read Data by using PIO Mode	

for (int i = 0; i < sec_num *(512+spare_size) ; i++) {	//PIO mode.
	// if (~autofmt_en) transfer size = sec_num*(512+spare_size)
	// if (autofmt_en) transfer size = sec_num*(512+(spare_size-fdm_num))
	// The parity data in spare area must be read from NFIECC register by MCU in PIO mode.

while (*NFI_PIO_RDY ==0x0)	//if the pio_rdy is not 1, keep polling.
pio_rdy = *NFI_PIO_RDY;	
*NFI_DATAW = nfi_data[i];	//if the pio_rdy is equal to 1, data is available for write in.
}	
Check end condition and write 2nd command to NAND device	
while (*NFI_ADDRCNTR != expected_nfi_tsf_num);	//polling when nf_tsf_num is not equal to expted transfer data number. This action is to guarantee all the data in NFI FIFO has been written into NAND device.
*NFI_CMD = 0x10;	// write second command. (Issue command when SW write this APB address) NFI_state from PROGDATA jumps to PROGBUSY.
*NFI_CNRNB = 0xf1	
Waiting for wr_done interrupt	//After Nand flash from busy to ready will issue ready_return IRQ, also the write_done IRQ. NFI state from PROGBUSY jumps to IDLE.

Page Read (DMA Mode)

Configure	**Memo**
ECC Engine Configuration	
if (hw_ecc_en) {	//if hw_ecc_en is needed, set ECC configuration. (reference NFIECC Functional spec)
dec_idle = *NFIECC_DECIDLE ;	//Polling IDLE signal until Decoder is available.
while (dec_idle==0) ;	
dec_idle = *NFIECC_DECIDLE ;	
*NFIECC_DECCNFG = 0x90743010;	//Configure Decoder parameter in NFI mode.
*NFIECC_ENCCON = 0x0 ;	//make sure Encoder is close.
*NFIECC_DECCON = 0x1 ;	//enable Decoder.

}	
NFI Configuration	
*NFI_CON = 0x3;	//Reset NFI before any command. (NFI state back to IDLE)
*NFI_PAGEFMT = 0x8800;	//Set device format according to NF device and fdm number.
*NFI_ACCCON = 0x41_0112;	//Set device access control according to NF device.
if (custom_mode) {	
*NFI_CNFG= (0x3003 | auto_fmt_en | hw_ecc_en | nfi_dma_burst)	
} else {	
*NFI_CNFG= (0x1003 | auto_fmt_en | hw_ecc_en | nfi_dma_burst)	//Setting NFI configuration according your usage.
}	
*NFI_INTR_EN = 0x0010;	//Set busy_return interrupt
*NFI_STRADDR = 0x1000_0000	//Read Data start address
*NFI_CON = 0x4000;	//Set length of Burst read
Write Command and Address to NAND Device	
*NFI_CMD = 0x00;	// read first command. (Issue command when SW write this APB address) NFI_state from IDLE state jumps to READBUSY state.
*NFI_COLADDR = col_addr ;	//column address
*NFI_ROWADDR = row_addr ;	//row address
*NFI_ADDRNOB = addres_byte_num;	//column and row number of bytes. (Issue address when the SW write this APB address).
while (*NFI_STA &0xf !== 0) ;	//polling when programming state is not equal to 0.
*NFI_CMD = 0x30;	// read second command. (Issue command when SW write this APB address)
*NFI_CNRNB = 0xf1	
Wait for ready_return interrupt	//After Nand flash from busy to ready will issue ready_return IRQ. NFI state from READBUSY jumps to READDATA.
Trigger the BWR register and NFI start to write the data into NAND device	
*NFI_CON = *NFI_CON | 0x0100;	//Set Burst read strobe.
wait for ahb_done interrupt	//After Nand flash from busy to ready will issue the IRQ.
nfi_tsf_num = *NFI_BYTELEN;	//**Polling bytelen**; nf_tsf_num should be equal to expected transfer data number.
while (nfi_tsf_num !== expected_nfi_tsf_num)	//polling when nf_tsf_num is not equal to expted transfer data number.
nfi_tsf_num = *NFI_BYTELEN;	
nfi_irq_sta = *NFI_INTR;	//read AHB_done IRQ and read_done IRQ. nfi_irq_sta should be 0x41;
if (hw_ecc_en)	//if hw_ecc_en is needed, set ECC configuration. (reference NFIECC Functional spec)
dec_done = *NFIECC_DECDONE ;	//Polling ECC Decoder Done signal.
while (dec_done !==0xf) ;	//Because the sec_num is 4; according to sec_num;
dec_done = *NFIECC_DECDONE;	
endif	

Page Read (PIO Mode)

Configure	Memo
ECC Engine Configuration	
if (hw_ecc_en) {	//if hw_ecc_en is needed, set ECC configuration. (reference NFIECC Functional spec)
dec_idle = *NFIECC_DECIDLE ;	//Polling IDLE signal until Decoder is available.
while (dec_idle==0) ;	
dec_idle = *NFIECC_DECIDLE ;	
*NFIECC_DECCNFG = 0x90743010;	//Configure Decoder parameter in NFI mode.
*NFIECC_ENCCON = 0x0 ;	//make sure Encoder is close.
*NFIECC_DECCON = 0x1 ;	//enable Decoder.
}	
NFI Configuration	
*NFI_CON = 0x3;	//Reset NFI before any command. (NFI state back to IDLE)
*NFI_PAGEFMT = 0x8800;	//Set device format according to NF device and fdm number.
*NFI_ACCCON = 0x41_0112;	//Set device access control according to NF device.
if (custom_mode) {	
*NFI_CNFG= (0x6003 | auto_fmt_en | hw_ecc_en | nfi_dma_burst)	
} else {	
*NFI_CNFG= (0x1003 | auto_fmt_en | hw_ecc_en | nfi_dma_burst)	//Setting NFI configuration according your usage.
}	
*NFI_INTR_EN = 0x0010;	//Set busy_return interrupt
*NFI_STRADDR = 0x1000_0000	//Read Data start address
*NFI_CON = 0x4000;	//Set length of Burst read
Write Command and Address to NAND Device	
*NFI_CMD = 0x00;	// read first command. (Issue command when SW write this APB address) NFI_state from IDLE state jumps to READBUSY state.
*NFI_COLADDR = col_addr ;	//column address
*NFI_ROWADDR = row_addr ;	//row address
*NFI_ADDRNOB = addres_byte_num;	//column and row number of bytes. (Issue address when the SW write this APB address).
while (*NFI_STA &0xf !== 0) ;	//polling when programming state is not equal to 0.
*NFI_CMD = 0x30;	// read second command. (Issue command when SW write this APB address)
*NFI_CNRNB = 0xf1	
Wait for ready_return interrupt	//After Nand flash from busy to ready will issue ready_return IRQ. NFI state from READBUSY jumps to READDATA.
Trigger the BWR register and NFI start to write the data into NAND device	
*NFI_CON = *NFI_CON | 0x0100;	//Set Burst read strobe.
for (int i = 1 ; i < sec_num*(512+fdm_size) ; i++){	//PIO mode.
pio_rdy = *NFI_PIO_RDY;	//poling pio_ready register.
while (pio_rdy==0x0)	//if the pio_rdy is not 1, keep polling.
pio_rdy = *NFI_PIO_RDY;	
*NFI_DATAR = nfi_data[i];	//if the pio_rdy is equal to 1, data is available for write in.
}	

while (nfi_irq_b==0x1);	//After NFI has read out all data from Nand Flash then issue read_done IRQ.
nfi_irq_sta = *NFI_INTR;	//read read_done IRQ. nfi_irq_sta should be 0x01;
nfi_tsf_num = *NFI_BYTELEN;	//**Polling bytelen**; nf_tsf_num should be equal to expected transfer data number. (make sure the FDM data is all read out)
while (nfi_tsf_num !== expected_nfi_tsf_num)	//polling when nf_tsf_num is not equal to expted transfer data number.
nfi_tsf_num = *NFI_BYTELEN;	
dec_done = *NFIECC_DECDONE ;	//Polling ECC Decoder Done signal.
while (dec_done !==0xf) ;	//Because the sec_num is 4; according to sec_num;
dec_done = *NFIECC_DECDONE;	
endif	

Two block erase (Pseudo-Code)

Configure	Memo
Set Interrupt Reg (0x10)	//Set busy to ready interrupt
Write command to NAND (0x60)	
Write address to NAND (First block address)	
Write command to NAND (0x60)	
Write address to NAND (Second block address)	
Write Command to NAND (0xd0)	
Waiting for Interrupt …	

Multiplane Page Program (Pseudo-Code)

Configure	Memo
This reference pseudo code is for TOSHIBA NAND device	
Write 1st Page to NAND Device	
Write command to NAND (0x80)	
Write address to NAND (The 1st address)	Ex: NFI_ROWADDR = 0x100
Program A page of data by using Custom Mode	//Please refer to the reference code of page program with custom_mode enabled
Set Interrupt Reg (0x10)	//Set busy to ready interrupt
Write_command(0x11)	
Set NFI CNRNB for waiting B2R Event	// Set hardware time-out for detecting the busy to ready of NAND Flash event
Wait for B2R interrupt	
Write 2nd Page to NAND Device	
Reset NFI	
Write command to NAND (0x80)	
Write address to NAND (The 2nd address)	Ex: NFI_ROWADDR = 0x20100
Program A page of data by using Custom Mode	//Please refer to the reference code of page program with custom_mode enabled
Set Interrupt Reg (0x10)	//Set busy to ready interrupt
Write_command(0x11)	
Set NFI CNRNB for waiting B2R Event	// Set hardware time-out for detecting the busy to ready of NAND Flash event
Wait for B2R interrupt	
Write 3rdPage to NAND Device	

Reset NFI	
Write command to NAND (0x80)	
Write address to NAND (The 3rd address)	Ex: NFI_ROWADDR = 0x101
Program A page of data by using Custom Mode	//Please refer to the reference code of page program with custom_mode enabled
Set Interrupt Reg (0x10)	//Set busy to ready interrupt
Write_command(0x11)	
Set NFI CNRNB for waiting B2R Event	// Set hardware time-out for detecting the busy to ready of NAND Flash event
Wait for B2R interrupt	
Write 4th Page to NAND Device	
Reset NFI	
Write command to NAND (0x80)	
Write address to NAND (The 4th address)	Ex: NFI_ROWADDR = 0x20101
Program A page of data by using Custom Mode	//Please refer to the reference code of page program with custom_mode enabled
Set Interrupt Reg (0x10)	//Set busy to ready interrupt
Write_command(0x11)	
Set NFI CNRNB for waiting B2R Event	// Set hardware time-out for detecting the busy to ready of NAND Flash event
Wait for B2R interrupt	

2.9 NFI ECC Controller

2.9.1 Features

- ECC (BCH code) acceleration is capable of 4 bits correction in one full or shorten ECC coded block size which is less than 8192 (<8192bits)
- Support data input in 8 bits in NFI mode and 32 bits in DMA / PIO mode and works in 122.88MHz.
- Support encoder and decoder work separately in DMA and PIO mode and automatic error correction.

2.9.2 Registers

Address	Name	Width	Register Function
1E003800	NFIECC_ENCCON	16	**NFIECC Encoder Control Register** This register is for Encoder control.
1E003804	NFIECC_ENCCNFG	32	**NFIECC Configure Register** This register is for NFIECC encoder configuration.
1E003808	NFIECC_ENCDIADDR	32	**NFIECC Encoder DI Memory Address Register** The register indicates the data start address of input data to the Encoder AHB mode.
1E00380C	NFIECC_ENCIDLE	16	**NFIECC Encoder Idle Status Register** This register is for NFIECC Encoder idle status.
1E003810	NFIECC_ENCPAR0	32	**NFIECC Parity0 Register** The register indicates the highest order of parity bits
1E003814	NFIECC_ENCPAR1	32	**NFIECC Parity1 Register** The register indicates the parity bits
1E003818	NFIECC_ENCPAR2	32	**NFIECC Parity2 Register** The register indicates the parity bits
1E00381C	NFIECC_ENCPAR3	32	**NFIECC Parity3 Register** The register indicates the parity bits
1E003820	NFIECC_ENCPAR4	32	**NFIECC Parity4 Register** The register indicates the parity bits
1E003824	NFIECC_ENCSTA	32	**NFIECC Encoder Status Register** This register is for NFIECC Encoder status for SW polling.
1E003828	NFIECC_ENCIRQEN	16	**NFIECC Encoder IRQ enable Register** This register is for software programmer to enable NFIECC IRQ signals (ignore in NFI mode)
1E00382C	NFIECC_ENCIRQSTA	16	**NFIECC Encoder IRQ status Register** This register is for software programmer tracking NFIECC IRQ status. (ignore in NFI mode)
1E003880	NFIECC_PIO_DIRDY	16	**NFIECC PIO Data Ready Register** This register indicates the data is ready for input
1E003884	NFIECC_PIO_DI	32	**NFIECC PIO Data Register** The register indicates PIO mode data input by MCU
1E003900	NFIECC_DECCON	16	**NFIECC Decoder Control Register** This register is for Decoder control.
1E003904	NFIECC_DECCNFG	32	**NFIECC Decoder Configure Register** This register is for NFIECC configuration.
1E003908	NFIECC_DECDIADDR	32	**NFIECC Decoder DI Memory Address Register** The register indicates the data start address of input data to the Decoder AHB mode.
1E00390C	NFIECC_DECIDLE	16	**NFIECC Decoder Idle Status Register** This register indicates the Decoder Idle status.
1E003910	NFIECC_DECFER	16	**NFIECC Decoder Found Error Status Register** This register is for NFIECC Decoder status.
1E003914	NFIECC_DECENUM	32	**NFIECC Decode Error Number Register** The register indicates the error number of the coded block.
1E003918	NFIECC_DECDONE	16	**NFIECC Decoder Error Status Register** This register is for NFIECC Decoder done status.
1E00391C	NFIECC_DECEL0	32	**NFIECC Decoder Error location0 Register**

			The register indicates the error location of the decoding result
1E003920	NFIECC_DECEL1	32	**NFIECC Decoder Error Location1 Register** The register indicates the error location of the decoding result.
1E003924	NFIECC_DECEL2	32	**NFIECC Decoder Error Location2 Register** The register indicates the error location of the decoding result.
1E003928	NFIECC_DECEL3	32	**NFIECC Decoder Error Location3 Register** The register indicates the error location of the decoding result.
1E00392C	NFIECC_DECEL4	32	**NFIECC Decoder Error Location4 Register** The register indicates the error location of the decoding result.
1E003930	NFIECC_DECEL5	32	**NFIECC Decoder Error Location5 Register** The register indicates the error location of the decoding result.
1E003934	NFIECC_DECIRQEN	16	**NFIECC Decoder IRQ enable Register** This register is for software programmer to enable NFIECC IRQ signals (ignore in NFI mode)
1E003938	NFIECC_DECIRQSTA	16	**NFIECC Decoder IRQ status Register** This register is for software programmer tracking NFIECC IRQ status. (ignore in NFI mode)
1E00393C	NFIECC_FDMADDR	32	**NFIECC FDM Register Address** The register indicates the address of FDM data in NFI module.
1E003940	NFIECC_DECFSM	32	**NFIECC Decoder FSM** The register indicates the finite state machine status of decoder.
1E003944	NFIECC_SYNSTA	32	**NFIECC Syndrome Status Register** This register is for NFIECC Syndrom status.
1E003948	NFIECC_NFIDIDECNFIDI	32	**NFIECC NFI input dataNFI input data Register** This register is for checking NFI input data.
1E00394C	NFIECC_SYN0	32	**NFIECC Syndrom Register** The register indicates the error location of the decoding result.

1E003800 NFIECC_ENCCON NFIECC Encoder Control Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne																ENC_EN
Type																R/W
Reset																0

Bit(s)	Name	Description
0	ENC_EN	**indicates the enable in NFI mode and start to work in AHB mode. In AHB mode, parity bits is remained in the PAR0~PAR4 register field until the ENC_EN is deasserted to 0.** 0: means disable the Encode block. 1: means enable the Encode block. In AHB mode, the Encoder starts to fetch data when the register changes from 0 to 1. In NFI mode, the register enables the Encode block, and then the Encoder module waits start signal and data from NFI.

1E003804 NFIECC_ENCCNFG NFIECC Configure Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne				ENC_MS												
Type				R/W												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne								ENC_BURST_EN			ENC_MODE			ENC_TNUM		
Type								R/W R/W			R/W			R/W		
Reset								0			0	0		0	0	0

Bit(s)	Name	Description
28:16	ENC_MS	**indicates the total bit size of message block including main data and control(FDM) data in the NFI mode. The spare_ECC_num parameter in old version has been merged into the message block_size parameter. If the block_size is equal to zero, the NFIECC do nothing.** The acceptable coded block size, which includes data and parity bits size, is 1~8191bits. Different ENC_TNUM results in different parity bits, and also results in different maximum message block size.
8	ENC_BURST_EN	**indicates the burst enalbe.** 0: means DMA mode uses single read. 1: means DMA mode uses burst read.
5:4	ENC_MODE	**indicates the data source from access through AHB bus or from NFI.** 00b: means source data from access through Bus. (DMA mode) 01b: means source data from NFI module. (NFI mode) 10b: means source data is written by MCU. (PIO mode) 11b: reserved mode.
2:0	ENC_TNUM	**indicates the correct capability in one block size. (Remove)** 0: means the NFIECC is capable of correct 4 bits in one block size. 1: means the NFIECC is capable of correct 6 bits in one block size. 2: means the NFIECC is capable of correct 8 bits in one block size. 3: means the NFIECC is capable of correct 10 bits in one block size. 4: means the NFIECC is capable of correct 12 bits in one block size.

1E003808 NFIECC_ENCDIADDR NFIECC Encoder DI Memory Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	ENC_DIADDR[29:14]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	ENC_DIADDR[13:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0		

Bit(s)	Name	Description
31:2	ENC_DIADDR	**indicates the memory address of input data to Encoder block in AHB mode. (4-Byte align)**

1E00380C NFIECC_ENCIDLE NFIECC Encoder Idle Status Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne																ENC_IDLE
Type																R
Reset																0

Bit(s)	Name	Description

0	ENC_IDLE	**indicates the Encode block in idle state and ready for new message block.** 0: means the Encode block is under working. 1: means the Encode block is in Idle state and available for new message block.

1E003810 NFIECC_ENCPAR0 NFIECC Parity0 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	ENC_PAR0[31:16]															
Type	R															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	ENC_PAR0[15:0]															
Type	R															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	ENC_PAR0	**indicates the highest order of output parity bits and the bit 0 is the highest order of parity bit. The PAR0~PAR4 register is remain the last message block parity bits until ENC_EN is deasserted. The parity bits should append after main data by order of {PAR0[31:0], PAR1[31:0], PAR2[31:0], PAR3[31:0], PAR4[31:4], 4'b0}, The redundant bit of parity bit will be padded by 0.**

1E003814 NFIECC_ENCPAR1 NFIECC Parity1 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	ENC_PAR1[31:16]															
Type	R															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	ENC_PAR1[15:0]															
Type	R															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	ENC_PAR1	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1E003818 NFIECC_ENCPAR2 NFIECC Parity2 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	ENC_PAR2[31:16]															
Type	R															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	ENC_PAR2[15:0]															
Type	R															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	ENC_PAR2	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1E00381C NFIECC_ENCPAR3 NFIECC Parity3 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	ENC_PAR3[31:16]															
Type	R															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	ENC_PAR3[15:0]															
Type	R															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	ENC_PAR3	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1E003820 NFIECC_ENCPAR4 NFIECC Parity4 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne					ENC_PAR4[27:16]											
Type					R											
Reset					0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	ENC_PAR4[15:0]															
Type	R															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
27:0	ENC_PAR4	**indicates the parity bits and the 31 is the highest order of parity bit.**

1E003824 NFIECC_ENCSTA NFIECC Encoder Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne			COUNT_MS													
Type			R													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	COUNT_PS										ENC_FSM					
Type	R										R					
Reset	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0

Bit(s)	Name	Description
29:16	COUNT_MS	**indicates the remaining un-processing message bits.**
15:7	COUNT_PS	**indicates the parity bits that have not read out from NFI.**
5:0	ENC_FSM	**indicates encoder finite state machine state** 6'd0: IDLE 6'd1: WAITIN 6'd2: BUSY 6'd4: PAROUT

1E003828 NFIECC_ENCIRQ NFIECC Encoder IRQ enable Register 0000

EN

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne																ENC_IRQEN
Type																R/W
Reset																0

Bit(s)	Name	Description
0	ENC_IRQEN	**Encoder IRQ mask: triggered when Encoder operation is completed.** 0: Disable 1: Enable

1E00382C NFIECC_ENCIRQSTA NFIECC Encoder IRQ status Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne																ENC_IRQSTA
Type																RC
Reset																0

Bit(s)	Name	Description
0	ENC_IRQSTA	**indicates interrupt status for Encoder processing.** 0: No interrupt is generated. 1: An interrupt is pending and waiting for service. Active when Encoder processing is done.

1E003880 NFIECC_PIO_DIRDY NFIECC PIO Data Ready Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne																PIO_DI_RDY
Type																R
Reset																0

Bit(s)	Name	Description
0	PIO_DI_RDY	**indicates the PIO mode (Encoder/Decoder) is ready for input data.** 0: ECC is busy. During busy state, NFIECC_PIO_DI should not be over-write. 1: ECC is ready for input data. In PIO mode, write next PIO_DI when pio_di_rdy is equal to 1.

1E003884 NFIECC_PIO_DI NFIECC PIO Data Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	PIO_DI[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	PIO_DI[15:0]															
Type	R/W															

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PIO_DI	**indicates the PIO mode data input.**

1E003900 NFIECC_DECCON NFIECC Decoder Control Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne																DEC_EN
Type																R/W
Reset																0

Bit(s)	Name	Description
0	DEC_EN	**indicates the enable in NFI mode and start to work in AHB mode. In AHB mode, the decode-status FER and error number registers and error location registers will be reset to 0 when DEC_EN is deasserted.** 0: means disable the Decode block. 1: means enable the Decode block. In AHB mode, the Decoder starts to fetch data when the register changes from 0 to 1. In NFI mode, the register enables the Decode block, and then the Decoder module waits start signal and data from NFI.

1E003904 NFIECC_DECCNFG NFIECC Decoder Configure Register 00003000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	DEC_EMPTY_EN			DEC_CS												
Type	R/W			R/W												
Reset	0			0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne			DEC_CON					DEC_BURST_EN			DEC_MODE			DEC_TNUM		
Type			R/W					R/W R/W			R/W			R/W		
Reset			1	1				0			0	0		0	0	0

Bit(s)	Name	Description
31	DEC_EMPTY_EN	**indicates the Decoder automatically detects the empty source data and by pass the auto-correction block (data are all equal to 1). (ignore in AHB_mode)** 0: means disenable the detection of empty source data. 1: means enable the detection of empty source data.
28:16	DEC_CS	**indicates the total bit size of coded block including protected data and parity bits. The acceptable coded block size is 1~8191bits. If the coded block size is equal to zero, the decoder does nothing. The detail figure shows in Figure 2.**
13:12	DEC_CON	**indicates the bypass configuration in decoding processor.** 0: is reserved 1: means only active syndrome calculator for error detecting purpose. ECC reports DONE and FER status after syndrome calculator is done. 2: means error-correction module is bypassed for being aware of error location purpose. ECC reports DONE, FER, EL and ERRNUM status after Chien search is done.

		3: means the ECC processor decoded data and auto-correction error data. The data address is signaled by DEC_DIADDR register in AHB mode and NFI_DIADDR in NFI mode. ECC reports DONE, FER, EL and ERRNUM status after error-correction is done.
8	DEC_BURST_EN	**indicates the burst enalbe.** 0: means DMA mode uses single read. 1: means DMA mode uses burst read.
5:4	DEC_MODE	**indicates the data source from access AHB bus or from NFI.** 00b: means source data from access through Bus. (DMA mode) 01b: means source data from NFI module. (NFI mode) 10b: means source data is written by MCU. (PIO mode) 11b: Reserved mode.
2:0	DEC_TNUM	**indicates the correct capability in one block size.** 0: means the Decoder is capable of correct 4 bits in one block size. 1: means the Decoder is capable of correct 6 bits in one block size. 2: means the Decoder is capable of correct 8 bits in one block size. 3: means the NFIECC is capable of correct 10 bits in one block size. 4: means the NFIECC is capable of correct 12 bits in one block size.

1E003908 NFIECC_DECDIADDR NFIECC Decoder DI Memory Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	DEC_DIADDR[29:14]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	DEC_DIADDR[13:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0		

Bit(s)	Name	Description
31:2	DEC_DIADDR	**indicates the memory address of input data to the Decoder block in AHB mode. (4-Byte align).**

1E00390C NFIECC_DECIDLE NFIECC Decoder Idle Status Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne																DEC_IDLE
Type																R
Reset																0

Bit(s)	Name	Description
0	DEC_IDLE	**indicates the Decode block is in idle state and ready for new coded block.** 0: means the Decode block is under working. 1: means the Decode block is in idle state and available for new coded block.

1E003910 NFIECC_DECFER NFIECC Decoder Found Error Status Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne									FER7	FER6	FER5	FER4	FER3	FER2	FER1	FER0
Type									R	R	R	R	R	R	R	R
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7	FER7	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
6	FER6	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
5	FER5	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
4	FER4	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
3	FER3	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
2	FER2	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
1	FER1	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
0	FER0	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.

1E003914 NFIECC_DECENUM NFIECC Decode Error Number Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	ERRNUM7				ERRNUM6				ERRNUM5				ERRNUM4			
Type	R				R				R				R			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	ERRNUM3				ERRNUM2				ERRNUM1				ERRNUM0			
Type	R				R				R				R			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:28	ERRNUM7	**indicates the error numbers of coded block in one start signal. 4'hf means the error is**

		uncorrectable. But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.
27:24	ERRNUM6	**indicates the error numbers of coded block in one start signal. 4'hf means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.
23:20	ERRNUM5	**indicates the error numbers of coded block in one start signal. 4'hf means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.
19:16	ERRNUM4	**indicates the error numbers of coded block in one start signal. 4'hf means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.
15:12	ERRNUM3	**indicates the error numbers of coded block in one start signal. 4'hf means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.
11:8	ERRNUM2	**indicates the error numbers of coded block in one start signal. 4'hf means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.
7:4	ERRNUM1	**indicates the error numbers of coded block in one start signal. 4'hf means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.
3:0	ERRNUM0	**indicates the error numbers of coded block in one start signal. 4'hf means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.

1E003918 NFIECC_DECDONE NFIECC Decoder Error Status Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne									DONE7	DONE6	DONE5	DONE4	DONE3	DONE2	DONE1	DONE0
Type									R	R	R	R	R	R	R	R
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7	DONE7	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
6	DONE6	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
5	DONE5	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.

4	DONE4	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
3	DONE3	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
2	DONE2	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
1	DONE1	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
0	DONE0	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.

1E00391C NFIECC_DECEL0 NFIECC Decoder Error location0 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne				DEC_EL10												
Type				R												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne				DEC_EL01												
Type				R												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28:16	DEC_EL10	**indicates the error location 1 of the decoding result.**
12:0	DEC_EL01	**indicates the error location 0 of the decoding result. The EL remains until the DEC_EN is deasserted to 0 in both AHB and NFI mode. When the error number is less than 12, error location registers will be filled from DEC_EL0, and the redundant register fields remain 0.**

1E003920 NFIECC_DECEL1 NFIECC Decoder Error Location1 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne				DEC_EL3												
Type				R												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne				DEC_EL2												
Type				R												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28:16	DEC_EL3	**indicates the error location 3 of the decoding result.**

12:0	DEC_EL2	**indicates the error location 2 of the decoding result.**

1E003924 NFIECC_DECEL2 NFIECC Decoder Error Location2 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne				DEC_EL5												
Type				R												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne				DEC_EL4												
Type				R												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28:16	DEC_EL5	**indicates the error location 5 of the decoding result.**
12:0	DEC_EL4	**indicates the error location 4 of the decoding result.**

1E003928 NFIECC_DECEL3 NFIECC Decoder Error Location3 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne				DEC_EL7												
Type				R												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne				DEC_EL6												
Type				R												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28:16	DEC_EL7	**indicates the error location 7 of the decoding result.**
12:0	DEC_EL6	**indicates the error location 6 of the decoding result.**

1E00392C NFIECC_DECEL4 NFIECC Decoder Error Location4 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne				DEC_EL9												
Type				R												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne				DEC_EL8												
Type				R												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28:16	DEC_EL9	**indicates the error location 9 of the decoding result.**
12:0	DEC_EL8	**indicates the error location 8 of the decoding result.**

1E003930 NFIECC_DECEL5 NFIECC Decoder Error Location5 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne				DEC_EL11												

Type				R												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne				DEC_EL10												
Type				R												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28:16	DEC_EL11	**indicates the error location 11 of the decoding result.**
12:0	DEC_EL10	**indicates the error location 10 of the decoding result.**

1E003934 NFIECC_DECIRQEN NFIECC Decoder IRQ enable Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne																DEC_IRQEN
Type																R/W
Reset																0

Bit(s)	Name	Description
0	DEC_IRQEN	**Decoder IRQ mask: triggered when Decoder operation is completed.** 0: Disable 1: Enable

1E003938 NFIECC_DECIRQSTA NFIECC Decoder IRQ status Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne																DEC_IRQSTA
Type																RC
Reset																0

Bit(s)	Name	Description
0	DEC_IRQSTA	**indicates Interrupt status for Decoder processing.** 0: No interrupt is generated. 1: An interrupt is pending and waiting for service. Active when Decoder processing is done.

1E00393C NFIECC_FDMADDR NFIECC FDM Register Address 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	FDM_ADDR[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	FDM_ADDR[15:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	FDM_ADDR	**indicates the APB register address of FDM data in NFI module.**

1E003940 NFIECC_DECFSM NFIECC Decoder FSM 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne				AUTOC_FSM								CHIEN_FSM				
Type				R/W								R				
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne				BMA_FSM							SYN_FSM					
Type				R/W							R					
Reset				0	0	0	0	0			0	0	0	0	0	0

Bit(s)	Name	Description
28:24	AUTOC_FSM	**indicates the status of auto-correction stage.** 5'd0: IDLE 5'd1: READ 5'd2: WRITE 5'd4: DONE
20:16	CHIEN_FSM	**indicates the status of Chien search stage.** 5'd0: IDLE 5'd1: BUSY 5'd2: DONE
12:8	BMA_FSM	**indicates the status of BMA stage.** 5'd0: IDLE 5'd1: BUSY 5'd2: DONE
5:0	SYN_FSM	**indicates the status of syndrome stage.** 6'd0: IDLE 6'd1: WAITIN 6'd2: BUSY 6'd4: DONE

1E003944 NFIECC_SYNSTA NFIECC Syndrome Status Register NA

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	SYN_SNUM						DIBW							NFI_SEC_NUMNFI_SEC_NUM		
Type	0						R							RR		
Reset	R	R	R				0	0	0	0	0	0		0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	NFI_STR_SET		SYN_COUNT_CS													
Type	R		R													
Reset	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:29	SYN_SNUM	**indicates the sector number recorded by syndrome.**
25:20	DIBW	**indicates input bandwidth.**
18:16	NFI_SEC_NUMNFI_SEC_NUM	**indicates the sector number from NFI.**
15	NFI_STR_SET	**indicates the NFI_STR signal from NFI.**

13:0	SYN_COUNT_CS	**indicates the remaining un-processing coded block bits.**

1E003948 NFIECC_NFIDIDE NFIECC NFI input dataNFI input data Register NA
CNFIDI

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	NFI_DINFI_DI[31:16]															
Type	0															
Reset	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	NFI_DINFI_DI[15:0]															
Type	0															
Reset	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR

Bit(s)	Name	Description
31:0	NFI_DINFI_DI	**indicates the latest 4 byte input data from nfi.**

1E00394C NFIECC_SYN0 NFIECC Syndrom Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne				DEC_SYN3												
Type				R												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne				DEC_SYN1												
Type				R												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28:16	DEC_SYN3	**informs the syndrome 3 from syndrome calculator.**
12:0	DEC_SYN1	**informs the syndrome 1 from syndrome calculator.**

2.9.3 Programming Guide

This section lists the program sequences for ECC operations.

Encoding in NFI mode

Caution: Before NFI address phase enable and configure ECC.

Configure	Memo
*NFIECC_ENCCNFG = 0x10400010;	//configure Encoder parameter in NFI mode.
*NFIECC_ENCCON = 0x1 ;	//enable Encoder.
while (NFI_STR==0x1) ;	//NFI_STR is happened in NFI address phase. NFI_STR is from NFI.
0 = *NFIECC_ENCIDLE ;	//It indicates the start is triggered and Encoder is in busy state.
while (*NFIECC_ENCIDLE==0x1) ;	//Wait all message data from NFI. After all data has input IDLE will be asserted.
parity = {*NFIECC_PAR0,*NFIECC_PAR1, *NFIECC_PAR2, *NFIECC_PAR3, *NFIECC_PAR4}	//If parity is necessary, Read out parity from APB register after IDLE=1.

Encoding in DMA mode

Configure	Memo
while (*NFIECC_ENCIDLE==1) ;	//polling IDLE signal until Encoder is available.
*NFIECC_ENCCNFG = 0x10400010;	//configure Encoder parameter in NFI mode.
*NFIECC_ENCIRQEN = 0x1;	//If IRQ is required when Encoder is done.
*NFIECC_ENCDIADDR= 0x10000000;	//Configure Data start address.
*NFIECC_ENCCON = 0x1 ;	//Encoder starts fetching data from ENCDIADDR.
0 = *NFIECC_ENCIDLE ;	//It indicates the start is triggered and Encoder is in busy state.
while (*NFIECC_ENCIDLE==0x1) ;	//After all data has fetched and encoded, IDLE will be asserted.
parity = {*NFIECC_PAR0,*NFIECC_PAR1, *NFIECC_PAR2, *NFIECC_PAR3, *NFIECC_PAR4}	// Read out parity from APB register after IDLE=1 and must append parity bits behind the original data for decoding.

Encoding in PIO mode

Configure	Memo
while (*NFIECC_ENCIDLE==1) ;	//polling IDLE signal until Encoder is available.
*NFIECC_ENCCNFG = 0x10400010;	//configure Encoder parameter in NFI mode.
*NFIECC_ENCIRQEN = 0x1;	//If IRQ is required when Encoder is done.
*NFIECC_ENCCON = 0x1 ;	//Encoder starts.
while (*NFIECC_PIO_DIRDY==0x1) {	//Wait the NFIECC_PIO_DIRDY is active then
*NFIECC_PIO_DI = DATA[i]; i++ }	//Input data 32bit by 32bit input NFIECC_PIO_DI APB register.
while (*NFIECC_ENCIDLE==0x1) ;	//After all data has been written in PIO_DI and encoder has been done, IDLE will be asserted.
parity = {*NFIECC_PAR0,*NFIECC_PAR1, *NFIECC_PAR2, *NFIECC_PAR3, *NFIECC_PAR4}	// Read out parity from APB register after IDLE=1 and must append parity bits behind the original data for decoding.

Decoding in NFI mode

Caution: Before NFI address phase enable and configure ECC.

Caution: When NFI AUTO_FMT_EN=0, ECC will correct all errors (include parity bits) found in Chien search. Be careful of those FDM data that was not protected by ECC. Those data would not be realized by ECC module and might be polluted by ECC module.

Caution: ECC correct limitation is error_limit = error_correct_capability. If the error number(data error number + parity error number) is bigger than the error_limit, ECC might decode error.

Configure	Memo
*NFIECC_DECCNFG = 0x90743010;	//configure Decoder parameter in NFI mode.
*NFIECC_FDMADDR = 0x800320A0;	//configure FDM0 APB address in NFI mode into FDMADDR. (NFI_BASE_ADDR+FDM0_OFFSET_ADDR)
*NFIECC_DECCON = 0x1 ;	//enable Decoder.
for i = 1:8	// 8 is equal to NFI read sector number.
while (NFI_STR==0x1) ;	//NFI_STR is happened in NFI address phase. NFI_STR is from NFI.
0 = *NFIECC_DECIDLE ;	//It indicates the start is triggered and Decoder is in busy state.
while (*NFIECC_DECIDLE==0x1) ;	//Wait all message data from NFI. After all data has input IDLE will be asserted and FER will be reported.
end for	

while (*NFIECC_DECDONE==0xff)	//Decoder and correction processor is done.
Additional Usage for detecting error number or uncorrectable error (execute after Done)	
while (*NFIECC_DECFSM==0x0)	//All Hardware is done
ERR_SEC = *NFIECC_FER	//Read Error status
for i = 0 : (SEC_NUM-1)	
if (ERR_SEC[i] == 1)	//If the sector has error
ERR_NUM = *NFIECC_DECENUM	//Read Error number
end if	
for i = 0 : (ERR_NUM-1)	//Read Error Location to check if location exceeds coded data size. If error location exceeds coded data size, it is an uncorrected decoding sector.
ErrorLocation[i] = *NFIECC_EL0+2i;	
end for	
end for	
*NFIECC_DECCON = 0x0	//Disable Decoder.

Decoding in DMA mode

Configure	**Memo**
while (*NFIECC_DECIDLE==1) ;	//polling IDLE signal until Decoder is available.
*NFIECC_DECCNFG = 0x90743010;	//configure Decoder parameter in NFI mode.
*NFIECC_DECIRQEN = 0x1;	//If IRQ is required when Encoder is done.
*NFIECC_DECDIADDR= 0x10000000;	//Configure Data start address.
*NFIECC_DECCON = 0x1 ;	//Decoder starts to fetch data from DECDIADDR.
0 = *NFIECC_DECIDLE ;	//It indicates the start is triggered and Decoder is in busy state.
while (*NFIECC_DECDONE==0x1)	//Decoder and correction processor is done.
Additional Usage for detecting error number or uncorrectable error (execute after Done)	
while (*NFIECC_DECFSM==0x0)	//All Hardware is done
ERR_SEC = *NFIECC_FER	//Read Error status
for i = 0 : (SEC_NUM-1)	//In DMA mode only support SEC_NUM=1
if (ERR_SEC[i] == 1)	//If the sector has error
ERR_NUM = *NFIECC_DECENUM	//Read Error number
end if	
for i = 0 : (ERR_NUM-1)	//Read Error Location to check if location exceeds coded data size. If error location exceeds coded data size, it is an uncorrected decoding sector.
ErrorLocation[i] = *NFIECC_EL0+2i;	
end for	
end for	
*NFIECC_DECCON = 0x0	//Disable Decoder.

Decoding in PIO mode

Configure	**Memo**
while (*NFIECC_DECIDLE==1) ;	//polling IDLE signal until Decoder is available.
*NFIECC_DECCNFG = 0x90743010;	//configure Decoder parameter in NFI mode.
*NFIECC_DECIRQEN = 0x1;	//If IRQ is required when Encoder is done.
*NFIECC_DECCON = 0x1 ;	//Decoder starts.
while (*NFIECC_PIO_DIRDY==0x1) {	//Wait the NFIECC_PIO_DIRDY is active then
*NFIECC_PIO_DI = DATA[i]; i++ }	//Input data 32bit by 32bit input NFIECC_PIO_DI APB register.
while (*NFIECC_DECDONE==0x1)	//Decoder and correction processor is done.
Additional Usage for detecting error number or uncorrectable error (execute after Done)	

while (*NFIECC_DECFSM==0x0)	//All Hardware is done
ERR_SEC = *NFIECC_FER	//Read Error status
for i = 0 : (SEC_NUM-1)	//In PIO mode only support SEC_NUM=1
if (ERR_SEC[i] == 1)	//If the sector has error
ERR_NUM = *NFIECC_DECENUM	//Read Error number
end if	
for i = 0 : (ERR_NUM-1)	//Read Error Location to check if location exceeds coded data size. If error location exceeds coded data size, it is an uncorrected decoding sector.
ErrorLocation[i] = *NFIECC_EL0+2i;	
end for	
end for	
*NFIECC_DECCON = 0x0	//Disable Decoder.

2.10 PCM Controller

2.10.1 Features

- Two clock sources are reserved for PCM circuit. (From internal clock generator, INT_PCM_CLK and EXT_PCM_CLK)
- PCM module can drive a clock out (with fraction-N dividor) to an external codec.
- Up to 4 channels PCM are available. 4 to 128 slots are configurable.
- Each channel supports a-law (8-bit)/u-law (8-bit)/raw-PCM (8-bit and 16-bit) transfer.
- Hardware converter of a-law<->raw-16 and u-law <-> raw-16 are implemented in design.
- Support long (8 cycle)/short (1 cycle)/configurable (intervals are configurable, use to emulate I^2S interface) FSYNC.
- DATA & FSYNC can be driven and sampled by either rising/falling of clock.
- Last bit of DTX can be configured as tri-stated on falling edge.
- Beginning of each slot is configurable by 10-bit registers on each channel.
- 32-byte FIFO are available for each channel
- PCM interface can emulate I2S interface (only 16-bit data-width supported).
- MSB/LSB order is configurable.
- Supports both a-law/u-law (8-bits) →linear PCM(16-bit) and linear PCM(16-bit) → a-law/u-law (8-bit)

2.10.2 Block Diagram

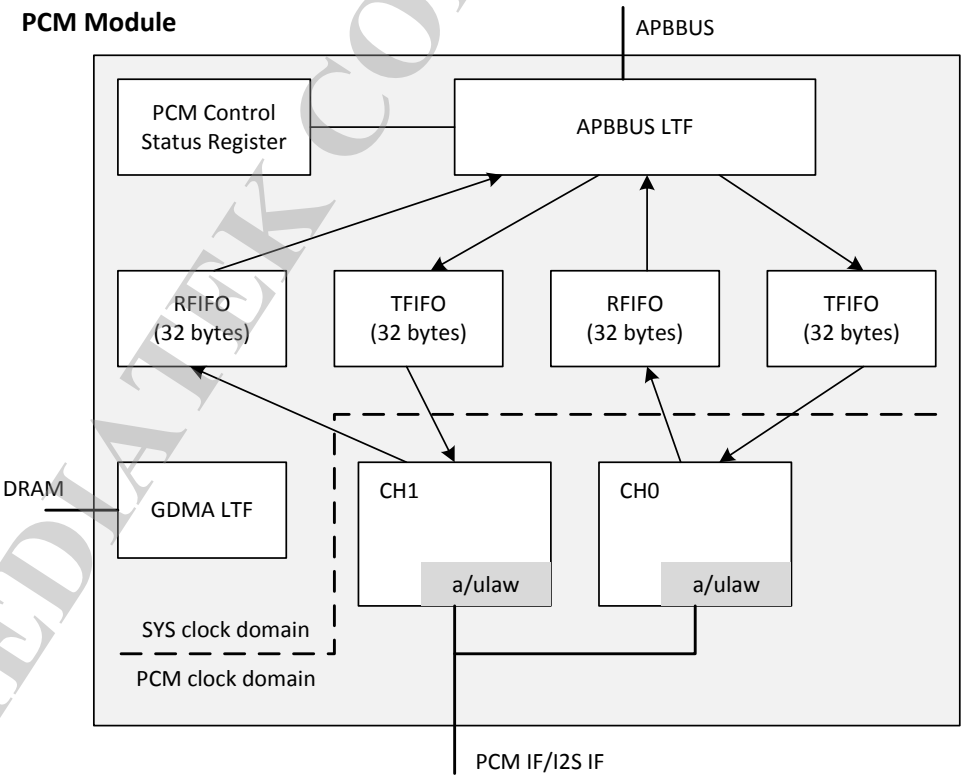


Figure 2-4 PCM Controller Block Diagram

Two clock domains are partitioned in this design. PCM converter (u-law < = > raw-16-bit and A-law < = > raw 16-bit) are implemented in PCM. The threshold of FIFO is configurable. When the threshold is reached, PCM (a) triggers the DMA interface to notify external DMA engine to transfer data, and (b) triggers an interrupt to the host.

The interrupt sources include:

- The threshold is reached.
- FIFO is under-run or over-run.
- A fault is detected at the DMA interface.

The A-law and u-law converter is implemented based on the ITU-G.711 A-law and u-law table. In this design, both A-law/u-law(8-bit) → linear PCM (16-bit) and linear PCM (16-bit) → A-law/u-law (8-bit) are supported.

The data-flow from codec to PCM-controller (Rx-flow) is shown as below:

- The PCM controller latches the data from DRX at the indicated time slot and then writes it to FIFO. If FIFO is full, the data is lost.
- When the Rx-FIFO reaches the threshold, two actions may be taken:
  - When DMA_ENA=1, DMA_REQ is asserted to request a burst transfer. It rechecks the FIFO threshold after DMA_END is asserted by GDMA. (GDMA should be configured before channel is enabled.)
  - Assert the interrupt source to notify the host. The host can check RFIFO_AVAIL information then get back the data from FIFO.

The data flow from the PCM controller to codec (Tx-flow) is shown below. After GDMA is configured, software should configure and enable the PCM channel. The empty FIFO should behave as follows.

- When DMA_ENA=1, DMA_REQ is triggered to request a burst transfer. It then re-checks the FIFO threshold after DMA_END is asserted by GDMA (a burst is completed).
- The Interrupt source is asserted to notify HOST. HOST writes the data to Tx-FIFO. After that, HOST rechecks TFIFO_EMPTY information, and then writes more data if available.

NOTE: When DMA_ENA=1, the burst size of GDMA should be less than the threshold value.

2.10.3 List of Registers

PCM Changes LOG

Revision	Date	Author	Change Log
0.1	2012/10/8	Paddy Wu	Initialization

Module name: PCM Base address: (+1E002000h)

Address	Name	Width	Register Function
1E002000	GLB_CFG	32	Global Config
1E002004	PCM_CFG	32	PCM configuration
1E002008	INT_STATUS	32	Interrupt status
1E00200C	INT_EN	32	Interrupt enable
1E002010	CHA0_FF_STATUS	32	Channel A0(represents channel 0) FIFO status
1E002014	CHB0_FF_STATUS	32	Channel B0(represents channel 1) FIFO status
1E002020	CHA0_CFG	32	Channel A0(represents channel 0) Config
1E002024	CHB0_CFG	32	Channel B0(represents channel 1) Config
1E002030	FSYNC_CFG	32	FSYNC config
1E002034	CHA0_CFG2	32	Channel A0(represents channel 0) Config
1E002038	CHB0_CFG2	32	Channel B0(represents channel 1) Config
1E002040	IP_INFO	32	IP version info
1E002044	RSV_REG16	32	SPARE REG 16 bits
1E002050	DIVCOMP_CFG	32	Dividor Compensation part config
1E002054	DIVINT_CFG	32	Dividor Integer part config
1E002060	DIGDELAY_CFG	32	Digital delay config
1E002080	CH0_FIFO	32	Channel 0 FIFO access point
1E002084	CH1_FIFO	32	Channel 1 FIFO access point
1E002088	CH2_FIFO	32	Channel 2 FIFO access point
1E00208C	CH3_FIFO	32	Channel 3 FIFO access point
1E002110	CHA0_FF_STATUS	32	Channel A1(represents channel 3) FIFO status
1E002114	CHB1_FF_STATUS	32	Channel B1(represents channel 4) FIFO status
1E002120	CHA1_CFG	32	Channel A1(represents channel 3) Config
1E002124	CHB1_CFG	32	Channel B1(represents channel 1) Config
1E002134	CHA1_CFG2	32	Channel A1(represents channel 3) Config
1E002138	CHB1_CFG2	32	Channel B1(represents channel 4) Config

1E002000 GLB_CFG Global Config 00440000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PCM_	DMA_E	LBK_E	EXT_LB	RSV0					RFF_THRES			RSV1	TFF_THRES		

	EN	N	N	K_EN												
Type	RW	RW	RW	RW	RO					RW			RO	RW		
Reset	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV2												CH_EN			
Type	RO												RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	PCM_EN	**PCM Enable** When disabled, all FSM of PCM are cleared to their default value. 0: disable 1: enable
30	DMA_EN	**DMA Enable** 0: Disable the DMA interface, transfer data using software. 1: Enable the DMA interface, transfer data using DMA. 0: disable 1: enable
29	LBK_EN	**loopback enable, loopback path is shown as (Asyn-TXFIFO ->DTX -> DRX->Asyn-RXFIFO)** 0: disable 1: enable
28	EXT_LBK_EN	**loopback enable, loopback path is shown as (Ext-Codec->DRX->DTX->Ext-Codec)** 0: disable 1: enable
27:23	RSV0	**Reserved**
22:20	RFF_THRES	**RXFIFO Threshold** When the threshold is reached, the host/DMA is notified to fill FIFO. The threshold should be >2 and <6. When data in FIFO is under the threshold, the following interrupts and GDMA are triggered. CH0T_THRES, CH0R_THRES, CH1T_THRES, CH1R_THRES (unit: word)
19	RSV1	**Reserved**
18:16	TFF_THRES	**TXFIFO Threshold** When the threshold is reached, the host/DMA is notified to fill FIFO. It should be >2 and <6. When data in FIFO is over the threshold, an interrupt and DMA are triggered. (unit: word)
15:4	RSV2	**Reserved**
3:0	CH_EN	**Channels 3 to 0 Tx and Rx Enable** 0: disable 1: enable

1E002004 PCM_CFG PCM configuration 03000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0	CLKOUT_EN	RSV1		EXT_FSYNC	LONG_SYNC	FSYNC_POL	DTX_TRI	RSV2[20:13]							
Type	RO	RW	RO		RW	RW	RW	RW	RO							
Reset	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name	RSV2[12:0]													SLOT_MODE		
Type	RO													RW		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	RSV0	**Reserved**
30	CLKOUT_EN	**PCM Clock Out Enable** 0: A PCM clock is provided from the external Codec/OSC. 1: A PCM clock is provided from the internal dividor. NOTE: Normally, the register should be asserted to 1. Also, it should be asserted after configuring the divider and enabling the divider clock. 0: EXT_CLK 1: INT_DIV
29:28	RSV1	**Reserved**
27	EXT_FSYNC	**FSYNC is provided externally** 0: FSYNC is generated by internal circuit. 1: FSYNC is provided externally
26	LONG_SYNC	**FSYNC Mode** 0: Short FSYNC 1: Long FSYNC
25	FSYNC_POL	**FSYNC Polarity** 0: FSYNC is low active 1: FSYNC is high active
24	DTX_TRI	**DTX Tri-State** Tristates DTX when the clock signal on the last bit is has a falling edge. 0: Non- tristate DTX 1: Tristate DTX
23:3	RSV2	**Reserved**
2:0	SLOT_MODE	**Sets the number of slots in each PCM frame.** 0: 4 slots, PCM clock out/in should be 256 KHz. 1: 8 slots, PCM clock out/in should be 512 KHz. 2: 16 slots, PCM clock out/in should be 1.024 MHz. 3: 32 slots, PCM clock out/in should be 2.048 MHz. 4: 64 slots, PCM clock out/in should be 4.096 MHz. 5:128 slots, PCM clock out/in should be 8.192 MHz. Other: Reserved. NOTE: When using the external clock, the frequency clock should be equal to PCM_clock out. Otherwise, the PCM_CLKin should be 8.192 MHz. 0: _4_SLOT 1: _8_SLOT 2: _16_SLOT 3: _32_SLOT 4: _64_SLOT 5: _128_SLOT

1E002008 INT_STATUS Interrupt status 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0[23:8]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV0[7:0]								CHT_DMA_FAULT	CHT_OVRUN	CHT_UNRUN	CHT_THRES	CHR_DMA_FAU	CHR_OVRUN	CHR_UNRUN	CHR_THRES

													LT			
Type	RO								W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:8	RSV0	**Reserved**
7	CHT_DMA_FAULT	**Channel Tx DMA Fault Interrupt, Asserts when a fault has been detected in a CH-Tx DMA signal.**
6	CHT_OVRUN	**Channel Tx FIFO Overrun Interrupt, Asserts when the CH-Tx FIFO is overrun.**
5	CHT_UNRUN	**Channel Tx FIFO Underrun Interrupt, Asserts when the CH-Tx FIFO is underrun.**
4	CHT_THRES	**Channel Tx Threshold Interrupt, Asserts when the CH-Tx FIFO is lower than the defined threshold.**
3	CHR_DMA_FAULT	**Channel Rx DMA Fault Interrupt, Asserts when a fault is detected in a CH-Rx DMA signal.**
2	CHR_OVRUN	**Channel Rx Overrun Interrupt, Asserts when the CH-Rx FIFO is overrun.**
1	CHR_UNRUN	**Channel Rx Underrun Interrupt, Asserts when the CH-Rx FIFO is underrun.**
0	CHR_THRES	**Channel Rx Threshold Interrupt, Asserts when the CH-Rx FIFO is lower than the defined threshold.**

1E00200C INT_EN Interrupt enable 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0[23:8]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV0[7:0]								INT7_EN	INT6_EN	INT5_EN	INT4_EN	INT3_EN	INT2_EN	INT1_EN	INT0_EN
Type	RO								RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:8	RSV0	**Reserved**
7	INT7_EN	**INT_STATUS[7] Enable,Enables the Channel Tx DMA Fault Interrupt. This interrupt asserts when a fault has been detected in a CH-Tx DMA signal.**
6	INT6_EN	**INT_STATUS[6] Enable,Enables the Channel Tx FIFO Overrun Interrupt. This interrupt asserts when the CH-Tx FIFO is overrun.**
5	INT5_EN	**INT_STATUS[5] Enable,Enables the Channel Tx FIFO Underrun Interrupt. This interrupt asserts when the CH-Tx FIFO is underrun.**
4	INT4_EN	**INT_STATUS[4] Enable,Enables the Channel Tx Threshold Interrupt. This interrupt when the CH-Tx FIFO is lower than the defined threshold.**
3	INT3_EN	**INT_STATUS[3] Enable,Enables the Channel Rx DMA Fault Interrupt. This interrupt when a fault is detected in a CH-Rx DMA signal.**
2	INT2_EN	**INT_STATUS[2] Enable,Enables the Channel Rx Overrun Interrupt. This interrupt when the CH-Rx FIFO is overrun.**
1	INT1_EN	**INT_STATUS[1] Enable,Enables the Channel Rx Underrun Interrupt. This interrupt when the CH-Rx FIFO is under-run.**
0	INT0_EN	**INT_STATUS[0] Enable,Enables the Channel Rx Threshold Interrupt. This interrupt asserts when the CH-Rx FIFO is lower than the defined threshold.**

1E002010 CHA0_FF_STATUS Channel A0(represents channel 0) FIFO status 00100008

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0								CHTX_DMA_FAULT	CHTX_OVRUN	CHTX_UNRUN	CHTX_THRES	CHRX_DMA_FAULT	CHRX_OVRUN	CHRX_UNRUN	CHRX_THRES
Type	RO								W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV1								CHRFF_AVCNT				CHTFF_EPCNT			
Type	RO								RO				RO			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

Bit(s)	Name	Description
31:24	RSV0	**Reserved**
23	CHTX_DMA_FAULT	**Tx DMA Fault Detected Interrupt,Asserts when a fault is detected in a Channel A0 Tx DMA signal.**
22	CHTX_OVRUN	**Tx Overrun Interrupt,Asserts when the Channel A0 Tx FIFO is overrun.**
21	CHTX_UNRUN	**Tx FIFO Underrun Interrupt,Asserts when the Channel A0 Tx FIFO is underrun.**
20	CHTX_THRES	**Tx FIFO Below Threshold Interrupt,Asserts when the Channel A0 FIFO is lower than the defined threshold.**
19	CHRX_DMA_FAULT	**Rx DMA Fault Detected Interrupt,Asserts when a fault is detected in a Channel A0 Rx DMA signal.**
18	CHRX_OVRUN	**Rx FIFO Overrun Interrupt,Asserts when the Channel A0 Rx FIFO is overrun.**
17	CHRX_UNRUN	**Rx FIFO Underrun Interrupt,Asserts when the Channel A0 Rx FIFO is underrun.**
16	CHRX_THRES	**Rx FIFO Below Threshold Interrupt,Asserts when the Channel A0 FIFO is lower than the defined threshold.**
15:8	RSV1	**Reserved**
7:4	CHRFF_AVCNT	**Channel A0 RXFIFO Available Space Count,Counts the available space for reads in channel A0 RXFIFO.(unit: word)**
3:0	CHTFF_EPCNT	**Channel A0 TXFIFO Available Space Count,Counts the available space for writes in channel A0 TXFIFO.(unit: word)**

1E002014 CHB0_FF_STATUS Channel B0(represents channel 1) FIFO status 00100008

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0								CHTX_DMA_FAULT	CHTX_OVRUN	CHTX_UNRUN	CHTX_THRES	CHRX_DMA_FAULT	CHRX_OVRUN	CHRX_UNRUN	CHRX_THRES
Type	RO								W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV1								CHRFF_AVCNT				CHTFF_EPCNT			
Type	RO								RO				RO			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

Bit(s)	Name	Description
31:24	RSV0	**Reserved**
23	CHTX_DMA_FAULT	**Tx DMA Fault Detected Interrupt,Asserts when a fault is detected in a Channel B0 Tx DMA signal.**
22	CHTX_OVRUN	**Tx Overrun Interrupt,Asserts when the Channel B0 Tx FIFO is overrun.**
21	CHTX_UNRUN	**Tx FIFO Underrun Interrupt,Asserts when the Channel B0 Tx FIFO is underrun.**
20	CHTX_THRES	**Tx FIFO Below Threshold Interrupt,Asserts when the Channel B0 FIFO is lower than the defined threshold.**
19	CHRX_DMA_FAULT	**Rx DMA Fault Detected Interrupt,Asserts when a fault is detected in a Channel B0 Rx DMA signal.**
18	CHRX_OVRUN	**Rx FIFO Overrun Interrupt,Asserts when the Channel B0 Rx FIFO is overrun.**
17	CHRX_UNRUN	**Rx FIFO Underrun Interrupt,Asserts when the Channel B0 Rx FIFO is underrun.**
16	CHRX_THRES	**Rx FIFO Below Threshold Interrupt,Asserts when the Channel B0 FIFO is lower than the defined threshold.**
15:8	RSV1	**Reserved**
7:4	CHRFF_AVCNT	**Channel B0 RXFIFO Available Space Count,Counts the available space for reads in channel B0 RXFIFO.(unit: word)**
3:0	CHTFF_EPCNT	**Channel B0 TXFIFO Available Space Count,Counts the available space for writes in channel B0 TXFIFO.(unit: word)**

1E002020 CHA0_CFG Channel A0(represents channel 0) Config 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0		CMP_MODE			RSV1[16:6]										
Type	RO		RW			RO										
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV1[5:0]						TS_START									
Type	RO						RW									
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
31:30	RSV0	**Reserved**
29:27	CMP_MODE	**Compression Mode** Sets the conversion method for the hardware converter to compress raw data. 000: Disable HW converter, linear raw data (16-bit) 010: Disable HW converter, linear raw data (8-bit), A-law or u-law (8-bit) 011: Reserved 100: Enable HW converter, raw data(16-bit) ? U-law mode (8-bit) (PCM bus in compressed format) 101: Enable HW converter, u-law mode (8-bit) ? raw data (16-bit) (PCM bus in raw, 16-bit format) 110: Enable HW converter, raw data (16-bit) ? A-law mode (8-bit) (PCM bus in compressed format) 111: Enable HW converter, A-law mode (8-bit) ? raw data (16-bit) (PCM bus in raw, 16-bit format) 0: DIS_CONV16 2: DIS_CONV8 4: EN_ULW2R 5: EN_R2ULW 6: EN_ALW2R 7: EN_R2ALW
26:10	RSV1	**Reserved**
9:0	TS_START	**Timeslot starting location** (unit: clock cycles)

1E002024 CHB0_CFG Channel B0(represents channel 1) Config 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0		CMP_MODE			RSV1[16:6]										
Type	RO		RW			RO										
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV1[5:0]						TS_START									
Type	RO						RW									
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
31:30	RSV0	**Reserved**
29:27	CMP_MODE	**Compression Mode** Sets the conversion method for the hardware converter to compress raw data. 000: Disable HW converter, linear raw data (16-bit) 010: Disable HW converter, linear raw data (8-bit), A-law or u-law (8-bit) 011: Reserved 100: Enable HW converter, raw data(16-bit) ? U-law mode (8-bit) (PCM bus in compressed format) 101: Enable HW converter, u-law mode (8-bit) ? raw data (16-bit) (PCM bus in raw, 16-bit format) 110: Enable HW converter, raw data (16-bit) ? A-law mode (8-bit) (PCM bus in compressed format) 111: Enable HW converter, A-law mode (8-bit) ? raw data (16-bit) (PCM bus in raw, 16-bit format) 0: DIS_CONV16 2: DIS_CONV8 4: EN_ULW2R 5: EN_R2ULW 6: EN_ALW2R 7: EN_R2ALW
26:10	RSV1	**Reserved**
9:0	TS_START	**Timeslot starting location** (unit: clock cycles)

1E002030 FSYNC_CFG FSYNC config 28000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CFG_FSYNC_EN	POS_CAP_DT	POS_DRV_DT	POS_CAP_FSYNC	POS_DRV_FSYNC	RSV0					RSV1[11:6]					
Type	RW	RW	RW	RW	RW	RO					RO					
Reset	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV1[5:0]						FSYNC_INTV									
Type	RO						RW									
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	CFG_FSYNC_EN	**Enables configurable FSYNC.**
30	POS_CAP_DT	**Positive Edge Capture Data, Sets the PCM controller to capture data on the**

		negative or positive edge of the PCM clock. NOTE: This configuration should be 0 if DTX_TRI=1.
29	POS_DRV_DT	**Positive Edge Drive Data, Sets the PCM controller to drive data on the negative or positive edge of the PCM clock.**
28	POS_CAP_FSYNC	**Positive Edge Capture FSYNC, Sets the PCM controller to capture FSYNC on the positive or negative edge of the PCM clock.**
27	POS_DRV_FSYNC	**Positive Edge Driver FSYNC, Sets the PCM controller to drive FSYNC on the negative or positive edge of the PCM clock.**
26:22	RSV0	**Reserved**
21:10	RSV1	**Reserved**
9:0	FSYNC_INTV	**Interval when FSYNC may be configured.** (unit: clock cycles)

1E002034 CHA0_CFG2 Channel A0(represents channel 0) Config 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0[27:12]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV0[11:0]												CH_RXFF_CLR	CH_TXFF_CLR	RSV1	CH_LSB
Type	RO												RW	RW	RO	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:4	RSV0	**Reserved**
3	CH_RXFF_CLR	**Channel A0 Rx FIFO Clear** 0: Normal operation 1: Clear this bit
2	CH_TXFF_CLR	**Channel A0 Tx FIFO Clear** 0: Normal operation 1: Clear this bit
1	RSV1	**Reserved**
0	CH_LSB	**Enable CH A0 Tx in LSB order.**

1E002038 CHB0_CFG2 Channel B0(represents channel 1) Config 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0[27:12]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV0[11:0]												CH_RXFF_CLR	CH_TXFF_CLR	RSV1	CH_LSB
Type	RO												RW	RW	RO	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:4	RSV0	**Reserved**
3	CH_RXFF_CLR	**Channel B0 Rx FIFO Clear** 0: Normal operation 1: Clear this bit
2	CH_TXFF_CLR	**Channel B0 Tx FIFO Clear** 0: Normal operation 1: Clear this bit
1	RSV1	**Reserved**
0	CH_LSB	**Enable CH B0 Tx in LSB order.**

1E002040 IP_INFO IP version info 00000401

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_CH								VER							
Type	RO								RO							
Reset	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
31:16	RSV0	**Reserved**
15:8	MAX_CH	**Maximum channel number.**
7:0	VER	**Version of this PCM Controller**

1E002044 RSV_REG16 SPARE REG 16 bits 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SPARE_REG															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RSV0	**Reserved**
15:0	SPARE_REG	**Spare register for future use**

1E002050 DIVCOMP_CFG Dividor Compensation part config 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CLK_E	RSV0[22:8]														

	N															
Type	RW	RO														
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV0[7:0]								DIVCOMP							
Type	RO								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	CLK_EN	**Clock Enable** Enables setting of the PCM interface clock based on DIVCOMP and DIVINT parameters.
30:8	RSV0	**Reserved**
7:0	DIVCOMP	**A parameter in an equation which determines FREQOUT. See DIVINT.**

1E002054 DIVINT_CFG Dividor Integer part config 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0[21:6]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV0[5:0]						DIVINT									
Type	RO						RW									
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:10	RSV0	**Reserved**
9:0	DIVINT	**A parameter in an equation which determines FREQOUT.** Formula: FREQOUT = 1/(FREQIN*2*(DIVINT+DIVCOMP /(2^8))) FREQIN is always fixed to 40 MHz.

1E002060 DIGDELAY_CFG Digital delay config 00000002

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TXD_CLR_GLT	CHEN_CLR_GLT	RSV0			TXD_GLT_ST	RSV1			CHEN_GLT_ST	RSV2			CHENP_GLT_ST	RSV3	CHENPD_GLT_ST
Type	RW	RW	RO			RW	RO			RW	RO			RW	RO	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TXD_DIGDLY_EN	RSV4		TXD_DLYVAL					CHEN_DIGDLY_EN	RSV5		CHEN_DLYVAL				
Type	RW	RO		RW					RW	RO		RW				
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bit(s)	Name	Description
31	TXD_CLR_GLT	**TXD Clear Glitch Flag** Clears the glitch detected flag for TXD. 0: No effect. 1: Clear the flag.
30	CHEN_CLR_GLT	**Channel Enable (CHEN) Clear Glitch Flag** Clears the glitch detected flag for CHEN. 0: No effect . 1: Clear the flag.
29:27	RSV0	**Reserved**
26	TXD_GLT_ST	**TXD Glitch Status** Indicates if a glitch is detected in a TXD signal. It can be cleared by bit[31]. 0: Not detected. 1: Detected
25:23	RSV1	**Reserved**
22	CHENN_GLT_ST	**CHEN Negative Glitch Status** Indicates if a glitch is detected in a CHEN signal. It can be cleared by bit[30] (negedge sample). 0: Not detected. 1: Detected
21:19	RSV2	**Reserved**
18	CHENP_GLT_ST	**CHEN Positive Glitch Status** Indicates if a glitch is detected in a CHEN signal. It can be cleared by bit[30] (posedge sample). 0: Not detected. 1: Detected
17	RSV3	**Reserved**
16	CHENPD_GLT_ST	**CHEN Positive Delay Glitch Status** Indicates if a glitch is detected in a CHEN signal. It can be cleared by bit[30] (posedge sample, delay 1 cycle). 0: Not detected. 1: Detected
15	TXD_DIGDLY_EN	**TXD Digital Delay Enable** Enables digital delay path. 0: Disable 1: Enable
14:13	RSV4	**Reserved**
12:8	TXD_DLYVAL	**Delay Count Value** The description is the same as the CHEN_DLYVAL field in this register. CHEN Digital Delay Enable, Enables the digital delay path. 0: Disable 1: Enable
7	CHEN_DIGDLY_EN	**CHEN Digital Delay Enable** Enables the digital delay path. 0: Disable 1: Enable
6:5	RSV5	**Reserved**
4:0	CHEN_DLYVAL	**Delay Count Value** The delay error = CLK_PERIOD * (SYNC_DELAY + SYNC_DELTA + (DLYCNT_CFG) + 1) For example, DLYCNT_CFG = 4, (SYNC_DELAY is always fixed to 4) Final Delay = CLK_PERIOD * (2 + (-1/0/+1) + (4) + 1) = CLK_PERIOD * (6/7/8)= CLK_PERIOD * (6 to 8) = 25 ns to 33.3 ns NOTE:

Period is 1/240 MHz = 4.1667 ns in MT7620.

1E002080 CH0_FIFO Channel 0 FIFO access point 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CH0_FIFO[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CH0_FIFO[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	CH0_FIFO	**Channel 0 FIFO access point**

1E002084 CH1_FIFO Channel 1 FIFO access point 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CH1_FIFO[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CH1_FIFO[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	CH1_FIFO	**Channel 1 FIFO access point**

1E002088 CH2_FIFO Channel 2 FIFO access point 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CH2_FIFO[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CH2_FIFO[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	CH2_FIFO	**Channel 2 FIFO access point**

1E00208C CH3_FIFO Channel 3 FIFO access point 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Name	CH3_FIFO[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CH3_FIFO[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	CH3_FIFO	**Channel 3 FIFO access point**

1E002110 CHA0_FF_STATUS Channel A1(represents channel 3) FIFO status 00100008

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0								CHTX_DMA_FAULT	CHTX_OVRUN	CHTX_UNRUN	CHTX_THRES	CHRX_DMA_FAULT	CHRX_OVRUN	CHRX_UNRUN	CHRX_THRES
Type	RO								W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV1								CHRFF_AVCNT				CHTFF_EPCNT			
Type	RO								RO				RO			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

Bit(s)	Name	Description
31:24	RSV0	**Reserved**
23	CHTX_DMA_FAULT	**Tx DMA Fault Detected Interrupt,Asserts when a fault is detected in a Channel A1 Tx DMA signal.**
22	CHTX_OVRUN	**Tx Overrun Interrupt,Asserts when the Channel A0 Tx FIFO is overrun.**
21	CHTX_UNRUN	**Tx FIFO Underrun Interrupt,Asserts when the Channel A1 Tx FIFO is underrun.**
20	CHTX_THRES	**Tx FIFO Below Threshold Interrupt,Asserts when the Channel A0 FIFO is lower than the defined threshold.**
19	CHRX_DMA_FAULT	**Rx DMA Fault Detected Interrupt,Asserts when a fault is detected in a Channel A1 Rx DMA signal.**
18	CHRX_OVRUN	**Rx FIFO Overrun Interrupt,Asserts when the Channel A1 Rx FIFO is overrun.**
17	CHRX_UNRUN	**Rx FIFO Underrun Interrupt,Asserts when the Channel A1 Rx FIFO is underrun.**
16	CHRX_THRES	**Rx FIFO Below Threshold Interrupt,Asserts when the Channel A1 FIFO is lower than the defined threshold.**
15:8	RSV1	**Reserved**
7:4	CHRFF_AVCNT	**Channel A1 RXFIFO Available Space Count,Counts the available space for reads in channel A1 RXFIFO.(unit: word)**
3:0	CHTFF_EPCNT	**Channel A1 TXFIFO Available Space Count,Counts the available space for writes in channel A1 TXFIFO.(unit: word)**

1E002114 CHB1_FF_STATUS Channel B1(represents channel 4) FIFO status 00100008

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0								CH	CH	CH	CH	CH	CH	CH	CH

									TX_DMA_FAULT	TX_OVRUN	TX_UNRUN	TX_THRES	RX_DMA_FAULT	RX_OVRUN	RX_UNRUN	RX_THRES
Type	RO								W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV1								CHRFF_AVCNT				CHTFF_EPCNT			
Type	RO								RO				RO			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

Bit(s)	Name	Description
31:24	RSV0	**Reserved**
23	CHTX_DMA_FAULT	**Tx DMA Fault Detected Interrupt,Asserts when a fault is detected in a Channel B1 Tx DMA signal.**
22	CHTX_OVRUN	**Tx Overrun Interrupt,Asserts when the Channel B0 Tx FIFO is overrun.**
21	CHTX_UNRUN	**Tx FIFO Underrun Interrupt,Asserts when the Channel B1 Tx FIFO is underrun.**
20	CHTX_THRES	**Tx FIFO Below Threshold Interrupt,Asserts when the Channel B1 FIFO is lower than the defined threshold.**
19	CHRX_DMA_FAULT	**Rx DMA Fault Detected Interrupt,Asserts when a fault is detected in a Channel B1 Rx DMA signal.**
18	CHRX_OVRUN	**Rx FIFO Overrun Interrupt,Asserts when the Channel B1 Rx FIFO is overrun.**
17	CHRX_UNRUN	**Rx FIFO Underrun Interrupt,Asserts when the Channel B1 Rx FIFO is underrun.**
16	CHRX_THRES	**Rx FIFO Below Threshold Interrupt,Asserts when the Channel B1 FIFO is lower than the defined threshold.**
15:8	RSV1	**Reserved**
7:4	CHRFF_AVCNT	**Channel B1 RXFIFO Available Space Count,Counts the available space for reads in channel B1 RXFIFO.(unit: word)**
3:0	CHTFF_EPCNT	**Channel B1 TXFIFO Available Space Count,Counts the available space for writes in channel B1 TXFIFO.(unit: word)**

1E002120 CHA1_CFG Channel A1(represents channel 3) Config 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0		CMP_MODE			RSV1[16:6]										
Type	RO		RW			RO										
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV1[5:0]						TS_START									
Type	RO						RW									
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
31:30	RSV0	**Reserved**
29:27	CMP_MODE	**Compression Mode** Sets the conversion method for the hardware converter to compress raw data. 000: Disable HW converter, linear raw data (16-bit) 010: Disable HW converter, linear raw data (8-bit), A-law or u-law (8-bit) 011: Reserved 100: Enable HW converter, raw data(16-bit) ? U-law mode (8-bit) (PCM bus in compressed format) 101: Enable HW converter, u-law mode (8-bit) ? raw data (16-bit) (PCM bus in raw, 16-bit format)

		110: Enable HW converter, raw data (16-bit) ? A-law mode (8-bit) (PCM bus in compressed format) 111: Enable HW converter, A-law mode (8-bit) ? raw data (16-bit) (PCM bus in raw, 16-bit format) 0: DIS_CONV16 2: DIS_CONV8 4: EN_ULW2R 5: EN_R2ULW 6: EN_ALW2R 7: EN_R2ALW
26:10	RSV1	**Reserved**
9:0	TS_START	**Timeslot starting location** (unit: clock cycles)

1E002124 CHB1_CFG Channel B1(represents channel 1) Config 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0		CMP_MODE			RSV1[16:6]										
Type	RO		RW			RO										
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV1[5:0]						TS_START									
Type	RO						RW									
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
31:30	RSV0	**Reserved**
29:27	CMP_MODE	**Compression Mode** Sets the conversion method for the hardware converter to compress raw data. 000: Disable HW converter, linear raw data (16-bit) 010: Disable HW converter, linear raw data (8-bit), A-law or u-law (8-bit) 011: Reserved 100: Enable HW converter, raw data(16-bit) ? U-law mode (8-bit) (PCM bus in compressed format) 101: Enable HW converter, u-law mode (8-bit) ? raw data (16-bit) (PCM bus in raw, 16-bit format) 110: Enable HW converter, raw data (16-bit) ? A-law mode (8-bit) (PCM bus in compressed format) 111: Enable HW converter, A-law mode (8-bit) ? raw data (16-bit) (PCM bus in raw, 16-bit format) 0: DIS_CONV16 2: DIS_CONV8 4: EN_ULW2R 5: EN_R2ULW 6: EN_ALW2R 7: EN_R2ALW
26:10	RSV1	**Reserved**
9:0	TS_START	**Timeslot starting location** (unit: clock cycles)

1E002134 CHA1_CFG2 Channel A1(represents channel 3) Config 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0[27:12]															
Type	RO															

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV0[11:0]												CH_RXFF_CLR	CH_TXFF_CLR	RSV1	CH_LSB
Type	RO												RW	RW	RO	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:4	RSV0	**Reserved**
3	CH_RXFF_CLR	**Channel A1 Rx FIFO Clear** 0: Normal operation 1: Clear this bit
2	CH_TXFF_CLR	**Channel A1 Tx FIFO Clear** 0: Normal operation 1: Clear this bit
1	RSV1	**Reserved**
0	CH_LSB	**Enable CH A1 Tx in LSB order.**

1E002138 CHB1_CFG2 Channel B1(represents channel 4) Config 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RSV0[27:12]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RSV0[11:0]												CH_RXFF_CLR	CH_TXFF_CLR	RSV1	CH_LSB
Type	RO												RW	RW	RO	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:4	RSV0	**Reserved**
3	CH_RXFF_CLR	**Channel B1 Rx FIFO Clear** 0: Normal operation 1: Clear this bit
2	CH_TXFF_CLR	**Channel B1 Tx FIFO Clear** 0: Normal operation 1: Clear this bit
1	RSV1	**Reserved**
0	CH_LSB	**Enable CH B1 Tx in LSB order.**

2.10.4 PCM Configuration

PCM Initialization Flow

1. Set PCM_CFG
2. Set CH0/1_CFG
3. Write PCM data to FIFO CH0/1_FIFO
4. Set GLB_CFG to enable the PCM and channel.
5. Set dividor clock
6. Enable clock
7. Monitor FF_STATUS to receive/transmit the other PCM data.

PCM Configuration Examples
Below are some examples of PCM configuration.

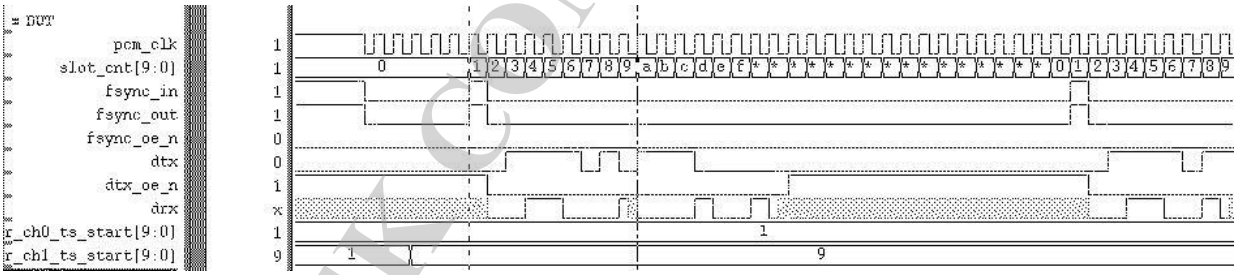
Case 1:

CFG_FSYNC Register: CFG_FSYNC_EN = 0 (PS: fsync is always driven at SLOT_CNT=1)

CH0_CFG Register: TS_START=1

CH1_CFG Register: TS_START=9

PCM_CFG Register: LONG_FSYNC=1'b0, FSYNC_POL=1'b1, DRX_TRI=1'b0, SLOT_MODE=3'b0



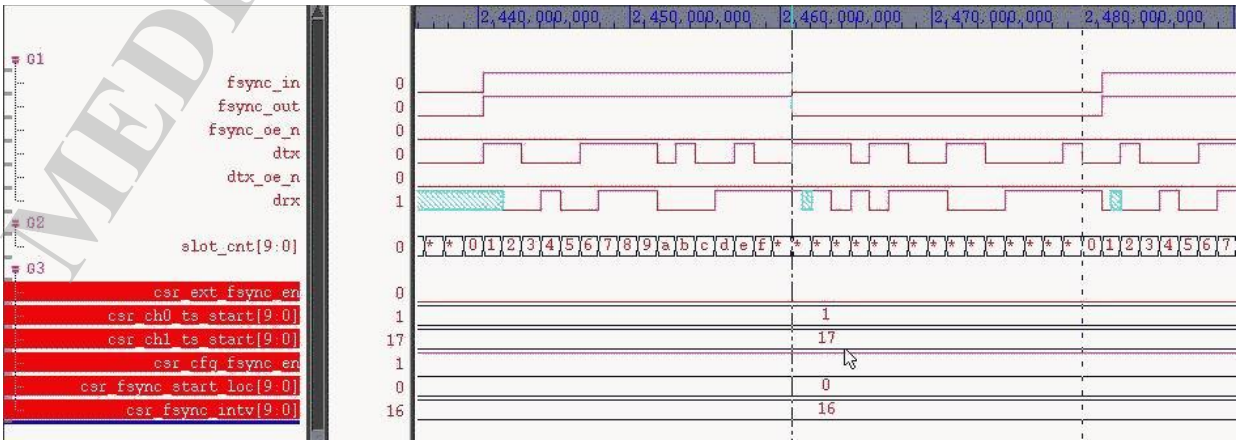
Case 2:

CFG_FSYNC Register: CFG_FSYNC_EN = 1, START_LOC=0, interval=16

CH0_CFG Register: TS_START=1

CH1_CFG Register: TS_START=17

PCM_CFG Register: LONG_FSYNC=1'b0, FSYNC_POL=1'b1, DRX_TRI=1'b0, SLOT_MODE=3'b0, RAW16-bits



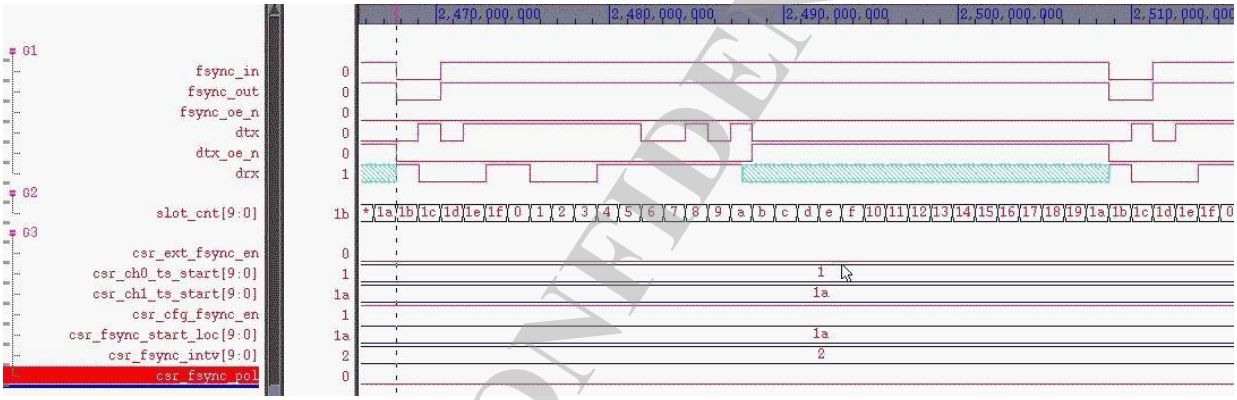
Case 3:

CFG_FSYNC Register: CFG_FSYNC_EN = 1, START_LOC=0x1A, interval=2

CH0_CFG Register: TS_START=1 (disable)

CH1_CFG Register: TS_START=0x1A

PCM_CFG Register: LONG_FSYNC=1'b0, FSYNC_POL=1'b0 (LOW active), DRX_TRI=1'b0, SLOT_MODE=3'b0, RAW16-bits



2.11 Generic DMA Controller

2.11.1 Features

- Supports 16 DMA channels
- Supports 32 bit address.
- Maximum 65535 byte transfer
- Programmable DMA burst size (1, 2, 4, 8, 16 double word burst)
- Supports memory to memory, memory to peripheral, peripheral to memory, peripheral to peripheral transfers.
- Supports continuous mode.
- Supports division of target transfer count into 1 to 256 segments
- Support for combining different channels into a chain.
- Programmable hardware channel priority.
- Interrupts for each channel.

2.11.2 Block Diagram

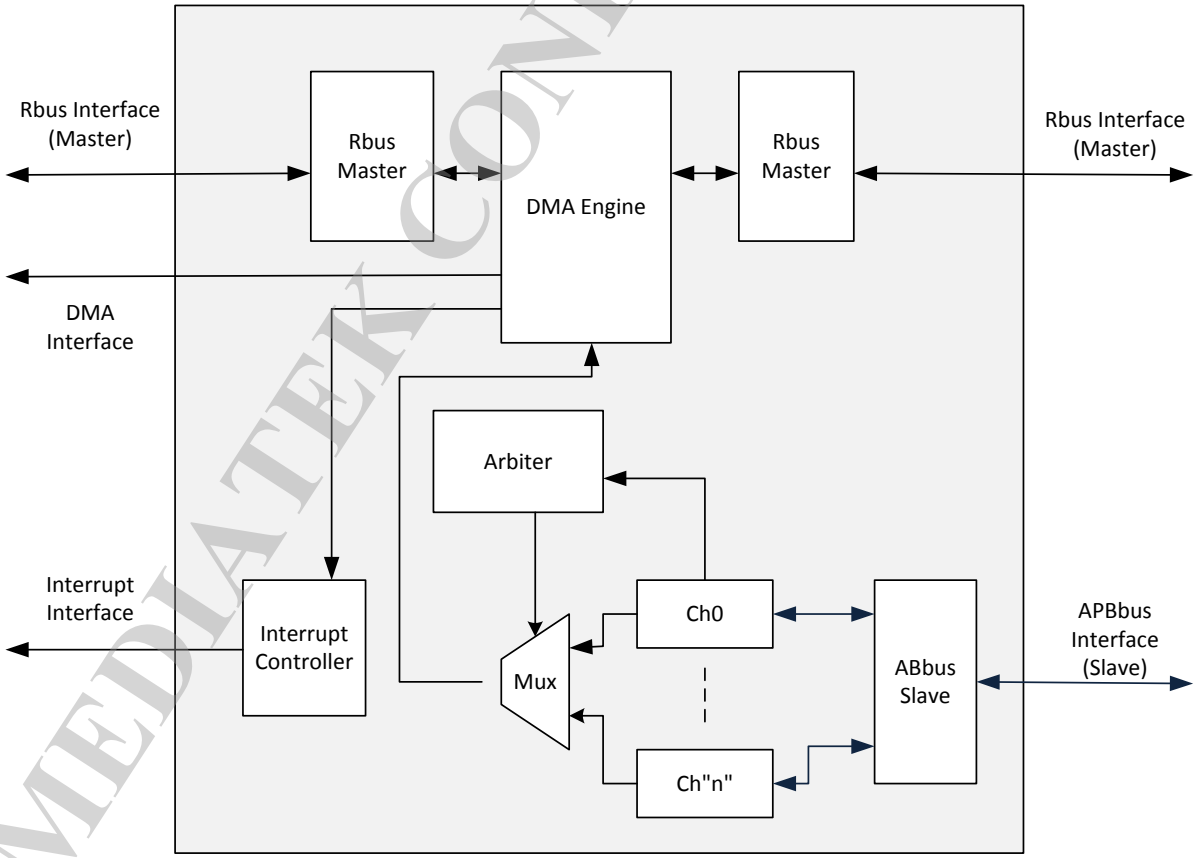


Figure 2-5 Generic DMA Controller Block Diagram

2.11.3 Peripheral Channel Connection

Channel number	Peripheral
0	Reserved
1	Reserved
2	I2S Controller (TXDMA)
3	I2S Controller (RXDMA)
4	PCM Controller (RDMA, channel-0)
5	PCM Controller (RDMA, channel-1)
6	PCM Controller (TDMA, channel-0)
7	PCM Controller (TDMA, channel-1)
8	PCM Controller (RDMA, channel-2)
9	PCM Controller (RDMA, channel-3)
10	PCM Controller (TDMA, channel-2)
11	PCM Controller (TDMA, channel-3)
12	SPI Controller (RXDMA)
13	SPI Controller (TXDMA)
8 to 15	Reserved

2.11.4 Registers

GDMA Changes LOG

Revision	Date	Author	Change Log
0.1	2012/10/15	Mark Wang	Initialization

Module name: GDMA Base address: (+1E002800h)

Address	Name	Width	Register Function
1E002800	GDMA_SA_0	32	Source Address of GDMA Channel 0
1E002804	GDMA_DA_0	32	Destination Address of GDMA Channel 0
1E002808	GDMA_CT0_0	32	Control Register 0 of GDMA Channel 0
1E00280C	GDMA_CT1_0	32	Control Register 1 of GDMA Channel 0
1E002810	GDMA_SA_1	32	Source Address of GDMA Channel 1
1E002814	GDMA_DA_1	32	Destination Address of GDMA Channel 1
1E002818	GDMA_CT0_1	32	Control Register 0 of GDMA Channel 1
1E00281C	GDMA_CT1_1	32	Control Register 1 of GDMA Channel 1
1E002820	GDMA_SA_2	32	Source Address of GDMA Channel 2
1E002824	GDMA_DA_2	32	Destination Address of GDMA Channel 2
1E002828	GDMA_CT0_2	32	Control Register 0 of GDMA Channel 2
1E00282C	GDMA_CT1_2	32	Control Register 1 of GDMA Channel 2
1E002830	GDMA_SA_3	32	Source Address of GDMA Channel 3
1E002834	GDMA_DA_3	32	Destination Address of GDMA Channel 3
1E002838	GDMA_CT0_3	32	Control Register 0 of GDMA Channel 3
1E00283C	GDMA_CT1_3	32	Control Register 1 of GDMA Channel 3
1E002840	GDMA_SA_4	32	Source Address of GDMA Channel 4
1E002844	GDMA_DA_4	32	Destination Address of GDMA Channel 4
1E002848	GDMA_CT0_4	32	Control Register 0 of GDMA Channel 4
1E00284C	GDMA_CT1_4	32	Control Register 1 of GDMA Channel 4
1E002850	GDMA_SA_5	32	Source Address of GDMA Channel 5
1E002854	GDMA_DA_5	32	Destination Address of GDMA Channel 5
1E002858	GDMA_CT0_5	32	Control Register 0 of GDMA Channel 5
1E00285C	GDMA_CT1_5	32	Control Register 1 of GDMA Channel 5
1E002860	GDMA_SA_6	32	Source Address of GDMA Channel 6
1E002864	GDMA_DA_6	32	Destination Address of GDMA Channel 6
1E002868	GDMA_CT0_6	32	Control Register 0 of GDMA Channel 6
1E00286C	GDMA_CT1_6	32	Control Register 1 of GDMA Channel 6
1E002870	GDMA_SA_7	32	Source Address of GDMA Channel 7
1E002874	GDMA_DA_7	32	Destination Address of GDMA Channel 7
1E002878	GDMA_CT0_7	32	Control Register 0 of GDMA Channel 7
1E00287C	GDMA_CT1_7	32	Control Register 1 of GDMA Channel 7
1E002880	GDMA_SA_8	32	Source Address of GDMA Channel 8
1E002884	GDMA_DA_8	32	Destination Address of GDMA Channel 8
1E002888	GDMA_CT0_8	32	Control Register 0 of GDMA Channel 8

1E00288C	GDMA_CT1_8	32	Control Register 1 of GDMA Channel 8
1E002890	GDMA_SA_9	32	Source Address of GDMA Channel 9
1E002894	GDMA_DA_9	32	Destination Address of GDMA Channel 9
1E002898	GDMA_CT0_9	32	Control Register 0 of GDMA Channel 9
1E00289C	GDMA_CT1_9	32	Control Register 1 of GDMA Channel 9
1E0028A0	GDMA_SA_10	32	Source Address of GDMA Channel 10
1E0028A4	GDMA_DA_10	32	Destination Address of GDMA Channel 10
1E0028A8	GDMA_CT0_10	32	Control Register 0 of GDMA Channel 10
1E0028AC	GDMA_CT1_10	32	Control Register 1 of GDMA Channel 10
1E0028B0	GDMA_SA_11	32	Source Address of GDMA Channel 11
1E0028B4	GDMA_DA_11	32	Destination Address of GDMA Channel 11
1E0028B8	GDMA_CT0_11	32	Control Register 0 of GDMA Channel 11
1E0028BC	GDMA_CT1_11	32	Control Register 1 of GDMA Channel 11
1E0028C0	GDMA_SA_12	32	Source Address of GDMA Channel 12
1E0028C4	GDMA_DA_12	32	Destination Address of GDMA Channel 12
1E0028C8	GDMA_CT0_12	32	Control Register 0 of GDMA Channel 12
1E0028CC	GDMA_CT1_12	32	Control Register 1 of GDMA Channel 12
1E0028D0	GDMA_SA_13	32	Source Address of GDMA Channel 13
1E0028D4	GDMA_DA_13	32	Destination Address of GDMA Channel 13
1E0028D8	GDMA_CT0_13	32	Control Register 0 of GDMA Channel 13
1E0028DC	GDMA_CT1_13	32	Control Register 1 of GDMA Channel 13
1E0028E0	GDMA_SA_14	32	Source Address of GDMA Channel 14
1E0028E4	GDMA_DA_14	32	Destination Address of GDMA Channel 14
1E0028E8	GDMA_CT0_14	32	Control Register 0 of GDMA Channel 14
1E0028EC	GDMA_CT1_14	32	Control Register 1 of GDMA Channel 14
1E0028F0	GDMA_SA_15	32	Source Address of GDMA Channel 15
1E0028F4	GDMA_DA_15	32	Destination Address of GDMA Channel 15
1E0028F8	GDMA_CT0_15	32	Control Register 0 of GDMA Channel 15
1E0028FC	GDMA_CT1_15	32	Control Register 1 of GDMA Channel 15
1E002A00	GDMA_UNMASK_INTSTS	32	Unmask Fail Interrupt Status
1E002A04	GDMA_DONE_INTSTS	32	Segment Done Interrupt Status
1E002A20	GDMA_GCT	32	Global Control
1E002A30	GDMA_PERI_ADDR_START_0	32	Peripheral Region 0 Starting Address
1E002A34	GDMA_PERI_ADDR_END_0	32	Peripheral Region 0 End Address
1E002A38	GDMA_PERI_ADDR_START_1	32	Peripheral Region 1 Starting Address
1E002A3C	GDMA_PERI_ADDR_END_1	32	Peripheral Region 1 End Address
1E002A40	GDMA_PERI_ADDR_START_2	32	Peripheral Region 2 Starting Address
1E002A44	GDMA_PERI_ADDR_END_2	32	Peripheral Region 2 End Address
1E002A48	GDMA_PERI_ADDR_START_3	32	Peripheral Region 3 Starting Address

1E002A4C	GDMA_PERI_ADDR_END_3	32	Peripheral Region 3 End Address

1E002800 GDMA_SA_0 Source Address of GDMA Channel 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SOURCE_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SOURCE_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SOURCE_ADDR	**Souce address**

1E002804 GDMA_DA_0 Destination Address of GDMA Channel 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEST_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DEST_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DEST_ADDR	**Destination address**

1E002808 GDMA_CT0_0 Control Register 0 of GDMA Channel 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TARGET_BYTE_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CURR_SEGMENT								SOURCE_ADDR_MODE	DEST_ADDR_MODE	BURST_SIZE			SEGMENT_DONE_INT_EN	CH_EN	SW_MODE_EN
Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**

15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MODE	**Sets the destination address mode** 0: Incremental mode 1: Fix mode
5:3	BURST_SIZE	**Sets the number of double words in each burst transaction** 0: 1 DW 1: 2 DWs 2: 4 DWs 3: 8 DWs 4: 16 DWs 5: Undefined 6: Undefined 7: Undefined
2	SEGMENT_DONE_INT_EN	**Enable the segment done interrupt. This interrupt asserts after transfer of each segment is done.** 0: Disable 1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of bytes transfferred reaches the TARGET_BYTE_CNT** 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.** 0: Hardware mode 1: Software mode

1E00280C GDMA_CT1_0 Control Register 1 of GDMA Channel 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED						NUM_SEGMENT				SOURCE_DMA_REQ					
Type	RO						RW				RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED	CONT_MODE_EN	DEST_DMA_REQ				NEXT_CH2UNMASK							CHERENT_INT_EN	CH_UNMASK_FAIL_INT_EN	CH_MASK
Type	RO	RW	RW				RW							RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2

		32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled
13:8	DEST_DMA_REQ	**Selects the destination DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The destination of the transfer is memory (always ready)
7:3	NEXT_CH2UNMASK	**Selects the channel to clear the CH_MASK bit. When the number of bytes transferred reaches the TARGET_BYTE_CNT, the hardware will clear the CH_MASK field of the NEXT_CH2UNMASK channel. If the hardware does not need to clear CH_MASK field of any channel, this field should be set to the channel itself.** 0: Channel 0 1: Channel 1 n: Channel n
2	COHERENT_INT_EN	**If COHERENT_INT_EN is set, GDMA will issue a dummy read to destination after the last write to destination to avoid data coherent problem. Note: DO NOT set this field if the destination is not MEM. (may corrupt data, if destination is a FIFO)** 0: Disable 1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

1E002810 GDMA_SA_1 Source Address of GDMA Channel 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SOURCE_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SOURCE_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SOURCE_ADDR	**Souce address**

1E002814 GDMA_DA_1 Destination Address of GDMA Channel 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEST_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name	DEST_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DEST_ADDR	**Destination address**

1E002818 GDMA_CT0_1 Control Register 0 of GDMA Channel 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TARGET_BYTE_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CURR_SEGMENT								SOURCE_ADDR_MODE	DEST_ADDR_MODE	BURST_SIZE			SEGMENT_DONE_INT_EN	CH_EN	SW_MODE_EN
Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MODE	**Sets the destination address mode** 0: Incremental mode 1: Fix mode
5:3	BURST_SIZE	**Sets the number of double words in each burst transaction** 0: 1 DW 1: 2 DWs 2: 4 DWs 3: 8 DWs 4: 16 DWs 5: Undefined 6: Undefined 7: Undefined
2	SEGMENT_DONE_INT_EN	**Enable the segment done interrupt. This interrupt asserts after transfer of each segment is done.** 0: Disable 1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of bytes transfferred reaches the TARGET_BYTE_CNT** 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.** 0: Hardware mode 1: Software mode

1E00281C GDMA_CT1_1 Control Register 1 of GDMA Channel 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED						NUM_SEGMENT				SOURCE_DMA_REQ					
Type	RO						RW				RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED	CONT_MODE_EN	DEST_DMA_REQ				NEXT_CH2UNMASK							COHERENT_INT_EN	CH_UNMASK_FAIL_INT_EN	CH_MASK
Type	RO	RW	RW				RW							RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled
13:8	DEST_DMA_REQ	**Selects the destination DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The destination of the transfer is memory (always ready)
7:3	NEXT_CH2UNMASK	**Selects the channel to clear the CH_MASK bit. When the number of bytes transferred reaches the TARGET_BYTE_CNT, the hardware will clear the CH_MASK field of the NEXT_CH2UNMASK channel. If the hardware does not need to clear CH_MASK field of any channel, this field should be set to the channel itself.** 0: Channel 0 1: Channel 1 n: Channel n
2	COHERENT_INT_EN	**If COHERENT_INT_EN is set, GDMA will issue a dummy read to destination after the last write to destination to avoid data coherent problem. Note: DO NOT set this field if the destination is not MEM. (may corrupt data, if destination is a FIFO)** 0: Disable 1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear**

by HW/SW.

0: Channel is not masked
1: Channel is masked

1E002820 GDMA_SA_2 Source Address of GDMA Channel 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SOURCE_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SOURCE_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SOURCE_ADDR	**Souce address**

1E002824 GDMA_DA_2 Destination Address of GDMA Channel 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEST_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DEST_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DEST_ADDR	**Destination address**

1E002828 GDMA_CT0_2 Control Register 0 of GDMA Channel 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TARGET_BYTE_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CURR_SEGMENT								SOURCE_ADDR_MODE	DEST_ADDR_MODE	BURST_SIZE			SEGMENT_DONE_INT_EN	CH_EN	SW_MODE_EN
Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MODE	**Sets the destination address mode** 0: Incremental mode 1: Fix mode
5:3	BURST_SIZE	**Sets the number of double words in each burst transaction** 0: 1 DW 1: 2 DWs 2: 4 DWs 3: 8 DWs 4: 16 DWs 5: Undefined 6: Undefined 7: Undefined
2	SEGMENT_DONE_INT_EN	**Enable the segment done interrupt. This interrupt asserts after transfer of each segment is done.** 0: Disable 1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of bytes transfferred reaches the TARGET_BYTE_CNT** 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.** 0: Hardware mode 1: Software mode

1E00282C GDMA_CT1_2 Control Register 1 of GDMA Channel 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED						NUM_SEGMENT				SOURCE_DMA_REQ					
Type	RO						RW				RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED	CONT_MODE_EN	DEST_DMA_REQ						NEXT_CH2UNMASK					CHERENT_INT_EN	CH_UNMASK_FAIL_INT_EN	CH_MASK
Type	RO	RW	RW						RW					RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0

		1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled
13:8	DEST_DMA_REQ	**Selects the destination DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The destination of the transfer is memory (always ready)
7:3	NEXT_CH2UNMASK	**Selects the channel to clear the CH_MASK bit. When the number of bytes transferred reaches the TARGET_BYTE_CNT, the hardware will clear the CH_MASK field of the NEXT_CH2UNMASK channel. If the hardware does not need to clear CH_MASK field of any channel, this field should be set to the channel itself.** 0: Channel 0 1: Channel 1 n: Channel n
2	COHERENT_INT_EN	**If COHERENT_INT_EN is set, GDMA will issue a dummy read to destination after the last write to destination to avoid data coherent problem. Note: DO NOT set this field if the destination is not MEM. (may corrupt data, if destination is a FIFO)** 0: Disable 1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

1E002830 GDMA_SA_3 Source Address of GDMA Channel 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SOURCE_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SOURCE_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SOURCE_ADDR	**Souce address**

1E002834 GDMA_DA_3 Destination Address of GDMA Channel 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEST_ADDR[31:16]															
Type	RW															

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DEST_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DEST_ADDR	**Destination address**

1E002838 GDMA_CT0_3 Control Register 0 of GDMA Channel 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TARGET_BYTE_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CURR_SEGMENT								SOURCE_ADDR_MODE	DEST_ADDR_MODE	BURST_SIZE			SEGMENT_DONE_INT_EN	CH_EN	SW_MODE_EN
Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MODE	**Sets the destination address mode** 0: Incremental mode 1: Fix mode
5:3	BURST_SIZE	**Sets the number of double words in each burst transaction** 0: 1 DW 1: 2 DWs 2: 4 DWs 3: 8 DWs 4: 16 DWs 5: Undefined 6: Undefined 7: Undefined
2	SEGMENT_DONE_INT_EN	**Enable the segment done interrupt. This interrupt asserts after transfer of each segment is done.** 0: Disable 1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of bytes transfferred reaches the TARGET_BYTE_CNT** 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.**

0: Hardware mode
1: Software mode

1E00283C GDMA_CT1_3 Control Register 1 of GDMA Channel 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED						NUM_SEGMENT				SOURCE_DMA_REQ					
Type	RO						RW				RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED	CONT_MODE_EN	DEST_DMA_REQ				NEXT_CH2UNMASK							COHERENT_INT_EN	CH_UNMASK_FAIL_INT_EN	CH_MASK
Type	RO	RW	RW				RW							RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled
13:8	DEST_DMA_REQ	**Selects the destination DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The destination of the transfer is memory (always ready)
7:3	NEXT_CH2UNMASK	**Selects the channel to clear the CH_MASK bit. When the number of bytes transferred reaches the TARGET_BYTE_CNT, the hardware will clear the CH_MASK field of the NEXT_CH2UNMASK channel. If the hardware does not need to clear CH_MASK field of any channel, this field should be set to the channel itself.** 0: Channel 0 1: Channel 1 n: Channel n
2	COHERENT_INT_EN	**If COHERENT_INT_EN is set, GDMA will issue a dummy read to destination after the last write to destination to avoid data coherent problem. Note: DO NOT set this field if the destination is not MEM. (may corrupt data, if destination is a FIFO)** 0: Disable 1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable

		1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

1E002840 GDMA_SA_4 Source Address of GDMA Channel 4 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SOURCE_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SOURCE_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SOURCE_ADDR	**Souce address**

1E002844 GDMA_DA_4 Destination Address of GDMA Channel 4 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEST_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DEST_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DEST_ADDR	**Destination address**

1E002848 GDMA_CT0_4 Control Register 0 of GDMA Channel 4 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TARGET_BYTE_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CURR_SEGMENT								SOURCE_ADDR_MODE	DEST_ADDR_MODE	BURST_SIZE			SEGMENT_DONE_INT_EN	CH_EN	SW_MODE_EN
Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MODE	**Sets the destination address mode** 0: Incremental mode 1: Fix mode
5:3	BURST_SIZE	**Sets the number of double words in each burst transaction** 0: 1 DW 1: 2 DWs 2: 4 DWs 3: 8 DWs 4: 16 DWs 5: Undefined 6: Undefined 7: Undefined
2	SEGMENT_DONE_INT_EN	**Enable the segment done interrupt. This interrupt asserts after transfer of each segment is done.** 0: Disable 1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of bytes transfferred reaches the TARGET_BYTE_CNT** 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.** 0: Hardware mode 1: Software mode

1E00284C GDMA_CT1_4 Control Register 1 of GDMA Channel 4 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED						NUM_SEGMENT				SOURCE_DMA_REQ					
Type	RO						RW				RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED	CONT_MODE_EN	DEST_DMA_REQ				NEXT_CH2UNMASK							COHERENT_INT_EN	CH_UNMASK_FAIL_INT_EN	CH_MASK
Type	RO	RW	RW				RW							RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**

21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled
13:8	DEST_DMA_REQ	**Selects the destination DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The destination of the transfer is memory (always ready)
7:3	NEXT_CH2UNMASK	**Selects the channel to clear the CH_MASK bit. When the number of bytes transferred reaches the TARGET_BYTE_CNT, the hardware will clear the CH_MASK field of the NEXT_CH2UNMASK channel. If the hardware does not need to clear CH_MASK field of any channel, this field should be set to the channel itself.** 0: Channel 0 1: Channel 1 n: Channel n
2	COHERENT_INT_EN	**If COHERENT_INT_EN is set, GDMA will issue a dummy read to destination after the last write to destination to avoid data coherent problem. Note: DO NOT set this field if the destination is not MEM. (may corrupt data, if destination is a FIFO)** 0: Disable 1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

1E002850 GDMA_SA_5 Source Address of GDMA Channel 5 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SOURCE_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SOURCE_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SOURCE_ADDR	**Souce address**

1E002854 GDMA_DA_5 Destination Address of GDMA Channel 5 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Name	DEST_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DEST_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DEST_ADDR	**Destination address**

1E002858 GDMA_CT0_5 Control Register 0 of GDMA Channel 5 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TARGET_BYTE_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CURR_SEGMENT								SOURCE_ADDR_MODE	DEST_ADDR_MODE	BURST_SIZE			SEGMENT_DONE_INT_EN	CH_EN	SW_MODE_EN
Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MODE	**Sets the destination address mode** 0: Incremental mode 1: Fix mode
5:3	BURST_SIZE	**Sets the number of double words in each burst transaction** 0: 1 DW 1: 2 DWs 2: 4 DWs 3: 8 DWs 4: 16 DWs 5: Undefined 6: Undefined 7: Undefined
2	SEGMENT_DONE_INT_EN	**Enable the segment done interrupt. This interrupt asserts after transfer of each segment is done.** 0: Disable 1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of bytes transfferred reaches the TARGET_BYTE_CNT** 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts**

when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.

0: Hardware mode
1: Software mode

1E00285C GDMA_CT1_5 Control Register 1 of GDMA Channel 5 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED						NUM_SEGMENT				SOURCE_DMA_REQ					
Type	RO						RW				RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED	CONT_MODE_EN	DEST_DMA_REQ				NEXT_CH2UNMASK							COHERENT_INT_EN	CH_UNMASK_FAIL_INT_EN	CH_MASK
Type	RO	RW	RW				RW							RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled
13:8	DEST_DMA_REQ	**Selects the destination DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The destination of the transfer is memory (always ready)
7:3	NEXT_CH2UNMASK	**Selects the channel to clear the CH_MASK bit. When the number of bytes transferred reaches the TARGET_BYTE_CNT, the hardware will clear the CH_MASK field of the NEXT_CH2UNMASK channel. If the hardware does not need to clear CH_MASK field of any channel, this field should be set to the channel itself.** 0: Channel 0 1: Channel 1 n: Channel n
2	COHERENT_INT_EN	**If COHERENT_INT_EN is set, GDMA will issue a dummy read to destination after the last write to destination to avoid data coherent problem. Note: DO NOT set this field if the destination is not MEM. (may corrupt data, if destination is a FIFO)** 0: Disable 1: Enable
1	CH_UNMASK_FAIL_I	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field**

	NT_EN	**of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

1E002860 GDMA_SA_6 Source Address of GDMA Channel 6 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SOURCE_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SOURCE_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SOURCE_ADDR	**Souce address**

1E002864 GDMA_DA_6 Destination Address of GDMA Channel 6 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEST_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DEST_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DEST_ADDR	**Destination address**

1E002868 GDMA_CT0_6 Control Register 0 of GDMA Channel 6 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TARGET_BYTE_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CURR_SEGMENT								SOURCE_ADDR_MODE	DEST_ADDR_MODE	BURST_SIZE			SEGMENT_DONE_INT_EN	CH_EN	SW_MODE_EN

Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MODE	**Sets the destination address mode** 0: Incremental mode 1: Fix mode
5:3	BURST_SIZE	**Sets the number of double words in each burst transaction** 0: 1 DW 1: 2 DWs 2: 4 DWs 3: 8 DWs 4: 16 DWs 5: Undefined 6: Undefined 7: Undefined
2	SEGMENT_DONE_INT_EN	**Enable the segment done interrupt. This interrupt asserts after transfer of each segment is done.** 0: Disable 1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of bytes transfferred reaches the TARGET_BYTE_CNT** 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.** 0: Hardware mode 1: Software mode

1E00286C GDMA_CT1_6 Control Register 1 of GDMA Channel 6 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED						NUM_SEGMENT				SOURCE_DMA_REQ					
Type	RO						RW				RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED	CONT_MODE_EN	DEST_DMA_REQ				NEXT_CH2UNMASK							COHERENT_INT_EN	CH_UNMASK_FAIL_INT_EN	CH_MASK
Type	RO	RW	RW				RW							RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the**

		TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled
13:8	DEST_DMA_REQ	**Selects the destination DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The destination of the transfer is memory (always ready)
7:3	NEXT_CH2UNMASK	**Selects the channel to clear the CH_MASK bit. When the number of bytes transferred reaches the TARGET_BYTE_CNT, the hardware will clear the CH_MASK field of the NEXT_CH2UNMASK channel. If the hardware does not need to clear CH_MASK field of any channel, this field should be set to the channel itself.** 0: Channel 0 1: Channel 1 n: Channel n
2	COHERENT_INT_EN	**If COHERENT_INT_EN is set, GDMA will issue a dummy read to destination after the last write to destination to avoid data coherent problem. Note: DO NOT set this field if the destination is not MEM. (may corrupt data, if destination is a FIFO)** 0: Disable 1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

1E002870 GDMA_SA_7 Source Address of GDMA Channel 7 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SOURCE_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SOURCE_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SOURCE_ADDR	**Souce address**

1E002874 GDMA_DA_7 Destination Address of GDMA Channel 7 0000000

0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEST_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DEST_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DEST_ADDR	**Destination address**

1E002878 GDMA_CT0_7 Control Register 0 of GDMA Channel 7 0000000 0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TARGET_BYTE_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CURR_SEGMENT								SOURCE_ADDR_MODE	DEST_ADDR_MODE	BURST_SIZE			SEGMENT_DONE_INT_EN	CH_EN	SW_MODE_EN
Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MODE	**Sets the destination address mode** 0: Incremental mode 1: Fix mode
5:3	BURST_SIZE	**Sets the number of double words in each burst transaction** 0: 1 DW 1: 2 DWs 2: 4 DWs 3: 8 DWs 4: 16 DWs 5: Undefined 6: Undefined 7: Undefined
2	SEGMENT_DONE_INT_EN	**Enable the segment done interrupt. This interrupt asserts after transfer of each segment is done.** 0: Disable 1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of bytes transfferred reaches the TARGET_BYTE_CNT** 0: Disable

		1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.** 0: Hardware mode 1: Software mode

1E00287C GDMA_CT1_7 Control Register 1 of GDMA Channel 7 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED						NUM_SEGMENT				SOURCE_DMA_REQ					
Type	RO						RW				RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED	CONT_MODE_EN	DEST_DMA_REQ				NEXT_CH2UNMASK							COHERENT_INT_EN	CH_UNMASK_FAIL_INT_EN	CH_MASK
Type	RO	RW	RW				RW							RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled
13:8	DEST_DMA_REQ	**Selects the destination DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The destination of the transfer is memory (always ready)
7:3	NEXT_CH2UNMASK	**Selects the channel to clear the CH_MASK bit. When the number of bytes transferred reaches the TARGET_BYTE_CNT, the hardware will clear the CH_MASK field of the NEXT_CH2UNMASK channel. If the hardware does not need to clear CH_MASK field of any channel, this field should be set to the channel itself.** 0: Channel 0 1: Channel 1 n: Channel n
2	COHERENT_INT_EN	**If COHERENT_INT_EN is set, GDMA will issue a dummy read to destination after the last write to destination to avoid data coherent problem. Note: DO NOT set this field if the destination is not MEM. (may corrupt data, if destination is a FIFO)** 0: Disable

		1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

1E002880 GDMA_SA_8 Source Address of GDMA Channel 8 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SOURCE_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SOURCE_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SOURCE_ADDR	**Souce address**

1E002884 GDMA_DA_8 Destination Address of GDMA Channel 8 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEST_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DEST_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DEST_ADDR	**Destination address**

1E002888 GDMA_CT0_8 Control Register 0 of GDMA Channel 8 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TARGET_BYTE_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CURR_SEGMENT								SOURCE_ADDR_MOD	DEST_ADDR_MOD	BURST_SIZE			SEGMENT_DONE_I	CH_EN	SW_MODE_EN

									ODE	E				NT_EN		
Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MODE	**Sets the destination address mode** 0: Incremental mode 1: Fix mode
5:3	BURST_SIZE	**Sets the number of double words in each burst transaction** 0: 1 DW 1: 2 DWs 2: 4 DWs 3: 8 DWs 4: 16 DWs 5: Undefined 6: Undefined 7: Undefined
2	SEGMENT_DONE_INT_EN	**Enable the segment done interrupt. This interrupt asserts after transfer of each segment is done.** 0: Disable 1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of bytes transfferred reaches the TARGET_BYTE_CNT** 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.** 0: Hardware mode 1: Software mode

1E00288C GDMA_CT1_8 Control Register 1 of GDMA Channel 8 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED						NUM_SEGMENT				SOURCE_DMA_REQ					
Type	RO						RW				RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED	CONT_MODE_EN	DEST_DMA_REQ				NEXT_CH2UNMASK							COHERENT_INT_EN	CH_UNMASK_FAIL_INT_EN	CH_MASK
Type	RO	RW	RW				RW							RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled
13:8	DEST_DMA_REQ	**Selects the destination DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The destination of the transfer is memory (always ready)
7:3	NEXT_CH2UNMASK	**Selects the channel to clear the CH_MASK bit. When the number of bytes transferred reaches the TARGET_BYTE_CNT, the hardware will clear the CH_MASK field of the NEXT_CH2UNMASK channel. If the hardware does not need to clear CH_MASK field of any channel, this field should be set to the channel itself.** 0: Channel 0 1: Channel 1 n: Channel n
2	COHERENT_INT_EN	**If COHERENT_INT_EN is set, GDMA will issue a dummy read to destination after the last write to destination to avoid data coherent problem. Note: DO NOT set this field if the destination is not MEM. (may corrupt data, if destination is a FIFO)** 0: Disable 1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

1E002890 GDMA_SA_9 Source Address of GDMA Channel 9 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SOURCE_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SOURCE_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SOURCE_ADDR	**Souce address**

1E002894 GDMA_DA_9 Destination Address of GDMA Channel 9 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEST_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DEST_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DEST_ADDR	**Destination address**

1E002898 GDMA_CT0_9 Control Register 0 of GDMA Channel 9 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TARGET_BYTE_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CURR_SEGMENT								SOURCE_ADDR_MODE	DEST_ADDR_MODE	BURST_SIZE			SEGMENT_DONE_INT_EN	CH_EN	SW_MODE_EN
Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MODE	**Sets the destination address mode** 0: Incremental mode 1: Fix mode
5:3	BURST_SIZE	**Sets the number of double words in each burst transaction** 0: 1 DW 1: 2 DWs 2: 4 DWs 3: 8 DWs 4: 16 DWs 5: Undefined 6: Undefined 7: Undefined
2	SEGMENT_DONE_INT_EN	**Enable the segment done interrupt. This interrupt asserts after transfer of each segment is done.** 0: Disable 1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of**

		bytes transfferred reaches the TARGET_BYTE_CNT 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.** 0: Hardware mode 1: Software mode

1E00289C GDMA_CT1_9 Control Register 1 of GDMA Channel 9 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED						NUM_SEGMENT				SOURCE_DMA_REQ					
Type	RO						RW				RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED	CONT_MODE_EN	DEST_DMA_REQ				NEXT_CH2UNMASK							COHERENT_INT_EN	CH_UNMASK_FAIL_INT_EN	CH_MASK
Type	RO	RW	RW				RW							RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled
13:8	DEST_DMA_REQ	**Selects the destination DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The destination of the transfer is memory (always ready)
7:3	NEXT_CH2UNMASK	**Selects the channel to clear the CH_MASK bit. When the number of bytes transferred reaches the TARGET_BYTE_CNT, the hardware will clear the CH_MASK field of the NEXT_CH2UNMASK channel. If the hardware does not need to clear CH_MASK field of any channel, this field should be set to the channel itself.** 0: Channel 0 1: Channel 1 n: Channel n
2	COHERENT_INT_EN	**If COHERENT_INT_EN is set, GDMA will issue a dummy read to destination after the last write to destination to avoid data coherent problem. Note: DO NOT set**

		this field if the destination is not MEM. (may corrupt data, if destination is a FIFO) 0: Disable 1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

1E0028A0 GDMA_SA_10 Source Address of GDMA Channel 10 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SOURCE_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SOURCE_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SOURCE_ADDR	**Souce address**

1E0028A4 GDMA_DA_10 Destination Address of GDMA Channel 10 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEST_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DEST_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DEST_ADDR	**Destination address**

1E0028A8 GDMA_CT0_10 Control Register 0 of GDMA Channel 10 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TARGET_BYTE_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CURR_SEGMENT								SOURCE_ADD	DEST_ADDR	BURST_SIZE			SEGMENT_D	CH_EN	SW_MODE_E

									DR_MODE	_MODE				ONE_INT_EN		N
Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MODE	**Sets the destination address mode** 0: Incremental mode 1: Fix mode
5:3	BURST_SIZE	**Sets the number of double words in each burst transaction** 0: 1 DW 1: 2 DWs 2: 4 DWs 3: 8 DWs 4: 16 DWs 5: Undefined 6: Undefined 7: Undefined
2	SEGMENT_DONE_INT_EN	**Enable the segment done interrupt. This interrupt asserts after transfer of each segment is done.** 0: Disable 1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of bytes transfferred reaches the TARGET_BYTE_CNT** 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.** 0: Hardware mode 1: Software mode

1E0028AC GDMA_CT1_10 Control Register 1 of GDMA Channel 10 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED						NUM_SEGMENT				SOURCE_DMA_REQ					
Type	RO						RW				RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED	CONT_MODE_EN	DEST_DMA_REQ				NEXT_CH2UNMASK							COHERENT_INT_EN	CH_UNMASK_FAIL_INT_EN	CH_MASK
Type	RO	RW	RW				RW							RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled
13:8	DEST_DMA_REQ	**Selects the destination DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The destination of the transfer is memory (always ready)
7:3	NEXT_CH2UNMASK	**Selects the channel to clear the CH_MASK bit. When the number of bytes transferred reaches the TARGET_BYTE_CNT, the hardware will clear the CH_MASK field of the NEXT_CH2UNMASK channel. If the hardware does not need to clear CH_MASK field of any channel, this field should be set to the channel itself.** 0: Channel 0 1: Channel 1 n: Channel n
2	COHERENT_INT_EN	**If COHERENT_INT_EN is set, GDMA will issue a dummy read to destination after the last write to destination to avoid data coherent problem. Note: DO NOT set this field if the destination is not MEM. (may corrupt data, if destination is a FIFO)** 0: Disable 1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

1E0028B0 GDMA_SA_11 Source Address of GDMA Channel 11 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SOURCE_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SOURCE_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SOURCE_ADDR	**Souce address**

1E0028B4 GDMA_DA_11 Destination Address of GDMA Channel 11 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEST_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DEST_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DEST_ADDR	**Destination address**

1E0028B8 GDMA_CT0_11 Control Register 0 of GDMA Channel 11 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TARGET_BYTE_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CURR_SEGMENT								SOURCE_ADDR_MODE	DEST_ADDR_MODE	BURST_SIZE			SEGMENT_DONE_INT_EN	CH_EN	SW_MODE_EN
Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MODE	**Sets the destination address mode** 0: Incremental mode 1: Fix mode
5:3	BURST_SIZE	**Sets the number of double words in each burst transaction** 0: 1 DW 1: 2 DWs 2: 4 DWs 3: 8 DWs 4: 16 DWs 5: Undefined 6: Undefined 7: Undefined
2	SEGMENT_DONE_INT_EN	**Enable the segment done interrupt. This interrupt asserts after transfer of each segment is done.** 0: Disable

		1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of bytes transfferred reaches the TARGET_BYTE_CNT** 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.** 0: Hardware mode 1: Software mode

1E0028BC GDMA_CT1_11 Control Register 1 of GDMA Channel 11 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED						NUM_SEGMENT				SOURCE_DMA_REQ					
Type	RO						RW				RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED	CONT_MODE_EN	DEST_DMA_REQ				NEXT_CH2UNMASK							COHERENT_INT_EN	CH_UNMASK_FAIL_INT_EN	CH_MASK
Type	RO	RW	RW				RW							RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled
13:8	DEST_DMA_REQ	**Selects the destination DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The destination of the transfer is memory (always ready)
7:3	NEXT_CH2UNMASK	**Selects the channel to clear the CH_MASK bit. When the number of bytes transferred reaches the TARGET_BYTE_CNT, the hardware will clear the CH_MASK field of the NEXT_CH2UNMASK channel. If the hardware does not need to clear CH_MASK field of any channel, this field should be set to the channel itself.** 0: Channel 0 1: Channel 1 n: Channel n

2	COHERENT_INT_EN	**If COHERENT_INT_EN is set, GDMA will issue a dummy read to destination after the last write to destination to avoid data coherent problem. Note: DO NOT set this field if the destination is not MEM. (may corrupt data, if destination is a FIFO)** 0: Disable 1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

1E0028C0 GDMA_SA_12 Source Address of GDMA Channel 12 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SOURCE_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SOURCE_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SOURCE_ADDR	**Souce address**

1E0028C4 GDMA_DA_12 Destination Address of GDMA Channel 12 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEST_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DEST_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DEST_ADDR	**Destination address**

1E0028C8 GDMA_CT0_12 Control Register 0 of GDMA Channel 12 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TARGET_BYTE_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CURR_SEGMENT								SOUR	DEST_	BURST_SIZE			SEGM	CH_EN	SW_M

									CE_ADDR_MODE	ADDR_MODE				ENT_DONE_INT_EN		ODE_EN
Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MODE	**Sets the destination address mode** 0: Incremental mode 1: Fix mode
5:3	BURST_SIZE	**Sets the number of double words in each burst transaction** 0: 1 DW 1: 2 DWs 2: 4 DWs 3: 8 DWs 4: 16 DWs 5: Undefined 6: Undefined 7: Undefined
2	SEGMENT_DONE_INT_EN	**Enable the segment done interrupt. This interrupt asserts after transfer of each segment is done.** 0: Disable 1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of bytes transfferred reaches the TARGET_BYTE_CNT** 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.** 0: Hardware mode 1: Software mode

1E0028CC GDMA_CT1_12 Control Register 1 of GDMA Channel 12 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED						NUM_SEGMENT				SOURCE_DMA_REQ					
Type	RO						RW				RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED	CONT_MODE_EN	DEST_DMA_REQ				NEXT_CH2UNMASK							COHERENT_INT_EN	CH_UNMASK_FAIL_INT_EN	CH_MASK
Type	RO	RW	RW				RW							RW	RW	RW

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled
13:8	DEST_DMA_REQ	**Selects the destination DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The destination of the transfer is memory (always ready)
7:3	NEXT_CH2UNMASK	**Selects the channel to clear the CH_MASK bit. When the number of bytes transferred reaches the TARGET_BYTE_CNT, the hardware will clear the CH_MASK field of the NEXT_CH2UNMASK channel. If the hardware does not need to clear CH_MASK field of any channel, this field should be set to the channel itself.** 0: Channel 0 1: Channel 1 n: Channel n
2	COHERENT_INT_EN	**If COHERENT_INT_EN is set, GDMA will issue a dummy read to destination after the last write to destination to avoid data coherent problem. Note: DO NOT set this field if the destination is not MEM. (may corrupt data, if destination is a FIFO)** 0: Disable 1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

1E0028D0 GDMA_SA_13 Source Address of GDMA Channel 13 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SOURCE_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SOURCE_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

31:0	SOURCE_ADDR	**Souce address**

1E0028D4 GDMA_DA_13 Destination Address of GDMA Channel 13 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEST_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DEST_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DEST_ADDR	**Destination address**

1E0028D8 GDMA_CT0_13 Control Register 0 of GDMA Channel 13 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TARGET_BYTE_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CURR_SEGMENT								SOURCE_ADDR_MODE	DEST_ADDR_MODE	BURST_SIZE			SEGMENT_DONE_INT_EN	CH_EN	SW_MODE_EN
Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MODE	**Sets the destination address mode** 0: Incremental mode 1: Fix mode
5:3	BURST_SIZE	**Sets the number of double words in each burst transaction** 0: 1 DW 1: 2 DWs 2: 4 DWs 3: 8 DWs 4: 16 DWs 5: Undefined 6: Undefined 7: Undefined
2	SEGMENT_DONE_IN	**Enable the segment done interrupt. This interrupt asserts after transfer of each**

	T_EN	**segment is done.** 0: Disable 1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of bytes transfferred reaches the TARGET_BYTE_CNT** 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.** 0: Hardware mode 1: Software mode

1E0028DC GDMA_CT1_13 Control Register 1 of GDMA Channel 13 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED						NUM_SEGMENT				SOURCE_DMA_REQ					
Type	RO						RW				RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED	CONT_MODE_EN	DEST_DMA_REQ				NEXT_CH2UNMASK							COHERENT_INT_EN	CH_UNMASK_FAIL_INT_EN	CH_MASK
Type	RO	RW	RW				RW							RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled
13:8	DEST_DMA_REQ	**Selects the destination DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The destination of the transfer is memory (always ready)
7:3	NEXT_CH2UNMASK	**Selects the channel to clear the CH_MASK bit. When the number of bytes transferred reaches the TARGET_BYTE_CNT, the hardware will clear the CH_MASK field of the NEXT_CH2UNMASK channel. If the hardware does not need to clear CH_MASK field of any channel, this field should be set to the channel itself.** 0: Channel 0

		1: Channel 1 n: Channel n
2	COHERENT_INT_EN	**If COHERENT_INT_EN is set, GDMA will issue a dummy read to destination after the last write to destination to avoid data coherent problem. Note: DO NOT set this field if the destination is not MEM. (may corrupt data, if destination is a FIFO)** 0: Disable 1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

1E0028E0 GDMA_SA_14 Source Address of GDMA Channel 14 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SOURCE_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SOURCE_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SOURCE_ADDR	**Souce address**

1E0028E4 GDMA_DA_14 Destination Address of GDMA Channel 14 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEST_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DEST_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DEST_ADDR	**Destination address**

1E0028E8 GDMA_CT0_14 Control Register 0 of GDMA Channel 14 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TARGET_BYTE_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CURR_SEGMENT								SOURCE_ADDR_MODE	DEST_ADDR_MODE	BURST_SIZE			SEGMENT_DONE_INT_EN	CH_EN	SW_MODE_EN
Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MODE	**Sets the destination address mode** 0: Incremental mode 1: Fix mode
5:3	BURST_SIZE	**Sets the number of double words in each burst transaction** 0: 1 DW 1: 2 DWs 2: 4 DWs 3: 8 DWs 4: 16 DWs 5: Undefined 6: Undefined 7: Undefined
2	SEGMENT_DONE_INT_EN	**Enable the segment done interrupt. This interrupt asserts after transfer of each segment is done.** 0: Disable 1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of bytes transfferred reaches the TARGET_BYTE_CNT** 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.** 0: Hardware mode 1: Software mode

1E0028EC GDMA_CT1_14 Control Register 1 of GDMA Channel 14 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED						NUM_SEGMENT				SOURCE_DMA_REQ					
Type	RO						RW				RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED	CONT_MODE_EN	DEST_DMA_REQ				NEXT_CH2UNMASK							CHERENT_INT_EN	CH_UNMASK_FAIL_IN	CH_MASK

															T_EN	
Type	RO	RW	RW						RW					RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled
13:8	DEST_DMA_REQ	**Selects the destination DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The destination of the transfer is memory (always ready)
7:3	NEXT_CH2UNMASK	**Selects the channel to clear the CH_MASK bit. When the number of bytes transferred reaches the TARGET_BYTE_CNT, the hardware will clear the CH_MASK field of the NEXT_CH2UNMASK channel. If the hardware does not need to clear CH_MASK field of any channel, this field should be set to the channel itself.** 0: Channel 0 1: Channel 1 n: Channel n
2	COHERENT_INT_EN	**If COHERENT_INT_EN is set, GDMA will issue a dummy read to destination after the last write to destination to avoid data coherent problem. Note: DO NOT set this field if the destination is not MEM. (may corrupt data, if destination is a FIFO)** 0: Disable 1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

1E0028F0 GDMA_SA_15 Source Address of GDMA Channel 15 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SOURCE_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SOURCE_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SOURCE_ADDR	**Souce address**

1E0028F4 GDMA_DA_15 Destination Address of GDMA Channel 15 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEST_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DEST_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DEST_ADDR	**Destination address**

1E0028F8 GDMA_CT0_15 Control Register 0 of GDMA Channel 15 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TARGET_BYTE_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CURR_SEGMENT								SOURCE_ADDR_MODE	DEST_ADDR_MODE	BURST_SIZE			SEGMENT_DONE_INT_EN	CH_EN	SW_MODE_EN
Type	RO								RW	RW	RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TARGET_BYTE_CNT	**The number of bytes to be transferred**
15:8	CURR_SEGMENT	**Indicates the current segment (0 to 255)**
7	SOURCE_ADDR_MODE	**Sets the source address mode** 0: Incremental mode 1: Fix mode
6	DEST_ADDR_MODE	**Sets the destination address mode** 0: Incremental mode 1: Fix mode
5:3	BURST_SIZE	**Sets the number of double words in each burst transaction** 0: 1 DW 1: 2 DWs 2: 4 DWs 3: 8 DWs 4: 16 DWs 5: Undefined 6: Undefined

		7: Undefined
2	SEGMENT_DONE_INT_EN	**Enable the segment done interrupt. This interrupt asserts after transfer of each segment is done.** 0: Disable 1: Enable
1	CH_EN	**If CONT_MODE_EN=0, this bit is de-asserted by hardware after the number of bytes transfferred reaches the TARGET_BYTE_CNT** 0: Disable 1: Enable
0	SW_MODE_EN	**Software mode enable. If software mode enable is set, the data transfer starts when the CH_EN bit is set. Otherwise, the data transfer starts when the DMA request is asserted.** 0: Hardware mode 1: Software mode

1E0028FC GDMA_CT1_15 Control Register 1 of GDMA Channel 15 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED						NUM_SEGMENT				SOURCE_DMA_REQ					
Type	RO						RW				RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED	CONT_MODE_EN	DEST_DMA_REQ						NEXT_CH2UNMASK					COHERENT_INT_EN	CH_UNMASK_FAIL_INT_EN	CH_MASK
Type	RO	RW	RW						RW					RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
25:22	NUM_SEGMENT	**the number of segments=2N, where N is the value of this field. Valid values for this field are N=0 to 8. The segment size=(TARGET_BYTE_CNT/2N). It the TARGET_BYTE_CNT is not a multiple of 2N, the segment size = {(TARGET_BYTE_CNT/2N) + 1}.**
21:16	SOURCE_DMA_REQ	**Selects the source DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The source of the transfer is memory (always ready)
14	CONT_MODE_EN	**If CONT_MODE_EN=1, HW will NOT clear CH_EN after the number of bytes transferred reachs TARGET_BYTE_CNT. Otherwise, HW will clear CH_EN will clear the CH_EN.** 0: Continuous mode is disabled 1: Continuous mode is enabled
13:8	DEST_DMA_REQ	**Selects the destination DMA request** 0: DMA_REQ0 1: DMA_REQ1 2: DMA_REQ2 32: The destination of the transfer is memory (always ready)
7:3	NEXT_CH2UNMASK	**Selects the channel to clear the CH_MASK bit. When the number of bytes transferred reaches the TARGET_BYTE_CNT, the hardware will clear the CH_MASK field of the NEXT_CH2UNMASK channel. If the hardware does not need to clear CH_MASK field of any channel, this field should be set to the**

		channel itself. 0: Channel 0 1: Channel 1 n: Channel n
2	COHERENT_INT_EN	**If COHERENT_INT_EN is set, GDMA will issue a dummy read to destination after the last write to destination to avoid data coherent problem. Note: DO NOT set this field if the destination is not MEM. (may corrupt data, if destination is a FIFO)** 0: Disable 1: Enable
1	CH_UNMASK_FAIL_INT_EN	**If this field is set, an interrupt will be assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.** 0: Disable 1: Enable
0	CH_MASK	**When this field is set, the transfer of this channel is gated untill this field is clear by HW/SW.** 0: Channel is not masked 1: Channel is masked

1E002A00 GDMA_UNMASK_INTSTS Unmask Fail Interrupt Status 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	UNMASK_FAIL_INTSTS[31:16]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	UNMASK_FAIL_INTSTS[15:0]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	UNMASK_FAIL_INTSTS	**This field is the bit-map of unmask fail interrupt status of each channel. The unmask fail interrupt will assert when HW detect the CH_MASK field of NEXT_CH2UNMASK channel is 1'b0 while trying to clear it.**

1E002A04 GDMA_DONE_INTSTS Segment Done Interrupt Status 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SEGMENT_DONE_INTSTS[31:16]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEGMENT_DONE_INTSTS[15:0]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SEGMENT_DONE_INTSTS	**This field is the bit-map of segment done interrupt status of each channel. The segment done interrupt will assert when each segment is transferred completely.**

1E002A20 GDMA_GCT Global Control 0000000E

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESERVED[26:11]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESERVED[10:0]											TOTAL_CH_NUM		IP_VER		ARB_MODE
Type	RO											RO		RO		RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0

Bit(s)	Name	Description
4:3	TOTAL_CH_NUM	**Total channel number supported** 0: 8 channels 1: 16 channels 2: 32 channels 3: Undefined
2:1	IP_VER	**GDMA core version**
0	ARB_MODE	**Arbitration mode selection** 0: channel 0 has highest priority and others are round-robin 1: All channel are round-robin

1E002A30 GDMA_PERI_ADDR_START_0 Peripheral Region 0 Starting Address 1C000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PERI_ADDR_START_0[31:16]															
Type	RW															
Reset	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PERI_ADDR_START_0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PERI_ADDR_START_0	**GDMA request will direct to peripheral bus if the request address >= PERI_ADDR_START_x & < PERI_ADDR_END_x**

1E002A34 GDMA_PERI_ADDR_END_0 Peripheral Region 0 End Address 20000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PERI_ADDR_END_0[31:16]															
Type	RW															
Reset	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PERI_ADDR_END_0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PERI_ADDR_END_0	**GDMA request will direct to peripheral bus if the request address >= PERI_ADDR_START_x & < PERI_ADDR_END_x**

1E002A38 GDMA_PERI_ADDR_START_1 Peripheral Region 1 Starting Address 1C000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PERI_ADDR_START_1[31:16]															
Type	RW															
Reset	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PERI_ADDR_START_1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PERI_ADDR_START_1	**GDMA request will direct to peripheral bus if the request address >= PERI_ADDR_START_x & < PERI_ADDR_END_x**

1E002A3C GDMA_PERI_ADDR_END_1 Peripheral Region 1 End Address 20000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PERI_ADDR_END_1[31:16]															
Type	RW															
Reset	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PERI_ADDR_END_1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PERI_ADDR_END_1	**GDMA request will direct to peripheral bus if the request address >= PERI_ADDR_START_x & < PERI_ADDR_END_x**

1E002A40 GDMA_PERI_ADDR_START_2 Peripheral Region 2 Starting Address 60000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PERI_ADDR_START_2[31:16]															
Type	RW															
Reset	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PERI_ADDR_START_2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PERI_ADDR_START_2	**GDMA request will direct to peripheral bus if the request address >= PERI_ADDR_START_x & < PERI_ADDR_END_x**

1E002A44 GDMA_PERI_ADDR_END_2 Peripheral Region 2 End Address 70000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PERI_ADDR_END_2[31:16]															
Type	RW															
Reset	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PERI_ADDR_END_2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PERI_ADDR_END_2	**GDMA request will direct to peripheral bus if the request address >= PERI_ADDR_START_x & < PERI_ADDR_END_x**

1E002A48 GDMA_PERI_ADDR_START_3 Peripheral Region 3 Starting Address 60000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PERI_ADDR_START_3[31:16]															
Type	RW															
Reset	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PERI_ADDR_START_3[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PERI_ADDR_START_3	**GDMA request will direct to peripheral bus if the request address >= PERI_ADDR_START_x & < PERI_ADDR_END_x**

1E002A4C GDMA_PERI_ADDR_END_3 Peripheral Region 3 End Address 70000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PERI_ADDR_END_3[31:16]															
Type	RW															
Reset	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PERI_ADDR_END_3[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PERI_ADDR_END_3	**GDMA request will direct to peripheral bus if the request address >= PERI_ADDR_START_x & < PERI_ADDR_END_x**

2.12 SPI Controller

2.12.1 Features

- Supports up to 2 SPI master operations
- Programmable clock polarity
- Programmable interface clock rate
- Programmable bit ordering
- Firmware-controlled SPI enable
- Programmable payload (address + data) length
- Supports 1/2/4 multi-IO SPI flash memory
- Supports command/user mode operation
- Supports SPI direct access
- Extends the addressable range from 24 bits to 32 bits for memory size larger than 128 Mb.

2.12.2 Block Diagram

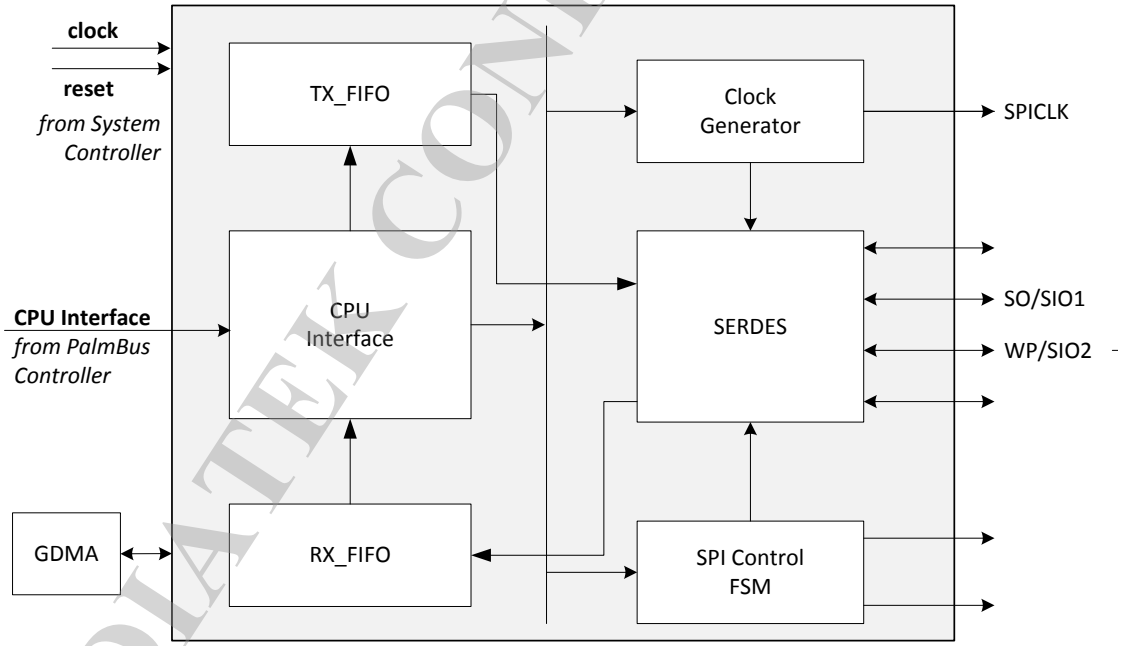


Figure 2-6 SPI Controller Block Diagram

2.12.3 Registers

SPI Changes LOG

Revision	Date	Author	Change Log
0.1	2012/8/29	Lancelot	Initialization
0.2	2012/11/6	Lancelot	1. Remove 0x38 SW_RST 2. Add CS_POLAR at 0x38
0.3	2012/11/23	Lancelot	Fix default value

Module name: SPI Base address: (+1E000B00h)

Address	Name	Width	Register Function
1E000B00	SPI_TRANS	32	SPI transaction control/status register
1E000B04	SPI_OP_ADDR	32	SPI opcode/address register
1E000B08	SPI_DIDO_0	32	SPI DI/DO data #0 register
1E000B0C	SPI_DIDO_1	32	SPI DI/DO data #1 register
1E000B10	SPI_DIDO_2	32	SPI DI/DO data #2 register
1E000B14	SPI_DIDO_3	32	SPI DI/DO data #3 register
1E000B18	SPI_DIDO_4	32	SPI DI/DO data #4 register
1E000B1C	SPI_DIDO_5	32	SPI DI/DO data #5 register
1E000B20	SPI_DIDO_6	32	SPI DI/DO data #6 register
1E000B24	SPI_DIDO_7	32	SPI DI/DO data #7 register
1E000B28	SPI_MASTER	32	SPI master mode register
1E000B2C	SPI_MORE_BUF	32	SPI more buf control register
1E000B30	SPI_QUEUE_CTL	32	SPI flash queue control register
1E000B34	SPI_STATUS	32	SPI controller status register
1E000B38	SPI_CS_POLAR	32	SPI chip select polarity
1E000B3C	SPI_SPACE	32	SPI flash space control register

1E000B00 SPI_TRANS SPI transaction control/status register 00160001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	spi_addr_ext								Reserved0			spi_addr_size		Reserved1		spi_master_busy
Type	RW								RO			RW		RO		RO
Reset	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved2							spi_master_start	miso_byte_cnt				mosi_byte_cnt			
Type	RO							WO	RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
31:24	spi_addr_ext	**SPI address extention** Address extenstion for 32-bit SPI address size. Usually this field specifies the first byte of the address phase to transmit to SPI device when more_buf_mode = 0 and spi_addr_size = 3. And spi_addr[31:24], spi_addr[23:16], and spi_addr[15:0] are respectively the second, third and fourth byte of the address phase
20:19	spi_addr_size	**SPI address size.** 0: reserved. 1: spi_addr[15:0] of SPI DI data register are valid (16-bit size). 2: spi_addr[23:0] of SPI DI data register are valid (24-bit size). 3: {spi_addr_ext[7:0], spi_addr[23:0]} of SPI DI data register are valid (32-bit size) Note: The spi_addr_size is valid only when more_buf_mode = 0.
16	spi_master_busy	**Transaction busy indication (Read-only). Writes to this bit are ignored.** 0: No SPI transaction is ongoing. Software may start a new SPI transaction by writing to the SPI transaction start bit within this register. 1: An SPI transaction presently is underway. Software must not try to start a new SPI transaction. Software may not alter the value of any field of the SPI master control registers.
8	spi_master_start	**SPI transaction start. Only writes to this field are meaningful, reads always return 0.** Writes: 0: No effect 1: Starts SPI transaction.
7:4	miso_byte_cnt	**SPI MISO (rx) byte count.** Determines the number of bytes received from the SPI device from the SPI opcode/address register and the SPI DI/DO data #0 register. Values of 0 ~ 8 are valid, other values are illegal. Note: The miso_byte_cnt is valid only when more_buf_mode = 0.
3:0	mosi_byte_cnt	**SPI MOSI (tx) byte count.** Determines the number of bytes transmitted from the SPI opcode/address register and the SPI DI/DO data #0 register to the SPI device. Values of 1 ~ 8 are valid, other values are illegal. Note: The mosi_byte_cnt is valid only when more_buf_mode = 0. The transmitted data sequence is as follows: spi_opcode, spi_addr (conditional) and d0_byte ~ d3_byte (conditional).

1E000B04 SPI_OP_ADDR SPI opcode/address register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	spi_addr[23:8]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	spi_addr[7:0]								spi_opcode							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:8	spi_addr	**SPI address. Usually this field specifies the 24-bits address to transmit to the SPI device when more_buf_mode = 0.** 1: (16-bits SPI address size), spi_addr[23:16] is the 1st byte of the address phase and spi_addr[15:8] is the 2nd byte of the address phase. 2: (24-bits SPI address size), spi_addr[31:24] is the 1st byte of the address phase and spi_addr[23:16] is the 2nd byte of the address phase and spi_addr[15:8] is the 3rd byte of the address phase. 3: (32-bits SPI address size), spi_addr[31:24] is the 2nd byte of the address phase and

		spi_addr[23:16] is the 3rd byte of the address phase and spi_addr[15:8] is the 4th byte of the address phase Note: For SPI read transaction and more_buf_mode = 0 Field [15:8] is also used to store the 6-th byte of data read phase. Field [23:16] is also used to store the 7-th byte of data read phase. Field [31:24] is also used to store the 8-th byte of data read phase.
7:0	spi_opcode	**SPI opcode. Usually this field specifies the 8-bits opcode (instruction) to transmit to the SPI device as the first byte of a SPI transaction when more_buf_mode = 0.** Note: For SPI read transaction and more_buf_mode = 0, this byte is also used to store the 5-th byte of data read phase according to the rx byte count miso_byte_cnt.

1E000B08 SPI_DIDO_0 SPI DI/DO data #0 register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	d3_byte								d2_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	d1_byte								d0_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	d3_byte	**The 4th data byte of data read/write phase.**
23:16	d2_byte	**The 3th data byte of data read/write phase.**
15:8	d1_byte	**The 2nd data byte of data read/write phase.**
7:0	d0_byte	**The 1st data byte of data read/write phase.**

1E000B0C SPI_DIDO_1 SPI DI/DO data #1 register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	d3_byte								d2_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	d1_byte								d0_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	d3_byte	**The 4th data byte of data read/write phase.**
23:16	d2_byte	**The 3th data byte of data read/write phase.**
15:8	d1_byte	**The 2nd data byte of data read/write phase.**
7:0	d0_byte	**The 1st data byte of data read/write phase.**

1E000B10 SPI_DIDO_2 SPI DI/DO data #2 register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	d3_byte								d2_byte							

Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	d1_byte								d0_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	d3_byte	**The 4th data byte of data read/write phase.**
23:16	d2_byte	**The 3th data byte of data read/write phase.**
15:8	d1_byte	**The 2nd data byte of data read/write phase.**
7:0	d0_byte	**The 1st data byte of data read/write phase.**

1E000B14 SPI_DIDO_3 SPI DI/DO data #3 register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	d3_byte								d2_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	d1_byte								d0_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	d3_byte	**The 4th data byte of data read/write phase.**
23:16	d2_byte	**The 3th data byte of data read/write phase.**
15:8	d1_byte	**The 2nd data byte of data read/write phase.**
7:0	d0_byte	**The 1st data byte of data read/write phase.**

1E000B18 SPI_DIDO_4 SPI DI/DO data #4 register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	d3_byte								d2_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	d1_byte								d0_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	d3_byte	**The 4th data byte of data read/write phase.**
23:16	d2_byte	**The 3th data byte of data read/write phase.**
15:8	d1_byte	**The 2nd data byte of data read/write phase.**
7:0	d0_byte	**The 1st data byte of data read/write phase.**

1E000B1C SPI_DIDO_5 SPI DI/DO data #5 register 0000000

0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	d3_byte								d2_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	d1_byte								d0_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	d3_byte	**The 4th data byte of data read/write phase.**
23:16	d2_byte	**The 3th data byte of data read/write phase.**
15:8	d1_byte	**The 2nd data byte of data read/write phase.**
7:0	d0_byte	**The 1st data byte of data read/write phase.**

1E000B20 SPI_DIDO_6 SPI DI/DO data #6 register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	d3_byte								d2_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	d1_byte								d0_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	d3_byte	**The 4th data byte of data read/write phase.**
23:16	d2_byte	**The 3th data byte of data read/write phase.**
15:8	d1_byte	**The 2nd data byte of data read/write phase.**
7:0	d0_byte	**The 1st data byte of data read/write phase.**

1E000B24 SPI_DIDO_7 SPI DI/DO data #7 register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	d3_byte								d2_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	d1_byte								d0_byte							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	d3_byte	**The 4th data byte of data read/write phase.**
23:16	d2_byte	**The 3th data byte of data read/write phase.**
15:8	d1_byte	**The 2nd data byte of data read/write phase.**
7:0	d0_byte	**The 1st data byte of data read/write phase.**

1E000B28 SPI_MASTER SPI master mode register 000D8880

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	rs_slave_sel			clk_mode	rs_clk_sel											
Type	RW			RW	RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	cs_dsel_cnt					full_duplex	int_en	spi_start_sel	spi_prefetch	bidir_mode	cpha	cpol	lsb_first	more_buf_mode	serial_mode	
Type	RW					RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	
Reset	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:29	rs_slave_sel	**select SPI device** 0: select SPI device 0 (default is flash) 1: select SPI device 1 ... 7: select SPI device 7
28	clk_mode	**This register is used to specify that period of SCLK HIGH is longer or period of SCLK LOW is longer when clock divisor(clk_sel) is odd.** 0: period of SCLK LOW is longer. 1: period of SCLL HIGH is longer.
27:16	rs_clk_sel	**Register Space SPI clock frequency select.** 0: SPI clock frequency is hclk/2. (50% duty cycle, duty cycle is the ratio of the output high time to the total cycle time) 1: SPI clock frequency is hclk/3. (33.33% or 66.67% duty cycle) 2: SPI clock frequency is hclk/4. (50% duty cycle) 3: SPI clock frequency is hclk/5. (40% or 60% duty cycle) 4095: SPI clock frequency is hclk/4097.
15:11	cs_dsel_cnt	**De-select time of SPI chip select is configured to occupy the number of cycles of AHB clock**
10	full_duplex	**Full duplex or half duplex mode.** 0: half duplex mode. 1: full duplex mode. Full duplex timing diagram Note: The full_duplex is valid only when more_buf_mode = 1. The transmission is always as half duplex when more_buf_mode = 0;
9	int_en	**Interrupt enable.** 0: disable SPI interrupt. 1: enable SPI interrupt.
8	spi_start_sel	**The interval between spi_cs_n and spi_sclk.** 0: 3 clk 1: 6 clk
7	spi_prefetch	**SPI pre-fetch buffer enable** 0: disable pre-fetch buffer. 1: enable pre-fetch buffer.
6	bidir_mode	**Bi-direction mode. In this mode, the SPI uses only one serial data pin for interface with external devices. The MOSI pin becomes the serial data I/O pin for the SPI transaction and MISO pin is not used. Bi-direction mode is used for the application with only 1 bi-direction serial pin for SPI transaction.** 0: normal mode (both MOSI and MISO pins are used).

		1: bi-direction mode (only MOSI pin is used). SPI host controller must operate in half duplex mode if bidir_mode = 1. Note: The bidir_mode is valid only when more_buf_mode = 1.
5	cpha	**(CPHA, clock phase). Initial SPI clock phase for SPI transaction.** There are four SPI modes used to latch data. These SPI modes latch data in one of four ways, and are defined by the logic state combinations of the CLK Polarity (CPOL) in relation to the CLK Phase (CPHA). The valid logic combinations identify and determine the SPI modes supported by the SPI device. SPI mode At CPOL=0 the base value of the clock is zero For CPHA=0 (mode 0), data is read on the clock's rising edge and data is changed on a falling edge. For CPHA=1 (mode 1), data is read on the clock's falling edge and data is changed on a rising edge. At CPOL=1 the base value of the clock is one (inversion of CPOL=0) For CPHA=0 (mode 2), data is read on clock's falling edge and data is changed on a rising edge. For CPHA=1 (mode 3), data is read on clock's rising edge and data is changed on a falling edge.
4	cpol	**cpol (CPOL, clock polarity). Initial SPI clock polarity for SPI transaction.**
3	lsb_first	**0: MSB(most significant bit) is transferred first for SPI transaction.** 1: LSB(least significant bit) is transferred first for SPI transaction.
2	more_buf_mode	**Select 2 words buffer or 8 words buffer for SPI transaction.** 0: SPI transfer data buffer size is only 2 words. In this mode, SPI DI/DO data #0 register and SPI opcode/address register are the data buffer for SPI transaction. And, SPI master follows mosi_byte_cnt and miso_byte_cnt to complete the transmission and reception, respectively. This kind of transaction must operate in half duplex mode. 1: SPI transfer data buffer size is 8 words. In this mode, SPI opcode/address register are the data buffer for SPI transaction and follows cmd_bit_cnt to complete the transaction. SPI DI/DO data #0~#7 register are the data buffer for SPI transaction and follows do_bit_cnt and di_bit_cnt to complete the transmission and reception, respectively. In half duplex mode, transmitted data are loaded from SPI opcode/address register and SPI DI/DO data #0~#7 registers. And, the received data will overwrite the SPI DI/DO data #0~#7 registers. In full duplex mode, SPI DI/DO data #0~#3 registers are used for transmission and SPI DI/DO #4~#7 registers are used for receipt.
1:0	serial_mode	**This mode is designed for Winbond SPI flash W25Q80/16/32 and W25X10/20/40/80/16/32/64 series.** 0: standard serial. 1: dual serial. 2: quad serial. 3: reserved. Note: The serial_mode is valid only when more_buf_mode = 0. The transaction mode is always as standard serial when more_buf_mode = 1.

1E000B2C SPI_MORE_BUF SPI more buf control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	Reserved0		cmd_bit_cnt						Reserved1			miso_bit_cnt[8:4]				
Type	RO		RW						RO			RW				
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	miso_bit_cnt[3:0]				Reserved2			mosi_bit_cnt								
Type	RW				RO			RW								
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

29:24	cmd_bit_cnt	**SPI command phase MOSI (tx) bit count. Determines the number of command bits transmitted from the SPI opcode/address register to the SPI device. Values of 0 ~ 32 are valid, but other values are illegal.** Note: The cmd_bit_cnt is valid only when more_buf_mode = 1 and the SPI opcode/address register is treated as a command register.
20:12	miso_bit_cnt	**SPI data phase MISO (rx) bit count. Determines the number of bits received from the SPI device into the SPI DI/DO data #0~#7 register. Values of 0 ~ 256 are valid, but other values are illegal. Maximum value is 256 for half duplex mode and 128 for full duplex mode. Please note that do_bit_cnt must be equal to di_bit_cnt in full duplex mode.** Note: The miso_bit_cnt is valid only when more_buf_mode = 1.
8:0	mosi_bit_cnt	**SPI data phase MOSI (tx) bit count. Determines the number of data bits transmitted from the SPI DI/DO data #0~#7 register to the SPI device. Values of 0 ~ 256 are valid, but other values are illegal. Maximum value is 256 for half duplex mode and 128 for full duplex mode.** Note: The mosi_bit_cnt is valid only when more_buf_mode = 1.

1E000B30 SPI_QUEUE_CTL SPI flash queue control register 00000A40

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	fs_page_sel						Reserved0[12:3]									
Type	RW						RO									
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved0[2:0]			fs_busy	fs_addr_size_r		fs_addr_size		fs_di_ph_byc				Reserved1	fast_spi_sel		
Type	RO			RO	RO		RW		RW				RO	RW		
Reset	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0

Bit(s)	Name	Description
31:26	fs_page_sel	**Flash Space Page Selection.** 0: (Page 0 space) 0x0000_0000 - 0x03ff_ffff 1: (Page 1 space) 0x0400_0000 - 0x07ff_ffff ... 63: (Page 63 space) 0xffc0_0000 - 0xffff_ffff
12	fs_busy	**Transaction busy indication (Read-only) in flash space. Writes to this bit are ignored.** 0: No SPI flash space access is ongoing. Software may change the configuration related to flash space. 1: SPI flash space access presently is underway. Software may not alter the configuration related to flash space.
11:10	fs_addr_size_r	**Latched fs_addr_size indication from internal spimc logic**
9:8	fs_addr_size	**SPI address. This field specifies the 24-bits/16-bits address to transmit to the SPI device for SPI Flash Space Read operation only.** 0: 25-bit SPI address size 1: 16-bit SPI address size Reserved. 2: 24-bit SPI address size (default for 3B SPI flash) 3: 26-bit SPI address size (default for 4B SPI flash) If the change of the fs_addr_size is needed, the sequence below must be followed. Otherwise, the new fs_addr_size configuration will not be updated to the internal spimc logic . Step 1: Set new fs_addr_size. Step 2: Transmit mode change command (ex. En4B or Ex4B of MX25L25635E) Note: 1. The value fs_addr_size is not valid in Register Space. 2. The Spimc now only supports 3-Byte mode (24 bits) and 4-Byte mode (25 or 26 bits) switch.

7:4	fs_di_ph_byc	**Determines the number of data bytes transmitted from the SPI master controller to the SPI device for SPI Flash Space Read operation. This field is similar to mosi_byte_cnt in STCSR but is used for setting of flash space access control path.** Note: this field should (if fs_addr_size_r = 2, 24-bit fs_addr_size) = 4 (OP + ADDR) if fast_spi_sel = 0 (0x03) = 5 (OP + ADDR + dummy) if fast_spi_sel = 1 (0x0b) = 5 (OP + ADDR + dummy) if fast_spi_sel = 2 (0x3b) = 5 (OP + ADDR + M7-0) if fast_spi_sel = 3 (0xbb) = 5 (OP + ADDR + dummy) if fast_spi_sel = 4 (0x6b) = 7 (OP + ADDR + M7-0 + dummy) if fast_spi_sel = 5 (0xeb) = 5 (OP + ADDR + M7-0) if fast_spi_sel = 6 (0xe3) (if fs_addr_size_r = 0 or 3, 25 or 26-bit fs_addr_size) = 5 (OP + ADDR) if fast_spi_sel = 0 (0x03) = 6 (OP + ADDR + dummy) if fast_spi_sel = 1 (0x0b) = 6 (OP + ADDR + dummy) if fast_spi_sel = 2 (0x3b) = 6 (OP + ADDR + M7-0) if fast_spi_sel = 3 (0xbb) = 6 (OP + ADDR + dummy) if fast_spi_sel = 4 (0x6b) = 8 (OP + ADDR + M7-0 + dummy) if fast_spi_sel = 5 (0xeb) = 6 (OP + ADDR + M7-0) if fast_spi_sel = 6 (0xe3)
2:0	fast_spi_sel	**Select SPI flash read instruction for Flash Space** 0: standard read data instruction (0x03). 1: standard fast read data instruction (0x0b). 2: fast read dual output instruction defined in Winbond W25Qxx series SPI flash (0x03b). 3: fast read dual I/O instruction defined in Winbond W25Qxx series SPI flash (0xbb). 4: fast read quad output instruction defined in Winbond W25Qxx series SPI flash (0x6b). 5: fast read quad I/O instruction defined in Winbond W25Qxx series SPI flash (0xeb). 6: burst read quad I/O instruction defined in Winbond W25Qxx series SPI flash (0xe3). Note: serial_mode and more_buf_mode are don't care for this flash space access control path.

1E000B34 SPI_STATUS SPI controller status register 00000030

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	Reserved0[25:10]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved0[9:0]										spi_flash_mode		Reserved1			spi_ok
Type	RO										RO		RO			RC
Reset	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0

Bit(s)	Name	Description
5:4	spi_flash_mode	**0: no SPI flash.** 1: standard SPI flash. 2: specific SPI flash with dual interface capability. 3: specific SPI flash with quad interface capability.
0	spi_ok	**When SPI transaction complete, SPI master controller will set this bit and assert SPI interrupt to notify software. Reading this register will clear this bit and de-assert SPI interrupt.**

1E000B38 SPI_CS_POL SPI chip select polarity 0000000

AR 0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	Reserved[23:8]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved[7:0]								cs_polar							
Type	RO								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7:0	cs_polar	**Chip select default polarity** set cs_polar[n]=1'b0 for cs[n] low active (SPI Flash) set cs_polar[n]=1'b1 for cs[n] high active

1E000B3C SPI_SPACE SPI flash space control register 00000030

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	Reserved[16:1]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved[0:0]	fs_slave_sel			fs_clk_sel											
Type	RO	RW			RW											
Reset	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0

Bit(s)	Name	Description
14:12	fs_slave_sel	**(Flash Space Slave Select)** 0: select SPI device #0. (default is flash) 1: select SPI device #1. ... 7: select SPI device #7.
11:0	fs_clk_sel	**Flash Space SPI clock frequency select.** 0: SPI clock frequency is hclk/2. (50% duty cycle, duty cycle is the ratio of the output high time to the total cycle time) 1: SPI clock frequency is hclk/3. (33.33% or 66.67% duty cycle) 2: SPI clock frequency is hclk/4. (50% duty cycle) 3: SPI clock frequency is hclk/5. (40% or 60% duty cycle) 4095: SPI clock frequency is hclk/4097.

2.13 I2S Controller

2.13.1 Features

- I2S transmitter/receiver, which can be configured as master or slave.
- Supports 16-bit data, sampling rates of 8 kHz, 16 kHz, 22.05 kHz, 44.1 kHz, and 48 kHz
- Support stereo audio data transfer.
- 32-byte FIFO are available for data transmission.
- Supports GDMA access
- Supports 12 Mhz bit clock from external source (when in slave mode)

2.13.2 Block Diagram

The I^2S transmitter block diagram is shown as below.

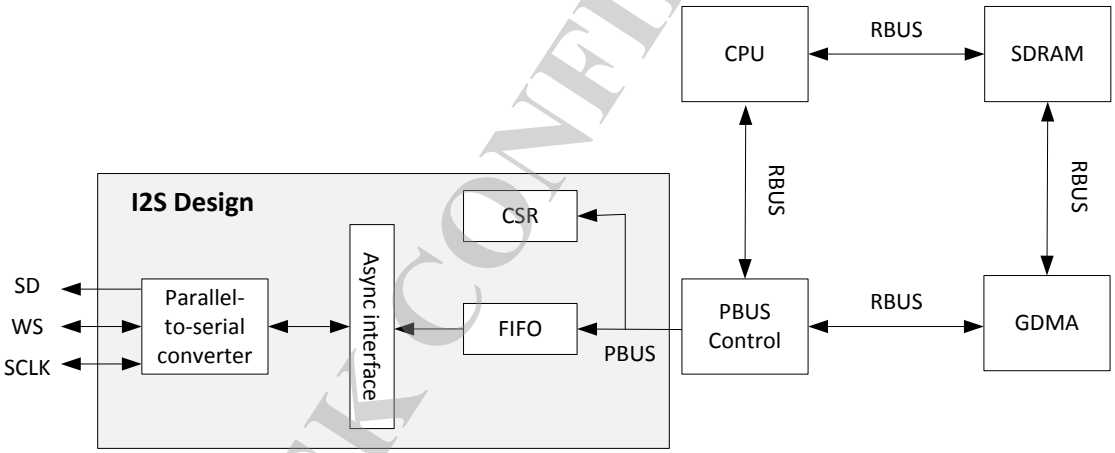


Figure 2-7 I^2S Transmitter Block Diagram

The I^2S interface consists of two separate cores, a transmitter and a receiver. Both can operate in either master or slave mode. The transmitter is only shown here in master or slave mode.

I^2S Signal Timing For I^2S Data Format

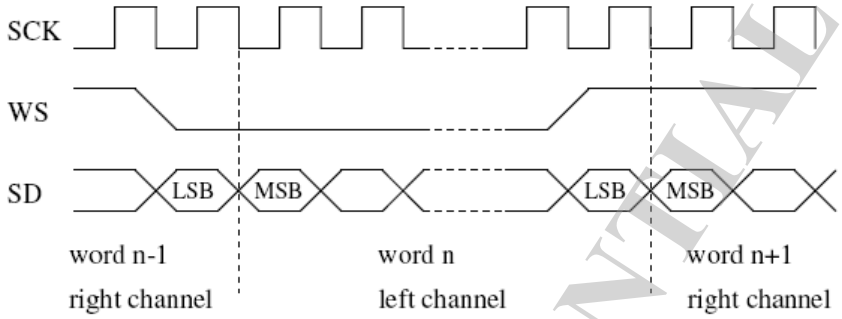


Figure 2-8 I2S Transmit/Receive

Serial data is transmitted in 2's complement with the MSB first. The transmitter always sends the MSB of the next word one clock period after the WS changes. Serial data sent by the transmitter may be synchronized with either the trailing (HIGH-to-LOW) or the leading (LOW-to-HIGH) edge of the clock signal. However, the serial data must be latched into the receiver on the leading edge of the serial clock signal, and so there are some restrictions when transmitting data that is synchronized with the leading edge.

The word select line indicates the channel being transmitted:

- WS = 0; channel 1 (left)
- WS = 1; channel 2 (right)

WS may change either on a trailing or leading edge of the serial clock, but it doesn't need to be symmetrical. In the slave, this signal is latched on the leading edge of the clock signal. The WS line changes one clock period before the MSB is transmitted. This allows the slave transmitter to derive synchronous timing of the serial data that will be set up for transmission. Furthermore, it enables the receiver to store the previous word and clear the input for the next Word.

2.13.3 Registers

Address	Name	Width	Register Function
1E000A00	I2S_CFG	32	**I2S Configuration** I2S Tx/Rx Configuration Register
1E000A04	INT_STATUS	32	**Interrupt Status** I2S Interrupt Status
1E000A08	INT_EN	32	**Interrupt Enable** I2S Interrupt Enable Control Register
1E000A0C	FF_STATUS	32	**FIFO Status** I2S Tx/Rx FIFO Status
1E000A10	TX_FIFO_WREG	32	**Transmit FIFO Write to Register** Tx Write Data Buffer
1E000A14	RX_FIFO_RREG	32	**Receive FIFO Read Register** DRAM PAD CONTROL 3
1E000A18	I2S_CFG1	32	**I2S Configuration 1** I2S Loopback Test Control Register
1E000A20	DIVCOMP_CFG	32	**Integer Part of the Dividor Register 1** Integer Part of the Dividor Register
1E000A28	DIVINT_CFG	32	**Integer Part of the Dividor Register 2** Integer Part of the Dividor Register

1E000A00 I2S_CFG I2S Configuration 00014040

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	I2S_EN	DMA_EN		BYTE_SWAP				TX_EN				RX_EN				SLAVE_MODE
Type	RW	RW		RW				RW				RW				RW
Reset	0	0		0				0				0				1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		RX_FF_THRES								TX_FF_THRES						
Type		RW								RW						
Reset		1	0	0						1	0	0				

Bit(s)	Name	Description
31	I2S_EN	**I2S Enable** Enables I2S. When disabled, all I2S control registers are cleared to their initial values. 0: Disable 1: Enable
30	DMA_EN	**DMA Enable** Enables DMA access. 0: Disable 1: Enable
28	BYTE_SWAP	**Swaps the order of data bytes in each 16-bit channel.** 0: No data swap 1: Data byte swap
24	TX_EN	**Transmitter on/off control** 0: Disable

		1: Enable
20	RX_EN	**Receiver on/off control** 0: Disable 1: Enable
16	SLAVE_MODE	**Sets master or slave mode.** 0: Master: using internal clock 1: Slave: using external clock
14:12	RX_FF_THRES	**Rx FIFO Threshold** When the threshold is reached, the host/DMA is notified to fill FIFO. 2<RX_FF_THRES<6 (unit: word)
6:4	TX_FF_THRES	**Tx FIFO Threshold** When the threshold is reached, the host/DMA is notified to fill FIFO. 2<TX_FF_THRES<6 (unit: word)

1E000A04 INT_STATUS Interrupt Status 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									RX_DMA_FAULT	RX_OVRUN	RX_UNRUN	RX_THRES	TX_DMA_FAULT	TX_OVRUN	TX_UNRUN	TX_THRES
Type									RW	RW	RW	RW	RW	RW	RW	RW
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7	RX_DMA_FAULT	**Rx DMA Fault Detected Interrupt** Asserts when a fault is detected in Rx DMA signals.
6	RX_OVRUN	**Rx Overrun Interrupt** Asserts when the Rx FIFO is overrun.
5	RX_UNRUN	**Rx Underrun Interrupt** Asserts when the Rx FIFO is underrun.
4	RX_THRES	**Rx FIFO Below Threshold Interrupt** Asserts when the Rx FIFO is lower than the defined threshold.
3	TX_DMA_FAULT	**Tx DMA Fault Detected Interrupt** Asserts when a fault is detected in Tx DMA signals.
2	TX_OVRUN	**Tx FIFO Overrun Interrupt** Asserts when the Tx FIFO is overrun.
1	TX_UNRUN	**Tx FIFO Underrun Interrupt** Asserts when the Tx FIFO is underrun.
0	TX_THRES	**Tx FIFO Below Threshold Interrupt** Asserts when the FIFO is lower than the defined threshold.

1E000A08 INT_EN Interrupt Enable 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									RX_INT3_EN	RX_INT2_EN	RX_INT1_EN	RX_INT0_EN	TX_INT3_EN	TX_INT2_EN	TX_INT1_EN	TX_INT0_EN
Type									RW	RW	RW	RW	RW	RW	RW	RW
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7	RX_INT3_EN	**INT_STATUS[7] Enable** Enables the Rx DMA Fault Detected Interrupt. This interrupt asserts when a fault is detected in Rx DMA signals.
6	RX_INT2_EN	**INT_STATUS[6] Enable** Enables the Rx Overrun Interrupt. This interrupt asserts when the Rx FIFO is overrun.
5	RX_INT1_EN	**INT_STATUS[5] Enable** Enables the Rx Underrun Interrupt. This interrupt asserts when the Rx FIFO is underrun.
4	RX_INT0_EN	**INT_STATUS[4] Enable** Enables the Rx FIFO Below Threshold Interrupt. This interrupt asserts when the Rx FIFO is lower than the defined threshold.
3	TX_INT3_EN	**INT_STATUS[3] Enable** Enables the Tx DMA Fault Detected Interrupt. This interrupt asserts when a fault is detected in Tx DMA signals.
2	TX_INT2_EN	**INT_STATUS[2] Enable** Enables the Tx FIFO Overrun Interrupt. This interrupt asserts when the Tx FIFO is overrun.
1	TX_INT1_EN	**INT_STATUS[1] Enable** Enables the Tx FIFO Underrun Interrupt. This interrupt asserts when the Tx FIFO is underrun.
0	TX_INT0_EN	**INT_STATUS[0] Enable** Enables the Tx FIFO Below Threshold Interrupt. This interrupt asserts when the FIFO is lower than the defined threshold.

1E000A0C FF_STATUS FIFO Status 00000008

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									RX_AVCNT				TX_EPCNT			
Type									RO				RO			
Reset									0	0	0	0	1	0	0	0

Bit(s)	Name	Description
7:4	RX_AVCNT	**Rx FIFO Available Space Count** Counts the available space for reads in Rx FIFO. (unit: word)
3:0	TX_EPCNT	**Tx FIFO Available Space Count** Counts the available space for writes in Tx FIFO.

(unit: word)

1E000A10 TX_FIFO_WREG Transmit FIFO Write to Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_FIFO_WDATA[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_FIFO_WDATA[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_FIFO_WDATA	**Tx FIFO Write Data Buffer** Buffers data to be written to the Tx FIFO.

1E000A14 RX_FIFO_RREG Receive FIFO Read Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RX_FIFO_RDATA[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RX_FIFO_RDATA[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	RX_FIFO_RDATA	**Rx FIFO Read Data Buffer** Buffers data read from the Rx FIFO.

1E000A18 I2S_CFG1 I2S Configuration 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	LBK_EN	EXT_LBK_EN														
Type	RW	RW														
Reset	0	0														
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Name	Description
31	LBK_EN	**Enables loopback mode.** 0: Normal mode 1: Loopback mode

		ASYNC_TXFIFIO -> Tx -> Rx -> ASYNC_RXFIFIO
30	EXT_LBK_EN	**Enables external loopback.** 0: Normal mode 1: Enables external loop back. External A/D -> Rx -> Tx -> External D/A

1E000A20 DIVCOMP_CFG Integer Part of the Divider Register 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CLK_EN															
Type	RW															
Reset	0															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								DIVCOMP								
Type								RW								
Reset								0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	CLK_EN	**Enables setting of the I2S clock based on DIVCOMP and DIVINT parameters.** 0: Disable 1: Enable
8:0	DIVCOMP	**A parameter in an equation which determines FREQOUT. See DIVINT_CFG.**

1E000A28 DIVINT_CFG Integer Part of the Divider Register 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name							DIVINT									
Type							RW									
Reset							0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
9:0	DIVINT	**Integer Divider** A parameter in an equation which determines FREQOUT: FREQOUT = FREQIN *(1/2) * {1 / [DIVINT+DIVCOMP/(512)]} FREQIN is always fixed to 40 MHz.

2.14 SPDIF TX

2.14.1 Registers

SPDIFTX Changes LOG

Revision	Date	Author	Change Log
1.0	20120825	Jiechao Wei	Initial Revision by RegisterMap_v1p4
1.1	20120919	Jiechao Wei	Update for final RISC(PBUS) interface
1.2	20121009	Jiechao Wei	Update default value
1.3	20121216	Jiechao Wei	Update for DRAM ping-pong structure

Module name: SPDIFTX Base address: (+1E000700h)

Address	Name	Width	Register Function
1E000700	IEC_CTRL	32	IEC CONTROL REGISTER
1E000704	IEC_BUF0_BS_SBLK	32	IEC BITSTREAM BUFFER START BLOCK
1E000708	IEC_BUF0_BS_EBLK	32	IEC BITSTREAM BUFFER END BLOCK
1E00070C	IEC_BUF0_NSADR	32	IEC NEXT BURST START ADDRESS
1E000710	IEC_BUF0_NEXT_UADR	32	IEC USER DATA NEXT START ADDRESS
1E000714	IEC_BUF0_INTR_NSNUM	32	IEC INTERRUPT SIZE
1E000718	IEC_BUF0_PCPD_PACK	32	IEC NEXT BURST LENGTH
1E00071C	IEC_BUF0_CH_CFG_TRIG	32	IEC CHANNEL CONFIGURATION TRIGGER
1E000720	IEC_BUF0_CH_CFG0	32	IEC CHANNEL CONFIGURATION 0
1E000724	IEC_BUF0_CH_CFG1	32	IEC CHANNEL CONFIGURATION 1
1E000728	IEC_BUF0_CH_CFG2	32	IEC CHANNEL CONFIGURATION 2
1E00072C	IEC_BUF0_CH_CFG3	32	IEC CHANNEL CONFIGURATION 3
1E000730	IEC_BUF0_CH_CFG4	32	IEC CHANNEL CONFIGURATION 4
1E000734	IEC_BUF0_CH_CFG5	32	IEC CHANNEL CONFIGURATION 5
1E000738	IEC_BUF0_CH_CFG6	32	IEC CHANNEL CONFIGURATION 6
1E00073C	IEC_ACLK_DIV	32	IEC AUDIO MASTER CLOCK DIVIDER
1E000740	IEC_APLL_CFG0	32	IEC AUDIO PLL CONFIGURATION 0
1E000744	IEC_APLL_CFG1	32	IEC AUDIO PLL CONFIGURATION 1
1E000748	IEC_APLL_CFG2	32	IEC AUDIO PLL CONFIGURATION 2
1E00074C	IEC_APLL_CFG3	32	IEC AUDIO PLL CONFIGURATION 3
1E000750	IEC_APLL_DEBUG	32	IEC AUDIO PLL DEBUG INFORMATION
1E000754	IEC_BUF1_BS_SBLK	32	IEC BITSTREAM BUFFER START BLOCK
1E000758	IEC_BUF1_BS_EB	32	IEC BITSTREAM BUFFER END BLOCK

	LK		
1E00075C	IEC_BUF1_NSADR	32	IEC NEXT BURST START ADDRESS
1E000760	IEC_BUF1_NEXT_UADR	32	IEC USER DATA NEXT START ADDRESS
1E000764	IEC_BUF1_INTR_NSNUM	32	IEC INTERRUPT SIZE
1E000768	IEC_BUF1_PCPD_PACK	32	IEC NEXT BURST LENGTH
1E00076C	IEC_BUF1_CH_CFG_TRIG	32	IEC CHANNEL CONFIGURATION TRIGGER
1E000770	IEC_BUF1_CH_CFG0	32	IEC CHANNEL CONFIGURATION 0
1E000774	IEC_BUF1_CH_CFG1	32	IEC CHANNEL CONFIGURATION 1
1E000778	IEC_BUF1_CH_CFG2	32	IEC CHANNEL CONFIGURATION 2
1E00077C	IEC_BUF1_CH_CFG3	32	IEC CHANNEL CONFIGURATION 3
1E000780	IEC_BUF1_CH_CFG4	32	IEC CHANNEL CONFIGURATION 4
1E000784	IEC_BUF1_CH_CFG5	32	IEC CHANNEL CONFIGURATION 5
1E000788	IEC_BUF1_CH_CFG6	32	IEC CHANNEL CONFIGURATION 6

1E000700 IEC_CTRL IEC CONTROL REGISTER 00000080

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RAW_EN															INTR_ENABLE
Type	RW															RW
Reset	0															0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	INTR_STATUS			DBUF_SEL	DBUF_DISABLE	DOWN_SAMPLE			MUTE_SPDF	BYTE_SWAP	RAW_SWAP	RAW_24	MUTE_SAMPLE	UDATA_EN	DATA_FMT	DATA_SRC
Type	W1C			RO	RW	RW			RW	RW	RW	RW	RW	RW	RW	RW
Reset	0			0	0	0	0	0	1	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	RAW_EN	**IEC958 raw data enable** 0: disable 1: enable
16	INTR_ENABLE	**IEC958 interrupt enable** 0: disable 1: enable
15	INTR_STATUS	**IEC958 interrupt status** 0: No interrupt 1: Interrupting

12	DBUF_SEL	**IEC958 DRAM ping-pong buffer indicator** 0: buffer0 is going 1: buffer1 is going
11	DBUF_DISABLE	**IEC958 DRAM ping-pong buffer disable** 0: enable 1: disable
10:8	DOWN_SAMPLE	**IEC958 down sample control** 0: no down sample (recommended for MT7621) 1: 2x down sample 3: 4x down sample
7	MUTE_SPDF	**mute IEC output SPDF signal** 0: normal 1: mute output SPDF signal
6	BYTE_SWAP	**IEC dram word data bytes switch mode**
5	RAW_SWAP	**IEC 24bit raw data bytes switch mode**
4	RAW_24	**IEC raw data 24bit mode**
3	MUTE_SAMPLE	**mute IEC output sample data** 0: normal 1: mute output sample data
2	UDATA_EN	**user data enable** 0: all user data are zero 1: load user data from DRAM (IEC_NEXT_UADR)
1	DATA_FMT	**output data format selection** 0: PCM data 1: encoded data
0	DATA_SRC	**data source selection** 0: cooked data (from PCM receiver or transmitter output) 1: raw data (from DRAM 61937 encoded audio data or 60958 plain PCM data)

1E000704 IEC_BUF0_BS_SBLK IEC BITSTREAM BUFFER START BLOCK 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					BS_SBLK[27:16]											
Type					RW											
Reset					0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BS_SBLK[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
27:0	BS_SBLK	**IEC958 bitstream buffer start block (double word size)**

1E000708 IEC_BUF0_BS_EBLK IEC BITSTREAM BUFFER END BLOCK 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					BS_EBLK[27:16]											
Type					RW											
Reset					0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BS_EBLK[15:0]															
Type	RW															

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
27:0	BS_EBLK	**IEC958 bitstream buffer end block (double word size)**

1E00070C IEC_BUF0_NSADR IEC NEXT BURST START ADDRESS 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			NSADR[29:16]													
Type			RW													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	NSADR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
29:0	NSADR	**next start address for next burst raw data (byte size)**

1E000710 IEC_BUF0_NEXT_UADR IEC USER DATA NEXT START ADDRESS 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			NUSADR[29:16]													
Type			RW													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	NUSADR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
29:0	NUSADR	**next start address for next user data, LSB 2 bits are ignored and considered zero (byte size)**

1E000714 IEC_BUF0_INTR_NSNUM IEC INTERRUPT SIZE 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				INTR_SIZE												
Type				RW												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				NSNUM												
Type				RW												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28:16	INTR_SIZE	**generating interrupt when how many samples remain for this burst**
12:0	NSNUM	**next sample number represented by next burst raw data**

1E000718 IEC_BUF0_PCPD_PACK IEC NEXT BURST LENGTH 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	NB_LEN															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BURST_INFO															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	NB_LEN	number of bits for next burst(Pd package)
15:0	BURST_INFO	burst information for IEC(Pc package)

1E00071C IEC_BUF0_CH_CFG_TRIG IEC CHANNEL CONFIGURATION TRIGGER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CH_CFG_TRIG								CH2_NUM							
Type	RW								RW							
Reset	0								0	0	0	0				
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Name	Description
31	CH_CFG_TRIG	channel status update trigger, write 1'b1 to trigger and read busy state or not
23:20	CH2_NUM	channel 2 (W) channel number

1E000720 IEC_BUF0_CH_CFG0 IEC CHANNEL CONFIGURATION 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CH_CFG0_RESERVE		CLK_ACCURACY		SAM_FREQ				CH1_NUM				SRC_NUM			
Type	RW		RW		RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CATEGORY								MODE		ADD_INFO			CP_RIGHT	DIGITAL	CONSUMER
Type	RW								RW		RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:30	CH_CFG0_RESERVE	channel status reserve bits

29:28	CLK_ACCURACY	**clock accuracy**
27:24	SAM_FREQ	**sampling frequency**
23:20	CH1_NUM	**channel 1 (B & M) channel number**
19:16	SRC_NUM	**source number**
15:8	CATEGORY	**category code**
7:6	MODE	**channel status mode 0**
5:3	ADD_INFO	**additional information**
2	CP_RIGHT	**copyright information**
1	DIGITAL	**digital (bit 1 of channel status)** 0: linear PCM samples 1: other purpose
0	CONSUMER	**bit 0 of channel status** 0: consumer use of channel status block 1: professional use of channel status block

1E000724 IEC_BUF0_CH_CFG1 IEC CHANNEL CONFIGURATION 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CH_CFG1_RESERVE[21:6]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CH_CFG1_RESERVE[5:0]						CGMS_A		ORIGINAL_FS				WORD_LEN			
Type	RW						RW		RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:10	CH_CFG1_RESERVE	**channel status reserve bits**
9:8	CGMS_A	**CGMS-A information**
7:4	ORIGINAL_FS	**original sample frequency**
3:0	WORD_LEN	**word length**

1E000728 IEC_BUF0_CH_CFG2 IEC CHANNEL CONFIGURATION 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CH_CFG2_RESERVE[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CH_CFG2_RESERVE[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	CH_CFG2_RESERVE	**channel status reserve bits**

1E00072C IEC_BUF0_CH IEC CHANNEL CONFIGURATION 3 0000000

_CFG3 0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CH_CFG3_RESERVE[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CH_CFG3_RESERVE[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	CH_CFG3_RESERVE	**channel status reserve bits**

1E000730 **IEC_BUF0_CH_CFG4** **IEC CHANNEL CONFIGURATION 4** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CH_CFG4_RESERVE[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CH_CFG4_RESERVE[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	CH_CFG4_RESERVE	**channel status reserve bits**

1E000734 **IEC_BUF0_CH_CFG5** **IEC CHANNEL CONFIGURATION 5** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CH_CFG5_RESERVE[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CH_CFG5_RESERVE[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	CH_CFG5_RESERVE	**channel status reserve bits**

1E000738 **IEC_BUF0_CH_CFG6** **IEC CHANNEL CONFIGURATION 6** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CH_CFG6_RESERVE[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CH_CFG6_RESERVE[15:0]															

Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	CH_CFG6_RESERVE	**channel status reserve bits**

1E00073C IEC_ACLK_DIV IEC AUDIO MASTER CLOCK DIVIDER 0018241F

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									MAS_DIV							
Type									RW							
Reset									0	0	0	1	1	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		IEC_DIV			BIT_DIV							LRC_DIV				
Type		RW			RW							RW				
Reset		0	1	0	0	1	0	0				1	1	1	1	1

Bit(s)	Name	Description
23:16	MAS_DIV	**audio master clock divider by audio pll (default 256xfs)**
14:12	IEC_DIV	**audio iec958 clock divider by audio master clock (default 128xfs)**
11:8	BIT_DIV	**audio bit clock divider by audio master clock (default 64xfs)**
4:0	LRC_DIV	**audio lrck divider by audio bit clock (default 1xfs)**

1E000740 IEC_APLL_CFG0 IEC AUDIO PLL CONFIGURATION 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	APLL_CFG0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	APLL_CFG0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	APLL_CFG0	**audio pll configuration register 0 (Need MT7621 update)**

1E000744 IEC_APLL_CFG1 IEC AUDIO PLL CONFIGURATION 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	APLL_CFG1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	APLL_CFG1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

31:0	APLL_CFG1	**audio pll configuration register 1 (Need MT7621 update)**

1E000748 IEC_APLL_CFG2 IEC AUDIO PLL CONFIGURATION 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	APLL_CFG2[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	APLL_CFG2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	APLL_CFG2	**audio pll configuration register 2 (Need MT7621 update)**

1E00074C IEC_APLL_CFG3 IEC AUDIO PLL CONFIGURATION 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	APLL_CFG3[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	APLL_CFG3[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	APLL_CFG3	**audio pll configuration register 3 (Need MT7621 update)**

1E000750 IEC_APLL_DEBUG IEC AUDIO PLL DEBUG INFORMATION 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									APLL_DEBUG							
Type									RO							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7:0	APLL_DEBUG	**audio pll debug information (Need MT7621 update)**

1E000754 IEC_BUF1_BS_SBLK IEC BITSTREAM BUFFER START BLOCK 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Name				BS_SBLK[28:16]												
Type				RW												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BS_SBLK[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28:0	BS_SBLK	**IEC958 bitstream buffer start block (double word size)**

1E000758 IEC_BUF1_BS_EBLK IEC BITSTREAM BUFFER END BLOCK 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				BS_EBLK[28:16]												
Type				RW												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BS_EBLK[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28:0	BS_EBLK	**IEC958 bitstream buffer end block (double word size)**

1E00075C IEC_BUF1_NSADR IEC NEXT BURST START ADDRESS 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		NSADR[30:16]														
Type		RW														
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	NSADR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
30:0	NSADR	**next start address for next burst raw data (byte size)**

1E000760 IEC_BUF1_NEXT_UADR IEC USER DATA NEXT START ADDRESS 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		NUSADR[30:16]														
Type		RW														
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	NUSADR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
30:0	NUSADR	next start address for next user data, LSB 2 bits are ignored and considered zero (byte size)

1E000764 IEC_BUF1_INTR_NSNUM IEC INTERRUPT SIZE 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				INTR_SIZE												
Type				RW												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				NSNUM												
Type				RW												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28:16	INTR_SIZE	generating interrupt when how many samples remain for this burst
12:0	NSNUM	next sample number represented by next burst raw data

1E000768 IEC_BUF1_PCPD_PACK IEC NEXT BURST LENGTH 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	NB_LEN															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BURST_INFO															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	NB_LEN	number of bits for next burst(Pd package)
15:0	BURST_INFO	burst information for IEC(Pc package)

1E00076C IEC_BUF1_CH_CFG_TRIG IEC CHANNEL CONFIGURATION TRIGGER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CH_CFG_TRIG								CH2_NUM							
Type	RW								RW							
Reset	0								0	0	0	0				
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Name	Description

31	CH_CFG_TRIG	**channel status update trigger, write 1'b1 to trigger and read busy state or not**
23:20	CH2_NUM	**channel 2 (W) channel number**

1E000770 IEC_BUF1_CH_CFG0 IEC CHANNEL CONFIGURATION 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CH_CFG0_RESERVE		CLK_ACCURACY		SAM_FREQ				CH1_NUM				SRC_NUM			
Type	RW		RW		RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CATEGORY								MODE		ADD_INFO			CP_RIGHT	DIGITAL	CONSUMER
Type	RW								RW		RW			RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:30	CH_CFG0_RESERVE	**channel status reserve bits**
29:28	CLK_ACCURACY	**clock accuracy**
27:24	SAM_FREQ	**sampling frequency**
23:20	CH1_NUM	**channel 1 (B & M) channel number**
19:16	SRC_NUM	**source number**
15:8	CATEGORY	**category code**
7:6	MODE	**channel status mode 0**
5:3	ADD_INFO	**additional information**
2	CP_RIGHT	**copyright information**
1	DIGITAL	**digital (bit 1 of channel status)** 0: linear PCM samples 1: other purpose
0	CONSUMER	**bit 0 of channel status** 0: consumer use of channel status block 1: professional use of channel status block

1E000774 IEC_BUF1_CH_CFG1 IEC CHANNEL CONFIGURATION 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CH_CFG1_RESERVE[21:6]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CH_CFG1_RESERVE[5:0]						CGMS_A		ORIGINAL_FS				WORD_LEN			
Type	RW						RW		RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:10	CH_CFG1_RESERVE	**channel status reserve bits**
9:8	CGMS_A	**CGMS-A information**
7:4	ORIGINAL_FS	**original sample frequency**

3:0	WORD_LEN	**word length**

1E000778 IEC_BUF1_CH_CFG2 IEC CHANNEL CONFIGURATION 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CH_CFG2_RESERVE[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CH_CFG2_RESERVE[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	CH_CFG2_RESERVE	**channel status reserve bits**

1E00077C IEC_BUF1_CH_CFG3 IEC CHANNEL CONFIGURATION 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CH_CFG3_RESERVE[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CH_CFG3_RESERVE[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	CH_CFG3_RESERVE	**channel status reserve bits**

1E000780 IEC_BUF1_CH_CFG4 IEC CHANNEL CONFIGURATION 4 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CH_CFG4_RESERVE[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CH_CFG4_RESERVE[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	CH_CFG4_RESERVE	**channel status reserve bits**

1E000784 IEC_BUF1_CH_CFG5 IEC CHANNEL CONFIGURATION 5 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Name	CH_CFG5_RESERVE[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CH_CFG5_RESERVE[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	CH_CFG5_RESERVE	**channel status reserve bits**

1E000788 IEC_BUF1_CH_CFG6 IEC CHANNEL CONFIGURATION 6 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CH_CFG6_RESERVE[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CH_CFG6_RESERVE[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	CH_CFG6_RESERVE	**channel status reserve bits**

2.15 Memory Controller

2.15.1 Features

- 1 SDRAM/DDR2 (16 b) chip selection
- 128 MB (SDRAM)/128 MB (DDR1)/256 MB (DDR2) per chip selection
- SDRAM transaction overlapping by early active and hidden pre-charge
- User SDRAM Init commands
- 4 banks per SDRAM chip select
- SDRAM burst length: 4 (fixed)
- DDR2 burst length: 4/8 (programmable)
- Wrap-4 transfer
- Bank-Raw-Column and Raw-Bank-Column address mapping

2.15.2 Registers

Address	Name	Width	Register Function
1E005000	ACTIM0	32	**DRAM AC TIMING SETTING 0** DRAM AC TIMING SETTING 0
1E005004	CONF1	32	**DRAM CONFIGURATION 1** DRAM CONFIGURATION 1
1E005008	CONF2	32	**DRAM CONFIGURATION 2** DRAM CONFIGURATION 2
1E00500C	PADCTL1	32	**DRAM PAD CONTROL 1** DRAM PAD CONTROL 1
1E005010	PADCTL2	32	**DRAM PAD CONTROL 2** DRAM PAD CONTROL 2
1E005014	PADCTL3	32	**DRAM PAD CONTROL 3** DRAM PAD CONTROL 3
1E005018	DELDLY1	32	**DQS INPUT DELAY CHAIN SETTING 1** DQS INPUT DELAY CHAIN SETTING 1
1E005020	DIFDLY1	32	**DQS INPUT DELAY CHAIN OFFSET SETTING 1** DQS INPUT DELAY CHAIN OFFSET SETTING 1
1E005028	DLLCONF	32	**DLL CONFIGURATION** DLL CONFIGURATION
1E00502C	TESTMODE	32	**TEST MODE CONFIGURATION 1** TEST MODE CONFIGURATION 1
1E00503C	TEST2_1	32	**TEST AGENT 2 CONFIGURATION 1** TEST AGENT 2 CONFIGURATION 1
1E005040	TEST2_2	32	**TEST AGENT 2 CONFIGURATION 2** TEST AGENT 2 CONFIGURATION 2
1E005044	TEST2_3	32	**TEST AGENT 2 CONFIGURATION 3** TEST AGENT 2 CONFIGURATION 3
1E005048	TEST2_4	32	**TEST AGENT 2 CONFIGURATION 4** TEST AGENT 2 CONFIGURATION 4
1E00507C	DDR2CTL	32	**DDR2 CONTROL REGISTER** DDR2 CONTROL REGISTER
1E005084	ZQCS	32	**ZQCS setting** ZQCS setting
1E005088	MRS	32	**MRS value setting** MRS value setting
1E00508C	CLK1DELAY	32	**Clock 1 output delay CONTROL** Clock 1 output delay CONTROL
1E005090	IOCTL	32	**IO CONTROL** IO misc control
1E005094	DQSIEN	32	**DQS INPUT RANGE FINE TUNER** DQS INPUT RANGE FINE TUNER
1E0050B8	DRVCTL0	32	**PAD DRIVING CONTROL SETTING 0** PAD DRIVING CONTROL SETTING 0
1E0050BC	DRVCTL1	32	**PAD DRIVING CONTROL SETTING 1** PAD DRIVING CONTROL SETTING 1

1E0050C0	**DLLSEL**	32	**DLL SELECTION SETTING** DLL SELECTION SETTING
1E0050CC	**TDSEL0**	32	**IO OUTPUT DUTY CONTROL 0** IO OUTPUT DUTY CONTROL 0
1E0050D0	**TDSEL1**	32	**IO OUTPUT DUTY CONTROL 1** IO OUTPUT DUTY CONTROL 1
1E0050D8	**MCKDLY**	32	**MEMORY CLOCK DELAY CHAIN SETTING** MEMORY CLOCK DELAY CHAIN SETTING
1E0050DC	**DQSCTL0**	32	**DQS INPUT RANGE CONTROL 0** DQS INPUT RANGE CONTROL 0
1E0050E0	**DQSCTL1**	32	**DQS INPUT RANGE CONTROL 1** DQS INPUT RANGE CONTROL 1
1E0050E4	**PADCTL4**	32	**PAD CONTROL 1** PAD CONTROL 4
1E0050E8	**PADCTL5**	32	**PAD CONTROL 2** PAD CONTROL 5
1E0050EC	**PADCTL6**	32	**PAD CONTROL 3** PAD CONTROL 6
1E0050F0	**PHYCTL1**	32	**DDR PHY CONTROL 1** DDR PHY CONTROL 1
1E0050F4	**GDDR3CTL1**	32	**GDDR3 CONTROL 1** GDDR3 CONTROL 1
1E0050F8	**PADCTL7**	32	**PAD CONTROL 4** PAD CONTROL 7
1E0050FC	**MISCTL0**	32	**MISC CONTROL 0** MISC CONTROL 0
1E005100	**OCDK**	32	**OCD CALIBRATION CONTROL** OCD CALIBRATION CONTROL
1E005104	**LBWDAT0**	32	**LOOP BACK DATA 0** LOOP BACK DATA 0
1E005108	**LBWDAT1**	32	**LOOP BACK DATA 1** LOOP BACK DATA 1
1E00510C	**LBWDAT2**	32	**LOOP BACK DATA 2** LOOP BACK DATA 2
1E005110	**RKCFG**	32	**RANK CONFIGURATION** RANK CONFIGURATION
1E005114	**CKPHDET**	32	**CLOCK PHASE DETECTION SETTING** CLOCK PHASE DETECTION SETTING
1E005124	**DQSGCTL**	32	**INPUT DQS GATING CONTROL** INPUT DQS GATING CONTROL
1E005130	**CLKENCTL**	32	**DRAM CLOCK ENABLE CONTROL** DRAM CLOCK ENABLE CONTROL
1E005140	**DQSGCTL1**	32	**DQS gating delay control 1** DQS gating delay control 1
1E005144	**DQSGCTL2**	32	**DQS gating delay control 2** DQS gating delay control 2
1E005168	**ARBCTL0**	32	**ARBITRATION CONTROL 0** ARBITRATION CONTROL 0
1E0051A8	**CMDDLY0**	32	**Command Delay CTL0** Command Delay CTL0

1E0051AC	**CMDDLY1**	32	**Command Delay CTL1** Command Delay CTL1
1E0051B0	**CMDDLY2**	32	**Command Delay CTL2** Command Delay CTL2
1E0051B4	**CMDDLY3**	32	**Command Delay CTL3** Command Delay CTL3
1E0051B8	**CMDDLY4**	32	**Command Delay CTL4** Command Delay CTL4
1E0051BC	**CMDDLY5**	32	**Command Delay CTL5** Command Delay CTL5
1E0051C0	**DQSCAL0**	32	**DQS CAL CONTROL 0** DQS CAL CONTROL 0
1E0051D8	**DMMonitor**	32	**Monitor parameter** Monitor parameter
1E0051DC	**DRAMC_PD_CTRL**	32	**PD mode parameter** PD mode parameter
1E0051E0	**LPDDR2**	32	**LPDDR2 setting** LPDDR2 setting
1E0051E4	**SPCMD**	32	**Special command mode** Special command mode
1E0051E8	**ACTIM1**	32	**DRAM AC TIMING SETTING 1** DRAM AC TIMING SETTING 1
1E0051EC	**PERFCTL0**	32	**PERFORMANCE CONTROL 0** PERFORMANCE CONTROL 0
1E0051F0	**AC_DERATING**	32	**AC TIME DERATING CONTROL** AC TIME DERATING CONTROL
1E0051F4	**RRRATE_CTL**	32	**REFRESH RATE CONTROL** REFRESH RATE CONTROL
1E0051F8	**WPATCMP_DAT**	32	**WRITE PATTERN COMPARE SETTING** WRITE PATTERN COMPARE SETTING
1E0051FC	**WPATCMP_CTL**	32	**WRITE PATTERN COMPARE CONTROL** WRITE PATTERN COMPARE CONTROL
1E005200	**DQODLY1**	32	**DQ output DELAY1 CHAIN setting** DQ output DELAY1 CHAIN setting
1E005204	**DQODLY2**	32	**DQ output DELAY2 CHAIN setting** DQ output DELAY2 CHAIN setting
1E005208	**DQODLY3**	32	**DQ output DELAY3 CHAIN setting** DQ output DELAY3 CHAIN setting
1E00520C	**DQODLY4**	32	**DQ output DELAY4 CHAIN setting** DQ output DELAY4 CHAIN setting
1E005210	**DQIDLY1**	32	**DQ input DELAY1 CHAIN setting** DQ input DELAY1 CHAIN setting
1E005214	**DQIDLY2**	32	**DQ input DELAY2 CHAIN setting** DQ input DELAY2 CHAIN setting
1E005218	**DQIDLY3**	32	**DQ input DELAY3 CHAIN setting** DQ input DELAY3 CHAIN setting
1E00521C	**DQIDLY4**	32	**DQ input DELAY4 CHAIN setting** DQ input DELAY4 CHAIN setting
1E005220	**DQIDLY5**	32	**DQ input DELAY5 CHAIN setting** DQ input DELAY5 CHAIN setting

1E005224	DQIDLY6	32	**DQ input DELAY6 CHAIN setting** DQ input DELAY6 CHAIN setting
1E005228	DQIDLY7	32	**DQ input DELAY7 CHAIN setting** DQ input DELAY7 CHAIN setting
1E00522C	DQIDLY8	32	**DQ input DELAY8 CHAIN setting** DQ input DELAY8 CHAIN setting
1E005280	R2R_page_hit_counter	32	**R2R_page_hit_counter** R2R_page_hit_counter
1E005284	R2R_page_miss_counter	32	**R2R_page_miss_counter** R2R_page_miss_counter
1E005288	R2R_interbank_counter	32	**R2R_interbank_counter** R2R_interbank_counter
1E00528C	R2W_page_hit_counter	32	**R2W_page_hit_counter** R2W_page_hit_counter
1E005290	R2W_page_miss_counter	32	**R2W_page_miss_counter** R2W_page_miss_counter
1E005294	R2W_interbank_counter	32	**R2W_interbank_counter** R2W_interbank_counter
1E005298	W2R_page_hit_counter	32	**W2R_page_hit_counter** W2R_page_hit_counter
1E00529C	W2R_page_miss_counter	32	**W2R_page_miss_counter** W2R_page_miss_counter
1E0052A0	W2R_interbank_counter	32	**W2R_interbank_counter** W2R_interbank_counter
1E0052A4	W2W_page_hit_counter	32	**W2W_page_hit_counter** W2W_page_hit_counter
1E0052A8	W2W_page_miss_counter	32	**W2W_page_miss_counter** W2W_page_miss_counter
1E0052AC	W2W_interbank_counter	32	**W2W_interbank_counter** W2W_interbank_counter
1E0052B0	dramc_idle_counter	32	**dramc_idle_counter** dramc_idle_counter
1E0052B4	freerun_26m_counter	32	**freerun_26m_counter** freerun_26m_counter
1E0052B8	refresh_pop_counter	32	**refresh_pop_counter** refresh_pop_counter
1E0052BC	JMETER_ST	32	**Jitter Meter Status**
1E0052C0	DQ_CAL_MAX_0	32	**DQ INPUT CALIBRATION per bit 3-0** DQ INPUT CALIBRATION per bit 3-0
1E0052C4	DQ_CAL_MAX_1	32	**DQ INPUT CALIBRATION per bit 7-4** DQ INPUT CALIBRATION per bit 7-4
1E0052C8	DQ_CAL_MAX_2	32	**DQ INPUT CALIBRATION per bit 11-8** DQ INPUT CALIBRATION per bit 11-8
1E0052CC	DQ_CAL_MAX_3	32	**DQ INPUT CALIBRATION per bit 15-12** DQ INPUT CALIBRATION per bit 15-12
1E0052D0	DQ_CAL_MAX_4	32	**DQ INPUT CALIBRATION per bit 19-16** DQ INPUT CALIBRATION per bit 19-16
1E0052D4	DQ_CAL_MAX_5	32	**DQ INPUT CALIBRATION per bit 23-20** DQ INPUT CALIBRATION per bit 23-20
1E0052D8	DQ_CAL_MAX_6	32	**DQ INPUT CALIBRATION per bit 27-34**

			DQ INPUT CALIBRATION per bit 27-24
1E0052DC	**DQ_CAL_MAX_7**	32	**DQ INPUT CALIBRATION per bit 31-28** DQ INPUT CALIBRATION per bit 31-28
1E0052E0	**DQS_CAL_MIN_0**	32	**DQS INPUT CALIBRATION per bit 3-0** DQS INPUT CALIBRATION per bit 3-0
1E0052E4	**DQS_CAL_MIN_1**	32	**DQS INPUT CALIBRATION per bit 7-4** DQS INPUT CALIBRATION per bit 7-4
1E0052E8	**DQS_CAL_MIN_2**	32	**DQS INPUT CALIBRATION per bit 11-8** DQS INPUT CALIBRATION per bit 11-8
1E0052EC	**DQS_CAL_MIN_3**	32	**DQS INPUT CALIBRATION per bit 15-12** DQS INPUT CALIBRATION per bit 15-12
1E0052F0	**DQS_CAL_MIN_4**	32	**DQS INPUT CALIBRATION per bit 19-16** DQS INPUT CALIBRATION per bit 19-16
1E0052F4	**DQS_CAL_MIN_5**	32	**DQS INPUT CALIBRATION per bit 23-20** DQS INPUT CALIBRATION per bit 23-20
1E0052F8	**DQS_CAL_MIN_6**	32	**DQS INPUT CALIBRATION per bit 27-34** DQS INPUT CALIBRATION per bit 27-24
1E0052FC	**DQS_CAL_MIN_7**	32	**DQS INPUT CALIBRATION per bit 31-28** DQS INPUT CALIBRATION per bit 31-28
1E005300	**DQS_CAL_MAX_0**	32	**DQS INPUT CALIBRATION per bit 3-0** DQS INPUT CALIBRATION per bit 3-0
1E005304	**DQS_CAL_MAX_1**	32	**DQS INPUT CALIBRATION per bit 7-4** DQS INPUT CALIBRATION per bit 7-4
1E005308	**DQS_CAL_MAX_2**	32	**DQS INPUT CALIBRATION per bit 11-8** DQS INPUT CALIBRATION per bit 11-8
1E00530C	**DQS_CAL_MAX_3**	32	**DQS INPUT CALIBRATION per bit 15-12** DQS INPUT CALIBRATION per bit 15-12
1E005310	**DQS_CAL_MAX_4**	32	**DQS INPUT CALIBRATION per bit 19-16** DQS INPUT CALIBRATION per bit 19-16
1E005314	**DQS_CAL_MAX_5**	32	**DQS INPUT CALIBRATION per bit 23-20** DQS INPUT CALIBRATION per bit 23-20
1E005318	**DQS_CAL_MAX_6**	32	**DQS INPUT CALIBRATION per bit 27-34** DQS INPUT CALIBRATION per bit 27-24
1E00531C	**DQS_CAL_MAX_7**	32	**DQS INPUT CALIBRATION per bit 31-28** DQS INPUT CALIBRATION per bit 31-28
1E005350	**DQICAL0**	32	**DQS INPUT CALIBRATION 0** DQS INPUT CALIBRATION 0
1E005354	**DQICAL1**	32	**DQS INPUT CALIBRATION 1** DQS INPUT CALIBRATION 1
1E005358	**DQICAL2**	32	**DQS INPUT CALIBRATION 2** DQS INPUT CALIBRATION 2
1E00535C	**DQICAL3**	32	**DQS INPUT CALIBRATION 3** DQS INPUT CALIBRATION 3
1E005370	**CMP_ERR**	32	**CMP ERROR** CMP ERROR
1E005374	**DQSIENDLY**	32	**DQS INPUT GATING DELAY VALUE** DQS INPUT GATING DELAY VALUE
1E00538C	**STBEN0**	32	**DQS RING COUNTER 0** DQS RING COUNTER 0
1E005390	**STBEN1**	32	**DQS RING COUNTER 1**

			DQS RING COUNTER 1
1E005394	**STBEN2**	32	**DQS RING COUNTER 2** DQS RING COUNTER 2
1E005398	**STBEN3**	32	**DQS RING COUNTER 3** DQS RING COUNTER 3
1E0053A0	**DQSDLY0**	32	**DQS INPUT DELAY SETTING 0** DQS INPUT DELAY SETTING 0
1E0053B8	**SPCMDRESP**	32	**SPECIAL COMMAND RESPONSE** SPECIAL COMMAND RESPONSE
1E0053BC	**IORGCNT**	32	**IO RING COUNTER** IO RING COUNTER
1E0053C0	**DQSGNWCNT0**	32	**DQS GATING WINODW COUNTER 0** DQS GATING WINODW COUNTER 0
1E0053C4	**DQSGNWCNT1**	32	**DQS GATING WINODW COUNTER 1** DQS GATING WINODW COUNTER 1
1E0053C8	**DQSGNWCNT2**	32	**DQS GATING WINODW COUNTER 2** DQS GATING WINODW COUNTER 2
1E0053CC	**DQSGNWCNT3**	32	**DQS GATING WINODW COUNTER 3** DQS GATING WINODW COUNTER 3
1E0053D0	**DQSGNWCNT4**	32	**DQS GATING WINODW COUNTER 4** DQS GATING WINODW COUNTER 4
1E0053D4	**DQSGNWCNT5**	32	**DQS GATING WINODW COUNTER 5** DQS GATING WINODW COUNTER 5
1E0053D8	**DQSSAMPLEV**	32	**DQS SAMPLE VALUE** DQS SAMPLE VALUE
1E0053DC	**DLLCNT0**	32	**DLL STATUS 0** DLL STATUS 0
1E0053E8	**CKPHCNT**	32	**CLOCK PHASE DETECTION RESULT** CLOCK PHASE DETECTION RESULT
1E0053FC	**TESTRPT**	32	**TEST AGENT STATUS** TEST AGENT STATUS
1E005600	**MEMPLL0**	32	**MEMPLL REGISTER SETTING 0** MEMPLL REGISTER SETTING 0
1E005604	**MEMPLL1**	32	**MEMPLL REGISTER SETTING 1** MEMPLL REGISTER SETTING 1
1E005608	**MEMPLL2**	32	**MEMPLL REGISTER SETTING 2** MEMPLL REGISTER SETTING 2
1E00560C	**MEMPLL3**	32	**MEMPLL REGISTER SETTING 3** MEMPLL REGISTER SETTING 3
1E005610	**MEMPLL4**	32	**MEMPLL REGISTER SETTING 4** MEMPLL REGISTER SETTING 4
1E005614	**MEMPLL5**	32	**MEMPLL REGISTER SETTING 5** MEMPLL REGISTER SETTING 5
1E005618	**MEMPLL6**	32	**MEMPLL REGISTER SETTING 6** MEMPLL REGISTER SETTING 6
1E00561C	**MEMPLL7**	32	**MEMPLL REGISTER SETTING 7** MEMPLL REGISTER SETTING 7
1E005620	**MEMPLL8**	32	**MEMPLL REGISTER SETTING 8** MEMPLL REGISTER SETTING 8
1E005624	**MEMPLL9**	32	**MEMPLL REGISTER SETTING 9**

			MEMPLL REGISTER SETTING 9
1E005628	**MEMPLL10**	32	**MEMPLL REGISTER SETTING 10** MEMPLL REGISTER SETTING 10
1E00562C	**MEMPLL11**	32	**MEMPLL REGISTER SETTING 11** MEMPLL REGISTER SETTING 11
1E005630	**MEMPLL12**	32	**MEMPLL REGISTER SETTING 12** MEMPLL REGISTER SETTING 12
1E005634	**MEMPLL13**	32	**MEMPLL REGISTER SETTING 13** MEMPLL REGISTER SETTING 13
1E005638	**MEMPLL14**	32	**MEMPLL REGISTER SETTING 14** MEMPLL REGISTER SETTING 14
1E005640	**MEMPLL_DIVIDER**	32	**MEMPLL DIVIDER REGISTER CONTROL** MEMPLL DIVIDER REGISTER CONTROL
1E005644	**VREF**	32	**VREF setting** VREF setting

1E005000 ACTIM0 DRAM AC TIMING SETTING 0 22560154

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TRCD				TRP				TFAW				TWR			
Type	RW				RW				RW				RW			
Reset	0	0	1	0	0	0	1	0	0	1	0	1	0	1	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BL2	CL3	CL25	CL2	TWTR				TRC				TRAS			
Type	RW	RW	RW	RW	RW				RW				RW			
Reset	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	0

Bit(s)	Name	Description
31:28	TRCD	**tRCD Timing setting** tRCD = (1 + TRCD) DRAMC clock cycles Note: DRAMC clock = 2 * DRAM clock when FDIV2 (0x7c[0]) = 1 DRAMC clock = 1 * DRAM clock when FDIV2 (0x7c[0]) = 0
27:24	TRP	**tRP Timing setting** tRP = (1 + TRP) DRAMC clock cycles
23:20	TFAW	**tFAW Timing setting** tFAW = (8 + TFAW) DRAMC clock cycles Note: 0x1e8[1] is added for TFAW[4]
19:16	TWR	**tWR Timing setting** for LPDDR2/DDR3: write command to precharge command TWR >= WL + tDQSS + (BL)/2 + tWR for LPDDR1: last data-in to precharge command tWR = (1 + TWR) DRAMC clock cycles
15	BL2	**When FDIV2 (0x7c[0]) = 1, 1 for burst length 4** When FDIV2 (0x7c[0]) = 0, set this to 0
14	CL3	**CAS Latency = 3 (FPGA), reserved**
13	CL25	**CAS Latency = 2.5 (FPGA), reserved**
12	CL2	**CAS Latency Timing setting for DDR1** CAS Latency = 2
11:8	TWTR	**tWTR Timing setting** tWTR = (1 + TWTR) DRAMC clock cycles under LPDDR1

		(WL + BL/2 + 1 + tWTR) = (3 + TWTR) DRAMC clock cycles under LPDDR2/DDR3 Note that TWTR value must be less or equal to 3 under LPDDR1, and less or equal to 'ha under other memories
7:4	TRC	**tRC Timing setting** tRC = (8 + TRC) DRAMC clock cycles Note: 0x1e8[0] is added for TRC[4]
3:0	TRAS	**tRAS Timing setting** tRAS = (8 + TRAS) DRAMC clock cycles

1E005004 CONF1 DRAM CONFIGURATION 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					TESTLP	SELFREF	DYNCLK	CLKDIS		TCMD			STRV_FRZ	STRV_EN	RDLOOP	AUTOINIT
Type					RW	RW	RW	RW		RW			RW	RW	RW	RW
Reset					0	0	0	0		0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CKEON		FW2R	FR2W	CMDHLD	BL4	MATYPE		TRRD				PAGDIS			DM64BITEN
Type	RW		RW	RW	RW	RW	RW		RW				RW			RW
Reset	0		0	0	0	0	0	0	0	0			0			0

Bit(s)	Name	Description
27	TESTLP	**Infinite self test loop enabling for test agent 1** 0: disable loop 1: enable loop
26	SELFREF	**Self-refresh mode enabling** 0: disable 1: eanble self-refresh
25	DYNCLK	
24	CLKDIS	
22:20	TCMD	**Test command** TCMD[2]: RAS_ TCMD[1]: CAS_ TCMD[0]: WE_
19	STRV_FRZ	
18	STRV_EN	
17	RDLOOP	
16	AUTOINIT	
15	CKEON	**CKE function enabling** 0: disable power down function, CKE will keep high 1: enable power down function, CKE will go down when idle
13	FW2R	**Fast write to read turnaround time (for DDR-I only)** 0: turnaround time is 0T 1: turnaround time is 1T
12	FR2W	
11	CMDHLD	
10	BL4	**When FDIV2 (0x7c[0]) = 0, 1 for DRAM burst length 4, 0 for burst length 8** When FDIV2 (0x7c[0]) = 1, 1 for DRAM burst length 8, 0 is reserved When FDIV2 (0x7c[0]) = 1, set 0x00[15] = 1 for burst length 4

9:8	MATYPE	**DRAM column address width** 00: 8 bits 01: 9 bits 10: 10 bits 11: 11 bits
7:6	TRRD	**tRRD Timing setting** tRRD = (1 + TRRD) DRAMC clock cycles Note: 0x1e8[3] is added for TRRD[2]
3	PAGDIS	**Page mode disabling** 0: disable page mode, every transaction is page-miss 1: enable page mode, page will keep opening after accessing
0	DM64BITEN	**DDR:** When FDIV2 (0x7c[0]) = 0, 1 for 64bit DRAM, 0 for 32bit DRAM When FDIV2 (0x7c[0]) = 1, 1 for 32bit DRAM, 0 for 16bit DRAM SDR: 0 for 64bit DRAM 1 for 128bit DRAM

1E005008 CONF2 DRAM CONFIGURATION 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TEST2W	TEST2R	TEST1			REFTHD										
Type	RW	RW	RW			RW										
Reset	0	0	0			0	0	0								
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									REFCNT							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	TEST2W	**Test Agent 2 write enabling** 0: Disable write cycle 1: Enable write cycle
30	TEST2R	**Test Agent 2 read enabling** 0: Disable read cycle 1: Enable read cycle
29	TEST1	**Test Agent 1 enabling** 0: Disable test agent 1 1: Enable test agent 1
26:24	REFTHD	**Refresh threshold value for promoting refresh request to high-priority** 0 means always high-priority
7:0	REFCNT	**Refresh period = (REFCNT * 16) DRAMC clock cycles** Setting the value according to DRAM spec and DRAMC frequency

1E00500C PADCTL1 DRAM PAD CONTROL 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CS1DLY				CLK0DLY											
Type	RW				RW											
Reset	0	0	0	0	0	0	0	0								
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name																
Type																
Reset																

Bit(s)	Name	Description
31:28	CS1DLY	**CS1 signal output delay** The larger value means larger delay, 1 step = 20ps
27:24	CLK0DLY	**DRAM clock 0 signal output delay** The larger value means larger delay, 1 step = 20ps

1E005010 PADCTL2 DRAM PAD CONTROL 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQM3DLY				DQM2DLY				DQM1DLY				DQM0DLY			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:12	DQM3DLY	**DRAM DQM[3] signal output delay** The larger value means larger delay, 1 step = 20ps
11:8	DQM2DLY	**DRAM DQM[2] signal output delay** The larger value means larger delay, 1 step = 20ps
7:4	DQM1DLY	**DRAM DQM[1] signal output delay** The larger value means larger delay, 1 step = 20ps
3:0	DQM0DLY	**DRAM DQM[0] signal output delay** The larger value means larger delay, 1 step = 20ps

1E005014 PADCTL3 DRAM PAD CONTROL 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS3ODLY				DQS2ODLY				DQS1ODLY				DQS0ODLY			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:12	DQS3ODLY	**DRAM DQS3 signal output delay** The larger value means larger delay, 1 step = 20ps
11:8	DQS2ODLY	**DRAM DQS2 signal output delay** The larger value means larger delay, 1 step = 20ps
7:4	DQS1ODLY	**DRAM DQS1 signal output delay** The larger value means larger delay, 1 step = 20ps
3:0	DQS0ODLY	**DRAM DQS0 signal output delay**

The larger value means larger delay, 1 step = 20ps

1E005018 DELDLY1 DQS INPUT DELAY CHAIN SETTING 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		DEL3DLY								DEL2DLY						
Type		RW								RW						
Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		DEL1DLY								DEL0DLY						
Type		RW								RW						
Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0

Bit(s)	Name	Description
30:24	DEL3DLY	**DQS3 input delay line setting** Total delay in typical case = (0.03 * DEL3DLY) ns
22:16	DEL2DLY	**DQS2 input delay line setting** Total delay in typical case = (0.03 * DEL2DLY) ns
14:8	DEL1DLY	**DQS1 input delay line setting** Total delay in typical case = (0.03 * DEL1DLY) ns
6:0	DEL0DLY	**DQS0 input delay line setting** Total delay in typical case = (0.03 * DEL0DLY) ns

1E005020 DIFDLY1 DQS INPUT DELAY CHAIN OFFSET SETTING 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		DIF3DLY								DIF2DLY						
Type		RW								RW						
Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		DIF1DLY								DIF0DLY						
Type		RW								RW						
Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0

Bit(s)	Name	Description
30:24	DIF3DLY	**Offset of DQS3 input delay line setting for auto mode** total delay in typical case = (0.03 * (DIF3DLY + DLLCNT)) ns, Binary-coded
22:16	DIF2DLY	**Offset of DQS2 input delay line setting for auto mode** total delay in typical case = (0.03 * (DIF2DLY + DLLCNT)) ns, Binary-coded
14:8	DIF1DLY	**Offset of DQS1 input delay line setting for auto mode** total delay in typical case = (0.03 * (DIF1DLY + DLLCNT)) ns, Binary-coded
6:0	DIF0DLY	**Offset of DQS0 input delay line setting for auto mode** total delay in typical case = (0.03 * (DIF0DLY + DLLCNT)) ns, Binary-coded

1E005028 DLLCONF DLL CONFIGURATION 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		DLL		MD							WC					

		FRZ		QS							KS EL2					
Type		RW		RW							RW					
Reset		0		0							0					
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Name	Description
30	DLLFRZ	**Auto-calibration value update when refresh cycle** 0: disable 1: enable
28	MDQS	**Manual mode for DQS input delay setting** 0: auto setting DQS input delay by DLL 1: manual setting DQS input delay by register
21	WCKSEL2	**Enable MIO_CK_DIV2 clocks input for MACRO_COM1 (data byte 2, 3)** 0: disable 1: enable

1E00502C TESTMODE TEST MODE CONFIGURATION 1 55000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TESTM_PAT0															
Type	RW															
Reset	0	1	0	1	0	1	0	1								
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Name	Description
31:24	TESTM_PAT0	**Test-pattern 0 for test mode**

1E00503C TEST2_1 TEST AGENT 2 CONFIGURATION 1 01200000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TEST2_PAT0								TEST2_BASE_28to5[23:16]							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TEST2_BASE_28to5[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	TEST2_PAT0	**Test-pattern 0 for test agent 2**
23:0	TEST2_BASE_28to5	**Test base address for test agent 2**

1E005040 TEST2_2 TEST AGENT 2 CONFIGURATION 2 00010000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TEST2_PAT1								TEST2_OFF_28to5[23:16]							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TEST2_OFF_28to5[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	TEST2_PAT1	**Test-pattern 1 for test agent 2**
23:0	TEST2_OFF_28to5	**Test offset address for test agent 2**

1E005044 TEST2_3 TEST AGENT 2 CONFIGURATION 3 08000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ADVPREEN	ADVREFEN	DMPGTIM							AGFZDIV2			TRFC			
Type	RW	RW	RW							RW			RW			
Reset	0	0	0	0	1	0	0	0		0			0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PERBIT	DQSUPDMODE	MANUDQSUPD	MANUDLLFRZ	DQDLYAUTO	DQSICALSTP			TESTAUDPAT	PSTWR2	DQSICALUPD	DQSICALEN	TESTCNT			
Type	RW	RW	RW	RW	RW	RW			RW	RW	RW	RW	RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	ADVPREEN	**Advanced precharge function enable** When page is idle for DMPGTIM cycles, the page is closed automatically 0: Disable advanced precharge function 1: Enable advanced precharge function
30	ADVREFEN	**Advanced refresh function enable** Used only for DDR3, DDR3 support refresh pull-in function, please refer DDR3 spec for detail 0: Disable advnaced precharge function 1: Enable advanced precharge function
29:24	DMPGTIM	**Advanced precharge function timer, use with AVDPREEN** unit: DRAMC clock
22	AGFZDIV2	**Agent half frequency mode enable**
19:16	TRFC	**tRFC Timing setting** tRFC = (11 + TRFC[7:0]) DRAMC clock cycles Note: 0x1e8[7:4] is added for TRFC[7:4]
15	PERBIT	**Per Bit HW calibration**
14	DQSUPDMODE	**0: Original manual mode for DQS input delay** 1: Update manual DQS input delay only while MANUDQSUPD=1 and DLLFRZ
13	MANUDQSUPD	**On-line manual DQS input delay adjust enabling** 0: disable DQS input delay adjust

		1: enable DQS input delay adjust, new value will be updated during refresh period
12	MANUDLLFRZ	**Manual freeze DLL counter** 0: DLL counter will be updated by hardware 1: DLL counter will be freezed for software reading
11	DQDLYAUTO	**DQ delay auto-update during calibration** 0: No update 1: Update
10:8	DQSICALSTP	**HW calibration step (=DQSICALSTP*2)**
7	TESTAUDPAT	**Select audio pattern as test pattern of test agent2** 0: ISI pattern 1: audio pattern
6	PSTWR2	
5	DQSICALUPD	**Update DQS input delay setting to calibrated value** 0: disable update 1: enable update
4	DQSICALEN	**HW calibration enable** 0: disable HW calibration 1: enable HW calibration
3:0	TESTCNT	**Test loop number of test agent2** loop number = 2^(TESTCNT)

1E005048 TEST2_4 TEST AGENT 2 CONFIGURATION 4 0000110D

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TZQCS															
Type	RW															
Reset	0	0	0	0	0	0	0	0								
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TESTAUDMODE	TESTAUDBITINV		TESTAUDINIT							TEST2DISSCRAM	TESTAUDINC				
Type	RW	RW		RW							RW	RW				
Reset	0	0		1	0	0	0	1			0	0	1	1	0	1

Bit(s)	Name	Description
31:24	TZQCS	**tZQCS Timing setting** tZQCS = (2 + TZQCS) DRAMC clock cycles
15	TESTAUDMODE	**Audio pattern: write after read enabling** 0: read only 1: write after read
14	TESTAUDBITINV	**Audio pattern bit inversion enabling** 0: No bit inversion 1: Bit inversion
12:8	TESTAUDINIT	**Initial bit inverse position for audio pattern**
5	TEST2DISSCRAM	**Test agent bypass scramble function** 0: not bypass scramble function 1: bypass scramble function
4:0	TESTAUDINC	**Bit inverse incremental value for audio pattern**

1E00507C DDR2CTL DDR2 CONTROL REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RODTE	WLAT			FIXRODT	RODT			EODT	TWODT						
Type	RW	RW			RW	RW			RW	RW						
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TR2W					TRTP			EROT	DATLAT			WOEN	ROEN	DDR2EN	FDIV2
Type	RW					RW			RW	RW			RW	RW	RW	RW
Reset	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	RODTE	**Read ODT extend 1T mode enable** 0: not extend 1T 1: extend 1T Unit: DRAM clock cycle
30:28	WLAT	**Write latency = WLAT + 1/WAT + 3 when FDIV2 = 0/1** for example, write latency = 5T, set WLAT = 4/2 when FDIV2 = 0/1 Unit: DRAM clock cycle
27	FIXRODT	**Fix READ cycle ODT signal** Fix the ODT signal value (to control the PAD termination) as always enabled 0: Not fix on ODT 1: Fix on ODT
26:24	RODT	**Read ODT timing control for DDR2** 000: For CL3 001: For CL4 and CL5 010: for CL6 and CL7
23	EODT	
22:16	TWODT	**Write ODT latency enabling for DDR2** 111111: for all cases others: Reserved
15:12	TR2W	**Read to write interval time = (TR2W[3:0] + 3) DRAMC clock cycles** TR2W values are the same when >= 'h9
10:8	TRTP	**tRTP = (TRTP + 1) DRAMC clock cycle** In LPDDR, tRTP = BL/2. HW design guarantees, no need to set TRTP In LPDDR2, tRTP begins (BL/2 - 2) clock cycles after the read command In DDR3, precharge can be issued after AL + tRTP
7	EROT	**Read ODT timing control for DDR2** 0: for CL3, CL5, CL7 1: for CL4, CL6
6:4	DATLAT	**Internal read data timing control** The register and 0xE4[4] (DATLAT3) 4'b0101: CL6, CL7 for DDR2 4'b0100: CL4, CL5 for DDR2 4'b0111: CL9, CL10 for GDDR3 4'b1000: CL11, CL12 for GDDR3
3	WOEN	**Write ODT enabling** 0: disable Write ODT 1: enable Write ODT
2	ROEN	**Read ODT enabling** 0: disable Read ODT 1: enable Read ODT

1	DDR2EN	**DDR2 enabling** 0: disable DDR2 function 1: enable DDR2 function
0	FDIV2	**Half frequency mode** 0: DRAMC clock cycle = DRAM clock cycle 1: DRAMC clock cycle = 2 * DRAM clock cycle

1E005084 ZQCS ZQCS setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ZQCSAD								ZQCSOP							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:8	ZQCSAD	
7:0	ZQCSOP	

1E005088 MRS MRS value setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	OCDADJ								MRSOP							
Type	RW								RW							
Reset	0								0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MRSBA			MRSMA												
Type	RW			RW												
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	OCDADJ	
23:16	MRSOP	**For LPDDR2 MRW's OP (MRS_OP[7:0])**
15:13	MRSBA	**For non-LPDDR2 MRW bank address**
12:0	MRSMA	**For MRR/MRW address**

1E00508C CLK1DELAY Clock 1 output delay CONTROL 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	OCDPAT												CLK1DLY			
Type	RW												RW			
Reset	0	0	0	0	0	0	0	0					0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															FIF	

															OLEN1	
Type															RW	
Reset															0	

Bit(s)	Name	Description
31:24	OCDPAT	
19:16	CLK1DLY	**DRAM clock 1 signal output delay** The larger value means larger delay, 1 step = 20ps
1	FIFOLEN1	**Read FIFO length in DDRPHY** 0: 8-level 1: 4-level

1E005090 IOCTL IO CONTROL 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SIOEN															
Type	RW															
Reset	0															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Name	Description
31	SIOEN	**DQS singal output enable** 0: Differentail output 1: Single-end output

1E005094 DQSIEN DQS INPUT RANGE FINE TUNER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		DQS3IEN								DQS2IEN						
Type		RW								RW						
Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		DQS1IEN								DQS0IEN						
Type		RW								RW						
Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0

Bit(s)	Name	Description
30:24	DQS3IEN	**DQS3 gating delay control** DRAMC uses the gated DQS to sample DQ. To get the right gated DQS, the gating signal need to be adjust to match DRAM, PCB, and others components. 1 step = 20ps.
22:16	DQS2IEN	**DQS2 gating delay control**
14:8	DQS1IEN	**DQS1 gating delay control**
6:0	DQS0IEN	**DQS0 gating delay control**

1E0050B8 DRVCTL0 PAD DRIVING CONTROL SETTING 0 AA22AA22

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQSDRVP				DQSDRVN				DQSRTTBPJ	DSODTP			DQSRTTBNJ	DSODTN		
Type	RW				RW				RW	RW			RW	RW		
Reset	1	0	1	0	1	0	1	0	0	0	1	0	0	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQDRVP				DQDRVN				DQRTTBPJ	DQODTP			DQRTTBNJ	DQODTN		
Type	RW				RW				RW	RW			RW	RW		
Reset	1	0	1	0	1	0	1	0	0	0	1	0	0	0	1	0

Bit(s)	Name	Description
31:28	DQSDRVP	**DQS P driving control, refer IBIST model for driving strength**
27:24	DQSDRVN	**DQS N driving control, refer IBIST model for driving strength**
23	DQSRTTBPJ	**DQS PAD RTTBPJ port control**
22:20	DSODTP	**DQSODT P driving control, refer IBIST model for driving strength**
19	DQSRTTBNJ	**DQS PAD RTTBNJ port control**
18:16	DSODTN	**DQSODT N driving control, refer IBIST model for driving strength**
15:12	DQDRVP	**DQ P driving control, refer IBIST model for driving strength**
11:8	DQDRVN	**DQ N driving control, refer IBIST model for driving strength**
7	DQRTTBPJ	**DQ PAD RTTBPJ port control**
6:4	DQODTP	**DQODT P driving control, refer IBIST model for driving strength**
3	DQRTTBNJ	**DQ PAD RTTBNJ port control**
2:0	DQODTN	**DQODT N driving control, refer IBIST model for driving strength**

1E0050BC DRVCTL1 PAD DRIVING CONTROL SETTING 1 AA22AA00

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CLKDRVP				CLKDRVN				CLKRTTBPJ	CKODTP			CLKRTTBNJ	CKODTN		
Type	RW				RW				RW	RW			RW	RW		
Reset	1	0	1	0	1	0	1	0	0	0	1	0	0	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CMDDRVP				CMDDRVN											
Type	RW				RW											
Reset	1	0	1	0	1	0	1	0								

Bit(s)	Name	Description
31:28	CLKDRVP	**CLK P driving control, refer IBIST model for driving strength**
27:24	CLKDRVN	**CLK N driving control, refer IBIST model for driving strength**
23	CLKRTTBPJ	**CLK PAD RTTBPJ port control**
22:20	CKODTP	**CLK ODT P driving control, refer IBIST model for driving strength**
19	CLKRTTBNJ	**CLK PAD RTTBNJ port control**

18:16	CKODTN	**CLK ODT N driving control, refer IBIST model for driving strength**
15:12	CMDDRVP	**CMD P driving control, refer IBIST model for driving strength**
11:8	CMDDRVN	**CMD N driving control, refer IBIST model for driving strength**

1E0050C0 DLLSEL DLL SELECTION SETTING 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DLL67SEL		DLL45SEL		DLL23SEL		DLL01SEL		DLLCNTSEL		AUTOKMODE		CMPEN	CMPCAL	CMPDRVNE	CMPDRVPE
Type	RW		RW		RW		RW		RW		RW		RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CMPDRVP				CMPDRVN				CMODTPE	CMPODTP			CMODTNE	CMPODTN		
Type	RW				RW				RW	RW			RW	RW		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:30	DLL67SEL	**DLL6 and DLL7 counter selection**
29:28	DLL45SEL	**DLL4 and DLL5 counter selection**
27:26	DLL23SEL	**DLL2 and DLL3 counter selection**
25:24	DLL01SEL	**DLL0 and DLL1 counter selection**
23:22	DLLCNTSEL	**DLL counter selection** Refer the previous bit fields for meaning
21:20	AUTOKMODE	**OCD/ODT calibration mode selection** 00: disable auto calibration 01: DRVP mode 10: DRVN mode 11: ODTP mode
19	CMPEN	**Compensation counter enabling** 0: disable 1: enable
18	CMPCAL	**Connect to CMP pad CALP**
17	CMPDRVNE	**Connect to CMP pad DRVNE**
16	CMPDRVPE	**Connect to CMP pad DRVPE**
15:12	CMPDRVP	**Connect to CMP pad DRVP[3:0]**
11:8	CMPDRVN	**Connect to CMP pad DRVN[3:0]**
7	CMODTPE	**CMP PAD ODTPE port**
6:4	CMPODTP	**CMP ODT P driving control**
3	CMODTNE	**CMP PAD ODTNE port**
2:0	CMPODTN	**CMP ODT N driving control**

1E0050CC TDSEL0 IO OUTPUT DUTY CONTROL 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					DQS3TDSEL								DQS2TDSEL			

Type					RW								RW			
Reset					0	0	0	0					0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					CMDTDSEL								CLKTDSEL			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0

Bit(s)	Name	Description
27:24	DQS3TDSEL	**DQS3 output duty control**
19:16	DQS2TDSEL	**DQS2 output duty control**
11:8	CMDTDSEL	**Command output duty control**
3:0	CLKTDSEL	**DRAM clock output duty control**

1E0050D0 TDSEL1 IO OUTPUT DUTY CONTROL 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					DQS1TDSEL								DQS0TDSEL			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQB3TDSEL				DQB2TDSEL				DQB1TDSEL				DQB0TDSEL			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
27:24	DQS1TDSEL	**DQS1 output duty control**
19:16	DQS0TDSEL	**DQS0 output duty control**
15:12	DQB3TDSEL	**DQ byte3 output duty control**
11:8	DQB2TDSEL	**DQ byte2 output duty control**
7:4	DQB1TDSEL	**DQ byte1 output duty control**
3:0	DQB0TDSEL	**DQ byte0 output duty control**

1E0050D8 MCKDLY MEMORY CLOCK DELAY CHAIN SETTING 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PINMUX			_16 BIT FULL						ODTREN			DISDQIEN			
Type	RW			RW						RW			RW			
Reset	0	0		0						0			0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FIXDQIEN															
Type	RW															
Reset	0	0	0	0												

Bit(s)	Name	Description
31:30	PINMUX	**PINMUX function** 00: DDR3 (PCB4L)/LPDDR2-POP (PCB8L) 01: DDR3 (PCB6L) 10:LPDDR2 (PCB6L)

		11: Reserve
28	_16BITFULL	**DRAM bus is 16-bit and FDIV2 = 0**
22	ODTREN	**Write ODT turn on when reading** 0: disable 1: enable
19:16	DISDQIEN	**Disable DQ input enable** 0: DQ input enable when necessary 1: DQ input disable
15:12	FIXDQIEN	**DQ input enable fixed mode** 0: DQ input enable when necessary 1: Keep DQ input always on

1E0050DC DQSCTL0 DQS INPUT RANGE CONTROL 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									DQS1CTL[11:4]							
Type									RW							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS1CTL[3:0]				DQS0CTL											
Type	RW				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
23:12	DQS1CTL	**DQS1 input range control, 1 hot encoding** Unit: 1/2 DRAM clock cycle
11:0	DQS0CTL	**DQS0 input range control, 1 hot encoding** Unit: 1/2 DRAM clock cycle

1E0050E0 DQSCTL1 DQS INPUT RANGE CONTROL 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				DQSIENMODE		DQSINCTL			DQS3CTL[11:4]							
Type				RW		RW			RW							
Reset				0		0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS3CTL[3:0]				DQS2CTL											
Type	RW				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28	DQSIENMODE	**DQS gating mode selection** 0: pulse mode 1: burst mode
26:24	DQSINCTL	**DQS input range control by M_CK** 0/1/ .../7 = delay 0/1/ .../7T Unit: DRAMC clock cycle
23:12	DQS3CTL	**DQS3 input range control, 1hot encoding**

		Unit: 1/2 DRAM clock cycle
11:0	DQS2CTL	**DQS2 input range control, 1hot encoding** Unit: 1/2 DRAM clock cycle

1E0050E4 PADCTL4 PAD CONTROL 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CLKPADCTL				CMDPADCTL				DQSPADCTL				DQPADCTL			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		DQSRTT				DQRTT			DDR3EN	ZQCSEN	BC4OTF	DATLAT3	CKEFIXOFF	CKEFIXON	GDDR3RST	
Type		RW				RW			RW	RW	RW	RW	RW	RW	RW	
Reset		0	0	0		0	0	0	0	0	0	0	0	0	0	

Bit(s)	Name	Description
31:28	CLKPADCTL	**Clock pad control**
27:24	CMDPADCTL	**CMD pad control**
23:20	DQSPADCTL	**DQS pad control**
19:16	DQPADCTL	**DQ pad control**
14:12	DQSRTT	**DQS termination control**
10:8	DQRTT	**DQ termination control**
7	DDR3EN	**enable DDR3 mode** 0: Not enable DDR3 1: Enable
6	ZQCSEN	**ZQCS enable (ZQ calibration short), for DDR3 only** 0: disable 1: enable
5	BC4OTF	**Burst chop 4 on the fly mode** 0: disable 1: enable
4	DATLAT3	**Read data latch timing control bit 3**
3	CKEFIXOFF	**CKE always off** 0: CKE hardware control 1: CKE always off
2	CKEFIXON	**CKE always on** 0: CKE hardware control 1: CKE always on
1	GDDR3RST	**GDDR3/DDR3 reset pin enable** 0: reset disable 1: reset enable

1E0050E8 PADCTL5 PAD CONTROL 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DQS3RDSEL								DQS2RDSEL					
Type			RW								RW					
Reset			0	0	0	0	0	0			0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DQS1RDSEL								DQS0RDSEL					
Type			RW								RW					
Reset			0	0	0	0	0	0			0	0	0	0	0	0

Bit(s)	Name	Description
29:24	DQS3RDSEL	**DQS 3 RDSEL (duty cycle control)** Unit: 50ps
21:16	DQS2RDSEL	**DQS 2 RDSEL (duty cycle control)** Unit: 50ps
13:8	DQS1RDSEL	**DQS 1 RDSEL (duty cycle control)** Unit: 50ps
5:0	DQS0RDSEL	**DQS 0 RDSEL (duty cycle control)** Unit: 50ps

1E0050EC PADCTL6 PAD CONTROL 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DQ3RDSEL								DQ2RDSEL					
Type			RW								RW					
Reset			0	0	0	0	0	0			0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DQ1RDSEL								DQ0RDSEL					
Type			RW								RW					
Reset			0	0	0	0	0	0			0	0	0	0	0	0

Bit(s)	Name	Description
29:24	DQ3RDSEL	**DQ Byte 3 RDSEL (duty cycle control)** Unit: 50ps
21:16	DQ2RDSEL	**DQ Byte 2 RDSEL (duty cycle control)** Unit: 50ps
13:8	DQ1RDSEL	**DQ Byte 1 RDSEL (duty cycle control)** Unit: 50ps
5:0	DQ0RDSEL	**DQ Byte 0 RDSEL (duty cycle control)** Unit: 50ps

1E0050F0 PHYCTL1 DDR PHY CONTROL 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQ4BMUX			PHYRST				FIXDQSIEN								
Type	RW			RW				RW								
Reset	0			0				0								
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Name	Description

31	DQ4BMUX	**DQ 4-bit multiplex for DDR3** 0: Disable 1: Enable
28	PHYRST	**PHY reset enable** 0: disable 1: enable
24	FIXDQSIEN	**DQS input enable always on** 0: Hardware control 1: always on

1E0050F4 GDDR3CTL1 GDDR3 CONTROL 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				PHYSYNCM			RDATRST	_8BKEN								
Type				RW			RW	RW								
Reset				0			0	0								
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Name	Description
28	PHYSYNCM	**SYNC MODE using inverted PHY_M_CK**
25	RDATRST	**Read data counter reset** 0: disable 1: enable reset
24	_8BKEN	**8-bank device enable** 0: for 4-bank device 1: for 8-bank device

1E0050F8 PADCTL7 PAD CONTROL 4 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name							DRAMOEN	LBTEST								
Type							RW	RW								
Reset							0	0								

Bit(s)	Name	Description
9	DRAMOEN	**DRAM pin output enable** 0: disable, DRAM pin will be Hi-Z 1: enable
8	LBTEST	**Loop-back test mode enable** 0: disable

1: enable

1E0050FC MISCTL0 MISC CONTROL 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		TXP				REFP_ARB_EN	REFA_ARB_EN	PBC_ARB_EN				ASYNCEN			INTLBT	MODE18V
Type		RW				RW	RW	RW				RW			RW	RW
Reset		0	0	0		0	0	0				0			0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Name	Description
30:28	TXP	**tXP Timing setting** tXP = (2 + TXP) DRAMC clock cycles
26	REFP_ARB_EN	**Per-bank refresh blocks EMI arbitration** 0: disable 1: enable
25	REFA_ARB_EN	**All-bank refresh blocks EMI arbitration** 0: disable 1: enable
24	PBC_ARB_EN	**Block page-miss requests in EMI arbitration** 0: disable 1: enable
20	ASYNCEN	**Asynchronous mode enabling between DRAMC & DDRPHY** 0: synchronous mode 1: asynchronous mode
17	INTLBT	**IO internal loop back** 0: disable 1: enable
16	MODE18V	**IO voltage operating condition** 0: 1.2V 1: 1.8V

1E005100 OCDK OCD CALIBRATION CONTROL 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	WDATKEY7	WDATKEY6		WDATITLV	WDATKEY5	WDATKEY4		DRVREF	WDATKEY3	WDATKEY2			WDATKEY1	WDATKEY0	INTREF_SEL	
Type	RW	RW		RW	RW	RW		RW	RW	RW			RW	RW	RW	
Reset	0	0		0	0	0		0	0	0			0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DRDELSWEN				DRDELSWSEL			AUTOCALDRV	AUTOKCNT							

Type	RW				RW			RW	RW							
Reset	0				0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	WDATKEY7	**Data encryption key bit 7**
30	WDATKEY6	**Data encryption key bit 6**
28	WDATITLV	**Data scramble enable** 0: disable 1: enable
27	WDATKEY5	**Data encryption key bit 5**
26	WDATKEY4	**Data encryption key bit 4**
24	DRVREF	**Driving change only when refresh cycle** 0: disable, change will be apply directly 1: enable, change will be apply during refresh
23	WDATKEY3	**Data encryption key bit 3**
22	WDATKEY2	**Data encryption key bit 2**
19	WDATKEY1	**Data encryption key bit 1**
18	WDATKEY0	**Data encryption key bit 0**
17:16	INTREF_SEL	**Calibration I/O PAD VREF selection** 00: 0.5*VDDQ 01: 0.6*VDDQ 10: 0.7*VDDQ 11: 0.8*VDDQ
15	DRDELSWEN	**Enable DQS input delay switching for different ranks** 0: disable 1: enable
11:9	DRDELSWSEL	**Timing control of DQS input delay switching for different ranks** Unit: DRAMC clock
8	AUTOCALDRV	**OCD calibration** 0: calibration disable 1: calibration enable
7:0	AUTOKCNT	**Auto calibration counter** Set timer for calibration cycle Unit: DRAMC clock

1E005104 LBWDAT0 LOOP BACK DATA 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	LBWDATA0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	LBWDATA0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	LBWDATA0	**Loop-back test mode data 0**

1E005108 LBWDAT1 LOOP BACK DATA 1 0000000

0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	LBWDATA1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	LBWDATA1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	LBWDATA1	**Loop-back test mode data 1**

1E00510C LBWDAT2 LOOP BACK DATA 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	LBWDATA2[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	LBWDATA2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	LBWDATA2	**Loop-back test mode data 2**

1E005110 RKCFG RANK CONFIGURATION 03051100

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name						RKSIZE							XRTW2W		XRTW2R	
Type						RW							RW		RW	
Reset						0	1	1					0	1	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		XRTR2W				XRTR2R			PBREFEN			MRS2RK	RKSWAP	RKMODE		
Type		RW				RW			RW			RW	RW	RW		
Reset		0	0	1		0	0	1	0			0	0	0	0	0

Bit(s)	Name	Description
26:24	RKSIZE	**Rank address selection** 000: ADDR[31] 001: ADDR[30] 010: ADDR[29] 011: ADDR[28] 100: ADDR[27] 101: ADDR[26] 110: ADDR[25] 111: ADDR[24]
19:18	XRTW2W	**cross rank timing W2W** Unit: DRAM controller clock

17:16	XRTW2R	**cross rank timing W2R** Unit: DRAM controller clock
14:12	XRTR2W	**cross rank timing R2W; note that XRTR2W = 6/7 have the same setting** Unit: DRAM controller clock
10:8	XRTR2R	**cross rank timing R2R** Unit: DRAM controller clock
7	PBREFEN	**Per-bank refresh enable for LPDDR2** 0: disable 1: enable
4	MRS2RK	**MRS commands are sent to 2 ranks simulataneously** 0: disable 1: enable
3	RKSWAP	**swap CS<->CS1** 0: disable 1: enable
2:0	RKMODE	**Multi-rank mode support** Set to non-zero for multi-rank

1E005114 CKPHDET CLOCK PHASE DETECTION SETTING 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									CKPHCNTEN							
Type									RW							
Reset									0							
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CKPHCHKCYC															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
23	CKPHCNTEN	**MEMPLL in/out clock phase detection counter enabling** 0: disable 1: enable
15:0	CKPHCHKCYC	**MEMPLL in/out clock phase detection counter cycle** Unit: DRAMC clock

1E005124 DQSGCTL INPUT DQS GATING CONTROL 80000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	NEWDQSG_SEL													BYPASS_DMPAD_COM1	BYPASS_DMPAD_COM0	DMYPAD_RXSEL
Type	RW													RW	RW	RW
Reset	1													0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQSG_FINE_DLY_COM1				DQSG_FINE_DLY_COM0						DQSG_COARSE_DLY_COM1				DQSG_COARSE_DLY_COM0	
Type	RW				RW						RW				RW	
Reset	0	0	0	0	0	0	0	0			0	0			0	0

Bit(s)	Name	Description
31	NEWDQSG_SEL	**DQS gating control method** 0: old 1: new
18	BYPASS_DMPAD_COM1	**Bypass dummy PAD for DQS2/3 gating signal** 0: not bypass 1: bypass
17	BYPASS_DMPAD_COM0	**Bypass dummy PAD for DQS0/1 gating signal** 0: not bypass 1: bypass
16	DMYPAD_RXSEL	**Select O/O1 pin of dummy PAD for gating signal input** 0: O pin 1: O1 pin
15:12	DQSG_FINE_DLY_COM1	**Fine tune delay setting for DQS2/3 gating signal before dummy PAD** Unit: 20ps
11:8	DQSG_FINE_DLY_COM0	**Fine tune delay setting for DQS1/0 gating signal before dummy PAD** Unit: 20ps
5:4	DQSG_COARSE_DLY_COM1	**Coarse tune delay setting for DQS2/3 gating signal before dummy PAD** Unit: 0.25/0.5T of DRAMC clock under 2X/1X mode
1:0	DQSG_COARSE_DLY_COM0	**Coarse tune delay setting for DQS0/1 gating signal before dummy PAD** Unit: 0.25/0.5T of DRAMC clock under 2X/1X mode

1E005130 CLKENCTL DRAM CLOCK ENABLE CONTROL 10000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			CLK1EN	CLK0EN												
Type			RW	RW												
Reset			0	1												
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Name	Description
29	CLK1EN	**DRAM clock 1 enable** 0: disable 1: enable
28	CLK0EN	**DRAM clock 0 enable** 0: disable 1: enable

1E005140 DQSGCTL1 DQS gating delay control 1 0000000

0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		DQSIPRE1DLY								DQSIPOS1DLY						
Type		RW								RW						
Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		DQSIPRE0DLY								DQSIPOS0DLY						
Type		RW								RW						
Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0

Bit(s)	Name	Description
30:24	DQSIPRE1DLY	**DQS PRE delay control for DQS1**
22:16	DQSIPOS1DLY	**DQS POS delay control for DQS1**
14:8	DQSIPRE0DLY	**DQS PRE delay control for DQS0**
6:0	DQSIPOS0DLY	**DQS POS delay control for DQS0**

1E005144 DQSGCTL2 DQS gating delay control 2 0000000 0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		DQSIPRE3DLY								DQSIPOS3DLY						
Type		RW								RW						
Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		DQSIPRE2DLY								DQSIPOS2DLY						
Type		RW								RW						
Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0

Bit(s)	Name	Description
30:24	DQSIPRE3DLY	**DQS PRE delay control for DQS3**
22:16	DQSIPOS3DLY	**DQS POS delay control for DQS3**
14:8	DQSIPRE2DLY	**DQS PRE delay control for DQS2**
6:0	DQSIPOS2DLY	**DQS POS delay control for DQS2**

1E005168 ARBCTL0 ARBITRATION CONTROL 0 0000000 0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									MAXPENDCNT							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7:0	MAXPENDCNT	**Maximum pending number to block the arbitration**

1E0051A8 CMDDLY0 Command Delay CTL0 0000000

0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					RA3DLY								RA2DLY			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					RA1DLY								RA0DLY			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0

Bit(s)	Name	Description
27:24	RA3DLY	**RA output delay chain setting for bit3** Unit: 20ps
19:16	RA2DLY	**RA output delay chain setting for bit2** Unit: 20ps
11:8	RA1DLY	**RA output delay chain setting for bit1** Unit: 20ps
3:0	RA0DLY	**RA output delay chain setting for bit0** Unit: 20ps

1E0051AC CMDDLY1 Command Delay CTL1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					RA7DLY								RA6DLY			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					RA5DLY								RA4DLY			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0

Bit(s)	Name	Description
27:24	RA7DLY	**RA output delay chain setting for bit7** Unit: 20ps
19:16	RA6DLY	**RA output delay chain setting for bit6** Unit: 20ps
11:8	RA5DLY	**RA output delay chain setting for bit5** Unit: 20ps
3:0	RA4DLY	**RA output delay chain setting for bit4** Unit: 20ps

1E0051B0 CMDDLY2 Command Delay CTL2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					RA11DLY								RA10DLY			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					RA9DLY								RA8DLY			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0

Bit(s)	Name	Description
27:24	RA11DLY	**RA output delay chain setting for bit11** Unit: 20ps
19:16	RA10DLY	**RA output delay chain setting for bit10** Unit: 20ps
11:8	RA9DLY	**RA output delay chain setting for bit9** Unit: 20ps
3:0	RA8DLY	**RA output delay chain setting for bit8** Unit: 20ps

1E0051B4 CMDDLY3 Command Delay CTL3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					BA2DLY								BA1DLY			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					BA0DLY								RA12DLY			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0

Bit(s)	Name	Description
27:24	BA2DLY	**BA output delay chain setting for bit2** Unit: 20ps
19:16	BA1DLY	**BA output delay chain setting for bit1** Unit: 20ps
11:8	BA0DLY	**BA output delay chain setting for bit0** Unit: 20ps
3:0	RA12DLY	**RA output delay chain setting for bit12** Unit: 20ps

1E0051B8 CMDDLY4 Command Delay CTL4 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				CASDLY								RASDLY				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				CKEDLY						CSMONSEL	CSMONEN	CSDLY				
Type				RW						RW	RW	RW				
Reset				0	0	0	0	0		0	0	0	0	0	0	0

Bit(s)	Name	Description
28:24	CASDLY	**CAS output delay chain setting** Unit: 20ps
20:16	RASDLY	**RAS output delay chain setting**

		Unit: 20ps
12:8	CKEDLY	**CKE output delay chain setting** Unit: 20ps
6	CSMONSEL	**DQSIEN monitor through CS select (only for 6517)**
5	CSMONEN	**DQSIEN monitor through CS enable (only for 6517)**
4:0	CSDLY	**CS output delay chain setting** Unit: 20ps

1E0051BC CMDDLY5 Command Delay CTL5 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		CSXMONSEL	CSXMONEN	OTDLY								RA13DLY				
Type		RW	RW	RW								RW				
Reset		0	0	0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				WEDLY												
Type				RW												
Reset				0	0	0	0	0								

Bit(s)	Name	Description
30	CSXMONSEL	**DQSIEN monitor through CS1 select (only for 6517)**
29	CSXMONEN	**DQSIEN monitor through CS1 enable (only for 6517)**
28:24	OTDLY	**OTD output delay chain setting** Unit: 20ps
20:16	RA13DLY	**RA output delay chain setting for bit13** Unit: 20ps
12:8	WEDLY	**WE output delay chain setting** Unit: 20ps

1E0051C0 DQSCAL0 DQS CAL CONTROL 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	STBCALEN			RA14DLY												
Type	RW			RW												
Reset	0			0	0	0	0	0								
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQSIENHLMTEN	DQSIENHLMT							DQSIENLLMTEN	DQSIENLLMT						
Type	RW	RW							RW	RW						
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

31	STBCALEN	**DQS strobe calibration enable** 0: disable 1: enable
28:24	RA14DLY	**RA output delay chain setting for bit14** Unit: 20ps
15	DQSIENHLMTEN	**DQS strobe calibration high-limit enable** 0: disable 1: enable
14:8	DQSIENHLMT	**DQS strobe calibration high-limit value**
7	DQSIENLLMTEN	**DQS strobe calibration low-limit enable** 0: disable 1: enable
6:0	DQSIENLLMT	**DQS strobe calibration low-limit value**

1E0051D8 DMMonitor Monitor parameter NA

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	JMTRCNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DSMONSEL										BUSMONEN_SW	MONPAUSE_SW		JMTR_EN
Type			RW										RW	RW		RW
Reset			10b'0000000000										0	0		0

Bit(s)	Name	Description
31:16	JMTRCNT	**Set monitor Period for Jitter Meter**
13:4	DSMONSEL	**DQSIEN monitor signal selection (only for 6517)**
3	BUSMONEN_SW	**Bus monitor enable. Can't use with BUSMONEN_HW at the same time.** 0: disable 1: enable
2	MONPAUSE_SW	**Pause Bus monitor Counter. Can't use with MONPAUSE_HW at the same time.** 0: disable 1: enable
0	JMTR_EN	**Jitter meter enable** 0: disable 1: enable

1E0051DC DRAMC_PD_CTRL PD mode parameter 00622840

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name						MIOCKCTRLOF	DCMEN	REFFRERUN	REFCNT_FR_CLK							

						F										
Type						RW	RW	RW	RW							
Reset						0	0	0	0	1	1	0	0	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TXREFCNT									DCMDLYREF						
Type	RW									RW						
Reset	0	0	1	0	1	0	0	0		1	0	0				

Bit(s)	Name	Description
26	MIOCKCTRLOFF	**dram clk gating parameter** 1 : always no gating 0 : controlled by dramc
25	DCMEN	**DRAMC non-freerun clock gating function** 0: disable 1: enable
24	REFFRERUN	**Using FREE-RUN CLK to count reflesh period**
23:16	REFCNT_FR_CLK	**Refresh period = (REFCNT_FR_CLK) DRAMC FREE-RUN clock cycles** Setting the value according to DRAM spec and DRAMC FREE-RUN frequency
15:8	TXREFCNT	**tXSR** 258T/3T~258T for DDR3/LPDDR2
6:4	DCMDLYREF	**Number of delay cycles to wake up DCM by the refresh command, which is counted by the FREE-RUN clock** Note that this value can't be set to 3'b000!

1E0051E0 LPDDR2 LPDDR2 setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ADRDECEN	SELO1ASO	DDRA14	LPDDR2EN	WDATRGO	FASTOE		DDRCS1		DDRODT	DDRCKE	DDRCS	DDRRAS	DDRCAS	DDRWE	DDRBA[2:2]
Type	RW	RW	RW	RW	RW	RW		RW		RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0		0		0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DDRBA[1:0]		DDRA													
Type	RW		RW													
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	ADRDECEN	**DRAM address decode** 0: by EMI 1: by DRAMC
30	SELO1ASO	**Select IO O1 as output** 0: select O 1: select O1
29	DDRA14	**DDR mode for A[14] pin (LPDDR2 DDR command rate)** 0: disable 1: enable
28	LPDDR2EN	**LPDDR2 enable** 0: disable 1: enable
27	WDATRGO	**Enable register output data by DRAMC** 0: disable

		1: enable
26	FASTOE	**Fast IO output enable** 0: disable 1: enable
24	DDRCS1	**DDR mode for CS1 pin (LPDDR2 DDR command rate)** 0: disable 1: enable
22	DDRODT	**DDR mode for ODT pin (LPDDR2 DDR command rate)** 0: disable 1: enable
21	DDRCKE	**DDR mode for CKE pin (LPDDR2 DDR command rate)** 0: disable 1: enable
20	DDRCS	**DDR mode for CS pin (LPDDR2 DDR command rate)** 0: disable 1: enable
19	DDRRAS	**DDR mode for RAS pin (LPDDR2 DDR command rate)** 0: disable 1: enable
18	DDRCAS	**DDR mode for CAS pin (LPDDR2 DDR command rate)** 0: disable 1: enable
17	DDRWE	**DDR mode for WE pin (LPDDR2 DDR command rate)** 0: disable 1: enable
16:14	DDRBA	**DDR mode for BA[2:0] pin (LPDDR2 DDR command rate)** 0: disable 1: enable
13:0	DDRA	**DDR mode for A[13:0] pin (LPDDR2 DDR command rate)** 0: disable 1: enable

1E0051E4 SPCMD Special command mode 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PADRG_RDSEL								ZQCSCNT							
Type	RW								RW							
Reset	0	0	0	0					0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			CMPPD				DQSGCNTRST	DQSGCNTEN			TCMDEN	ZQCEN	AREFEN	PREAEN	MRREN	MRWEN
Type			RW				RW	RW			RW	RW	RW	RW	RW	RW
Reset			0				0	0			0	0	0	0	0	0

Bit(s)	Name	Description
31:28	PADRG_RDSEL	**PAD Ring RDSEL (only for 6517)**
23:16	ZQCSCNT	**Every refresh number to issue ZQCS commands, only for DDR3**
13	CMPPD	**Power down control of CMP IO**
9	DQSGCNTRST	**DQS gating window counter Reset**
8	DQSGCNTEN	**DQS gating window counter Enable**

5	TCMDEN	**Test command enable** 0: disable 1: enable
4	ZQCEN	**ZQ calibration enable** 0: disable 1: enable
3	AREFEN	**Auto Refresh command enable** 0: disable 1: enable
2	PREAEN	**Precharge all command enable** 0: disable 1: enable
1	MRREN	**Mode register read command enable** 0: disable 1: enable
0	MRWEN	**Mode register write command enable** 0: disable 1: enable

1E0051E8 ACTIM1 DRAM AC TIMING SETTING 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name							TRPAB		REFRCNT							
Type							RW		RW							
Reset							0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TRFCPB								TRFC_BIT7_4				TRRD_BIT2		TFAW_BIT4	TRC_BIT4
Type	RW								RW				RW		RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0

Bit(s)	Name	Description
25:24	TRPAB	**All-bank precharge timing for LPDDR2** tRPAB = TRP + TRPAB
23:16	REFRCNT	**Every refresh number to issue MRR commands for refresh rates, only for LPDDR2**
15:8	TRFCPB	**tRFCPB Timing setting** tRFCPB = (11 + TRFCPB[7:0]) DRAMC clock cycles
7:4	TRFC_BIT7_4	**tRFC Timing setting for bit 7 ~ 4**
3	TRRD_BIT2	**tRRD Timing setting for bit 2**
1	TFAW_BIT4	**tFAW Timing setting for bit 4**
0	TRC_BIT4	**tRC Timing setting for bit 4**

1E0051EC PERFCTL0 PERFORMANCE CONTROL 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																DISDMOEDIS

Type																RW
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				CS2RANK		RWAGEEN	RWLLATEN	RWHPRIEN	RWOFOWNUM			RWOFOEN				DUALSCHEN
Type				RW		RW	RW	RW	RW			RW				RW
Reset				0		0	0	0	0	0	0	0				0

Bit(s)	Name	Description
16	DISDMOEDIS	**For power saving, the self-refresh may disable the IO output enable** 0: enable the power saving function 1: disable the power saving function
12	CS2RANK	**CS0 is also applied to CS1** 0: disable 1: enable
10	RWAGEEN	**Support EMI read/write aging tag** 0: Not support 1: Support
9	RWLLATEN	**Support EMI read/write low-latency** 0: Not support 1: Support
8	RWHPRIEN	**Support EMI read/write high-priority** 0: Not support 1: Support
7:5	RWOFOWNUM	**Coniinous write transactions allowed**
4	RWOFOEN	**Enable read/write out of order control** 0: disable 1: enable
0	DUALSCHEN	**Enable dual schedulers, only effective under FDIV2 = 1** 0: disable 1: enable

1E0051F0 AC_DERATING AC TIME DERATING CONTROL 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		TRRD_DERATE					TRPAB_DERATE		TRP_DERATE				TRAS_DERATE			
Type		RW					RW		RW				RW			
Reset		0	0	0			0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				TRC_DERATE					TRCD_DERATE							ACDERATEEN
Type				RW					RW							RW
Reset				0	0	0	0	0	0	0	0	0				0

Bit(s)	Name	Description
30:28	TRRD_DERATE	**tRRD de-rate timing setting** tRRD = (1 + TRRD_DERATE) DRAMC clock cycles
25:24	TRPAB_DERATE	**All-bank precharge de-rate timing for LPDDR2**

		tRPAB = TRP_DERATE + TRPAB_DERATE
23:20	TRP_DERATE	**tRP de-rate timing setting** tRP = (1 + TRP_DERATE) DRAMC clock cycles
19:16	TRAS_DERATE	**tRAS de-rate timing setting** tRAS = (8 + TRAS_DERATE) DRAMC clock cycles
12:8	TRC_DERATE	**tRC de-rate timing setting** tRC = (8 + TRC_DERATE) DRAMC clock cycles
7:4	TRCD_DERATE	**tRCD de-rate timing setting** tRCD = (1 + TRCD_DERATE) DRAMC clock cycles
0	ACDERATEEN	**Enable LPDDR2 AC timing de-rating control, effective when REFRESH_RATE >= 6** 0: disable 1: enable

1E0051F4 RRRATE_CTL REFRESH RATE CONTROL 00020100

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name												RR_BIT2_SEL				
Type												RW				
Reset												0	0	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				RR_BIT1_SEL								RR_BIT0_SEL				
Type				RW								RW				
Reset				0	0	0	0	1				0	0	0	0	0

Bit(s)	Name	Description
20:16	RR_BIT2_SEL	**Refresh rate data bit 2 selection from 32-bit input read data** 00000: select bit 0 00001: select bit 1 00010: select bit 2 . 11111: select bit 31
12:8	RR_BIT1_SEL	**Refresh rate data bit 1 selection from 32-bit input read data** 00000: select bit 0 00001: select bit 1 00010: select bit 2 . 11111: select bit 31
4:0	RR_BIT0_SEL	**Refresh rate data bit 0 selection from 32-bit input read data** 00000: select bit 0 00001: select bit 1 00010: select bit 2 . 11111: select bit 31

1E0051F8 WPATCMP_DAT WRITE PATTERN COMPARE SETTING 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	WPATCMP[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	WPATCMP[15:0]															

Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	WPATCMP	**Write data pattern to be compared for interrupting write commands**

1E0051FC WPATCMP_CTL WRITE PATTERN COMPARE CONTROL 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name												WPAT_STCLR			WPAT_BLKCYC	
Type												RW			RW	
Reset												0			0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	WPAT_INVEN								WPAT_CMPEN							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
20	WPAT_STCLR	**Clear the write pattern hit counter WPAT_HIT_CNT (REG.3FC[7:0])**
17:16	WPAT_BLKCYC	**Block write command cycles during interruption** 0/1/2/3: 1/2/3/4T
15:8	WPAT_INVEN	**Inversion control for 8 sets of 32-bit write compare data WPATCMP**
7:0	WPAT_CMPEN	**Enable write data compare sequence** bit 0: PAT0, PAT1, PAT2, PAT3, PAT4, PAT5, PAT6, PAT7 bit 1: PAT7, PAT0, PAT1, PAT2, PAT3, PAT4, PAT5, PAT6 bit 2: PAT6, PAT7, PAT0, PAT1, PAT2, PAT3, PAT4, PAT5 . bit7: PAT1, PAT2, PAT3, PAT4, PAT5, PAT6, PAT7, PAT0

1E005200 DQODLY1 DQ output DELAY1 CHAIN setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQ7DLY				DQ6DLY				DQ5DLY				DQ4DLY			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQ3DLY				DQ2DLY				DQ1DLY				DQ0DLY			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:28	DQ7DLY	**DQ output delay chain setting for bit7** Unit: 20ps
27:24	DQ6DLY	**DQ output delay chain setting for bit6** Unit: 20ps
23:20	DQ5DLY	**DQ output delay chain setting for bit5** Unit: 20ps

19:16	DQ4DLY	**DQ output delay chain setting for bit4** Unit: 20ps
15:12	DQ3DLY	**DQ output delay chain setting for bit3** Unit: 20ps
11:8	DQ2DLY	**DQ output delay chain setting for bit2** Unit: 20ps
7:4	DQ1DLY	**DQ output delay chain setting for bit1** Unit: 20ps
3:0	DQ0DLY	**DQ output delay chain setting for bit0** Unit: 20ps

1E005204 DQODLY2 DQ output DELAY2 CHAIN setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQ15DLY				DQ14DLY				DQ13DLY				DQ12DLY			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQ11DLY				DQ10DLY				DQ9DLY				DQ8DLY			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:28	DQ15DLY	**DQ output delay chain setting for bit15** Unit: 20ps
27:24	DQ14DLY	**DQ output delay chain setting for bit14** Unit: 20ps
23:20	DQ13DLY	**DQ output delay chain setting for bit13** Unit: 20ps
19:16	DQ12DLY	**DQ output delay chain setting for bit12** Unit: 20ps
15:12	DQ11DLY	**DQ output delay chain setting for bit11** Unit: 20ps
11:8	DQ10DLY	**DQ output delay chain setting for bit10** Unit: 20ps
7:4	DQ9DLY	**DQ output delay chain setting for bit9** Unit: 20ps
3:0	DQ8DLY	**DQ output delay chain setting for bit8** Unit: 20ps

1E005208 DQODLY3 DQ output DELAY3 CHAIN setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQ23DLY				DQ22DLY				DQ21DLY				DQ20DLY			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQ19DLY				DQ18DLY				DQ17DLY				DQ16DLY			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:28	DQ23DLY	**DQ output delay chain setting for bit23** Unit: 20ps
27:24	DQ22DLY	**DQ output delay chain setting for bit22** Unit: 20ps
23:20	DQ21DLY	**DQ output delay chain setting for bit21** Unit: 20ps
19:16	DQ20DLY	**DQ output delay chain setting for bit20** Unit: 20ps
15:12	DQ19DLY	**DQ output delay chain setting for bit19** Unit: 20ps
11:8	DQ18DLY	**DQ output delay chain setting for bit18** Unit: 20ps
7:4	DQ17DLY	**DQ output delay chain setting for bit17** Unit: 20ps
3:0	DQ16DLY	**DQ output delay chain setting for bit16** Unit: 20ps

1E00520C DQODLY4 DQ output DELAY4 CHAIN setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQ31DLY				DQ30DLY				DQ29DLY				DQ28DLY			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQ27DLY				DQ26DLY				DQ25DLY				DQ24DLY			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:28	DQ31DLY	**DQ output delay chain setting for bit31** Unit: 20ps
27:24	DQ30DLY	**DQ output delay chain setting for bit30** Unit: 20ps
23:20	DQ29DLY	**DQ output delay chain setting for bit29** Unit: 20ps
19:16	DQ28DLY	**DQ output delay chain setting for bit28** Unit: 20ps
15:12	DQ27DLY	**DQ output delay chain setting for bit27** Unit: 20ps
11:8	DQ26DLY	**DQ output delay chain setting for bit26** Unit: 20ps
7:4	DQ25DLY	**DQ output delay chain setting for bit25** Unit: 20ps
3:0	DQ24DLY	**DQ output delay chain setting for bit24** Unit: 20ps

1E005210 DQIDLY1 DQ input DELAY1 CHAIN setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					DQ3DEL								DQ2DEL			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					DQ1DEL								DQ0DEL			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0

Bit(s)	Name	Description
27:24	DQ3DEL	**DQ input delay chain setting for bit3** Unit: 20ps
19:16	DQ2DEL	**DQ input delay chain setting for bit2** Unit: 20ps
11:8	DQ1DEL	**DQ input delay chain setting for bit1** Unit: 20ps
3:0	DQ0DEL	**DQ input delay chain setting for bit0** Unit: 20ps

1E005214 DQIDLY2 DQ input DELAY2 CHAIN setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					DQ7DEL								DQ6DEL			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					DQ5DEL								DQ4DEL			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0

Bit(s)	Name	Description
27:24	DQ7DEL	**DQ input delay chain setting for bit7** Unit: 20ps
19:16	DQ6DEL	**DQ input delay chain setting for bit6** Unit: 20ps
11:8	DQ5DEL	**DQ input delay chain setting for bit5** Unit: 20ps
3:0	DQ4DEL	**DQ input delay chain setting for bit4** Unit: 20ps

1E005218 DQIDLY3 DQ input DELAY3 CHAIN setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					DQ11DEL								DQ10DEL			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					DQ9DEL								DQ8DEL			

Type					RW								RW			
Reset					0	0	0	0					0	0	0	0

Bit(s)	Name	Description
27:24	DQ11DEL	**DQ input delay chain setting for bit11** Unit: 20ps
19:16	DQ10DEL	**DQ input delay chain setting for bit10** Unit: 20ps
11:8	DQ9DEL	**DQ input delay chain setting for bit9** Unit: 20ps
3:0	DQ8DEL	**DQ input delay chain setting for bit8** Unit: 20ps

1E00521C DQIDLY4 DQ input DELAY4 CHAIN setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					DQ15DEL								DQ14DEL			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					DQ13DEL								DQ12DEL			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0

Bit(s)	Name	Description
27:24	DQ15DEL	**DQ input delay chain setting for bit15** Unit: 20ps
19:16	DQ14DEL	**DQ input delay chain setting for bit14** Unit: 20ps
11:8	DQ13DEL	**DQ input delay chain setting for bit13** Unit: 20ps
3:0	DQ12DEL	**DQ input delay chain setting for bit12** Unit: 20ps

1E005220 DQIDLY5 DQ input DELAY5 CHAIN setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					DQ19DEL								DQ18DEL			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					DQ17DEL								DQ16DEL			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0

Bit(s)	Name	Description
27:24	DQ19DEL	**DQ input delay chain setting for bit19** Unit: 20ps
19:16	DQ18DEL	**DQ input delay chain setting for bit18** Unit: 20ps

11:8	DQ17DEL	**DQ input delay chain setting for bit17** Unit: 20ps
3:0	DQ16DEL	**DQ input delay chain setting for bit16** Unit: 20ps

1E005224 DQIDLY6 DQ input DELAY6 CHAIN setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					DQ23DEL								DQ22DEL			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					DQ21DEL								DQ20DEL			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0

Bit(s)	Name	Description
27:24	DQ23DEL	**DQ input delay chain setting for bit23** Unit: 20ps
19:16	DQ22DEL	**DQ input delay chain setting for bit22** Unit: 20ps
11:8	DQ21DEL	**DQ input delay chain setting for bit21** Unit: 20ps
3:0	DQ20DEL	**DQ input delay chain setting for bit20** Unit: 20ps

1E005228 DQIDLY7 DQ input DELAY7 CHAIN setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					DQ27DEL								DQ26DEL			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					DQ25DEL								DQ24DEL			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0

Bit(s)	Name	Description
27:24	DQ27DEL	**DQ input delay chain setting for bit27** Unit: 20ps
19:16	DQ26DEL	**DQ input delay chain setting for bit26** Unit: 20ps
11:8	DQ25DEL	**DQ input delay chain setting for bit25** Unit: 20ps
3:0	DQ24DEL	**DQ input delay chain setting for bit24** Unit: 20ps

1E00522C DQIDLY8 DQ input DELAY8 CHAIN setting 0000000

0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					DQ31DEL								DQ30DEL			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					DQ29DEL								DQ28DEL			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0

Bit(s)	Name	Description
27:24	DQ31DEL	**DQ input delay chain setting for bit31** Unit: 20ps
19:16	DQ30DEL	**DQ input delay chain setting for bit30** Unit: 20ps
11:8	DQ29DEL	**DQ input delay chain setting for bit29** Unit: 20ps
3:0	DQ28DEL	**DQ input delay chain setting for bit28** Unit: 20ps

1E005280 R2R_page_hit_counter R2R_page_hit_counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	R2R_page_hit_counter[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	R2R_page_hit_counter[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	R2R_page_hit_counter	**R2R_page_hit_counter**

1E005284 R2R_page_miss_counter R2R_page_miss_counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	R2R_page_miss_counter[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	R2R_page_miss_counter[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	R2R_page_miss_counter	**R2R_page_miss_counter**

1E005288 R2R_interbank_counter R2R_interbank_counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	R2R_interbank_counter[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	R2R_interbank_counter[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	R2R_interbank_counter	**R2R_interbank_counter**

1E00528C R2W_page_hit_counter R2W_page_hit_counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	R2W_page_hit_counter[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	R2W_page_hit_counter[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	R2W_page_hit_counter	**R2W_page_hit_counter**

1E005290 R2W_page_miss_counter R2W_page_miss_counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	R2W_page_miss_counter[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	R2W_page_miss_counter[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	R2W_page_miss_counter	**R2W_page_miss_counter**

1E005294 R2W_interbank_counter R2W_interbank_counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	R2W_interbank_counter[31:16]															

Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	R2W_interbank_counter[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	R2W_interbank_counter	**R2W_interbank_counter**

1E005298 W2R_page_hit_counter W2R_page_hit_counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	W2R_page_hit_counter[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	W2R_page_hit_counter[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	W2R_page_hit_counter	**W2R_page_hit_counter**

1E00529C W2R_page_miss_counter W2R_page_miss_counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	W2R_page_miss_counter[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	W2R_page_miss_counter[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	W2R_page_miss_counter	**W2R_page_miss_counter**

1E0052A0 W2R_interbank_counter W2R_interbank_counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	W2R_interbank_counter[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	W2R_interbank_counter[15:0]															
Type	RO															

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	W2R_interbank_counter	**W2R_interbank_counter**

1E0052A4 W2W_page_hit_counter W2W_page_hit_counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	W2W_page_hit_counter[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	W2W_page_hit_counter[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	W2W_page_hit_counter	**W2W_page_hit_counter**

1E0052A8 W2W_page_miss_counter W2W_page_miss_counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	W2W_page_miss_counter[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	W2W_page_miss_counter[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	W2W_page_miss_counter	**W2W_page_miss_counter**

1E0052AC W2W_interbank_counter W2W_interbank_counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	W2W_interbank_counter[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	W2W_interbank_counter[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	W2W_interbank_counter	**W2W_interbank_counter**

1E0052B0 dramc_idle_counter dramc_idle_counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	dramc_idle_counter[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	dramc_idle_counter[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	dramc_idle_counter	**dramc_idle_counter**

1E0052B4 freerun_26m_counter freerun_26m_counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	freerun_26m_counter[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	freerun_26m_counter[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	freerun_26m_counter	**freerun_26m_counter**

1E0052B8 refresh_pop_counter refresh_pop_counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	refresh_pop_counter[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	refresh_pop_counter[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	refresh_pop_counter	**refresh_pop_counter**

1E0052BC JMETER_ST Jitter Meter Status 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	JM	ONES_CNT														

	TR_DONE															
Type	RO	RO														
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		ZEROS_CNT														
Type		RO														
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	JMTR_DONE	**Jitter meter result is updated.** 0: not ready 1: update result.
30:16	ONES_CNT	**ones counter result**
14:0	ZEROS_CNT	**zeros counter result**

1E0052C0 DQ_CAL_MAX_0 DQ INPUT CALIBRATION per bit 3-0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQ0_3_DLY_MAX								DQ0_2_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQ0_1_DLY_MAX								DQ0_0_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQ0_3_DLY_MAX	**DQ bit3 input maximum delay**
23:16	DQ0_2_DLY_MAX	**DQ bit2 input maximum delay**
15:8	DQ0_1_DLY_MAX	**DQ bit1 input maximum delay**
7:0	DQ0_0_DLY_MAX	**DQ bit0 input maximum delay**

1E0052C4 DQ_CAL_MAX_1 DQ INPUT CALIBRATION per bit 7-4 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQ0_7_DLY_MAX								DQ0_6_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQ0_5_DLY_MAX								DQ0_4_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQ0_7_DLY_MAX	**DQ bit7 input maximum delay**
23:16	DQ0_6_DLY_MAX	**DQ bit6 input maximum delay**
15:8	DQ0_5_DLY_MAX	**DQ bit5 input maximum delay**
7:0	DQ0_4_DLY_MAX	**DQ bit4 input maximum delay**

1E0052C8 DQ_CAL_MAX_2 DQ INPUT CALIBRATION per bit 11-8 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQ1_3_DLY_MAX								DQ1_2_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQ1_1_DLY_MAX								DQ1_0_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQ1_3_DLY_MAX	**DQ bit11 input maximum delay**
23:16	DQ1_2_DLY_MAX	**DQ bit10 input maximum delay**
15:8	DQ1_1_DLY_MAX	**DQ bit9 input maximum delay**
7:0	DQ1_0_DLY_MAX	**DQ bit8 input maximum delay**

1E0052CC DQ_CAL_MAX_3 DQ INPUT CALIBRATION per bit 15-12 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQ1_7_DLY_MAX								DQ1_6_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQ1_5_DLY_MAX								DQ1_4_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQ1_7_DLY_MAX	**DQ bit15 input maximum delay**
23:16	DQ1_6_DLY_MAX	**DQ bit14 input maximum delay**
15:8	DQ1_5_DLY_MAX	**DQ bit13 input maximum delay**
7:0	DQ1_4_DLY_MAX	**DQ bit12 input maximum delay**

1E0052D0 DQ_CAL_MAX_4 DQ INPUT CALIBRATION per bit 19-16 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQ2_3_DLY_MAX								DQ2_2_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQ2_1_DLY_MAX								DQ2_0_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQ2_3_DLY_MAX	**DQ bit19 input maximum delay**

23:16	DQ2_2_DLY_MAX	**DQ bit18 input maximum delay**
15:8	DQ2_1_DLY_MAX	**DQ bit17 input maximum delay**
7:0	DQ2_0_DLY_MAX	**DQ bit16 input maximum delay**

1E0052D4 DQ_CAL_MAX_5 DQ INPUT CALIBRATION per bit 23-20 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQ2_7_DLY_MAX								DQ2_6_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQ2_5_DLY_MAX								DQ2_4_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQ2_7_DLY_MAX	**DQ bit23 input maximum delay**
23:16	DQ2_6_DLY_MAX	**DQ bit22 input maximum delay**
15:8	DQ2_5_DLY_MAX	**DQ bit21 input maximum delay**
7:0	DQ2_4_DLY_MAX	**DQ bit20 input maximum delay**

1E0052D8 DQ_CAL_MAX_6 DQ INPUT CALIBRATION per bit 27-34 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQ3_3_DLY_MAX								DQ3_2_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQ3_1_DLY_MAX								DQ3_0_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQ3_3_DLY_MAX	**DQ bit27 input maximum delay**
23:16	DQ3_2_DLY_MAX	**DQ bit26 input maximum delay**
15:8	DQ3_1_DLY_MAX	**DQ bit25 input maximum delay**
7:0	DQ3_0_DLY_MAX	**DQ bit24 input maximum delay**

1E0052DC DQ_CAL_MAX_7 DQ INPUT CALIBRATION per bit 31-28 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQ3_7_DLY_MAX								DQ3_6_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQ3_5_DLY_MAX								DQ3_4_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQ3_7_DLY_MAX	**DQ bit31 input maximum delay**
23:16	DQ3_6_DLY_MAX	**DQ bit30 input maximum delay**
15:8	DQ3_5_DLY_MAX	**DQ bit29 input maximum delay**
7:0	DQ3_4_DLY_MAX	**DQ bit28 input maximum delay**

1E0052E0 DQS_CAL_MIN_0 DQS INPUT CALIBRATION per bit 3-0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQS0_3_DLY_MIN								DQS0_2_DLY_MIN							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS0_1_DLY_MIN								DQS0_0_DLY_MIN							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQS0_3_DLY_MIN	**DQS bit3 input minimum delay**
23:16	DQS0_2_DLY_MIN	**DQS bit2 input minimum delay**
15:8	DQS0_1_DLY_MIN	**DQS bit1 input minimum delay**
7:0	DQS0_0_DLY_MIN	**DQS bit0 input minimum delay**

1E0052E4 DQS_CAL_MIN_1 DQS INPUT CALIBRATION per bit 7-4 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQS0_7_DLY_MIN								DQS0_6_DLY_MIN							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS0_5_DLY_MIN								DQS0_4_DLY_MIN							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQS0_7_DLY_MIN	**DQS bit7 input minimum delay**
23:16	DQS0_6_DLY_MIN	**DQS bit6 input minimum delay**
15:8	DQS0_5_DLY_MIN	**DQS bit5 input minimum delay**
7:0	DQS0_4_DLY_MIN	**DQS bit4 input minimum delay**

1E0052E8 DQS_CAL_MIN_2 DQS INPUT CALIBRATION per bit 11-8 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQS1_3_DLY_MIN								DQS1_2_DLY_MIN							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS1_1_DLY_MIN								DQS1_0_DLY_MIN							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQS1_3_DLY_MIN	**DQS bit11 input minimum delay**
23:16	DQS1_2_DLY_MIN	**DQS bit10 input minimum delay**
15:8	DQS1_1_DLY_MIN	**DQS bit9 input minimum delay**
7:0	DQS1_0_DLY_MIN	**DQS bit8 input minimum delay**

1E0052EC **DQS_CAL_MIN_3** **DQS INPUT CALIBRATION per bit 15-12** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQS1_7_DLY_MIN								DQS1_6_DLY_MIN							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS1_5_DLY_MIN								DQS1_4_DLY_MIN							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQS1_7_DLY_MIN	**DQS bit15 input minimum delay**
23:16	DQS1_6_DLY_MIN	**DQS bit14 input minimum delay**
15:8	DQS1_5_DLY_MIN	**DQS bit13 input minimum delay**
7:0	DQS1_4_DLY_MIN	**DQS bit12 input minimum delay**

1E0052F0 **DQS_CAL_MIN_4** **DQS INPUT CALIBRATION per bit 19-16** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQS2_3_DLY_MIN								DQS2_2_DLY_MIN							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS2_1_DLY_MIN								DQS2_0_DLY_MIN							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQS2_3_DLY_MIN	**DQS bit19 input minimum delay**
23:16	DQS2_2_DLY_MIN	**DQS bit18 input minimum delay**
15:8	DQS2_1_DLY_MIN	**DQS bit17 input minimum delay**
7:0	DQS2_0_DLY_MIN	**DQS bit16 input minimum delay**

1E0052F4 **DQS_CAL_MIN_5** **DQS INPUT CALIBRATION per bit 23-20** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQS2_7_DLY_MIN								DQS2_6_DLY_MIN							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS2_5_DLY_MIN								DQS2_4_DLY_MIN							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQS2_7_DLY_MIN	**DQS bit23 input minimum delay**
23:16	DQS2_6_DLY_MIN	**DQS bit22 input minimum delay**
15:8	DQS2_5_DLY_MIN	**DQS bit21 input minimum delay**
7:0	DQS2_4_DLY_MIN	**DQS bit20 input minimum delay**

1E0052F8 DQS_CAL_MIN_6 DQS INPUT CALIBRATION per bit 27-34 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQS3_3_DLY_MIN								DQS3_2_DLY_MIN							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS3_1_DLY_MIN								DQS3_0_DLY_MIN							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQS3_3_DLY_MIN	**DQS bit27 input minimum delay**
23:16	DQS3_2_DLY_MIN	**DQS bit26 input minimum delay**
15:8	DQS3_1_DLY_MIN	**DQS bit25 input minimum delay**
7:0	DQS3_0_DLY_MIN	**DQS bit24 input minimum delay**

1E0052FC DQS_CAL_MIN_7 DQS INPUT CALIBRATION per bit 31-28 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQS3_7_DLY_MIN								DQS3_6_DLY_MIN							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS3_5_DLY_MIN								DQS3_4_DLY_MIN							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQS3_7_DLY_MIN	**DQS bit31 input minimum delay**
23:16	DQS3_6_DLY_MIN	**DQS bit30 input minimum delay**
15:8	DQS3_5_DLY_MIN	**DQS bit29 input minimum delay**
7:0	DQS3_4_DLY_MIN	**DQS bit28 input minimum delay**

1E005300 DQS_CAL_MAX_0 DQS INPUT CALIBRATION per bit 3-0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQS0_3_DLY_MAX								DQS0_2_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS0_1_DLY_MAX								DQS0_0_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQS0_3_DLY_MAX	**DQS bit3 input maximum delay**
23:16	DQS0_2_DLY_MAX	**DQS bit2 input maximum delay**
15:8	DQS0_1_DLY_MAX	**DQS bit1 input maximum delay**
7:0	DQS0_0_DLY_MAX	**DQS bit0 input maximum delay**

1E005304 DQS_CAL_MAX_1 DQS INPUT CALIBRATION per bit 7-4 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQS0_7_DLY_MAX								DQS0_6_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS0_5_DLY_MAX								DQS0_4_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQS0_7_DLY_MAX	**DQS bit7 input maximum delay**
23:16	DQS0_6_DLY_MAX	**DQS bit6 input maximum delay**
15:8	DQS0_5_DLY_MAX	**DQS bit5 input maximum delay**
7:0	DQS0_4_DLY_MAX	**DQS bit4 input maximum delay**

1E005308 DQS_CAL_MAX_2 DQS INPUT CALIBRATION per bit 11-8 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQS1_3_DLY_MAX								DQS1_2_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS1_1_DLY_MAX								DQS1_0_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQS1_3_DLY_MAX	**DQS bit11 input maximum delay**
23:16	DQS1_2_DLY_MAX	**DQS bit10 input maximum delay**
15:8	DQS1_1_DLY_MAX	**DQS bit9 input maximum delay**

7:0	DQS1_0_DLY_MAX	**DQS bit8 input maximum delay**

1E00530C DQS_CAL_MAX_3 DQS INPUT CALIBRATION per bit 15-12 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQS1_7_DLY_MAX								DQS1_6_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS1_5_DLY_MAX								DQS1_4_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQS1_7_DLY_MAX	**DQS bit15 input maximum delay**
23:16	DQS1_6_DLY_MAX	**DQS bit14 input maximum delay**
15:8	DQS1_5_DLY_MAX	**DQS bit13 input maximum delay**
7:0	DQS1_4_DLY_MAX	**DQS bit12 input maximum delay**

1E005310 DQS_CAL_MAX_4 DQS INPUT CALIBRATION per bit 19-16 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQS2_3_DLY_MAX								DQS2_2_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS2_1_DLY_MAX								DQS2_0_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQS2_3_DLY_MAX	**DQS bit19 input maximum delay**
23:16	DQS2_2_DLY_MAX	**DQS bit18 input maximum delay**
15:8	DQS2_1_DLY_MAX	**DQS bit17 input maximum delay**
7:0	DQS2_0_DLY_MAX	**DQS bit16 input maximum delay**

1E005314 DQS_CAL_MAX_5 DQS INPUT CALIBRATION per bit 23-20 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQS2_7_DLY_MAX								DQS2_6_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS2_5_DLY_MAX								DQS2_4_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

31:24	DQS2_7_DLY_MAX	**DQS bit23 input maximum delay**
23:16	DQS2_6_DLY_MAX	**DQS bit22 input maximum delay**
15:8	DQS2_5_DLY_MAX	**DQS bit21 input maximum delay**
7:0	DQS2_4_DLY_MAX	**DQS bit20 input maximum delay**

1E005318 DQS_CAL_MAX_6 DQS INPUT CALIBRATION per bit 27-34 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQS3_3_DLY_MAX								DQS3_2_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS3_1_DLY_MAX								DQS3_0_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQS3_3_DLY_MAX	**DQS bit27 input maximum delay**
23:16	DQS3_2_DLY_MAX	**DQS bit26 input maximum delay**
15:8	DQS3_1_DLY_MAX	**DQS bit25 input maximum delay**
7:0	DQS3_0_DLY_MAX	**DQS bit24 input maximum delay**

1E00531C DQS_CAL_MAX_7 DQS INPUT CALIBRATION per bit 31-28 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DQS3_7_DLY_MAX								DQS3_6_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DQS3_5_DLY_MAX								DQS3_4_DLY_MAX							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DQS3_7_DLY_MAX	**DQS bit31 input maximum delay**
23:16	DQS3_6_DLY_MAX	**DQS bit30 input maximum delay**
15:8	DQS3_5_DLY_MAX	**DQS bit29 input maximum delay**
7:0	DQS3_4_DLY_MAX	**DQS bit28 input maximum delay**

1E005350 DQICAL0 DQS INPUT CALIBRATION 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		DQ3_DLY_MAX								DQ2_DLY_MAX						
Type		RO								RO						
Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		DQ1_DLY_MAX								DQ0_DLY_MAX						
Type		RO								RO						

Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0

Bit(s)	Name	Description
30:24	DQ3_DLY_MAX	**DQ byte3 input maximum delay**
22:16	DQ2_DLY_MAX	**DQ byte2 input maximum delay**
14:8	DQ1_DLY_MAX	**DQ byte1 input maximum delay**
6:0	DQ0_DLY_MAX	**DQ byte0 input maximum delay**

1E005354 DQICAL1 DQS INPUT CALIBRATION 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		DQS3_DLY_MIN								DQS2_DLY_MIN						
Type		RO								RO						
Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		DQS1_DLY_MIN								DQS0_DLY_MIN						
Type		RO								RO						
Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0

Bit(s)	Name	Description
30:24	DQS3_DLY_MIN	**DQS3 input minimum delay**
22:16	DQS2_DLY_MIN	**DQS2 input minimum delay**
14:8	DQS1_DLY_MIN	**DQS1 input minimum delay**
6:0	DQS0_DLY_MIN	**DQS0 input minimum delay**

1E005358 DQICAL2 DQS INPUT CALIBRATION 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		DQS3_DLY_MAX								DQS2_DLY_MAX						
Type		RO								RO						
Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		DQS1_DLY_MAX								DQS0_DLY_MAX						
Type		RO								RO						
Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0

Bit(s)	Name	Description
30:24	DQS3_DLY_MAX	**DQS3 input maximum delay**
22:16	DQS2_DLY_MAX	**DQS2 input maximum delay**
14:8	DQS1_DLY_MAX	**DQS1 input maximum delay**
6:0	DQS0_DLY_MAX	**DQS0 input maximum delay**

1E00535C DQICAL3 DQS INPUT CALIBRATION 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		DQS3_DLY_AVG								DQS2_DLY_AVG						
Type		RO								RO						

Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		DQS1_DLY_AVG								DQS0_DLY_AVG						
Type		RO								RO						
Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0

Bit(s)	Name	Description
30:24	DQS3_DLY_AVG	**DQS3 input delay average**
22:16	DQS2_DLY_AVG	**DQS2 input delay average**
14:8	DQS1_DLY_AVG	**DQS1 input delay average**
6:0	DQS0_DLY_AVG	**DQS0 input delay average**

1E005370 CMP_ERR CMP ERROR 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CMP_ERR[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CMP_ERR[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	CMP_ERR	**bitwise auto test fail**

1E005374 DQSIENDLY DQS INPUT GATING DELAY VALUE 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		DQS3IENDLY								DQS2IENDLY						
Type		RO								RO						
Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		DQS1IENDLY								DQS0IENDLY						
Type		RO								RO						
Reset		0	0	0	0	0	0	0		0	0	0	0	0	0	0

Bit(s)	Name	Description
30:24	DQS3IENDLY	**DQS input gating delay for DQS3**
22:16	DQS2IENDLY	**DQS input gating delay for DQS2**
14:8	DQS1IENDLY	**DQS input gating delay for DQS1**
6:0	DQS0IENDLY	**DQS input gating delay for DQS0**

1E00538C STBEN0 DQS RING COUNTER 0 00000003

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	STBEN0[31:16]															
Type	RO															

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	STBEN0[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Bit(s)	Name	Description
31:0	STBEN0	**DQS0 ring counter**

1E005390 STBEN1 DQS RING COUNTER 1 00000003

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	STBEN1[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	STBEN1[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Bit(s)	Name	Description
31:0	STBEN1	**DQS1 ring counter**

1E005394 STBEN2 DQS RING COUNTER 2 00000003

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	STBEN2[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	STBEN2[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Bit(s)	Name	Description
31:0	STBEN2	**DQS2 ring counter**

1E005398 STBEN3 DQS RING COUNTER 3 00000003

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	STBEN3[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	STBEN3[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Bit(s)	Name	Description

31:0	STBEN3	**DQS3 ring counter**

1E0053A0 DQSDLY0 DQS INPUT DELAY SETTING 0 0F0F0F0F

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		DEL3DLY								DEL2DLY						
Type		RO								RO						
Reset		0	0	0	1	1	1	1		0	0	0	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		DEL1DLY								DEL0DLY						
Type		RO								RO						
Reset		0	0	0	1	1	1	1		0	0	0	1	1	1	1

Bit(s)	Name	Description
30:24	DEL3DLY	**DQS input delay for DQS3**
22:16	DEL2DLY	**DQS input delay for DQS2**
14:8	DEL1DLY	**DQS input delay for DQS1**
6:0	DEL0DLY	**DQS input delay for DQS0**

1E0053B8 SPCMDRESP SPECIAL COMMAND RESPONSE 00000300

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																SREF_STATE
Type																RO
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						REFRESH_RATE					TCMD_RESPONSE	ZQC_RESPONSE	AREF_RESPONSE	PREA_RESPONSE	MRR_RESPONSE	MRW_RESPONSE
Type						RO					RO	RO	RO	RO	RO	RO
Reset						0	1	1			0	0	0	0	0	0

Bit(s)	Name	Description
16	SREF_STATE	**Self-refresh status** 0: not enter 1: enter
10:8	REFRESH_RATE	**Refresh rate reading from LPDDR2** 001: 4 x tREFI 010: 2 x tREFI 011: 1 x tREFI 101: 0.25 x tREFI 110: 2.25 x tREFI Others: Refer to LPDDR2 spec.
5	TCMD_RESPONSE	**TCMD command response**
4	ZQC_RESPONSE	**ZQC command response**
3	AREF_RESPONSE	**AREF command response**

2	PREA_RESPONSE	**PREA command response**
1	MRR_RESPONSE	**MRR command response**
0	MRW_RESPONSE	**MRW command response**

1E0053BC IORGCNT IO RING COUNTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	IO_RING_COUNTER															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	IO_RING_COUNTER_K															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	IO_RING_COUNTER	**270 degree I/O clock offset counter for group 5**
15:0	IO_RING_COUNTER_K	**180 degree I/O clock offset counter for group 5**

1E0053C0 DQSGNWCNT0 DQS GATING WINODW COUNTER 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	dqs1r_gating_counter								dqs1f_gating_counter							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	dqs0r_gating_counter								dqs0f_gating_counter							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	dqs1r_gating_counter	**rsing dqs gating counter for group 1**
23:16	dqs1f_gating_counter	**falling dqs gating counter for group 1**
15:8	dqs0r_gating_counter	**rsing dqs gating counter for group 0**
7:0	dqs0f_gating_counter	**falling dqs gating counter for group 0**

1E0053C4 DQSGNWCNT1 DQS GATING WINODW COUNTER 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	dqs3r_gating_counter								dqs3f_gating_counter							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	dqs2r_gating_counter								dqs2f_gating_counter							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	dqs3r_gating_counter	**rsing dqs gating counter for group 3**
23:16	dqs3f_gating_counter	**falling dqs gating counter for group 3**
15:8	dqs2r_gating_counter	**rsing dqs gating counter for group 2**
7:0	dqs2f_gating_counter	**falling dqs gating counter for group 2**

1E0053C8 DQSGNWCNT2 DQS GATING WINODW COUNTER 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	dqs0r_pre_gating_counter								dqs0f_pre_gating_counter							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	dqs0r_pos_gating_counter								dqs0f_pos_gating_counter							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	dqs0r_pre_gating_counter	**rsing pre dqs gating counter for group 0**
23:16	dqs0f_pre_gating_counter	**falling pre dqs gating counter for group 0**
15:8	dqs0r_pos_gating_counter	**rsing pos dqs gating counter for group 0**
7:0	dqs0f_pos_gating_counter	**falling pos dqs gating counter for group 0**

1E0053CC DQSGNWCNT3 DQS GATING WINODW COUNTER 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	dqs1r_pre_gating_counter								dqs1f_pre_gating_counter							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	dqs1r_pos_gating_counter								dqs1f_pos_gating_counter							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	dqs1r_pre_gating_counter	**rsing pre dqs gating counter for group 1**
23:16	dqs1f_pre_gating_counter	**falling pre dqs gating counter for group 1**
15:8	dqs1r_pos_gating_counter	**rsing pos dqs gating counter for group 1**
7:0	dqs1f_pos_gating_counter	**falling pos dqs gating counter for group 1**

1E0053D0 DQSGNWCNT DQS GATING WINODW COUNTER 4 0000000

	4															0
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	dqs2r_pre_gating_counter								dqs2f_pre_gating_counter							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	dqs2r_pos_gating_counter								dqs2f_pos_gating_counter							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	dqs2r_pre_gating_counter	**rsing pre dqs gating counter for group 2**
23:16	dqs2f_pre_gating_counter	**falling pre dqs gating counter for group 2**
15:8	dqs2r_pos_gating_counter	**rsing pos dqs gating counter for group 2**
7:0	dqs2f_pos_gating_counter	**falling pos dqs gating counter for group 2**

1E0053D4 DQSGNWCNT5 DQS GATING WINODW COUNTER 5 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	dqs3r_pre_gating_counter								dqs3f_pre_gating_counter							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	dqs3r_pos_gating_counter								dqs3f_pos_gating_counter							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	dqs3r_pre_gating_counter	**rsing pre dqs gating counter for group 3**
23:16	dqs3f_pre_gating_counter	**falling pre dqs gating counter for group 3**
15:8	dqs3r_pos_gating_counter	**rsing pos dqs gating counter for group 3**
7:0	dqs3f_pos_gating_counter	**falling pos dqs gating counter for group 3**

1E0053D8 DQSSAMPLEV DQS SAMPLE VALUE 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name							CMPCNT						sample_out1_D	sample_out1_D	sample_out1_D	sample_out1_D

													QS 3	QS 2	QS 1	QS 0
Type							RO						RO	RO	RO	RO
Reset							0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
9:4	CMPCNT	**CMP counter value**
3	sample_out1_DQS3	**Sampled value for DQS3** 0: late 1: early
2	sample_out1_DQS2	**Sampled value for DQS2** 0: late 1: early
1	sample_out1_DQS1	**Sampled value for DQS1** 0: late 1: early
0	sample_out1_DQS0	**Sampled value for DQS0** 0: late 1: early

1E0053DC DLLCNT0 DLL STATUS 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CMPOT															
Type	RO															
Reset	0															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Name	Description
31	CMPOT	**CMP pad calibration result**

1E0053E8 CKPHCNT CLOCK PHASE DETECTION RESULT 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CKPHCHKCNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	CKPHCHKCNT	**MEMPLL in/out clock phase detection result counter**

1E0053FC TESTRPT TEST AGENT STATUS 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				CALI_DONE_MON				LB_CMP_FAIL						DLE_CNT_OK		
Type				RO				RO						RO		
Reset				0				0						0		
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		DM_CMP_ERR				DM_CMP_CPT			WPAT_HIT_CNT							
Type		RO				RO			RO							
Reset		0				0			0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28	CALI_DONE_MON	**calibration result is updated, SW can disable calibration**
24	LB_CMP_FAIL	**Loop-back test mode compare fail**
18	DLE_CNT_OK	**DLE counter is right for test agent 2**
14	DM_CMP_ERR	**Read data compare error for test agent 2**
10	DM_CMP_CPT	**Read data compare ready for test agent 2**
7:0	WPAT_HIT_CNT	**Write pattern hit counter. Clear by WPAT_STCLR (REG.1FC[20])**

1E005600 MEMPLL0 MEMPLL REGISTER SETTING 0 D000000F

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_MEMPLL_FBDIV2		RG_MEMPLL_ACCEN	RG_MEMPLL_LF	RG_MEMPLL_BR	RG_MEMPLL_BP	RG_MEMPLL_FMEN	RG_MEMPLL_LVRODEN	RG_MEMPLL_DIVEN						RG_MEMPLL_MONCKEN	RG_MEMPLL_MONEN
Type	RW		RW	RW	RW	RW	RW	RW	RW						RW	RW
Reset	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		RG_MEMPLL_RST	RG_MEMPLL_POSDIV		RG_MEMPLL_PREDIV		RG_MEMPLL_CKCTRL		RG_MEMPLL_FBDIV							RG_MEMPLL_PWD
Type		RW	RW		RW		RW		RW							RW
Reset		0	0	0	0	0	0	0	0	0	0	0	1	1	1	1

Bit(s)	Name	Description
31:30	RG_MEMPLL_FBDIV2	**Feedback clock select** 2'b00: VCO/1 2'b01: VCO/2 2'b11: VCO/4

29	RG_MEMPLL_ACCEN	**Fast Slew Enable** 1'b0: Disable 1'b1: Enable
28	RG_MEMPLL_LF	**Frequency Band Control** always set 1
27	RG_MEMPLL_BR	**Resistance adjustment for Bandwidth** 1'b0: BW = Fref/10 1'b1: BW = Fref/20
26	RG_MEMPLL_BP	**Capacitance adjustment for Bandiwdth** 1'b0: When RG_APLL_BR=1'b0 1'b1: When RG_APLL_BR=1'b1
25	RG_MEMPLL_FMEN	**PLL REF/FB monitor clock enable** 1'b0: disable 1'b1: enable
24	RG_MEMPLL_LVRODEN	**REGV12 LVR overdrive enable** 1'b0: disable 1'b1: enable
23:18	RG_MEMPLL_DIVEN	**Time domain cap multiplication ratio** 3'd0: x1 3'd1: x2 3'd6: x64
17	RG_MEMPLL_MONCKEN	**Monitor clock enable** 1'b0: Disable 1'b1: Enable
16	RG_MEMPLL_MONEN	**Control voltage monitor enable** 1'b0: Disable 1'b1: Enable
14	RG_MEMPLL_RST	**PLL reset control** 1'b0: reset disable 1'b1: reset enable
13:12	RG_MEMPLL_POSDIV	**Post-divider ratio for single-phase output** 2'b00: VCO/1 2'b01: VCO/2 2'b11: VCO/4
11:10	RG_MEMPLL_PREDIV	**Pre-divider ratio** 2'b00: Fref = Fin/1 2'b01: Fref = Fin/2 2'b11: Fref = Fin/4
9:8	RG_MEMPLL_CKCTRL	**Fast Slew Time Control** 2'b00: 2^9 * Tin 2'b01: 2^8 * Tin 2'b10: 2^7 * Tin 2'b11: 2^6 * Tin
7:1	RG_MEMPLL_FBDIV	**Feedback divide ratio** 7'd0: /1 7'd1: /2 7'd127: /128
0	RG_MEMPLL_PWD	**Power Down** 1'b0: Power On 1'b1: Power Down (toggle from 1->0 to initialize)

1E005604 **MEMPLL1** **MEMPLL REGISTER SETTING 1** **C0000001**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Name	RG_DMSS_PCW_NCPO[22:7]															
Type	RW															
Reset	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_DMSS_PCW_NCPO[6:0]							RG_DMSS_PCW_NCPO_CHG						RG_MEMPLL_RST_DLY		RG_MEMPLL_VODEN
Type	RW							RW						RW		RW
Reset	0	0	0	0	0	0	0	0						0	0	1

Bit(s)	Name	Description
31:9	RG_DMSS_PCW_NCPO	**DDS NCPO PCW**
8	RG_DMSS_PCW_NCPO_CHG	**DDS PCW change asynchrounous clock**
2:1	RG_MEMPLL_RST_DLY	**ICO reset signal** 2'b00: reset delay min 2'b11: reset delay max
0	RG_MEMPLL_VODEN	**CHP OverDrive Enable** 1'b0: Disable 1'b1: Enable

1E005608 MEMPLL2 MEMPLL REGISTER SETTING 2 4AC6001C

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_DMSS_SSC_DELTA1								RG_DMSS_SSC_PHINI	RG_DMSS_SSC_TRI_EN	RG_DMSS_SSC_EN	RG_DMSS_MONCK_EN	RG_DMSS_PI_C			RG_DMSS_PI_RST_SEL
Type	RW								RW	RW	RW	RW	RW			RW
Reset	0	1	0	0	1	0	1	0	1	1	0	0	0	1	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_DMSS_PI_PL_EN	RG_DMSS_HF_EN	RG_DMSS_PREDIV2	RG_DMSS_FIFO_START_MAN	RG_DMSS_NCPO_EN	RG_DMSS_RSTB	RG_DMSS_PWDB	RG_MEMPLL_DDSEN	RG_DMSS_PCW_NCPO							
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0

Bit(s)	Name	Description
31:24	RG_DMSS_SSC_DELTA1	**DDS SSC first spread disturbance amplitude**

23	RG_DMSS_SSC_PHINI	**FNPLL SSC initial spreading direction** 1'b0: Upward 1'b1: Downward
22	RG_DMSS_SSC_TRI_EN	**DDS SSC modulation type** 1'b0: Square wave 1'b1: Triangular wave
21	RG_DMSS_SSC_EN	**DDS SSC enable** 1'b0: Disable 1'b1: Enable
20	RG_DMSS_MONCK_EN	**DDS monitor clock enable** 1'b0: Disable 1'b1: Enable
19:17	RG_DMSS_PI_C	**DMSS PI cap select** 0: 165f 1: 150f 2: 135f 3: 120f 4: 105f 5: 90f 6: 75f 7: 60f
16	RG_DMSS_PI_RST_SEL	**DDS PI reset selection** 0:analog reset 1:digital reset
15	RG_DMSS_PI_PL_EN	**DDS PI pull low function enable bar** 1'b0: Enable 1'b1: Disable
14	RG_DMSS_HF_EN	**DDS high frequency mode enable** 1'b0: When RG_LC_DDS_PREDIV2=1'b0 1'b1: When RG_LC_DDS_PREDIV2=1'b1
13	RG_DMSS_PREDIV2	**DDS predivider** 1'b0: /1 1'b1: /2
12	RG_DMSS_FIFO_START_MAN	**DDS FIFO enable** 1'b0: Disable 1'b1: Enable
11	RG_DMSS_NCPO_EN	**DDS NCPO enable** 1'b0: Disable 1'b1: Enable
10	RG_DMSS_RSTB	**DDS NCPO reset bar** 1'b0: Reset 1'b1: Enable
9	RG_DMSS_PWDB	**DDS power down bar** 1'b0: Power down 1'b1: Power on
8	RG_MEMPLL_DDSEN	**PLL DDS feedback enable** 1'b0: Integer-N mode 1'b1: Fractional-N mode
7:0	RG_DMSS_PCW_NCPO	**DDS NCPO PCW**

1E00560C **MEMPLL3** **MEMPLL REGISTER SETTING 3** **90004A00**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_DMSS_SSC_PRD								RG_DMSS_SSC_DELTA[15:8]							

Type	RW								RW							
Reset	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_DMSS_SSC_DELTA[7:0]								RG_DMSS_SSC_DELTA1							
Type	RW								RW							
Reset	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	RG_DMSS_SSC_PRD	**DDS SSC modulation period**
23:8	RG_DMSS_SSC_DELTA	**DDS SSC disturbance amplitude**
7:0	RG_DMSS_SSC_DELTA1	**DDS SSC first spread disturbance amplitude**

1E005610 MEMPLL4 MEMPLL REGISTER SETTING 4 000D0801

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_DMSS_REV								RG_MEMPLL_DIV_EN	RG_MEMPLL_DIV						
Type	RW								RW	RW						
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_DMSS_LVRODEN	RG_DMSS_FRAC_MUTE			RG_DMSS_SEL_EXT	RG_DMSS_POSTDIV2	RG_DMSS_CLK_PH_INV	RG_DMSS_LPF_EN	RG_DMSS_SSC_PRD							
Type	RW	RW			RW	RW	RW	RW	RW							
Reset	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
31:24	RG_DMSS_REV	**dummy reg**
23	RG_MEMPLL_DIV_EN	**Enable the Divider for (APLL+DDS)**
22:16	RG_MEMPLL_DIV	**Control bits of Divider for (APLL+DDS)** RG_MEMPLL_DIV[6]=1: /1 RG_MEMPLL_DIV[5:0]: /(N+2)
15	RG_DMSS_LVRODEN	**REGV12 LVR overdrive enable** 1'b0: disable 1'b1: enable
14:12	RG_DMSS_FRAC_MUTE	**REV**
11	RG_DMSS_SEL_EXT	**DDS output pulse width** 0: 1T 1: 2T
10	RG_DMSS_POSTDIV2	**DMSS output clock div by 2** 0: disable 1: enable
9	RG_DMSS_CLK_PH_INV	**DMSS phase inverter** 0: normal

		1: inverter
8	RG_DMSS_LPF_EN	**DMSS regualtor low pass filter enable** 0: disable 1: enable
7:0	RG_DMSS_SSC_PRD	**DDS SSC modulation period**

1E005614 MEMPLL5 MEMPLL REGISTER SETTING 5 50008013

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_MEMPLL2_FBDIV2		RG_MEMPLL2_ACCEN	RG_MEMPLL2_LF	RG_MEMPLL2_BR	RG_MEMPLL2_BP	RG_MEMPLL2_FMEN	RG_MEMPLL2_LVRODEN	RG_MEMPLL2_DIVEN						RG_MEMPLL2_MONCKEN	RG_MEMPLL2_MONEN
Type	RW		RW	RW	RW	RW	RW	RW	RW						RW	RW
Reset	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_MEMPLL2_EXFBDIV_EN	RG_MEMPLL2_RST	RG_MEMPLL2_POSDIV		RG_MEMPLL2_PREDIV		RG_MEMPLL2_CKCTRL		RG_MEMPLL2_FBDIV							RG_MEMPLL2_PWD
Type	RW	RW	RW		RW		RW		RW							RW
Reset	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1

Bit(s)	Name	Description
31:30	RG_MEMPLL2_FBDIV2	**Feedback clock select** 2'b00: VCO/1 2'b01: VCO/2 2'b11: VCO/4
29	RG_MEMPLL2_ACCEN	**Fast Slew Enable** 1'b0: Disable 1'b1: Enable
28	RG_MEMPLL2_LF	**Frequency Band Control** always set 1
27	RG_MEMPLL2_BR	**Resistance adjustment for Bandwidth** 1'b0: BW = Fref/10 1'b1: BW = Fref/20
26	RG_MEMPLL2_BP	**Capacitance adjustment for Bandiwdth** 1'b0: When RG_APLL_BR=1'b0 1'b1: When RG_APLL_BR=1'b1
25	RG_MEMPLL2_FMEN	**PLL REF/FB monitor clock enable** 1'b0: disable 1'b1: enable
24	RG_MEMPLL2_LVRODEN	**REGV12 LVR overdrive enable** 1'b0: disable 1'b1: enable
23:18	RG_MEMPLL2_DIVEN	**Time domain cap multiplication ratio** 3'd0: x1 3'd1: x2

		3'd6: x64
17	RG_MEMPLL2_MONCKEN	**Monitor clock enable** 1'b0: Disable 1'b1: Enable
16	RG_MEMPLL2_MONEN	**Control voltage monitor enable** 1'b0: Disable 1'b1: Enable
15	RG_MEMPLL2_EXFBDIV_EN	**Mux For Feeback clock** 1'b0: VCO loop 2'b1:outer loop
14	RG_MEMPLL2_RST	**PLL reset control** 1'b0: reset disable 1'b1: reset enable
13:12	RG_MEMPLL2_POSDIV	**Post-divider ratio for single-phase output** 2'b00: VCO/1 2'b01: VCO/2 2'b11: VCO/4
11:10	RG_MEMPLL2_PREDIV	**not use**
9:8	RG_MEMPLL2_CKCTRL	**Fast Slew Time Control** 2'b00: 2^9 * Tin 2'b01: 2^8 * Tin 2'b10: 2^7 * Tin 2'b11: 2^6 * Tin
7:1	RG_MEMPLL2_FBDIV	**Feedback divide ratio** 7'd0: /1 7'd1: /2 7'd127: /128
0	RG_MEMPLL2_PWD	**Power Down** 1'b0: Power On 1'b1: Power Down (toggle from 1->0 to initialize)

1E005618 MEMPLL6 MEMPLL REGISTER SETTING 6 00001001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_MEMPLL2_REV															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		RG_MEMPLL2_TEST_DIV		RG_MEMPLL2_FB_MCK_SEL	RG_MEMPLL2_LDO_LVRODEN	RG_MEMPLL2_M4PDIV		RG_MEMPLL2_M8PDIVMON_EN	RG_MEMPLL2_SEL_MON					RG_MEMPLL2_RST_DLY		RG_MEMPLL2_VODEN
Type		RW		RW	RW	RW		RW	RW					RW		RW
Reset		0	0	1	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
31:16	RG_MEMPLL2_REV	**dummy reg**
14:13	RG_MEMPLL2_TEST_	**Monitor clock divider for testmode**

	DIV	2'b00: /1 2'b01: /2 2'b10: forbidden 2'b11: /4
12	RG_MEMPLL2_FB_MCK_SEL	**Mux For Feeback clock** 1'b0: internal loop 2'b1:outer loop
11	RG_MEMPLL2_LDO_LVRODEN	**REGV12 LVR overdrive enable** 1'b0: disable 1'b1: enable
10:9	RG_MEMPLL2_M4PDIV	**Multi-phase divider ratio for 4-phase output** 2'b00: VCO/2 2'b01: VCO/4 2'b10: VCO/8
8	RG_MEMPLL2_M8PDIVMON_EN	
7:3	RG_MEMPLL2_SEL_MON	
2:1	RG_MEMPLL2_RST_DLY	**ICO reset signal** 2'b00: reset delay min 2'b11: reset delay max
0	RG_MEMPLL2_VODEN	**CHP OverDrive Enable** 1'b0: Disable 1'b1: Enable

1E00561C MEMPLL7 MEMPLL REGISTER SETTING 7 13000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_MEMPLL3_FBDIV							RG_MEMPLL3_PWD	RG_MEMPLL2_DL_REV							
Type	RW							RW	RW							
Reset	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_MEMPLL2_FB_DL								RG_MEMPLL2_REF_DL							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:25	RG_MEMPLL3_FBDIV	**Feedback divide ratio** 7'd0: /1 7'd1: /2 7'd127: /128
24	RG_MEMPLL3_PWD	**Power Down** 1'b0: Power On 1'b1: Power Down (toggle from 1->0 to initialize)
23:16	RG_MEMPLL2_DL_REV	**REV reg**
15:8	RG_MEMPLL2_FB_DL	**MEMPLL2 skew adjust between reference clock and feedback clock**
7:0	RG_MEMPLL2_REF_DL	**MEMPLL2 skew adjust between reference clock and feedback clock**

1E005620 MEMPLL8 MEMPLL REGISTER SETTING 8 01500080

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_MEMPLL3_SEL_MON					RG_MEMPLL3_RST_DLY		RG_MEMPLL3_VODEN	RG_MEMPLL3_FBDIV2		RG_MEMPLL3_ACCEN	RG_MEMPLL3_LF	RG_MEMPLL3_BR	RG_MEMPLL3_BP	RG_MEMPLL3_FMEN	RG_MEMPLL3_LVRODEN
Type	RW					RW		RW	RW		RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_MEMPLL3_DIVEN						RG_MEMPLL3_MONCKEN	RG_MEMPLL3_MONEN	RG_MEMPLL3_EXFBDIV_EN	RG_MEMPLL3_RST	RG_MEMPLL3_POSDIV		RG_MEMPLL3_PREDIV		RG_MEMPLL3_CKCTRL	
Type	RW						RW	RW	RW	RW	RW		RW		RW	
Reset	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:27	RG_MEMPLL3_SEL_MON	
26:25	RG_MEMPLL3_RST_DLY	**ICO reset signal** 2'b00: reset delay min 2'b11: reset delay max
24	RG_MEMPLL3_VODEN	**CHP OverDrive Enable** 1'b0: Disable 1'b1: Enable
23:22	RG_MEMPLL3_FBDIV2	**Feedback clock select** 2'b00: VCO/1 2'b01: VCO/2 2'b11: VCO/4
21	RG_MEMPLL3_ACCEN	**Fast Slew Enable** 1'b0: Disable 1'b1: Enable
20	RG_MEMPLL3_LF	**Frequency Band Control** always set 1
19	RG_MEMPLL3_BR	**Resistance adjustment for Bandwidth** 1'b0: BW = Fref/10 1'b1: BW = Fref/20
18	RG_MEMPLL3_BP	**Capacitance adjustment for Bandiwdth** 1'b0: When RG_APLL_BR=1'b0 1'b1: When RG_APLL_BR=1'b1
17	RG_MEMPLL3_FMEN	**PLL REF/FB monitor clock enable** 1'b0: disable 1'b1: enable
16	RG_MEMPLL3_LVRODEN	**REGV12 LVR overdrive enable** 1'b0: disable 1'b1: enable

15:10	RG_MEMPLL3_DIVEN	**Time domain cap multiplication ratio** 3'd0: x1 3'd1: x2 3'd6: x64
9	RG_MEMPLL3_MONCKEN	**Monitor clock enable** 1'b0: Disable 1'b1: Enable
8	RG_MEMPLL3_MONEN	**Control voltage monitor enable** 1'b0: Disable 1'b1: Enable
7	RG_MEMPLL3_EXFBDIV_EN	**Mux For Feeback clock** 1'b0: VCO loop 2'b1:outer loop
6	RG_MEMPLL3_RST	**PLL reset control** 1'b0: reset disable 1'b1: reset enable
5:4	RG_MEMPLL3_POSDIV	**Post-divider ratio for single-phase output** 2'b00: VCO/1 2'b01: VCO/2 2'b11: VCO/4
3:2	RG_MEMPLL3_PREDIV	**not use**
1:0	RG_MEMPLL3_CKCTRL	**Fast Slew Time Control** 2'b00: 2^9 * Tin 2'b01: 2^8 * Tin 2'b10: 2^7 * Tin 2'b11: 2^6 * Tin

1E005624 MEMPLL9 MEMPLL REGISTER SETTING 9 00000010

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_MEMPLL3_REF_DL								RG_MEMPLL3_REV[15:8]							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_MEMPLL3_REV[7:0]									RG_MEMPLL3_TEST_DIV		RG_MEMPLL3_FB_MCK_SEL	RG_MEMPLL3_LDO_LVRODEN	RG_MEMPLL3_M4PDIV		RG_MEMPLL3_M8PDIVMON_EN
Type	RW									RW		RW	RW	RW		RW
Reset	0	0	0	0	0	0	0	0		0	0	1	0	0	0	0

Bit(s)	Name	Description
31:24	RG_MEMPLL3_REF_DL	**MEMPLL3 skew adjust between reference clock and feedback clock**
23:8	RG_MEMPLL3_REV	**dummy reg**
6:5	RG_MEMPLL3_TEST_DIV	**Monitor clock divider for testmode** 2'b00: /1 2'b01: /2 2'b10: forbidden

		2'b11: /4
4	RG_MEMPLL3_FB_MCK_SEL	**Mux For Feeback clock** 1'b0: internal loop 2'b1:outer loop
3	RG_MEMPLL3_LDO_LVRODEN	**REGV12 LVR overdrive enable** 1'b0: disable 1'b1: enable
2:1	RG_MEMPLL3_M4PDIV	**Multi-phase divider ratio for 4-phase output** 2'b00: VCO/2 2'b01: VCO/4 2'b10: VCO/8
0	RG_MEMPLL3_M8PDIVMON_EN	

1E005628 MEMPLL10 MEMPLL REGISTER SETTING 10 80130000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_MEMPLL4_EXFBDIV_EN	RG_MEMPLL4_RST	RG_MEMPLL4_POSDIV		RG_MEMPLL4_PREDIV		RG_MEMPLL4_CKCTRL		RG_MEMPLL4_FBDIV							RG_MEMPLL4_PWD
Type	RW	RW	RW		RW		RW		RW							RW
Reset	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_MEMPLL3_DL_REV								RG_MEMPLL3_FB_DL							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	RG_MEMPLL4_EXFBDIV_EN	**Mux For Feeback clock** 1'b0: VCO loop 2'b1:outer loop
30	RG_MEMPLL4_RST	**PLL reset control** 1'b0: reset disable 1'b1: reset enable
29:28	RG_MEMPLL4_POSDIV	**Post-divider ratio for single-phase output** 2'b00: VCO/1 2'b01: VCO/2 2'b11: VCO/4
27:26	RG_MEMPLL4_PREDIV	**not use**
25:24	RG_MEMPLL4_CKCTRL	**Fast Slew Time Control** 2'b00: 2^9 * Tin 2'b01: 2^8 * Tin 2'b10: 2^7 * Tin 2'b11: 2^6 * Tin
23:17	RG_MEMPLL4_FBDIV	**Feedback divide ratio** 7'd0: /1 7'd1: /2 7'd127: /128
16	RG_MEMPLL4_PWD	**Power Down**

		1'b0: Power On 1'b1: Power Down (toggle from 1->0 to initialize)
15:8	RG_MEMPLL3_DL_REV	**REV reg**
7:0	RG_MEMPLL3_FB_DL	**MEMPLL3 skew adjust between reference clock and feedback clock**

1E00562C MEMPLL11 MEMPLL REGISTER SETTING 11 10015000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		RG_MEMPLL4_TEST_DIV		RG_MEMPLL4_FB_MCK_SEL	RG_MEMPLL4_LDO_LVRODEN	RG_MEMPLL4_M4PDIV		RG_MEMPLL4_M8PDIVMON_EN	RG_MEMPLL4_SEL_MON					RG_MEMPLL4_RST_DLY		RG_MEMPLL4_VODEN
Type		RW		RW	RW	RW		RW	RW					RW		RW
Reset		0	0	1	0	0	0	0	0	0	0	0	0	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_MEMPLL4_FBDIV2		RG_MEMPLL4_ACCEN	RG_MEMPLL4_LF	RG_MEMPLL4_BR	RG_MEMPLL4_BP	RG_MEMPLL4_FMEN	RG_MEMPLL4_LVRODEN	RG_MEMPLL4_DIVEN						RG_MEMPLL4_MONCKEN	RG_MEMPLL4_MONEN
Type	RW		RW	RW	RW	RW	RW	RW	RW						RW	RW
Reset	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
30:29	RG_MEMPLL4_TEST_DIV	**Monitor clock divider for testmode** 2'b00: /1 2'b01: /2 2'b10: forbidden 2'b11: /4
28	RG_MEMPLL4_FB_MCK_SEL	**Mux For Feeback clock** 1'b0: internal loop 2'b1:outer loop
27	RG_MEMPLL4_LDO_LVRODEN	**REGV12 LVR overdrive enable** 1'b0: disable 1'b1: enable
26:25	RG_MEMPLL4_M4PDIV	**Multi-phase divider ratio for 4-phase output** 2'b00: VCO/2 2'b01: VCO/4 2'b10: VCO/8
24	RG_MEMPLL4_M8PDIVMON_EN	
23:19	RG_MEMPLL4_SEL_MON	
18:17	RG_MEMPLL4_RST_DLY	**ICO reset signal** 2'b00: reset delay min 2'b11: reset delay max

16	RG_MEMPLL4_VODEN	**CHP OverDrive Enable** 1'b0: Disable 1'b1: Enable
15:14	RG_MEMPLL4_FBDIV2	**Feedback clock select** 2'b00: VCO/1 2'b01: VCO/2 2'b11: VCO/4
13	RG_MEMPLL4_ACCEN	**Fast Slew Enable** 1'b0: Disable 1'b1: Enable
12	RG_MEMPLL4_LF	**Frequency Band Control** always set 1
11	RG_MEMPLL4_BR	**Resistance adjustment for Bandwidth** 1'b0: BW = Fref/10 1'b1: BW = Fref/20
10	RG_MEMPLL4_BP	**Capacitance adjustment for Bandiwdth** 1'b0: When RG_APLL_BR=1'b0 1'b1: When RG_APLL_BR=1'b1
9	RG_MEMPLL4_FMEN	**PLL REF/FB monitor clock enable** 1'b0: disable 1'b1: enable
8	RG_MEMPLL4_LVRODEN	**REGV12 LVR overdrive enable** 1'b0: disable 1'b1: enable
7:2	RG_MEMPLL4_DIVEN	**Time domain cap multiplication ratio** 3'd0: x1 3'd1: x2 3'd6: x64
1	RG_MEMPLL4_MONCKEN	**Monitor clock enable** 1'b0: Disable 1'b1: Enable
0	RG_MEMPLL4_MONEN	**Control voltage monitor enable** 1'b0: Disable 1'b1: Enable

1E005630 MEMPLL12 MEMPLL REGISTER SETTING 12 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_MEMPLL4_FB_DL								RG_MEMPLL4_REF_DL							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_MEMPLL4_REV															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	RG_MEMPLL4_FB_DL	**MEMPLL4 skew adjust between reference clock and feedback clock**
23:16	RG_MEMPLL4_REF_DL	**MEMPLL4 skew adjust between reference clock and feedback clock**
15:0	RG_MEMPLL4_REV	**dummy reg**

1E005634 MEMPLL13 MEMPLL REGISTER SETTING 13 02005B00

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			RG_MEMPLL_TEST_DIV		RG_MEMPLL_REFMON		RG_MEMPLL_CKMON_PD	RG_MEMPLL_CKMON_AMPADJ			RG_MEMPLL_MONSEL					
Type			RW		RW		RW	RW			RW					
Reset			0	0	0	0	1	0	0	0	0	0	0	0	0	
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_MEMPLL_LDO_LVRODEN	RG_MEMPLL_SEL_CK	RG_MEMPLL_REFCK_SEL			RG_MEMPLL_REFCK_MONEN	RG_MEMPLL_BIAS_RST	RG_MEMPLL_BIAS_PWD	RG_MEMPLL4_DL_REV							
Type	RW	RW	RW			RW	RW	RW	RW							
Reset	0	1	0	1	1	0	1	1	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
29:28	RG_MEMPLL_TEST_DIV	**Monitor clock divider for testmode** 2'b00: /1 2'b01: /2 2'b10: forbidden 2'b11: /4
27:26	RG_MEMPLL_REFMON	**Monitor clock for testmode** 2'b01 Refernece clock
25	RG_MEMPLL_CKMON_PD	
24:22	RG_MEMPLL_CKMON_AMPADJ	
21:17	RG_MEMPLL_MONSEL	
15	RG_MEMPLL_LDO_LVRODEN	**REGV12 LVR overdrive enable** 1'b0: disable 1'b1: enable
14	RG_MEMPLL_SEL_CK	**Monitor clock for debug enable** 1'b0: Disable 1'b1: Enable
13:11	RG_MEMPLL_REFCK_SEL	**MEMPLL input clock selection** 000 : XTAL 001 : FNPLL+Divider
10	RG_MEMPLL_REFCK_MONEN	**MEMPLL2/3/4 refck monitor enable** 1'b0: disable 1'b1: enable
9	RG_MEMPLL_BIAS_RST	**Constant-Gm Bias Reset** 1'b0: For performance 1'b1: Reset for Fast Power On
8	RG_MEMPLL_BIAS_PWD	**Constant-Gm Bias Power Down** 1'b0:Power On

		1'b1: Power Down
7:0	RG_MEMPLL4_DL_REV	**REV reg**

1E005638 MEMPLL14 MEMPLL REGISTER SETTING 14 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name															RG_MEMPLL_TOP_REV[15:14]	
Type															RW	
Reset															0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_MEMPLL_TOP_REV[13:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0		

Bit(s)	Name	Description
17:2	RG_MEMPLL_TOP_REV	**dummy reg**

1E005640 MEMPLL_DIVIDER MEMPLL DIVIDER REGISTER CONTROL 00000003

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name											R_DMPLL2CLK_EN	R_DMALLCLK_EN		R_DM1PLL_SYNC_MODE	R_DMBYP_PLL3	R_DMBYP_PLL4
Type											RW	RW		RW	RW	RW
Reset											0	0		0	1	1

Bit(s)	Name	Description
5	R_DMPLL2CLK_EN	**Enable 4-phase output clocks of PLL core 2** 0: disable 1: enable
4	R_DMALLCLK_EN	**Enable 4-phase output clocks of all clocks** 0: disable 1: enable
2	R_DM1PLL_SYNC_MODE	**Synchronous mode under 1-PLL clock scheme** 0: asynchronous mode 1: synchronous mode
1	R_DMBYP_PLL3	**Bypass PLL core 3** 0: not bypass 1: bypass
0	R_DMBYP_PLL4	**Bypass PLL core 4** 0: not bypass

1: bypass

1E005644 VREF VREF setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	INTREF1_REFN		INTREF1_REFP		INTREF1_DS			EN_INTREF1	INTREF0_REFN		INTREF0_REFP		INTREF0_DS			EN_INTREF0
Type	RW		RW		RW			RW	RW		RW		RW			RW
Reset	0	0	0	0	0	0		0	0	0	0	0	0	0		0

Bit(s)	Name	Description
15:14	INTREF1_REFN	**Fine tune Vref to lower level Control for internal VREF 1** 00: weakest 11: strongest to lower level
13:12	INTREF1_REFP	**Fine tune Vref to higher level Control for internal VREF 1** 00: weakest 11: strongest to higher level
11:10	INTREF1_DS	**Current consumption control for internal VREF 1** 00: weakest 11: strongest current consumption
8	EN_INTREF1	**Internal VREF 1 Enalbe control** 1 : enable 0 : disable
7:6	INTREF0_REFN	**Fine tune Vref to lower level Control for internal VREF 0** 00: weakest 11: strongest to lower level
5:4	INTREF0_REFP	**Fine tune Vref to higher level Control for internal VREF 0** 00: weakest 11: strongest to higher level
3:2	INTREF0_DS	**Current consumption control for internal VREF 0** 00: weakest 11: strongest current consumption
0	EN_INTREF0	**Internal VREF 0 Enalbe control** 1 : enable 0 : disable

2.16 RBUS Matrix and QoS Arbiter

2.16.1 Features

- 8 channel QoS Arbiter
- Configurable Bandwidth and Duedate for each agent
- QoS classifier can be programmed for RR, BW RR, Fixed Priority and QoS Arb

2.16.2 Block Diagram

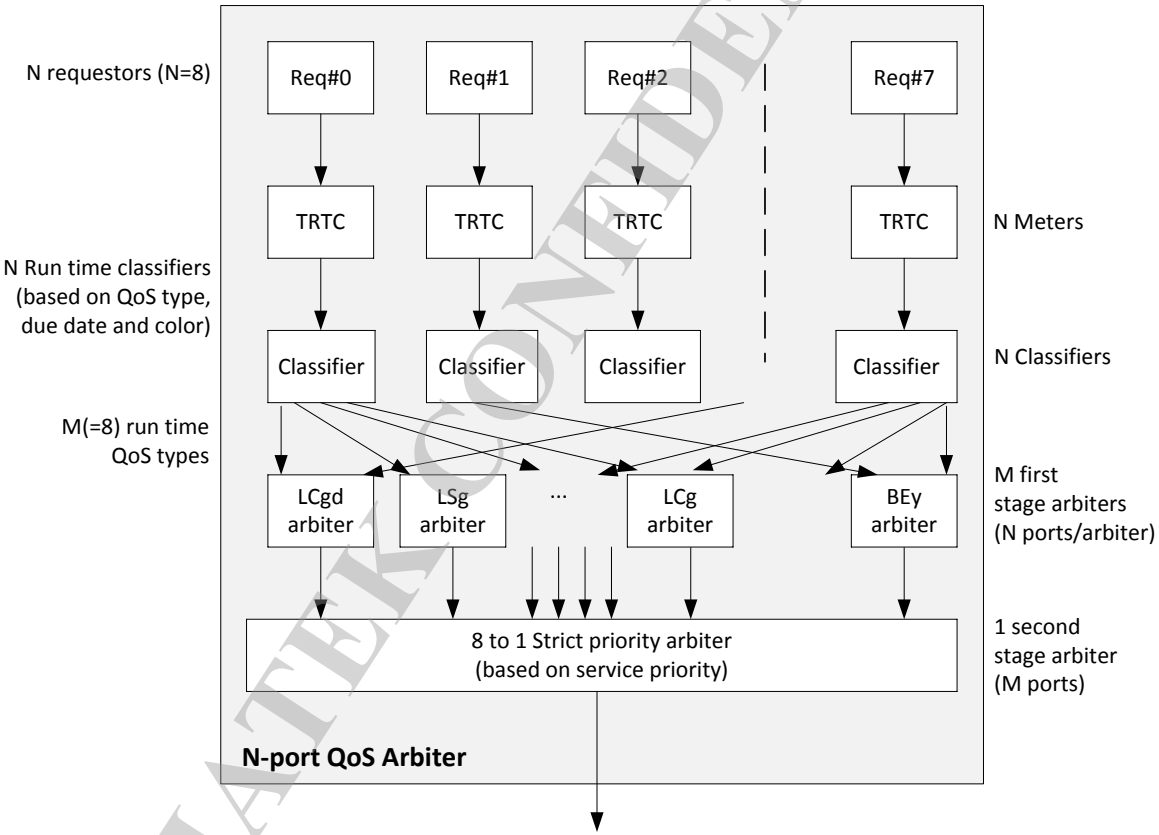


Figure 2-9 QoS Arbitration Block Diagram

2.16.3 Registers of QoS Control

DMA_CFG_ARB Changes LOG

Revision	Date	Author	Change Log
0.1	2012/10/5	Lancelot	Initialization
0.2	2012/10/22	Lancelot	Modify DMA debug message

Module name: DMA_CFG_ARB Base address: (+1E000800h)

Address	Name	Width	Register Function
1E000800	DMA_ARB_CFG	32	DMA 8 to 1 arbiter setting
1E000804	DMA_AG_BW	32	DMA Channel BW/QoS_Type/DueDate Setting
1E000808	DMA_AG_MAP	32	DMA channel (AG) mapping
1E00080C	DMA_ROUTE	32	DMA Routing
1E000810	DMA_DBG	32	DMA Debug
1E000814	DMA_STATE	32	DMA Debug State
1E000818	DMA_BW	32	DMA Bandwidth
1E00081C	DMA_LAT	32	DMA Latency
1E000820	R2P_MONITOR	32	Rbus to Pbus monitor
1E000824	R2P_ERR_ADDR	32	Rbus to Pbus ERR address

1E000800 DMA_ARB_CFG DMA 8 to 1 arbiter setting 04FAC688

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name						preempt_en	trtc_en	class_en	cls_priority[23:16]							
Type						RW	RW	RW	RW							
Reset						1	0	0	1	1	1	1	1	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	cls_priority[15:0]															
Type	RW															
Reset	1	1	0	0	0	1	1	0	1	0	0	0	1	0	0	0

Bit(s)	Name	Description
26	preempt_en	**Preemption Enable** Request preemption, higher priority requestor may change current request transaction 0: Disable Preemption 1: Enable Preemption
25	trtc_en	**Two Rate Three Color Bandwidth (TRTC) Meter Enable** 0: Disable TRTC 1: Enable TRTC
24	class_en	**QoS Classifier Enable** 0: Disable CLASS 1: Enable CLASS TRTC (0) CLASS (0) Round Robin TRTC (0) CLASS (1) Fixed Priority TRTC (1) CLASS (0) BW RR TRTC (1) CLASS (1) QoS Arb

23:0	cls_priority	**Class Priority** This field is used for class priority for second arbitration. {BEy(3'd7), LCg(3'd6), BSy(3'd5), LSy(3'd4), BEg (3'd3), BSg (3'd2), LSg(3'd1), LCgd(3'd0)}

1E000804 DMA_AG_BW DMA Channel BW/QoS_Type/DueDate Setting 02208020

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ag_wr	ag_sel					ag_qos_type		ag_duedate							
Type	WO	RW					RW		RW							
Reset	0	0	0	0			1	0	0	0	1	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ag_pir								ag_cir							
Type	RW								RW							
Reset	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

Bit(s)	Name	Description
31	ag_wr	**Agent Write** 0: Read 1: Write
30:28	ag_sel	**DMA Agent Select** Selects a DMA agent to configure. 0: SDXC/ND/SPDIF 1: HS_GDMS/GDMS/UTIF 2: USB 3.0 3: FE(0) 4: FE(1) 5: PPE 6: PCIe 7: Crypto
25:24	ag_qos_type	**Agent QoS Type** 0: Latency critical 1: Latency sensitive 2: Bandwidth sensitive (default) 3: Best Effort
23:16	ag_duedate	**Due date for latency critical agent** (unit: system bus clock cycle - system bus is 300 MHz or 225 MHz depending on bootstrap value.)
15:8	ag_pir	**Peak Information Rate for the Agent** The PIR is greater than or equal to the CIR. Bandwidth which exceeds PIR is marked red. 0x00: 0 MB/s 0x01: 4 MB/s ... 0x80: 512 MB/s (default) ... 0xFF: 1020 MB/s (Max)
7:0	ag_cir	**Committed Information Rate for the Agent** Bandwidth which falls below the CIR is marked green. BW which exceeds the CIR but is below the EIR is marked yellow. 0x00: 0 MB/s 0x01: 4 MB/s ... 0x20: 128 MB/s (default) ... 0xFF: 1020 MB/s (Max)

1E000808 DMA_AG_MAP DMA channel (AG) mapping 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									dma_ch_ag_map							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7:0	dma_ch_ag_map	**DMA AG map for 8 channel** 1'b0 -> This channel will go to DRAM arbiter 1'b1-> This channel will go to IOCU

1E00080C DMA_ROUTE DMA Routing 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																dma_route
Type																RW
Reset																0

Bit(s)	Name	Description
0	dma_route	**DMA routing** 0: DMA will access to DRAM 1: DMA will access to CSR

1E000810 DMA_DBG DMA Debug 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name														dma_sel		
Type														RW		
Reset														0	0	0

Bit(s)	Name	Description
2:0	dma_sel	**DMA channel select for debug message dump**

1E000814 DMA_STATE DMA Debug State 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						dma_rw	dma_state		dma_length							
Type						RO	RO		RO							
Reset						0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
10	dma_rw	**DMA channel RW**
9:8	dma_state	**DMA channel State** 2'b00: IDLE 2'b01: REQ 2'b10: ACK 2'b11: DATA
7:0	dma_length	**DMA channel burst length (Byte)**

1E000818 DMA_BW DMA Bandwidth 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	bw_rst		avg_bw										peak_bw[9:6]			
Type	WO		RO										RO			
Reset	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	peak_bw[5:0]						dma_bw									
Type	RO						RO									
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	bw_rst	**Write 1 will reset BW values.**
29:20	avg_bw	**Average BW (MB/S)**
19:10	peak_bw	**Peak BW (MB/S)**
9:0	dma_bw	**DMA channel BW (MB/S)**

1E00081C DMA_LAT DMA Latency 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	lat_rst		avg_lat										peak_lat[9:6]			
Type	WO		RO										RO			
Reset	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	peak_lat[5:0]						rd_lat									
Type	RO						RO									
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	lat_rst	**Write 1 will reset latency values**
29:20	avg_lat	**Average read latency (T)**
19:10	peak_lat	**Peak read latency (T)**
9:0	rd_lat	**DMA channel read latency (T)**

1E000820 R2P_MONITOR Rbus to Pbus monitor 00001023

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																r2p_inc_cnt
Type																W1C
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	r2p_err_cnt						r2p_inc_clr									
Type	RO						RW									
Reset	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1

Bit(s)	Name	Description
16	r2p_inc_cnt	**R2P Interrupt Clear** Write 1 to clear this interrupt.
15:10	r2p_err_cnt	**R2P error counter**
9:0	r2p_inc_clr	**R2P Interrupt Countdown Timer** Sets a delay timer which begins counting down when an R2P error is detected. When the timer reaches zero the R2P interrupt is then triggered. 10'd0: Disable R2P monitoring 10'd1: 20 us 10'd2: 40 us 10'd1023: 40 ms

1E000824 R2P_ERR_ADDR Rbus to Pbus ERR address 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	r2p_err_addr[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	r2p_err_addr[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	r2p_err_addr	**R2P address record for previous error found**

2.16.4 Registers of Rbus Matrix

Rbus_Matrix_CTRL Changes LOG

Revision	Date	Author	Change Log
0.1	2012/10/2	Lancelot	Initialization
0.2	2013/1/3	Lancelot	Add sleep count

Module name: Rbus_Matrix_CTRL Base address: (+1E000400h)

Address	Name	Width	Register Function
1E000400	OCP_CFG0	32	OCP to Rbus configuration
1E000404	OCP_CFG1	32	Read bypass write mask
1E000410	DYN_CFG0	32	Dynamic cpu/ocp frequency control
1E000414	DYN_CFG1	32	CPU sleep step frequency control
1E000418	DYN_CFG2	32	Dyn CFG Probe
1E00041C	DYN_CFG3	32	Sleep Counter
1E000420	IOCU_CFG	32	IOCU MreqInfo Setting

1E000400 OCP_CFG0 OCP to Rbus configuration 00000007

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													sync_method	ocp_sync_cmd	rbus_async	rd_bypass_wr
Type													RW	RW	RW	RW
Reset													0	1	1	1

Bit(s)	Name	Description
3	sync_method	**OCP Synchronization Command Method** 0: All empty (Wait until all FIFOs are empty) 1: CMD empty (Wait until the CMD FIFO is empty)
2	ocp_sync_cmd	**OCP Synchronization Command Method Enable** Remaps this RD CMD to address 0x0000_0000. Initiate DRAM control before enabling this option. 0: Disable 1: Enable
1	rbus_async	**Async Mode for RBUS** 0: Set HW to switch between sync or async mode dynamically. 1: Force RBUS to A.sync mode.
0	rd_bypass_wr	**Read Bypass Write Enable** Allows read commands to bypass write commands for OCP_IF when the address does not conflict. 0: Disable

1: Enable

1E000404 OCP_CFG1 Read bypass write mask FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	rd_bypass_wr_mask[31:16]															
Type	RW															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	rd_bypass_wr_mask[15:0]															
Type	RW															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:0	rd_bypass_wr_mask	**Mask bit for read bypass write address**

1E000410 DYN_CFG0 Dynamic cpu/ocp frequency control 00000A01

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					cpu_fdiv								cpu_ffrac			
Type					RW								RW			
Reset					1	0	1	0					0	0	0	1

Bit(s)	Name	Description
11:8	cpu_fdiv	**CPU Frequency Divider** The frequency divider is used to generate the CPU frequency. The value must be larger than or equal to CPU_FFRAC. Valid values range from 1 to 15.
3:0	cpu_ffrac	**CPU Frequency Fractional** A parameter used in conjunction with the CPU frequency divider to determine the CPU frequency. Input a value in the following equation to determine the CPU frequency. Valid values range from 0 to 15. CPU frequency = (CPU_FFRAC/CPU_FDIV)*PLL_FREQ NOTE: If the chip runs in USB OHCI mode, the OCP frequency cannot be lower than 30 MHz. It means that PLL_FREQ*(CPU_FFRAC/CPU_FDIV)/(CPU_OCP_RATIO+1) >= 30 MHz.

1E000414 DYN_CFG1 CPU sleep step frequency control 00200006

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	slp_en	step_en			step_cnt											
Type	RW	RW			RW											
Reset	0	0			0	0	0	0	0	0	1	0				
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													step_ffrac			
Type													RW			

Reset													0	1	1	0

Bit(s)	Name	Description
31	slp_en	**Sleep Mode Enable** Enables sleep mode when MIPS SI_Sleep is asserted. 0: Disable 1: Enable Sleep Mode CPU Frequency = (1/CPU_FDIV)*PLL_FREQ
30	step_en	**Step Jump Enable** Enables step jump after MIPS exits sleep mode. The CPU will jump to the normal frequency in increments defined by STEP_FFRAC.bit[4:0] of this register. 0: Disable 1: Enable
27:20	step_cnt	**Step Counter** Sets the period of each step jump. When the counter counts down to zero, the CPU clock automatically changes to the next step frequency. The count period unit is 1 us.
3:0	step_ffrac	**Step Frequency Fraction** Sets the fractional size of the increment in CPU frequency after the CPU exits from sleep mode and returns to normal operation. This step is only valid when SLP_STEP_EN is enabled. FRAC_VALUE = PREVIOUS_FRAC_VALUE + STEP_FFRAC CPU Frequency = (FRAC_VALUE/CPU_FDIV)*PLL_FREQ

1E000418 DYN_CFG2 Dyn CFG Probe 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	dyn_probe[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	dyn_probe[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	dyn_probe	**Dynamic freq probe** [26:25] dfc_fsm [18:16] cpu_ocp_ratio [11:8] cpu_fdiv [3:0] cpu_ffrac

1E00041C DYN_CFG3 Sleep Counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				slp_cnt_rst	Sleep_counter[27:16]											
Type				WO	RW											
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Sleep_counter[15:0]															

Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28	slp_cnt_rst	**Sleep Counter Reset**
27:0	Sleep_counter	**Sleep Counter**

1E000420 IOCU_CFG IOCU MreqInfo Setting 11111111

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	mreq_ag7				mreq_ag6				mreq_ag5				mreq_ag4			
Type	RW				RW				RW				RW			
Reset	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	mreq_ag3				mreq_ag2				mreq_ag1				mreq_ag0			
Type	RW				RW				RW				RW			
Reset	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1

Bit(s)	Name	Description
31:28	mreq_ag7	**IO_MreqInfo Setting for Ag7** Please reference AG0 configuration
27:24	mreq_ag6	**IO_MreqInfo Setting for Ag6** Please reference AG0 configuration
23:20	mreq_ag5	**IO_MreqInfo Setting for Ag5** Please reference AG0 configuration
19:16	mreq_ag4	**IO_MreqInfo Setting for Ag4** Please reference AG0 configuration
15:12	mreq_ag3	**IO_MreqInfo Setting for Ag3** Please reference AG0 configuration
11:8	mreq_ag2	**IO_MreqInfo Setting for Ag2** Please reference AG0 configuration
7:4	mreq_ag1	**IO_MreqInfo Setting for Ag1** Please reference AG0 configuration
3:0	mreq_ag0	**IO_MreqInfo Setting for Ag0** 0: Non-Coherent, L2 cacheable no L2 allocate 1: Coherent, cacheable no L2 allocate 2: Non-Coherent, uncacheable 3: Illegal 4: Non-Coherent, L2 cacheable, L2 allocate 5: Coherent, cacheable, L2 allocate 6: Illegal 7-15 Reserved

2.17 External MC Arbiter

2.17.1 Registers

EXT_MC_ARB Changes LOG

Revision	Date	Author	Change Log
0.1	2012/10/5	Lancelot	Initialization

Module name: EXT_MC_ARB Base address: (+1E006000h)

Address	Name	Width	Register Function
1E006000	MC_ARB_CFG	32	MC 2 to 1 arbiter setting
1E006004	MC_AG_BW	32	MC Channel BW/QoS_Type/DueDate Setting
1E006010	DRAM_SIZE	32	DRAM Size config

1E006000 MC_ARB_CFG MC 2 to 1 arbiter setting 07FAC688

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name						preempt_en	trtc_en	class_en	cls_priority[23:16]							
Type						RW	RW	RW	RW							
Reset						1	1	1	1	1	1	1	1	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	cls_priority[15:0]															
Type	RW															
Reset	1	1	0	0	0	1	1	0	1	0	0	0	1	0	0	0

Bit(s)	Name	Description
26	preempt_en	**Preemption Enable** Request preemption, higher priority requestor may change current request transaction 0: Disable Preemption 1: Enable Preemption
25	trtc_en	**Two Rate Three Color Bandwidth (TRTC) Meter Enable** 0: Disable TRTC 1: Enable TRTC
24	class_en	**QoS Classifier Enable** 0: Disable CLASS 1: Enable CLASS TRTC (0) CLASS (0) Round Robin TRTC (0) CLASS (1) Fixed Priority TRTC (1) CLASS (0) BW RR TRTC (1) CLASS (1) QoS Arb
23:0	cls_priority	**Class Priority** This field is used for class priority for second arbitration. {BEy(3'd7), LCg(3'd6), BSy(3'd5), LSy(3'd4), BEg (3'd3), BSg (3'd2), LSg(3'd1), LCgd(3'd0)}

1E006004 MC_AG_BW MC Channel BW/QoS_Type/DueDate Setting 0210FF40

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ag_wr	ag_sel					ag_qos_type		ag_duedate							
Type	WO	RW					RW		RW							
Reset	0	0	0	0			1	0	0	0	0	1	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ag_pir								ag_cir							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	0	1	0	0	0	0	0	0

Bit(s)	Name	Description
31	ag_wr	**Agent Write** 0: Read 1: Write
30:28	ag_sel	**DMA Agent Select** Selects a DMA agent to configure. 0: CPU (Rbus0) 1: DMA (Rbus1)
25:24	ag_qos_type	**Agent QoS Type** 0: Latency critical 1: Latency sensitive (CPU) 2: Bandwidth sensitive (DMA) 3: Best Effort
23:16	ag_duedate	**Due date for latency critical agent** (unit: system bus clock cycle - system bus is 300 MHz or 225 MHz depending on bootstrap value.)
15:8	ag_pir	**Peak Information Rate for the Agent** The PIR is greater than or equal to the CIR. Bandwidth which exceeds PIR is marked red. 0x00: 0 MB/s 0x01: 8 MB/s ... 0x40: 512 MB/s ... 0xFF: 2040 MB/s (Max)
7:0	ag_cir	**Committed Information Rate for the Agent** Bandwidth which falls below the CIR is marked green. BW which exceeds the CIR but is below the EIR is marked yellow. 0x00: 0 MB/s 0x01: 8 MB/s ... 0x40: 512 MB/s (default) ... 0xFF: 2040 MB/s (Max)

1E006010 DRAM_SIZE DRAM Size config 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																dram_size_1g
Type																RW
Reset																0

Bit(s)	Name	Description
0	dram_size_1g	**Dram size:** 0: dram size is small or equal to 512MB 1: dram size is equal to 1GB

2.18 Analog Macro Control

2.18.1 Registers

ANA_CTRL Changes LOG

Revision	Date	Author	Change Log
1.1	2012/2	Diary	draft
1.2	2012/11/28	Jeffrey	change base address and review.

Module name: ANA_CTRL Base address: (+1E000F00h)

Address	Name	Width	Register Function
1E000F00	XTAL_CTRL_1	32	XTAL control and status 1
1E000F04	XTAL_CTRL_2	32	XTAL control and status 2
1E000F08	XDRV_CTRL_1	32	XDRV control and status 1
1E000F0C	CBG_CTRL_1	32	CBG control and status 1
1E000F10	PLLGP_CTRL_1	32	PLLGP control and status 1
1E000F14	PLLGP_CTRL_2	32	PLLGP control and status 2
1E000F18	PLLGP_CTRL_3	32	PLLGP control and status 3
1E000F1C	PLLGP_CTRL_4	32	PLLGP control and status 4
1E000F20	PLLGP_CTRL_5	32	PLLGP control and status 5
1E000F24	PLLGP_CTRL_6	32	PLLGP control and status 6
1E000F28	PLLGP_CTRL_7	32	PLLGP control and status 7
1E000F2C	PLLGP_CTRL_8	32	PLLGP control and status 8
1E000F30	DA_CTRL_1	32	DA control and status 1

1E000F00 XTAL_CTRL_1 XTAL control and status 1 00008000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_XPTL_RESERVE2_D0															RG_XPTL_AMP_VTH[4:4]
Type	RW															RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0				0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_XPTL_AMP_VTH[3:0]				RG_XPTL_AMPMON_EN	RG_XPTL_CHG	RG_XPTL_CAP_ENB						RG_XPTL_VREG_LPF_ENB	RG_XPTL_GM_PWD		
Type	RW				RW	RW	RW						RW	RW		
Reset	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:20	RG_XPTL_RESERVE2	RG_XPTL_RESERVE2_D0

	_D0	12'b000000000000: NORMAL/SCAN/OLT mode
16:12	RG_XPTL_AMP_VTH	**XSTAL amplitude monitor threshold select** 00001: 0.9V 00010: 1.2V 00100: 1.5V 01000: 1.8V 10000: 2.1V
11	RG_XPTL_AMPMON_EN	**XSTAL amplitude monitor enable** 1'b1: NORMAL/SCAN/OLT mode
10	RG_XPTL_CHG	**GM pwd and CAP RG change** 1'b0: NORMAL/SCAN/OLT mode
9:4	RG_XPTL_CAP_ENB	**on-chip capacitor disable** 6'b000000: NORMAL/SCAN/OLT mode
3	RG_XPTL_VREG_LPF_ENB	**Regulator LPF disable** 1'b0: NORMAL/SCAN/OLT mode
2:0	RG_XPTL_GM_PWD	**XSTAL gain adjustment** 3'b000: NORMAL/SCAN/OLT mode

1E000F04 XTAL_CTRL_2 XTAL control and status 2 00000088

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			RG_XPTL_DIV2_EN	RG_XPTL_MON_DC_EN	RG_XPTL_MON_CK_EN	RG_XPTL_MON_DRV			RG_XPTL_DXP_CK_EN	RG_XPTL_DXP_DRV			RG_XPTL_AXP_CK_EN	RG_XPTL_AXP_DRV		
Type			RW	RW	RW	RW			RW	RW			RW	RW		
Reset			0	0	0	0	0	0	1	0	0	0	1	0	0	0

Bit(s)	Name	Description
13	RG_XPTL_DIV2_EN	1'b0: NORMAL/SCAN/OLT mode
12	RG_XPTL_MON_DC_EN	1'b0: NORMAL/SCAN/OLT mode
11	RG_XPTL_MON_CK_EN	1'b0: NORMAL/SCAN/OLT mode
10:8	RG_XPTL_MON_DRV	3'b000: NORMAL/SCAN/OLT mode
7	RG_XPTL_DXP_CK_EN	**enable rg for DXP ck** 1'b1: NORMAL/SCAN/OLT mode
6:4	RG_XPTL_DXP_DRV	**XSTAL gain adjustment** 3'b000: NORMAL/SCAN/OLT mode
3	RG_XPTL_AXP_CK_EN	**enable rg for AXP ck** 1'b1: NORMAL/SCAN/OLT mode
2:0	RG_XPTL_AXP_DRV	**XSTAL gain adjustment** 3'b000: NORMAL/SCAN/OLT mode

1E000F08 XDRV_CTRL_1 XDRV control and status 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name								RG_XDRV_EN								RG_XDRV_OBUF_EN
Type								RW								RW
Reset								0								0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		RG_XDRV_DRV			RG_XDRV_RSV											
Type		RW			RW											
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
24	RG_XDRV_EN	**CLOCK SQUARE PWD signal** 0: Clock square power down 1: Clock square power on 1'b1: NORMAL mode. 1'b0: SCAN/OLT mode.
16	RG_XDRV_OBUF_EN	**CLOCK SQUARE PWD signal** 0: Clock square power down 1: Clock square power on 1'b1: NORMAL mode. 1'b0: SCAN/OLT mode.
14:12	RG_XDRV_DRV	**XSTAL gain adjustment** 3'b000: NORMAL/SCAN/OLT mode
11:0	RG_XDRV_RSV	12'b000000000000: NORMAL/SCAN/OLT mode

1E000F0C CBG_CTRL_1 CBG control and status 1 00802157

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_CBG_AVOUTSEL				RG_CBG_SET				RG_CBG_TRIM							
Type	RW				RW				RW							
Reset	0	0	0	0	0	0	0	0	1	0	0	0				
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			RG_CBG_RASEL				RG_CBG_RBSEL		RG_CBG_CHPSET				RG_CBG_PWSET			
Type			RW				RW		RW				RW			
Reset			1	0			0	1	0	1	0	1	0	1	1	1

Bit(s)	Name	Description
31:28	RG_CBG_AVOUTSEL	**Analog monitor out selection** 0000: disbale 0001: 0010: 0011: 0100: USB3 (ADA_SSUSB_MONOUT) 0101: PLLGP (ADA_PLLGP_MON_DC) 0110: 0111: XPTL (ADA_XPTL_MON_DC) 1000: PCIe_x1 P2 (ADA_PE1_NS_VTOUT_P2) 1001: PCIe_x2 P1/P0 (ADA_PE1_NS_VTOUT) others: reserved

		4'b0000: NORMAL/SCAN/OLT mode
27:24	RG_CBG_SET	**CBG set** 4'b0000: NORMAL/SCAN/OLT mode
23:20	RG_CBG_TRIM	**Bandgap VRT reference voltage selection** ??? 4'b1000: NORMAL/SCAN/OLT mode
13:12	RG_CBG_RASEL	**Bandgap RA selection** 2'b10: NORMAL/SCAN/OLT mode
9:8	RG_CBG_RBSEL	**Bandgap RB selection** 2'b01: NORMAL/SCAN/OLT mode
7:4	RG_CBG_CHPSET	**Bandgap chopper set** [3] reserved [2] Enable 0: disable 1: enable [1:0] refclk dividing ratio 00: /2 01: /4 10: /8 11: /16 4'b0101: NORMAL/SCAN/OLT mode
3:0	RG_CBG_PWSET	**Bandgap power force mode set** [3] force mode enable [2] Bandgap power down bar [1] Current mirror power down bar [0] Low pass enable 4'b0111: NORMAL/SCAN/OLT mode

1E000F10 PLLGP_CTRL_1 PLLGP control and status 1 801510E0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_GMPLL_PWD									RG_GMPLL_FBDIV						
Type	RW									RW						
Reset	1									0	0	1	0	1	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			RG_GMPLL_PREDIV				RG_GMPLL_POSDIV		RG_GMPLL_CKCTRL		RG_GMPLL_FBSEL			RG_GMPLL_DIVEN		
Type			RW				RW		RW		RW			RW		
Reset			0	1			0	0	1	1	1	0		0	0	0

Bit(s)	Name	Description
31	RG_GMPLL_PWD	**Power Down** 1'b0: Power On 1'b1: Power Down 1'b1: NORMAL/SCAN mode. 1'b0: OLT mode.
22:16	RG_GMPLL_FBDIV	**Feedback divide ratio (N+1 Divider)** 7'd0: /1 7'd1: /2 7'd127: /128
13:12	RG_GMPLL_PREDIV	**Pre-divider ratio** 2'b00: Fref = Fin/1

		2'b01: Fref = Fin/2 2'b1X: Fref = Fin/4
9:8	RG_GMPLL_POSDIV	**Post-divider ratio for single-phase output** 2'b00: VCO/1 2'b01: VCO/2 2'b1X: VCO/4
7:6	RG_GMPLL_CKCTRL	**Fast Slew Time Control** 2'b00: 2^9 * Tin 2'b01: 2^8 * Tin 2'b10: 2^7 * Tin 2'b11: 2^6 * Tin
5:4	RG_GMPLL_FBSEL	**Feedback clock select** (Fvco output < 700MHz => 2'b00 Fvco output > 700MHz => 2'b01) 2'b00: Fvco/1 2'b01: Fvco/2 2'b1X: Fvco/4
2:0	RG_GMPLL_DIVEN	**Time domain cap multiplication ratio** 3'd0: x1 3'd1: x2 3'd6: x64

1E000F14 PLLGP_CTRL_2 PLLGP control and status 2 01401D61

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_GMPLL_BP	RG_GMPLL_BR	RG_GMPLL_HF	RG_GMPLL_LF	RG_GMPLL_FPEN	RG_GMPLL_MONEN	RG_GMPLL_ACCEN	RG_GMPLL_VODEN		RG_GMPLL_MONCKEN		RG_GMPLL_FMEN	RG_GMPLL_RESERVE			
Type	RW	RW	RW	RW	RW	RW	RW	RW		RW		RW	RW			
Reset	0	0	0	0	0	0	0	1		1		0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		RG_APLL_FBDIV							RG_APLL_PREDIV		RG_APLL_POSDIV		RG_APLL_CKCTRL		RG_APLL_FBSEL	
Type		RW							RW		RW		RW		RW	
Reset		0	0	1	1	1	0	1	0	1	1	0	0	0	0	1

Bit(s)	Name	Description
31	RG_GMPLL_BP	**Capacitance adjustment for Bandiwdth** 1'b0: When RG_APLL_BR=1'b0 1'b1: When RG_APLL_BR=1'b1 1'b0: When RG_APLL_BR=1'b0 1'b1: When RG_APLL_BR=1'b1
30	RG_GMPLL_BR	**Resistance adjustment for Bandwidth** 1'b0: BW = Fref/10 1'b1: BW = Fref/20 1'b0: BW = Fref/10 1'b1: BW = Fref/20
29	RG_GMPLL_HF	**Boost to 1.4GHz don't quarantee MP** 1'b0: Normal 1'b1: 1.4GHz Band 1'b0: Normal 1'b1: 1.4GHz Band
28	RG_GMPLL_LF	**Frequency Band Control**

		1'b0: Band > 700MHz 1'b1: Band < 700MHz 1'b0: Band > 700MHz 1'b1: Band < 700MHz
27	RG_GMPLL_FPEN	**PLL four phase output enable** 1'b0: 2 phase output 1'b1: 4 phase output 1'b0: 2 phase output 1'b1: 4 phase output
26	RG_GMPLL_MONEN	**Monitor for debug enable** 1'b0: Disable 1'b1: Enable 1'b0: Disable 1'b1: Enable
25	RG_GMPLL_ACCEN	**Fast Slew Enable** 1'b0: Disable 1'b1: Enable 1'b0: Disable 1'b1: Enable
24	RG_GMPLL_VODEN	**CHP OverDrive Enable** 1'b0: Disable 1'b1: Enable 1'b0: Disable, and OLT mode 1'b1: Enable
22	RG_GMPLL_MONCKEN	
20	RG_GMPLL_FMEN	**PLL Ref/FB monitor clock enable** 1'b0: Disable 1'b1: Enable 1'b0: Disable 1'b1: Enable
19:16	RG_GMPLL_RESERVE	
14:8	RG_APLL_FBDIV	**Feedback divide ratio** 7'd0: /1 7'd1: /2 7'd127: /128 7'd0: /1 7'd1: /2 7'd127: /128
7:6	RG_APLL_PREDIV	**Pre-divider ratio** 2'b00: /1 2'b01: /2 2'b1X: /4 2'b00: /1 2'b01: /2 2'b1X: /4
5:4	RG_APLL_POSDIV	**Post-divider ratio** 2'b00: /1 2'b01: /2 2'b1X: /4 2'b00: /1 2'b01: /2 2'b1X: /4
3:2	RG_APLL_CKCTRL	**Fast Slew & K-Band Time Control** 2'b00: 2^9 * Tref 2'b01: 2^8 * Tref 2'b10: 2^7 * Tref 2'b11: 2^6 * Tref 2'b00: 2^9 * Tref 2'b01: 2^8 * Tref

		2'b10: 2^7 * Tref 2'b11: 2^6 * Tref
1:0	RG_APLL_FBSEL	**Feedback clock select** 2'b00: Fvco/1 2'b01: Fvco/2 2'b1X: Fvco/4 (Fvco output < 700MHz => 2'b00 Fvco output > 700MHz => 2'b01) 2'b00: Fvco/1 2'b01: Fvco/2 2'b1X: Fvco/4

1E000F18 PLLGP_CTRL_3 PLLGP control and status 3 38233D0E

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_APLL_BP				RG_APLL_FPEN	RG_APLL_DIVEN				RG_APLL_BR				RG_APLL_BIC		
Type	RW				RW	RW				RW				RW		
Reset	0	0	1	1	1	0	0	0		0	1	0		0	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			RG_APLL_BC		RG_APLL_BIR							RG_APLL_ACCEN	RG_APLL_AUTOK_VCO	RG_APLL_AUTOK_LOAD	RG_APLL_LOAD_RSTB	
Type			RW		RW							RW	RW	RW	RW	
Reset			1	1	1	1	0	1				0	1	1	1	

Bit(s)	Name	Description
31:28	RG_APLL_BP	**P-path capacitance adjustment** MSB=1pF LSB=125fF
27	RG_APLL_FPEN	**PLL four phase output enable** 1'b0: 2 phase output 1'b1: 4 phase output 1'b0: 2 phase output 1'b1: 4 phase output
26:24	RG_APLL_DIVEN	**Time domain cap multiplication ratio** 3'd0: x1 3'd1: x2 3'd6: x64 3'd0: x1 3'd1: x2 3'd6: x64
22:20	RG_APLL_BR	**P-path resistance adjustment** 100:20kohm 010:40kohm 001:60kohm 000:80kohm 100: 20kohm 010: 40kohm 001: 60kohm 000: 80kohm

18:16	RG_APLL_BIC	**I-path current adjustment** 6.25u 6.25u 3.125u
13:12	RG_APLL_BC	**I-path capacitance adjustment** 00:0.5pF 01:1pF 10:1.5pF 11:2pF 00: 0.5pF 01: 1pF 10: 1.5pF 11: 2pF
11:8	RG_APLL_BIR	**P-path current adjustment** 25u 25u 12.5u 6.25u
4	RG_APLL_ACCEN	**Fast Slew Enable** 1'b0: Disable 1'b1: Enable 1'b0: Disable 1'b1: Enable
3	RG_APLL_AUTOK_VCO	**Band calibration enable** 1'b0: Disable 1'b1: Enable 1'b0: Disable 1'b1: Enable
2	RG_APLL_AUTOK_LOAD	**Load last-time band calibration result** 1'b0: Use register setting 1'b1: Load 1'b0: Use register setting 1'b1: Load
1	RG_APLL_LOAD_RSTB	**Band calibration result register reset** 1'b0: Reset 1'b1: Normal 1'b0: Reset 1'b1: Normal

1E000F1C PLLGP_CTRL_4 PLLGP control and status 4 80120004

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_APLL_PWD		RG_APLL_BAND									RG_APLL_VODEN	RG_APLL_CKSEL	RG_APLL_FMEN	RG_APLL_DDSEN	RG_APLL_RESERVE[4:4]
Type	RW		RW									RW	RW	RW	RW	RW
Reset	1		0	0	0	0	0	0				1	0	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_APLL_RESERVE[3:0]							RG_APLL_DDS_RSTB	RG_APLL_NCPO_EN	RG_APLL_PCW_NCPO_CHG	RG_APLL_FIFO_START_MAN	RG_APLL_DDS_HF_EN	RG_APLL_DDS_PREDIV2	RG_APLL_DDS_PI_PL_ENB	RG_APLL_DDS_RST_SEL	RG_APLL_DDS_PWDB
Type	RW							RW	RW	RW	RW	RW	RW	RW	RW	RW

Reset	0	0	0	0				0	0	0	0	0	0	1	0	0

Bit(s)	Name	Description
31	RG_APLL_PWD	**Power Down** 1'b0: Power On 1'b1: Power Down 1'b0: Power On/ OLT mode 1'b1: Power Down/ SCAN mode
29:24	RG_APLL_BAND	**Manual PLL Band Selection** 6'b000001: Lowest Band 6'b111111: Highest Band 6'b000001: Lowest Band 6'b111111: Highest Band
20	RG_APLL_VODEN	**CHP OverDrive Enable** 1'b0: Disable 1'b1: Enable 1'b0: Disable/OLT mode 1'b1: Enable
19	RG_APLL_CKSEL	**00: XTAL_CK = 40MHz** 01: ARMPLL_CK 00: XTAL_CK = 40MHz 01: ARMPLL_CK
18	RG_APLL_FMEN	**PLL Ref/FB monitor clock enable** 1'b0: Disable 1'b1: Enable 1'b0: Disable 1'b1: Enable
17	RG_APLL_DDSEN	**DDS Feedback Enable** 1'b0: Disable 1'b1: Enable 1'b0: Disable 1'b1: Enable
16:12	RG_APLL_RESERVE	
8	RG_APLL_DDS_RSTB	**APLL_DDS NCPO PI reset bar** 1'b1: SCAN / OLT mode
7	RG_APLL_NCPO_EN	**APLL_DDS NCPO enable** 1'b1: SCAN / OLT mode
6	RG_APLL_PCW_NCPO_CHG	**APLL_DDS PCW asynchrounous clock**
5	RG_APLL_FIFO_START_MAN	**APLL_DDS FIFO manual start**
4	RG_APLL_DDS_HF_EN	**APLL_DDS high frequency mode enable**
3	RG_APLL_DDS_PREDIV2	**APLL_DDS predivide by 2**
2	RG_APLL_DDS_PI_PL_ENB	**APLL_DDS PI pull low function enable bar**
1	RG_APLL_DDS_RST_SEL	**APLL_DDS PI reset selection (0:INA/INB 1:DIG_RST)**
0	RG_APLL_DDS_PWDB	**APLL_DDS PI power down bar** 1'b1: OLT mode. 1'b0: SCAN mode.

1E000F20 **PLLGP_CTRL_5** **PLLGP control and status 5** **1C7DBF48**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		RG_APLL_PCW_NCPO[30:16]														
Type		RW														
Reset		0	0	1	1	1	0	0	0	1	1	1	1	1	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_APLL_PCW_NCPO[15:0]															
Type	RW															
Reset	1	0	1	1	1	1	1	1	0	1	0	0	1	0	0	0

Bit(s)	Name	Description
30:0	RG_APLL_PCW_NCPO	**APLL_DDS NCPO PCW**

1E000F24 PLLGP_CTRL_6 PLLGP control and status 6 01484001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name								RG_APLL_CLK_PH_INV	RG_APLL_DDS_PI_C				RG_PLLGP_BIAS_RST	RG_PLLGP_BIAS_PWD	RG_PLLGP_ABIST_DIV1	RG_PLLGP_ABIST_DIV2
Type								RW	RW				RW	RW	RW	RW
Reset								1	0	1	0	0	1	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_PLLGP_TEST_EN	RG_PLLGP_ABIST_PWD	RG_PLLGP_ABIST_DIV						RG_PLLGP_MONSEL							RG_PLLGP_SEL_CK
Type	RW	RW	RW						RW							RW
Reset	0	1	0	0	0	0	0	0	0	0	0	0				1

Bit(s)	Name	Description
24	RG_APLL_CLK_PH_INV	**APLL_DDS clock inversion**
23:20	RG_APLL_DDS_PI_C	**APLL_DDS PI capacitor adjustment**
19	RG_PLLGP_BIAS_RST	**Constant-Gm Bias Reset** 1'b0: Use bias LPF for reference current 1'b1: Bybpass bias LPF initially
18	RG_PLLGP_BIAS_PWD	**Constant-Gm Bias Power Down** 1'b0:Power On 1'b1: Power Down 1'b0: OLT mode. 1'b1: SCAN mode.
17	RG_PLLGP_ABIST_DIV1	**ABIST clock div 1**
16	RG_PLLGP_ABIST_DIV2	**ABIST clock div 2**
15	RG_PLLGP_TEST_EN	**PLL group test mode enable** 0:disable 1:enable
14	RG_PLLGP_ABIST_P	**PLL group ABIST power down**

	WD	0: normal 1: power down 1'b0: OLT mode. 1'b1: SCAN mode.
13:8	RG_PLLGP_ABIST_DIV	**Selected clock / (RG_PLL_ABIST_DIV[5:0]+2)/2**
7:4	RG_PLLGP_MONSEL	**PLL group monitor control**
0	RG_PLLGP_SEL_CK	**PLL group test mode control** 0: AIO mode 1: CLK mode

1E000F28 PLLGP_CTRL_7 PLLGP control and status 7 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					RG_APLL_DDS_DMY											
Type					RW											
Reset					0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												RG_APLL_MONEN	RG_APLL_MONCKEN	RG_APLL_DDS_MONEN	RG_APLL_DIV34	RG_PLLGP_DET_EN
Type												RW	RW	RW	RW	RW
Reset												0	0	0	0	1

Bit(s)	Name	Description
27:16	RG_APLL_DDS_DMY	**APLL_DDS dummy registers**
4	RG_APLL_MONEN	
3	RG_APLL_MONCKEN	
2	RG_APLL_DDS_MONEN	
1	RG_APLL_DIV34	
0	RG_PLLGP_DET_EN	

1E000F2C PLLGP_CTRL_8 PLLGP control and status 8 00011501

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_PLLGP_RESERVE															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		RG_USBDIV														RG_USBDIV_EN
Type		RW														RW
Reset		0	0	1	0	1	0	1								1

Bit(s)	Name	Description
31:16	RG_PLLGP_RESERVE	**dummy reg**
14:8	RG_USBDIV	**Feedback divide ratio (N+1 Divider)** 7'd0: /1 7'd1: /2 7'd127: /128
0	RG_USBDIV_EN	

1E000F30 DA_CTRL_1 DA control and status 1 0000001F

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												DA_CBG_BG_EN	DA_CBG_IMR_EN	DA_CBG_LPF_EN	DA_XPTL_VREG_EN	DA_XPTL_CLKSQ_EN
Type												RW	RW	RW	RW	RW
Reset												1	1	1	1	1

Bit(s)	Name	Description
4	DA_CBG_BG_EN	**Bandgap enable** 1'b0: disable 1'b1: enable
3	DA_CBG_IMR_EN	**Current mirror enable** 1'b0: disable 1'b1: enable
2	DA_CBG_LPF_EN	**Low pass filter enable** 1'b0: disable (time constant=0.3us, settle needs 2us) 1'b1: enable (time constant=4us, settle needs 25us. BW= 40KHz)
1	DA_XPTL_VREG_EN	**Crystal PWD signal** 0: Crystal power down 1: Crystal power on
0	DA_XPTL_CLKSQ_EN	**CLOCK SQUARE EN signal** 0: Clock square power down 1: Clock square power on

3. List

Abbrev.	Description
AC	Access Category
ACK	Acknowledge/ Acknowledgement
ACL	Access Control List
ACPR	Adjacent Channel Power Ratio
AD/DA	Analog to Digital/Digital to Analog converter
ADC	Analog-to-Digital Converter
AES	Advanced Encryption Standard
AFC	Automatic Frequency Calibration
AGC	Auto Gain Control
AIFS	Arbitration Inter-Frame Space
AIFSN	Arbitration Inter-Frame Spacing Number
ALC	Automatic Level Control
A-MPDU	Aggregate MAC Protocol Data Unit
A-MSDU	Aggregation of MAC Service Data Units
AP	Access Point
ASIC	Application-Specific Integrated Circuit
ASME	American Society of Mechanical Engineers
ASYNC	Asynchronous
BA	Block Acknowledgement
BAC	Block Acknowledgement Control
BAR	Base Address Register
BBP	Baseband Processor
BGSEL	Band Gap Select
BIST	Built-In Self-Test
BSC	Basic Spacing between Centers
BJT	Bipolar Junction Transistor
BSSID	Basic Service Set Identifier
BW	Bandwidth
CAS	Column Address Strobe
CCA	Clear Channel Assessment
CCK	Complementary Code Keying
CCMP	Counter Mode with Cipher Block Chaining Message Authentication Code Protocol
CCX	Cisco Compatible Extensions
CF-END	Control Frame End
CF-ACK	Control Frame Acknowledgement
CLK	Clock
CPU	Central Processing Unit

Abbrev.	Description
CRC	Cyclic Redundancy Check
CSR	Control Status Register
CTS	Clear to Send
CW	Contention Window
CWmax	Maximum Contention Window
CWmin	Minimum Contention Window
DAC	Digital-To-Analog Converter
DCF	Distributed Coordination Function
DDONE	DMA Done
DDR	Double Data Rate
DFT	Discrete Fourier Transform
DIFS	DCF Inter-Frame Space
DMA	Direct Memory Access
DQ	DRAM Data
DQS	Data Strobe
DSCP	Differentiated Services Code Point
DSP	Digital Signal Processor
DW	DWORD
EAP	Expert Antenna Processor
ED	Energy Detection
EDCA	Enhanced Distributed Channel Access
EECS	EEPROM chip select
EEDI	EEPROM data input
EEDO	EEPROM data output
EEPROM	Electrically Erasable Programmable Read-Only Memory
eFUSE	electrical Fuse
EESK	EEPROM source clock
EIFS	Extended Inter-Frame Space
EIV	Extend Initialization Vector
EVM	Error Vector Magnitude
FDS	Frequency Domain Spreading
FEM	Front-End Module
FEQ	Frequency Equalization
FIFO	First In First Out
FSM	Finite-State Machine
GDM	GTP Director Module
GEM	GPON Encapsulation Method
GF	Green Field
GND	Ground
GP	General Purpose

Abbrev.	Description
GPO	General Purpose Output
GPON	Gigabit Passive Optical Network
GPIO	General Purpose Input/Output
GPRS	General Packet Radio Service
GTP	GPRS Tunneling Protocol
HCCA	HCF Controlled Channel Access
HCF	Hybrid Coordination Function
HT	High Throughput
HTC	High Throughput Control
I	In phase
ICV	Integrity Check Value
IFS	Inter-Frame Space
iNIC	Intelligent Network Interface Card
IV	Initialization Vector
I^2C	Inter-Integrated Circuit
I^2S	Integrated Inter-Chip Sound
I/O	Input/Output
IPI	Idle Power Indicator
IQ	In phase/Quadrature phase
JEDEC	Joint Electron Devices Engineering Council
JTAG	Joint Test Action Group
kbps	kilo (1000) bits per second
KB	Kilo (1024) Bytes
LCP	Linear Complementarity Problem
LDO	Low-Dropout Regulator
LDODIG	LDO for DIGital part output voltage
LED	Light-Emitting Diode
LTSSM	Link Training and Status State Machine
LNA	Low Noise Amplifier
LO	Local Oscillator
L-SIG	Legacy Signal Field
MAC	Medium Access Control
MCU	Microcontroller Unit
MCS	Modulation and Coding Scheme
MDC	Management Data Clock
MDIO	Management Data Input/Output
MEM	Memory
MFB	MCS Feedback
MFS	MFB Sequence
MIC	Message Integrity Code
MIMO	Multiple-Input Multiple-Output
MLD	Multicast Listener Discovery

Abbrev.	Description
MLNA	Monolithic Low Noise Amplifier
MM	Mixed Mode
MOSFET	Metal Oxide Semiconductor Field Effect Transistor
MPDU	MAC Protocol Data Units
MSB	Most Significant Bit
NAV	Network Allocation Vector
NAS	Network-Attached Server
NAT	Network Address Translation
NDP	Null Data Packet
NVM	Non-Volatile Memory
OCP	Open Core Protocol
ODT	On-die Termination
Oen	Output Enable
OFDM	Orthogonal Frequency-Division Multiplexing
OoS	Out-of-Service
OSC	Open Sound Control
PA	Power Amplifier
PAPE	Provider Authentication Policy Extension
PBC	Push Button Configuration
PBF	Packet Buffer
PCB	Printed Circuit Board
PCF	Point Coordination Function
PCM	Pulse-Code Modulation
PD	Preamble Detection
PFD	Phase-Frequency Detector
PHY	Physical Layer
PIFS	PCF Interframe Space
PLCP	Physical Layer Convergence Protocol
PLL	Phase-Locked Loop
PME	Physical Medium Entities
PMU	Power Management Unit
PN	Packet Number
PPLL	Programmable PLL
PROM	Programmable Read-Only Memory
PSDU	Physical layer Service Data Unit
PSI	Power supply Strength Indication
PSM	Power Save Mode
PTN	Packet Transport Network
QoS	Quality of Service
Q	Quadrature

Abbrev.	Description
R2P	Rbus to Pbus
RDG	Reverse Direction Grant
RAM	Random Access Memory
RC	Root Complex
RF	Radio Frequency
RGMII	Reduced Gigabit Media Independent Interface
RH	Relative Humidity
RoHS	Restriction on Hazardous Substances
ROM	Read-Only Memory
ROS	Rx Offset
RSSI	Received Signal Strength Indication (Indicator)
RTS	Request to Send
RvMII	Reverse Media Independent Interface
Rx	Receive
RXD	Received Data
RXINFO	Receive Information
RXWI	Receive Wireless Information
S	Stream
SDHC	Secure Digital High Capacity
SDIO	Secure Digital Input Output
SDRAM	Synchronous Dynamic Random Access Memory
SEC	Security
SGI	Short Guard Interval
SIFS	Short Inter-Frame Space
SoC	System-on-a-Chip
SPI	Serial Peripheral Interface
SRAM	Static Random Access Memory
SSCG	Spread Spectrum Clock Generator
STBC	Space–Time Block Code
SW	Switch Regulator

Abbrev.	Description
TA	Transmitter Address
TBTT	Target Beacon Transmission Time
TDLS	Tunnel Direct Link Setup
TKIP	Temporal Key Integrity Protocol
TOS	Tx Offset
TRSW	Tx/Rx Switch
TSF	Timing Synchronization Function
TSSI	Transmit Signal Strength Indication
Tx	Transmit
TxBF	Transmit Beamforming
TXD	Transmitted Data
TXDAC	Transmit Digital-Analog Converter
TXINFO	Transmit Information
TXOP	Opportunity to Transmit
TXWI	Tx Wireless Information
UART	Universal Asynchronous Rx/ Tx
USB	Universal Serial Bus
UTIF	Universal Test Interface
VGA	Variable Gain Amplifier
VCO	Voltage Controlled Oscillator
VIH	High Level Input Voltage
VIL	Low Level Input Voltage
VoIP	Voice over IP
VPID	Virtual Path Identifier
WCID	Wireless Client Identification
WEP	Wired Equivalent
WI	Wireless Information
WIV	Wireless Information Valid
WMM	Wi-Fi Multimedia
WPA	Wi-Fi Protected Access
WPDMA	Wireless Polarization Division Multiple Access
WS	Word Select

4. Revision History

Rev	Date	From	Description
1.0	2013/06/7	Leon Chung	Preliminary

This product is not designed for use in medical, life support applications. Do not use this product in these types of equipments or applications .This document is subject to change without notice and Ralink assumes no responsibility for any inaccuracies that nay be contained in this document. Ralink reserves the right to make change in the products to improve function, performance, reliability, and to attempt to supply the best product possible.